



## Hassliebe

# Nutzerzentrierte Entwicklung von Business-Software in der Praxis

Anja Endmann

Globalisierung, hochdynamische Märkte, zunehmende Verbreitung von Internet und mobilen Endgeräten – all diese Faktoren bedingen den hohen Konkurrenzdruck, dem sich die Softwareunternehmen ausgeliefert sehen. Viele Anbieter versuchen, wettbewerbsfähig zu bleiben, indem sie für ihr Produkt eine höhere Leistung und einen stetig wachsenden Umfang an Features zur Verfügung stellen. Dabei wird viel Geld in die schnelle Umsetzung neuer Features und Funktionen investiert, die die Anwendungen immer komplexer und unübersichtlicher werden lassen und dem Endanwender die Arbeit mit dem Produkt erschweren. Softwareanwender sind mittlerweile häufig mit der Nutzung überfordert, dadurch demotiviert und frustriert. Diese Auswirkungen spiegeln sich beispielsweise in mangelnder Motivation und längeren Bearbeitungszeiten von relativ einfachen Arbeitsaufgaben wider. Auf Dauer kann dieses Vorgehen daher nicht zum Erfolg führen. Wie also kann ein Produkt nachhaltig wettbewerbsfähig und erfolgreich sein? Welche Merkmale und Funktionen werden dafür benötigt?

## Der Schlüssel zum Erfolg

Die Integration des User Centered Design (UCD)-Prozesses in die Produktentwicklung ist der Schlüssel zum Erfolg beim Nutzer. Der UCD-Prozess setzt den Nutzer mit seiner User Experience (UX) in den Fokus der Entwicklung – bei gleichzeitiger Berücksichtigung der Ziele, Erwartungen und Vorstellungen der Projektbeteiligten (Stakeholder). Diese Fokussierung gibt Aufschluss über die potenziellen Nutzer einer Software, deren Erwartungen, (Arbeits-)Aufgaben, Bedürfnisse, Denkstrukturen, Handlungsweisen, den jeweiligen Nutzungskontext und die Erwartungen an die Marke der zu entwickelnden Software. Daraus lässt sich ableiten, welche Funktionen und Merkmale ein Produkt mindestens beinhalten muss. Ebenso, wie sie umgesetzt sein müssen, damit das Produkt effizient eingesetzt werden kann und als attraktiv sowie gebrauchstauglich empfunden wird. Die aufgeführten Aspekte lassen sich in die vier Bereiche Markenerfahrung (Brand Experience), Erwünschtheit (Desirability), Gebrauchstauglichkeit (Usability) und Nutzen (Utility) zusammenfassen (s. Abb. 1).

Erst wenn ein Produkt neben den notwendigen Features und Funktionen eine positive User Experience schafft, hat es einen Mehrwert für den jeweiligen Einsatzkontext. Es zeichnet sich gegenüber Konkurrenzprodukten durch eine geringere Anlernzeit der Nutzer und eine höhere Arbeitseffizienz aus. Die Nutzer fühlen sich in ihrer Arbeit unterstützt und sind motivierter. Über einen längeren Nutzungszeitraum kann dadurch ein Vertrauen in die Marke entstehen. Dies wiederum steigert die Markentreue beim Nutzer und die Wettbewerbsfähigkeit des Produktes.

Die Integration des UCD-Prozesses in die Produktentwicklung ist der Weg zur Entwicklung eines nutzerorientierten, wettbewerbsfähigen Produktes. Der in diesem Artikel be-



Abb. 1: Definition von UX

schriebene nutzerorientierte Gestaltungsprozess beruht auf dem standardisierten UCD-Prozess der DIN EN ISO 9241-210 und wurde in eine in der Praxis geprüfte Version übertragen. Er beschreibt ein allgemeingültiges Vorgehen.

In die in der Praxis geprüfte Version des UCD-Prozesses wurde die Implement-Phase aufgenommen. Die Unterstützung der Entwicklung und der Review des Entwicklungsstandes haben für die praktische Projektdurchführung eine entscheidende Bedeutung; nur so kann gewährleistet werden, dass die Softwarelösung den im Vorfeld definierten Anforderungen entspricht. Auf Grund von limitierten Budgetierungen durchläuft eine Iteration in der Praxis darüber hinaus nur ein bis maximal zwei Schleifen.

