

WORKING PAPER

29/2006

Usability Testing von ERP-Systemen

Univ.-Prof. Dipl.-Ing. Dr. Hans Hinterhuber

A. Univ.-Prof. Mag. Dr. Kurt Promberger

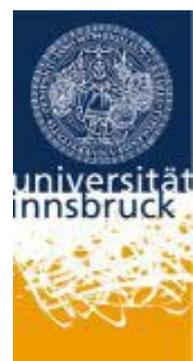
Mag. Felix Piazzolo

Unter Mitarbeit von

Katrin Kneissl

Leopold-Franzens-Universität Innsbruck
Fakultät für Betriebswirtschaft
Institut für Strategisches Management,
Marketing und Tourismus
Universitätsstr. 15
A – 6020 Innsbruck, Austria

Tel. +43-512-507-7601
Internet: www.verwaltungsmanagement.at
E-Mail: verwaltungsmanagement@uibk.ac.at



Abstract

In den vergangenen Jahren und Jahrzehnten hat eine enorme Entwicklung im Bereich der Unternehmensinformationstechnologie stattgefunden. Nicht alle Errungenschaften wurden vom Markt sofort angenommen bzw. waren im ersten Stadium praxistauglich, wie es bei „Management Information Systemen“ zum Beispiel der Fall war. Dafür gibt es eine Vielzahl von Gründen, welche technischer, kultureller, sozialer und individueller Art sein können. Oft wird behauptet, dass die Entwicklungsgeschwindigkeit der Informationstechnologie (IT) zu hoch sei. Der potenzielle Kunde, zum Beispiel eine Unternehmung inklusive seiner MitarbeiterInnen, unabhängig von den eingenommenen Positionen und Funktionen, könne solchen Entwicklungssprüngen kaum folgen. Ob dies wirklich der Fall ist, muss in Frage gestellt werden. Denn eine Weiterentwicklung im Bereich der Informationssysteme bedeutet nicht unbedingt, dass die Bedürfnisse der Adressaten hochwertiger erfüllt werden können. In der IT-Branche gibt es laufend Innovationenschübe, doch sind die resultierenden Lösungen wirklich gebrauchstauglich?

Unternehmen und Individuen agieren, sofern sie nicht durch externe Faktoren polarisiert werden, primär nutzenorientiert. Jede Unternehmung und Person steht täglich vor vielen kleineren sowie größeren Entscheidungen, die zu treffen sind. Solange die Kosten, die Funktionalität und die Software Usability nicht den Vorstellungen der Entscheider entsprechen, wird sich ein (IT-) Produkt am Markt sehr schwer tun und nachhaltig nicht erfolgreich agieren können. Je nach Lebenszyklus der einzelnen Lösungen bzw. der Problemstellungen verschiebt sich die Gewichtung der einzelnen Kriterien, sprich die Konzentration auf die Kosten, die Funktionalitäten und die Gebrauchstauglichkeit.

Unternehmen aus allen Branchen haben in den letzten Jahren erhebliche Investitionen für die Einführung von Enterprise Resource Planning (ERP)-Systemen getätigt. Sie ermöglichen das Abbilden der Geschäftsprozesse verschiedener Unternehmensbereiche wie Beschaffung, Materialwirtschaft, Vertrieb sowie Produktion, und schlussendlich die nahtlose Integration aller unternehmensspezifischen Informationen. Die Vorteile des Einsatzes von ERP-Systemen sind für Betriebe attraktiv: Substantielle Nutzensvorteile in den Bereichen Unternehmenscontrolling, Geschäftsprozessoptimierung, Kunden- und Marktorientierung und Mitarbeitereffizienz stellen sich ein. Trotz der Bedeutung, die ERP-Systemen heute zuteil wird, wurde der Aspekt der Usability in einem ERP-spezifischen Kontext bislang kaum betrachtet. Es verwundert, warum Unternehmen der Usability ihrer Informationssysteme so wenig Beachtung schenken, zumal ihre Mitarbeiter tagtäglich einen beträchtlichen Teil ihrer Arbeitszeit damit verbringen und ihre Effizienz mit der Usability der eingesetzten Software in einem sehr engen Zusammenhang steht.

Im Mittelpunkt dieser Arbeit steht neben der theoretischen Behandlung und praktischen Bedeutung des Themas Usability die Entwicklung eines für ERP-Systeme geeigneten Instrumentariums zur Messung der Gebrauchstauglichkeit aus der Sicht der Anwender. In einer Untersuchung an der Leopold-Franzens-Universität Innsbruck wurden die ERP-Produkte SAP und Semiramis in Hinblick auf deren Software Usability miteinander verglichen.

Verzeichnisstruktur

I	Inhaltsverzeichnis	4
II	Bibliografie	75
III	Tabellenverzeichnis	81
IV	Grafikverzeichnis	82
V	Anhang A: Fragebogen	83
VI	Anhang B: Fragen zum User Profil	92
VII	Anhang C: Deskriptive Statistik	94
VIII	Anhang D: Signifikanztests	97

I Inhaltsverzeichnis

1	SCHNITTSTELLE DES ERFOLGS	6
2	EINFÜHRUNG ZU ERP-SYSTEMEN	9
2.1	Merkmale von ERP-Systemen	9
2.2	Nutzen von ERP-Software	10
3	SOFTWARE USABILITY	12
3.1	Begriffsabgrenzungen	12
3.1.1	User Interface	13
3.1.2	Human Computer Interaction	15
3.1.3	Software Ergonomie	18
3.1.4	Software Usability	19
3.2	Stellenwert von Usability	23
3.2.1	Druck von Benutzerseite	23
3.2.2	User Satisfaction als Maßstab für die Akzeptanz der Software	24
3.3	Nutzen von Usability	25
3.3.1	Nutzen für das Unternehmen	26
3.3.2	Nutzen für den Softwareanbieter.....	27
3.4	User Centred Design	29
3.4.1	Prozess des User Centred Design	30
3.4.2	Prinzipien des User Centred Design	32
3.4.2.1	Allgemeine Entwurfsprinzipien	33
3.4.2.2	Hersteller Styleguides	34
3.4.2.3	Standards und Normen für HCI und Usability	35
3.5	Usability Testing	38
3.5.1	Motive für den Einsatz von Usability Testing	39
3.5.2	Ziele von Usability Testing	39
3.5.3	Methoden und Verfahren.....	40
3.5.3.1	Task Based Testing	41
3.5.3.2	Thinking Aloud	43
3.5.3.3	Benutzerbefragung.....	43
4	ENTWICKLUNG DES USABILITY TESTS FÜR ERP-SYSTEME (UTE)	45
4.1	Anforderungen	45
4.1.1	Anforderungen an den Fragebogen	45
4.1.2	Anforderungen an die Ergebnisqualität.....	46
4.2	Fragebogenentwicklung	47
4.2.1	Analyse bestehender Fragebögen	47
4.2.2	Entwicklung eines Fragebogens für die Evaluation von ERP-Software.....	50
4.3	Datenerhebung	51

5	PRAXISANWENDUNG DES UTE	52
5.1	Vorgehen	52
5.2	Systemumfeld	53
5.3	Testergebnisse	54
5.3.1	Emotion	55
5.3.2	Softwaresteuerung	57
5.3.3	Effizienz	58
5.3.4	Systemunterstützung.....	61
5.3.5	Erlernbarkeit	64
5.4	Endergebnisse und Rückschlüsse	66
6	ZUKUNFTSAUSSICHTEN	74

1 Schnittstelle des Erfolgs

Im letzten Jahrzehnt haben Unternehmen aus allen Branchen weltweit erhebliche Investitionen für die Einführung von Enterprise Resource Planning (ERP)-Systemen getätigt.¹ Diese Softwaresysteme ermöglichen das Abbilden der Geschäftsprozesse der verschiedenen Unternehmensbereiche wie Beschaffung, Materialwirtschaft, Vertrieb oder Produktion, und schlussendlich die nahtlose Integration aller unternehmensspezifischen Informationen.²

Die Vorteile des Einsatzes von ERP-Systemen sind für Betriebe attraktiv: Neben optimierter Lagerhaltung und verkürzten Durchlaufzeiten stellen sich substantielle Mehrnutzen in den Bereichen Unternehmenscontrolling, Geschäftsprozessoptimierung, Kunden- und Marktorientierung und Mitarbeitereffizienz ein.³ Das erklärt auch den sich in den letzten Jahren rasant entwickelnden Markt für ERP-Softwaresysteme, der nach wie vor große Wachstumspotentiale birgt. Analysten erwarten bis 2009 einen jährlichen Umsatzzuwachs von sieben Prozent.⁴

Ob die erwarteten Vorteile von ERP-Systemen auch tatsächlich eintreten, hängt vor allem von der effektiven Nutzung der Informationssysteme und von der Zufriedenheit der Benutzer ab. *Somers, Nelson* und *Karimi* belegen, dass User Satisfaction eine der wichtigsten Determinanten für den Erfolg eines ERP-Systems darstellt.⁵

Ein integriertes ERP-System erfordert einen relativ hohen Lernaufwand. Probleme in der einfachen Bedienbarkeit sind keine Seltenheit. *Somers, Nelson* und *Karimi* bekräftigen die Notwendigkeit einer Reduktion der Komplexität der Systeme und einer einfacheren Adaptierbarkeit der User Interfaces.⁶ Ebenso weiß man über die Relevanz der Softwaredokumentation für die Benutzerzufriedenheit mit dem Softwareprodukt, welche sich positiv auf die effektive Nutzung des Informationssystems auswirkt, Bescheid.⁷

Trotz der Bedeutung, die ERP-Systemen heute zuteil wird, wurde der Aspekt der Usability in einem ERP-spezifischen Kontext bislang kaum betrachtet. Es verwundert, warum Unternehmen der Usability ihrer Informationssysteme so wenig Beachtung schenken, zumal ihre Mitarbeiter tagtäglich einen beträchtlichen Teil ihrer Arbeitszeit damit verbringen⁸ und ihre Effizienz mit der Usability der eingesetzten Software in einem engen Zusammenhang steht:

¹ Vgl. *Somers/Nelson/Karimi* (2003): S. 595.

² Vgl. *Martin/Lempp/Mauterer* (2003): S. 75; *Somers/Nelson/Karimi* (2003): S. 596.

³ Vgl. *Martin* (2003): S. 66.

⁴ Vgl. *Reilly, Kevin* (2005): AMR Research Releases Report Showing Overall European Market for ERP Vendors to Grow 7% Annually Through 2009. Online im Internet. URL: <<http://www.amrresearch.com/Content/View.asp?pmillid=18386>>, Download am 02.02.2006.

⁵ Vgl. *Somers/Nelson/Karimi* (2003): S. 595.

⁶ Vgl. ebenda: S. 613.

⁷ Vgl. ebenda; *Mirani/King* (1994): S. 141.

⁸ Vgl. *Nielsen, Jakob/Pernice Coyne, Kara* (2001): A Useful Investment. Online im Internet. URL: <http://www.cio.com/archive/021501/et_pundits.html>, Download am 22.01.2006.

„Usability is not a luxury but a basic ingredient in software systems: People's productivity and comfort relate directly to the usability of the software they use.“⁹

Die Nutzen, welche Software Usability für ein Unternehmen generiert, sind vielschichtig:

- Usability reduziert Schulungs- und Trainingskosten.
- Usability senkt die Fehlerraten.
- Usability steigert die Produktivität.
- Usability reduziert die Ausgaben für den Support.
- Usability senkt die Wartungskosten.
- Usability steigert die Zufriedenheit der Benutzer.
- Usability erhöht die Arbeitszufriedenheit der Mitarbeiter.

Kein Softwareanbieter kann es sich heute erlauben, ohne die Beachtung grundlegender softwareergonomischer Kriterien, Anwenderprogramme zu erstellen.¹⁰ Gleichzeitig resultieren noch immer 60 Prozent der Softwaremängel aus Schwächen der Software Usability, während nur 15 Prozent von Funktionalitätsproblemen herrühren.¹¹ Nielsen kommt zum Schluss, dass mindestens 10 Prozent des Budgets für eine Softwareentwicklung für Usability ausgegeben werden sollten.¹²

Usability bringt aber nicht nur Nutzen für die Unternehmen, welche die Software einsetzen. Die Vorteile eines Usability-Fokus sind auch für Softwareanbieter enorm, denkt man zum Beispiel an Einsparungsmöglichkeiten im Zusammenhang mit dem Personalbedarf in einem Support Center. Der Anruf eines Kunden kostet dem Softwarehersteller rund 30 bis 100 USD, abhängig von der Komplexität der Software. Mehr als die Hälfte der Anrufe in Support Center sind die Folge mangelnder Usability.¹³

Die Usability von ERP-Systemen, und von Softwareprodukten im Allgemeinen, wird in den nächsten Jahren wesentlich an Bedeutung gewinnen. In einer Welt des Ubiquitous Computing¹⁴ verwenden immer mehr Anwender eine immer größere Fülle an Programmen. Bereits 1995 nutzten zwei Drittel der Windows-Benutzer fünf Programme gleichzeitig und viele verwendeten zehn oder mehr Programme am Tag.¹⁵

Die Notwendigkeit für Softwarehersteller, sich am Benutzer zu orientieren und User Centred Design konsequent einzusetzen, kann nicht geleugnet werden. Oder mit den Worten von Zemanek, Computerpionier und Gründer der Österreichischen Computer Gesellschaft, gesprochen:

⁹ Juristo/Windl/Constantine (2001): S. 20.

¹⁰ Vgl. Holzinger (2001b): S. 15.

¹¹ Vgl. Bevan (1999): S. 94.

¹² Vgl. Nielsen, Jakob (2003): Return on Investment for Usability. Online im Internet. URL: <<http://www.useit.com/alertbox/20030107.html>>, Upload am 07.01.2003, Download am 03.02.2006.

¹³ Vgl. Nielsen (2000): S. 10.

¹⁴ Mit Ubiquitous Computing wird die Allgegenwärtigkeit von Informationssystemen bezeichnet.

¹⁵ Vgl. Dray (1995): S. 17.

„Wir Informatiker hatten es leicht während der letzten fünfundzwanzig Jahre - wir konzentrierten uns auf die Maschinen. Aber innerhalb der nächsten Jahre werden die Informatiker sich mehr auf den Menschen, der mit dieser Maschine arbeitet, konzentrieren müssen.“¹⁶

Durch „Usability lacks“ ausgelöste Ineffizienzen sind zu vermeiden und, wie in diesem Working Paper thematisiert, auch vermeidbar. Die Schnittstelle Mensch-Computer bleibt in den Unternehmensbereichen, in welchen keine Vollautomatisierung möglich bzw. erwünscht ist, ein Schlüsselkriterium für den nachhaltigen Erfolg einer ERP-Software am Markt.

Nach einer kurzen Einführung zur Thematik ERP werden die zwei Kernbereiche dieser Arbeit behandelt. Im Fokus steht einerseits die wissenschaftliche theoretische Abhandlung des Themas Software Usability und ihrer einzelnen Teildisziplinen sowie andererseits die Entwicklung und Durchführung eines für ERP-Systeme zugeschnittenen Usability Tests, welcher seit Ende 2004 an der Leopold-Franzens-Universität Innsbruck eingesetzt wird. Es wurden die in der Hochschullehre intensiv eingesetzten ERP-Systeme SAP R/3 und Semiramis als Untersuchungsobjekte ausgewählt. Dem Ausbildungsbereich der Universität standen dabei folgende Systeme zur Verfügung: Einerseits eine SAP R/3 R4.6C Installation in Verbindung mit dem User Interface (UI) SAP GUI 6.40 bzw. 6.20 und andererseits eine Semiramis 4.1 bzw. Semiramis 2 Installation, welche als Front End den MS Internet Explorer 6.0 verwendet. Im Wintersemester 2005 wurde jeweils das neuere Produkt eingesetzt.

¹⁶ Zemanek nach Holzinger/Weidmann/Thurnher (2005): S. 28.

2 Einführung zu ERP-Systemen

Enterprise Resource Planning Systeme, kurz ERP-Systeme genannt, bezeichnen eine integrierte betriebswirtschaftlich-administrative Standardanwendungssoftware.¹⁷ Sie steuern und kontrollieren den gesamten betrieblichen Ablauf in den verschiedenen Unternehmensbereichen.¹⁸ Front und Back Office Systeme werden in einem zentralen System integriert. Idealerweise werden alle Geschäftsprozesse im ERP-System abgebildet und gesteuert. Redundanzen werden bestmöglich vermieden. Die Integration der verschiedenen Unternehmens- und Funktionsbereiche eröffnet dem Management vielfältige Nutzenpotenziale. Trotzdem ist bei IT-Investitionen grundsätzlich auf die Kosten-Nutzen-Relation zu achten.

Der ERP-Markt ist seit 2 Jahren wieder spürbar in Fahrt gekommen. Rund 9 Milliarden USD investieren Unternehmen jährlich in ERP-Systeme.¹⁹ *Davenport* gibt an, dass zusätzlich ein Betrag in ähnlicher Höhe, allein für ERP-spezifisches Consulting aufgewendet wird.²⁰

2.1 Merkmale von ERP-Systemen

Betriebswirtschaftlich orientierte Software muss folgende Aspekte erfüllen, um als ERP-System klassifiziert zu werden:

- **Integration:** ERP-Systeme versuchen, das ganze Unternehmen mit seiner Aufbau- und Ablauforganisation IT-mäßig abzubilden (siehe Abbildung 1).²¹ Die administrativen Funktionen eines Unternehmens, nämlich Finanz- und Rechnungswesen, Beschaffung, Produktion, Lagerhaltung und Vertrieb, werden in ERP-Systemen weitgehend abgedeckt. Aufgrund der Tatsache, dass es nur eine Datenbasis für alle Anwendungsbereiche gibt, wird redundante Datenhaltung abgewendet und Datenkonsistenz garantiert. Durch die Integration der Unternehmensdaten werden auch die Aktivitäten des strategischen und operativen Managements unterstützt.²² Betriebswirtschaftliche Standardsoftware orientiert sich an durchgängigen Geschäftsprozessen. Deshalb leistet der Einsatz von ERP-Software einen bedeutenden Beitrag für die Steuerung und Optimierung von Geschäftsprozessen.²³

¹⁷ Vgl. Promberger/Schlager-Weidinger/Traxl (2003): S. 20; von Arb, Reto (1997): Vorgehensweise und Erfahrungen bei der Einführung von Enterprise-Management-Systemen dargestellt am Beispiel von SAP R/3. Online im Internet. URL: <<http://www.digital-publications.ch/vonarb/vonarb.zip>>, Download am 12.02.2006.

¹⁸ Vgl. Promberger/Schlager-Weidinger/Traxl (2003): S. 39.

¹⁹ Vgl. Reilly, Kevin (2005): AMR Research Releases Report Showing Overall European Market for ERP Vendors to Grow 7% Annually Through 2009. Online im Internet. URL: <<http://www.amrresearch.com/Content/View.asp?pmillid=18386>>, Download am 02.02.2006.

²⁰ Vgl. Davenport (1998): S. 122.

²¹ Vgl. Färinger, Peter (2000): Enterprise Resource Planning. Online im Internet. URL: <<http://ebusiness.or.at/erp.htm>>, Download am 12.02.2006.

²² Vgl. von Arb, Reto (1997): Vorgehensweise und Erfahrungen bei der Einführung von Enterprise-Management-Systemen dargestellt am Beispiel von SAP R/3. Online im Internet. URL: <<http://www.digital-publications.ch/vonarb/vonarb.zip>>, Download am 12.02.2006.

²³ Vgl. Färinger, Peter (2000): Enterprise Resource Planning. Online im Internet. URL: <<http://ebusiness.or.at/erp.htm>>, Download am 12.02.2006.

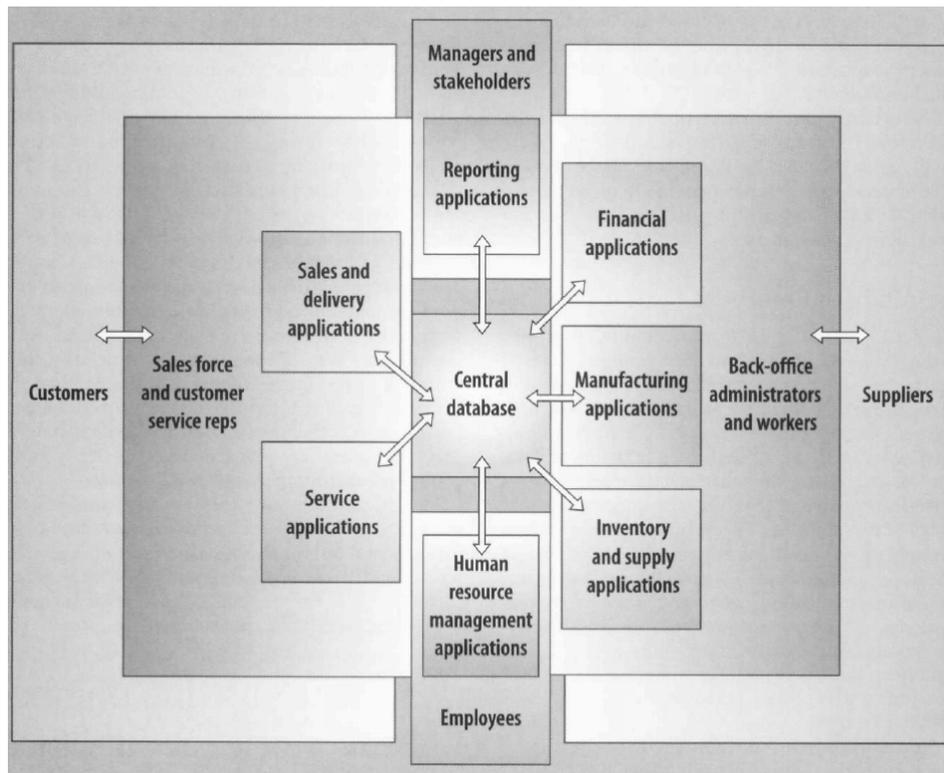


Abbildung 1: Integration der verschiedenen Unternehmensbereiche in einem ERP-System²⁴

- **Flexibilität:** Betriebswirtschaftliche Standardsoftware kennzeichnet sich durch einen gewissen Grad an technischer und betriebswirtschaftlicher Anpassungsfähigkeit. So lassen sich ERP-Systeme meist auf mehreren Hardwareplattformen und Betriebssystemen einsetzen. Die Orientierung an Client/Server-Architekturen und offenen Standards schafft weitgehende Skalierungsmöglichkeiten. Eine betriebswirtschaftliche Anpassung (Customizing) kann entweder durch Parameterersetzung oder durch Eingriffe an definierten Schnittstellen erfolgen.²⁵
- **Internationalität:** ERP-Software unterstützt mehrere Sprachen und die Anwendung verschiedener länderspezifischer Gesetzgebungen und erfüllt somit zentrale Anforderungen von international tätigen Unternehmen.²⁶

Die prozessorientierte Sichtweise von ERP-Systemen ermöglicht ein durchgängiges Workflow-Management, wobei funktionale Gegebenheiten berücksichtigt werden können und müssen.

2.2 Nutzen von ERP-Software

Die Funktion von Enterprise Resource Planning in einem Unternehmen kann als die eines Problemlösers verstanden werden: Der Einsatz einer ERP-Software beendet die Fragmentierung von

²⁴ Grafik entnommen aus Davenport (1998): S. 124.

²⁵ Vgl. von Arb, Reto (1997): Vorgehensweise und Erfahrungen bei der Einführung von Enterprise-Management-Systemen dargestellt am Beispiel von SAP R/3. Online im Internet. URL: <<http://www.digital-publications.ch/vonarb/vonarb.zip>>, Download am 12.02.2006.

²⁶ Vgl. ebenda.

Informationen in Unternehmen. Daraus resultiert auch der Nutzen des Einsatzes eines ERP-Systems.²⁷

Das Aufrechterhalten vieler verschiedener Systeme in einem Unternehmen verursacht hohe Kosten: Die redundante Datenhaltung und der notwendige Datenaustausch von einem System ins andere bzw. das Programmieren von Schnittstellen zwischen zwei Systemen für automatisierte Datentransfers beanspruchen betriebliche Ressourcen in einem nicht unbeträchtlichen Ausmaß. Ist das Vertriebssystem eines Unternehmens nicht an das Buchhaltungssystem geknüpft, können die Entscheidungen des Managements nicht auf einem Verständnis der Profitabilität von einzelnen Produkten bzw. Kunden gründen.²⁸ Wie *Davenport* meint: „If a company's systems are fragmented, its business is fragmented.“²⁹

Zu den Vorteilen von Enterprise Resource Planning (ERP) für Unternehmen zählen zum einen direkt monetär messbare Nutzeneffekte, wie Bestandssenkung, Durchlaufzeitverkürzung oder eine Reduktion an Wartungskosten. Zum anderen lassen sich qualitativ messbare Nutzenaspekte erzielen. Eine Studie mit 260 Unternehmen, die ERP-Systeme eingeführt haben, zeigt, dass ein Großteil des Nutzens qualitativ gemessen werden kann. Die Unternehmen konnten einen substantiellen Mehrnutzen gegenüber der Situation vor der ERP-Einführung erzielen. Die Vorteile für die Unternehmen stellten sich vor allem in den Bereichen Unternehmenscontrolling, Geschäftsprozessoptimierung, Kunden- und Marktorientierung sowie Mitarbeitereffizienz ein.³⁰

Geschäftsprozesse durchschneiden horizontal die einzelnen Funktionsbereiche einer Unternehmung. ERP-Systeme bewirken eine hohe Prozessorientierung und können somit durch ihre Implementierung zu einer radikalen funktions- und hierarchieübergreifenden Neugestaltung der Unternehmung führen. Sie kommt einer Neuorientierung, in vielen Fällen sogar einer Neuschöpfung der Unternehmung gleich und ist auf die Erreichung dreier Ziele angelegt:³¹

1. Bessere und schnellere Zufriedenstellung der Stakeholder, insbesondere der internen und externen Kunden.
2. Entwicklung und Absicherung von Kernkompetenzen und
3. Erhöhung der Produktivität.

Durch die laufende Verbesserung der Geschäftsprozesse lassen sich Produktivitätsfortschritte erzielen, die für Unternehmen notwendige Handlungsspielräume schaffen können.

²⁷ Vgl. Davenport (1998): S. 123.

²⁸ Vgl. ebenda.

²⁹ Ebenda.

³⁰ Vgl. Martin (2003): S. 66.

³¹ Vgl. Hinterhuber (2004): S. 49.

3 Software Usability

Bei der Entwicklung von Softwaresystemen rückte die Qualität der Interaktion zwischen Mensch und Computer zunehmend in den Fokus der Softwareanbieter. Das lässt sich alleine an der Tatsache messen, dass der Anteil des User Interface (Benutzerschnittstelle) an der Software ständig angewachsen ist. Dies bezieht sich sowohl auf den Kodierungsumfang³² als auch auf den zeitlichen Aufwand bei der Entwicklung der Software. Die Qualität der Benutzerschnittstelle ist in einem kompetitiven Markt für die Auswahl von Softwaresystemen relevant.³³ Die Akzeptanz einer Software bei den tatsächlichen wie potentiellen Anwendern bestimmt sich neben ihrem Funktionsumfang immer mehr durch ihre Nutzungsqualität. Eine hohe Qualität des User Interface ist demzufolge neben der Aufgabenangemessenheit³⁴ und einem stabilen Betrieb eine entscheidende Determinante für den Erfolg einer Software.³⁵ Den Anwendern ist es kaum möglich zwischen Funktionalität und User Interface zu unterscheiden. Nehmen die Anwender das User Interface als gebrauchstauglich wahr, so werden sie auch das gesamte Softwaresystem als gebrauchstauglich einschätzen.³⁶ Software Usability, oder Gebrauchstauglichkeit, wie es in der deutschen Übersetzung genannt wird, dient als Maßstab zur Bestimmung der Nutzungsqualität einer Software.³⁷

Einführend finden sich in diesem Kapitel zunächst terminologische Abgrenzungen. Nach der Darstellung des Stellenwerts von Software Usability wird der Nutzen von Usability, einerseits für die Unternehmen, welche ERP-Systeme mit hoher Usability einsetzen, andererseits für die Anbieter dieser Produkte, diskutiert. Schließlich wird der User Centred Design Prozess thematisiert. Im Anschluss daran wird auf die Prinzipien und Methoden des User Centred Design eingegangen. User Centred Design zielt darauf, Systeme von möglichst hoher ergonomischer Güte zu entwickeln. Zentraler Bestandteil des User Centred Design ist das Usability Testing, das Methoden und Verfahren zur Überprüfung von Software Usability zur Verfügung stellt.

3.1 Begriffsabgrenzungen

Das User Interface stellt für die Benutzer einer Software den sichtbaren Teil des Systems dar; was dahinter auch an technischer Perfektion liegen mag, die überwiegende Mehrheit der Endbenutzer hat davon keine Ahnung. Deshalb ist die Bedeutung des User Interface für die vom Benutzer erfahrene Nutzungsqualität, die Software Usability, einzigartig.³⁸

³² Vgl. ACM SIGCHI (1996): Curricula for Human Computer Interaction. Chapter 2: Human-Computer Interaction. Online im Internet. URL: <<http://www.acm.org/sigchi/cdg/cdg2.html>>, Download am 19.01.2006; Usability Forum: HCI-Geschichte. Online im Internet. URL: <<http://www.usability-forum.com/bereiche/einleitung/geschichte.shtml>>, Download am 27.01.2006.

³³ Vgl. Holzinger (2001b): S. 16.

³⁴ Die Aufgabenangemessenheit ist gegeben, sofern die Software die Erledigung einer Aufgabe des Anwenders unterstützt. Vgl. Rohr (1988): S. 27.

³⁵ Vgl. Preim (1999): S. 1f; ACM SIGCHI (1996): Curricula for Human Computer Interaction. Chapter 2: Human-Computer Interaction. Online im Internet. URL: <<http://www.acm.org/sigchi/cdg/cdg2.html>>, Download am 19.01.2006.

³⁶ Vgl. Dray (1995): S. 17.

³⁷ Vgl. Nielsen nach Eichinger, Armin: Usability - Vorbemerkungen. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-vorbemerkungen.html>>, Download am 18.01.2006.

³⁸ Vgl. Holzinger, Andreas (2006): Usability - Erfolgsfaktor für Unternehmen. Interdisziplinäre Erkenntnisse aus Psychologie und Informatik bestimmen die Zukunft. Online im Internet. URL: <[http://www.competence-site.de/wissensmanagement.nsf/%208A644AB5EEF5063FC12570EC00351C51/\\$File/g47_holzinger_competencesite01.pdf](http://www.competence-site.de/wissensmanagement.nsf/%208A644AB5EEF5063FC12570EC00351C51/$File/g47_holzinger_competencesite01.pdf)>, Download am 01.02.2006.

Einleitend werden die wesentlichen Charakteristika des User Interface umrissen. Daran anknüpfend wird ein Einblick in jenes Teilgebiet der Informatik gegeben, welches sich mit der Kommunikation zwischen Mensch und Maschine befasst, die Human Computer Interaction (HCI, Mensch-Computer-Interaktion). Es stellt die Grundlagen für die Forschung zur Usability von Softwareprodukten bereit. Das mit der HCI verwandte Fach der Ergonomie wird anschließend vorgestellt, wobei sich die Erläuterungen dazu auf die Ergonomie von Softwareprodukten beschränken. Abschließend wird jener Begriff definiert, auf dessen Thematik sich diese Arbeit im Kern bezieht, Software Usability.

3.1.1 User Interface

Der Anteil des User Interface an der Entwicklung eines Softwaresystems, was Dauer und Kosten betrifft, ist ständig im Wachsen begriffen. Daran lässt sich die Bedeutung erkennen, welche der Benutzerschnittstelle heute beigemessen wird.³⁹

- Der Anteil des User Interface am Sourcecode beträgt typischerweise 50 bis 60 Prozent.⁴⁰
- Mindestens 29 Prozent des Softwareentwicklungsbudgets sind für das Design des User Interface reserviert.
- Der zeitliche Aufwand für die Gestaltung des User Interface in Relation zum gesamten Softwareentwicklungsprojekt beträgt 40 Prozent.

Das User Interface bildet die zentrale Schnittstelle zwischen Mensch und Computer.⁴¹ Dabei geht es nicht ausschließlich darum, wie sich die visuelle Oberfläche des Softwaresystems gestaltet. Es umfasst alle Aspekte des Systemdesigns, welche die Interaktion zwischen Anwender und System beeinflussen, beispielsweise die Aufgabenangemessenheit des Systems, die Steuerung, die Navigation im System und die Integration verschiedener Anwendungen.⁴²

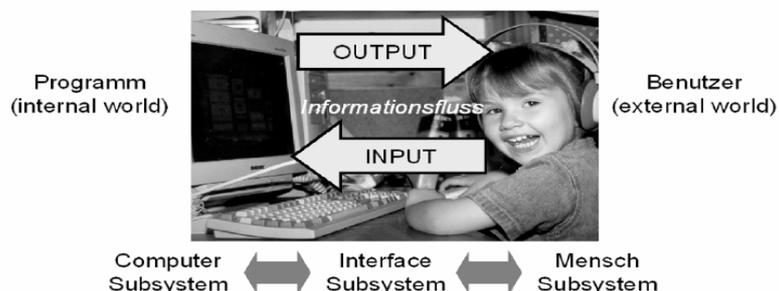


Abbildung 2: User Interface⁴³

³⁹ Vgl. Dray (1995): S. 17.

⁴⁰ Vgl. ebenda; ACM SIGCHI (1996): Curricula for Human Computer Interaction. Chapter 2: Human-Computer Interaction. Online im Internet. URL: <<http://www.acm.org/sigchi/cdg/cdg2.html>>, Download am 19.01.2006; Usability Forum: HCI-Geschichte. Online im Internet. URL: <<http://www.usability-forum.com/bereiche/einleitung/geschichte.shtml>>, Download am 27.01.2006.

⁴¹ Vgl. Holzinger (2001a): S. 87.

⁴² Vgl. Dray (1995): S. 17.

⁴³ Grafik übernommen aus Holzinger (2001a): S. 87.

Die Benutzeroberfläche integriert funktionale und ästhetische Elemente:

Tabelle 1: Elemente der Benutzeroberfläche⁴⁴

	Beschreibung des Elements	Regel für das Element
Orientierungselement	Dient dem leichteren Zurechtfinden auf der Oberfläche. Beispiele: Übersichten, Gliederungen, Aufzählungszeichen, Hervorhebungen, Farbbereiche	Benutzer muss erkennen können, wo er sich befindet und aus welchen Bereichen die Software besteht.
Navigationselement	Ermöglicht dem Benutzer, sich auf dem User Interface zu bewegen und gezielt bestimmte Bereiche anzuspringen. Beispiel: Navigationsleiste	Die Navigation soll konsistent, logisch, übersichtlich und rasch sein.
Inhaltselement	Präsentiert Informationen, welche dem Benutzer vermittelt werden sollen. Beispiele: Texte, Bilder, Animationen, Töne, Videos	Die Inhaltselemente müssen für das System aufbereitet sein. Text sollte etwa kurz und prägnant sein.
Interaktionselement	Befähigt den Benutzer, bestimmte Aktionen zu erledigen. Beispiele: Auswahlménüs, Buttons	Ein Interaktionselement muss den intuitiven Erwartungen des Benutzers entsprechen.
Motivationselement	Wirkt fördernd für die Benutzerakzeptanz und die Freude am Benutzen der Software. Beispiele: Bilder, Animationen	Ein Motivationselement soll die Neugier der Benutzer steigern.

Es existieren verschiedene Modelle, um User Interfaces softwaretechnisch zu realisieren:⁴⁵

- **Schichtenmodelle** unterteilen das System in „Schichten“. Das älteste und bekannteste dieser Modelle ist das Seeheim-Modell, welches das User Interface in drei Schichten aufteilt.



Abbildung 3: Das Seeheim-Modell für User Interfaces⁴⁶

- Ein- und Ausgabegeräte stehen in den **Gerätemodellen** im Vordergrund.
- **Objektorientierte Modelle** begreifen das User Interface als ein System von Interaktionsobjekten.

⁴⁴ Eigene Darstellung, Inhalte übernommen aus Holzinger, Andreas (2006): Usability - Erfolgsfaktor für Unternehmen. Interdisziplinäre Erkenntnisse aus Psychologie und Informatik bestimmen die Zukunft. Online im Internet. URL: <[http://www.competence-site.de/wissensmanagement.nsf/%20A644AB5EEF5063FC12570EC00351C51/\\$File/q47_holzinger_competencesite01.pdf](http://www.competence-site.de/wissensmanagement.nsf/%20A644AB5EEF5063FC12570EC00351C51/$File/q47_holzinger_competencesite01.pdf)>, Download am 01.02.2006.

⁴⁵ Vgl. Preim (1999): S. 276ff.

⁴⁶ Eigene Darstellung, teilweise übernommen aus Preim (1999): S. 277.

Diese Modelle bilden die Grundlage für User Interface Management Systeme (UIMS), Konstruktionswerkzeuge zur Unterstützung der Entwicklung von Benutzerschnittstellen.⁴⁷ Mit Hilfe von UIMS können einerseits das Layout, andererseits die Handhabung, also sowohl statische wie auch dynamische Komponenten von Softwaresystemen, spezifiziert werden.⁴⁸

Erst als Graphical User Interfaces (GUIs) kostengünstig hergestellt werden konnten, nahm ihre Bedeutung für Softwaresysteme zu. Mittlerweile ist ein gebrauchstaugliches User Interface unerlässlich.⁴⁹ GUIs „bezeichnen eine Generation Benutzerschnittstellen, die sich aus der Idee der Visualisierung von Daten- und Kontrollmanipulation entwickelt hat.“⁵⁰ Auch WIMP (Windows, Icon, Menue, Pointing Device) -Schnittstellen genannt, bestehen sie aus einer visuellen Anzeigeeinheit, Interaktionsformen - basierend auf Fenstern, Piktogrammen und Menus - und einem Zeigehilfsmittel. Zur visuellen Anzeigeeinheit zählen ein Bildschirm zur Anzeige von Informationen sowie eine Tastatur zur Eingabe von Kontroll- und Manipulationsfunktionen sowie Daten. Die Vorteile von GUIs sind nachstehend aufgelistet:⁵¹

- **Hochauflösende Bildschirme** ermöglichen die Präsentation und Manipulation von grafischer Information.
- **Fenstersysteme** erlauben die Kontrolle mehrerer, einander überlappender Bildschirmausschnitte. Die Benutzer können sich somit in unterschiedlichem Kontext mit geringem Interaktionsaufwand bewegen.
- **Piktogramme** stellen Daten-, Manipulations- und Kontrolleingaben durch eine vereinfachte grafische Darstellung dar.
- **Menus** ermöglichen dem Anwender die Auswahl von Eingaben aus einer Liste von Befehlen.
- **Zeigegeräte**, beispielsweise die Maus, dienen dem Auffinden, Selektieren und der Manipulation von Information.

