

Usability Engineering als Wettbewerbsvorteil für die Investitionsgüterindustrie

Claus Oetter

Fachverband Software / VDMA
Lyoner Strasse 18
60528 Frankfurt am Main
claus.oetter@vdma.org
www.vdma.org/software

Abstract

Software bestimmt immer mehr die Funktionalitäten von Maschinen und Anlagen. Damit nehmen zwangsläufig auch die Bedienschnittstellen zu. Leider wird jedoch bei der Softwareentwicklung häufig nicht ausreichend auf die Gestaltung von anwendergerechter Bedienung geachtet. Die Folgen sind höhere Ausfallzeiten durch

Fehlbedienung sowie ein Sinken der Qualität in der Produktion. Die beiden Disziplinen Informatik und Ingenieurwissenschaft müssen besser zusammenwachsen und voneinander lernen, um den heutigen Ansprüchen an moderne Produktionssysteme gerecht zu werden.

Keywords

Usability-Engineering, Software-Engineering, Software-Qualität, Prototyping, Interdisziplinäre Projekte, .

1.0 Einleitung

In den Unternehmen, Produkten und Anlagen der Investitionsgüterindustrie ist Software nicht mehr wegzudenken. Software bestimmt die Funktionalität und den Kundennutzen entscheidend. Ein erheblicher Anteil der Wertschöpfung entfällt heute auf die Software. Europa bezieht zwar als Nettoimporteur weitaus mehr klassische Software für Betriebssysteme, Office und Netzwerke, als es selbst exportiert. Dennoch ist Europa eine höchst erfolgreiche, in manchen Bereichen weltweit führende High-Tech- und Softwareschmiede. Mindestens in drei Marktsegmenten lässt sich ein Vorsprung oder ein Aufholen europäischer Unternehmen gegenüber amerikanischen und japanischen Konkurrenten nachweisen: Maschinen- und Anlagenbau mit High-Tech-Komponenten, Telekommunikation und Unternehmenssoftware. Im Maschinen- und Anlagenbau hat Europa den traditionellen Vorsprung durch die Weiterentwicklung der Produkte unter Einsatz der Informationstechnologie konsequent ausgebaut. Europa zählt bei

Software als Bestandteil von Konsumgütern, Maschinen und Anlagen zu den massiven Exporteuren. Dort, wo klassische Technik mit Informations- und Kommunikationstechnik verknüpft wird, hat Europa eine führende Rolle übernommen. Diesen Vorsprung gilt es zu halten durch einen konsequenten Einsatz von Qualität steigernden und Kosten senkenden Maßnahmen bei der Softwareentwicklung. Schaut man sich die Durchdringung von Vorgehensmodellen in den frühen Softwareprojekt-Phasen an, so wird deutlich, das sich ca. 60 % der deutschen Unternehmen auf CMM Level 1 befinden. Dies sind Zahlen, die ein sofortiges Umdenken in den betroffenen Unternehmen notwendig machen. Im Folgenden werden einige Teilaspekte erörtert und speziell für die Investitionsgüterindustrie beleuchtet.

2.0 Stellenwert früher projektphasen

Die Interessenslage von IT-Experten und Anwendern unterscheidet sich. Während Nutzer an Programmen interessiert sind, die einfach zu bedienen sind und ihnen die Arbeit erleichtern, steht für Informatiker eher eine elegante und innovative Programmierung im Vordergrund. Anwender und IT-Experten sprechen keine gemeinsame Sprache. Man versteht nur wenig vom Fachgebiet und der Begriffswelt des Anderen. Gerade bei sehr großen Projekten, die an verschiedenen Standorten für internationale Kunden abgewickelt werden, sind die Gefahren eines Projekt-Desasters sehr groß. Dies liegt zum einen daran, dass die frühen Projektphasen gerade in der Investitionsgüterindustrie noch nicht den Stellenwert haben, den sie haben müßten. Zum anderen fehlt sehr oft der richtige Bezug zur Software. In einer Industrie, wo mit Stahl gearbeitet wird, ist ein Leichtgewicht wie Software schlecht zu bewerten. Die frühen Projektphasen sind das Fundament einer guten und effizienten Software-Entwicklung. Für die eigentliche Software-Entwicklung