Für jedes Produkt wird der Prozess individuell und iterativ durchlaufen, da sowohl Umfang als auch die eingesetzten Methoden in den einzelnen Phasen auf das jeweilige Projekt anzupassen sind. Zu betonen ist an dieser Stelle, dass alle in den Prozess involvierten Stakeholder zusammen über die (Zwischen-)Ergebnisse der einzelnen Phasen kollaborieren sollten. Die regelmäßige Abstimmung gewährleistet, dass alle den gleichen Kenntnisstand haben, auch wenn sie nicht in alle Phasen gleichermaßen integriert sind.

Im Folgenden werden die fünf Phasen (s. Abb. 2) detailliert beschrieben und mit konkreten Handlungsalternativen unter-

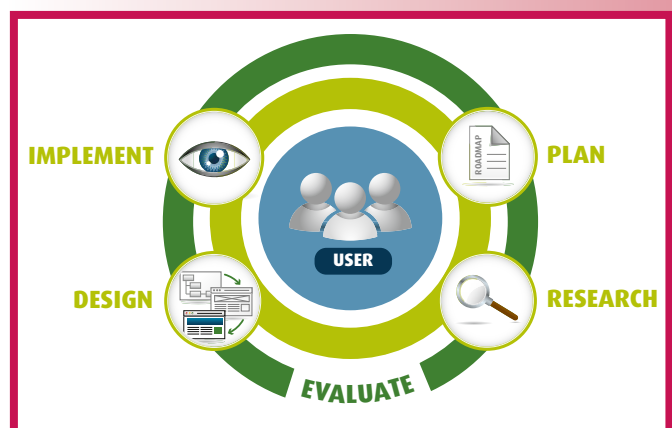


Abb. 2: Phasen des UCD-Prozesses



## Plan-Phase – „Definiere deine Roadmap!“

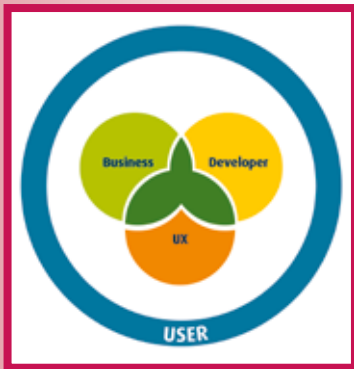


Abb. 3: Schnittmengen zwischen den „Stakeholdern“

Den Ausgangspunkt des UCD-Prozesses bildet eine Design Challenge. Diese Herausforderung ist beispielsweise eine bestimmte Problemstellung. Oder ein bereits bestehendes Produkt, das mit Hilfe des UCD-Prozesses nutzerorientiert überarbeitet werden soll. Um Projektziel, Inhalt, Zeitplan, die Methode der Projektumsetzung/-weiterführung sowie die Zuständigkeiten und Verantwortlichkeiten während des gesamten Projektverlaufes definieren zu können, findet häufig ein ein- bis dreitägiger Kickoff-Workshop zwischen den Projektbeteiligten statt (s. Abb. 3).

Durch das Zusammentreffen aller Parteien fühlen sich alle gleichermaßen für das Gelingen des Projektes verantwortlich. Des Weiteren haben sie den gleichen Kenntnisstand und durch die Definition der Zuständigkeiten können lange Bearbeitungswege, Missverständnisse und teure Fehlentwicklungen vermieden werden.

Als Arbeitsergebnis dieser Phase entsteht eine Roadmap, die den Zeitplan und die (finanzielle, temporäre und personale) Ressourcenplanung des Projektverlaufes aufzeigt. Ergebnisse während dieses Workshops bereits erste Ideen, können diese in Scribbles festgehalten und für die spätere Design-Phase aufbewahrt werden.