Das User Interface bildet den grundlegenden Gestaltungsgegenstand in den Disziplinen der Human Computer Interaction und der Software Ergonomie⁵², auf welche anschließend näher eingegangen wird.

3.1.2 Human Computer Interaction

Die Forschungsdisziplin der Human Computer Interaction (HCI), im deutschen auch Mensch-Computer-Interaktion genannt, widmet sich dem Zusammenspiel zwischen und der Gestaltung der Interaktionsformen von Mensch und Computer. Zentrales Element einer Interaktion ist, dass der Anwender mit dem Computer interagiert, mit dem Zweck eine bestimmte Aufgabe zu erledigen.⁵³

⁴⁷ Vgl. Preim (1999): S. 534; Hoppe (1988): S. 280.

⁴⁸ Vgl. Stary (1994): S. 291.

⁴⁹ Vgl. Holzinger (2001a): S. 79; Dray (1995): S. 17.

⁵⁰ Stary (1994): S. 132.

⁵¹ Vgl. ebenda: S. 132f.

⁵² Vgl. Maaß (1995): S. 199.

⁵³ Vgl. Dix et al. (1993): S. 3; Geiser (1990): S. 9.

Es handelt sich dabei um ein relativ junges Fachgebiet, das erst ab 1980 mit der Verbreitung von grafischen Benutzeroberflächen (Xerox Star 1983, Apple Lisa 1985) entstand.⁵⁴ Die Interessensgruppe *Special Interest Group on Computer Human Interaction* der *Association for Computing Machinery* (ACM SIGCHI) definiert HCI folgendermaßen:

„Human computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them.“⁵⁵

HCI als interdisziplinäres Fachgebiet verknüpft Erkenntnisse aus den Bereichen der Informatik, im Besonderen die Gestaltung von Anwendungen und User Interfaces, der Psychologie, speziell aus der Kognitionswissenschaft und der Analyse des Anwenderverhaltens, der Soziologie und der Anthropologie sowie des Industrial Design von Softwaresystemen. HCI vereint somit wissenschaftliche, technische und gestalterische Aspekte. Die im Folgenden aufgelisteten Themen sollen veranschaulichen, mit welchen Fragestellungen sich die Disziplin HCI auseinandersetzt:⁵⁶

- Leistungsfähigkeit von Menschen und Computer
- Struktur der Kommunikation zwischen Menschen und Computer
- Menschliche Fähigkeiten zur Benutzung von Maschinen, dazu zählt auch die Erlernbarkeit von User Interfaces
- Algorithmen für und Programmierung von User Interfaces
- Aspekte des Software Engineering in Bezug auf die Gestaltung und die Implementierung von User Interfaces
- Prozesse der Spezifikation, des Designs und der Implementierung von User Interfaces
- Gestalterische Gesichtspunkte

Die Interaktion zwischen Mensch und Computer wird in Abbildung 3 dargestellt. Folgende Anmerkungen sollen zu einem besseren Verständnis beitragen:

- **Use and Context:** Die Anwendungen (Application Areas), welche mit Hilfe der Software ausgeführt werden, sind für das User Interface Design von entscheidender Bedeutung. Daneben können auch der soziale und organisationale Kontext (Social Organization and Work), beispielsweise das Image des Unternehmens, Einfluss auf das User Interface haben. Ziel der Gestaltung des User Interface ist die Entwicklung einer passenden Schnittstelle; das schließt allenfalls auch notwendige Anpassungen (Human Machine Fit and Adaptations) mit ein.
- **Human Characteristics:** Die Grundlagen menschlicher Informationsverarbeitung (Human Information Processing) sowie menschlicher Kommunikation und Interaktion (Language,

⁵⁴ Vgl. Holzinger, Andreas (2006): Usability - Erfolgsfaktor für Unternehmen. Interdisziplinäre Erkenntnisse aus Psychologie und Informatik bestimmen die Zukunft. Online im Internet. URL: <[http://www.competence-site.de/wissensmanagement.nsf/%208A644AB5EEF5063FC12570EC00351C51/\\$File/g47_holzinger_competencesite01.pdf](http://www.competence-site.de/wissensmanagement.nsf/%208A644AB5EEF5063FC12570EC00351C51/$File/g47_holzinger_competencesite01.pdf)>, Download am 01.02.2006.

⁵⁵ ACM SIGCHI (1996): Curricula for Human Computer Interaction. Chapter 2: Human-Computer Interaction. Online im Internet. URL: <<http://www.acm.org/sigchi/cdg/cdg2.html>>, Download am 19.01.2006.

⁵⁶ Vgl. ebenda.

Communication and Interaction), aber auch das Wissen um anthropometrische und physiologische Charakteristika (Ergonomics) sind für die Disziplin der HCI von entscheidender Bedeutung.

- **Computer System and Interface Architecture:** Computer sind mit speziellen Komponenten ausgestattet, um die Interaktion zu ermöglichen. Dazu zählen Ein- und Ausgabegeräte (Input and Output Devices), Dialogtechniken (Dialogue Techniques), Dialoggattung (Dialogue Genre), grafische Datenverarbeitung (Computer Graphics) und Dialogarchitektur (Dialogue Architecture).
- **Development Process:** Die Gestaltung von User Interfaces beinhaltet sowohl gestalterische (Design Approaches) wie auch technische Aspekte (Implementation Techniques and Tools). Eine Evaluation der Benutzerschnittstelle (Evaluation Techniques) kann Verbesserungspotentiale aufdecken. Ebenso nützlich für die Gestaltung von User Interfaces erweisen sich Beispielsysteme und Fallstudien (Example Systems and Case Studies).

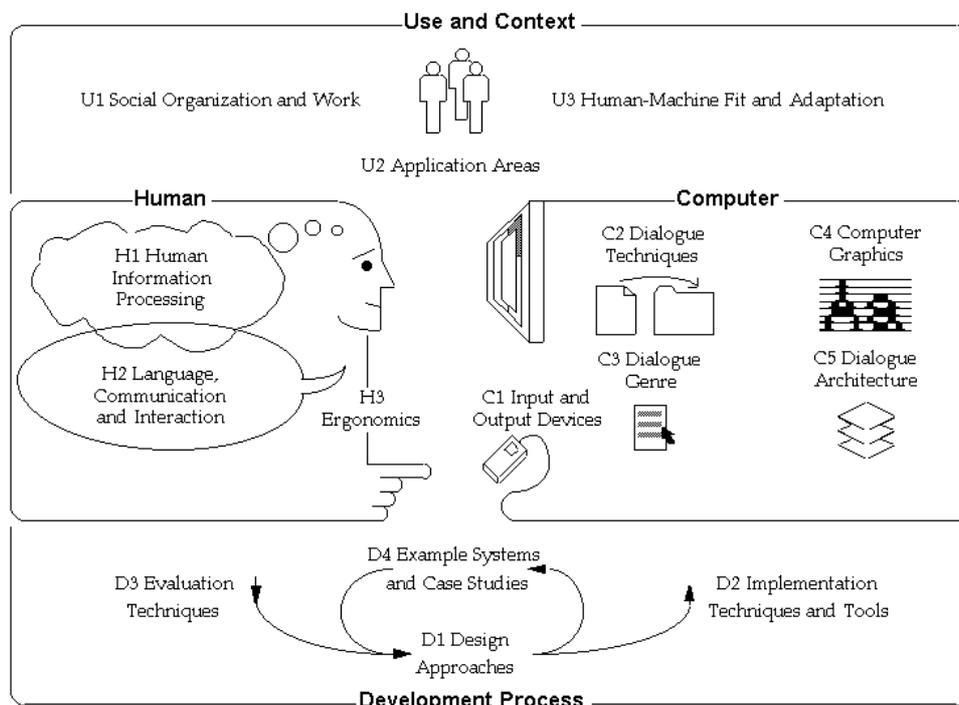


Abbildung 4: Human Computer Interaction⁵⁷

Der Unterschied zwischen HCI und Software Ergonomie, jene Wissenschaft, welche im nächsten Abschnitt behandelt wird, besteht darin, dass die Forschungsrichtung der HCI ein größeres Themengebiet als die Software Ergonomie umfasst. So werden auch hardwareergonomische Aspekte, beispielsweise die Gestaltung von Tastaturen oder ergonomischen Bildschirmen, behandelt.⁵⁸

⁵⁷ Grafik übernommen aus ACM SIGCHI (1996): Curricula for Human Computer Interaction. Chapter 2: Human-Computer Interaction. Online im Internet. URL: <http://www.acm.org/sigchi/cdg/cdg2.html>, Download am 19.01.2006.

⁵⁸ Vgl. Streitz (1988): S. 3.

3.1.3 Software Ergonomie

Eng verwandt, aber nicht deckungsgleich mit der HCI ist die Ergonomie⁵⁹. Diese Wissenschaft beschäftigt sich mit der Tätigkeit des Menschen mit technischen Hilfsmitteln und der Arbeitsumgebung im Allgemeinen. Anatomische, physiologische und psychologische Kenntnisse werden von der Ergonomie angewendet, um die zwei wesentlichen Ziele zu erreichen, welche einerseits die optimale Nutzung der menschlichen Ressourcen und andererseits die Sicherung von menschlicher Gesundheit und Wohlbefinden sind.⁶⁰ Erforderlich für das Erreichen dieser Ziele ist die Anpassung der Eigenschaften von Software an die psychischen Eigenschaften der damit arbeitenden Menschen, um auf diese Art und Weise erhöhten Belastungen der Anwender entgegenzuwirken.⁶¹

Die Software Ergonomie erforscht die Merkmale benutzer- und aufgabengerechter Software und entwickelt Verfahren und Softwareunterstützung für die Gestaltung von User Interfaces. Anliegen der Software Ergonomie ist es, das Zusammenwirken aller Komponenten, die die Arbeitssituation der Anwender bestimmen, nämlich Mensch, Aufgabe, Technik und organisatorischer Rahmen, zu optimieren.⁶² Softwareergonomische Fragestellungen beschränken sich nicht nur auf solche, welche die Präsentation interaktiver Software betreffen; technische Merkmale wie Zuverlässigkeit oder Effizienz werden bei softwareergonomischen Überlegungen allerdings vorausgesetzt.⁶³ Das Fachgebiet der Software Ergonomie erzielt durch das Zusammenwirken verschiedener Forschungsströmungen Erkenntnisse für die ergonomisch angemessene Systementwicklung und -gestaltung:⁶⁴

- Die **technisch orientierte Forschung** versucht, durch technische Weiterentwicklungen Fortschritte in der Interaktion zwischen Mensch und Computer zu erzielen. Formulare und Menüs als systemgeführte Dialogtechniken wurden als Alternative zur Kommando-Interaktion eingeführt. Das Prinzip der direkten Manipulation und damit verbunden der Einsatz von Fenstertechnik erleichterten die Computerhandhabung beträchtlich. Hypertext-Systeme, welche dem Benutzer ein einfaches Bewegen in hochvernetzten Informationsstrukturen ermöglichen, oder neue Eingabe- und Ausgabetechniken, beispielsweise die Handschrifteneingabe, Gestenerkennung oder die akustische Sprachverarbeitung, spielen ebenso eine entscheidende Rolle für ein verbessertes Zusammenspiel zwischen Mensch und Computer. Ein weiteres Teilgebiet der technisch orientierten Forschung zur Software Ergonomie entwirft Werkzeuge zur Entwicklung von User Interfaces, User Interface Management Systeme.
- Menschliche Wahrnehmung, Denken, Problemlösen und Lernen bilden die wesentlichen Aspekte, mit welchen die **kognitiv-psychologische Forschung** die Systemgestaltung betrachtet. Diese Forschungsströmung brachte etwa Hinweise zur optimalen Gestaltung von

⁵⁹ Vgl. Dix et al. (1993): S. 99; Streitz (1988): S. 3.

⁶⁰ Vgl. The Ergonomics Society: Ergonomics. Definition. Online im Internet. URL: <<http://www.ergonomics.org.uk/ergonomics/definition.htm>>, Download am 26.01.2006.

⁶¹ Vgl. Bräutigam, Lothar (2000): Software-Ergonomie. Online im Internet. URL: <<http://www.sozialnetz-hessen.de/ca/ph/het/Hauptpunkt/aaaaaaaaaahfi/Unterpunkt/aaaaaaaaaaaiih/HauptframeID/aaaaaaaaaaaalul/HauptframeTemplate/aaaaaaaaaa aaapq/>>, Download am 19.01.2006.

⁶² Vgl. Maaß (1993): S. 191.

⁶³ Vgl. ebenda.

⁶⁴ Vgl. ebenda: S. 192ff.

Piktogrammen als bildliche Darstellung von Systemobjekten, zum konsistenten Aufbau von Bildschirmmasken und zur maximalen Tiefe von Menübäumen hervor. Die Analyse von Benutzungsfehlern und Hilfeanforderungen von Anwendern lieferte neue Ansätze für die Unterstützung der Anwender durch Systemausgaben und Hilfefunktionen, mit der Softwaredokumentation sowie in Schulungen.

- Die **arbeitspsychologische Forschung** zur Software Ergonomie betrachtet den Computereinsatz im Zusammenhang menschengerechter Arbeitsgestaltung. Kriterien humaner Arbeitstätigkeiten sind nach *Ulich*⁶⁵ Schädigungsfreiheit, Beeinträchtigungslosigkeit, Persönlichkeitsförderlichkeit und Zumutbarkeit. Möglichst leicht handhabbare Systeme zu schaffen, die Maxime der frühen Benutzerfreundlichkeitsforschung, wird nicht länger angestrebt. Menschliche Handlungs-, Gestaltungs- und Entscheidungsspielräume sind erwiesenermaßen die wichtigsten Voraussetzungen für Arbeitszufriedenheit und für eine qualifikationsfördernde Arbeitsumgebung.

Auf Software Usability, den Maßstab zur Bewertung der Software Ergonomie, soll im folgenden Abschnitt eingegangen werden.

3.1.4 Software Usability

"Usability is the measure of the quality of the user experience when interacting with something - whether a Web site, a traditional software application, or any other device the user can operate in some way or another."⁶⁶

Usability, oder Gebrauchstauglichkeit⁶⁷, wie es in der offiziellen deutschen Übersetzung nach DIN EN ISO 9241-11 1998 heißt, beschreibt die Qualität der Interaktion zwischen Mensch und Maschine. Trotz der breit angelegten Definition von *Nielsen* bezieht sich Usability meist auf computerverwandte Kontexte.

Die Qualität interaktiver Systeme wird mit einer Fülle von Begriffen beschrieben: Neben Usability dienen auch Bedienbarkeit, Bedienerfreundlichkeit, User Friendliness, Ease of Use, Benutzerfreundlichkeit, Gebrauchstauglichkeit, Handhabbarkeit, Nutzungsqualität sowie User Experience und Benutzerzufriedenheit der Darstellung der Qualität der Mensch Computer Interaktion.⁶⁸ Diese Begriffe werden in der Literatur größtenteils synonym verwendet. Zu den Begriffen Usability und User Friendliness findet man beispielsweise eine Vielzahl von Definitionen. Vergleicht man die Definitionen zu Usability mit denen zu User Friendliness, dann zeigen sich jedoch keine bedeutenden Unterschiede.⁶⁹

⁶⁵ Vgl. Ulich (1988): S. 49.

⁶⁶ Nielsen nach Eichinger, Armin: Usability - Vorbemerkungen. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-vorbemerkungen.html>>, Download am 18.01.2006.

⁶⁷ Deutsches Institut für Normung (2003): EN ISO 9241-11: 1998. S. 4.

⁶⁸ Vgl. Geis, Thomas: Willkommen im Usability Begriffszoo. Online im Internet. URL: <<http://www.fit-fuer-usability.de/1x1/basics/begriffszoo.html>>, Download am 19.01.2006.

⁶⁹ Vgl. Englisch (1993): 21ff.

Der Name „Usability“ wurde ursprünglich geprägt, um den Begriff der „User Friendliness“ abzulösen, welcher in den frühen 1980er Jahren mit einer Menge unerwünschter Nebenbedeutungen assoziiert wurde. Allerdings ist der Bedeutungsumfang von Usability heute ähnlich vage, wie es jener von User Friendliness war, den er eigentlich verdrängen sollte.⁷⁰

Software Usability kennzeichnet die Nutzungsqualität von Software für den Benutzer. Software-technische Merkmale sind in diese Bewertung nur insofern eingeschlossen, als diese Auswirkungen auf die Softwarenutzung haben. Anstatt der Bewertung der technischen Qualität der Software wird ihre Wirkung am Arbeitsplatz und auf den Benutzer festgestellt. Das Produkt wird also danach beurteilt, ob und inwieweit es am Arbeitsplatz des Benutzers anforderungsgemäß und zuverlässig funktioniert.⁷¹

Die erfahrene Nutzungsqualität setzt sich aus vielen teilweise interdependenten Faktoren zusammen:

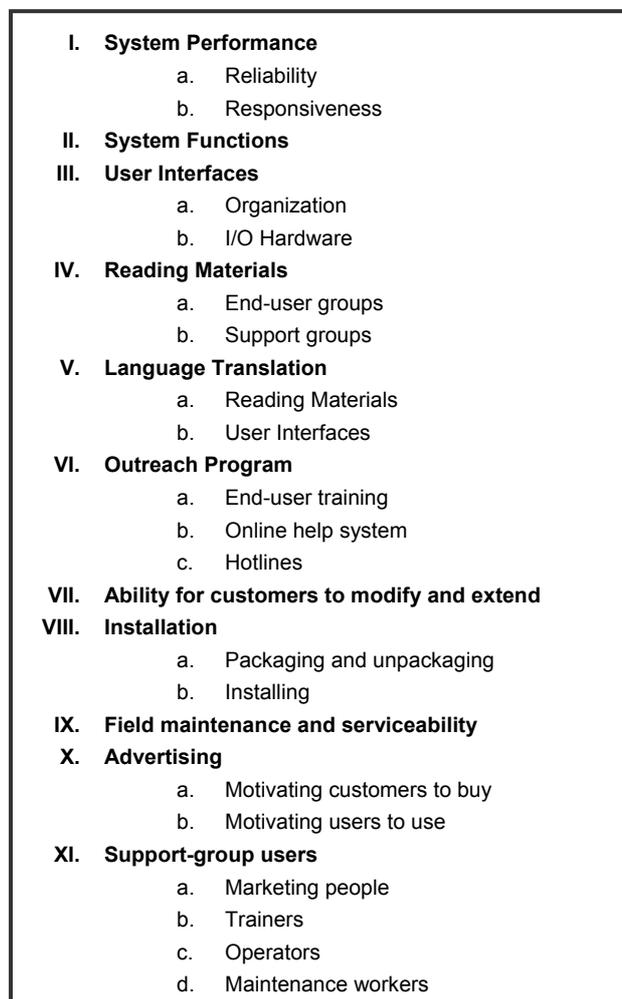


Abbildung 5: Usability-Komponenten⁷²

⁷⁰ Vgl. Bevan, Nigel/Kirakowski, Jurek/Maissel, Jonathan (1991): What is Usability? Online im Internet. URL: <<http://www.usability.serco.com/papers/whatis92.pdf>>, Download am 02.02.2006.

⁷¹ Vgl. DATech (2004): DATech-Prüfhandbuch Gebrauchstauglichkeit. Leitfaden für die ergonomische Evaluierung von Software auf Grundlage von DIN EN ISO 9241, Teile 10 und 11. Online im Internet. URL: <http://www.datech.de/share/files/Pruefhandbuch_ISO_9241.pdf>, Download am 23.01.2006, S. 9.

⁷² Eigene Darstellung, teilweise übernommen aus Gould/Boies/Ukelson (1997): S: 233.

Die grundsätzlichen Leitkriterien zur Software Usability finden sich in der Norm DIN EN ISO 9241-11 1998. Danach ist Gebrauchstauglichkeit wie folgt definiert:

„Das Ausmaß, in dem ein Produkt durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und zufriedenstellend zu erreichen“⁷³

Demnach kann Usability als das Attribut einer Interaktion zwischen Benutzer und Software verstanden werden. Usability ist immer in Abhängigkeit vom Produkt und der Situation, welche durch die Ziele des Benutzers sowie dem jeweiligen Kontext begründet wird, zu betrachten. DIN EN ISO 9241-11 präsentiert Effizienz, Effektivität und Benutzerzufriedenheit als Kriterien zur Bestimmung der Software Usability.⁷⁴

- **Effizienz der Software:** Stehen die vom Benutzer eingesetzten Ressourcen in Relation zum Ergebnis?
- **Effektivität der Software:** Ermöglicht die Software dem Benutzer das Erledigen seiner Aufgaben?
- **Zufriedenheit der Anwender einer Software:** Wie zufrieden ist der Benutzer mit dem durch die Software erreichten Arbeitsziel? Ist der Benutzer gegenüber dem Softwareeinsatz positiv eingestellt?

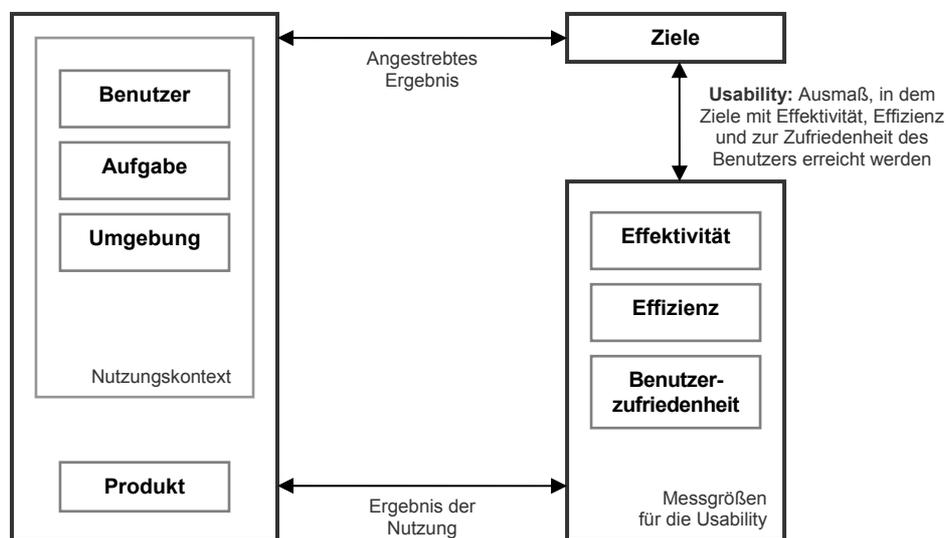


Abbildung 6: Anwendungsrahmen der Software Usability nach DIN EN ISO 9241-11⁷⁵

⁷³ Deutsches Institut für Normung (2003): EN ISO 9241-11: 1998. S. 4.

⁷⁴ Vgl. Eichinger, Armin: Usability - Definition. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-definition.html>>, Download am 18.01.2006.

⁷⁵ Eigene Darstellung, teilweise übernommen aus Deutsches Institut für Normung (2003): EN ISO 9241-11: 1998. S. 6.

Zur Bewertung der Usability einer Software sind demnach folgende Elemente notwendig:⁷⁶

- Darstellung der Ziele der Interaktion
- Darstellung der Benutzer
- Darstellung der Aufgaben, das sind jene Handlungen, die auszuführen sind, um ein Ziel zu erreichen
- Darstellung der Ausstattung (im speziellen Software und Hardware)
- Darstellung der Umgebung, welche die Eigenschaften der sozialen und physikalischen Umwelt, beispielsweise die Organisationsstruktur oder die Raumtemperatur, beinhaltet
- Auswahl von Usability Messgrößen, welche sich auf Effizienz, Effektivität und Zufriedenheit beziehen

Dumas und *Redish* geben eine alternative Definition des Begriffes Usability: Menschen, welche die Software benutzen, können diese schnell und einfach bedienen, um ihre Aufgaben damit zu erledigen. Die Definition basiert auf folgenden vier Elementen:⁷⁷

- I. **Usability bedeutet das Fokussieren auf den Anwender.** Um ein Produkt mit hoher Usability zu entwickeln, bedarf es des Verstehens von und der Zusammenarbeit mit tatsächlichen oder potentiellen Anwendern.
- II. **Menschen verwenden Produkte, um produktiv zu sein.** Die Leistungsziele der Anwender müssen von den Entwicklern eines Produkts verstanden werden, um die Usability eines Produkts garantieren zu können.
- III. **Die Anwender sind Menschen, die das Erledigen ihrer Aufgaben anstreben.** Sie verknüpfen Usability mit Leistungsfähigkeit. Eine Software ist ein Instrument, das Menschen bei der Durchführung ihrer Aufgaben helfen soll. Die Bereitschaft der Anwender, Zeit damit zu verbringen, das Benutzen solch eines Instruments zu erlernen, ist sehr gering.
- IV. **Die Anwender entscheiden, ob ein Produkt leicht zu bedienen ist,** nicht die Entwickler der Software. Die Anwender wägen Zeit und Mühe, die für das Erlernen von Funktionalitäten eines Produkts erforderlich sind, mit dem erwarteten Nutzen dieser Funktionalitäten ab. Sie bewerten somit, ob sich der Aufwand lohnt. Bei Produkten mit geringer Usability nutzen die Anwender nur einen geringen Teil der zur Verfügung stehenden Funktionalitäten.

Zunächst wurde Software Usability lediglich als Ergänzung des Nutzens (Utility) bzw. der „Funktionalität“⁷⁸ eines Softwareprodukts betrachtet: Der funktionale Nutzen soll erfassen, ob die Software die für das Unternehmen benötigten Funktionen prinzipiell zur Verfügung stellt und somit die technischen Anforderungen des Unternehmens erfüllt. Demgegenüber würde Usability der Frage nachgehen, wie diese Funktionen bereitgestellt werden und wie gut diese durch den Anwender genutzt werden können. Funktionalität und Usability kombiniert, würden die Nützlichkeit (Usefulness)

⁷⁶ Vgl. Deutsches Institut für Normung (2003): EN ISO 9241-11: 1998. S. 14; Eichinger, Armin: Usability - Definition. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-definition.html>>, Download am 18.01.2006.

⁷⁷ Vgl. Dumas/Redish (1999): S. 4ff.

⁷⁸ Ebenda: S. 4.

eines Softwareprodukts ergeben. Usability wäre demnach eine Qualität des Produktes, welche unabhängig vom Nutzungskontext erfasst werden könnte.⁷⁹

Die Definition von Software Usability, welche dieser Arbeit zugrunde liegt, orientiert sich an der Systembenutzung, am Benutzer und am Nutzungskontext.⁸⁰ Sie versteht Usability als Nutzungsqualität in dem Sinne, dass ein starker Zusammenhang zwischen Softwaresystem und Nutzungskontext besteht. Es muss folglich das Ziel der Softwareanbieter sein, Systeme mit hoher Effektivität und Effizienz zu gestalten. Denn ist eine Software effektiv und effizient, werden die Benutzer ohne Beeinträchtigung mit dem System arbeiten und eine positive Einstellung zu diesem entwickeln können. Die Anwender werden zufrieden sein. Im Mittelpunkt dieser Definition steht die Ziel- und Aufgabenorientierung. Diese Auffassung von Usability teilt auch die überwiegende Mehrzahl der Usability-Experten.⁸¹

3.2 Stellenwert von Usability

Der Stellenwert, den Software Usability heute in der Entwicklung von Softwaresystemen genießt, hat sich anfänglich daraus entwickelt, dass die Benutzer ihrem Wunsch nach einer verbesserten Software Ergonomie zunehmend Ausdruck verliehen. Der Druck von Benutzerseite diente gleichsam als Katalysator für die breite Akzeptanz von Usability. Darauf wird in diesem Kapitel zunächst eingegangen. Anschließend wird der Stellenwert von Usability als besten Maßstab für die Akzeptanz eines Softwaresystems diskutiert. Zahlreiche Studien deuten daraufhin, dass Usability die nützlichste Determinante für den Erfolg eines Softwaresystems darstellt.

3.2.1 Druck von Benutzerseite

Bis Mitte der 1980er Jahre setzten sich ergonomische Erkenntnisse kaum durch. Die Benutzer waren mit unzuverlässigen Systemen und beträchtlichen Antwortzeiten konfrontiert. Erheblich zum Usability-Fokus der Herstellerfirmen beigetragen haben die Benutzer, welche zunehmend nach Produkten mit verbesserter Ergonomie nachgefragt haben. Beispielhaft sei hier die Initiative von *SHARE*, dem Interessensverband der IBM-Anwender, genannt: Sie stellten im Report of the Interactive Systems Task Force konkrete Forderungen an IBM (Requirements for systems 1985-1990). Darin machten sie deutlich, dass die Benutzerfreundlichkeit (Ease of Use) der von ihnen geforderten Funktionen noch wichtiger sei als die Funktionen an sich. Angesichts der Tatsache, dass zukünftig Softwarebenutzer nicht ausschließlich EDV-Experten wären, sollten die Systeme konsistent, integriert, robust, leistungsfähig, konfigurierbar, leicht zu erlernen und leicht zu benutzen sein.⁸²

⁷⁹ Vgl. Burmester/Hassenzahl/Koller (2002): S. 32f.

⁸⁰ Vgl. Bevan, Nigel/Kirakowski, Jurek/Maissel, Jonathan (1991): What is Usability? Online im Internet. URL: <<http://www.usability.serco.com/papers/whatis92.pdf>>, Download am 02.02.2006.

⁸¹ Vgl. Burmester/Hassenzahl/Koller (2002): S. 33.

⁸² Vgl. Hellbardt, Günter: Vorlesung Software-Ergonomie. Texte und Folien. Einführung. Online im Internet. URL: <http://www1.informatik.uni-jena.de/lehre/SoftErg/vor_e100.htm#einfg>, Download am 23.01.2006.

Karat, Designerin für Human Interfaces am Thomas J. Watson Research Center der IBM, verlieh ihrer Forderung nach erhöhter Usability in ihren User's Bill of Rights Ausdruck.⁸³

- I. Der Anwender hat immer Recht. Existiert ein Bedienungsproblem, so ist das System das Problem, nicht der Anwender.
- II. Der Anwender hat das Recht auf einfach zu installierende Software und Hardware.
- III. Der Anwender hat das Recht auf ein System, das genau die vom Hersteller versprochene Leistung besitzt.
- IV. Der Anwender hat das Recht auf einfache Bedienungsanleitung, die das System hilft zu verstehen und anzuwenden.
- V. Der Anwender hat das Recht auf eine Kontrolle des IT-Systems und muss entsprechende Informationen erhalten.
- VI. Der Anwender hat das Recht, verständliche und präzise Informationen über die gerade laufende Aufgabe bis zur vollständigen Abarbeitung zu erhalten.
- VII. Der Anwender hat das Recht, über die Systemanforderungen für einen erfolgreichen Software- oder Hardwareeinsatz informiert zu werden.
- VIII. Der Anwender hat das Recht, über die Grenzen der Systemleistungen Bescheid zu wissen.
- IX. Der Anwender hat das Recht, mit dem Technikanbieter zu kommunizieren und eine hilfreiche Antwort zu erhalten.
- X. Der Anwender sollte der Herr über Hardware und Software sein, und nicht umgekehrt. Produkte müssen natürlich und intuitiv sein.

Abbildung 7: User's Bill of Rights nach Karat⁸⁴

Der Lobbyismus von Benutzerseite artikuliert anfänglich hauptsächlich die Forderung nach möglichst einfachen Systemen für scheinbar ahnungslose Benutzer. Schenkt man jener Definition von Software Usability Beachtung, welche auch für diese Arbeit herangezogen wurde, so erkennt man, dass die leichte Handhabbarkeit eines Systems nicht zur Maxime in der Softwareentwicklung erhoben werden kann. Lässt ein Softwareprodukt an Effektivität und Effizienz zu wünschen übrig, so wird der Benutzer die Software nicht als gebrauchstauglich empfinden, auch wenn sie leicht zu bedienen ist. Irreführend ist nämlich jene Annahme vom nichtwissenden Benutzer. Im Gegenteil, Benutzer sind in ihrem Fach meist qualifizierte Personen, welche vom Softwaresystem bei ihrer Arbeit angemessen zu unterstützen sind.⁸⁵ Studien zeigen außerdem, dass der Ease of Use eines Softwareprodukts Auswirkungen auf den tatsächlichen Gebrauch hat, diese aber im Vergleich zu jenen, welche die Usefulness eines Systems nach sich zieht, gering sind.⁸⁶

3.2.2 User Satisfaction als Maßstab für die Akzeptanz der Software

„End-user satisfaction is [...] the affective attitude towards a specific computer application by someone who interacts with the application directly.“⁸⁷

Ähnlich wird der Begriff der User Satisfaction in der DIN EN ISO 9241-11 beschrieben, nämlich als positive Einstellung gegenüber der Nutzung der Software.⁸⁸

⁸³ Vgl. Wildstrom (1998): S. 18.

⁸⁴ Eigene Darstellung, Inhalte übernommen aus Karat nach Wildstrom (1998): S. 18.

⁸⁵ Vgl. Maaß (1993): S. 194.

⁸⁶ Vgl. Au/Ngai/Cheng (2002); Gelderman (1998); Adams/Nelson/Todd (1992); Delone/McLean (1992).

⁸⁷ Doll/Torkzadeh (1988): S. 261.

⁸⁸ Vgl. Deutsches Institut für Normung (2003): EN ISO 9241-11: 1998. S. 4.

Cyert und *March* (1963) waren die ersten, welche das Konzept der User Satisfaction als Maßstab für den Erfolg von Softwaresystemen präsentierten. Es stellt heute wohl den meist verbreiteten Ansatz zur Bewertung des Erfolgs einer Software dar.⁸⁹ *Somers, Nelson* und *Karimi* beweisen, dass das Konstrukt der User Satisfaction auch im Falle von ERP-Systemen eine der wichtigsten Determinanten für deren Erfolg darstellt und eine geeignete Methode ist, um die Effizienz und Effektivität eines ERP-Systems beurteilen zu können.⁹⁰

Softwaresysteme können die Erwartungen, welche an sie gestellt werden, oft nicht erfüllen, nicht etwa weil sie technische Mängel aufweisen würden, sondern weil psychologische und organisationale Aspekte bei Entwicklung, Einführung und Anwendung der Software zu wenig beachtet wurden.⁹¹ Nur 10 Prozent der Implementationen neuer Informationssysteme scheitern an technischen Aspekten.⁹²

Der Erfolg von Informationssystemen kann also nur im jeweiligen Anwendungskontext beurteilt werden.⁹³ Das gilt auch für ERP-Systeme: Während viele Unternehmen durch den Einsatz von ERP-Systemen Leistungszuwächse verbuchen konnten, wurden andernorts Implementationen von ERP-Systemen nicht einmal zu Ende geführt. Eine von fünf ERP-Implementationen scheitert gänzlich.⁹⁴

Die Mitarbeiter eines Unternehmens erweisen sich als das kritische Element für den Erfolg von Informationssystemen. Ein Informationssystem kann nicht im vollen Ausmaß genutzt werden, wenn es die Erwartungen und Bedürfnisse der Endbenutzer nicht erfüllt.⁹⁵ Die Anwender haben also einen entscheidenden Einfluss auf den Nutzen, welchen ein Unternehmen durch den Einsatz eines Softwaresystems erzielen kann.⁹⁶

3.3 Nutzen von Usability

Schätzungen zufolge werden 20 Prozent der am Computer verbrachten Arbeitszeit auf Grund vermeidbarer Benutzungsprobleme nicht sinnvoll genutzt. Nicht nur die Kunden ergonomisch schlecht gestalteter Software haben Kosten infolge mangelnder Usability zu tragen; auf der Seite der Softwareanbieter sind es überlaufene Support-Hotlines und drohender Imageverlust, die Kosten darstellen.⁹⁷

Der Nutzen von Usability lässt sich in Erträgen des Einsatzes von Usability Engineering messen. Grundsätzlich setzt sich der Nutzen von Usability aus drei Komponenten zusammen: eine Steigerung der Produktivität beim Einsatz der Software, eine Reduktion anfallender Kosten bei ihrer Entwicklung

⁸⁹ Vgl. *Au/Ngai/Cheng* (2002): S. 452.

⁹⁰ Vgl. *Somers/Nelson/Karimi* (2003): S. 613.

⁹¹ Vgl. *Markus/Keil* (1994): S. 11.

⁹² Vgl. *Bikson/Gutek* (1984) nach *Somers/Nelson/Karimi* (2003): S. 596.

⁹³ Vgl. *Au/Ngai/Cheng* (2002): S. 453.

⁹⁴ Vgl. *Soh/Tien/Tay-Yap* (2000) nach *Somers/Nelson/Karimi* (2003): S. 596; *Trunick* (1999): S. 25.

⁹⁵ Vgl. *Martinsons/Chong* (1999): S. 146f.

⁹⁶ Vgl. *Somers/Nelson/Karimi* (2003): S. 596.

⁹⁷ Vgl. *Geis/Hartwig* (1998): S. 168.

und eine bessere Wettbewerbsfähigkeit des Produktes.⁹⁸ Softwareanbieter bestätigen, dass die Nutzen von Usability die dafür aufzuwendenden Kosten deutlich überwiegen. Laut IBM, spart momentan jeder in Usability investierte Euro 10 bis 100 Euro.⁹⁹ Dieser Nutzen rechtfertigt die hohen Anteile am Softwareentwicklungsbudget, welche für Usability-Aspekte für gewöhnlich reserviert werden. Heute noch bei etwa 10 Prozent, werden diese zukünftig auf rund 20 Prozent anwachsen.¹⁰⁰

Der Wert von Usability bei Softwareprodukten, welche intensives und teures Training verlangen, was im Speziellen auf ERP-Systeme zutrifft, ist besonders groß.¹⁰¹ Im Folgenden werden die Nutzen für Unternehmen, welche Softwaresysteme mit hoher Usability einsetzen, sowie die Nutzen für die Softwareanbieter dargestellt.

