

Entscheidungsrahmen für die softwareergonomische Einobjekt-Evaluation

Sascha STOWASSER

*Institut für Arbeitswissenschaft und Betriebsorganisation, Universität Karlsruhe (TH)
Kaiserstraße 12, D-76131 Karlsruhe*

Kursfassung. Der Beitrag beschreibt die systematische Formulierung eines Rahmenkonzepts zur Beschreibung von Softwareevaluationen auf einer Metaebene. Das Rahmenkonzept enthält relevante Merkmale von Evaluationen und beschreibt die Zusammenhänge zwischen Anforderungen (Soll-Zustand) und den Ausprägungen (Ist-Zustand) der Softwarekomponenten. Die Bewertung wird auf Grundlage einer metrischen Intervallskala, dem Zielerreichungsgrad "anforderungsbezogene Erfüllung" durchgeführt.

Schlüsselwörter: Softwarebewertung, Softwareergonomie, Evaluationsforschung

1. Hintergrund und Zielsetzung von Einobjekt-Evaluationen

Eine Mensch-Rechner-Schnittstelle, die nicht ergonomisch gestaltet ist, führt oft dazu, dass sich die Benutzer über undurchschaubare Programmabläufe, schwer erlernbare Handhabung, unzureichende Funktionalität, Verkomplizierung einfacher Vorgänge und Ausbleiben der erhofften Vorteile durch die Einführung der Software beklagen. Um die Konformität der Software mit vorliegenden Gestaltungsnormen und –empfehlungen bereits bei der Software-Entwicklung oder –beschaffung zu überprüfen, ist eine methodische Konzeption und Durchführung von Evaluationen unerlässlich.

Im Rahmen von softwareergonomischen Einobjekt-Evaluationen ist es das Ziel, den Ist-Zustand eines Evaluationsobjektes zu untersuchen, mit Soll-Zuständen eines Bezugssystems (z.B. softwareergonomische Normen, Pflichtenhefte des Anwenders) zu vergleichen und die Zielerfüllung der Gestaltung zu bewerten. Aufgrund der Bewertung anhand einer festen Wertvorstellung kann die Einobjekt-Evaluation auch als absolute Vergleichsaufgabe bezeichnet werden. Anlass der Einobjekt-Evaluation kann seitens des Herstellers die Qualitäts- oder Normkonformitätsprüfung eines prototypischen oder bereits fertig gestellten Softwaresystems sein. Aus Anwender- und Benutzersicht wird eine Einobjekt-Evaluation hauptsächlich bei der Eignungsprüfung einer Software und gleichfalls bei Normkonformitätsprüfungen durchgeführt.

Im Gegensatz zur Einobjekt-Evaluation zielt die softwareergonomische Mehrobjekt-Evaluation nicht nur auf die Bewertung eines Evaluationsobjektes ab, sondern untersucht vergleichend mehrere Evaluationsobjekte. Ziel hierbei ist es, qualitative oder quantitative Unterschiede zwischen den einzelnen Softwarerealisierungen aufzudecken. Als Evaluationsobjekte können hier sowohl Prototypen während des Entwicklungsprozesses als auch bereits vollständig realisierte Softwareprodukte beurteilt werden.

2. Entscheidungsrahmen zur softwareergonomischen Einobjekt-Evaluation

2.1 Evaluationsobjekt und Objektkomponenten

Als Gegenstand der softwareergonomischen Evaluation gilt das Evaluationsobjekt (vgl. Abbildung 1). In der Regel wird das zu betrachtende Evaluationsobjekt in verschiedene Objektkomponenten zerlegt (Englisch 1993). So definiert beispielsweise das IFIP-Modell für Benutzungsschnittstellen die folgenden Komponenten einer Software (vgl. Dzida 1988, S. 15 ff.): Ein- und Ausgabekomponente, Dialogkomponente, Werkzeugkomponente und/sowie Organisationskomponente. Wenn auch das globale Konzept der Software durch dieses Modell bereits unterteilt wurde, so reicht diese Differenzierung nicht aus, um ein konkretes Softwareprodukt hinsichtlich der softwareergonomischen Kriterien zu untersuchen (Oppermann u.a. 1992, S. 57). Eine weitere Differenzierung der Komponenten ist daher notwendig.