## Research-Phase – „Kenne deine Nutzer!“

„Valide Nutzerdaten sind die Grundlage für jedes erfolgreiche Softwareprodukt“, erläutert Katharina Göring, Director User Experience der Software AG-Tochter itCampus. „Sie eliminieren Unsicherheiten bei der Frage, was entwickelt werden muss, und das resultiert in effizienter Umsetzung. Das heißt, Problemlösung erreicht man nur mit einem sehr guten Verständnis des Problemraums.“

Im Zentrum der Research-Phase steht daher die umfassende Erforschung der Nutzer mit deren individuellen Anforderungen, Zielen, Aufgaben, Pain Points (Schwierigkeiten), Bedürfnissen, Meinungen und Wünschen. Um fundierte Kenntnisse zu diesen Aspekten zu erhalten, werden potenzielle Nutzer befragt und deren Verhalten wird dokumentiert. Gängige Methoden sind beispielsweise Fragebögen, Interviews, Benutzertagebücher, Beobachtungen oder Diskussionen mit Fokusgruppen. Auch Stakeholder mit direktem Kontakt zu potenziellen Anwendern liefern wichtigen Input, etwa in Stakeholder-Interviews und -Fokusgruppen. Unabhängig von der Verfügbarkeit potenzieller Nutzer und/oder Stakeholdern können darüber hinaus weitere Methoden wie die Analyse eines bestehenden Systems, vorhandene Dokumentationen, Nutzerstatistiken, Web-Analytics, Konkurrenz- sowie Marktanalysen eingesetzt werden.

Die Erkenntnisse analysiert man nun und überträgt sie in Personas, Szenarien, Storyboards, Use Cases und/oder User Stories. Das Beispiel einer Applikation für Inventuren, die *Invento-*

*ry-App*, dient im Folgenden zur näheren Erläuterung dieser für die weitere Projektentwicklung entscheidenden Instrumente.

Eine *Persona* ist eine fiktive Person. Sie steht stellvertretend für die Bedürfnisse einer meist unbekannteren Nutzergruppe. Durch ihren individuellen Charakter, ihre Motivation, Bedürfnisse, Ziele, Interessen, Herangehens- und Verhaltensweisen wird sie zu einer greifbaren, realen Person, in die sich der Designer oder Entwickler hinein versetzen kann. Für ein Produkt können ein oder mehrere Personas entwickelt werden, die sich klar voneinander abgrenzen. So könnte die Persona einer stu-



**Hanna Accuracy**  
*„Ich möchte mich nicht lange einarbeiten müssen. Dafür fehlt mir die Zeit. Ich möchte so schnell wie möglich zum Ziel gelangen.“*

---

<b>Alter</b> 23 Jahre	<b>Hintergrund</b> Hanna studiert seit zwei Jahren BWL. Sie arbeitet viel am PC und nutzt das Internet sowohl zu Hause als auch mobil für Recherchen. Sie hat fundierte Kenntnisse in MS Office, das sie für das Anfertigen von Belegarbeiten und Präsentationen verwendet. Darüber hinaus hat sie ebenfalls solide Kenntnisse für ein Marktanalysetool. In ihrer Arbeit legt sie sehr großen Wert auf Exaktheit und Genauigkeit. In ihrer Freizeit ist sie gern mobil unterwegs, um mit ihren Freunden in Kontakt zu bleiben und stets auf dem neusten Stand zu sein.
<b>Familienstand</b> Ledig, wohnt in einer WG	
<b>Beruf</b> Studentin	
<b>Hobbys</b> Lesen, Freunde treffen, Ausgehen	<b>Ziele</b> Hanna versucht durch Gelegenheitsjobs im Einzelhandel ihr Bafög aufzubessern. Sie möchte in kurzer Zeit ohne enorme Anstrengungen einen kleinen Zuverdienst erringen.

Abb. 4: Beschreibung einer Persona für die „Inventory-App“

dentischen Hilfskraft für die „Inventory-App“ wie in Abbildung 4 beschrieben sein.

Das heißt, die definierte Persona interagiert mit dem Produkt in einem realen Kontext, der in Form einer Kurzgeschichte – einem Szenario – visualisiert wird. Dieses realistische (Zukunfts-)Bild erlaubt einen Perspektivwechsel von der Produkt-zur Nutzersicht. „Hanna Accuracy“ unterstützt in unserem Fall einen Textilhersteller bei der Inventur. Dafür lässt sich beispielhaft folgendes Szenario entwickeln:

▼ Hanna unterstützt einen Textilhersteller bei der Inventur und hat dafür ein iPad erhalten. Auf einer Liste werden ihr die Räume und Regale angezeigt, für die sie heute die Inventur durchführen soll. Sie kann aus der Liste jeweils den Raum, das zu prüfende Regal und den darin befindlichen Artikel manuell oder durch das Scannen des jeweiligen Strichcodes auswählen. Nachdem sie den Lagerbestand eines Produktes gezählt hat, kann sie das Zählergebnis für das Produkt direkt eintragen. Bereits vorgenommene Einträge kann sie jederzeit korrigieren. Um sicherzustellen, dass sich das richtige Produkt im Regal befindet, kann Hanna die Detailansicht des Produktes vergrößern. Am Ende der Inventur gibt Hanna das iPad zurück. Ihre Ergebnisse werden automatisch mit dem System synchronisiert, sobald das Gerät online ist.

