

Autor: Thomas Dermin (29.01.2016)

# Von der Usability-Relevanz in Gesellschaft und Wirtschaft

Usability (vgl. S. 5-9) ist als Kaufkriterium für KMU (kleine und mittlere Unternehmen) nicht nur wirtschaftlich relevant geworden (vgl. [MM+11], S. 140), sondern sie ergibt sich auch als eine notwendige Konsequenz der modernen Informationsgesellschaft. Die zunehmende technologische Komplexität ist für unsere menschlichen kognitiven Prozesse an der Belastbarkeitsgrenze angelangt und ein technologischer Beleg liefert das Passwortmanagement aus der Praxis. Der Benutzer von heute muss nicht nur einen Überblick über seine zahlreichen Systemaccounts (sowohl im Beruf als auch im Privatem) bewahren, sondern darüber hinaus über ein verantwortungsvolles Sicherheitsbewusstsein während der Passwörterstellung verfügen. Üblicherweise, wenn auch nicht allein als höchste Sicherheitsmaßnahme ausreichend, verwenden Softwaresysteme (insbesondere im Web) häufig ein Benutzername-Passwort-Wertepaar, um dem Benutzer gegenüber dem System zu authentifizieren. Das Resultat ist für den Endverbraucher eine nicht mehr überschaubare und leicht merkbare Menge an Benutzername-Passwort-Kombinationen. Aus pragmatischen Gründen verleitet dies dem technisch nicht versierten Benutzer dazu, beispielsweise ein bereits gemerktes Passwort mit hoher Sicherheit (eine hohe Anzahl an Stellen, Zeichen und entsprechender Komplexität) durch den Austausch spezifischer Zeichen wiederzuverwenden.

Ein praktisches Sicherheitsproblem.

*„[...] users tend to generate passwords that are easy to remember but also easy to crack“  
[VP+07]*

Auch die beliebte Passwortspeicherungsfunktion des Browsers wird gerne genutzt, ungeachtet dessen, dass dieser ohne ein sicheres Masterpasswort für andere Benutzer möglicherweise einsehbar ist. Diese Funktionalität basiert im Prinzip auf das Konzept des „Single-Sign-On“, welches als zentrale Authentifizierungsstelle nur noch eine Benutzername-Passwort-Kombination zur Freischaltung sämtlicher Dienste (z. B. eines Unternehmensnetzwerkes) ermöglicht und somit zu einer kognitiven Reduktion führt.

*„Web single sign-on (SSO) systems enable web users to leverage one single account on a service provider to sign on to multiple websites, reducing the number of passwords and registration information a user needs to manage.“ [SP+12]*

Für den Benutzer stellt dies eine deutliche kognitive Erleichterung dar und auch für die Administration ist durch die zentrale Passwortverwaltung eine Änderung mit weniger Aufwand verbunden. Wird jedoch das Masterpasswort kompromittiert, werden somit logischerweise alle darin befindlichen Benutzername-Passwort-Kombinationen obsolet. Das Single-Sign-On-Prinzip ist ein Beleg für den Sieg im Kampf „Sicherheit“ gegen „Usability“ (Merkbarkeit) von Passwörtern in der Praxis.

Ergänzend zur kognitiven Entlastung ist auch der sogenannte „Joy-of-Use“ (vgl. S. 10-11) ein Faktor, der zumindest genau dann in den Vordergrund des Käufers rückt, wenn die finanzielle und technische Frage bei der Auswahl ähnlicher Systeme keine notwendige Relevanz aufweist. In diesem Falle stellt die für den Benutzer wahrgenommene intuitive Bedienung während der Benutzung ein Mehrwert dar (vgl. [HV03], S. 7-21; zitiert nach [SB11], S. 92-93), was zum Kauf des Produktes führen kann.

Ein Beispiel dafür wäre die Auswahl zwischen einem Android-Smartphone und einem Apple iPhone. Natürlich kommen in der Praxis weitere Einflussfaktoren zusammen, wie z. B. die Unternehmenspolitik oder auch ethisch-moralische Werte, wenn Geld nicht mehr als entscheidendes Kaufkriterium im Vordergrund steht.

Aus der Perspektive eines Anforderungsanalysten stellt ein erfolgreich und sinnvoll angewandtes Usability Engineering (vgl. S. 12) als kontextausgerichtetes und nutzerfokussiertes Entwicklungsparadigma langfristig eine Kostenoptimierung im Sinne des „Total Cost of Ownership“ dar.

**Exkurs:** Argumente für Usability-Engineering liefert Kalbach wie folgt (vgl. [HV03], S.7-21; zitiert nach [SB11], S.92-93):

- **Erfolgssicherheit:** Die Korrektur aufgedeckter Usability-Probleme in der frühen Designphase ist bis zu zehnmals kostengünstiger, als in der späten Entwicklungsphase.
- **Zeitersparnis:** Während der Entwicklung werden problemlastige Alternativen (bei Designentscheidungen) vermieden und somit Zeit gespart. In der späteren Nutzung spart der Benutzer durch die Effizienz des Systems selbst Zeit ein.
- **Kostensparnis:** Aus der Erfolgssicherheit und Zeitersparnis folgt die implizite Kostensparnis.
- **Steigerung der Umsätze:** Produkte mit hoher Gebrauchstauglichkeit (engl. usability) können ein wichtiges Kaufargument darstellen. Dies kann den Umsatz des Unternehmens steigern.
- **Innovationspotenzial:** Durch interdisziplinäre Entwicklungsarbeit (z. B. durch Zusammenarbeit von Designern, Entwicklern und Geisteswissenschaftlern) werden eindimensionalen Sichten aufgebrochen und können durch den regen Austausch neue Ideen hervorbringen (Synergieeffekte).
- **Stressreduktion:** Alternative Produkte mit hoher Gebrauchstauglichkeit vermindern den Ärger bei der Benutzung (mentaler Systemvergleich) und können somit zu einer Stressreduktion führen. Ein funktional gleiches Produkt mit höherer Usability kann sich positiv auf die Produktwahrnehmung auswirken.

Je früher relevantere Anforderungen gesammelt werden, desto kostengünstiger ist die Entwicklung für die gesamte Projektlaufzeit (vgl. [S02], S. 3-5). Dabei werden nur notwendige Funktionalitäten und Qualitätseigenschaften eines Systems, passend zu den Benutzerzielen im seinem Nutzungskontext, als Anforderungen erhoben. Dies wirkt sich positiv sowohl auf die Effizienz der Entwicklungszeit (zielorientierter), als auch auf die spätere Zeitersparnis während der Nutzung aus.