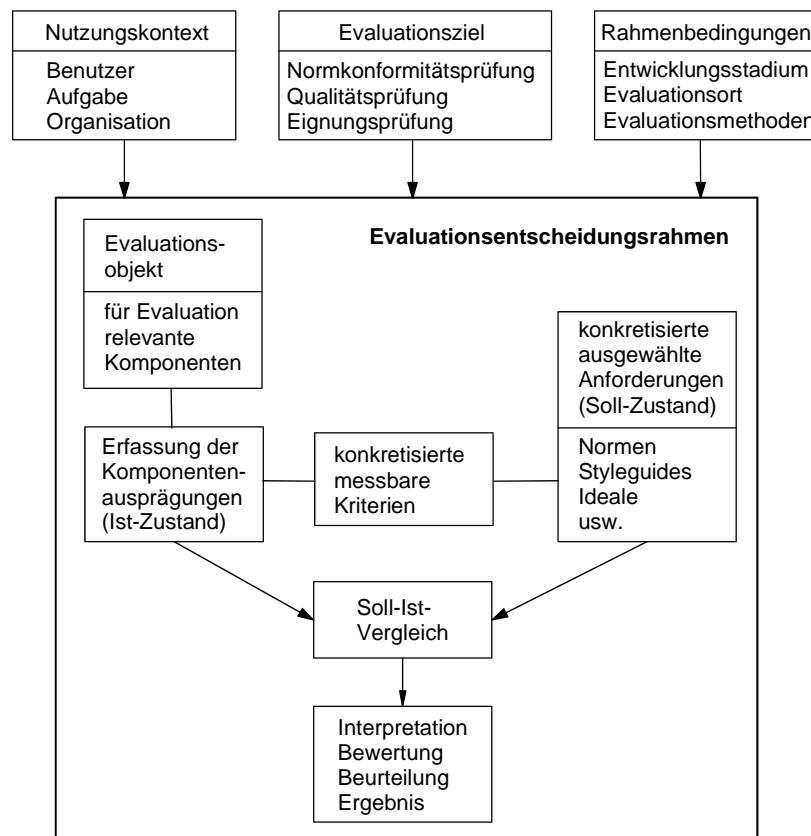


Abbildung 1: Zusammenhänge der Einobjekt-Evaluation

Einige Modellansätze zur Beschreibung von Software sind daher in mehrere Hierarchiestufen geschichtet, womit es ermöglicht wird, zur detaillierten Beschreibung der Software geeignete Unterkomponenten zu bilden. Zur konkreten Evaluation der Software wird Bezug zu denjenigen Komponenten genommen, die nicht weiter unterteilt werden. Auf diesem Niveau finden dann konkrete Bewertungen über charakteristische Ausprägungen der Software auf diesem Niveau statt. Die nicht weiter unterteilten Komponenten werden deshalb auch charakterisierbare Komponenten genannt. Der Detaillierungsgrad der Evaluation ist somit von der Charakterisierbarkeit der Komponenten und damit von der verwendeten Komponenteneinteilung abhängig.

2.2 Kriterienableitung und -operationalisierung

Die Bewertung des Evaluationsobjekts basiert auf einem Evaluationsablauf, nämlich auf der Auswahl eines Kriterienrahmens mit den zugehörigen softwareergonomischen Kriterien (Anforderungen), der Zuordnung von Maßen (auch Messgrößen genannt) zu den messbaren Kriterien, der Messung der relevanten Komponentenausprägungen und der Beurteilung des Vergleichs zwischen Ist- und Soll-Zustand. Als Kriterienrahmen zur softwareergonomischen Evaluation hat sich die Benutzungsfreundlichkeit nach DIN EN ISO 9241-10 bzw. die Gebrauchstauglichkeit nach DIN EN ISO 9241-11 bewährt. Weitere (durchaus für spezielle Evaluationszwecke eigens zusammengestellte) Kriterienrahmen sind in großer Vielzahl möglich und können aus Normen, Styleguides sowie Idealvorstellungen gebildet werden.

Ein Kriterienrahmen ist eine (meist) in mehrere Ebenen hierarchisch untergliederte Zusammenstellung von softwareergonomischen Kriterien zur Evaluation der Software. Die Kriterien der ersten Ebene stellen sog. Zentralkriterien dar, wie z.B. die in DIN EN ISO 9241-11 definierten Kriterien Effektivität, Effizienz und Zufriedenheit oder die Dialoggestaltungskriterien der DIN EN ISO 9241-10 (z.B. Erwartungskonformität, Steuerbarkeit usw.). Für eine konkrete Evaluation weisen diese Kriterien eine zu große Generalität auf. Sie müssen erst interpretiert und konkretisiert werden, um sie zu operationalisieren. Daher werden die Kriterien so weit in untergeordnete Kriterien zerlegt, bis eine Messbarkeit des Kriteriums gegeben ist, d.h. ihnen konkrete Maße zugeordnet werden können.