müssen Regeln aufgestellt, befolgt und überwacht werden. Auch wenn dies bei vielen Programmierern auf wenig Akzeptanz stößt, ist ein strukturiertes Vorgehen bei der Software-Entwicklung in IT-Projekten unumgänglich.

2.1 Tools - ein Allheilmittel?

Viele Unternehmen versuchen, die Produktivität und Qualität ihrer Softwareentwicklung durch den Kauf von CASE- oder Konfigurationsmanagement-Tools zu verbessern. Dabei wird vergessen, dass dokumentierte Prozesse, die eine Nachvollziehbarkeit von Analyse- und Designphase gewährleisten und in denen zu fest definierten Zeitpunkten Qualitätskontrollen stattfinden, in erheblichem Maße die Qualität der Softwareentwicklung verbessern.

3.0 Usability Engineering bei interdisziplinären Projekten

Die Anforderungen an die IT-Landschaft als integraler Bestandteil eines Produktionssystems sind extrem hoch. Aber nur der Einsatz der entsprechenden Software reicht nicht aus. Dies ist wie bei einem Hausbau: Nur im Team und mit guten Plänen gelingt es auch, ein stabiles Gebäude zu erstellen. Genau das Gleiche gilt im Bereich der Entwicklung tragfähiger Software-Systeme im Maschinen- und Anlagenbau. Es gibt kaum eine andere Disziplin, bei der so viele Fakultäten so eng zusammen arbeiten müssen. Es ist der richtige Mix aus Maschinenbau, Elektrotechnik und Software (auch Mechatronik genannt), der zum Erfolg führt. Hier ist Software-Engineering das Zauberwort. Interdisziplinäres Denken, Reden und Handeln bestimmt den Erfolg der Entwicklungsmannschaft. Grundsätzlich stellen mechatronische

Systeme die gleichen Anforderungen an den Engineeringprozess wie an Standard-IT-Programme. Hinzu kommt allerdings, dass hier natürlich Mechanik mit im Spiel ist, und somit eine Fehlbedienung oder ein Fehlverhalten der Software nicht nur zum Kippen eines Bits führt, sondern eventuell zum mechanischen Crash.

Sicherheitskritische Funktionalitäten müssen sehr viel besser getestet und gegen beabsichtigtes oder unbeabsichtigtes falsches Bedienen gesichert sein. Zur späteren Auswertung einer Fehlfunktion müssen somit nicht nur die Bedienhandlungen protokolliert werden sondern auch der Zustand der mechanischen Komponenten abrufbar sein.

3.1 Interdisziplinäre Teamauswahl

Der wichtigste Aspekt für das erfolgreiche Abwickeln eines Softwareprojektes ist die ganzheitliche Sichtweise auf die Technik und ihre Beziehung zu dem Unternehmen. Um erfolgreiche Projekte abwickeln zu können, müssen die drei Schlüsselfaktoren Organisation, Mitarbeiter sowie die Entwicklungsprozesse optimal aufeinander abgestimmt sein. Die Auswahl der Teammitglieder sollte mit größter Sorgfalt erfolgen, da hier ein zentraler Erfolgsfaktor für den reibungslosen Projektverlauf liegt. Zusätzlich zum fachlichen Wissen sollten die Mitarbeiter in der Lage sein, funktionsübergreifend und in Analogien zu denken. In der heutigen Zeit wird es immer wichtiger, interdisziplinär und bereichsübergreifend arbeiten zu können. Eine gute Kommunikation ist eine Grundvoraussetzung für ein funktionierendes Team. Doch muß es auch Regeln geben, an die sich jeder halten muß. Vor allem in der Analyse- und Designphase schleichen sich hier Fehler ein, die später nicht mehr oder nur schwer zu korrigieren sind. Durch immer kürzer werdende Entwicklungszyklen

bei gleichzeitig stetig wachsendem Qualitätsanspruch sehen sich die Entwicklungsteams einem extremen Zeitdruck gegenüber, unter dem sehr oft vor allem die frühe Projektphase leidet.