Dies ist eine Win-Win-Situation für das Software-Unternehmen und den späteren Anwender. Doch nicht nur die Kosteneffizienz ist ein Argument für Usability Engineering. Durch ein „Andersdenken“ in der Entwicklung können neue Potenziale freigesetzt werden, indem bisher nicht beachtete Anforderungen ermittelt und umgesetzt werden, welche die Konkurrenz bisher vernachlässigt hat. Ähnlich wie beim „Design Thinking“ sind die nutzerzentrierten Aktivitäten des Usability-Engineerings entscheidend, um Produkte maßgeschneidert an die Bedürfnisse der jeweiligen Zielgruppe anzupassen. Ebenso bildet die interaktive Herangehensweise ein gemeinsames Methodencharakteristika. Insbesondere beim „Prototyping“ findet eine Prinzipien-Überschneidung statt (vgl. S. 13-14). Ideen werden basierend auf empirischen Erkenntnissen hin prototypisch umgesetzt und gegenüber dem Benutzer validiert (vgl. [RF13], S. 52-60). Hierfür ist eine positive Fehlerkultur, genau wie beim Design Thinking, notwendig. Aufgrund der ähnlichen Charakteristiken zum Design Thinking kann Usability Engineering ebenfalls als Innovationsmethode betrachtet werden (vgl. [B08], zitiert nach [RL14]).

Doch ist Usability-Engineering, im Gegensatz zur klassischen Softwareentwicklung, eine technische oder soziale Innovation? Nach Howaldt und Schwarz wird die soziale Innovation wie folgt definiert:

*„Eine soziale Innovation ist eine von bestimmten Akteuren (...) ausgehende intentionale (...) Neukonfiguration sozialer Praktiken in bestimmten Handlungsfeldern mit dem Ziel, Probleme oder Bedürfnisse besser zu lösen (...) als dies auf der Grundlage etablierter Praktiken möglich ist.“ „Das Neue vollzieht sich nicht im Medium technologischer Artefakte, sondern auf der Ebene sozialer Praktiken (des Regierens, Organisierens, Versorgens, Konsumierens, der Partnerschaft, der Verhandlung etc.).“ ([HoSc10], S. 89)*

Im Kern geht es bei der sozialen Innovation um eine Neukonfiguration sozialer Praktiken, um Probleme im bestimmten Kontext besser zu lösen als dies bisher methodisch der Fall war. Dies entspricht dem Wesen des Usability Engineering, denn hier wird die soziale Praxis der Softwareentwicklung geändert, in dem die Partizipation, Befragung, sowie Beobachtung von Nutzern die bisherige Vorgehensweise stark verändert. Die Probleme die durch Usability Engineering gelöst werden sind, aus der Sicht der Wirtschaft, zum einen die Kostenoptimierung und nachhaltige Innovationsstrategie von Softwareunternehmen. Zum anderen findet eine Reduktion der wahrnehmbaren Komplexität statt, in dem die Software sich auf die wesentliche Aufgabe des Benutzers im Kontext konzentriert und somit für Ihn eine kognitive Entlastung schafft.

### **Zusammenfassung**

Usability Engineering kommt dem Bedürfnis einer Komplexitätsreduktion (entstanden durch die moderne Informationsgesellschaft) mittels kognitiver Entlastung entgegen und reduziert durch die Aufgaben- sowie Kontextfokussierung unnötige Nutzungszeit. Außerdem kann der Stress sinken und die Freude während der Benutzung zunehmen. Für Softwareunternehmen kann durch die Effizienz eine Kostenoptimierung entstehen und durch Usability Engineering-Methoden wie z. B. Das Prototyping, die dem Design Thinking-Prinzipien sehr nahe kommen, können Innovationspotenziale ausgeschöpft werden. Perspektivisch können sich so Unternehmen durch eigene Innovationen nachhaltig auf dem globalen Markt stabilisieren. Somit ist es eine Win-Win-Situation, denn sowohl wirtschaftlich, als auch gesellschaftlich ist ein Mehrwert vorhanden. Usability Engineering ist nicht nur eine Methode, sondern viel mehr eine andere Denkweise, die eine „Neukonfiguration sozialer Praktiken“ in der Entwicklung einfordert. Ein Paradigma gemacht für die Zukunft.

## Quellen

- [SB11]: Florian Sarodnick, Henning Brau (2011): Methoden der Usability Evaluation. Wissenschaftliche Grundlagen und praktische Anwendung. 2, Hans Huber, Bern, 2011.
- [S02]: Alistair Sutcliffe (2002): User-Centred Requirements Engineering. Theory and Practice. Springer, London, 2002.
- [HV03]: Robert Baggen, Andreas Beu, Kai-Christoph Hamborg, Marc Hassenzahl, Sven Heinsen, Jens Heuer, Ralph Hinderberger, James Kalbach, Adriane Kempken, Olde Lorenzen-Schmidt, Matthias Müller-Prove, Kerstin Röse, Christian Scheier, Petra Vogt (2003): Usability praktisch umsetzen. Carl Hanser, 2003.
- [RF13]: Michael Richter, Markus Flückiger (2013): Usability Engineering kompakt. Benutzbare Produkte gezielt entwickeln. 3, Berlin, 2013.
- [SP+12]: San-Tsai Sun, Eric Pospisil, Ildar Muslukhov, Nuray Dindar, Kirstie Hawkey, Konstantin Beznosov (2012): Investigating User's Perspective of Web Single Sign-On: Conceptual Gaps, Alternative Design and Acceptance Model. In: ACM Transactions on Internet Technology, V, S. 35, 2012.
- [VP+07]: Kim-Phuong L. Vua, Robert W. Proctorb, Abhilasha Bhargav-Spantzelb, Bik-Lam (Belin) Taib, Joshua Cookb, E. Eugene Schultz (2007): Improving password security and memorability to protect personal and organizational information. In: International Journal Human-Computer Studies, S. 744–757, 2007.
- [HS10]: Jürgen Howaldt, Michael Schwarz (2010): Soziale Innovation – Konzepte, Forschungsfelder und -perspektiven. In: Howaldt, Jürgen; Jacobsen, Heike (Hrsg.): Soziale Innovation – Auf dem Weg zu einem postindustriellen Innovationsparadigma, S. 87-108, VS Verlag, Wiesbaden, 2010.
- [RL14]: Petra Schaper-Rinkel, Petra-Wagner-Luptacik (2014): Design Thinking. In: SI-DRIVE-Projekt. Theoretical approaches to Social Innovation – A Critical Literature Review, S. 97-103, 2014.
- [B08]: Tim Brown (2008): Design Thinking. In: Harvard Business Review, S. 84-92, 2008.
- [MM+11]: Institut für Mittelstandsforschung an der Universität Mannheim, Lehrstuhl für Wirtschaftsinformatik IV der Universität Mannheim, Fachhochschule Kaiserslautern, ERGOSIGN GmbH (2011): Gebrauchstauglichkeit von Anwendungssoftware als Wettbewerbsfaktor für kleine und mittlere Unternehmen (KMU). 2011, (Abschlussbericht des Forschungsprojekts).
- [BD+14]: Anna Butzin, Anna Davies, Dmitri Domanski, Steven Dhondt, Jürgen Howaldt, Christoph Kaletka, Alexander Kesselring, Ralf Kopp, Jeremy Millard, Peter Oeij, Dieter Rehfeld, Petra Schaper-Rinkel, Michael Schwarz, Anette Scoppetta, Petra Wagner-Luptacik, Matthias Weber (2014): SI-DRIVE-Projekt. Theoretical approaches to Social Innovation – A Critical Literature Review. TU Dortmund 2014.
- [HoSc10]: Jürgen Howaldt, Michael Schwarz (2010): Soziale Innovation im Fokus. Skizze eines gesellschaftstheoretisch inspirierten Forschungskonzepts. transcript, 2010.