3.3.1 Nutzen für das Unternehmen

Der Nutzen ergonomisch gestalteter Software setzt sich für Anwender aus folgenden Komponenten zusammen:

- **Usability erhöht die Erlernbarkeit:** Die Qualität der Benutzerschnittstelle beeinflusst den entstehenden Schulungs- und Beratungsbedarf und somit die Kosten, welche für Schulung und Beratung aufgewendet werden müssen.¹⁰² Bei verbesserter Usability kann eine Reduktion der Schulungskosten um 50 Prozent erwartet werden.¹⁰³ Neben den Erlernzeiten verkürzt Usability auch die Wiedererlernzeiten.¹⁰⁴
- **Usability erhöht die Produktivität:** Mitarbeiter arbeiten mit gesteigerter Produktivität, wenn sie von benutzerorientierter Software unterstützt werden.¹⁰⁵ In einer branchenübergreifenden Studie unter 4.000 US-amerikanischen Mitarbeitern gaben 90 Prozent der Manager an, dass sich die Produktivität ihrer Mitarbeiter sehr gesteigert hätte.¹⁰⁶
- **Usability reduziert die Fehlerrate:** Dazu tragen auch die geringeren Fehlerraten bei ergonomisch gestalteter Software bei, wie eine Fallstudie der New York Stock Exchange zeigt: Hier konnte die Fehlerrate um den Faktor 10 gesenkt werden.¹⁰⁷ Eine weitere Fallstudie beschreibt, wie ein Unternehmen mit dem Re-Design der User Interfaces nach Usability-

⁹⁸ Vgl. Mutschler, Bela/Reichert, Manfred (2004): Usability-Metriken als Nachweis der Wirtschaftlichkeit von Verbesserungen der Mensch-Maschine-Schnittstelle. Online im Internet. URL: <<http://www.mutschler.info/downloads/MetriKon-Mutschler-Reichert-Usability-Metriken.pdf>>, Download am 20.01.2006.

⁹⁹ Vgl. IBM: Cost justifying ease of use. Complex solutions are problems. Online im Internet. URL: <http://www-3.ibm.com/ibm/easy/eou_ext.nsf/Publish/23>, Download am 20.01.2006.

¹⁰⁰ Vgl. Mutschler, Bela/Reichert, Manfred (2004): Usability-Metriken als Nachweis der Wirtschaftlichkeit von Verbesserungen der Mensch-Maschine-Schnittstelle. Online im Internet. URL: <<http://www.mutschler.info/downloads/MetriKon-Mutschler-Reichert-Usability-Metriken.pdf>>, Download am 20.01.2006.

¹⁰¹ Vgl. Black, Jane (2002): Usability Is Next to Profitability. Online im Internet. URL: <http://www.businessweek.com/technology/content/dec2002/tc2002124_2181.htm>, Download am 20.01.2006.

¹⁰² Vgl. Preim (1999): S. 2; Donahue (2001): S. 34f; Kalbach (2003): S. 9.

¹⁰³ Vgl. Nielsen, Jakob/Pernice Coyne, Kara (2001): A Useful Investment. Online im Internet. URL: <http://www.cio.com/archive/021501/et_pundits.html>, Download am 22.01.2006.

¹⁰⁴ Vgl. Usability Forum: Was bringt Usability? Online im Internet. URL: <<http://www.usability-forum.com/bereiche/kosten.shtml>>, Download am 18.01.2006.

¹⁰⁵ Vgl. Donahue (2001): S. 34.

¹⁰⁶ Vgl. Schneider (1985) nach Harrison/Henneman/Blatt (1994): S. 213.

¹⁰⁷ Vgl. Gibbs (1997): S. 70f.

Prinzipien eine 25-prozentige Steigerung der Effizienz erreichte und die Benutzerfehler um 25 Prozent reduzierte.¹⁰⁸

- **Usability senkt Support-Kosten:** Zeichnet sich eine Software durch hohe Usability aus, sind die Anwender weniger häufig gezwungen, den Support zu kontaktieren.¹⁰⁹
- **Usability reduziert Wartungskosten:** 80 Prozent der Kosten, welche in einem Softwarelebenszyklus anfallen, sind der Wartung zuzurechnen. Der Großteil des Wartungsbedarfs entsteht auf Grund nicht beachteter Anforderungen der Anwender und anderen Problemen im Zusammenhang mit mangelnder Usability.¹¹⁰
- **Usability erhöht die Benutzerzufriedenheit:** Software, welche die Bedürfnisse der Benutzer berücksichtigt, steigert die Benutzerzufriedenheit deutlich.¹¹¹ Neben der Zufriedenheit des Benutzers erweist sich auch die Benutzerfreude mit der Arbeit an einem Softwaresystem als entscheidendes Kriterium für die Bewertung der Software durch die Benutzer.¹¹²
- **Usability erhöht die Arbeitszufriedenheit:** Eine Studie über ergonomisch gestaltete Arbeitsumgebungen brachte folgendes Ergebnis: Jene Mitarbeiter, welche unter ergonomisch schlechteren Bedingungen arbeiteten, wiesen höhere Abwesenheitsraten, weniger Zufriedenheit mit der Arbeit und eine höhere Fluktuation auf.¹¹³

3.3.2 Nutzen für den Softwareanbieter

Für die Softwareanbieter macht sich das Investieren in die Usability ihrer Produkte bezahlt. Folgende Nutzenkategorien lassen sich auf Seite der Hersteller feststellen:

- **Usability bringt neue Ideen:** Usability Engineering nimmt neue von Anwendern gewünschte Ideen auf. Somit entsteht Potential, Produktinnovationen zu lancieren, welche vom Markt nachgefragt werden.¹¹⁴
- **Usability bewirkt Erfolgssicherheit:** Usability-orientierte Softwareentwicklung erhöht die Wahrscheinlichkeit für die Akzeptanz eines Produkts am Markt.¹¹⁵
- **Usability verbessert Planbarkeit:** Eine Studie zu Softwareentwicklungskosten zeigte, dass 63 Prozent der großen Softwareprojekte ihr Budgetziel nicht einhalten konnten. Die vier meist genannten Gründe für die höheren Ausgaben konnten auf eine mangelnde Benutzerorientierung im Entwicklungsprozess zurückgeführt werden; dazu zählten häufige Änderungsvorschläge durch User, übersehene Aufgabenstellungen, fehlendes Wissen des Users über seine

¹⁰⁸ Vgl. Dray (1995): S. 18.

¹⁰⁹ Vgl. Donahue (2001): S. 35.

¹¹⁰ Vgl. Pressman (1992) nach Usability Professionals' Organization (UPA): The ROI of Usability. Online im Internet. URL: <http://www.upassoc.org/usability_resources/usability_in_the_real_world/roi_of_usability.html>, Download am 24.01.2006.

¹¹¹ Vgl. Schneider (1985) nach Harrison/Henneman/Blatt (1994): S. 210.

¹¹² *Burmester, Hassenzahl* und *Koller* entwickelten ein Zwei-Komponenten-Modell für die Bewertung der Nutzungsqualität einer Software, bestehend aus der wahrgenommenen pragmatischen und hedonistischen Qualität. Die erste der beiden setzt sich aus Usability und Nutzen zusammen und kann als aufgabenbezogene Qualität betrachtet werden. Die hedonistische Qualität hingegen bezieht sich auf die Wahrnehmung nicht-zielorientierter Aspekte, welche beispielsweise mit Attributen wie innovativ, originell, aufregend oder exklusiv beschrieben werden können. Vgl. *Burmester/Hassenzahl/Koller* (2002): S. 32ff.

¹¹³ Vgl. Schneider (1985) nach Harrison/Henneman/Blatt (1994): S. 210.

¹¹⁴ Vgl. Kalbach (2003): S. 11.

¹¹⁵ Vgl. ebenda: S. 8; Eichinger, Armin: Usability - Kosten. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-kosten.html>>, Download am 18.01.2006.

Anforderungen an die Software und eine unzureichende Kommunikation zwischen Anwendern und Usability-Experten.¹¹⁶

- **Usability senkt Entwicklungskosten:** Frühes Erkennen von Problemen, zu Zeitpunkten, an denen die Korrektur noch leichter und kostengünstiger ist, reduziert die Kosten der Fehlerbeseitigung.¹¹⁷ Ist ein Softwaresystem bereits in der Entwicklungsphase, muss für die Korrektur eines Problems zehnmal mehr ausgegeben werden, als dafür in der Designphase nötig gewesen wäre. Wird man erst nach Markteinführung des Systems auf den Fehler aufmerksam, so kostet die Beseitigung dieses Fehlers rund hundertmal mehr als in der Designphase.¹¹⁸

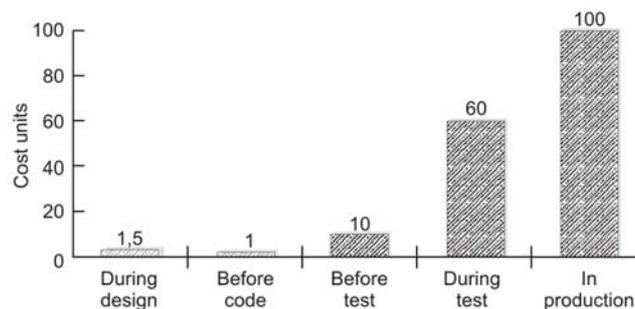


Abbildung 8: Relative Kosten der Fehlerbeseitigung in den verschiedenen Entwicklungsphasen¹¹⁹

- **Usability reduziert die Entwicklungszeit:** Funktionalitäten, die für die Anwender von geringerer Wichtigkeit sind, müssen gar nicht erst entwickelt werden bzw. können auf später verschoben werden.¹²⁰
- **Usability verringert die Dokumentationskosten:** Ergonomisch gestaltete Software benötigt weniger Dokumentation. Außerdem erleichtert sich das Erstellen der benötigten Dokumentation; es ist somit günstiger als bei nicht ergonomisch gestalteter Software.¹²¹
- **Usability bringt neue Kunden:** Erfüllt die Software die Bedürfnisse der Zielgruppe, so fällt es leichter, diese am Markt zu positionieren und neue Kunden zu gewinnen.¹²² Kunden von Softwareprodukten betrachten die Leistungsfähigkeit und die Benutzungsfreundlichkeit als entscheidende Kriterien für die Kaufentscheidung.¹²³
- **Usability steigert Markenwert und Loyalität:** Benutzer werten Usability als wichtige Anforderung an ein Softwaresystem.¹²⁴ Ein Softwareprodukt, das die Anforderungen seiner Kunden erfüllt, erzielt eine höhere Kundenbindung. Softwareanbietern gelingt es, das Vertrauen der Kunden in das Unternehmen zu erhöhen.¹²⁵

¹¹⁶ Vgl. Barker, Dean (2000): Cost Benefits of Usability Engineering. Online im Internet. URL: <<http://www.interfacearchitecture.net/articles/benefits.htm>>, Download am 10.02.2005.

¹¹⁷ Vgl. Donahue (2001): S. 33; Kalbach (2003): S. 9.

¹¹⁸ Vgl. Gilb (1988): S. 221.

¹¹⁹ Eigene Darstellung, teilweise übernommen aus Gilb (1988): S. 221.

¹²⁰ Vgl. Kalbach (2003): S. 9.

¹²¹ Vgl. Donahue (2001): S. 35.

¹²² Vgl. Kalbach (2003): S.10.

¹²³ Vgl. Harrison/Hennemann/Blatt (1994): S. 221.

¹²⁴ Vgl. Donahue (2001): S. 36.

¹²⁵ Vgl. Kalbach (2003): S. 10f.

- **Usability bringt Vorteile im Marketing:** Usability dient mehr und mehr als Differenzierungsinstrument im Wettbewerb der Softwareanbieter.¹²⁶ Außerdem riskieren Softwareanbieter, welche Produkte mit schlechter Usability auf den Markt bringen, negative Kritiken.¹²⁷

Nach der Darstellung der verschiedenen Nutzenkategorien soll im folgenden Abschnitt der Prozess des User Centred Design in den Mittelpunkt gerückt werden. Dieser hat zum Ziel, dass eine angemessene Usability einer Software garantiert werden kann.

3.4 User Centred Design

Die zahlreichen ISO Qualitätsstandards zur Software Ergonomie erheben Usability zum höchsten Ziel für den gesamten Produktlebenszyklus eines Softwaresystems. Dadurch ergibt sich für Usability eine strategisch wichtige Rolle speziell im Entwicklungsprozess, aber auch in allen anderen Lebenszyklusphasen einer Software. Um das Metaziel, die Sicherstellung hoher ergonomischer Softwarequalität, zu erreichen, bedarf es einem User Centred Design Prozess und dem Gebrauch geeigneter Techniken zum Evaluieren von Software Usability.¹²⁸

Neben dem User Centred Design hat sich in der Literatur der Begriff des Usability Engineering durchgesetzt. Unterschiede zwischen diesen beiden Begrifflichkeiten lassen sich kaum feststellen. *Preim*¹²⁹ definiert Usability Engineering als Prozess zur Entwicklung ergonomisch ausgereifter Benutzerschnittstellen. *Eichinger*¹³⁰ fasst den Begriff des Usability Engineering enger; es sei ein Prozess, bei dem die Usability eines Produktes definiert, gemessen und verbessert wird. Er umfasst somit alle Aktivitäten innerhalb eines Produktlebenszyklus, die einer gesteigerten Usability dienen. In dieser Arbeit werden die beiden Bezeichnungen synonym verwendet.

In der Praxis zeigt sich die Bedeutung von Schwächen in der Software Usability. Während nur 15 Prozent der Softwareprobleme mit den Funktionalitäten einer Software zusammenhängen, stehen 60 Prozent mit Usability in Beziehung. Es zeigt sich auch, dass jene Probleme, welche aus einer falschen Definition der Anforderungen resultieren, die kostenintensivsten sind, was ihre Korrektur betrifft. Obwohl bekannt ist, dass User Centred Design die Wahrscheinlichkeit des Erfolgs eines Softwareprodukts drastisch erhöht, werden die Methoden des User Centred Design längst nicht von allen Softwareanbietern angewandt.¹³¹

¹²⁶ Vgl. Mutschler, Bela/Reichert, Manfred (2004): Usability-Metriken als Nachweis der Wirtschaftlichkeit von Verbesserungen der Mensch-Maschine-Schnittstelle. Online im Internet. URL: <<http://www.mutschler.info/downloads/MetriKon-Mutschler-Reichert-Usability-Metriken.pdf>>, Download am 20.01.2006; Doanhue (2001): S. 36.

¹²⁷ Vgl. Kalbach (2003): S. 11; Donahue (2001): S. 36.

¹²⁸ Vgl. Bevan, Nigel/Curson, Ian (1998): Planning and Implementing User-Centred Design. Online im Internet. URL: <<http://www.usability.serco.com/papers/ucdtut97.pdf>>, Download am 04.02.2006.

¹²⁹ Vgl. Preim (1999): S. 534.

¹³⁰ Vgl. Eichinger, Armin: Usability Engineering. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-engineering.html>>, Download am 18.01.2006; Usability Forum: Usability Engineering. Online im Internet. URL: <<http://www.usability-forum.com/bereiche/usabilityengineering.shtml>>, Download am 18.01.2006.

¹³¹ Vgl. Bevan (1999): S. 95.

User Centred Design birgt große Kosteneinsparungschancen für Softwareanbieter. Je konsequenter ein User Centred Design Prozess verfolgt wird und je früher Methoden des Usability Testing angewendet werden, desto günstiger gestaltet sich die allfällige Fehlerbehebung: Die Beseitigung eines Problems kostet in der Entwicklungsphase zehnmals mehr als in der Designphase, ist das Produkt bereits am Markt, kostet es hundertmal mehr als in der Designphase.¹³²

Analog dazu formulieren *Gould, Boies* und *Ukelson* vier grundlegende Regeln für User Centred Design:

- **Frühes und kontinuierliches Fokussieren der Benutzer:**
Gespräche mit Benutzern führen anstatt deskriptive Informationen über ihre Aufgaben sammeln
- **Empirische Überprüfung der Usability:**
Usability messen und geeignete Anpassungen am System machen, um somit den Entwicklungsprozess rasant anzutreiben
- **Iterativer Designprozess:**
Notwendige Änderungen identifizieren, die Fähigkeit besitzen, diese durchzusetzen und eine grundlegende Bereitschaft für Änderungen an den Tag legen
- **Integriertes Design:**
Usability-Aspekte gleichzeitig entwickeln anstatt einzelne Komponenten isoliert zu betrachten

Abbildung 9: Vier Grundregeln des User Centred Design¹³³

Nachfolgend wird der Prozess des User Centred Design beschrieben. Daran anknüpfend wird auf Prinzipien und Methoden eingegangen.

3.4.1 Prozess des User Centred Design

Der Prozess des User Centred Design kann modellhaft als iterativer Ablauf folgender Aktivitäten betrachtet werden: (1) Zunächst wird der Nutzungskontext analysiert. Dieser ist durch die zu erledigenden Aufgaben, die Benutzer sowie die Arbeitsumgebung charakterisiert. (2) Daraus lassen sich die Benutzungsanforderungen ableiten. (3) Auf der Grundlage dieser konkreten Anforderungen kann mit dem Entwickeln von Designlösungen begonnen werden. (4) Anschließend werden die Designlösungen daraufhin getestet, inwieweit sie die vorher definierten Anforderungen erfüllen. Damit schließt sich der Kreislauf des User Centred Design (siehe Abbildung 10).¹³⁴ Sie werden solange wiederholt, bis das Softwaresystem die Anforderungen erfüllt, wobei zu beachten ist, dass sich die Usability-Anforderungen an die Designlösung im Laufe des Softwareentwicklungsprozesses zunehmend spezifizieren.¹³⁵

¹³² Vgl. Gilb (1988): S. 221.

¹³³ Eigene Darstellung, Inhalte übernommen aus Gould/Boies/Ukelson (1997): S. 233ff.

¹³⁴ Vgl. Bevan (1999): S. 93; Bevan, Nigel/Curson, Ian (1998): Planning and Implementing User-Centred Design. Online im Internet. URL: <http://www.usability_serco.com/papers/ucdtu97.pdf>, Download am 04.02.2006.

¹³⁵ Vgl. Bevan (1999): S. 93.

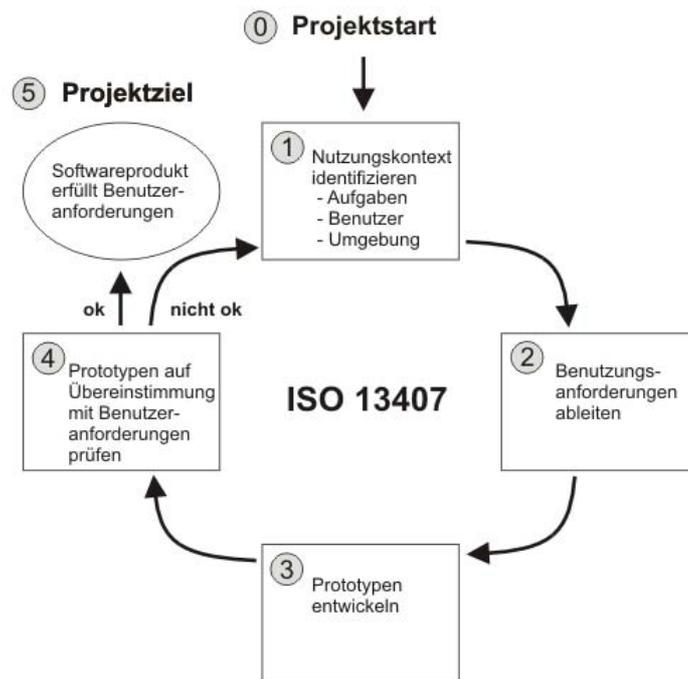


Abbildung 10: Aktivitäten des User Centred Design¹³⁶

Das entscheidende Element im Rahmen des User Centred Design ist die Festlegung der angestrebten Usability-Ziele. Ohne konkrete Anforderungen an die Usability einer Software lassen sich keine Aussagen über die aktuelle Usability treffen (siehe Abbildung 11). Die Ziele werden in einem mehrstufigen, folgende Schritte umfassenden Prozess gebildet:¹³⁷

- **Analyse der Aufgabe, der Benutzer und der Umgebung:** Es gilt herauszufinden, zu welchem Zweck ein Produkt eingesetzt wird, in welchem Ausmaß dieser Zweck momentan erreicht wird und welchen Einfluss der Prototyp darauf hat. Ziel der Aufgabenanalyse ist es, die Hauptaufgaben und ihre Häufigkeit zu bestimmen. Dies geschieht beispielsweise durch Beobachtung der Benutzer im Arbeitsumfeld, Befragung der Zielgruppe oder einer Videoanalyse des Arbeitsablaufes. Da Usability-Ziele auf die relevanten Eigenschaften der Benutzer abgestimmt sein sollen, wird ein Benutzerprofil erstellt, das bei der Entwicklung und Festsetzung der Ziele berücksichtigt werden soll.
- **Festlegung der Usability-Attribute:** Die Usability-Dimensionen, also Effizienz, Effektivität und Zufriedenheit, werden in diesem Schritt in Attribute zerlegt. Fehlerraten, Flexibilität oder Erlernbarkeit stellen mögliche Attribute dar. Entscheidend ist, dass diese Kriterien operationalisiert, das heißt in messbare Größen verwandelt, werden können.
- **Bestimmung eines kritischen Levels für die Usability-Attribute:** Für jedes Usability-Attribut wird ein kritischer Wert festgesetzt, welcher die akzeptierten Ausprägungen eines Attributs festlegt. Zum einen können absolute Grenzwerte definiert werden („*Ein Benutzer sollte die*

¹³⁶ Eigene Darstellung, teilweise übernommen aus Bevan (1999): S. 94; Geis/Hartwig (1998): S. 169.

¹³⁷ Vgl. Eichinger, Armin: Usability Engineering (2). Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-engineering2.html>>, Download am 18.01.2006.

Aufgabe mit nicht mehr als einem Fehler beenden.“), zum anderen können auch relative Werte angewendet werden („Ein Benutzer sollte mit dem Produkt X innerhalb von fünf Minuten weniger Fehler machen als mit Produkt Y.“) Die Quellen für solche kritischen Werte bilden beispielsweise frühere Usability Tests, Expertenurteile, Marktforschungsdaten und Feldstudien.

Nachdem die angestrebten Usability-Ziele definiert wurden, wird über den Einsatz der Messinstrumente entschieden. Ein Messinstrument beschreibt die Methode zur Bestimmung der Werte der Usability-Attribute. Die quantitativen Ausprägungen können sowohl subjektive, etwa Benutzerzufriedenheit, als auch objektive Werte, beispielsweise Zeitaufwand oder Fehlerhäufigkeit, darstellen. Zu den beliebtesten Messinstrumenten zählen Fragebögen, Protokollierungssoftware oder -videos und Benchmark-Aufgaben. Mit Hilfe der ausgewählten Messinstrumente können Usability-Probleme diagnostiziert werden. Sie bilden also das Herzstück im Prozess des User Centred Design (siehe dazu im speziellen Kapitel 3.5 *Usability Testing*).¹³⁸

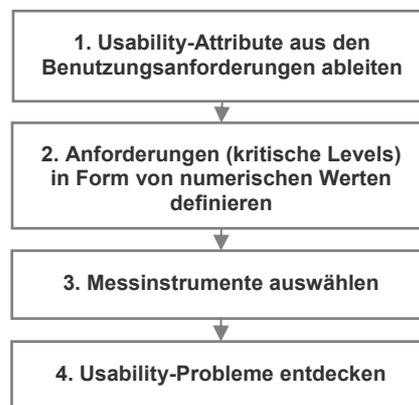


Abbildung 11: Schritt 4 im User Centred Design Prozess ¹³⁹

3.4.2 Prinzipien des User Centred Design

In diesem Abschnitt werden die wesentlichen Prinzipien des User Centred Design vorgestellt. Die Beschreibung allgemeiner Entwurfsprinzipien steht am Anfang des Kapitels. Im weiteren Verlauf wird auf internationale Usability Standards und Hersteller Styleguides eingegangen.

Prinzipien, Normen und Guidelines können Usability Testing, den wohl wichtigsten Bestandteil des User Centred Design, nicht ersetzen, sie bergen aber großes Potential, es zu ergänzen. Das Befolgen von Prinzipien und Guidelines garantiert keine makellosen Produkte, es reduziert jedoch die

¹³⁸ Vgl. Eichinger, Armin: Usability Engineering (2). Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-engineering2.html>>, Download am 18.01.2006.

¹³⁹ Eigene Darstellung, teilweise übernommen aus Eichinger, Armin: Usability Engineering. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-engineering.html>>, Download am 18.01.2006.

Wahrscheinlichkeit, dass im Zuge von Usability Tests schwerwiegende Usability Probleme auftreten.¹⁴⁰

Wesentliche Beachtung muss der Tatsache geschenkt werden, dass das Ziel einer hohen Software Usability mit anderen Zielen abgewogen werden muss. Prinzipien des User Centred Design isoliert zu betrachten, wäre nicht sinnvoll. Beispielsweise stellen Passwortabfragen und andere Formen der Authentifizierung ganz offensichtlich eine Beschränkung der Benutzerfreundlichkeit einer Software dar; dennoch sind sie zweifellos nützlich.¹⁴¹

3.4.2.1 Allgemeine Entwurfsprinzipien

Ein Entwurfsprinzip ist eine sehr allgemein formulierte Anweisung, die zumeist auf Erkenntnissen der menschlichen Lernfähigkeit und des ergonomischen Arbeitens basiert. Als Beispiel sei hier das Prinzip der Konsistenz genannt, das für das Design des User Interface wie auch für die Softwaredokumentation Gültigkeit hat. Bei der Auswahl der Worte, Formate, Grafiken und Vorgehensweisen innerhalb des Systems soll auf Konsistenz geachtet werden. Dieses Prinzip basiert auf Forschungsergebnissen, welche zeigen, dass Menschen schneller lernen und sich Neuerlerntes besser aneignen können, sofern ihre Eindrücke und ihr Handeln konsistent sind.¹⁴²

In der Literatur gibt es eine Vielzahl an Checklisten mit Entwurfsprinzipien für User Interfaces. Die Bekannteste ist jene von *Smith* und *Mosier*¹⁴³; sie war auch Ausgangspunkt für die Entwicklung der internationalen Normen, herausgegeben von der International Organization of Standardization.¹⁴⁴ Im Folgenden sind die Entwurfsprinzipien nach *Preim* wiedergegeben, eine relativ kurze Auflistung von Prinzipien, welche auch mit Aspekten aus den Grundsätzen der Dialoggestaltung nach EN ISO 9241-10 im Verhältnis stehen:

¹⁴⁰ Vgl. Dumas/Redish (1999): S. 55; Bevan, Nigel/Kirakowski, Jurek/Maissel, Jonathan (1991): What is Usability? Online im Internet. URL: <<http://www.usability.serco.com/papers/whatis92.pdf>>, Download am 02.02.2006.

¹⁴¹ Vgl. Preim (1999): S. 55.

¹⁴² Vgl. Dumas/Redish (1999): S. 53.

¹⁴³ Vgl. Smith, Sidney/Mosier, Jane (1986): Guidelines For Designing User Interface Software. Online im Internet. URL: <<http://hcibib.org/sam/>>, Download am 11.02.2006.

¹⁴⁴ Vgl. Bevan, Nigel/Kirakowski, Jurek/Maissel, Jonathan (1991): What is Usability? Online im Internet. URL: <<http://www.usability.serco.com/papers/whatis92.pdf>>, Download am 02.02.2006.

- II. Informiere dich über potentielle Benutzer und ihre Aufgaben.
- III. Hilf Benutzern, ein mentales Modell zu entwickeln.
- IV. Sprich die Sprache des Benutzers.
- V. Mach Systemzustände sichtbar und unterscheidbar.
- VI. Verdeutliche die jeweils möglichen Aktionen.
- VII. Strukturiere die Benutzungsschnittstelle.
- VIII. Stelle eine erkennbare Rückkoppelung sicher.
- IX. Gestalte die Schnittstelle adaptierbar.
- X. Kombiniere visuelle Interaktion mit sprachbasierter Interaktion.
- XI. Vermeide, dass Benutzer sich zu viele Dinge merken müssen.
- XII. Ermögliche es, Fehler zu erkennen, zu diagnostizieren und zu beheben.
- XIII. Vermeide es, den Benutzer zu überraschen.
- XIV. Beachte die wichtigsten Bedienhandlungen besonders.
- XV. Erkläre die Bedienung des Programms durch Beispiele und weniger durch Formalismen.

Abbildung 12: Entwurfsprinzipien nach Preim¹⁴⁵

3.4.2.2 Hersteller Styleguides

Hersteller Styleguides bilden Regelwerke, welche Softwareanbieter herausgegeben haben, um ein einheitliches Look & Feel ihrer Systeme zu beschreiben.¹⁴⁶ Beispiele sind Apple's Human Interface Guidelines¹⁴⁷ oder SAA Common User Access von IBM¹⁴⁸. Im Gegensatz zu Entwurfsprinzipien, sind Styleguides konkreter; sie enthalten Einzelheiten der Gestaltung und Benutzung von Bedienelementen und sind zumeist plattformspezifisch.¹⁴⁹ Ein allgemeines Entwurfsprinzip stellt ein Ziel dar; es werden aber keine Aussagen darüber getroffen, wie man das Ziel erreicht. Eine Guideline formuliert ein konkretes Ziel für spezifische Benutzer, Arbeitsumgebungen und Technologien. Für ein menügesteuertes Produkt beispielsweise könnte das Prinzip der Konsistenz in viele verschiedene Guidelines aufgespalten werden. Eine könnte sich etwa darauf beziehen, dass das Verlassen jedes einzelnen Menüs konsistent sein soll.¹⁵⁰

Hersteller Styleguides vereinen wissenschaftliche Forderungen und praktische Empfehlungen zur Gestaltung des User Interface. Der erste Teil, der generelle Entwurfsprinzipien für Schnittstellen enthält, welche von der Software Ergonomie in den letzten Jahren erarbeitet wurden, beispielsweise Einfachheit, Konsistenz und Benutzerkontrolle, wird um einen zweiten Teil ergänzt, in welchem detaillierte Angaben zum User Interface Design festgesetzt sind: Welche Interaktionselement einzuplanen sind, wie diese aussehen sollen und wie sie zu bedienen sind.¹⁵¹ Der Hauptteil umfasst

¹⁴⁵ Vgl. Preim (1999): S. 55ff.

¹⁴⁶ Vgl. Maaß (1993): S. 199.

¹⁴⁷ Vgl. Apple Computer (2005): Introduction to Apple Human Interface Guidelines. URL: <<http://developer.apple.com/documentation/UserExperience/Conceptual/OSXHIGuidelines/>>, Download am 08.02.2006.

¹⁴⁸ Vgl. Berry, R. (1988): Common User Access - A consistent and usable human-computer interface for the SAA environments. Online im Internet. URL: <<http://www.research.ibm.com/journal/sj/273/ibmsj2703E.pdf>>, Download am 08.02.2006.

¹⁴⁹ Vgl. Preim (1999): S. 55.

¹⁵⁰ Vgl. Dumas/Redish (1999): S. 53.

¹⁵¹ Vgl. Maaß (1993): S. 199.

also konkrete Richtlinien für die Verwendung von Interaktionstechniken, beispielsweise den Einsatz von Tastaturkürzel, die Gestaltung von Icons und die Platzierung von Fenstern.¹⁵²

Die genannten Regelwerke treffen ausschließlich Aussagen zur Gestaltung der Systemhandhabung. Somit konzentrieren sie sich auf nur einen Aspekt des User Centred Design. Die Aufgabenangemessenheit als Gestaltungsgrundsatz wird nicht beachtet.¹⁵³ Im Gegensatz zu den Hersteller Styleguides befassen sich nationale und internationale Normen mit diesem Gestaltungsprinzip.

3.4.2.3 Standards und Normen für HCI und Usability

Usability Standards und Normen bestimmen die für ein Softwaresystem wichtigen Eigenschaften und enthalten zudem auch Richtlinien für den Entwicklungsprozess.¹⁵⁴ Sie können, je nachdem welche Aspekte eines Softwaresystems sie primär ansprechen, in folgende Teilgruppen untergliedert werden:¹⁵⁵

- **Anwendung der Software (Use in context):** Normen zur Effektivität, Effizienz und Benutzerzufriedenheit in einem bestimmten Anwendungskontext
- **User Interface und Interaktion:** Normen zur Produktqualität
- **User Centred Design Prozess:** Normen zum Softwaregestaltungsprozess
- **Usability im Softwarelebenszyklus:** Normen zur Fähigkeit eines Softwareanbieters, Usability im gesamten Lebenszyklus einer Software zu garantieren

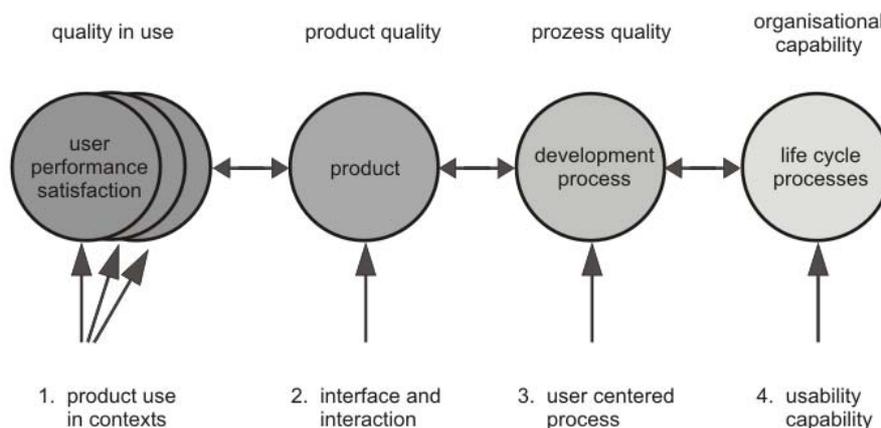


Abbildung 13: Kategorisierung der Usability Normen¹⁵⁶

¹⁵² Vgl. Preim (1999): S. 532.

¹⁵³ Vgl. Maaß (1993): S. 199.

¹⁵⁴ Vgl. Preim (1999): S. 529.

¹⁵⁵ Vgl. Serco (2001): User centred design standards. Online im Internet. URL: <<http://usabilitynet.org/trump/resources/standards.htm>>, Download am 04.02.2006.

¹⁵⁶ Grafik übernommen aus Serco (2001): User centred design standards. Online im Internet. URL: <<http://usabilitynet.org/trump/resources/standards.htm>>, Download am 04.02.2006.

Folgende Tabelle enthält eine Übersicht der internationalen Normen zur Software Usability:

Tabelle 2: Internationale Normen zur Software Usability¹⁵⁷

	Prinzipien und Empfehlungen	Spezifizierungen
Kontextuelle Anwendung	ISO/IEC 9126-1: Software Engineering - Product quality - Part 1: Quality model ISO/IEC TR 9126-4: Software Engineering - Product quality - Part 4: Quality in use metrics ISO 9241-11: Guidance on Usability	ISO 20282: Usability of everyday products
User Interface und Interaktion	ISO/IEC TR 9126-2: Software Engineering - Product quality – Part 2 External metrics ISO/IEC TR 9126-3: Software Engineering - Product quality – Part 3 Internal metrics ISO 9241: Ergonomic requirements for office work with visual display terminals. Parts 10-17 ISO 11064: Ergonomic design of control centres ISO 14915: Software ergonomics for multimedia user interfaces IEC TR 61997: Guidelines for the user interfaces in multimedia equipment for general purpose use	ISO 9241: Ergonomic requirements for office work with visual display terminals. Parts 3-9 ISO/IEC 10741-1: Dialogue interaction - Cursor control for text editing ISO/IEC 11581: Icon symbols and functions ISO 13406: Ergonomic requirements for work with visual displays based on flat panels ISO/IEC 14754: Pen-based interfaces - Common Gestures for text editing with pen-based systems ISO/IEC 18789: Information Technology - User interface for mobile tools ISO 18789: Ergonomic requirements and measurement techniques for electronic visual displays
Software-dokumentation	ISO/IEC 18019: Guidelines for the design and preparation of software user documentation	ISO/IEC 15910: Software user documentation process
Softwaregestaltungsprozess	ISO 13407: Human centred design processes for interactive systems ISO TR 16982: Usability methods supporting human centred design	ISO/IEC 14598: Information Technology - Evaluation of Software Products
Usability im Software-lebenszyklus	ISO TR 18529: Ergonomics of human system interaction – Human centred lifecycle process descriptions	
Andere	ISO 9241-1: Part 1: General Introduction ISO 9241-2: Part 2: Guidance on task requirements ISO 10075-1: Ergonomic principles related to mental workload - General terms and definitions ISO DTS 16071: Guidance on accessibility for human computer interfaces	

Für die Softwaregestaltung ist Teil 10 der Norm EN ISO 9241¹⁵⁸ von überragender Bedeutung. Die Dialoggestaltung bezieht sich auf die Interaktion des Benutzers mit dem Softwaresystem. In der EN ISO 9241-10 werden sieben übergeordnete Grundsätze für die Dialoggestaltung genannt, die bei der Entwicklung einzubeziehen sind:

¹⁵⁷ Eigene Darstellung, Daten übernommen aus Serco (2001): User centred design standards. Online im Internet. URL: <<http://usabilitynet.org/trump/resources/standards.htm>>, Download am 04.02.2006.

¹⁵⁸ Vgl. Deutsches Institut für Normung (2003): EN ISO 9241-10: 1996. S. 3ff.

- „Ein Dialog ist **aufgabenangemessen**, wenn er den Benutzer unterstützt, seine Arbeitsaufgabe effektiv und effizient zu erledigen.“¹⁵⁹ Voraussetzung für die Erfüllung des Grundsatzes der Aufgabenangemessenheit ist die Kenntnis von Benutzer und seiner Aufgabe.¹⁶⁰ So sollten in einem Formular keine mühsam auszufüllenden Pflichtangaben verlangt werden, die für den relevanten Vorgang nicht wesentlich sind.¹⁶¹
- „Ein Dialog ist **selbstbeschreibungsfähig**, wenn jeder einzelne Dialogschritt durch Rückmeldung des Dialogsystems unmittelbar verständlich ist oder dem Benutzer auf Anfrage erklärt wird.“¹⁶² Ein Benutzer sollte stets Bescheid wissen, wo er sich gerade im System befindet und was er als nächstes machen muss, um sein Arbeitsziel zu erreichen.¹⁶³
- „Ein Dialog ist **steuerbar**, wenn der Benutzer in der Lage ist, den Dialogablauf zu starten sowie seine Richtung und Geschwindigkeit zu beeinflussen, bis das Ziel erreicht ist.“¹⁶⁴ Bei einem Dateidownload soll es dem Benutzer beispielsweise möglich sein, diesen zu unterbrechen und später fortzusetzen.¹⁶⁵
- „Ein Dialog ist **erwartungskonform**, wenn er konsistent ist und den Merkmalen des Benutzers entspricht, z.B. seinen Kenntnissen aus dem Arbeitsgebiet, seiner Ausbildung und seiner Erfahrung sowie den allgemein anerkannten Konventionen.“¹⁶⁶ Ziel ist es, das Erstaunen des Benutzers zu minimieren.¹⁶⁷ Sind Wörter unterstrichen, so soll es sich bei diesen um Hypertext-Links handeln.¹⁶⁸
- „Ein Dialog ist **fehlertolerant**, wenn das beabsichtigte Arbeitsergebnis trotz erkennbar fehlerhaften Eingaben entweder mit keinen oder mit minimalem Korrekturaufwand seitens des Benutzers erreicht werden kann.“¹⁶⁹
- „Ein Dialog ist **individualisierbar**, wenn das Dialogsystem Anpassungen an die Erfordernisse der Arbeitsaufgabe sowie an die individuellen Fähigkeiten und Vorlieben des Benutzers zulässt.“¹⁷⁰
- „Ein Dialog ist **lernförderlich**, wenn er den Benutzer beim Erlernen des Dialogsystems unterstützt und anleitet.“¹⁷¹ Für die Lernförderlichkeit ist eine gute Übersicht über verfügbare Optionen, eine ausreichende Rückkoppelung des Systems und Fehlermeldungen, welche leicht interpretiert werden können, wichtig.¹⁷²

¹⁵⁹ Deutsches Institut für Normung (2003): EN ISO 9241-10: 1996. S. 4.