2.2 Vergleich zwischen Anforderung und Ist-Zustand an ein Evaluationsobjekt

Ziel des Soll-Ist-Vergleiches einer Einobjekt-Evaluation ist es, Unterschiede oder Übereinstimmungen zwischen den zu vergleichenden Elementen, d.h. den Anforderungen an die relevanten Softwarekomponenten (Zielbezug) und den Ist-Zuständen (Objektbezug) dieser Komponenten zu ermitteln. Hierzu werden Ausprägungen und Anforderungen mittels einer Vergleichsoperation in Bezug gesetzt. Beim Soll-Ist-Vergleich einer Einobjekt-Evaluation können zwei Vergleichsprozesse unterschieden werden:

- Vergleich des Ist-Zustands mit Kriterienanforderungen aus Normen, Styleguides, Anforderungsanalysen und Pflichtenheften. Hierbei wird beurteilt, ob Forderungen bei der Softwaregestaltung eingehalten wurden.
- Vergleich des Ist-Zustands mit Kriterienanforderungen aus Idealvorstellungen, die ein Evaluator z.B. von bestimmten Realisierungsformen besitzt und die in geeigneter Weise operationalisiert sind. Hierbei wird beurteilt, ob die Software Schwachstellen besitzt und ein Evaluator Verbesserungsvorschläge bzw. bessere Realisierungsmöglichkeiten erkennt.

Durch Interpretation der Vergleichsergebnisse wird festgestellt, in welchem Ausmaß die zu untersuchende Objektkomponente die an sie gestellte Anforderung bezüglich aller als relevant betrachteten Kriterien erfüllt. Aufgrund der Vielzahl von Möglichkeiten zur Definition von softwareergonomischen Kriterien und zur Festlegung ihrer Anforderungen sind verschiedene Ansätze der Vergleichsbewertung und -interpretation denkbar.

2.3 Zielerreichungsgrad "Anforderungsbezogene Erfüllung"

Eine Möglichkeit zur Formalisierung des Vergleichs ist die Definition des Zielerreichungsgrades "anforderungsbezogene Erfüllung ZAE". Der Zielerreichungsgrad ordnet dem Vergleich zwischen Ausprägung und Anforderung einen (Prozent-)Wert zu, der das Verhältnis der Kriteriumsausprägung zur -anforderung in Bezug auf eine Softwarekomponente erfasst. Je näher dieser Zielerreichungsgrad sich 100 % nähert, desto mehr sind die Anforderungen an die Softwaregestaltung erfüllt. Diese generelle Beschreibung lässt sich in Abhängigkeit vom Skalentyp der Anforderungen bzw. der Kriterienmaße weiter ausformulieren.

Für intervall- bzw. verhältnisskalierte Kriterienanforderungen ist die folgende Berechnungsweise denkbar, welche z.B. bei der Bewertung von Ausführungszeiten verwendet werden kann. Hierbei wird eine obere zulässige Wertebereichsgrenze durch einen maximal zulässigen Grenzwert MXW definiert, der nicht überschritten werden sollte. Darüber liegende Werte sind unerwünscht und werden mit einem Zielerreichungsgrad von 0 % bewertet.

$$ZAE_{mi} = \begin{cases} 100\%, & \dots \text{ falls } AUS_{mi} \leq ANF_{mi} \\ \left(1 - \frac{AUS_{mi} - ANF_{mi}}{MXW_{mi} - ANF_{mi}}\right) \cdot 100\%, & \dots \text{ falls } ANF_{mi} < AUS_{mi} < MXW_{mi} \\ 0\%, & \dots \text{ falls } AUS_{mi} \geq MXW_{mi} \end{cases}$$

mit ANF_{mi} Anforderung an Komponente m bezüglich Kriterium i

AUS_{mi} Ausprägung des Kriterium i bei Komponente m

MXW_{mi} Maximal zulässiger Wert des Kriterium i bei Komponente m

3. Diskussion

Das vorliegende Konzept bietet einen Rahmen zur umfassenden Beschreibung möglicher Einobjekt-Evaluationen auf einer Metaebene. Die Bewertung auf Grundlage einer metrischen Intervallskala (hier: Zielerreichungsgrad "anforderungsbezogene Erfüllung") kommt der anzustrebenden Form einer quantitativen Evaluation entgegen. Die dargestellte Berechnungsformel zur Ermittlung des Zielerreichungsgrads ist nur ein Beispiel aus einer Vielzahl von Möglichkeiten zur Formulierung des Zielerreichungsgrads. In der Praxis sind vom Evaluator geeignete Berechnungsansätze in Abhängigkeit von der Evaluationsauslegung fallweise abzuleiten bzw. auszuwählen.

5. Literatur

1. DIN EN ISO 9241-10, Juli 1996: *Grundsätze der Dialoggestaltung*.
2. DIN EN ISO 9241-11, Januar 1999: *Anforderungen an die Gebrauchstauglichkeit*.
3. Dzida, W. 1983, Das IFIP-Modell für Benutzerschnittstellen, *Office Management*, 31, 6-8. (Sonderheft "Mensch-Maschine-Kommunikation")
4. Englisch, J. 1993, *Ergonomie von Softwareprodukten*. Mannheim u.a.: BI-Wissenschaftsverlag. (Angewandte Informatik, Band 5)
5. Oppermann, R., Murchner, B., Reiterer & H., Koch, M. 1992, *Software-ergonomische Evaluation*. Berlin, New York: Walter de Gruyter. (Mensch-Computer-Kommunikation, Band 5)