# Usability

Der Begriff „Usability“ (dt. Gebrauchstauglichkeit) ist nach der internationalen Norm DIN EN ISO 9241-11 wie folgt definiert:

*„Usability ist das Ausmaß, in dem ein Produkt durch bestimmte Nutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und zufriedenstellend zu erreichen.“ (DIN EN ISO 9241-11, 1999, S.94; zitiert nach [SB11], S.37)*

Maßgeblich hierfür ist der Nutzer (man kann auch Zielgruppe sagen) und sein Nutzungskontext, der ein bestimmtes Ziel erreichen oder ein vorhandenes Problem (mit einem Produkt) lösen möchte. Notwendig sind also Informationen über relevante Nutzercharakteristiken, die über empirische Methoden gesammelt werden (vgl. [BB14]). Darüber hinaus sind allgemeine psychologische Kenntnisse über die menschlichen kognitiven Prozesse notwendig. Einerseits dienlich für die Befragung der Nutzer (vgl. [BB14]), andererseits um valide Testverfahren planen und durchführen zu können (vgl. [My14]).

Der Anwendungskontext wird im „Anwendungsrahmen für Usability“ nach DIN EN ISO 9241-11 untergliedert in den Punkten (vgl. DIN EN ISO 9241-11, 1999; zitiert nach [SB11], S. 38, Abb. 2)

- Benutzer,
- Arbeitsaufgabe,
- Arbeitsmittel und
- Umgebung.

Neben den Nutzercharakteristiken ist somit eine umfassende Analyse der durchzuführenden Arbeitsaufgaben (im Business: Geschäftsprozesse) zwingend erforderlich. Bezogen auf Websites ist unter „Arbeitsmittel“ nicht mehr nur der PC gemeint, sondern man kann davon ausgehen, dass der Nutzer mittlerweile auch über seine mobile Geräte (wie beispielsweise Smartphones oder Tablet-PC's) Inhalte unterwegs abrufen möchte („Umgebung“). Exemplarisch hierfür wäre ein bevorstehender Restaurantbesuch: Der Nutzer möchte nicht nur im Vorfeld die Adresse des Restaurants ermitteln, sondern darüber hinaus die Menükarte betrachten, um beispielsweise Gerichte vor dem Besuch gedanklich zu selektieren.

Mit „Effektivität“ ist die allgemeine Erreichbarkeit eines Ziels gemeint, entsprechend den Kenntnissen und Fähigkeiten des Benutzers in seinem Nutzungskontext. Beispielsweise wäre dies bei einem Online-Shop das Bestellen und der Einkauf von Waren. Die „Effizienz“ gibt Aussage darüber, mit welchem Aufwand dies geschieht. Quantifizierbar wäre beispielsweise die benötigte Zeit für den Abschluss einer erfolgreichen Bestellungen. Eine andere Möglichkeit wäre die Messung über das Verhältnis von falschen Klicks gegenüber dem richtigen „Mausklicks“ (Klickpfad). Die letzte Messkomponente, die „Zufriedenheit“, ist überwiegend subjektiver Natur (vgl. [HV03], S. 3). Ein wichtiger Einflussfaktor ist dabei die Erwartung des Benutzers gegenüber dem Produkt (siehe auch DIN EN ISO 9241-10 „Erwartungskonformität“).

## Weitere Usability-Definitionen

Der IEEE-Standard für Terminologie des Software-Engineering definiert dagegen Usability wie folgt:

*“Usability. The ease with which a user can learn to operate, prepare inputs for, and interpret outputs of a system or component.” ([M90], S.80; zitiert nach [AK+03], S.324)*

Hier wird die Usability auf das EVA-Prinzip (Eingabe-Verarbeitung-Ausgabe) reduziert. Die Fähigkeiten des Benutzers die korrekte Eingabe und Interpretation der Ausgabe (bei Bedarf erlernen zu können) sind nach dieser Definition entscheidend.

Der Usability-Experte Jakob Nielsen (<https://www.nngroup.com/articles/>) betrachtet Usability als ein mehrdimensionales und mehrfach attribuiertes Modell, bestehend aus den Eigenschaften (vgl. [N93], S.26)

- Erlernbarkeit,
- Effizienz,
- Erinnerungsfähigkeit,
- Fehlerdimension sowie Zufriedenheit.

Die qualitative Ausprägung der Erinnerungsfähigkeit ist für Nielsen die Grundvoraussetzung für die Erlernbarkeit. Dies beinhaltet einerseits die Effizienz, denn der Benutzer möchte möglichst schnell die Bedienung erlernen. Andererseits versteckt sich hier auch teilweise die Zufriedenheit, wenn beispielsweise nach längerer Abwesenheit die Bedienung immer noch prägnant im Langzeitgedächtnis des Benutzer abgespeichert ist und somit nicht ein erneuter Lernaufwand erforderlich ist. Hinter der Fehlerdimension befindet sich die Fehlertoleranz und Vermeidung, wie sie auch in den Grundsätzen der Dialoggestaltung nach DIN EN ISO 9241-10 unter dem Punkt „5. Fehlertoleranz“ wiederzufinden ist.