¹⁶⁰ Vgl. Preim (1999): S. 73.

¹⁶¹ Vgl. Wirth, Thomas (2005): Die EN ISO 9241 - 10. Online im Internet. URL: <<http://kommdesign.de/texte/din.htm>>, Download am 19.01.2006.

¹⁶² Deutsches Institut für Normung (2003): EN ISO 9241-10: 1996. S. 5.

¹⁶³ Vgl. Schneider, Wolfgang (2000): Selbstbeschreibungsfähigkeit. Online im Internet. URL: <<http://www.sozialnetz-hessen.de/ca/ph/het/Hauptpunkt/aaaaaaaaaaaaahfi/Unterpunkt/aaaaaaaaaaaaaiih/HauptframeID/aaaaaaaaaaaaaluw/HauptframeTemplate/aaaaaaaaaaaaaapq/>>, Download am 05.02.2006.

¹⁶⁴ Deutsches Institut für Normung (2003): EN ISO 9241-10: 1996. S. 6.

¹⁶⁵ Vgl. Preim (1999): S. 73; Wirth, Thomas (2005): Die EN ISO 9241 - 10. Online im Internet. URL: <<http://kommdesign.de/texte/din.htm>>, Download am 19.01.2006.

¹⁶⁶ Deutsches Institut für Normung (2003): EN ISO 9241-10: 1996. S. 6.

¹⁶⁷ Vgl. Preim (1999): S. 73.

¹⁶⁸ Vgl. Wirth, Thomas (2005): Die EN ISO 9241 - 10. Online im Internet. URL: <<http://kommdesign.de/texte/din.htm>>, Download am 19.01.2006.

¹⁶⁹ Deutsches Institut für Normung (2003): EN ISO 9241-10: 1996. S. 7.

¹⁷⁰ Ebenda: EN ISO 9241-10: 1996. S. 8.

¹⁷¹ Ebenda: EN ISO 9241-10: 1996. S. 9.

¹⁷² Vgl. Preim (1999): S. 73f.

Eine Ergänzung der Grundsätze der Dialoggestaltung findet sich in den detaillierten Normen der Teile 12 bis 17 der EN ISO 9241.

Neben den internationalen Normen und Standards der International Organization for Standardization widmet sich auch eine EU-Richtlinie dem Thema der Software Usability. Die 1990 erlassene EU-Bildschirmrichtlinie (90/270/EWG)¹⁷³ setzt Mindestvorschriften und legt die Pflichten der Arbeitgeber fest. Sie befasst sich vor allem mit der physischen Arbeitsumgebung, enthält aber auch Anforderungen betreffend der Aufgabenangemessenheit, der leichten Benutzbarkeit und der Prinzipien der Software Ergonomie, welche angewandt werden müssen. Verglichen mit den ISO Normen enthält die Richtlinie konkretere Vorschriften, welche sich allerdings hauptsächlich auf gesundheitliche Aspekte beziehen. Standards zur Effektivität und Effizienz eines Softwaresystems sind in der Richtlinie nicht festgelegt.¹⁷⁴

3.5 Usability Testing

„A usability test is a procedure for determining whether the quantitative goals defined in the usability specification have been achieved. The term “test” is a broad term that encompasses any method for assessing whether goals have been met.“¹⁷⁵

Usability Testing bezeichnet also jenes Vorgehen, während dessen geprüft wird, ob die vorab definierten Usability-Ziele erreicht wurden. „Testing“ ist dabei ein allgemeiner Name für viele verschiedene Methoden, die dabei Anwendung finden können.¹⁷⁶

Dumas und *Redish* formulieren den Begriff Usability Testing ein wenig provokant: Während sich das Testen der Funktionalität bzw. des Nutzens einer Software darauf konzentriert, ob das Produkt die gewünschten technischen Spezifikationen besitzt, rückt beim Usability Testing die Frage in den Mittelpunkt, inwieweit die Anwender die Funktionalitäten der Software überhaupt finden und mit diesen ihre Arbeit effektiv erledigen können.¹⁷⁷

¹⁷³ Vgl. EUR-Lex: Richtlinie 90/270/EWG des Rates vom 29. Mai 1990 über die Mindestvorschriften bezüglich der Sicherheit und des Gesundheitsschutzes bei der Arbeit an Bildschirmgeräten (Fünfte Einzelrichtlinie im Sinne von Artikel 16 Absatz 1 der Richtlinie 89/391/EWG). Online im Internet. URL: <<http://europa.eu.int/eur-lex/lex/LexUriServ/LexUriServ.do?uri=CELEX:31990L0270:DE:HTML>>, Download am 11.02.2006.

¹⁷⁴ Vgl. Bevan, Nigel (1995): Human-Computer Interaction Standards. Online im Internet. URL: <<http://www.usability.serco.com/papers/hcistd95.pdf>>, Download am 02.02.2006.

¹⁷⁵ Wixon/Wilson (1997): S. 669.

¹⁷⁶ Vgl. Eichinger, Armin: Usability Tests. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-tests.html>>, Download am 18.01.2006.

¹⁷⁷ Vgl. Dumas/Redish (1999): S. 4.

3.5.1 Motive für den Einsatz von Usability Testing

Oppermann¹⁷⁸ nennt drei Anlässe für den Einsatz von Evaluationsverfahren:

- **Während des Systemdesign** (auch formative Evaluation genannt¹⁷⁹): Ein entstehendes Softwareprodukt wird vom Softwarehersteller hinsichtlich Usability-Kriterien überprüft. Die Evaluation von softwareergonomischen Eigenschaften kann bereits in der Phase des Systementwurfs einsetzen. Eine Evaluation in diesem Stadium prüft, ob es sich um ein angemessenes Gestaltungskonzept handelt; sie wird mit Hilfe einer Prüfliste von angestrebten Usability-Eigenschaften durchgeführt werden. Konkreter wird die Evaluation des Prototyps ausfallen, die sich von der Evaluation eines fertigen Systems im Grunde genommen nicht unterscheidet.¹⁸⁰
- **Evaluation von Systemausprägungen** (auch als summative Evaluation bekannt¹⁸¹): Ein bestehendes System wird bewertet. Als mögliche Anwender kommen Softwarehersteller, Unternehmen, welche die Software einsetzen, oder Software-Prüfstellen in Frage. Während bei der Evaluation eines Produktkonzeptes bewertet wird, ob die softwareergonomischen Prinzipien und deren logische Konsistenz realisiert wurden, wird bei einer Produktevaluation darauf geachtet, ob diese Prinzipien durchgängig Anwendung finden.¹⁸²
- **Zum Zweck eines Systemvergleichs**: Mehrere Softwareprodukte werden im Rahmen eines Systemvergleichs bewertet, beispielsweise vor einer Kaufentscheidung. Die Anwendergruppen sind wiederum Softwareanbieter, Unternehmen als zukünftige Kunden einer der getesteten Lösungen oder Software-Prüfstellen.

3.5.2 Ziele von Usability Testing

Die Ziele der Evaluation von Software Usability stehen in einem direkten Verhältnis zu den Beziehungen zwischen den drei Faktoren Benutzer, Aufgabe und Software:¹⁸³

- **Aufgabenbewältigung**: Bei der Evaluation von Softwareprodukten interessiert vor allem, inwieweit die Aufgabenerledigung durch die Software unterstützt oder behindert wird. Es wird versucht, das Ausmaß der Unterstützung des Benutzers an der Aufgabenbewältigung über die beiden folgenden Beziehungen zu erfassen.
- **Benutzung**: Es soll herausgefunden werden, wie groß der Aufwand der Systemnutzung ist. Einflussnehmende Faktoren sind beispielsweise die Erlernbarkeit der Software und die Anpassung des Systems an Benutzereigenschaften.
- **Funktionalität**: Die Aufgabenangemessenheit der Software wird durch das Ausmaß der Unterstützung der Aufgaben durch das System bestimmt. Ob das System die Aufgaben

¹⁷⁸ Vgl. Oppermann et al. (1992): S. 7.

¹⁷⁹ Vgl. Holzinger (2001): S. 224.

¹⁸⁰ Vgl. Oppermann (1988): S. 326.

¹⁸¹ Vgl. Holzinger (2001): S. 224.

¹⁸² Vgl. Oppermann (1988): S. 327.

¹⁸³ Vgl. Oppermann et al. (1992): S. 6f.

hinreichend genau abbilden kann oder inwieweit der Benutzer das System umgestalten kann, sind Aspekte, welche für die Funktionalität entscheidend sind.

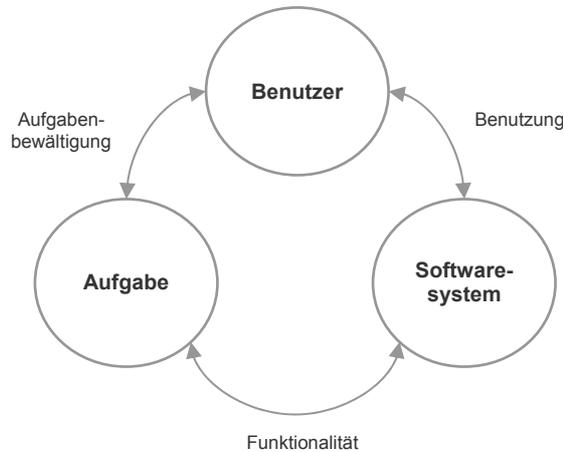


Abbildung 14: Verhältnis zwischen Software, Benutzer und Aufgabe¹⁸⁴

3.5.3 Methoden und Verfahren

In allen Phasen der Entwicklung und Nutzung von Softwaresystemen, nämlich Analyse, Design, Programmierung, Verifizierung, Benutzung, Wartung/Erweiterung und Wiederverwendung, können verschiedene Evaluierungsverfahren (siehe Abbildung 15) zum Einsatz kommen.¹⁸⁵

	Inspection Methods			Test Methods		
	Heuristic Evaluation	Cognitive Walkthrough	Action Analysis	Thinking Aloud	Field Observation	Questionnaires
Applicably in Phase	all	all	design	design	final testing	all
Required Time	low	medium	high	high	medium	low
Needed Users	none	none	none	3+	20+	30+
Required Evaluators	3+	3+	1-2	1	1+	1
Required Equipment	low	low	low	high	medium	low
Required Expertise	medium	high	high	medium	high	low
Intrusive	no	no	no	yes	yes	no

Abbildung 15: Methoden des Usability Testing¹⁸⁶

¹⁸⁴ Eigene Darstellung, teilweise übernommen aus Oppermann (1992): S. 6.

¹⁸⁵ Vgl. Stary (1994): S. 317.

¹⁸⁶ Grafik übernommen aus Holzinger (2005): S. 72.

Die Methoden des Usability Testing können nach verschiedenen Aspekten geordnet werden, etwa ob sie nach subjektiven oder objektiven Kriterien bewerten¹⁸⁷, oder ob Anwender oder Experten die Systeme evaluieren¹⁸⁸. Diese Arbeit orientiert sich an der Trennung der Methoden in experten- und benutzerzentrierte.

Evaluationsverfahren, welche die Benutzer mit einbeziehen, sind die elementarsten Methoden und unabdingbar. Es können Informationen darüber gewonnen werden, wie Menschen Systeme einsetzen und auf welche Probleme sie dabei stoßen.¹⁸⁹ Zu den bekanntesten der **benutzerzentrierten Methoden** (Test Methods) zählen Befragungen sowie Evaluationsmethoden, welche experimentell im Labor oder als Field Observations durchgeführt werden können. Diese Verfahren umfassen beispielsweise solche, bei denen bestimmte Aufgaben anhand objektiv quantifizierbarer Messgrößen bewertet werden (z.B. Zeit für die Erledigung einer bestimmten Aufgabe), und wiederum andere, welche verschiedene Systeme anhand von standardisierten Aufgaben testen, sogenannte Benchmark-Tests.¹⁹⁰ Eine subjektive Bewertung stellen jene Evaluationen dar, welche von den Testpersonen lautes Denken (Thinking Aloud) verlangen.¹⁹¹

Expertenzentrierte Verfahren sollten immer in Kombination mit benutzerzentrierten angewendet werden. Die am häufigsten eingesetzten Methoden der Expertenevaluation (Inspection Methods) sind die Heuristische Evaluation, der Cognitive Walkthrough sowie die Action Analysis (auch bekannt unter Keystroke-Level Analysis).¹⁹²

Im Folgenden wird auf ausgewählte Methoden aus dem Kreis jener, welche den Benutzer bewerten lassen, eingegangen.

3.5.3.1 Task Based Testing

Task Based Testing, in Form von Laborexperimenten oder Feldstudien, entspricht dem klassischen Bild von Usability Tests: Die Testpersonen bearbeiten Aufgaben am System. Diese Aufgabenerledigung wird bewertet und evaluiert. Wie solch ein Test vor sich geht, wird in Abbildung 16 veranschaulicht.

¹⁸⁷ Vgl. Oppermann (1988): S. 327ff.

¹⁸⁸ Vgl. Holzinger (2005): S. 72.

¹⁸⁹ Vgl. Holzinger (2005): S. 73.

¹⁹⁰ Vgl. Oppermann et al. (1992): S. 10ff; Holzinger (2005): S. 73f.

¹⁹¹ Vgl. Oppermann (1988): S. 328.

¹⁹² Vgl. Holzinger (2005): S. 72f.

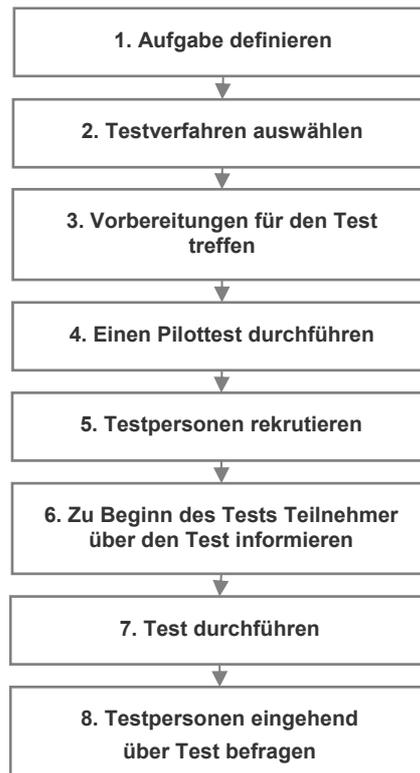


Abbildung 16: Task Based Testing¹⁹³

Grundlage für den ersten Schritt der **Definition einer Aufgabe** bilden die relevanten Benutzereigenschaften sowie die Ergebnisse der Aufgabenanalyse.¹⁹⁴ Benutzer- und aufgabenspezifische Aspekte wurden bereits in der zweiten Phase des User Centred Design Prozesses betrachtet. Aufgaben können so gestaltet sein, dass den Testpersonen das gewünschte Resultat gezeigt wird und sie aufgefordert werden, dieses zu reproduzieren. *Wixon* und *Wilson*¹⁹⁵ bezeichnen solche Tests als „results-based tests“. Im Rahmen von „process-based tests“ hingegen wird eine Aufgabensequenz definiert, welche von den Testpersonen abgearbeitet werden soll.

Die zum Einsatz kommenden **Methoden des Usability Testing** werden ebenfalls auf Grundlage der Usability-Ziele ausgewählt.¹⁹⁶ Man unterscheidet grundsätzlich zwischen zwei Messgrößen, welche bei einem Usability Test erhoben werden können: (1) Indikatoren, welche das Verhalten direkt abbilden (z.B. Zeit, um eine Aufgabe zu erledigen, Anzahl an Fehlern), (2) Indikatoren, welche das Verhalten reflektierend abbilden (z.B. spontane Antworten der Testpersonen). Innerhalb eines Usability Tests kommen teilweise auch mehrere Methoden zum Einsatz. Nachdem die Testpersonen beispielsweise eine Aufgabe mit Hilfe der Software erledigt haben, wird eine ähnliche Aufgabe gestellt, wobei dann Thinking Aloud als Methode angewandt wird.¹⁹⁷

¹⁹³ Eigene Darstellung, teilweise übernommen aus Eichinger, Armin: Usability Tests. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-tests.html>>, Download am 18.01.2006.

¹⁹⁴ Vgl. Eichinger, Armin: Usability Tests. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-tests.html>>, Download am 18.01.2006.

¹⁹⁵ Wixon/Wilson (1997): S. 670.

¹⁹⁶ Vgl. Eichinger, Armin: Usability Tests. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-tests.html>>, Download am 18.01.2006.

¹⁹⁷ Vgl. Wixon/Wilson (1997): S. 671.

Die notwendigen **Vorbereitungen** für einen Usability Test umfassen beispielsweise die Anfertigung von Fragebögen und Formularen.¹⁹⁸ Der **Pilottest** sollte so früh im Voraus abgehalten werden, dass zwischen Pilottest und dem tatsächlichen Test genügend Zeit bleibt, um nötige Veränderungen an den Testmaterialien zu bewerkstelligen.¹⁹⁹

Welche Personen am Test teilnehmen sollten, ergibt sich aus der Benutzercharakterisierung im Rahmen des User Centred Design. Für eine Field Observation benötigt man relativ wenige **Testpersonen**; fünf bis zehn reichen etwa schon aus um die Software Usability umfassend beurteilen zu können.²⁰⁰

Der Testleiter sollte die Teilnehmer am Anfang des Tests **über Dauer und Zweck in Kenntnis setzen**. Der Ablauf des Tests sollte für die Teilnehmer vorhersehbar sein. Ein mitunter wichtiger Hinweis an die Teilnehmer ist, dass ihr Verhalten nur hinsichtlich der Produkteigenschaften interpretiert wird und nicht Rückschlüsse über ihre Fähigkeiten gemacht werden.²⁰¹ **Nach der Durchführung des Tests** können **Fragen an die Testteilnehmer** hinsichtlich subjektiver Meßgrößen (Zufriedenheit, erster Eindruck, etc.) gestellt werden.

3.5.3.2 Thinking Aloud

Thinking Aloud ermöglicht herauszufinden, wie der Benutzer mit dem Produkt umgeht und welche Überlegungen er bei der Benutzung anstellt. Während des Testdurchlaufs führt die Testperson eine Aufgabe durch. Parallel zur Interaktion mit der Software soll die Testperson alles, was sie denkt und macht, verbal äußern. Falls Arbeitsschritte, die vom System bei der Aufgabenerledigung vorgegeben werden, von den durch die Testperson erwarteten Schritten abweichen, wird dies mit Hilfe von Thinking Aloud offensichtlich. Der wesentliche Nutzen dieser Methode ist also ein vertieftes Verständnis vom mentalen Modell des Benutzers und dessen Umgang mit dem Produkt. Thinking Aloud kann leicht und kostengünstig zu jedem Zeitpunkt im Entwicklungsprozess eingesetzt werden. Verzerrend wirken kann mitunter das Problem der sozialen Erwünschtheit; wenn Testpersonen meinen, bestimmten Erwartungen entsprechen zu müssen.²⁰²

3.5.3.3 Benutzerbefragung

Befragungen, in Form von qualitativen Interviews oder standardisierten Fragebögen²⁰³, welche von den Testpersonen auszufüllen sind, stellen eine indirekte Evaluationsmethode dar. Denn es wird nicht das User Interface an sich bewertet; es werden ausschließlich Meinungen und Einstellungen der

¹⁹⁸ Vgl. Wixon/Wilson (1997): S. 671f.

¹⁹⁹ Vgl. ebenda: S. 672.

²⁰⁰ Vgl. ebenda.

²⁰¹ Vgl. Eichinger, Armin: Usability Tests. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-tests.html>>, Download am 18.01.2006.

²⁰² Vgl. Hunkirchen, Peter (2005): Bitte laut denken: "Thinking Aloud". Online im Internet. URL: <<http://www.fit-fuer-usability.de/1x1/messen/thinking.html>>, Download am 02.02.2006; Fachrichtung Informationswissenschaft Saarbrücken (2006): Arbeitsbereich Usability Engineering. Methoden und Verfahren. Die Nutzerzentrierten Methoden. Online im Internet. URL: <http://usability.is.uni-sb.de/methoden/nutzer_methoden.php>, Download am 02.02.2006.

²⁰³ Vgl. Fachrichtung Informationswissenschaft Saarbrücken (2006): Arbeitsbereich Usability Engineering. Methoden und Verfahren. Die Nutzerzentrierten Methoden. Online im Internet. URL: <http://usability.is.uni-sb.de/methoden/nutzer_methoden.php>, Download am 02.02.2006.

Benutzer zum User Interface gesammelt.²⁰⁴ Diese Fragen beziehen sich auf bestimmte Systemeigenschaften. Der Benutzer antwortet auf der Grundlage seiner Erfahrungen mit der Software.²⁰⁵

Als Vorteile einer Befragung gelten der im Vergleich zu den anderen Testmethoden geringe Aufwand in der Durchführung, die Anwendbarkeit in allen Lebenszyklusphasen des Softwareprodukts sowie die Zweckmäßigkeit für das Auffinden unstrukturierter Probleme. Nachteile können daraus resultieren, dass sie anfällig für Übertreibungen sind, dass sie suggestive Fragestellungen enthalten können, und dass sie eine große Datenmenge produzieren, was den Aufwand der Auswertung in die Höhe treibt.²⁰⁶

²⁰⁴ Vgl. Holzinger (2005): S. 74.

²⁰⁵ Vgl. Oppermann et al. (1992): S. 10.

²⁰⁶ Vgl. ebenda.

4 Entwicklung des Usability Tests für ERP-Systeme (UTE)

Das Interesse am Entwurf eines ERP-spezifischen Usability Tests kam auf, als die Leopold-Franzens-Universität Innsbruck zwei verschiedene ERP-Systeme, SAP R/3 und Semiramis 2 bzw. 4.1, in der Hochschullehre einsetzte; beide Systeme werden bis dato eingesetzt. Die intensive Beschäftigung mit den genannten ERP-Systemen warf die Frage auf, welches System eine qualitativ höhere Software Usability aufweist.

Das Testumfeld war bekannt: Die Testpersonen würden aus jenen Studierenden rekrutiert werden, welche im Rahmen von Lehrveranstaltungen an der Universität Erfahrungen an einem der beiden Systeme sammeln konnten. Bei den Testpersonen würde es sich also nicht um Neulinge am System handeln, sondern um Benutzer, welche zumindest einen guten Überblick über die Systemfunktionalitäten vorweisen können. Auf Grund der Tatsache, dass die ERP-spezifischen Lehrveranstaltungen an der Universität Innsbruck versuchen, die betrieblichen Prozesse so umfassend wie möglich im System abzubilden und eine Einführung in die wesentlichsten Module / Frameworks der Software zu geben, kann kein Anspruch darauf erhoben werden, dass die Studierende ein detailliertes Wissen in den einzelnen Modulen mit sich bringen. Aber gerade dieser Aspekt birgt enorme Chancen für das Usability Testing von ERP-Software: Der Überblick ermöglicht es den Testpersonen, festzustellen, ob software-ergonomische Prinzipien durchgängig, über das ganze System hinweg, beachtet wurden. In experimentellen Testanordnungen oder Feldstudien zur Gebrauchstauglichkeit von ERP-Systemen wäre dies nur sehr eingeschränkt möglich, am ehesten in kleinen bzw. mittelständischen Betrieben, in welchen der einzelne Mitarbeiter viele Rollen einnimmt und an einem relativ großen Anteil der gesamten betrieblichen Prozesse mitwirkt.

Ein in der Durchführung kostengünstiger und leicht handhabbarer Usability Test sollte geschaffen werden, um einen Vergleich zwischen verschiedenen ERP-Systemen zu ermöglichen. Die Methode der Befragung mittels standardisiertem Fragebogen stellte ein geeignetes Mittel dar, um diesen Aspekten gerecht zu werden.

4.1 Anforderungen

Für die Entwicklung des Fragebogens sollten sowohl die Anforderungen an den Fragebogen als auch jene an die Ergebnisqualität berücksichtigt werden.

4.1.1 Anforderungen an den Fragebogen

Es sollte ein Fragebogen entwickelt werden, mit dem die Usability von ERP-Systemen bewertet und verglichen werden kann. Der Produktvergleich der eingesetzten ERP-Systeme war das angestrebte Ziel des Usability Tests.

Folgende Anforderungen gelten für die Entwicklung eines Fragebogens zur Evaluierung von Software Usability:²⁰⁷

- **Anonymität:** Die Benutzerbefragung sollte anonym sein. Der Schutz der Anonymität ermuntert Testpersonen, Nutzungsprobleme zu beschreiben. Eine Kritik der Software sollte nicht für Rückschlüsse auf den persönlichen Wissensstand verwendet werden können, was die Ergebnisse der Befragung beeinflussen könnte.
- **Befragung ohne Beteiligung von Experten:** Der Fragebogen soll an Benutzerarbeitsplätzen ohne Beteiligung von Experten nutzbar sein, um Antworten im Sinne sozialer Erwünschtheit ausschließen zu können.
- **Akzeptanz:** Das Verhältnis zwischen Aufwand und Nutzen soll ein vernünftiges Maß annehmen. Nur so kann eine Akzeptanz bei den befragten Testpersonen erwartet werden.

4.1.2 Anforderungen an die Ergebnisqualität

Der wichtigste Anspruch hinsichtlich der Ergebnisqualität ist die **Verständlichkeit der Usability-Attribute** für die Benutzer. Bei der Entwicklung des Fragebogens muss darauf geachtet werden, dass die Items des Fragebogens zu den Untersuchungsbereichen inhaltlich passen und für Benutzer verständlich sind.²⁰⁸

Zur **Bestimmung der psychometrischen Qualität** eines Benutzerfragebogens für Software muss angemerkt werden, dass die klassische Testtheorie im Grunde genommen nicht anwendbar ist. Zeitliche Stabilität ist bei der Messung von Usability-Attributen kaum gegeben. Die Bewertung dieser Attribute ist stark kontextabhängig.²⁰⁹ Ein psychometrisch zu bestimmendes Qualitätskriterium, beispielsweise die Retest-Reliabilität einer Messung, ist folglich nicht anwendbar. Ein Fragebogen sollte aber bei der Aufdeckung von Usability-Problemen dennoch ein gewisses Maß an Reliabilität erfüllen: Bei wiederholter Anwendung in einem bestimmten Kontext, der durch eine Kombination aus Benutzer, Aufgabe und Softwaresystem bestimmt wird, sollte er stets dieselben Probleme identifizieren.²¹⁰

²⁰⁷ Vgl. Dzida, Wolfgang et al. (2000): Gebrauchstauglichkeit von Software. ErgoNorm: Ein Verfahren zur Konformitätsprüfung von Software auf der Grundlage von DIN EN ISO 9241 Teile 10 und 11. Online im Internet. URL: <http://www.ergonomic.de/files/abschlussbericht_de_lang.pdf>, Download am 10.02.2006.

²⁰⁸ Vgl. ebenda.

²⁰⁹ Vgl. Deutsches Institut für Normung (2003): EN ISO 9241-11: 1998. S. 3.

²¹⁰ Vgl. Dzida, Wolfgang et al. (2000): Gebrauchstauglichkeit von Software. ErgoNorm: Ein Verfahren zur Konformitätsprüfung von Software auf der Grundlage von DIN EN ISO 9241 Teile 10 und 11. Online im Internet. URL: <http://www.ergonomic.de/files/abschlussbericht_de_lang.pdf>, Download am 10.02.2006.

4.2 Fragebogenentwicklung

Bei der Entwicklung des Fragebogens wurde auf Fragen aus bewährten Fragebögen zurückgegriffen, nicht zuletzt deshalb, weil diese bereits auf Inhaltsvalidität und Verständlichkeit durch den Benutzer überprüft wurden.²¹¹

Die Fragebögen, darunter SUMI (Software Usability Measurement Inventory), SUS (System Usability Scale), ErgoNorm und IsoMetrics, werden im Folgenden vorgestellt. Anschließend wird auf den neu entwickelten Usability Test eingegangen, der hauptsächlich dem Produktvergleich zwischen ERP-Systemen dient.

4.2.1 Analyse bestehender Fragebögen

An der University of Cork, Irland, entwickelt, werden für den **SUMI** drei Einsatzgebiete empfohlen: (1) Bewertung von Systemen im Entwicklungsstadium, (2) Produktvergleich und (3) Formulierung von Gestaltungszielen für die Weiterentwicklung einer Software.²¹² Der vollständig standardisierte Fragebogen enthält 50 Items, welche fünf Subskalen zugeordnet sind. Die Subskalen „Effizienz“, „Affekt“, „Hilfe und Unterstützung“, „Kontrollierbarkeit“ und „Erlernbarkeit“ enthalten jeweils zehn Items. Die Items sind dreistufig mit den Antwortmöglichkeiten „Stimme zu“, „Weiß nicht“ und „Stimme nicht zu“. Zusätzlich gibt es eine Global-Skala, welche sich aus 25 der 50 Items zusammensetzt und das Konstrukt Usability am besten repräsentiert.²¹³

Die Validität der Bewertungen durch SUMI wurde in Feld- und Laboruntersuchungen sowie in Fallstudien getestet. Es zeigt sich, dass der Fragebogen in der Lage ist, zwischen verschiedenen Textverarbeitungs- und Datensystemen mit unterschiedlicher ergonomischer Qualität zu unterscheiden.²¹⁴

Darüber, wie lange der Benutzer mit dem Ausfüllen des Fragebogens beschäftigt ist, findet man unterschiedliche Angaben. Diese reichen von drei²¹⁵ bis zehn²¹⁶ Minuten. Zwölf Testpersonen reichen aus, um zuverlässige Ergebnisse zu erhalten. Eine Studie, welche einen Systemvergleich zum Inhalt hat, soll mindestens 30 Fragebögen je Untersuchungsgruppe auswerten.²¹⁷

²¹¹ Vgl. Dzida, Wolfgang et al. (2000): Gebrauchstauglichkeit von Software. ErgoNorm: Ein Verfahren zur Konformitätsprüfung von Software auf der Grundlage von DIN EN ISO 9241 Teile 10 und 11. Online im Internet. URL: <http://www.ergonomic.de/files/abschlussbericht_de_lang.pdf>, Download am 10.02.2006.

²¹² Vgl. SUMI: What is SUMI? Online im Internet. URL: <<http://www.ucc.ie/hfrg/questionnaires/sumi/whatis.html>>, Download am 11.02.2006.

²¹³ Vgl. Hamburg/Gediga/Hassenzahl (2003): S. 179.

²¹⁴ Vgl. SUMI: Validity of SUMI. Online im Internet. URL: <<http://www.ucc.ie/hfrg/questionnaires/sumi/sumipapp.html#validity>>, Download am 11.02.2006.

²¹⁵ Vgl. SUMI: What is SUMI? Online im Internet. URL: <<http://www.ucc.ie/hfrg/questionnaires/sumi/whatis.html>>, Download am 11.02.2006.

²¹⁶ Vgl. Hamburg/Gediga/Hassenzahl (2003): S. 180.

²¹⁷ Vgl. SUMI: What is SUMI? Online im Internet. URL: <<http://www.ucc.ie/hfrg/questionnaires/sumi/whatis.html>>, Download am 11.02.2006.

Der **SUS** trägt der Tatsache Rechnung, dass es für Unternehmen als Kunden von Software Lösungen oft weder kosteneffizient noch praktisch ist, einen aufwändigen Usability Test zum Zwecke des Vergleichs zwischen Softwarealternativen durchzuführen. Alles was benötigt wird, ist ein genereller Eindruck der Software Usability der verschiedenen Systeme. Der Fragebogen enthält zehn Items, die auf einer fünfstufigen Likert-Skala bewertet werden. Bei der Entwicklung des Fragebogens wurden die Items aus einem Pool von insgesamt 50 ausgewählt, wobei jene herangezogen wurden, welche in einem Pretest die eindeutigsten Antworten erhielten. Die Statements decken eine Vielzahl an Usability-Aspekten ab, beispielsweise den Bedarf an Support und Schulungen sowie die Komplexität des Systems (siehe Abbildung 17). Das Evaluationsverfahren nach SUS führt zu einem einzigen Wert, der die generelle Usability des Systems ausdrückt. Das Ergebnis der Befragung mit SUS korreliert mit jenem, das mit Hilfe des SUMI Fragebogens erzielt wird.²¹⁸

- I. I think that I would like to use this system frequently
- II. I found the system unnecessarily complex
- III. I thought the system was easy to use
- IV. I think that I would need the support of a technical person to be able to use this system
- V. I found the various functions in this system were well integrated
- VI. I thought there was too much inconsistency in this system
- VII. I would imagine that most people would learn to use this system very quickly
- VIII. I found the system very cumbersome to use
- IX. I felt very confident using the system
- X. I needed to learn a lot of things before I could get going with this system

Abbildung 17: SUS-Items²¹⁹

Das **ErgoNorm**-Prüfverfahren setzt sich die Normkonformitätsprüfung nach DIN EN ISO 9241-10 und 9241-11 zum Ziel. Die Befragung mittels Fragebogen im Rahmen dieses Evaluationsverfahrens soll Auskunft darüber geben, wie die Effektivität und Effizienz der Software von den Benutzern beurteilt wird. Diese Beurteilungen werden als Indikatoren für die subjektive Zufriedenstellung angesehen. Im Rahmen der Entwicklung des Fragebogens wurden Experten auf dem Gebiet der Software Ergonomie gebeten, Items zu den Dialogprinzipien der DIN EN ISO 9241-10 zu generieren. Der erste Itempool bestand aus insgesamt 192 Items; schließlich wurden 27 Fragen ausgewählt, welche den tatsächlichen Fragebogen bilden sollten. Zuerst wird die Testperson gebeten, die Aufgabe zu beschreiben, an welche sie während des Ausfüllens des Fragebogens denkt. Bei jeder der 27 Fragen kann die Testperson vorab entscheiden, ob diese Frage für die von ihr beschriebene Aufgabe überhaupt relevant ist. Stellt die Testperson für sie bedeutsame Probleme fest, hat sie die Möglichkeit, die Checkbox „Ich empfinde dies als sehr störend“ anzuhaken (siehe Abbildung 18).²²⁰

²¹⁸ Vgl. Brooke, John: A quick and dirty usability scale. Online im Internet. URL:

<<http://www.usability.serco.com/trump/documents/Suschapt.doc>>, Download am 11.02.2006.

²¹⁹ Eigene Darstellung, Inhalte übernommen aus Brooke, John: A quick and dirty usability scale. Online im Internet. URL:

<<http://www.usability.serco.com/trump/documents/Suschapt.doc>>, Download am 11.02.2006.

²²⁰ Vgl. Dzida, Wolfgang et al. (2000): Gebrauchstauglichkeit von Software. ErgoNorm: Ein Verfahren zur Konformitätsprüfung von Software auf der Grundlage von DIN EN ISO 9241 Teile 10 und 11. Online im Internet. URL: <http://www.ergonomic.de/files/abschlussbericht_de_lang.pdf>, Download am 10.02.2006.

2. Müssen Sie Eingaben oder Dialogschritte machen, die eigentlich überflüssig wären?

ja nein Frage trifft nicht zu

wenn "ja":

Bitte benennen Sie die in ihren Augen überflüssigen Eingaben und Dialogschritte.

.....

Ich empfinde dies als sehr störend

Abbildung 18: ErgoNorm-Fragebogen²²¹

Ebenso wie das ErgoNorm-Verfahren strebt auch der **IsoMetrics**-Fragebogen an, die zu untersuchende Software auf der Basis der DIN EN ISO 9241-10 zu evaluieren. So umfasst auch dieser Fragebogen sieben Subskalen, welche den Gestaltungsgrundsätzen entsprechen. Insgesamt müssen von den Testpersonen 75 Items per Ratingskala bewertet werden. Eine zusätzliche Ratingskala gibt den Testpersonen die Möglichkeit, die Wichtigkeit dieses Aspekts am Gesamteindruck der Software zu beschreiben. Außerdem werden die Testpersonen aufgefordert, konkrete Beispiele zu nennen, welche mit diesem Item im Zusammenhang stehen (siehe Abbildung 19). Das Ausfüllen der hier vorgestellten Langversion benötigt etwa zwei Stunden.²²²

	stimmt nicht	stimmt wenig	stimmt mittelmäßig	stimmt ziemlich	stimmt sehr	Keine Angabe
A.8 Es müssen zuviele Eingabeschritte für die Bearbeitung mancher Aufgaben durchgeführt werden.	1	2	3	4	5	

	nicht wichtig	wenig wichtig	mittelmäßig wichtig	ziemlich wichtig	sehr wichtig	Keine Angabe
Wie wichtig ist dieser Aspekt für Ihren Gesamteindruck von der Software?	1	2	3	4	5	

Können Sie konkrete Beispiele nennen, bei denen Sie dieser Aussage nicht zustimmen können?

Abbildung 19: IsoMetrics-Fragebogen²²³

²²¹ Grafik übernommen aus Dzida, Wolfgang et al. (2000): Gebrauchstauglichkeit von Software. ErgoNorm: Ein Verfahren zur Konformitätsprüfung von Software auf der Grundlage von DIN EN ISO 9241 Teile 10 und 11. Online im Internet. URL: <http://www.ergonomic.de/files/abschlussbericht_de_lang.pdf>, Download am 10.02.2006.

²²² Vgl. Hamborg/Gediga/Hassenzahl (2003): S. 180ff.

²²³ Grafik übernommen aus Willumeit, Heinz/Hamborg, Kai/Geidga, Günther (1997): IsoMetrics. Fragebogen zur Evaluation von grafischen Benutzungsschnittstellen. (Lang-Version). Online im Internet. URL: <<http://www.isometrics.uni-osnabrueck.de/boegen/www/Isometrl.ps>>, Download am 11.02.2006.