4. Grundlagen der Softwarergonomie

Historisch gesehen bekamen die Maschinen und Anlagen mit Einführung der Prozessrechner eine neue Qualität. Sie wurden flexibler und leistungsfähiger. Während früher die Bedienelemente direkt am Ort des Geschehens waren, sind sie jetzt auf einer zentralen Bedieneinheit konzentriert. Durch die Kompaktheit der Darstellung wurde dem Bediener die Begreifbarkeit der Maschine und des Prozesses erschwert. Konnten Kurvensteuerungen noch »durchschaut« werden, so geben abstrahierte Darstellungen mitunter nicht nur dem Bediener Rätsel auf. Erschwerend kommt hinzu, dass »Software« eher der Maschine angepasst ist. Der Bediener, der die Maschine sicher und zuverlässig beherrschen muss, wird häufig vergessen oder überfordert. Grundsätzliche Wahrnehmungsmechanismen und Verhaltensweisen des Menschen werden missachtet, die Bedienung wird fehlerträchtig und frustrationsfördernd – die Maschine wird unbedienbar!

4.1 Beherrschung hoher Komplexität

Durch die Vielfalt software-technisch realisierbarer Programme wird der Computer zu einem äußerst vielfältigen Instrument. Während beim traditionellen Handwerkzeug eine unmittelbare und damit auch einfach verständliche Verbindung zwischen der menschlichen Aktion und dem Effekt gegeben ist, ist bei Maschinen nicht immer ein unmittelbarer Zusammenhang zwischen den Bedieneingriffen und dem bewirkten Effekt zu erkennen. Insbesondere ist bei ihrer Bedienung die Kenntnis bzw. ein Gefühl über die Dynamik der

Aussagen der Testwähler. Als Erfolg kann daher vorwiegend die hohe Akzeptanz und Zufriedenheit gegenüber dem Wahl-User-Interface und dem elektronischen Wahlablauf als solchen angesehen werden, die sich auch bei den Wählern zeigte, die vor der Testwahl sehr unsicher waren und zum Teil sogar ängstlich dem Wahlsystem gegenüber standen. Sie beurteilten es insgesamt als intuitiv verständlich, selbsterklärend und vertrauenserweckend.

4.0 Literatur:

- 1 A. Beu, C. Görner, C & F. Koller (Hg.): Der Bildschirmarbeitsplatz. Softwareentwicklung mit DIN EN ISO 9241, Berlin 1999.
- 2 Burmester, M.: Warum muss Software gebrauchstauglich sein? Gesellschaft Arbeit und Ergonomie-Online e.V., Stand 2000, www.sozialnetz.de gelesen am 16.06.2003.
- 3 Gediga, G. & Hamborg, K.-C.: Ergonomische Evaluation von Software: Methoden und Modelle im Software-Entwicklungsprozess. In: Zeitschrift für Psychologie 210 1 (2002), 40-57.

»Es ist erlaubt digitale und Kopien in Papierform des ganzen Papers oder Teilen davon für den persönlichen Gebrauch oder zur Verwendung in Lehrveranstaltungen zu erstellen. Der Verkauf oder gewerbliche Vertrieb ist untersagt. Rückfragen sind zu stellen an den Vorstand des GC-UPA e.V. (Postfach 80 06 46, 70506 Stuttgart).
Proceedings of the 2nd annual GC-UPA Track Paderborn, September 2004
© 2004 German Chapter of the UPA e.V.«