Der internationale Standard für Softwareproduktqualität in der Informationstechnologie kategorisiert Usability als Qualitätseigenschaft eines Systems, welches aus den folgenden Komponenten besteht (vgl. [ISO9126], S.7-10):

- Verständlichkeit:  
Die Software ist selbstbeschreibend und den Aufgaben angemessen gestaltet.
- Erlernbarkeit:  
Die Fähigkeit der Software, den Benutzer beim Erlernen zu unterstützen.
- Durchführbarkeit:  
Der Benutzer kann mit der Software arbeiten und den Dialogfluss kontrollieren.
- Attraktivität:  
Die Software ist für den Benutzer visuell attraktiv gestaltet und verwendet dabei Prinzipien aus dem Grafikdesign.
- Konventionen:  
Die Softwaregestaltung richtet sich nach bereits vorherrschenden Standards, Konventionen & Style-Guide's für die jeweilige Domäne.

Madan und Dubey haben in ihrer 2014 publizierten Literaturanalyse zum Thema “Usability Evaluationsmethoden“ ([MD12]) zwei weitere Usability-Modelle aus der Literatur recherchiert. Zum einen das Usability-Modell nach Eason, zum anderen das Modell nach Shackel (vgl. [MD12]).

## Zusammenfassung & Fazit

Für den Einstieg in die Usability-Thematik ist der internationale Standard DIN EN ISO 9241 maßgebend, denn sowohl in der Literatur als auch in der Industrie hat sich diese derzeit meist zitierte Norm bereits durchgesetzt (vgl. [JI+03], S. 53). Außerdem bildet eine Norm ein allgemeiner Konsens, welcher durch die Beteiligung zahlreicher Experten und Arbeitsgruppen (wie z. B. Konsortien) entstanden ist. Man kann also davon ausgehen, dass eine Norm ein intensives Prüfverfahren durchlaufen hat. Für Usability-Evaluationsstudien und -Tests bildet die Kenntnis weiterer Usability-Dimensionen bzw. Usability-Attribute eine gute Indikatoren-Grundlage für die Abbildung quantifizierbarer Messungen. Ein Problem, das alle Modelle besitzen, ist die hohe Abstraktion, die durch die Verallgemeinerung entsteht. Dadurch ist zwar ein Verstehen von Prinzipien möglich, aber die kontextuelle Analyse wird als durch die Domänenabhängigkeit immer wieder eine anfallende Arbeitsaufgabe eines Usability-Experten sein. Ein weiterer Nachteil ist, dass die Beschaffung eines Standards mit Kosten verbunden ist und der Lernende somit auf Sekundärliteratur ausweichen oder aber Geld investieren muss.

## Quellen

[SB11]: Florian Sarodnick, Henning Brau (2011): Methoden der Usability Evaluation. Wissenschaftliche Grundlagen und praktische Anwendung, 2, Hans Huber, Bern, 2011.

[N93]: Jakob Nielsen (1993): Usability Engineering. Morgan Kaufmann, San Diego, 1993.

[BB14]: Leila Akreimi, Nina Baur, et. al. (2014): Handbuch Methoden der empirischen Sozialforschung. Springer, 2014.

[My14]: David G. Myers (2014): Psychologie, 3, Springer, Berlin, 2014.

[HV03]: Robert Baggen, Andreas Beu, Kai-Christoph Hamborg, Marc Hassenzahl, Sven Heinsen, Jens Heuer, Ralph Hinderberger, James Kalbach, Adriane Kempken, Olde Lorenzen-Schmidt, Matthias Müller-Prove, Kerstin Röse, Christian Scheier, Petra Vogt (2003): Usability praktisch umsetzen. Carl Hanser, 2003.

[IEEE90]: IEEE Members (1990): IEEE Standard Glossary of Software Engineering Terminology. of and Engineers, The Institute Electrical Electronics (Hrsg.): USA, 1990.

[ISO9126]: ISO/IEC Members (2000): ISO/IEC FDIS 9126-1. Information technology – Software product quality. Part 1: Quality Model. ISO/IEC, (Hrsg.): 2000.

[MD12]: Ankita Madan, Sanjay Kumar Dubey (2012): Usability Evaluation Methods: A Literature Review. In: International Journal of Engineering Science and Technology (IJEST), S. 590-599, 2012.

[AK+03]: Alain Abran, Adel Kheli, Witold Suryn (2003): Usability Meanings and Interpretations in ISO Standard. In: Software Quality Journal, S. 323-336, 2003.

[JI+03]: Timo Jokela, Netta Iivari, Juha Matero, Minna Karukka (2003): The Standard of User-Centred Design and the Standard Definition of Usability: Analyzing ISO 13407 against ISO 9241-11. In: Proceedings of the Latin American conference on Human-Computer Interaction, S. 53-60, 2003.

# Mobile Usability

Die grundlegenden Usability-Prinzipien lassen sich auch im mobilen Kontext übertragen und anwenden. Allerdings existieren, im Vergleich zu stationären Desktop-Anwendungen, zusätzliche Anforderungen, die als Entwicklungs Herausforderungen zu betrachten sind (vgl. [NB13], S. 10-13 u. S. 39-41). Diese Anforderungen ergeben sich durch einem dynamischen Nutzungskontext und folgenden Limitierungen, sowie veränderten Eigenschaften mobiler Geräte (vgl. [HF+13], S. 2):

- **Mobiler Kontext:**  
Durch die Mobilität entsteht ein dynamischer Nutzungskontext und daraus resultiert eine sich wechselnde Umgebung- und Nutzungssituation.
- **Konnektivität:**  
Mobile Geräte verfügen über eine geringere Bandbreite, Datengeschwindigkeit und auch die Zuverlässigkeit der Verbindung ist deutlich geringer im Vergleich zu Desktop-Anwendungen.
- **Displaygröße:**  
Durch die kleineren Displays ist die Anzahl der darzustellenden Informationen wesentlich geringerer. Für das Design stellt diese Hardware-Limitierung eine große Herausforderung dar.
- **Displayauflösung:**  
Auf älteren mobilen Geräten ist die Displayauflösung unwesentlich geringerer.
- **Energie & Leistung:**  
Die zur Verfügung stehende Energie (Akkulaufzeit) und die davon abhängige Leistung des mobilen Gerätes ist durch die zwingende Ökonomie beschränkt.
- **Dateneingabe:**  
Die klassischen Eingabemethoden wie die Maus und die Tastatur gelten nicht für den mobilen Bereich. Egal ob Touch- oder Stifteingabe: Die Fehlerrate ist bei mobilen Geräten wesentlich höher und die Rate der Dateneingabe geringerer.