Mit Hilfe des IsoMetrics-Fragebogens können Hinweise auf Schwachstellen der Software aus Benutzersicht sowie auf die Gewichtung der Items durch die Benutzer gewonnen werden. Studien konnten zeigen, dass mit diesem Verfahren zwischen Systemen unterschiedlicher softwareergonomischer Qualität differenziert werden kann.²²⁴

4.2.2 Entwicklung eines Fragebogens für die Evaluation von ERP-Software

Im entworfenen Fragebogen zum Usability Testing von ERP-Systemen sind die Items, in fünf Subskalen, so genannte Untersuchungsbereiche, aufgeteilt. Es sind dies **Emotion**, **Softwaresteuerung**, **Effizienz**, **Systemunterstützung** und **Erlernbarkeit**. Jeder dieser Untersuchungsbereiche umfasst eine Reihe von Items, welche auf einer fünfstufigen Likert-Skala von der Testperson bewertet werden müssen. Zusätzlich zur Bewertung des Items wird die Testperson danach gefragt, inwieweit sie diese im Item angesprochene Systemeigenschaft als wichtig empfindet (siehe Abbildung 20).

Mir macht die Arbeit mit der Software Spaß	Bewertung 1 2 3 4 5	Subjektive Wichtigkeit 1 2 3 4 5
	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>
	1 = Stimme voll und ganz zu 2 = Stimme zu 3 = Teils/teils 4 = Stimme nicht zu 5 = Stimme überhaupt nicht zu	1 = Sehr wichtig 2 = Wichtig 3 = Unentschlossen 4 = Unwichtig 5 = Absolut unwichtig

Abbildung 20: Beispiel für ein Item aus dem Fragebogen²²⁵

Der Fragebogen umfasst zehn einführende Fragen zur Charakterisierung der Testpersonen und 53 Statements, welche bewertet und nach ihrer subjektiven Wichtigkeit beurteilt werden müssen. Mit der Ausnahme einer einzigen Frage handelt es sich um einen vollständig standardisierten Fragebogen. Der Fragebogen ist im Anhang A dieser Arbeit vollständig abgebildet.

Die Frage nach der subjektiven Wichtigkeit der einzelnen Items kann zukünftig dazu verwendet werden, jene Items herauszufiltern, welche vom Benutzer als besonders wichtig empfunden werden, bzw. eine Gewichtung der einzelnen Items innerhalb eines Untersuchungsbereiches vorzunehmen. Dies eröffnet in der Folge die Möglichkeit, einerseits eine Kurzversion des Fragebogens zu entwickeln und andererseits die Aussagequalität der Untersuchung zu optimieren.

²²⁴ Vgl. Hamborg/Gediga/Hassenzahl (2003): S. 181.

²²⁵ Eigene Darstellung.

4.3 Datenerhebung

Die Datenerhebung folgt der Methodik der schriftlichen Befragung mittels standardisiertem Fragebogen. Alternativ zum Paper-and-Pen Test kann die Befragung auch online durchgeführt werden. Das Ausfüllen des Tests benötigt etwa 20 Minuten, wobei kein Zeitdruck ausgeübt wird.

5 Praxisanwendung des UTE

Das Usability-Testverfahren zur Evaluierung von ERP-Systemen wurde bislang ausschließlich an der Universität Innsbruck angewandt, wobei eine Ausweitung auf andere Bildungsinstitutionen und Unternehmungen geplant ist.

Zunächst wird die praktische Vorgehensweise der Untersuchung erläutert. Dabei wird auf die Testpersonen und das Testumfeld eingegangen. Anschließend werden die Ergebnisse mit signifikanten Unterschieden zwischen den zwei ERP-Systemen präsentiert und die Rückschlüsse kurz erörtert. Die kompletten Ergebnisse der Untersuchung sind in den Anhängen B, C und D dieser Arbeit zu finden.

5.1 Vorgehen

Die Erhebung der Daten erfolgte in drei aufeinander folgenden Semestern im Rahmen der Hochschullehre an der Leopold-Franzens-Universität Innsbruck. Der Test kam im Wintersemester 2004-05 erstmalig zum Einsatz. Im Sommersemester 2005 und im Wintersemester 2005-06 wurde die Befragung fortgeführt.

Im ersten Test setzten sich die Testpersonen aus Studierenden der Kurse „Unternehmensplanspiel (UPEK) mit Semiramis“ und „Integriertes Management mit SAP R/3“ zusammen. Die beiden folgenden Erhebungen wurden auch auf den im Sommersemester 2005 neu angebotenen Kurs „Integriertes Management mit Semiramis“ ausgedehnt.

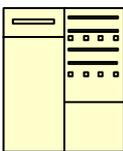
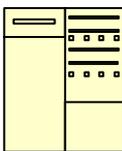
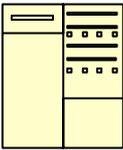
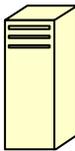
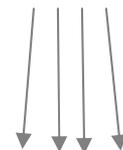
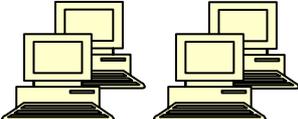
Der Test wurde jeweils am Ende des Semesters durchgeführt. Die Testpersonen hatten zu diesem Zeitpunkt einen Kurs besucht, der fünf Semesterwochenstunden (oder zehn ECTS credits, European Credits Transfer System) umfasste und die intensive Arbeit am jeweiligen ERP-System beinhaltete. Die Kursziele deckten sich insoweit, als dass in der Schulung an beiden Systemen die wesentlichsten betrieblichen Prozesse, beispielsweise Beschaffung, Vertrieb und Lagerlogistik, mit der Software abgebildet wurden. In der Lehrveranstaltung „Integriertes Management mit SAP R/3“ wurde zudem ein Einblick in die Kostenrechnungsbereiche gewährt, wogegen im Kurs „Integriertes Management mit Semiramis“ die aus hardwaretechnischer Sicht ressourcenbelastenden Bereiche Produktionsplanung, Produktion und Workflow-Management hinzukamen. Inwieweit dies die Bewertungen, vor allem im Untersuchungsbereich Effizienz, beeinflusst, ist noch nicht klar. Eine Untersuchung in diese Richtung ist für das Sommersemester 2006 in Planung. Der entwickelte Test zur Evaluierung von ERP Software Usability wäre aber selbst dann geeignet, wenn verschiedene Kursinhalte präsentiert werden würden. Schließlich ist die Bewertung der verschiedenen Statements kaum von der Kenntnis bestimmter Module / Frameworks in den beiden Systemen abhängig. Es wird aber angenommen, dass sich die Vergleichbarkeit der Ergebnisse der beiden Testgruppen erhöht, wenn sie einen beinahe identischen Wissensstand bezüglich der Geschäftsprozessabbildung in einem ERP-System mitbringen.

Über die drei Semester hinweg brachte die Befragung 197 ausgefüllte Fragebögen ein; 116 von Semiramis- und 91 von SAP-Anwendern. Es ist darauf hinzuweisen, dass ab dem Wintersemester 2005-06 Semiramis 4 zum Einsatz kam. In den beiden ersten Test-Semestern wurde Semiramis 2 in der Hochschullehre eingesetzt. Deshalb findet sich neben einer Gesamtauswertung der Testergebnisse der drei Semester, eine Aufspaltung der Ergebnisse für Wintersemester 2005-06 und die ersten beiden Test-Semester (siehe Anhang B, C und D). Für die ersten beiden Untersuchungssemester liegen 68 Fragebögen für die Software Semiramis und 45 für SAP vor. Im Wintersemester 2005-06 konnten 48 Fragebögen von Semiramis-Anwendern und 36 von SAP-Benutzern gesammelt werden.

5.2 Systemumfeld

Das Systemumfeld der Universität Innsbruck ist im Folgenden dargestellt:

Tabelle 3: Systemumfeld

Hersteller	SAP AG	Semiramis Software AG (KTW Group)	
System	SAP R/3 R4.6C	Semiramis 4.1	Semiramis 2 (V1R2M1)
Spezielle Client Installation für UI notwendig	Ja	Nein	Nein
Benutzeroberfläche	SAP GUI 6.40 (WS05) SAP GUI 6.20 (WS04, SS05)	MS Internet Explorer 6.0 (HTA-Modus)	MS Internet Explorer 6.0 (HTA-Modus)
	Wintersemester 2005 Sommersemester 2005 Wintersemester 2004	Wintersemester 2005 - -	- Sommersemester 2005 Wintersemester 2004
Architektur	Dreistufig	Dreistufig	Zweistufig
Datenbankserver			
Applikationsserver			
Client			

5.3 Testergebnisse

Im Folgenden wird auf jene Items eingegangen, bei welchen Semiramis- und SAP-Anwender signifikant unterschiedlich antworteten. Die Ergebnisse sind nach den bereits bekannten Subskalen des Fragebogens gegliedert.

Im Anhang finden sich weitere Testergebnisse. Dort findet man zunächst die Auswertungen der Fragen zum User Profil. In diesem Bereich konnten keine signifikanten Unterschiede zwischen den Testgruppen festgestellt werden. Anschließend sind die Mittelwerte der Bewertungsverteilungen sowie der subjektiven Wichtigkeit in tabellarischer Form wiedergegeben (siehe Tabellen 4 und 5).²²⁶ Dabei stellen niedrigere Werte grundsätzlich das bessere Ergebnis dar (Skala von 1 bis 5). Ebenso werden Mediane und Modi für alle abgefragten Items aufgelistet (siehe Tabelle 6). Die wissenschaftlich relevanteste Untersuchung ist die Feststellung von signifikanten Unterschieden bei der Bewertung der einzelnen Items. Ob Unterschiede in den Bewertungsverteilungen zwischen SAP- und Semiramis-Anwendern bestehen, zeigt Tabelle 7 klar und deutlich.

²²⁶ Ob eine Likert-Skala ein ordinal- oder intervallskaliertes Merkmal darstellt, ist strittig. Die übliche Forschungspraxis verzichtet auf eine empirische Überprüfung der Skalenaxiomatik. Es wird die Annahme getroffen, das jeweilige Merkmal würde auf einer Intervallskala gemessen. Unter dieser Annahme wurden auch für die Items des Fragebogens Mittelwerte berechnet. Vgl. Bortz/Döring (2003): S. 74.

5.3.1 Emotion

Die Subskala Emotion misst die allgemeine emotionale Reaktion des Benutzers auf die Arbeit mit der Software.²²⁷

Die Subskala Emotion umfasst dreizehn Items. In der durchgeführten Testreihe an der Universität Innsbruck konnte bei sieben dieser Items ein hoch signifikanter Unterschied (auf einem Signifikanzniveau von $\alpha = 0,01$) zwischen SAP und Semiramis festgestellt werden:

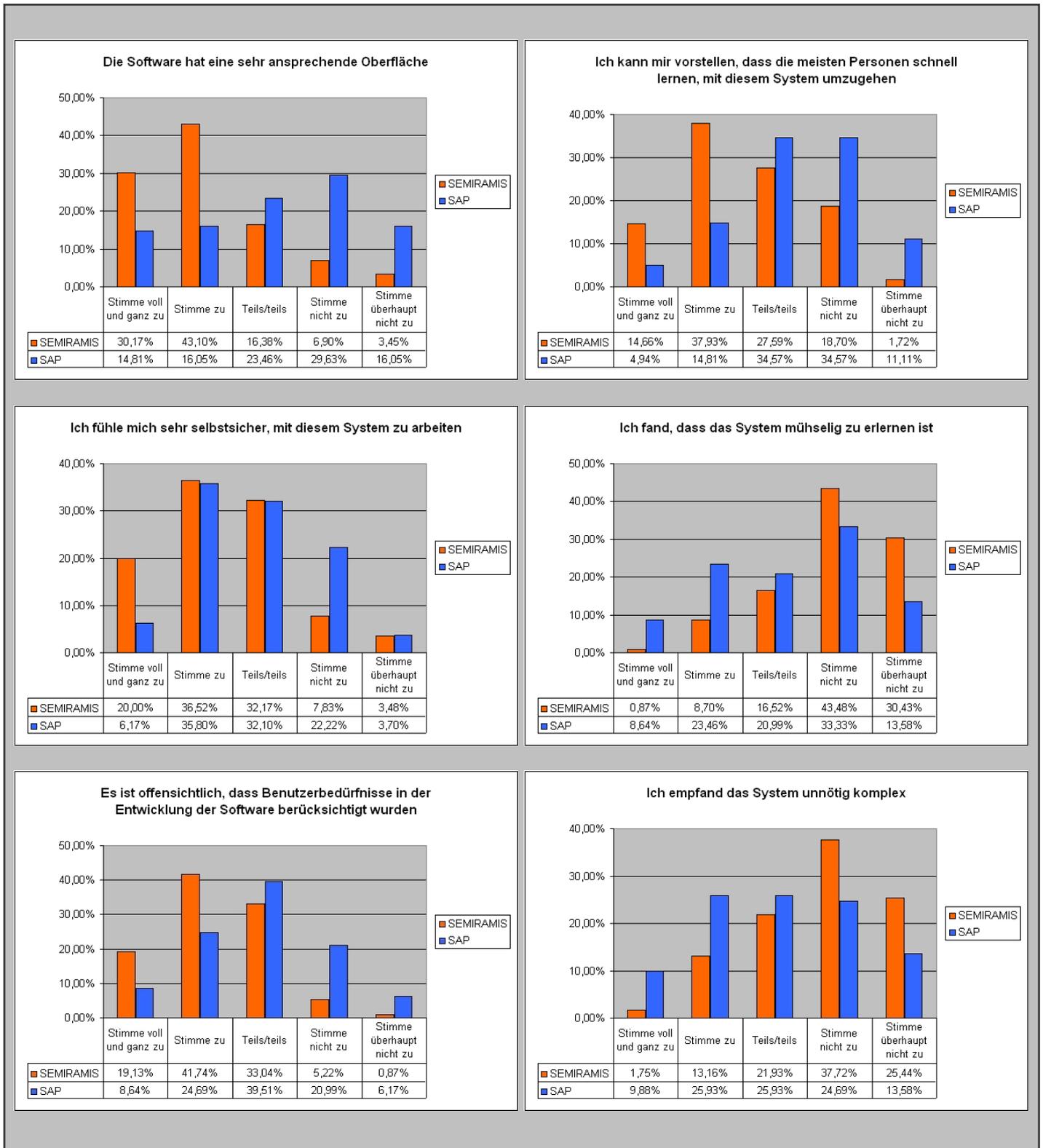
- **User Interface:** Das User Interface von Semiramis wurde deutlich besser eingestuft als jenes von SAP. Ein Vergleich der Mittelwerte unterstreicht das: Semiramis erzielte einen Mittelwert von 2,10, während die Benutzeroberfläche von SAP durchschnittlich mit 3,16 bewertet wurde.²²⁸
- **Einschätzung der Erlernbarkeit des Systems durch andere:** Rund 55 Prozent der SAP-Anwender schätzten die Erlernbarkeit des Systems für andere als zeitaufwändig ein. Unter den Semiramis-Anwendern hingegen gab mehr als die Hälfte der Benutzer an, dass das System für andere schnell zu erlernen sei.
- **Selbstsicherheit in der Benutzung der Software:** Ein Fünftel der Anwender von Semiramis fühlt sich bei der Arbeit am System sehr selbstsicher; weitere 36 Prozent fühlen sich selbstsicher. Nur 11 Prozent gaben an, nicht selbstsicher zu sein. Die Bewertung von SAP ist nicht sehr deutlich.
- **Erlernbarkeit:** Fast drei Viertel der Semiramis-Anwender empfand das System als nicht mühselig zu erlernen. Von Studierenden, welche mit SAP arbeiteten, wurde die Erlernbarkeit weniger deutlich eingestuft.
- **Einschätzung, dass Benutzerbedürfnisse bei der Softwareentwicklung beachtet wurden:** Während die SAP-User unentschlossen waren, ob Benutzerbedürfnisse in der Entwicklung eine Rolle gespielt haben, stimmten 60 Prozent der Studierenden der Semiramis-Kurse dem Item zu.
- **Geringe Komplexität:** Knapp zwei Drittel der Semiramis-User verneinten, dass das System eine zu hohe Komplexität aufweisen würde. Bei den SAP-Anwendern fand dieses Item weder eine deutliche Zustimmung noch eine deutliche Ablehnung.
- **Einfache Handhabung:** 63 Prozent der Semiramis-Benutzer gaben an, dass die Handhabung des Systems einfach ist. Die Beurteilung der Handhabung von SAP fiel nicht eindeutig aus. Knapp 30 Prozent der SAP-User empfanden die Handhabung als einfach. Fast 40 Prozent der Anwender waren unentschlossen.

In zukünftigen Tests gilt es herauszufinden, inwieweit die hier diskutierten Items kausale Zusammenhänge aufweisen. So könnten etwa die wahrgenommene Komplexität und die Benutzeroberfläche einen Einfluss auf die Einfachheit der Handhabung haben. Diese könnte sich

²²⁷ Vgl. Kirakowski, Jurek (1994): The Use of Questionnaire Methods for Usability Assessment. Online im Internet. URL: <http://sumi.ucc.ie/sumipapp.html#sumidev>, Download am 12.02.2006.

²²⁸ Ob eine Likert-Skala ein ordinal- oder intervallskaliertes Merkmal darstellt, ist strittig. Die übliche Forschungspraxis verzichtet auf eine empirische Überprüfung der Skalenaxiomatik. Es wird die Annahme getroffen, das jeweilige Merkmal würde auf einer Intervallskala gemessen. Unter dieser Annahme wurden auch für die Items des Fragebogens Mittelwerte berechnet. Vgl. Bortz/Döring (2003): S. 74.

wiederum auf die Erlernbarkeit auswirken. Die Erlernbarkeit könnte eine entscheidende Determinante für die Selbstsicherheit bei der Arbeit am System sein, ebenso wie für die Einschätzung, dass Benutzerbedürfnisse bei der Entwicklung der Software beachtet wurden.



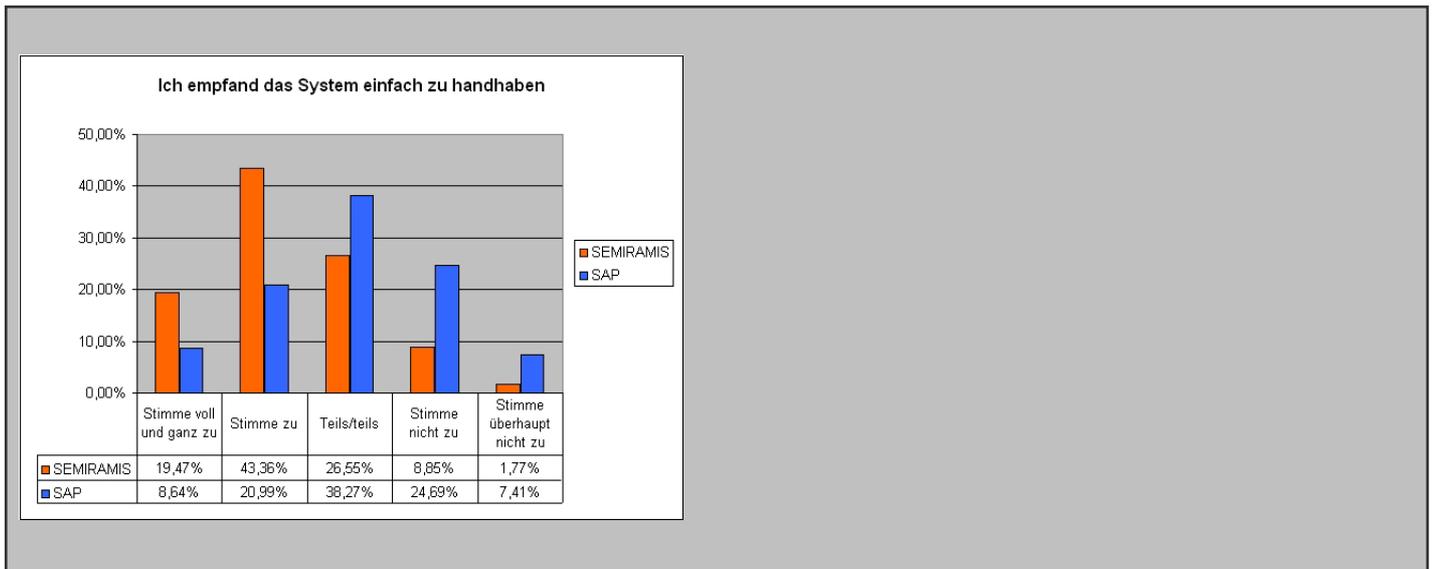


Abbildung 21: Emotion - Items mit signifikantem Unterschied

Alle signifikanten Unterschiede dieser Subskala fielen positiv für die ERP-Software Semiramis aus.

5.3.2 Softwaresteuerung

Die Dimension der Softwaresteuerung misst, inwieweit sich der Benutzer in der Lage fühlt, das System bei der Erledigung einer Aufgabe zu kontrollieren.²²⁹ Der Benutzer sollte Herr über die Software sein, nicht umgekehrt.²³⁰

Von den insgesamt acht untersuchten Items der Subskala Softwaresteuerung weisen drei einen hoch signifikanten Unterschied zwischen den beiden ERP-Systemen auf. Semiramis konnte sich in diesem Untersuchungsbereich von SAP dreimal positiv abheben.

- **Gefühl, bei der Softwarenutzung „alles im Griff“ zu haben:** Etwas mehr als die Hälfte der Semiramis-User gibt an, bei der Arbeit im System alles im Griff zu haben. Die Verteilung bei SAP ist weder eindeutig positiv noch eindeutig negativ. Über 40 Prozent waren sich nicht sicher.
- **Gefühl, die Software zu beherrschen:** Hier bietet sich ein sehr ähnliches Bild wie beim letzten Item.
- **Unterbrechen einer Aufgabe:** Rund 47 Prozent der SAP- und 69 Prozent der Semiramis-User geben an, dass sie eine Aufgabe unterbrechen und später fortsetzen können, ohne alles neu eingeben zu müssen.

²²⁹ Vgl. Kirakowski, Jurek (1994): The Use of Questionnaire Methods for Usability Assessment. Online im Internet. URL: <http://sumi.ucc.ie/sumipapp.html#sumidev>, Download am 12.02.2006.

²³⁰ Vgl. Karat nach Wildstrom (1998): S. 18.

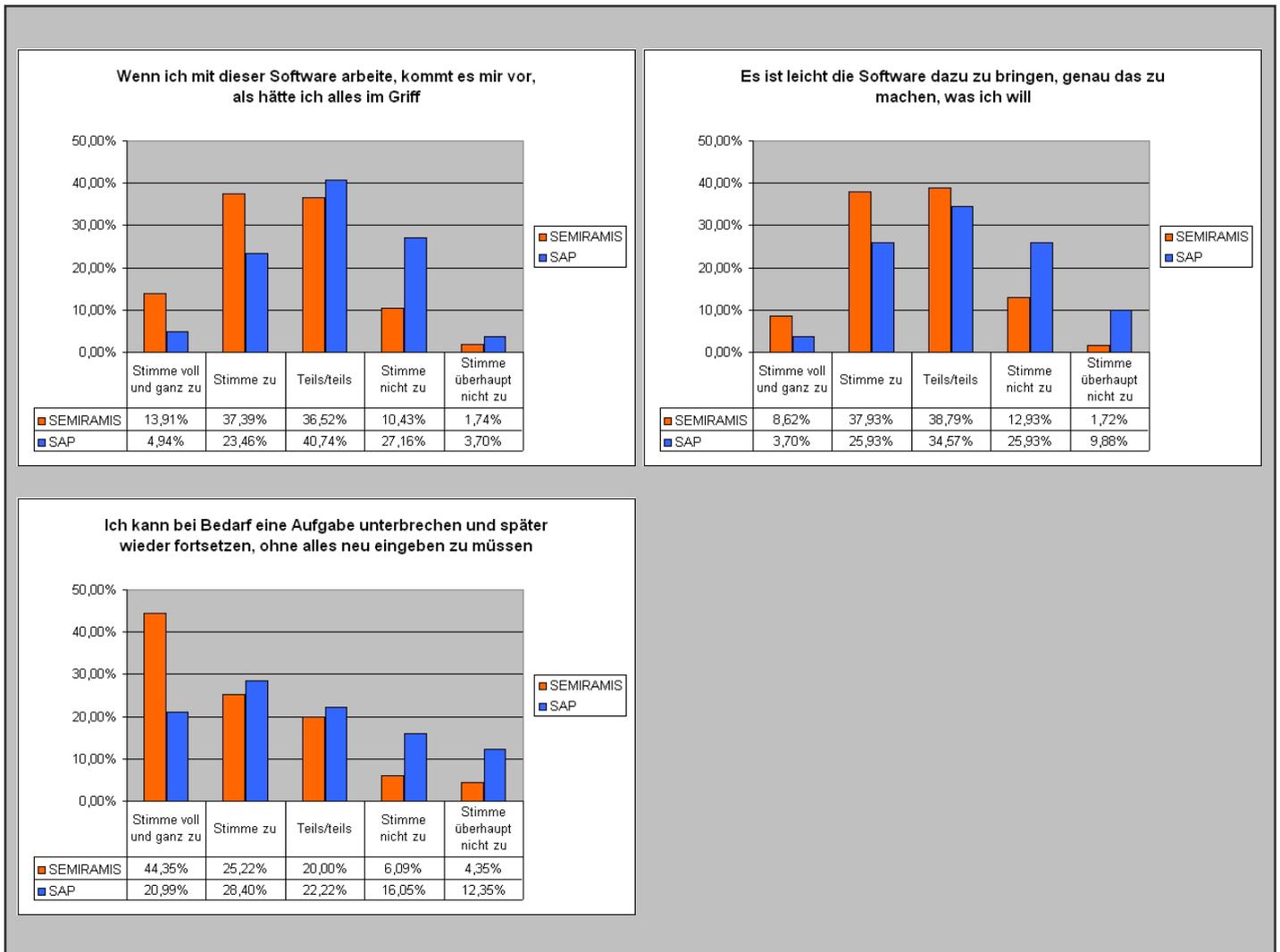


Abbildung 22: Softwaresteuerung - Items mit signifikantem Unterschied

5.3.3 Effizienz

In der DIN EN ISO 9241-11 ist Effizienz als eine der drei Säulen für Software Usability definiert. Nach dem Deutschen Institut für Normung ist Effizienz "der im Verhältnis zur Genauigkeit und Vollständigkeit eingesetzte Aufwand, mit dem Benutzer ein bestimmtes Ziel erreichen"²³¹. Effizienz steht für das Input-/Outputverhältnis und bestimmt folglich das Empfinden des Benutzers, ob ihn die Software bei der Erledigung seiner Aufgaben behilflich ist.²³²

Die Subskala Effizienz des für ERP-spezifisches Usability Testing entwickelten Fragebogens enthält dreizehn Items. SAP und Semiramis unterscheiden sich bei zwei Items in einem signifikanten Ausmaß. Beide konnte Semiramis für sich entscheiden:

²³¹ Deutsches Institut für Normung (2003): EN ISO 9241-11: 1998. S. 4.

²³² Vgl. Kirakowski, Jurek (1994): The Use of Questionnaire Methods for Usability Assessment. Online im Internet. URL: <<http://sumi.ucc.ie/sumipapp.html#sumidev>>, Download am 12.02.2006.

- **Nur wenige Tastenschläge/Mausklicks zur sinnvollen Benutzung notwendig:** Beinahe die Hälfte der SAP-User gab an, dass dies möglich ist; genauso urteilten etwa 60 Prozent der Semiramis-Anwender. Auf SAP-Seite meinten aber auch rund 30 Prozent, dass die sinnvolle Bedienung der Software mit wenigen Tastenschlägen bzw. Mausklicks nicht möglich ist.
- **Zu viele Arbeitsschritte zur Erledigung von Aufgaben:** Während die SAP-Anwender ein sehr uneinheitliches Bild aufweisen, ist das Ergebnis auf Semiramis-Seite etwas deutlicher. Rund 46 Prozent sagten aus, dass dieses Item nicht zutrifft.

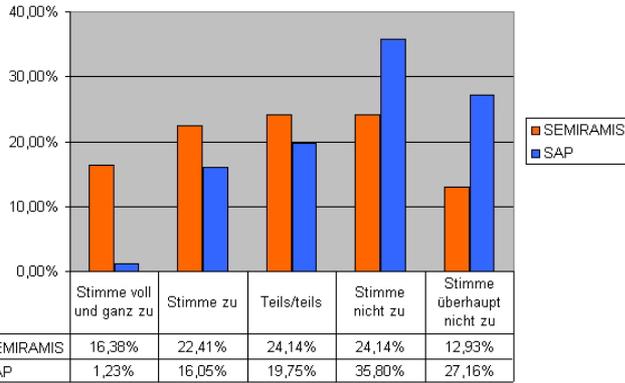
Innerhalb des Unterbereichs Effizienz finden sich daneben noch sieben Items, welche eine hohe Signifikanz aufzeigen:

- **Zu langsame Reaktion der Software auf Inputs:** Fast zwei Drittel der SAP-User meinten, dass die Reaktion von SAP auf Inputs nicht zu langsam ist. Die Einschätzung bezüglich dieses Items fiel unter den Semiramis-Anwendern recht unterschiedlich aus.
- **Akzeptable Geschwindigkeit der Software:** SAP war auch bei diesem Item erfolgreich. 70 Prozent stimmten diesem Item zu. Die Software Semiramis konnte bei diesem Item im Vergleich einen Wert von 51 Prozent erreichen.
- **Logischer Aufbau des Benutzermenüs:** 80 Prozent der Semiramis-Anwender stimmten dem Item zu, bei SAP lag die Zustimmung bei nur 54 Prozent.
- **Erkennbarkeit der Funktionsoptionen der Anwendungen:** Die Testergebnisse lassen darauf schließen, dass die Funktionsoptionen der Anwendungen in Semiramis schneller und klarer erkennbar sind als in SAP.
- **Klare Darstellung von Systeminformationen:** Die SAP-Benutzer zeichneten bei diesem Item ein unklares Bild; unter den Semiramis-Usern stimmten 57 Prozent zu.
- **Passende Anordnung von Menüpunkten oder Funktionen:** Auch bei diesem Item zeigt sich keine eindeutige Verteilung für SAP. Die Semiramis-Anwender pflichteten zu 70 Prozent dem Item positiv bei. Nur 5 Prozent meinten, dass sie Menüpunkte und Funktionen nicht dort finden, wo sie ihrer Meinung nach auch sein sollten.
- **Erforderlicher Aufwand für Arbeitsergebnis:** Semiramis wird von 67 Prozent als effizientes System eingestuft, SAP von nur 47 Prozent.

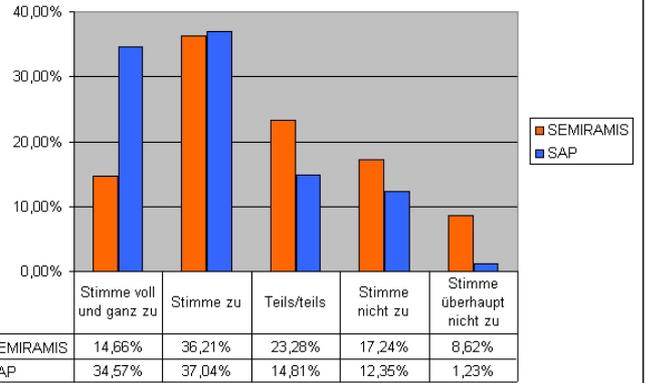
In diesem Untersuchungsbereich konnte SAP zweimal signifikant besser abschneiden als Semiramis. Der Vorsprung in diesem Bereich hat sich jedoch, wie in Anhang D ersichtlich, im letzten Semester verringert. Das gilt jedoch genauso für einige Items, welche Semiramis für sich entscheiden konnte. Diese Subskala wird weniger von Semiramis dominiert.

Es wäre auch hier von hohem Interesse, die kausalen Zusammenhänge innerhalb des Untersuchungsbereiches Effizienz näher zu analysieren.

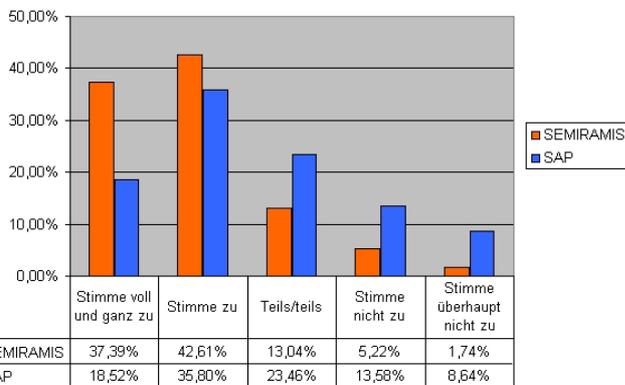
Die Software reagiert zu langsam auf Inputs



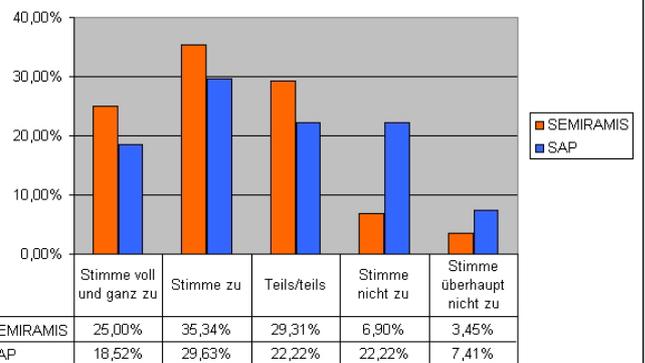
Die Geschwindigkeit der Software ist schnell genug



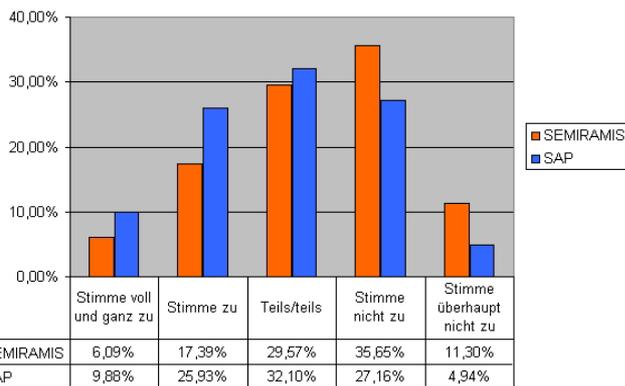
Der Aufbau des Benutzermenüs ist logisch strukturiert



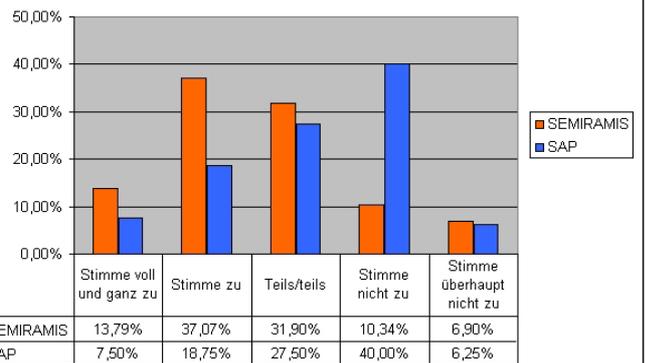
Zur sinnvollen Benutzung der Software sind nur wenige Tastenschläge notwendig



Man braucht zu viele Arbeitsschritte, um Aufgaben zu erledigen



Es ist sofort erkennbar, welche Funktionsoptionen die einzelnen Anwendungen bieten



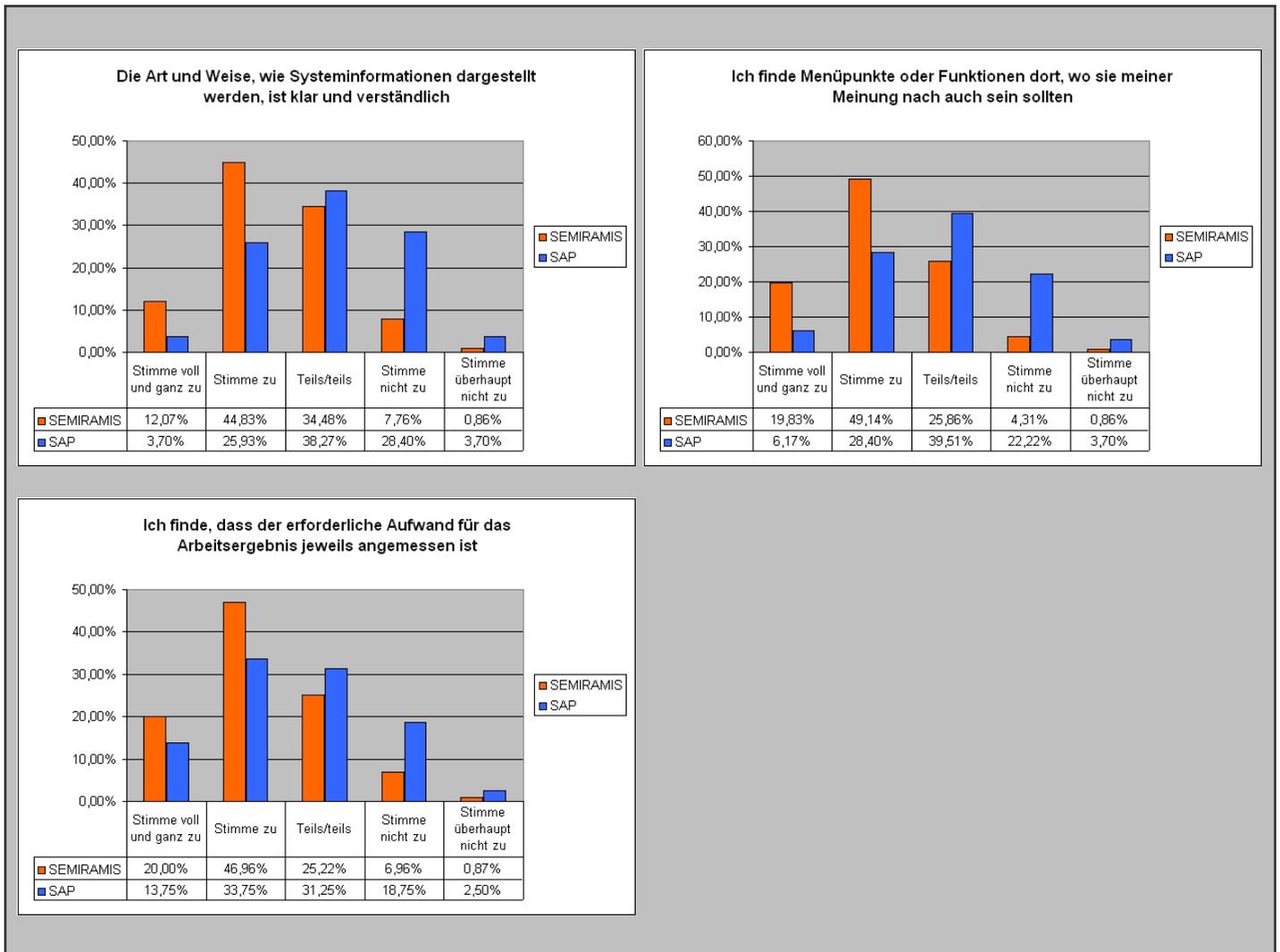


Abbildung 23: Effizienz - Items mit signifikantem Unterschied

5.3.4 Systemunterstützung

Die Subskala Systemunterstützung misst, inwieweit die Software selbst erklärend ist. Das beinhaltet auch die Zweckmäßigkeit der Softwaredokumentation sowie der Hilfestellungen, welche vom System angeboten werden.²³³

Für folgendes Item konnte ein signifikanter Unterschied zugunsten von Semiramis bewiesen werden.