Um die Komplexität der Szenarien zu vereinfachen, können sie in Storyboards, Use Cases oder User Stories transformiert werden. Diese beschreiben die Interaktion des Nutzers mit dem System in unterschiedlichem Detaillierungsgrad. Nach Fertigstellung werden die Storyboards, Use Cases oder User Stories zu Storylines in Gruppen sortiert. Die so entstandenen Kapitel priorisiert man nun zeitlich entsprechend der Ergebnisse der Anforderungsanalyse.

Nachfolgend beschränkt sich der Artikel auf *User Stories*. Diese eignen sich vor allem für eine agile Softwareentwicklung, zum Beispiel mit Scrum. Je nach Komplexität kann eine Anforderung in einer oder mehreren User Stories abgebildet



werden. In Abhängigkeit vom Kontext lässt sich diese Anforderung zudem aus unterschiedlichen Perspektiven der verschiedenen Personas näher beschreiben. Jede so entstandene User Story besteht aus einem Satz, der vor der Implementierung um Akzeptanzkriterien erweitert und mit den Softwareentwicklern durchlaufen wird. Eine User Story für die oben genannte Persona wäre zum Beispiel:

- ▼ Als studentische Hilfskraft (als Persona Hanna Accuracy) möchte ich direkt den Raum auswählen, in dem ich mich gerade befinde, um eine langwierige Suche nach den Artikeln im Bestand zu vermeiden.

Personas und User Stories helfen den Stakeholdern, die Perspektive des Nutzers einzunehmen, auch wenn sie nicht direkt an der Nutzerforschung beteiligt waren. So bilden die Ergebnisse der Research-Phase die Basis für das Vorgehen und Entscheidungen in den nächsten Phasen. Sie verhindern darüber hinaus eine beliebige Anpassung und Ausdehnung des Designs und der Implementierung.

## Design-Phase – „Visualisiere dein Design Schritt für Schritt!“

Für die in der Research-Phase definierten Personas und User Stories wird nun entsprechend der vorgenommenen Priorisierung ein Design entwickelt. Üblicherweise geht man hier in drei Schritten vor:

- ▼ Im ersten Schritt wird die Informationsarchitektur des Produktes definiert.

- ▼ Darauf aufbauend werden das Interaktions- und das visuelle Design erstellt.

In der Entwicklung des Designs gilt es, eine Vielzahl von Richtlinien wie endgerätespezifische Regeln, Richtlinien zur Barrierefreiheit und Regeln zur Internationalisierbarkeit zu berücksichtigen.

Die Informationsarchitektur lässt sich beispielsweise in Form einer Sitemap darstellen. Die Sitemap stellt die Verknüpfung der einzelnen Seiten dar. Jede Seite kann wiederum um den Inhalt, die jeweils verfügbaren Funktionen, Links und Objekte erweitert werden.

Das Interaktionsdesign baut auf der Informationsarchitektur sowie der vorgenommenen Priorisierung der wichtigsten User Stories auf. Es visualisiert die Struktur der einzelnen Seiten in Form von Scribbles oder Wireframes. Handlungssequenzen können in einer Abfolge von Wireframes oder auch in einem klickbaren Prototyp abgebildet werden. Der Prototyp vermittelt einen ersten Eindruck des Workflows und der Interaktion des Nutzers mit der Anwendung.

Nach der Fertigstellung des Interaktionsdesigns erfolgt ein Review mit den Stakeholdern, um die Umsetzbarkeit entsprechend des verfügbaren Budgets zu überprüfen. Denn gerade in diesem Prozessschritt können Änderungen noch kostengünstig und schnell vorgenommen werden, da in der Regel bislang kaum Programmierkosten aufgelaufen sind. Abbildung 5 zeigt Wireframes für unser Beispiel, die „Inventory-App“.