Außerdem besitzt die psychologische Last, auch kognitive Last genannt, im mobilen Kontext eine höhere Usability-Bedeutung (vgl. [HF+13]). Dies resultiert aus dem folgenden neuen Verhaltensmuster, welche bei der Verwendung von mobilen Anwendungen beim Benutzer zu beobachten ist: Während der Benutzer bei Desktop-Anwendungen sich auf seine Aufgabe vollständig fokussiert, werden mobile Anwendungen häufiger parallel zu anderen Tätigkeiten durchgeführt. Dadurch steht der mobilen Anwendung eine verminderte kognitive Leistungsfähigkeit des Benutzers zur Verfügung. Ein Beispiel ist die mobile Autonavigation. Während die Autofahrt die primäre Aufgabe des Benutzers ist, soll die Navigationsanwendung den Benutzer unterstützen. Dies führt dazu, dass die Navigation über eine Sprachausgabe verfügen muss (andere Eingabemethode), damit der Benutzer weiterhin konzentriert Auto fahren kann (vgl. [HF+13], S. 4-5). Die Ein- und Ausgaben müssen also zielgerichtet erfolgen und gleichzeitig darf die Aufmerksamkeit des Fahrer nicht überstrapaziert werden, um für den Verkehr keine Gefahr darzustellen.



Zu empfehlen ist der Forschungsbeitrag von 2013 mit dem Titel “Usability Heuristics for Touchscreen-based Mobile Devices: Update“ von Inostroza und Rusu et al. ([IR+13]), welche die Usability-Heuristiken nach Nielsen ([N93]) mit ihren eigenen entwickelten Heuristiken für mobile Endgeräte mit Touch-Eingabe validiert haben. Auch hier wurden die oben genannten spezifischen Eigenschaften und Limitierungen mobiler Geräte berücksichtigt, insbesondere die Vermeidung von kognitiver Last spiegelt sich im Punkt „Minimize the user’s memory load“ wider (vgl. [IR+13]).

## Quellen

[NB13]: Jakob Nielsen, Raluca Budiu (2013): Mobile Usability. Für iPhone, iPad, Android und Kindle. mitp, Heidelberg, 2013.

[N93]: Jakob Nielsen (1993): Usability Engineering. Morgan Kaufmann, San Diego, 1993.

[HF+13]: Rachel Harrison, Derek Flood, David Duce (2013): Usability of mobile applications: literature review and rationale for a new usability model. In: Journal of Interaction Science, S. 1-16, 2013.

[IR+13]: Rodolfo Inostroza, Silvana Roncagliolo, Cristian Rusu, Virginica Rusu (2013): Usability Heuristics for Touchscreen-based Mobile Devices: Update. In: First Chilean Conference on Human – Computer Interaction (ChileCHI ’13), S. 24-29, ACM, 2013.

# User Experience Design

Mit „User Experience Design“ ist die Gestaltung positiver Benutzererlebnisse gemeint und der Begriff „User Experience“ bezeichnet sämtliche für den Benutzer erfahrbare Berührungspunkte eines Produktes (vgl. ISO 9241-210). Dies kann beispielsweise bei der Website beginnen und auch die langfristige, persönliche Produktidentifizierung mit einschließen (ein gutes Beispiel hierfür wäre das iPhone von Apple). Eine wesentliche Erfahrung wird dabei während der Produktnutzung gemacht und hier sorgt eine möglichst hohe Usability-Ausprägung für ein maximales, positives Benutzererlebnis. Allgemein kann die User Experience in drei Erfahrungsbereichen unterteilt werden (vgl. [SB11], S. 22):

- **Vor der Nutzung:**  
Der Nutzer setzt Erwartungen voraus.
- **Während der Nutzung:**  
Die Usability wird als wesentlicher Qualitätsfaktor wahrgenommen.
- **Nach der Nutzung:**  
Der langfristige Aufbau durch emotionale Bindung, sowie Akzeptanz.

Für das User Experience Design ist die Unterscheidung zwischen positiven und negativen Erfahrungen entscheidend. Ein gutes User Experience Design vermittelt primär positive Gefühle und versucht negative Emotionen zu vermeiden (vgl. [M12]). Ein Teil der User Experience lässt sich steuern (z. B. durch ein ganzheitliches Design im Marketing), ein anderer Teil, wie beispielsweise die Meinung anderer Personen im unmittelbaren Umfeld des Benutzers, ist nicht direkt beeinflussbar (vgl. [SB11], S. 22). Grundvoraussetzung für die Erzeugung positiver Benutzererlebnisse ist die Erforschung der Benutzererwartungen durch die Analyse des Benutzers, seiner Ziele und seines Nutzungskontextes (vgl. [M12], S. 2-13).

In diesem Zusammenhang fällt auch der Begriff „Joy-of-Use“, der den funktionalen Usability-Charakter um eine emotionale Komponente (Spaß und Begeisterung) erweitert (vgl. [M12], S. 17 u. [D11], S. 17). Joy-of-Use wird dadurch generiert, indem die Erwartungen, Bedürfnisse und individuelle Kompetenz des Benutzers derart berücksichtigt werden, sodass eine Zufriedenheit bei der Benutzung entsteht, dass wiederum ein positives Erlebnis darstellt (vgl. [S10], S. 291). Nach Heinecke ist die intuitive Bedienung die Vorbedingung für ein möglichst positives Benutzungserlebnis und ist somit ein wichtiger Einflussfaktor für die User Experience (vgl. [H12], S. 34). Für das Konzept der Intuitivität existieren bisher nur Ansätze aus der Forschung, aber als wichtigstes Gestaltungsprinzip kann die Aufgabenorientierung des Benutzers genannt werden (vgl. [H12], S. 34).

## Quellen

[SB11]: Florian Sarodnick, Henning Brau (2011): Methoden der Usability Evaluation. Wissenschaftliche Grundlagen und praktische Anwendung. 2, Hans Huber, Bern, 2011.

[M12]: Christian Moser (2012): User Experience Design. Mit erlebniszentrierter Softwareentwicklung zu Produkten, die begeistern. Springer Vieweg, Berlin, 2012.

[D11]: Rainer Dorau (2011): Emotionales Interaktionsdesign. Gesten und Mimik interaktiver Systeme. Springer, Berlin, 2011.

[S10]: Torsten Stapelkamp (2010): Interaction- und Interfacedesign. Web-, Game-, Produkt und Servicedesign Usability und Interface als Corporate Identity. Springer, Berlin, 2010.