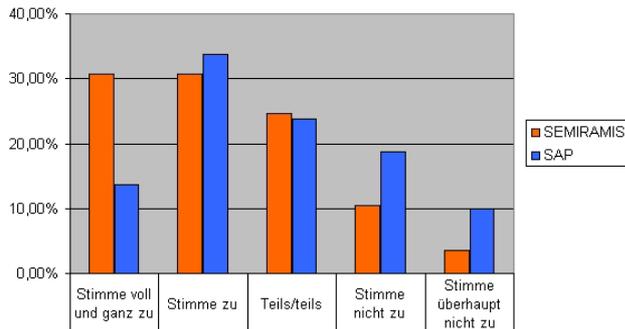
- **Verstehen der angebotenen Informationen der Software:** Knapp 60 Prozent der Semiramis-Anwender gaben an, die angebotenen Informationen zu verstehen und auf diese reagieren zu können. Bei SAP konnten 47 Prozent dies bestätigen.

²³³ Vgl. Kirakowski, Jurek (1994): The Use of Questionnaire Methods for Usability Assessment. Online im Internet. URL: <<http://sumi.ucc.ie/sumipapp.html#sumidev>>, Download am 12.02.2006.

Sechs weitere der insgesamt zwölf Items zeigten einen hoch signifikanten Unterschied zwischen SAP und Semiramis auf. Diese Bereiche konnte Semiramis für sich gewinnen:

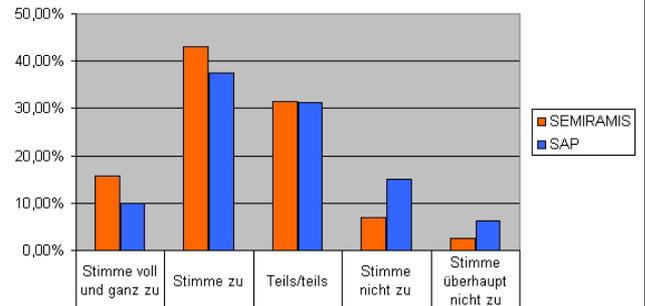
- **Hilfreiche Anweisungen und Hinweise:** Beinahe zwei Drittel der Semiramis-User zeigte sich mit den Systemanweisungen zufrieden, bei SAP war es die Hälfte der Anwender.
- **Einlesen für Softwarenutzung erforderlich:** Die Maxima beider Anwendergruppen liegen für dieses Item bei der Antwortkategorie „Teils/teils“. Es gaben jedoch deutlich mehr SAP- als Semiramis-Anwender an, dass man sich zu sehr einlesen müsse, um mit der Software arbeiten zu können.
- **Hilfreiche Fehlerwarnmeldungen:** Die Fehlerwarnmeldungen in Semiramis wurden von 55 Prozent als hilfreich bezeichnet, nur 20 Prozent hatte eine gegenteilige Meinung. Die Verteilung der Einschätzung der SAP-User zu diesem Item brachte kein eindeutiges positives oder negatives Ergebnis.
- **Erlernen zusätzlicher Dinge nötig:** Während nur die Hälfte der SAP-User dieses Item als nicht zutreffend bezeichnete, waren es unter den Semiramis-Anwendern 70 Prozent.
- **Zur Aufgabenerledigung notwendige Informationen übersichtlich verfügbar:** Auch dieses Item wurde für die Software SAP nicht aussagekräftig beantwortet. Auf Semiramis-Seite stimmten 56 Prozent zu; nur 12 Prozent gaben an, dass dies nicht zutreffend ist.
- **Häufiges Konsultieren von Kollegen oder Verwendung des Handbuchs:** Bei diesem Item zeigt sich ein beinahe deckungsgleiches Bild wie beim vorherigen.

Die Anweisungen und Hinweise sind hilfreich



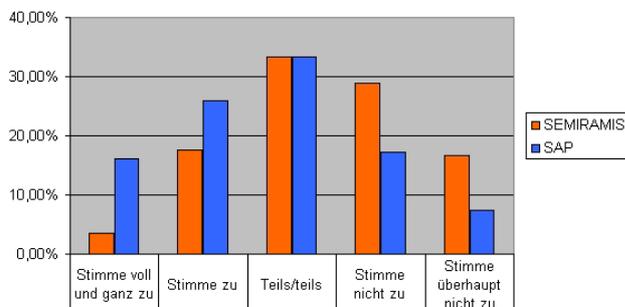
SEMIRAMIS	30,70%	30,70%	24,56%	10,53%	3,51%
SAP	13,75%	33,75%	23,75%	18,75%	10,00%

Ich kann die Informationen, welche die Software anbietet, verstehen und darauf reagieren



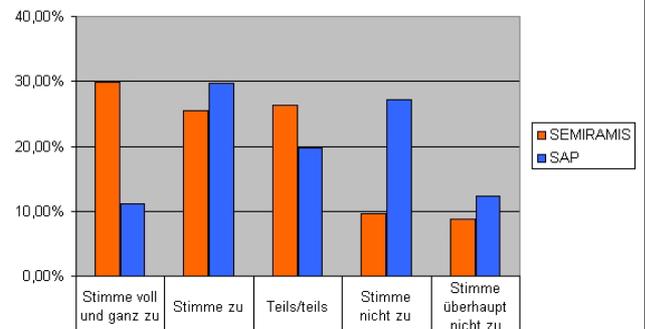
SEMIRAMIS	15,79%	42,98%	31,58%	7,02%	2,63%
SAP	10,00%	37,50%	31,25%	15,00%	6,25%

Man muss sich zu sehr einlesen, um mit der Software arbeiten zu können



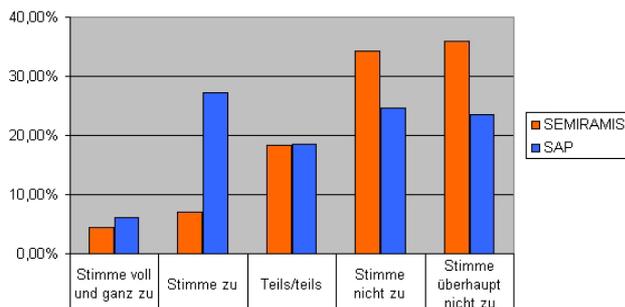
SEMIRAMIS	3,51%	17,54%	33,33%	28,95%	16,67%
SAP	16,05%	25,93%	33,33%	17,28%	7,41%

Die Fehlerwarnmeldungen sind ausreichend hilfreich



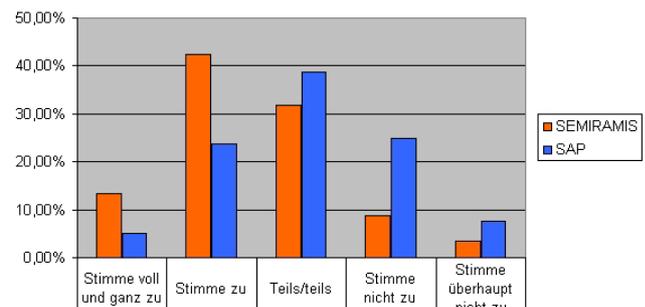
SEMIRAMIS	29,82%	25,44%	26,32%	9,85%	8,77%
SAP	11,11%	29,63%	19,75%	27,16%	12,35%

Ich musste erst viele zusätzliche Dinge erlernen, um mit diesem System zurecht zu kommen



SEMIRAMIS	4,39%	7,02%	18,42%	34,21%	35,96%
SAP	6,17%	27,16%	18,52%	24,69%	23,46%

Die Informationen, die zur Erledigung der Aufgabe notwendig sind, sind auf dem Bildschirm übersichtlich verfügbar



SEMIRAMIS	13,27%	42,48%	31,86%	8,85%	3,54%
SAP	5,00%	23,75%	38,75%	25,00%	7,50%

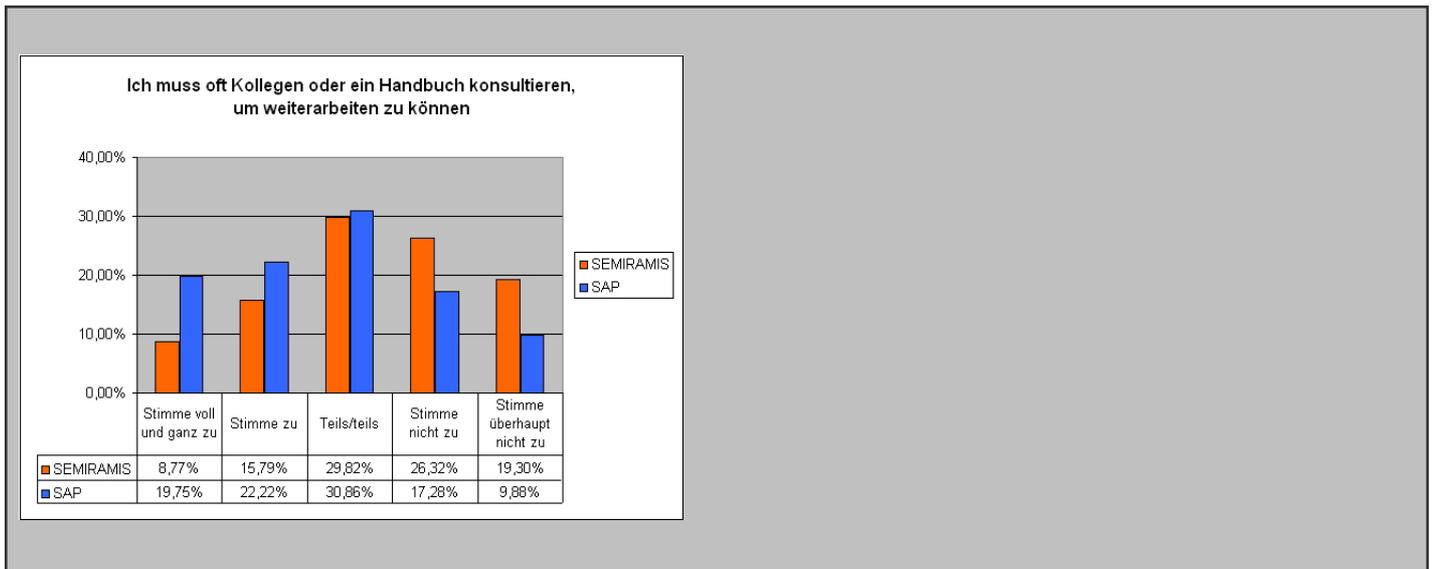


Abbildung 24: Systemunterstützung - Items mit signifikantem Unterschied

5.3.5 Erlernbarkeit

Die Dimension Erlernbarkeit drückt aus, inwieweit das System ein schnelles und einfaches Lernen ermöglicht.²³⁴ Dieses Attribut bezieht sich nicht nur auf das erstmalige Erlernen von Systemfunktionen, sondern umfasst auch das Wiedererlernen sowie das Neuerlernen zusätzlicher Funktionen.

Die sieben Items, aus welchem sich die Subskala Erlernbarkeit zusammensetzt, brachten eine signifikante und drei hoch signifikante Unterschiede zwischen SAP und Semiramis hervor. Einen signifikanten Unterschied brachte folgendes Item:

- **Selbststudium von Funktionen möglich:** Während sich die Semiramis-Anwender nicht darüber im Klaren waren, ob ein Aneignen von Funktionen im Selbststudium möglich wäre, stritt dies rund die Hälfte der SAP-User ab.

Hoch signifikante Unterschiede zu Gunsten von Semiramis gab es in diesen drei Bereichen:

- **Einfaches Erlernen neuer Funktionen:** Rund 70 Prozent der Semiramis-User empfanden das Lernen von neuen Funktionen im System als einfach, nur 6 Prozent stimmten dem nicht zu. SAP hingegen wurde von 18 Prozent als zu schwierig zu erlernen beurteilt; nur 43 Prozent meinten, dass das Erlernen neuer Funktionen in SAP einfach war.
- **Erlernen aller Funktionen:** 43 Prozent der SAP-Anwender stimmten voll und ganz zu, dass sie niemals alle Funktionen der Software erlernen werden; weitere 19 Prozent stimmten dem zu. Für die Semiramis-Anwender zeigt sich ein weniger deutliches Bild.

²³⁴ Vgl. Kirakowski, Jurek (1994): The Use of Questionnaire Methods for Usability Assessment. Online im Internet. URL: <<http://sumi.ucc.ie/sumipapp.html#sumidev>>, Download am 12.02.2006.

- **Gefahrloses Ausprobieren im System möglich:** 53 Prozent meinten, dass Semiramis ihnen die Möglichkeit bietet, auch einmal etwas gefahrlos auszuprobieren. Nur 13 Prozent meinten, dies ist nicht möglich. Die Verteilung der SAP-Anwender weist ihr Maximum in der mittleren Kategorie auf; tendenziell geben aber auch SAP-User an, dass das System ein solches Ausprobieren toleriert.

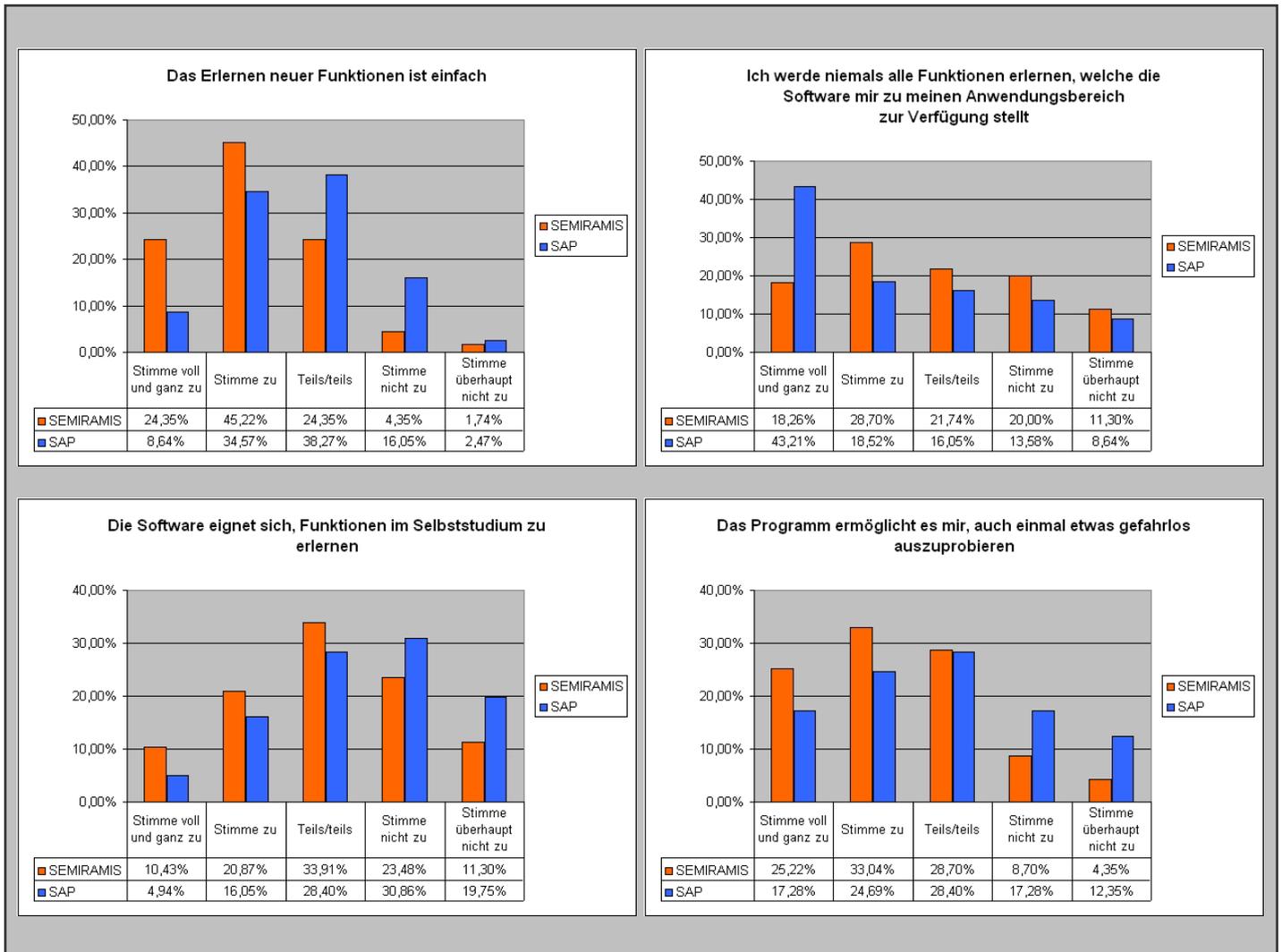


Abbildung 25: Erlernbarkeit - Items mit signifikantem Unterschied

5.4 Endergebnisse und Rückschlüsse

Die nähere Betrachtung jener Items mit signifikantem Unterschied zwischen SAP und Semiramis hat bereits nahe gelegt, dass sich, wie sich im Usability Test herausgestellt hat, Semiramis gegenüber SAP mit einer höheren Software Usability Qualität auszeichnet. Der Vergleich der Indizes für die Subskalen belegt dies (siehe Abbildung 27).²³⁵ Insgesamt kam es bei der Beantwortung der Items zu 4 signifikanten und 26 hoch signifikanten Bewertungsunterschieden. Davon konnte Semiramis 28 für sich entscheiden. SAP konnte bei zwei Items glänzen. Semiramis konnte besonders in den Bereichen Emotion und Erlernbarkeit punkten. Die geringsten Differenzen zwischen den beiden getesteten ERP-Systemen weisen die Subskalen Softwaresteuerung und Effizienz auf.

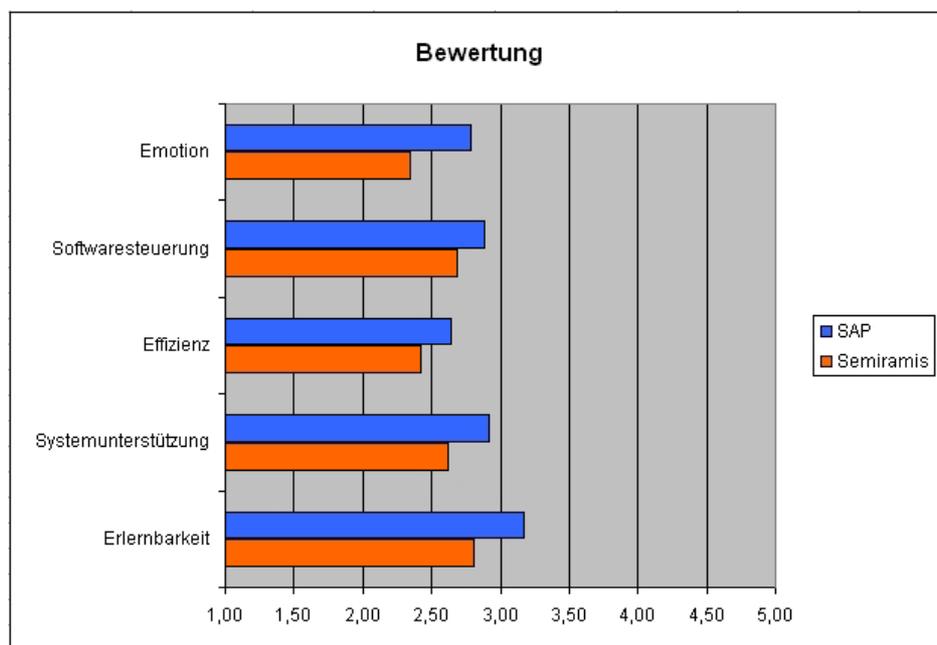


Abbildung 26: Bewertung der Subskalen im Vergleich zwischen SAP und Semiramis²³⁶

Das soeben gewonnene Bild, dass Semiramis vor allem in den Bereichen Emotion und Erlernbarkeit punktet, bestätigt sich auch bei der Analyse der zehn bestgereihten Items durch die Semiramis-User. Davon sind vier Items der Subskala Emotion zuzuordnen, zwei dem Untersuchungsbereich Erlernbarkeit. Weitere zwei Items entstammen der Subskala Effizienz, jeweils eines dem Untersuchungsbereich Softwaresteuerung bzw. Systemunterstützung.

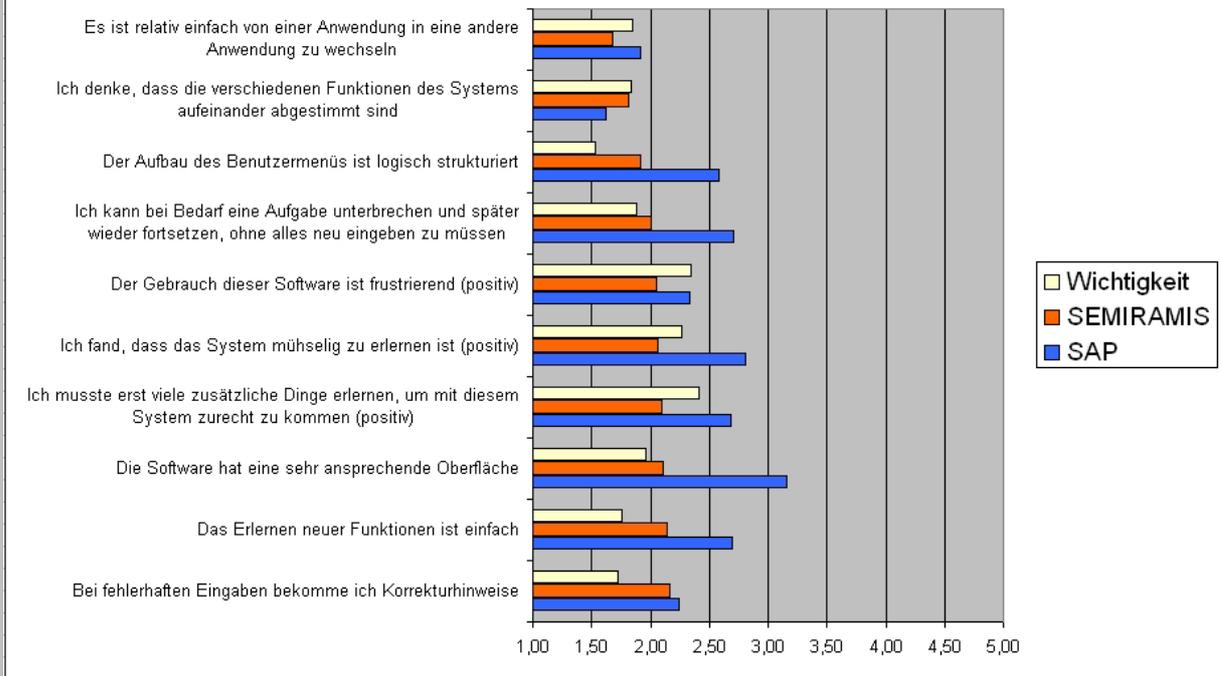
²³⁵ Ob eine Likert-Skala ein ordinal- oder intervallskaliertes Merkmal darstellt, ist strittig. Die übliche Forschungspraxis verzichtet auf eine empirische Überprüfung der Skalenaxiomatik. Es wird die Annahme getroffen, das jeweilige Merkmal würde auf einer Intervallskala gemessen. Unter dieser Annahme wurden auch für die Items des Fragebogens Mittelwerte berechnet. Vgl. Bortz/Döring (2003): S. 74.

²³⁶ Items mit ähnlichen Inhalten, beispielsweise „Ich würde es nicht mögen, diese Software jeden Tag zu benutzen“ und „Ich glaube, ich würde die Software gerne regelmäßig benutzen“, wurden daraufhin überprüft, ob sie korrelieren. Als Itempaare mit Korrelationen gelten fortan: 1.1 und 1.3, 1.2 und 1.5, 1.6 und 1.8 sowie 2.1 und 2.4. Jedes dieser Items wurde in die Berechnung der Mittelwerte für die Subskalen mit 50 Prozent einbezogen, um Verzerrungen zugunsten dieser Usability-Aspekte zu verhindern.

Semiramis war SAP speziell bezüglich der Aspekte User Interface, logischer Aufbau des Benutzermenüs, Möglichkeit eine Aufgabe zu unterbrechen und Erlernbarkeit überlegen. Unter den von Semiramis-Benutzern schlechter bewerteten Items fällt vor allem die Bewertung zur Reaktion der Software auf Inputs auf. SAP schneidet in diesem Punkt über die drei Semester hinweg signifikant besser ab. Seit der Umstellung der Hardwarearchitektur (Wintersemester 2005-06) sind die Unterschiede dabei jedoch nicht mehr so eindeutig (siehe Anhang D). Ob die positive Entwicklung des zuletzt untersuchten Semesters bezüglich Semiramis anhält, ist abzuwarten.

Die Abbildung 27 stellt sowohl für die zehn besten wie auch für die zehn schlechtesten Bewertungen der ERP-Software Semiramis die erzielten Mittelwerte dar. Zusätzlich zu den durchschnittlichen Bewertungen der Semiramis-User finden sich jene der SAP-Anwender sowie die subjektive Wichtigkeit der Items für Semiramis-User.

Semiramis: Die am besten bewerteten Items



Semiramis: Die am schlechtesten bewerteten Items

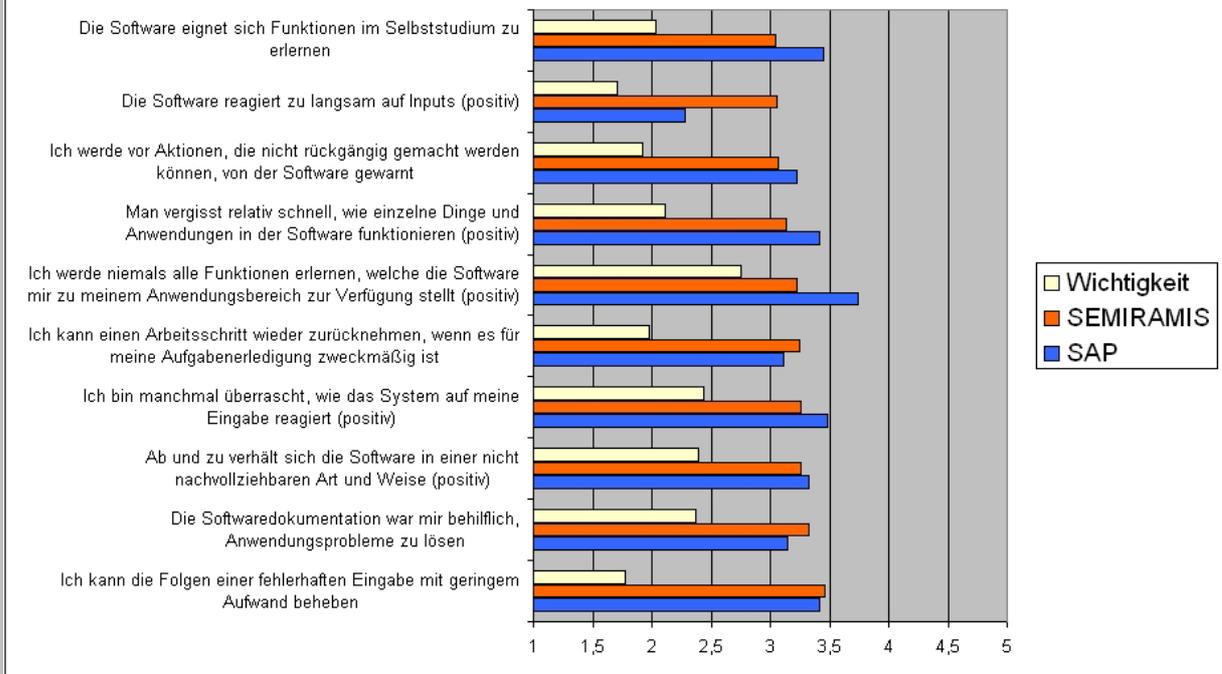


Abbildung 27: Semiramis - Die zehn besten und schlechtesten Bewertungen²³⁷

²³⁷ Die subjektive Wichtigkeit bezieht sich jeweils auf das Urteil der Semiramis-User.

Die am besten bewerteten Items bezüglich SAP umfassen jeweils vier Items aus den Bereichen Effizienz und Emotion. Die restlichen zwei der Top 10 sind der Softwaresteuerung bzw. der Systemunterstützung zuzuordnen.

Die subjektiv wahrgenommene Effizienz der Software SAP wird besonders deutlich, wenn man die Aspekte der Geschwindigkeit sowie der Reaktion der Software auf Inputs betrachtet. Semiramis liegt bei diesen zwei Items hinter SAP. Die Testergebnisse zeigen, dass SAP-Anwender vor allem Usability-Attribute, welche mit Erlernbarkeit im Zusammenhang stehen, schlechter bewerteten. Unter den zehn schlechtesten Bewertungen für die Software SAP finden sich fünf aus der Subskala Erlernbarkeit und zusätzlich ein Item aus dem Bereich Emotion, welches auch auf die subjektiv wahrgenommene Erlernbarkeit deutet (*„Ich kann mir vorstellen, dass die meisten Personen sehr schnell lernen mit diesem System umzugehen“*).

Die Abbildung 28 zeigt neben den durchschnittlichen Bewertungen für SAP zusätzlich wiederum die Vergleichswerte des zweiten untersuchten ERP-Systems sowie die subjektive Wichtigkeit der einzelnen Items für SAP-Anwender.

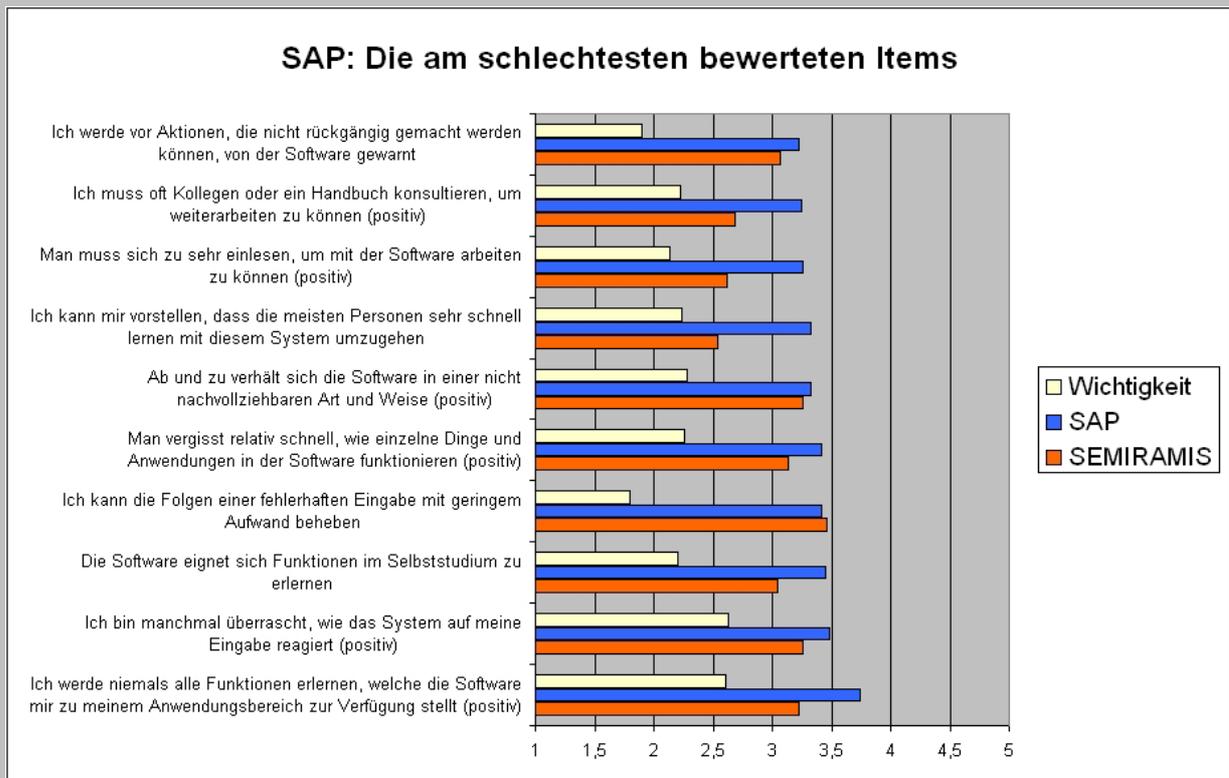
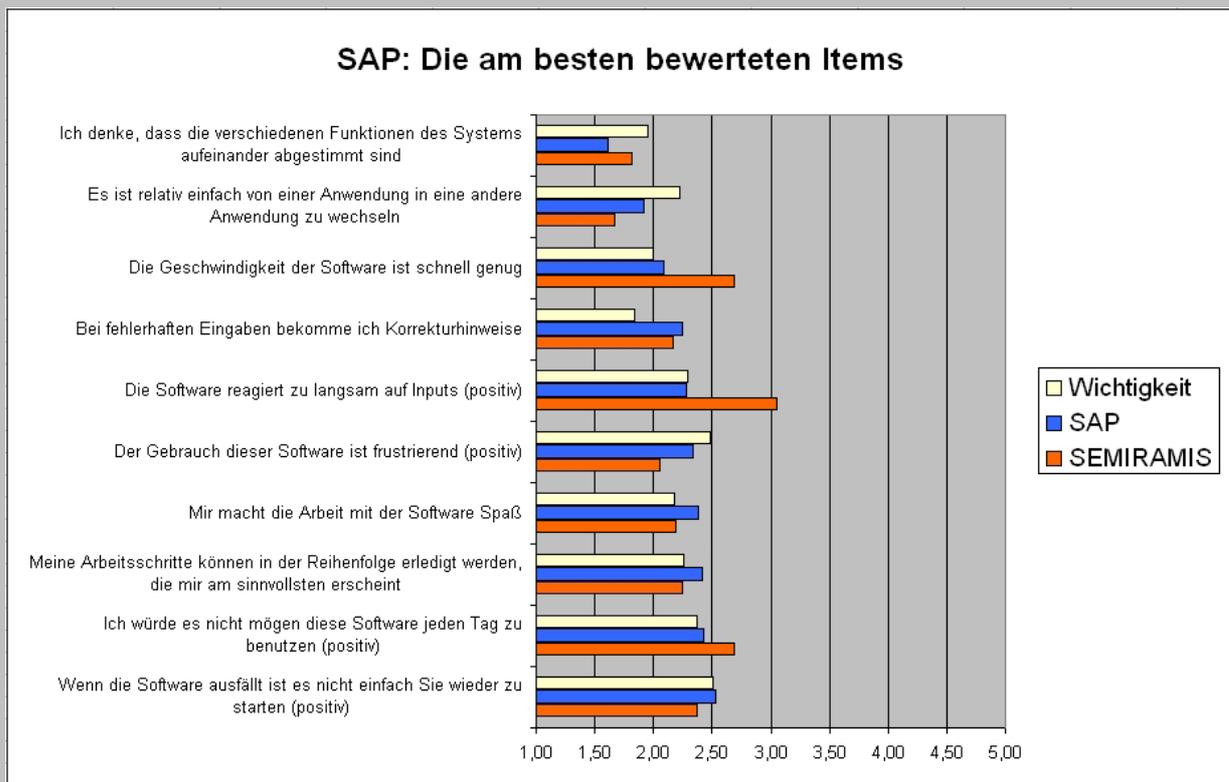


Abbildung 28: SAP - Die zehn besten und schlechtesten Bewertungen²³⁸

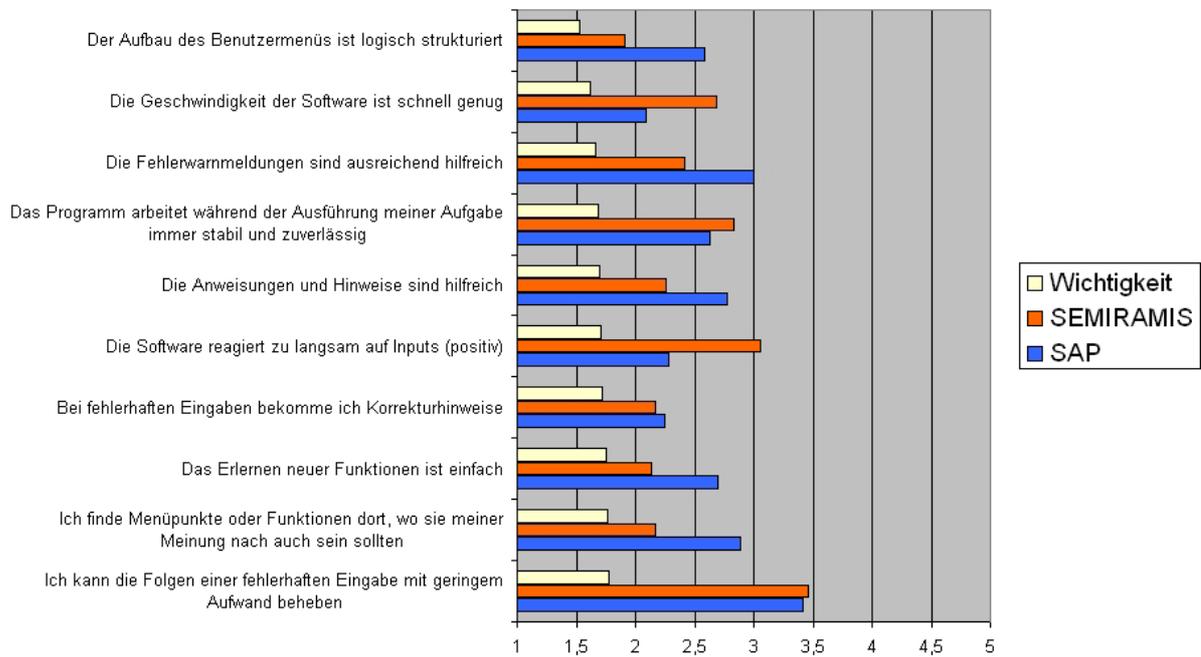
²³⁸ Die dargestellten Mittelwerte der Wichtigkeit der einzelnen Items repräsentieren jene der SAP-Anwender.

Eine Analyse der subjektiven Wichtigkeit der Items zeigt, dass Semiramis-User die Effizienz der Software als sehr wichtig einschätzen, die Software den Benutzeranforderungen in den relevanten Aspekten aber nicht beständig nachkommen kann. Hingegen wurden für die Semiramis-Anwender wichtige Usability-Attribute bezüglich des Benutzermenüs, der Fehlermeldungen und der Systemunterstützung im Allgemeinen sowie der Erlernbarkeit sehr gut bewertet (siehe Abbildung 29).

Die SAP-Anwender wünschten sich hilfreiche Fehlerwarnmeldungen sowie Anweisungen und Hinweise der Software, außerdem einen logischen Aufbau des Benutzermenüs, ein benutzungsfreundliches System und eine leichte Erlernbarkeit. In all diesen Aspekten wurde SAP von Semiramis übertroffen. Auffallend ist, dass die Effizienz der Software für SAP-Anwender nicht wichtig zu sein scheint. Es wird angenommen, dass sie ihnen nicht wichtig zu sein scheint, weil diese Systemeigenschaft als gegeben wahrgenommen wird.

Bei den Usability-Attributen, welche als wenig wichtig eingestuft wurden, fällt im Vergleich zwischen SAP und Semiramis auf, dass sechs der zehn unwichtigsten Items deckungsgleich sind.

Semiramis: Die subjektiv wichtigsten Items



Semiramis: Die subjektiv unwichtigsten Items

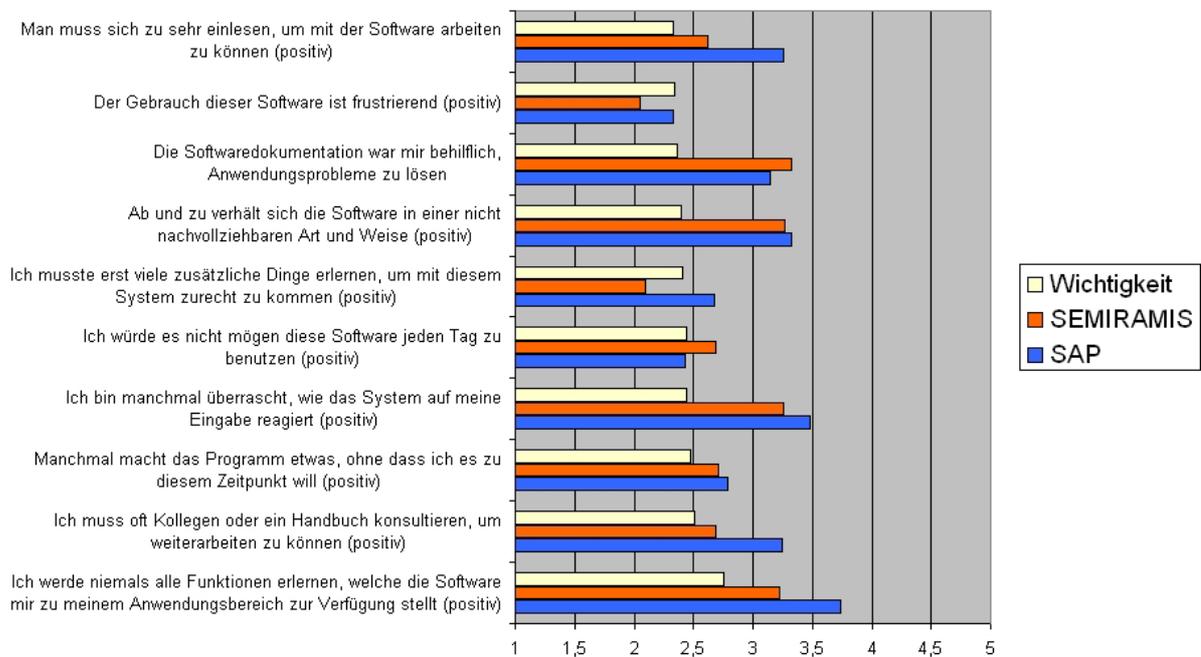
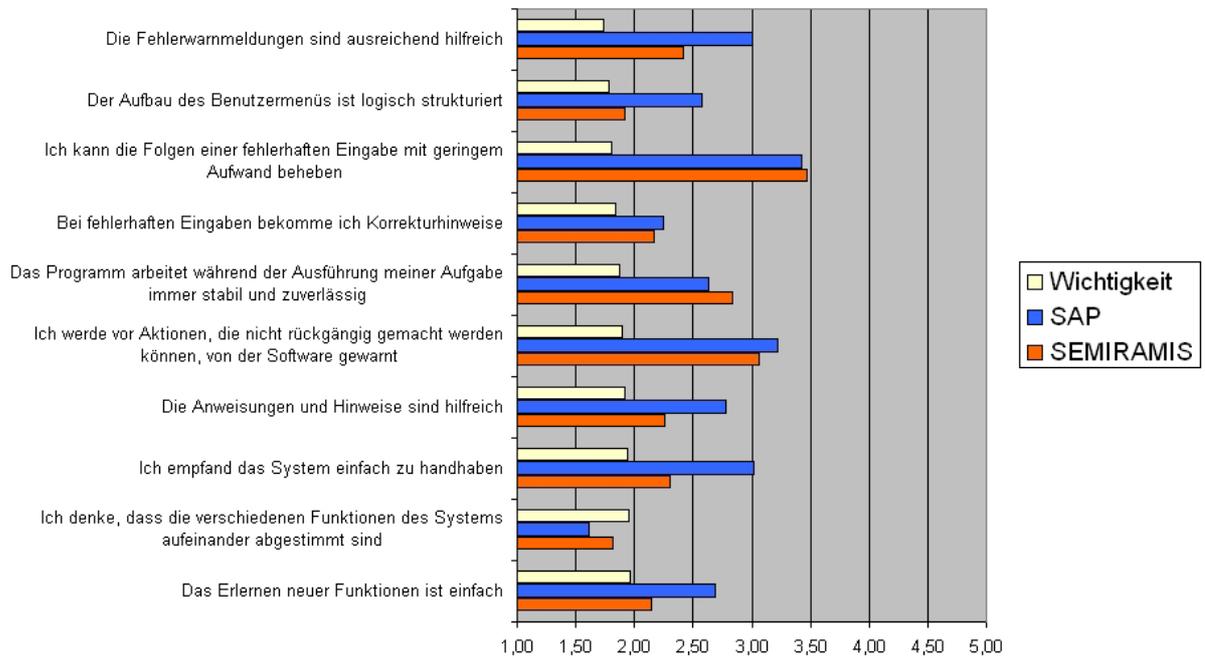


Abbildung 29: Semiramis - Die zehn wichtigsten und unwichtigsten Items

SAP: Die subjektiv wichtigsten Items



SAP: Die subjektiv unwichtigsten Items

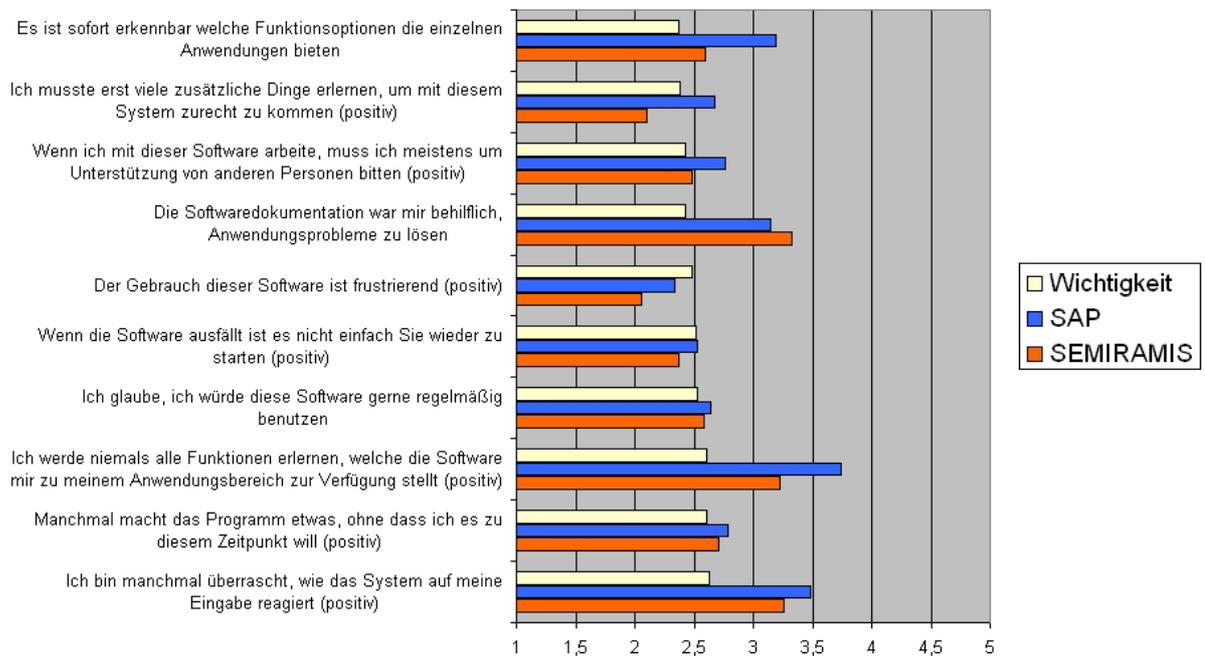


Abbildung 30: SAP - Die zehn wichtigsten und unwichtigsten Items

6 Zukunftsaussichten

Usability zählt schon heute zu den wichtigsten Qualitätskriterien einer Software und wird in Zukunft entscheidend im Wettbewerb sein. Die Kundenzufriedenheit mit Software-Systemen ist in einem kompetitiven Markt für die Auswahl von ERP-Systemen von höchster Relevanz. Dass ein System die entsprechenden vom Unternehmen geforderten Funktionalitäten mitbringt, ist selbstverständlich, ebenso die Systemstabilität.

Nicht der Benutzer sollte sich an ein System anpassen müssen, sondern das System sollte dem Benutzer intuitiv und natürlich erscheinen, um ein Anpassen gar nicht erst nötig werden zu lassen. Die Unternehmen werden zukünftig vermehrt den Anspruch an ERP-Systeme stellen, dass sie die Organisation bestmöglich in ihrer Arbeit unterstützen. Daher wird das Thema Usability für die Softwareanbieter im generellen und für die Anbieter von ERP-Software im speziellen immer interessanter.

Trotz der zentralen Stellung von Usability für den Erfolg einer Software fällt auf, dass die Forschung zum Thema Usability noch in den Kinderschuhen steckt. *Nielsen* glaubt, „dass der Grund hierfür darin zu suchen ist, dass für viele Universitätsabteilungen eine gute HCI-Forschung zum einen zu banal und zum anderen aber auch zu ressourcenintensiv ist. Viele Akademiker verschmähen Forschungsthemen, die zu eng an die Anforderungen der realen Welt geknüpft sind.“²³⁹

Der an der Universität Innsbruck entwickelte Usability Test für ERP-Systeme soll in den nächsten Semestern weiterentwickelt werden, um dann auch im außeruniversitären Bereich Anwendung finden zu können. Denn Unternehmen, welche ERP-Software einsetzen, wenden dafür beträchtliche finanzielle Ressourcen auf, scheinen aber, die Usability der eingesetzten Software unkritisch zu betrachten. Ziel ist es, Unternehmen verstärkt darauf aufmerksam zu machen, wie entscheidend Software Usability für die effiziente und effektive Nutzung eines ERP-Systems ist. Der Usability Test für ERP-Systeme soll Unternehmen schließlich die Möglichkeit geben, die Usability von zwei oder mehr zur Auswahl stehender Softwareprodukte bzw. der eingesetzten Software zu überprüfen.

Zuvor gilt es den Test in weiteren Untersuchungsreihen an der Universität Innsbruck sowie anderen Bildungsinstitutionen weiterzuentwickeln und weiterzuführen, um daraus wichtige Schlüsse für die Anwendung in der Praxis zu ziehen. Interessant erscheinen vor allem kausale Zusammenhänge zwischen einzelnen Testitems sowie die Frage, wie sich die Indizes für die Subskalen zusammensetzen sollten.

Ein hoher Grad an Software Usability muss in naher Zukunft zum Standard eines jeden ERP-Systems werden. Der Markt wird dies immer nachhaltiger fordern. Softwareanbieter, die erfolgreich werden bzw. bleiben wollen, werden dieser Forderung mittelfristig nachkommen müssen.

²³⁹ Nielsen, Jakob (2002): Top-Labore im Bereich der HCI-Forschung. Online im Internet. URL: <<http://www.usability.ch/Alertbox/20020331.htm>>, Download am 13.02.2006.

II Bibliografie

Bücher, Zeitschriften und Zeitungen

- Adams**, Dennis/Nelson, R./Todd, Peter (1992): Perceived Usefulness, Ease of Use and Usage of Information Technology: A Replication. In: MIS Quarterly 1992. 16(2). S. 227 - 247.
- Au**, Norman/Ngai, Eric/Cheng, Edwin (2002): A critical review of end-user information system satisfaction research and a new research framework. In: Omega 2002. 30(6). S: 451 - 478.
- Bevan**, Nigel (1999): Quality in use: Meeting user needs for quality. In: The Journal of Systems and Software 1999. 49(1). S. 89 - 96.
- Bortz**, Jürgen/Döring, Nikola (2003): Forschungsmethoden und Evaluation für Human- und Sozialwissenschaftler. Springer; Berlin, Heidelberg, New York; 2003.
- Burmester**, Michael/Hassenzahl, Marc/Koller, Franz (2002): Usability ist nicht alles - Wege zu attraktiven Produkten. In: i-com 2002. 1(1). S. 32 - 40.
- Davenport**, Thomas (1998): Putting the Enterprise into the Enterprise System. In: Harvard Business Review 1998. 76(4). S. 121 - 131.
- Delone**, William/McLean, Ephraim (1992): Information system success: The quest for the dependent variable. In: Information Systems Research 1992. 3(1). S. 60 - 95.
- Deutsches Institut für Normung** (2003): Software-Ergonomie. Empfehlungen für die Programmierung und Auswahl von Software. DIN Taschenbuch 354. Beuth; Berlin, Wien, Zürich; 2003.
- Dix**, Alan et al. (1993): Human Computer Interaction. Prentice Hall; New York u.a.; 1993.
- Doll**, William/Torkzadeh, Gholamreza (1988): The Measurement of End-User Computing Satisfaction. In: MIS Quarterly 1988. 12(2). S. 259 - 274.
- Donahue**, George (2001): Usability and the Bottom Line. In: IEEE Software 2001. 18(1). S. 31 - 37.
- Dray**, Susan (1995): The Importance of Designing Usable Systems. In: interactions 1995. 2(1). S. 17 - 20.
- Dumas**, Joseph/Redish, Janice (1999): A Practical Guide to Usability Testing. Intellect; Exeter, Portland; 1999.
- Eckstein**, Peter (2000): Angewandte Statistik mit SPSS. Praktische Einführung für Wirtschaftswissenschaftler. Gabler; Wiesbaden; 2000.
- Englisch**, Joachim (1993): Ergonomie von Softwareprodukten. Methodische Entwicklung von Evaluationsverfahren. BI-Wissenschaftsverlag; Mannheim u.a.; 1993.
- Geis**, Thomas/Hartwig, Ronald (1998): Auf die Finger geschaut. Neue ISO-Norm für benutzergerechte interaktive Systeme. In: c't Magazin für Computertechnik 1998(14). S. 168 - 172.
- Geiser**, Georg (1990): Mensch-Maschine-Kommunikation. Oldenbourg; München, Wien, 1990.
- Gelderman**, Maarten (1998): The relation between user satisfaction and usage of information systems and performance. In: Information and Management 1998. 34. S. 11 - 18.
- Gibbs**, W. Wayt (1997): Taking Computers to Task. In: Scientific American 1997. 277(7). S. 64 - 71.
- Gilb**, Tom (1988): Principles of Software Engineering Management. Addison-Wesley; Wokingham u.a.; 1988.

- Gould**, John/Boies, Stephen/Ukelson, Jacob (1997): How To Design Usable Systems. In: Helander, Martin/Landauer, Thomas/Prabhu, Prasad (Hrsg.): Handbook of Human Computer Interaction. Elsevier; Amsterdam u.a.; 1997. S. 231 - 254.
- Hamborg**, Kai-Christoph/Gediga, Günther/Hassenzahl, Marc (2003): Fragebogen zur Evaluation. In: Heinsen, Sven/Vogt, Petra (Hrsg.): Usability praktisch umsetzen. Handbuch für Software, Web, Mobile Devices und andere innovative Produkte. Carl Hanser Verlag; München, Wien; 2003. S. 171 - 186.
- Harrison**, Mary/Henneman, Richard/Blatt, Louis (1994): Design of a Human Factors Cost-Justification Tool. In: Bias, Randolph/Mayhew, Deborah (Hrsg.): Cost-Justifying Usability. Academic Press; Boston u.a.; 1994.
- Hinterhuber**, H. Hans (2004): Strategische Unternehmensführung, I Strategisches Denken; De Gruyter; Berlin; 2004.
- Holzinger**, Andreas (2001a): Basiswissen Multimedia. Band 3: Design. Vogel; Würzburg; 2001.
- Holzinger**, Andreas (2001b): Mensch-Computer-Interaktion - ein interdisziplinäres Thema. OCG-Außenstelle Graz mit einem HCI-Syposium offiziell eröffnet. In: computer kommunikativ 2001. 26(1). S. 14 - 16.
- Holzinger**, Andreas (2005): Usability Engineering Methods for Software Developers. In: Communications of the ACM 2005. 48(1). S. 71 - 74.
- Holzinger**, Andreas/Weidmann, Karl-Heinz, Thurnher, Bettina (2005): Aufbruch in das Zeitalter des „New Computing“. Der Arbeitskreis Human Computer Interaction und Usability Engineering (HCI&UE) stellt sich vor. In: OCG-Journal. Das Magazin der österreichischen Computergesellschaft 2005. 30(1). S. 28 - 29.
- Hoppe**, Heinz (1988): Werkzeuge für die Prototypenentwicklung von Benutzerschnittstellen. In: Balzert, Helmut et al. (Hrsg.): Einführung in die Software-Ergonomie. De Gruyter; Berlin, New York, 1988. S. 280 - 297.
- Juristo**, Natalia/Windl, Helmut/Constantine, Larry (2001): Introducing Usability. In: IEEE Software 2001. 18(1). S. 20 - 21.
- Kalbach**, James (2003): Von Usability überzeugen. In: Heinsen, Sven/Vogt, Petra (Hrsg.): Usability praktisch umsetzen. Handbuch für Software, Web, Mobile Devices und andere innovative Produkte. Carl Hanser Verlag; München, Wien; 2003. S. 7 - 21.
- Maaß**, Susanne (1993): Software-Ergonomie. Benutzer- und aufgabenorientierte Softwaregestaltung. In: Informatik-Spektrum 1993. 16(4). S. 191 - 205.
- Markus**, M. Lynne/Keil, Mark (1994): If We Build It, They Will Come: Designing Information Systems That People Want to Use. In: Sloan Management Review 1994. 35(4). S. 11 - 25.
- Martin**, Reiner (2003): Rechnen sich ERP-Systeme? Die Software allein ist nicht das entscheidende Erfolgskriterium. In: Neue Zürcher Zeitung. 23.09.2003(220). S. 66.
- Martin**, Reiner/Lempp, Peter/Mauterer, Heiko (2003): Wie Software wirklich Nutzen bringt. In: Harvard Business manager 2003. 25(6). S. 71 - 77.
- Martinsons**, Maris/Chong, Patrick (1999): The Influence of Human Factors and Specialist Involvement on Information Systems Success. In: Human Relations 1999. 52(1). S. 123 - 152.
- Mirani**, Rajesh/King, William (2003): Impacts of End-User and Information Center Characteristics on End-User Computing Support. In: Journal of Management Information Systems 1994. 11(1). S. 141 - 166.
- Nielsen**, Jakob (2000): Designing Web Usability. The Practice of Simplicity. New Riders Publishing; Indianapolis; 2000.

- Oppermann**, Reinhard (1988): Software-ergonomische Evaluationsverfahren. In: Balzert, Helmut et al. (Hrsg.): Einführung in die Software-Ergonomie. De Gruyter; Berlin, New York, 1988. S. 323 - 342.
- Oppermann**, Reinhard et al. (1992): Software-ergonomische Evaluation. Der Leitfaden EVADIS II. De Gruyter; Berlin, New York; 1992.
- Preim**, Bernhard (1999): Entwicklung interaktiver Systeme. Grundlagen, Fallbeispiele und innovative Anwendungsfelder. Springer; Berlin, Heidelberg; 1999.
- Promberger**, Kurt/Schlager-Weidinger, Norbert/Traxl, Markus (2003). Verwaltungsmodernisierung durch Enterprise Resource Planning Systeme. Steigerung der Leistungsfähigkeit öffentlicher Verwaltungen durch den Einsatz betriebswirtschaftlicher Standardsoftware. Neuer Wissenschaftlicher Verlag; Wien, Graz; 2003.
- Rohr**, Gabriele (1988): Grundlagen menschlicher Informationsverarbeitung. In: Balzert, Helmut et al. (Hrsg.): Einführung in die Software-Ergonomie. De Gruyter; Berlin, New York, 1988. S. 27 - 48.
- Somers**, Toni/Nelson, Klara/Karimi, Jahangir (2003): Confirmatory Factor Analysis of the End-User Computing Satisfaction Instrument: Replication within an ERP Domain. In: Decision Sciences 2003. 34(3). S. 595 - 621.
- Stary**, Christian (1994): Interaktive Systeme. Software-Entwicklung und Software-Ergonomie. Vieweg; Braunschweig/Wiesbaden; 1994.
- Streitz**, Norbert (1988): Fragestellungen und Forschungsstrategien der Software-Ergonomie. In: Balzert, Helmut et al. (Hrsg.): Einführung in die Software-Ergonomie. De Gruyter; Berlin, New York, 1988. S. 3 - 24.
- Trunick**, Perry (1999): ERP: Promise or pipe dream? In: Transportation and Distribution 1999. 40(1). S. 23 - 26.
- Ulich**, Eberhard (1988): Arbeits- und organisationspsychologische Aspekte. In: Balzert, Helmut et al. (Hrsg.): Einführung in die Software-Ergonomie. De Gruyter; Berlin, New York, 1988. S. 49 - 66.
- Wildstrom**, Stephen (1998): A Computer User's Manifesto. In: Business Week 28.09.1998. 3597. S. 18.
- Wixon**, Dennis/Wilson, Chauncey (1997): The Usability Engineering Framework for Product Design and Evaluation. In: Helander, Martin/Landauer, Thomas/Prabhu, Prasad (Hrsg.): Handbook of Human Computer Interaction. Elsevier; Amsterdam u.a.; 1997. S. 653 - 688.

Internetquellen

- ACM SIGCHI** (1996): Curricula for Human Computer Interaction. Chapter 2: Human Computer Interaction. Online im Internet. URL: <<http://www.acm.org/sigchi/cdg/cdg2.html>>, Download am 19.01.2006.
- Barker**, Dean (2000): Cost Benefits of Usability Engineering. Online im Internet. URL: <<http://www.interfacearchitecture.net/articles/benefits.htm>>, Download am 10.02.2005.
- Berry**, R. (1988): Common User Access - A consistent and usable human computer interface for the SAA environments. Online im Internet. URL: <<http://www.research.ibm.com/journal/sj/273/ibmsj2703E.pdf>>, Download am 08.02.2006.
- Bevan**, Nigel (1995): Human Computer Interaction Standards. Online im Internet. URL: <<http://www.usability.serco.com/papers/hcistd95.pdf>>, Download am 02.02.2006.
- Bevan**, Nigel/Curson, Ian (1998): Planning and Implementing User-Centred Design. Online im Internet. URL: <<http://www.usability.serco.com/papers/ucdtut97.pdf>>, Download am 04.02.2006.

- Bevan**, Nigel/Kirakowski, Jurek/Maisel, Jonathan (1991): What is Usability? Online im Internet. URL: <<http://www.usability.serco.com/papers/whatis92.pdf>>, Download am 02.02.2006.
- Black**, Jane (2002): Usability Is Next to Profitability. Online im Internet. URL: <http://www.businessweek.com/technology/content/dec2002/tc2002124_2181.htm>, Download am 20.01.2006.
- Bräutigam**, Lothar (2000): Software-Ergonomie. Online im Internet. URL: <<http://www.sozialnetz-hessen.de/ca/ph/het/Hauptpunkt/aaaaaaaaaaaaahfi/Unterpunkt/aaaaaaaaaaaaaiih/HauptframeID/aaaaaaaaaaaaaalul/HauptframeTemplate/aaaaaaaaaaaaaapq/>>, Download am 19.01.2006.
- Brooke**, John: A quick and dirty usability scale. Online im Internet. URL: <<http://www.usability.serco.com/trump/documents/Suschapt.doc>>, Download am 11.02.2006.
- DATEch** (2004): DATEch-Prüfhandbuch Gebrauchstauglichkeit. Leitfaden für die ergonomische Evaluierung von Software auf Grundlage von DIN EN ISO 9241, Teile 10 und 11. Online im Internet. URL: <http://www.datech.de/share/files/Pruefhandbuch_ISO_9241.pdf>, Download am 23.01.2006.
- Dzida**, Wolfgang et al. (2000): Gebrauchstauglichkeit von Software. ErgoNorm: Ein Verfahren zur Konformitätsprüfung von Software auf der Grundlage von DIN EN ISO 9241 Teile 10 und 11. Online im Internet. URL: <http://www.ergonomic.de/files/abschlussbericht_de_lang.pdf>, Download am 10.02.2006.
- Fachrichtung Informationswissenschaft Saarbrücken** (2006): Arbeitsbereich Usability Engineering. Methoden und Verfahren. Die Nutzerzentrierten Methoden. Online im Internet. URL: <http://usability.is.uni-sb.de/methoden/nutzer_methoden.php>, Download am 02.02.2006.
- Färbinger**, Peter (2000): Enterprise Resource Planning. Online im Internet. URL: <<http://ebusiness.or.at/erp.htm>>, Download am 12.02.2006.
- Eichinger**, Armin: Usability Engineering. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-engineering.html>>, Download am 18.01.2006
- Eichinger**, Armin: Usability - Definition. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-definition.html>>, Download am 18.01.2006.
- Eichinger**, Armin: Usability - Kosten. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-kosten.html>>, Download am 18.01.2006.
- Eichinger**, Armin: Usability Tests. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-tests.html>>, Download am 18.01.2006.
- Eichinger**, Armin: Usability - Vorbemerkungen. Online im Internet. URL: <<http://pcptpp030.psychologie.uni-regensburg.de/student2001/Skripten/Zimmer/u-vorbemerkungen.html>>, Download am 18.01.2006.
- EUR-Lex**: Richtlinie 90/270/EWG des Rates vom 29. Mai 1990 über die Mindestvorschriften bezüglich der Sicherheit und des Gesundheitsschutzes bei der Arbeit an Bildschirmgeräten (Fünfte Einzelrichtlinie im Sinne von Artikel 16 Absatz 1 der Richtlinie 89/391/EWG). Online im Internet. URL: <<http://europa.eu.int/eur-lex/lex/LexUriServ/LexUriServ.do?uri=CELEX:31990L0270:DE:HTML>>, Download am 11.02.2006.
- Geis**, Thomas: Das DATEch-Prüfhandbuch Gebrauchstauglichkeit: Ein Verfahren zur Überprüfung der Normkonformität mit DIN EN ISO 9241-10, -11. Online im Internet. URL: <<http://www.fit-fuer-usability.de/1x1/messen/datech.html>>, Download am 23.01.2006.
- Geis**, Thomas: Willkommen im Usability Begriffszoo. Online im Internet. URL: <<http://www.fit-fuer-usability.de/1x1/basics/begriffszoo.html>>, Download am 19.01.2006.

- Hellbardt**, Günter: Vorlesung Software-Ergonomie. Texte und Folien. Einführung. Online im Internet. URL: <http://www1.informatik.uni-jena.de/lehre/SoftErg/vor_e100.htm#einfg>, Download am 23.01.2006.
- Holzinger**, Andreas (2006): Usability - Erfolgsfaktor für Unternehmen. Interdisziplinäre Erkenntnisse aus Psychologie und Informatik bestimmen die Zukunft. Online im Internet. URL: <[http://www.competence-site.de/wissensmanagement.nsf/8A644AB5EEF5063FC12570EC00351C51/\\$File/g47_holzinger_competencesite01.pdf](http://www.competence-site.de/wissensmanagement.nsf/8A644AB5EEF5063FC12570EC00351C51/$File/g47_holzinger_competencesite01.pdf)>, Download am 01.02.2006.
- Hunkirchen**, Peter (2005): Bitte laut denken: "Thinking Aloud". Online im Internet. URL: <<http://www.fit-fuer-usability.de/1x1/messen/thinking.html>>, Download am 02.02.2006.
- IBM**: Cost justifying ease of use. Complex solutions are problems. Online im Internet. URL: <http://www-3.ibm.com/ibm/easy/eou_ext.nsf/Publish/23>, Download am 20.01.2006.
- Institut für Software-Ergonomie und Usability** (2002): Was ist Usability? Online im Internet. URL: <<http://www.usability.ch/Deutsch/usab.htm>>, Download am 01.02.2006.
- Kirakowski**, Jurek (1994): The Use of Questionnaire Methods for Usability Assessment. Online im Internet. URL: <<http://sumi.ucc.ie/sumipapp.html#sumidev>>, Download am 12.02.2006.
- Mutschler**, Bela/Reichert, Manfred (2004): Usability-Metriken als Nachweis der Wirtschaftlichkeit von Verbesserungen der Mensch-Maschine-Schnittstelle. Online im Internet. URL: <<http://www.mutschler.info/downloads/MetriKon-Mutschler-Reichert-Usability-Metriken.pdf>>, Download am 20.01.2006
- Nielsen**, Jakob (2002): Top-Labore im Bereich der HCI-Forschung. Online im Internet. URL: <<http://www.usability.ch/Alertbox/20020331.htm>>, Download am 13.02.2006.
- Nielsen**, Jakob (2003): Return on Investment for Usability. Online im Internet. URL: <<http://www.useit.com/alertbox/20030107.html>>, Upload am 07.01.2003, Download am 03.02.2006.
- Nielsen**, Jakob/Coyne, Kara (2001): A Useful Investment. Online im Internet. URL: <http://www.cio.com/archive/021501/et_pundits.html>, Download am 22.01.2006.
- Reilly**, Kevin (2005): AMR Research Releases Report Showing Overall European Market for ERP Vendors to Grow 7% Annually Through 2009. Online im Internet. URL: <<http://www.amrresearch.com/Content/View.asp?pmillid=18386>>, Download am 02.02.2006.
- Schneider**, Wolfgang (2000): Selbstbeschreibungsfähigkeit. Online im Internet. URL: <<http://www.sozialnetz-hessen.de/ca/ph/het/Hauptpunkt/aaaaaaaaaaaaahfi/Unterpunkt/aaaaaaaaaaaaaiih/HauptframeID/aaaaaaaaaaaaaluw/HauptframeTemplate/aaaaaaaaaaaaaapg/>>, Download am 05.02.2006.
- Serco** (2001): User centred design standards. Online im Internet. URL: <<http://usabilitynet.org/trump/resources/standards.htm>>, Download am 04.02.2006.
- Smith**, Sidney/Mosier, Jane (1986): Guidelines For Designing User Interface Software. Online im Internet. URL: <<http://hcibib.org/sam/>>, Download am 11.02.2006.
- SUMI**: Validity of SUMI. Online im Internet. URL: <<http://www.ucc.ie/hfrg/questionnaires/sumi/sumipapp.html#validity>>, Download am 11.02.2006.
- SUMI**: What is SUMI? Online im Internet. URL: <<http://www.ucc.ie/hfrg/questionnaires/sumi/whatis.html>>, Download am 11.02.2006.
- The Ergonomics Society**: Ergonomics. Definition. Online im Internet. URL: <<http://www.ergonomics.org.uk/ergonomics/definition.htm>>, Download am 26.01.2006.
- Usability Forum**: HCI-Geschichte. Online im Internet. URL: <<http://www.usability-forum.com/bereiche/einleitung/geschichte.shtml>>, Download am 27.01.2006.

Usability Forum: Usability Engineering. Online im Internet. URL: <<http://www.usability-forum.com/bereiche/usabilityengineering.shtml>>, Download am 18.01.2006.

Usability Forum: Was bringt Usability? Online im Internet. URL: <<http://www.usability-forum.com/bereiche/kosten.shtml>>, Download am 18.01.2006.

Usability Professionals' Organization (UPA): The ROI of Usability. Online im Internet. URL: <http://www.upassoc.org/usability_resources/usability_in_the_real_world/roi_of_usability.html>, Download am 24.01.2006.

von Arb, Reto (1997): Vorgehensweise und Erfahrungen bei der Einführung von Enterprise-Management-Systemen dargestellt am Beispiel von SAP R/3. Online im Internet. URL: <<http://www.digital-publications.ch/vonarb/vonarb.zip>>, Download am 12.02.2006.

Willumeit, Heinz/Hamborg, Kai/Geidga, Günther (1997): IsoMetrics. Fragebogen zur Evaluation von grafischen Benutzungsschnittstellen. (Lang-Version). Online im Internet. URL: <<http://www.isometrics.uni-osnabrueck.de/boegen/www/Isometrl.ps>>, Download am 11.02.2006.

Wirth, Thomas (2005): Die EN ISO 9241 - 10. Online im Internet. URL: <<http://kommdesign.de/texte/din.htm>>, Download am 19.01.2006.

III Tabellenverzeichnis

Tabelle 1: Elemente der Benutzeroberfläche	14
Tabelle 2: Internationale Normen zur Software Usability	36
Tabelle 3: Systemumfeld	53
Tabelle 4: Mittelwerte der Bewertung im Vergleich zwischen SAP und Semiramis.....	94
Tabelle 5: Mittelwerte der Subjektiven Wichtigkeit im Vergleich zwischen SAP und Semiramis.....	95
Tabelle 6: Median und Modus der Bewertung im Vergleich zwischen SAP und Semiramis	96
Tabelle 7: Mann-Whitney-U-Test für 2 unabhängige Stichproben - SAP- und Semiramis-Anwender..	97

IV Abbildungsverzeichnis

Abbildung 1: Integration der verschiedenen Unternehmensbereiche in einem ERP-System.....	10
Abbildung 2: User Interface	13
Abbildung 3: Das Seeheim-Modell für User Interfaces	14
Abbildung 4: Human Computer Interaction	17
Abbildung 5: Usability-Komponenten	20
Abbildung 6: Anwendungsrahmen der Software Usability nach DIN EN ISO 9241-11.....	21
Abbildung 7: User's Bill of Rights nach Karat.....	24
Abbildung 8: Relative Kosten der Fehlerbeseitigung in den verschiedenen Entwicklungsphasen.....	28
Abbildung 9: Vier Grundregeln des User Centred Design	30
Abbildung 10: Aktivitäten des User Centred Design	31
Abbildung 11: Schritt 4 im User Centred Design Prozess	32
Abbildung 12: Entwurfsprinzipien nach Preim.....	34
Abbildung 13: Kategorisierung der Usability Normen	35
Abbildung 14: Verhältnis zwischen Software, Benutzer und Aufgabe	40
Abbildung 15: Methoden des Usability Testing	40
Abbildung 16: Task Based Testing.....	42
Abbildung 17: SUS-Items	48
Abbildung 18: ErgoNorm-Fragebogen	49
Abbildung 19: IsoMetrics-Fragebogen	49
Abbildung 20: Beispiel für ein Item aus dem Fragebogen	50
Abbildung 21: Emotion - Items mit signifikantem Unterschied	57
Abbildung 22: Softwaresteuerung - Items mit signifikantem Unterschied	58
Abbildung 23: Effizienz - Items mit signifikantem Unterschied.....	61
Abbildung 24: Systemunterstützung - Items mit signifikantem Unterschied	64
Abbildung 25: Erlernbarkeit - Items mit signifikantem Unterschied.....	65
Abbildung 26: Bewertung der Subskalen im Vergleich zwischen SAP und Semiramis.....	66
Abbildung 27: Semiramis - Die zehn besten und schlechtesten Bewertungen.....	68
Abbildung 28: SAP - Die zehn besten und schlechtesten Bewertungen	70
Abbildung 29: Semiramis - Die zehn wichtigsten und unwichtigsten Items	72
Abbildung 30: SAP - Die zehn wichtigsten und unwichtigsten Items	73

V Anhang A: Fragebogen



Usability Test

Vorlage für die Durchführung des Usability Tests an der Universität Innsbruck im Rahmen folgender Lehrveranstaltungen:

Unternehmensplanspiel UPEK (Semiramis)
Integriertes Management mit Semiramis
Integriertes Management mit SAP R/3

Die Daten werden anonym erfasst und haben keinen Einfluss auf universitäre Leistungsbeurteilungen!

A Angaben zum Userprofil

Hinweise zur Markierung:

Markierung: markieren Sie die gewählte Antwort deutlich mit **X**

Kurstitel:

- Unternehmensplanspiel UPEK (Semiramis)
- Integriertes Management mit Semiramis
- Integriertes Management mit SAP R/3

Alter:

Geschlecht:

- Männlich
- Weiblich

Vorkenntnisse mit einem ERP-System (unabhängig von Semiramis oder SAP):

- Ja
- Nein

Wann haben Sie persönlich das erste Mal mit ERP-Systemen zu tun gehabt:

- Vor 3 Monaten
- Vor 3 – 6 Monaten
- Vor 6 Monaten – 1 Jahr
- Vor mehr als 1 Jahr



In einer normalen Woche haben Sie wie lange mit der Software gearbeitet:

- Weniger als 2 Stunden
- 2 – 5 Stunden
- 5 – 8 Stunden
- Mehr als 8 Stunden

An wie vielen Tutorienterminen haben Sie teilgenommen:

- Keinem
- 1 Termin
- 2 – 3 Termine
- Mehr als 3 Termine

Seit wann verwenden Sie für Ihre persönlichen Aufgaben/Arbeiten Computer:

- Weniger als 1 Jahr
- Zwischen 1 – 3 Jahren
- Zwischen 3 – 5 Jahren
- Mehr als 5 Jahre

Inwieweit hat Ihnen die Software geholfen, betriebswirtschaftliche Standardprozesse zu erlernen bzw. nachzuvollziehen:

- Sehr geholfen
- Geholfen
- Teils/teils
- Kaum geholfen
- Nicht geholfen

Haben Sie bezüglich der Schulung als solches noch Kommentare (pos/neg)?