[H12]: Andreas M. Heinecke (2012): Mensch-Computer-Interaktion. Basiswissen für Entwickler und Gestalter. 2, Springer, Wiesbaden, 2012.

# Usability Engineering

Usability Engineering ist ein prozessorientiertes Entwicklungsparadigma, um eine möglichst hohe Ausprägung von Usability beim Endprodukt sicherzustellen. Dabei wird der Nutzungskontext und das Ziel des Benutzers frühzeitig erforscht, um auf dieser Basis relevante Anforderungen, Funktionalitäten und Bedienkonzepte abzuleiten (vgl. [BC+12], S. 6 u. [D10] S. 12-13). Durch den geeigneten und frühzeitigen Methodeneinsatz des Usability Engineering wird einerseits der Benutzer in die Entwicklung miteinbezogen (partizipativer Anteil). Andererseits werden durch Usability-Evaluationsstudien die aktuelle Usability-Ausprägung gemessen, um die Systementwicklung hinsichtlich Usability besser lenken und optimieren zu können (vgl. [SB11], S. 87-88). Zusammengefasst ist Usability Engineering somit nicht nur ein Sammelbegriff für jene Methoden und Techniken (vgl. [RF13], S. 7), sondern ist darüber hinaus ein qualitatives Entwicklungsvorhaben. Ein allgemeines Vorgehen zu definieren ist jedoch wenig sinnvoll, denn die Auswahl der Methoden ist individuell vom Entwicklungsprojekt abhängig (vgl. [RF13], S. 13). Insbesondere ist die Methodenauswahl von den Ressourcen abhängig, denn der Zeitbedarf der Methoden ist unterschiedlich ausgeprägt (z. B. heuristische Evaluation vs. Usability-Tests [vgl. S. 15-18]). Ein weiterer wichtiger Punkt bezüglich der Benutzerschnittstelle ist die Vermeidung von zu hoher Komplexität im Bedienkonzept (vgl. [RF13], S. 8). Nach Nielsen ist es beim Usability Engineering entscheidend, die jeweiligen Methoden frühzeitig vor der Umsetzung der Benutzerschnittstelle zu praktizieren (vgl. [N93], S. 71).

## Quellen

[SB11]: Florian Sarodnick, Henning Brau (2011): Methoden der Usability Evaluation. Wissenschaftliche Grundlagen und praktische Anwendung, 2, Hans Huber, Bern, 2011.

[RF13]: Michael Richter, Markus Flückiger (2013): Usability Engineering kompakt. Benutzbare Produkte gezielt entwickeln, 3, Berlin, 2013.

[N93]: Jakob Nielsen (1993): Usability Engineering. Morgan Kaufmann, San Diego, 1993.

[BC+12]: Kay Behrenbruch, Christian Bogner, Holger Fischer, Thomas Geis, Claudia Geitner, Rüdiger Heimgärtner, Britta Hofmann, Peter Hunkirchen, Oliver Kluge, Britta Litzenberg, Knut Polkehn, Yuliya Pysarenko, Dirk Zimmermann (2012): German UPA Qualitätsstandard für Usability Engineering. e.V., German UPA (Hrsg.): 2012.

[D10]: (2010): Leitpfaden Usability. DAkkS, Deutsche\_Akkreditierungsstelle (Hrsg.): 2010.

# Prototyping

Das Prototyping ist eine Methode die im Rahmen des Usability Engineering-Prozess eingesetzt wird, um Benutzerschnittstellen überprüfen und verbessern zu können, noch bevor die finale Implementierung durchgeführt wurde (vgl. [RF13], S. 52). Durch das Prototyping kann die Entwicklung beschleunigt, Kosten gespart und das Verständnis für den Aufbau der Benutzerschnittstelle erhöht werden. Die Idee ist es, Konzepte möglichst frühzeitig und iterativ mit potenziellen Anwendern zu evaluieren (vgl. [Nie93], S. 93-94). Aber auch zu Diskussionszwecken innerhalb der Entwicklungsgruppe eignet sich Prototyping (vgl. [Gal07], S. 771). Methodisch steht zu Beginn die Fragestellung: Was ist zu evaluieren und welches Ziel soll dabei angestrebt werden ([RF13], S. 53)? Mit Prototyping können beispielsweise Anforderungen validiert, Bedienkonzepte evaluiert und Optimierungen von bestehenden Benutzerschnittstellen durchgeführt werden (vgl. [RF13], S. 59). Beim Prototyping wird iterativ vorgegangen, d.h. aus jeder Überprüfung folgt eine Auswertung der Ergebnisse sowie die daraus resultierenden Änderungen am Bedienkonzept ([RF13], S. 60).

## Quellen

[RF13]: Michael Richter, Markus Flückiger (2013): Usability Engineering kompakt. Benutzbare Produkte gezielt entwickeln. 3, Berlin, 2013.

[N93]: Jakob Nielsen (1993): Usability Engineering. Morgan Kaufmann, San Diego, 1993.

[Gal07]: Wilbert O. Galitz. The Essential Guide to User Interface Design. An Introduction to GUI Design Principles and Techniques. 3. Au. San Francisco: Wiley, 2007.

# User Interface Prototyp

Unter einem User Interface Prototyp ist die zu entwickelnde Benutzerschnittstelle zu verstehen, welcher zwischen Mensch und Maschine vermittelt und im Kontext des Usability Engineering zur Überprüfung eingesetzt wird ([RF13], S. 52). Allgemein wird zwischen vertikalen und horizontalen Prototypen unterschieden: Ein vertikaler Prototyp setzt nur einige ausgewählte Features (in vollständiger Ausprägung) um, ein horizontaler Prototyp dagegen deckt sämtliche Features (allerdings nicht tiefgehend) ab (vgl. [Nie93], S. 94). Darüber hinaus existieren noch die Dimensionen der Darstellungstiefe (Ausprägung der tatsächlichen visuellen Erscheinung), das Ausmaß an Interaktivität (statisch oder dynamisch), die verwendeten Daten (real oder fiktiv) und die allgemeine, technische Reife (wiederverwendbar oder zweckgebunden; vgl. [RF13], S. 53). Je nach Ziel und Entwicklungsstand können Prototypen als gezeichnete Papiervorlage, als Video, als Animation oder zum Teil bereits implementiert vorliegen. Papier-Prototypen besitzen den Vorteil, das solche in sehr kurzer Zeit erstellt werden können. Dadurch können verschiedene Varianten effizient ausprobiert und in gemeinsamer Diskussion untersucht werden. Außerdem können Papier-Prototypen auch von nicht technischen Mitarbeitern erstellt werden und regen insgesamt die Kreativität im Entwicklungsprozess an (vgl. [Sny03], S. 12). Allerdings können nicht alle Interaktionen mittels Stift und Papier dargestellt werden. Auch Usability-Test's (vgl. S. 15-18) sind mit solchen Prototypen nicht oder nur sehr begrenzt einsatzfähig (vgl. [Gal07], S. 772). Je mehr Systemanteile umgesetzt sind, desto realistischer kann getestet werden. Im Vergleich dazu besitzen Tests mit Papier-Prototypen einen Workshop-Charakter (vgl. [SB11], S. 166). In diesem Zusammenhang wird von sogenannten „high und low fidelity“ Prototypen gesprochen (vgl. [Sny03], S. 20).

## Quellen

[RF13]: Michael Richter, Markus Flückiger (2013): Usability Engineering kompakt. Benutzbare Produkte gezielt entwickeln. 3, Berlin, 2013.

[N93]: Jakob Nielsen (1993): Usability Engineering. Morgan Kaufmann, San Diego, 1993.

[SB11]: Florian Sarodnick, Henning Brau (2011): Methoden der Usability Evaluation. Wissenschaftliche Grundlagen und praktische Anwendung. 2, Hans Huber, Bern, 2011.

[Gal07]: Wilbert O. Galitz. The Essential Guide to User Interface Design. An Introduction to GUI Design Principles and Techniques. 3. Au. San Francisco: Wiley, 2007.

[Sny03]: Carolyn Snyder. Paper Prototyping. The Fast and Easy Way to Design and Rene User Interfaces. San Francisco: Morgan Kaufmann, 2003.

# Usability-Test

Der Usability-Test ist eine empirische Methode zur Evaluierung von Gebrauchstauglichkeit. Primär geht es um das Testen von Systemen mit realen Nutzern und realitätsnahen Aufgaben, um so beispielsweise Schwachstellen aufdecken zu können oder auch um Messungen (zum Beispiel die Dauer zum Lösen von Aufgaben) vorzunehmen. Die gewonnenen Erkenntnisse sollen zur Verbesserung des Systems genutzt werden. Usability-Test's können entweder induktiv oder deduktiv durchgeführt werden. Induktive Tests werden im formativen Evaluationsansatz angewendet, um ein noch zu entwickelndes Produkt zu verbessern. Im Gegensatz dazu geht es bei deduktiven Test's um den Vergleich und die Leistungsbeurteilung mehrerer Systeme wie zum Beispiel unterschiedlicher Varianten (vgl. [SB11], S. 162-163).

Für die Durchführung eines Usability-Test's muss vorher genau überlegt werden, wer und wie getestet werden soll. Diese Planungsdetails werden in einem Usability-Testplan festgehalten (vgl. [Bar11], S. 142). Der Testplan dient dabei innerhalb des Projektmanagements als Meilenstein, innerhalb des Entwicklungsteams als Kommunikationsbasis. Letzteres ist wichtig, um die Qualität eines geplanten Usability-Test's sicherzustellen (vgl. [RC08], S. 65-67). Die folgenden Punkte fassen stichpunktartig die wichtigsten Informationen eines Testplans kurz zusammen (vgl. [Bar11], S. 145-146):

- Zieldefinition: Was ist das Ziel und welche Intention steckt dahinter?
- Untersuchungsgegenstand: Was soll getestet werden und welche Kriterien sind relevant?
- Methodik und Datenerhebung: Welche Methoden werden verwendet und welche Daten werden erhoben?
- Testpersonen: Wer soll getestet werden (Kriterien) und wie lautet die Anzahl?
- Testaufgaben: Welche Aufgaben dienen dem Testen?
- Testumgebung: Wo soll getestet werden?
- Werkzeuge: Womit soll getestet werden?
- Auslieferung: Welches Ergebnis wird in welcher Präsentationsform geliefert?

Wie schon erwähnt, können entweder Schwächen aufgedeckt oder die aktuelle Leistung kontrolliert werden (vgl. [SB11], S.163), doch das Testen vollständiger Systemaspekte ist aufwendig und nicht immer zweckmäßig. Deswegen sollte der Test sich auf ein oder mehrere Aspekte beschränken, wie zum Beispiel die Analyse von Handlungs- und Bedienlogiken oder anderen Gestaltungsfeldern (vgl. [SB11], S.213).

Untersucht werden können entweder ein oder mehrere Systeme. Letzteres ist als AB-Test (Vergleichstest) bekannt, bei dem beispielsweise eine alte Systemversion nach gezielter Änderung mit der neuen Version verglichen wird. Die oder das System kann vom halbfertigen Prototypen bis zu vollständigen Systemen reichen, wobei letzteres nicht dienlich im Sinne des Usability Engineering ist (vgl. [SB11], S.163-166). Für die Datenerhebung existieren verschiedene Methoden (siehe [SB11]).

Eine davon ist die Videoanalyse, von der allerdings aufgrund des hohen Zeitbedarf für die Auswertung abgeraten wird. Auch Eingabeprotokolle können dienlich sein (abgespeicherte Abläufe des Benutzers bzw. Systems in Logdateien), dies kann jedoch ebenso aufwändig in der Auswertung (insbesondere der Interpretation) sein. Eine weitere Methode ist das laute Denken (engl. thinking aloud), bei der die Gedanken und Gefühle während der Testdurchführung durch die Testperson verbalisiert werden. Ihren Ursprung hat die Methode des lauten Denkens in der Psychologie und aufgrund dieser zusätzlichen, kognitiven Belastung können keine valide Performance-Messungen durchgeführt werden. Ein weiterer Nachteil ist die unnatürliche Art und Weise, wie sich die Testpersonen dadurch verhalten. Das laute Denken erfordert psychologisches Feingefühl, denn die Testperson darf nicht so beeinflusst werden, sodass ihre Gedanken und Handlungen einer bestimmten Richtung gelenkt werden. Ebenso ist die Interpretation dieser Gedanken immer kritisch in der Auswertung zu betrachten, da der Benutzer schnell eigene Theorien über die Probleme entwickelt. Deswegen sollte lautes Denken mit anderen Daten (wie zum Beispiel Verhaltensdaten) analysiert werden. Eine andere Form des lauten Denkens ist die Co-discovery-Methode, bei dem zwei Personen eine Aufgabe bearbeiten. Der Dialog ist natürlicher als das laute Denken im Alleingang. Eine andere Ausprägung ist die coaching-Methode, bei der nicht zwei Testpersonen, sondern eine Testperson mit dem Testleiter (der normalerweise sich passiv verhalten sollte) die Testaufgaben gemeinsam lösen sollen. Über Aufmerksamkeitsanalysen wie zum Beispiel Eye-Tracking (Messungen von Blicken und Blickbewegungen) können weitere Hinweise über die kognitiven Prozesse erschlossen werden. Weitere Formen der Messung, wie zum Beispiel Zeit- und Fehlerdaten, können ebenfalls Hinweise auf mögliche Probleme liefern. Beispielsweise kann die Zeit für die Bearbeitung oder auch das Verhältnis zwischen erfolgreichen Handlungen und Fehlern gemessen sowie berechnet werden (vgl. [SB11], S.169-173).