B Kriterien / Untersuchungsbereiche:

1. Emotion
2. Softwaresteuerung
3. Effizienz
4. Systemunterstützung
5. Erlernbarkeit

Hinweise zur Markierung:

1. Spalte: Beantwortung der Frage (Bewertung der Frage)
 2. Spalte: Subjektive Wichtigkeit der Frage (Ist die Frage wichtig für Sie)
 Markierung: markieren Sie die gewählte Antwort deutlich mit **X**

Bitte lesen Sie die einzelnen Fragen der jeweiligen Untersuchungsbereiche genau durch und beantworten Sie dann die Fragen nach bestem Gewissen

<p><u>Beispiel:</u></p> <p>Die Software hat immer so reagiert, wie ich es erwartet habe</p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th colspan="5">Bewertung</th> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">2</td> <td style="text-align: center;">3</td> <td style="text-align: center;">4</td> <td style="text-align: center;">5</td> </tr> </table> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><input type="radio"/></td> </tr> </table>	Bewertung					1	2	3	4	5	<input type="radio"/>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th colspan="5">Subjektive Wichtigkeit</th> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">2</td> <td style="text-align: center;">3</td> <td style="text-align: center;">4</td> <td style="text-align: center;">5</td> </tr> </table> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><input type="radio"/></td> </tr> </table>	Subjektive Wichtigkeit					1	2	3	4	5	<input type="radio"/>								
Bewertung																																
1	2	3	4	5																												
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>																												
Subjektive Wichtigkeit																																
1	2	3	4	5																												
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>																												

Bewertung

1. Stimme voll und ganz zu
2. Stimme zu
3. Teils/teils
4. Stimme nicht zu
5. Stimme überhaupt nicht zu

Subjektive Wichtigkeit

1. Sehr wichtig
2. Wichtig
3. Unentschlossen
4. Unwichtig
5. Absolut unwichtig



1. Emotion:

	Bewertung					Subjektive Wichtigkeit				
	1	2	3	4	5	1	2	3	4	5
1. Mir macht die Arbeit mit der Software Spaß	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
2. Ich würde es nicht mögen, diese Software jeden Tag zu benutzen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
3. Der Gebrauch dieser Software ist frustrierend	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
4. Die Software hat eine sehr ansprechende Oberfläche	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
5. Ich glaube, ich würde diese Software gerne regelmäßig benutzen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
6. Ich kann mir vorstellen, dass die meisten Personen sehr schnell lernen mit diesem System umzugehen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
7. Ich fühle mich sehr selbstsicher, mit diesem System zu arbeiten	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
8. Ich fand, dass das System mühselig zu erlernen ist	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
9. Es ist offensichtlich, dass Benutzerbedürfnisse in der Entwicklung der Software berücksichtigt wurden	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
10. Ich bin manchmal überrascht, wie das Programm auf meine Eingabe reagiert	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
11. Ich empfand das System unnötig komplex	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
12. Ich denke, dass die verschiedenen Funktionen des Systems aufeinander abgestimmt sind	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
13. Ich empfand das System einfach zu handhaben	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					



2. Softwaresteuerung:

	Bewertung					Subjektive Wichtigkeit				
	1	2	3	4	5	1	2	3	4	5
1. Wenn ich mit dieser Software arbeite, kommt es mir vor als hätte ich alles im Griff	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
2. Manchmal macht das Programm etwas, ohne dass ich es zu diesem Zeitpunkt will	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
3. Meine Arbeitsschritte können in der Reihenfolge erledigt werden, die mir am sinnvollsten erscheint	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
4. Es ist leicht die Software dazu zu bringen, genau das zu machen, was ich will	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
5. Ich kann bei Bedarf eine Aufgabe unterbrechen und später wieder fortsetzen, ohne alles neu eingeben zu müssen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
6. Ich kann einen Arbeitsschritt wieder zurücknehmen, wenn es für meine Aufgabenerledigung zweckmäßig ist	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
7. Ich kann am Computer alles so einstellen, dass mir das Lesen und Arbeiten leichter fällt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
8. Ich kann die Folgen einer fehlerhaften Eingabe mit geringem Aufwand beheben	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					



3. Effizienz:

	Bewertung					Subjektive Wichtigkeit				
	1	2	3	4	5	1	2	3	4	5
1. Die Software reagiert zu langsam auf Inputs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
2. Wenn die Software ausfällt, ist es nicht einfach sie wieder zu starten	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
3. Die Geschwindigkeit der Software ist schnell genug	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
4. Der Aufbau des Benutzermenüs ist logisch strukturiert	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
5. Zur sinnvollen Benutzung der Software sind nur wenige Tastenschläge bzw. Mausklicks notwendig	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
6. Man braucht zu viele Arbeitsschritte um Aufgaben zu erledigen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
7. Es ist relativ einfach von einer Anwendung in eine andere Anwendung zu wechseln	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
8. Es ist sofort erkennbar welche Funktionsoptionen die einzelnen Anwendungen bieten	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
9. Die Art und Weise, wie Systeminformationen dargestellt werden, ist klar und verständlich	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
10. Das Programm arbeitet während der Ausführung meiner Aufgabe immer stabil und zuverlässig	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
11. Ich finde Menüpunkte oder Funktionen dort, wo sie meiner Meinung nach auch sein sollten	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
12. Ich finde, dass der erforderliche Aufwand für das Arbeitsergebnis jeweils angemessen ist	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
13. Ich muss Werte und Texte eingeben, die der Computer eigentlich wissen könnte	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					

4. Systemunterstützung:

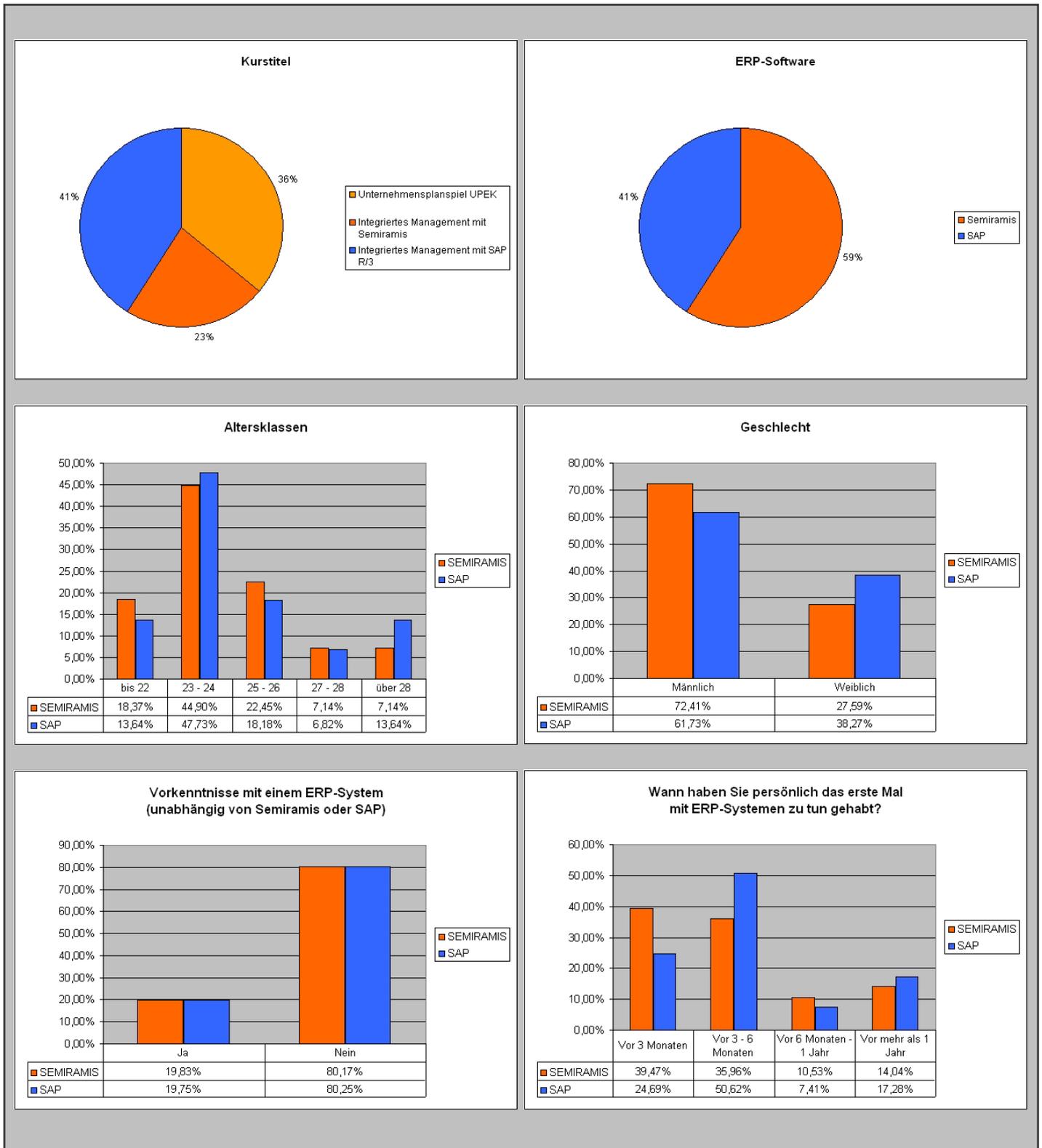
	Bewertung					Subjektive Wichtigkeit				
	1	2	3	4	5	1	2	3	4	5
1. Die Anweisungen und Hinweise sind hilfreich	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
2. Die Softwaredokumentation ist sehr informativ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
3. Ich kann die Informationen, welche die Software anbietet, verstehen und darauf reagieren	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
4. Man muss sich zu sehr einlesen, um mit der Software arbeiten zu können	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
5. Die Fehlerwarnmeldungen sind ausreichend hilfreich	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
6. Ich musste erst viele zusätzliche Dinge erlernen, um mit diesem System zurecht zu kommen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
7. Die Softwaredokumentation war mir behilflich, Anwendungsprobleme zu lösen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
8. Bei fehlerhaften Eingaben bekommen ich Korrekturhinweise	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
9. Die Informationen, die zur Erledigung der Aufgabe notwendig sind, sind auf dem Bildschirm übersichtlich verfügbar	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
10. Ich kann bei der Arbeit mit dem Programm erkennen, welche Eingabe als nächstes von mir erwartet wird	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
11. Ich werde vor Aktionen, die nicht rückgängig gemacht werden können, von der Software gewarnt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					
12. Ich muss oft Kollegen oder ein Handbuch konsultieren, um weiterarbeiten zu können	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>					



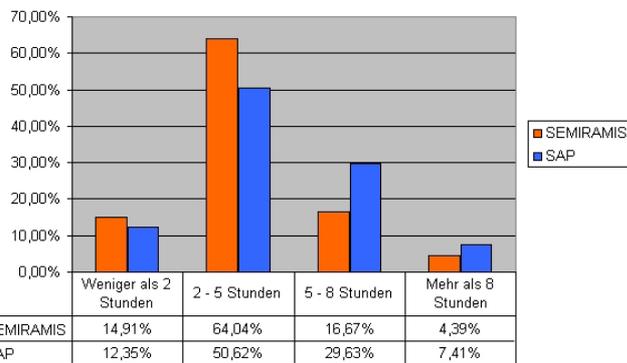
5. Erlernbarkeit:

	Bewertung	Subjektive Wichtigkeit
	1 2 3 4 5	1 2 3 4 5
1. Das Erlernen neuer Funktionen ist einfach	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>
2. Ich werde niemals alle Funktionen erlernen, welche die Software mir zu meinem Anwendungsbereich zur Verfügung stellt	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>
3. Die Software eignet sich, Funktionen im Selbststudium zu erlernen	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>
4. Man vergisst relativ schnell, wie einzelne Dinge und Anwendungen in der Software funktionieren bzw. zu bedienen sind	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>
5. Ab und zu verhält sich die Software in einer nicht nachvollziehbaren Art und Weise	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>
6. Wenn ich mit dieser Software arbeite, muss ich meistens um Unterstützung von anderen Personen bitten	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>
7. Das Programm ermöglicht es mir, auch einmal etwas gefahrlos auszuprobieren	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>

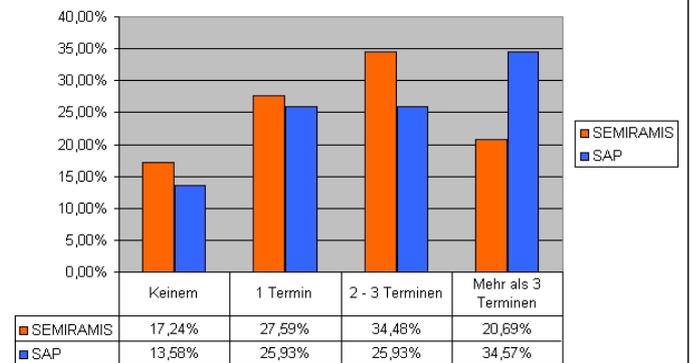
VI Anhang B: Fragen zum User Profil



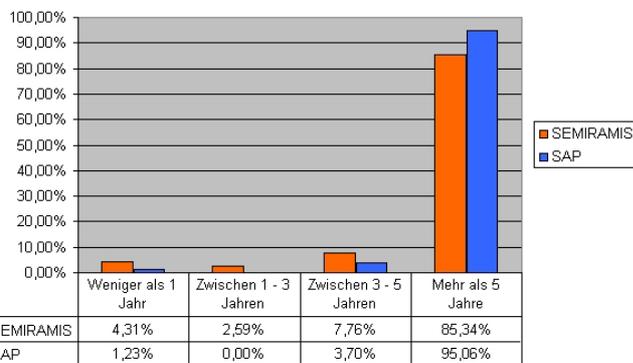
In einer normalen Woche haben Sie wie lange mit der Software gearbeitet?



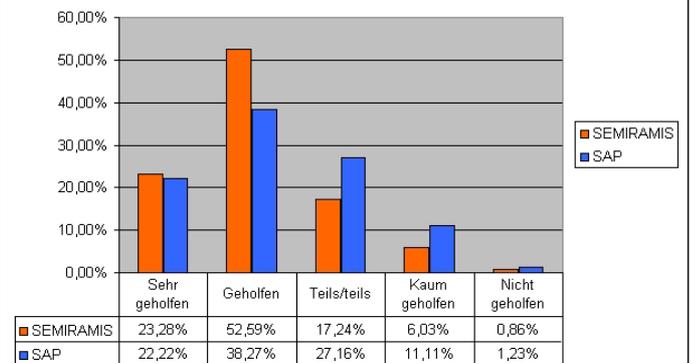
An wie vielen Tutorienterminen haben Sie teilgenommen?



Seit wann verwenden Sie für Ihre persönlichen Aufgaben/Arbeiten Computer?



Inwieweit hat Ihnen die Software geholfen, betriebswirtschaftliche Standardprozesse nachzuvollziehen?



VII Anhang C: Deskriptive Statistik

Tabelle 4: Mittelwerte der Bewertung im Vergleich zwischen SAP und Semiramis

Untersuchungs- bereiche ^{240/241}		Semiramis - SAP ²⁴²		Semiramis 4 - SAP ²⁴³		Semiramis 2 - SAP ²⁴⁴	
		Semiramis	SAP	Semiramis	SAP	Semiramis	SAP
		\bar{x}	\bar{x}	\bar{x}	\bar{x}	\bar{x}	\bar{x}
Emotion	1.1	2,19	2,38	2,13	2,33	2,24	2,42
	1.2*	2,69	2,43	2,71	2,50	2,68	2,38
	1.3*	2,05	2,33	2,21	2,33	1,94	2,33
	1.4	2,10	3,16	2,10	2,89	2,10	3,38
	1.5	2,59	2,64	2,56	2,58	2,60	2,68
	1.6	2,54	3,32	2,58	3,03	2,51	3,56
	1.7	2,38	2,81	2,36	2,67	2,40	2,93
	1.8*	2,06	2,80	2,02	2,61	2,09	2,96
	1.9	2,27	2,91	2,32	2,75	2,24	3,04
	1.10*	3,26	3,48	3,32	3,25	3,22	3,67
	1.11*	2,28	2,94	2,15	2,94	2,37	2,93
	1.12	1,82	1,62	1,98	1,58	1,71	1,64
	1.13	2,30	3,01	2,38	2,81	2,25	3,18
Software- steuerung	2.1	2,49	3,01	2,48	3,00	2,49	3,02
	2.2*	2,71	2,79	2,81	2,86	2,63	2,73
	2.3	2,25	2,41	2,21	2,17	2,28	2,60
	2.4	2,61	3,12	2,63	2,86	2,60	3,33
	2.5	2,01	2,70	1,89	2,58	2,09	2,80
	2.6	3,25	3,11	3,21	3,17	3,28	3,07
	2.7	2,55	2,68	2,29	2,75	2,74	2,62
	2.8	3,47	3,42	3,33	3,36	3,56	3,47
Effizienz	3.1*	3,05	2,28	2,92	2,36	3,15	2,22
	3.2*	2,37	2,53	2,40	2,51	2,35	2,55
	3.3	2,69	2,09	2,46	1,92	2,85	2,22
	3.4	1,91	2,58	2,17	2,39	1,73	2,73
	3.5	2,28	2,70	2,40	2,44	2,21	2,91
	3.6*	2,71	3,09	2,94	3,03	2,55	3,13
	3.7	1,67	1,91	1,73	1,83	1,63	1,98
	3.8	2,59	3,19	2,58	3,00	2,60	3,34
	3.9	2,41	3,02	2,52	3,03	2,32	3,02
	3.10	2,84	2,63	2,90	2,06	2,79	3,09
	3.11	2,17	2,89	2,21	2,92	2,15	2,87
	3.12	2,22	2,63	2,30	2,46	2,16	2,76
	3.13*	2,57	2,84	2,69	2,56	2,49	3,07
System- unterstützung	4.1	2,25	2,78	2,54	2,86	2,06	2,70
	4.2	2,93	2,72	3,20	2,77	2,75	2,67
	4.3	2,38	2,70	2,54	2,56	2,26	2,82
	4.4*	2,62	3,26	2,41	3,28	2,76	3,24
	4.5	2,42	3,00	3,04	3,03	2,00	2,98
	4.6*	2,10	2,68	2,30	2,78	1,96	2,60
	4.7	3,32	3,15	3,36	2,86	3,30	3,38
	4.8	2,17	2,25	2,15	2,28	2,18	2,22
	4.9	2,47	3,06	2,46	3,11	2,48	3,02
	4.10	3,00	2,99	3,04	3,06	2,97	2,93
	4.11	3,06	3,22	3,24	3,25	2,94	3,20
4.12*	2,68	3,25	2,89	2,97	2,54	3,47	
Erlernbarkeit	5.1	2,14	2,69	2,29	2,53	2,03	2,82
	5.2*	3,23	3,74	3,31	3,75	3,16	3,73
	5.3	3,04	3,44	3,08	3,14	3,01	3,69
	5.4*	3,13	3,42	3,21	3,28	3,07	3,53
	5.5*	3,26	3,32	3,49	3,22	3,10	3,40
	5.6*	2,49	2,77	2,65	2,67	2,37	2,84
	5.7	2,34	2,83	2,19	3,03	2,45	2,67

²⁴⁰ Ob eine Likert-Skala ein ordinal- oder intervallskaliertes Merkmal darstellt, ist strittig. Die übliche Forschungspraxis verzichtet auf eine empirische Überprüfung der Skalenaxiomatik. Es wird die Annahme getroffen, das jeweilige Merkmal würde auf einer Intervallskala gemessen. Unter dieser Annahme wurden auch für die Items des Fragebogens Mittelwerte berechnet. Vgl. Bortz/Döring (2003): S. 74.

²⁴¹ Bei der Auswertung jener Statements, welche negativ formuliert sind (*), beispielsweise Statement 1.2 („Ich würde es nicht mögen, diese Software jeden Tag zu benutzen“), wurde die Polung beachtet. Die Bewertung der negativ formulierten Statements wurde nach der Formel $6 - x$ umgerechnet. Ergab die Verteilung einen Mittelwert von 5, so ist hier der Wert 1 ausgewiesen. Es gilt folglich, je kleiner die Ausprägung der Mittelwerte ist, desto besser hat die Software bezüglich des jeweiligen Statements abgeschnitten. Vgl. Hamborg/Gediga/Hassenzahl (2003): S. 175f.

²⁴² Die Befragung der Studierenden erfolgte im Wintersemester 2004-05, im Sommersemester 2005 und im Wintersemester 2005-06 an der Leopold-Franzens-Universität Innsbruck. Insgesamt wurden 197 Studierende befragt; unter ihnen waren 116 Semiramis- und 91 SAP-Anwender.

²⁴³ Ab dem Wintersemester 2005-06 wurde Semiramis 4 in der Lehre eingesetzt. Die Mittelwerte beziehen sich auf eine Erhebung von 48 Studierenden in den Semiramis-Kursen und 36 Studierenden im SAP-Kurs an der LFU Innsbruck.

²⁴⁴ Semiramis 2 wurde im Wintersemester 2004-05 sowie im Sommersemester 2005 in den Semiramis-Kursen verwendet. In diesen beiden Semestern wurden 68 Studierende mit Semiramis- und 45 mit SAP-Kenntnissen befragt.

Usability Testing von ERP-Systemen

Tabelle 5: Mittelwerte der Subjektiven Wichtigkeit im Vergleich zwischen SAP und Semiramis

Untersuchungsbereiche ²⁴⁵		Semiramis - SAP ²⁴⁶		Semiramis 4 - SAP ²⁴⁷		Semiramis 2 - SAP ²⁴⁸	
		Semiramis	SAP	Semiramis	SAP	Semiramis	SAP
		\bar{x}	\bar{x}	\bar{x}	\bar{x}	\bar{x}	\bar{x}
Emotion	1.1	1,82	2,17	1,69	2,11	1,91	2,22
	1.2	2,44	2,37	2,47	2,25	2,42	2,47
	1.3	2,34	2,48	2,27	2,53	2,39	2,44
	1.4	1,96	2,21	2,10	2,08	1,85	2,31
	1.5	2,22	2,53	2,27	2,43	2,18	2,60
	1.6	1,82	2,24	1,88	1,94	1,79	2,48
	1.7	1,97	2,30	1,94	2,17	2,00	2,40
	1.8	2,27	2,25	2,11	2,39	2,38	2,14
	1.9	2,01	2,05	2,11	1,94	1,94	2,13
	1.10	2,44	2,63	2,36	2,64	2,50	2,62
	1.11	2,29	2,25	2,30	2,11	2,28	2,36
	1.12	1,84	1,95	1,83	1,67	1,85	2,18
	1.13	1,79	1,94	1,76	1,69	1,82	2,13
Software- steuerung	2.1	1,95	2,17	1,83	2,11	2,03	2,22
	2.2	2,47	2,60	2,44	2,56	2,50	2,64
	2.3	1,99	2,26	1,98	2,09	2,00	2,40
	2.4	2,13	2,16	2,15	2,11	2,12	2,20
	2.5	1,88	2,22	1,87	1,97	1,89	2,42
	2.6	1,97	2,14	2,06	1,81	1,91	2,40
	2.7	2,08	2,21	2,27	2,14	1,94	2,27
	2.8	1,77	1,80	1,77	1,67	1,77	1,91
Effizienz	3.1	1,71	2,30	1,83	2,25	1,62	2,33
	3.2	2,28	2,51	2,13	2,60	2,39	2,44
	3.3	1,62	2,00	1,65	1,97	1,61	2,02
	3.4	1,53	1,78	1,63	1,69	1,46	1,84
	3.5	1,87	2,09	1,98	1,97	1,79	2,18
	3.6	2,13	2,07	2,23	1,75	2,06	2,33
	3.7	1,84	2,22	1,90	1,97	1,80	2,42
	3.8	1,95	2,38	1,94	2,08	1,95	2,61
	3.9	1,91	2,26	1,92	2,31	1,91	2,22
	3.10	1,68	1,88	1,69	1,61	1,68	2,09
	3.11	1,76	2,00	1,73	1,97	1,79	2,02
	3.12	1,99	2,05	2,02	2,06	1,97	2,04
	3.13	2,25	2,35	2,38	2,36	2,17	2,33
System- unterstützung	4.1	1,70	1,91	1,72	2,08	1,69	1,77
	4.2	2,23	2,27	2,11	2,34	2,32	2,21
	4.3	1,86	2,05	1,89	1,89	1,84	2,18
	4.4	2,33	2,14	2,28	1,89	2,36	2,33
	4.5	1,66	1,74	1,74	1,75	1,61	1,73
	4.6	2,41	2,38	2,35	2,25	2,45	2,49
	4.7	2,37	2,43	2,23	2,47	2,46	2,40
	4.8	1,72	1,84	1,63	1,86	1,79	1,82
	4.9	2,03	2,05	2,04	1,94	2,02	2,13
	4.10	1,96	2,01	1,98	2,08	1,95	1,96
	4.11	1,92	1,90	1,96	1,81	1,89	1,98
	4.12	2,50	2,22	2,52	2,19	2,49	2,24
Erlernbarkeit	5.1	1,75	1,96	1,85	1,92	1,68	2,00
	5.2	2,75	2,60	2,71	2,64	2,78	2,58
	5.3	2,04	2,20	2,10	2,28	1,98	2,13
	5.4	2,12	2,26	2,31	2,17	1,97	2,33
	5.5	2,39	2,28	2,34	2,19	2,43	2,36
	5.6	2,19	2,43	2,21	2,42	2,17	2,44
	5.7	2,14	2,31	2,06	2,58	2,20	2,09

²⁴⁵ Ob eine Likert-Skala ein ordinal- oder intervallskaliertes Merkmal darstellt, ist strittig. Die übliche Forschungspraxis verzichtet auf eine empirische Überprüfung der Skalenaxiomatik. Es wird die Annahme getroffen, das jeweilige Merkmal würde auf einer Intervallskala gemessen. Unter dieser Annahme wurden auch für die Items des Fragebogens Mittelwerte berechnet. Vgl. Bortz/Döring (2003): S. 74.

²⁴⁶ Die Befragung der Studierenden erfolgte im Wintersemester 2004-05, im Sommersemester 2005 und im Wintersemester 2005-06 an der Leopold-Franzens-Universität Innsbruck. Insgesamt wurden 197 Studierende befragt; unter ihnen waren 116 Semiramis- und 91 SAP-Anwender.

²⁴⁷ Ab dem Wintersemester 2005-06 wurde Semiramis 4 in der Lehre eingesetzt. Die Mittelwerte beziehen sich auf eine Erhebung von 48 Studierenden in den Semiramis-Kursen und 36 Studierenden im SAP-Kurs an der LFU Innsbruck.

²⁴⁸ Semiramis 2 wurde im Wintersemester 2004-05 sowie im Sommersemester 2005 in den Semiramis-Kursen verwendet. In diesen beiden Semestern wurden 68 Studierende mit Semiramis- und 45 mit SAP-Kenntnissen befragt.

Tabelle 6: Median und Modus der Bewertung im Vergleich zwischen SAP und Semiramis

Untersuchungs- bereiche ²⁴⁹		Semiramis - SAP ²⁵⁰				Semiramis 4 - SAP ²⁵¹				Semiramis 2 - SAP ²⁵²			
		Semiramis		SAP		Semiramis		SAP		Semiramis		SAP	
		x_{med}	x_{mod} ²⁵³	x_{med}	x_{mod}	x_{med}	x_{mod}	x_{med}	x_{mod}	x_{med}	x_{mod}	x_{med}	x_{mod}
Emotion	1.1	2	2	2	3	2	2	2	3	2	2	2	2
	1.2*	3	2	2	2	2,5	2	2	2	3	3	2	1
	1.3*	2	1	2	2	2	3	2	2	2	1	2	2
	1.4	2	2	3	4	2	1	3	4	2	2	4	4
	1.5	2	2	3	3	2	2	3	3	3	2	3	3
	1.6	2	2	3	3,5	2,5	2	3	3	2	2	4	4
	1.7	2	2	3	2	2	2	3	2	2	3	3	2,5
	1.8*	2	2	3	2	2	2	3	2	2	2	3	2
	1.9	2	2	3	3	2	2	3	2	2	2	3	3
	1.10*	3	4	4	5	3	4	3	3,5	3	3	4	5
	1.11*	2	2	3	3,5	2	2	3	3	2	2	3	4
	1.12	2	1,5	1	1	2	1,5	2	1	2	1	1	1
	1.13	2	2	3	3	2	2	3	3	2	2	3	3
Software- steuerung	2.1	2	2	3	3	3	3	3	3	2	2	3	3
	2.2*	3	2	3	2	3	1/4	2,5	2	2,5	2	3	3
	2.3	2	2	2	2	2	2	2	2	2	2	3	2,5
	2.4	3	3	3	3	3	3	3	2,5	2	2	3	3
	2.5	2	1	3	2	2	1	2,5	1	2	1	3	2
	2.6	3	4	3	4	3	3	3	3	3	4	3	4
	2.7	3	3	3	3	2	2	3	3	3	3	3	3
	2.8	4	4	3	3	3	3	3	3	4	4	4	4
Effizienz	3.1*	3	2,5	2	2	3	2	2	2	3	4	2	1
	3.2*	2	3	2	2	2	3	2	2	2	2	2	2
	3.3	2	2	2	2	2	2	2	2	3	2	2	1
	3.4	2	2	2	2	2	2	2	2	2	1	3	2
	3.5	2	2	3	2	2	2	2	2	2	1,5	3	4
	3.6*	3	2	3	3	3	3	3	3	2	2	3	3
	3.7	1,5	1	2	1	2	1,5	1	1	1	1	2	1
	3.8	2	2	3	4	3	3	3	4	2	2	4	4
	3.9	2	2	3	3	2	2	3	3	2	2	3	3
	3.10	3	3	2	2	3	3	2	2	3	3	3	2,5
	3.11	2	2	3	3	2	2	3	3	2	2	3	3
	3.12	2	2	2	2	2	2	2	2	2	2	3	3
	3.13*	2	2	2	2	2	2	2	2	2	2	3	3
System- unterstützung	4.1	2	1,5	3	2	2	2	3	2	2	1	3	2
	4.2	3	3	3	3	3	2	3	3	3	3	3	3
	4.3	2	2	3	2	3	3	2	2	2	2	3	3
	4.4*	3	3	3	3	2	3	3	3	3	3	3	3
	4.5	2	1	3	2	3	3	2	2	2	1	3	2
	4.6*	2	1	3	4	2	1	3	3	2	1	2	1/4
	4.7	3	3	3	3	3	3	3	3	3	3	3	3
	4.8	2	2	2	2	2	2	2	2	2	2	2	2
	4.9	2	2	3	3	2	2	3	4	2	2	3	3
	4.10	3	3	3	3	3	2	3	3	3	3	3	2
	4.11	3	5	3	5	3	5	3	2/5	3	1	4	4
4.12*	3	3	3	3	3	3	3	3	2	2	3	3	
Erlernbarkeit	5.1	2	2	3	3	2	2	2,5	2	2	2	3	3
	5.2*	3	4	4	5	3	4	4	5	3	4	4	5
	5.3	3	3	4	4	3	3	3	3	3	3	4	4
	5.4*	3	3	4	4	3	3	3,5	4	3	3	4	4
	5.5*	3	3	3	3	4	4	3	2	3	3	3	3
	5.6*	2	2	3	2	3	3	2,5	2	2	2	3	2
	5.7	2	2	3	3	2	2	3	3	2	3	3	3

²⁴⁹ Bei der Auswertung jener Statements, welche negativ formuliert sind (*), beispielsweise Statement 1.2 („Ich würde es nicht mögen, diese Software jeden Tag zu benutzen“), wurde die Polung beachtet. Die Bewertung der negativ formulierten Statements wurde nach der Formel $6 - x$ umgerechnet. Ergab die Verteilung einen Median von 5, so ist hier ein Median von 1 ausgewiesen. Gleiches gilt für den Modus. Es gilt folglich, je kleiner die Ausprägung von Median bzw. Modus, desto besser hat die Software bezüglich des jeweiligen Statements abgeschnitten. Vgl. Hamborg/Gediga/Hassenzahl (2003): S. 175f.

²⁵⁰ Die Befragung der Studierenden erfolgte im Wintersemester 2004-05, im Sommersemester 2005 und im Wintersemester 2005-06 an der Leopold-Franzens-Universität Innsbruck. Insgesamt wurden 197 Studierende befragt; unter ihnen waren 116 Semiramis- und 91 SAP-Anwender.

²⁵¹ Ab dem Wintersemester 2005-06 wurde Semiramis 4 in der Lehre eingesetzt. Die Mediane und Modi beziehen sich auf eine Erhebung von 48 Studierenden in den Semiramis-Kursen und 36 Studierenden im SAP-Kurs an der LFU Innsbruck.

²⁵² Semiramis 2 wurde im Wintersemester 2004-05 sowie im Sommersemester 2005 in den Semiramis-Kursen verwendet. In diesen beiden Semestern wurden 68 Studierende mit Semiramis- und 45 mit SAP-Kenntnissen befragt.

²⁵³ Ist eine Verteilung bimodal, so sind beide Modi angegeben. Konzentrieren sich zwei Maxima in einer Verteilung auf benachbarte Kategorien, so wird der Modus als mittlerer Wert zwischen den beiden am häufigsten besetzten Kategorien gebildet.

VIII Anhang D: Signifikanztests

Tabelle 7: Mann-Whitney-U-Test für 2 unabhängige Stichproben²⁵⁴ - SAP- und Semiramis-Anwender

		Semiramis - SAP			Semiramis 4 - SAP			Semiramis 2 - SAP		
		H ₀ verwerfen ²⁵⁵	α *	„Testsieger“	H ₀ verwerfen	α *	„Testsieger“	H ₀ verwerfen	α *	„Testsieger“
Emotion	1.1	-	,149	-	-	,198	-	-	,415	-
	1.2	-	,109	-	-	,499	-	-	,126	-
	1.3	-	,105	-	-	,612	-	-	,097	-
	1.4	**	,000	Semiramis	**	,007	Semiramis	**	,000	Semiramis
	1.5	-	,507	-	-	,736	-	-	,653	-
	1.6	**	,000	Semiramis	*	,043	Semiramis	**	,000	Semiramis
	1.7	**	,003	Semiramis	-	,087	-	*	,011	Semiramis
	1.8	**	,000	Semiramis	*	,011	Semiramis	**	,000	Semiramis
	1.9	**	,000	Semiramis	-	,058	-	**	,000	Semiramis
	1.10	-	,165	-	-	,824	-	**	,000	Semiramis
	1.11	**	,000	Semiramis	**	,001	Semiramis	*	,025	Semiramis
	1.12	-	,052	-	*	,015	SAP	-	,568	-
	1.13	**	,000	Semiramis	*	,044	Semiramis	**	,000	Semiramis
Software- steuerung	2.1	**	,000	Semiramis	**	,008	Semiramis	**	,007	Semiramis
	2.2	-	,650	-	-	,835	-	-	,691	-
	2.3	-	,184	-	-	,964	-	-	,085	-
	2.4	**	,001	Semiramis	-	,322	-	**	,000	Semiramis
	2.5	**	,000	Semiramis	*	,014	Semiramis	**	,003	Semiramis
	2.6	-	,365	-	-	,842	-	-	,303	-
	2.7	-	,361	-	*	,031	Semiramis	-	,613	-
	2.8	-	,795	-	-	,915	-	-	,670	-
Effizienz	3.1	**	,000	SAP	-	,108	-	**	,000	SAP
	3.2	-	,411	-	-	,633	-	-	,534	-
	3.3	**	,000	SAP	*	,033	SAP	**	,005	SAP
	3.4	**	,000	Semiramis	-	,375	-	**	,000	Semiramis
	3.5	*	,018	Semiramis	-	,894	-	**	,004	Semiramis
	3.6	*	,017	Semiramis	-	,684	-	**	,009	Semiramis
	3.7	-	,091	-	-	,934	-	*	,031	Semiramis
	3.8	**	,000	Semiramis	*	,039	Semiramis	**	,001	Semiramis
	3.9	**	,000	Semiramis	**	,010	Semiramis	**	,000	Semiramis
	3.10	-	,137	-	**	,001	SAP	-	,227	-
	3.11	**	,000	Semiramis	**	,001	Semiramis	**	,000	Semiramis
	3.12	**	,005	Semiramis	-	,533	-	**	,002	Semiramis
	3.13	-	,110	-	-	,635	-	*	,011	Semiramis
System- unterstützung	4.1	**	,003	Semiramis	-	,264	-	**	,004	Semiramis
	4.2	-	,259	-	-	,138	-	-	,670	-
	4.3	*	,033	Semiramis	-	,629	-	**	,002	Semiramis
	4.4	**	,000	Semiramis	**	,002	Semiramis	*	,022	Semiramis
	4.5	**	,001	Semiramis	-	,973	-	**	,000	Semiramis
	4.6	**	,001	Semiramis	-	,062	-	*	,015	Semiramis
	4.7	-	,235	-	-	,059	-	-	,864	-
	4.8	-	,475	-	-	,721	-	-	,523	-
	4.9	**	,000	Semiramis	**	,003	Semiramis	**	,004	Semiramis
	4.10	-	,958	-	-	,862	-	-	,830	-
	4.11	-	,455	-	-	,996	-	-	,378	-
	4.12	**	,002	Semiramis	-	,814	-	**	,000	Semiramis
Erlernbarkeit	5.1	**	,000	Semiramis	-	,231	-	**	,000	Semiramis
	5.2	**	,004	Semiramis	-	,094	-	*	,017	Semiramis
	5.3	*	,017	Semiramis	-	,803	-	**	,004	Semiramis
	5.4	-	,091	-	-	,787	-	*	,042	Semiramis
	5.5	-	,772	-	-	,297	-	-	,199	-
	5.6	-	,078	-	-	,955	-	*	,032	Semiramis
	5.7	**	,007	Semiramis	**	,001	Semiramis	-	,390	-

²⁵⁴ Auf den beiden Signifikanzniveaus von $\alpha = 0,05$ und $\alpha = 0,01$ soll die folgende Homogenitätshypothese (H₀) überprüft werden: Die Bewertungsverteilungen der Statements sind unter Semiramis und SAP-Anwendern homogen, d.h. für die beiden Anwendergruppen gibt es keinen Unterschied in der Bewertung der beiden ERP-Softwaresysteme. Die Nullhypothese wird verworfen, sobald $\alpha^* \leq \alpha$ gilt, wobei α^* das empirische und α das vorgegebene Signifikanzniveau beschreibt. Vgl. Eckstein (2000): S. 162f.

²⁵⁵ Ein Stern (*) gibt an, dass die Nullhypothese verworfen und die Alternativhypothese akzeptiert werden kann, d.h. es besteht ein signifikanter Unterschied in der Bewertung des jeweiligen Statements zwischen den beiden Anwendergruppen. Wird die Nullhypothese auf einem Signifikanzniveau von $\alpha = 0,01$ verworfen, so spricht man von hoch signifikanten Unterschieden zwischen den beiden Gruppen; diese sind mit zwei Sternen (**) gekennzeichnet.