Beim sogenannten Attention-Tracking können (im Vergleich zu aufwändigen Aufmerksamkeitsanalysen) beispielsweise die Bewegungen und Klicks der Computer-Maus im Hintergrund erfasst werden. Allerdings entscheidend sind die Klicks (als die bewusste Handlung), da Mausbewegungen häufig unwillkürlicher Natur sind (vgl. [SB11], S.178-179). Für die quantitative Datenerhebungen eignen sich Fragebögen, die genau wie der Usability-Test selbst, hinsichtlich der Objektivität (Ergebnis unabhängig vom Versuchsleiter), Reliabilität (Wiederholbarkeit bei gleichen Bedingungen) sowie Validität (Übereinstimmung der Messdaten mit dem Zielen) zu bewerten sind (vgl. [SB11], S.181-183). Für die Usability-Evaluation existieren darüber hinaus auch standardisierte Fragebögen für spezifische Zwecke, auf die hier nicht weiter eingegangen werden, da sie aus dem Rahmen dieses Beitrags fallen (vgl. [SB11], S.187).



Innerhalb der Planung bildet die Frage nach der Auswahl und Anzahl der Testpersonen eine wichtige Grundlage des Usability-Test's. Zunächst einmal müssen die Personen repräsentativ sein, d.h. der Zielgruppe bestmöglich entsprechen. Liegen jedoch keine Informationen über die Zielgruppe vor (wie zum Beispiel im konsumorientierten Bereich wie das Restaurant), dann sollten zumindest Alter, Geschlecht, Ausbildung/Beruf sowie elektronische Datenverarbeitungserfahrungen empirisch ermittelt werden. Fünf bis sechs Testpersonen sollen nach dem aktuellen Stand der Wissenschaft dabei reichen, um ca. 80 % der Usability-Probleme aufzudecken. Die tatsächliche Anzahl jedoch hängt von den Rahmenbedingungen des Projektes ab wie zum Beispiel das Budget und die Anzahl der Test's. Aber auch der Anwendungskontext ist ein wesentlicher Einflussfaktor. Während ein sicherheitskritischer Bereich intensiver getestet werden sollte, so ist der Konsumentenbereich eher zu vernachlässigen (vgl. [SB11], S.167).

Auch die Gestaltung und die Auswahl der Testaufgaben kann einen positiven oder negativen Einfluss auf die Resultate einnehmen. Hier kann zwischen einem strukturierten und unstrukturierten Aufgabendesign differenziert. Eine strukturierte Aufgabe (enthält viele Anweisungen für den Benutzer) deckt qualitativ weniger schwere Usability-Probleme auf, als unstrukturierte Aufgaben. Quantitativ jedoch werden damit mehr Probleme aufgedeckt (vgl. [AM08]). Wichtig ist bei beiden Aufgabentypen, dass der Aufgabenkontext des zukünftigen Benutzer möglichst real ist (vgl. [SB11], S.165).

Für die Durchführung können entweder speziell eingerichtete Usability-Labore dienen oder aber auch mobile Testlabore im Feld verwendet werden. Während ein Usability-Labor nach Einrichtung eine optimale Testinfrastruktur bietet, ermöglichen mobile Labore das Testen im realen Nutzungskontexten (vgl. [SB11], S.168-169). Als Spezialfall gelten Remote-Usability-Tests, bei denen der Versuchsleiter und die Testpersonen lokal unabhängig voneinander dem Test beiwohnen. Bei entsprechender Automatisierung ist auch eine zeitunabhängige Durchführung möglich, doch ist hier der Aufwand einer nachträglichen Datenauswertung wesentlich höher (vgl. [SB11], S.179-181).

Auf Werkzeuge und Auslieferungen wird nicht weiter eingegangen, da ersteres von den gewählten Datenerhebungsmethoden abhängig ist und letzteres vom individuellen Projektziel bestimmt wird. Nach Semiu, Hector und Hammed ([SHH12]) liegen die wesentlichen Herausforderungen bei Usability-Test's in der Planung und Analyse. Dazu gehört die Definition des Testgegenstandes und der bewusste Einsatz von Methoden sowie deren (positiven oder auch negativen) Einfluss zueinander. Was ist zu messen, wie wird gemessen und ist die Messung valide und zuverlässig. Auch die subjektive Komponente des Versuchsleiter ist zu berücksichtigen, denn die Problembefunde unterscheiden sich konsequenterweise. Schließlich sind die Anzahl der Testpersonen, die ausgewählte Aufgaben sowie der Ort der Testdurchführen weitere wesentliche Faktoren im Testdesign (vgl. [SHH12]).

## Quellen

[SB11]: Florian Sarodnick, Henning Brau (2011): Methoden der Usability Evaluation. Wissenschaftliche Grundlagen und praktische Anwendung. 2, Hans Huber, Bern, 2011.

[Bar11] Carol M. Barnum. Usability Testing Essentials. Ready, Set...Test. USA: Morgan Kaufmann, 2011.

[RC08] Jerrey Rubin und Dana Chisnell. Handbook of Usability Testing. How to Plan, Design and Conduct Eective Tests. 2. Au. Indianapolis, Indiana: Wiley, 2008.

[SHH12] Akanmu Semiu, Okere P. Hector und Adebambo Hammed. Current Issues of Usability characteristics and Usability testing. In: First International Conference on Behavioural and Social Science Research (ICBSSR 2012). Universiti Tunku Abdul Rahman, 2012, S. 15.

[AM08] Majed Alshamari und Pam Mayhew. Task Design: Its Impact on Usability Testing. In: The Third International Conference on Internet and Web Applications and Services. IEEE computer society, 2008, S. 583589.