Nach Freigabe des Interaktionsdesigns entsteht nun im dritten Schritt das visuelle Design. Dieses verbindet in sogenannten Mock-ups das Interaktionsdesign mit dem Look & Feel der

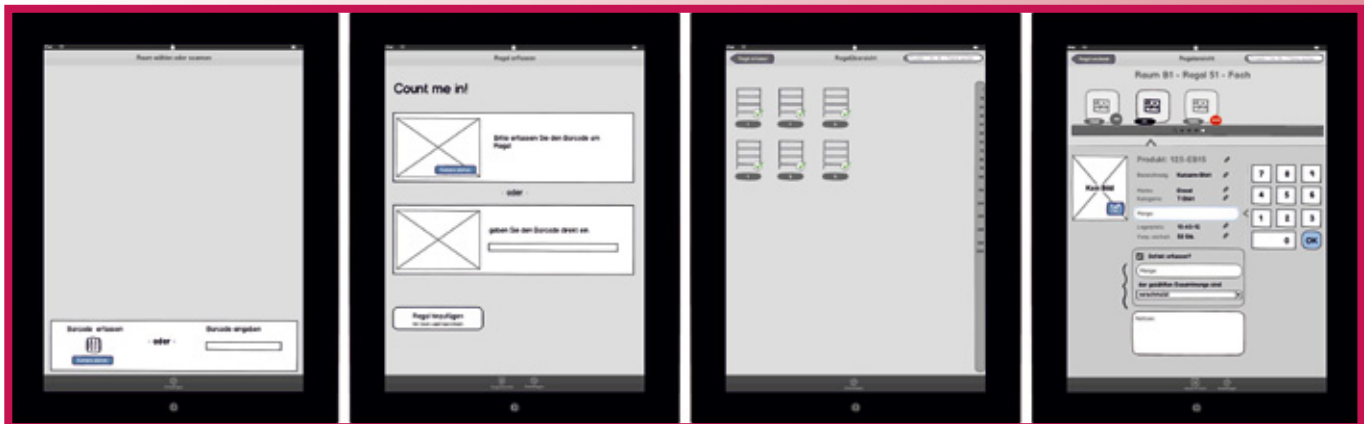


Abb. 5: Interaktionsdesign der „Inventory-App“ in Wireframes



Abb. 6: Visuelles Design der „Inventory-App“ in Mock-ups



Anwendung und orientiert sich dabei am Corporate Design des Unternehmens bzw. der gewünschten zielgruppengerechten Anmutung. Da die Erstellung und Änderung dieser detaillierten Entwürfe (Mock-ups) sehr zeitintensiv ist, werden sie nur für wenige zentrale und im Interaktionsdesign finalisierte Seiten erstellt. Gegebenenfalls lohnt sich die Erstellung eines Prototyps des visuellen Designs, etwa um zentrale Funktionen zu evaluieren. Für unser Beispiel könnten Mock-ups etwa wie in Abbildung 6 aussehen.

## Implement-Phase – „Setze dein Design Schritt für Schritt um!“

Um die Entwicklung eines Produktes zu beschleunigen, können die Entwickler bereits parallel zur Design-Phase mit dem technischen Unterbau des Produktes beginnen. Darüber hinaus werden ihnen bereits finalisierte (Zwischen-)Ergebnisse zur Verfügung gestellt. Als Erstes erhalten die Entwickler die erstellten Personas, die priorisierte Abfolge der User Stories, dann die Informationsarchitektur und daraufhin das Interaktionsdesignkonzept. Während durch den UX-Designer parallel das visuelle Designkonzept entwickelt wird, können die Softwareentwickler bereits mit der Implementierung des Interaktionsdesigns starten.

Nach der Fertigstellung des visuellen Designs erhalten sie auch die Mock-ups. Darüber hinaus werden alle notwendigen Deliverables wie Icons, Buttons und Images im definierten Format zur Verfügung gestellt. Zusätzliche Implementierungshilfe gibt ein Styleguide. Dieser enthält eine Beschreibung der einzelnen Elemente der Benutzungsoberfläche, der Navigation, aller verwendeten Muster sowie die Abmessungen der einzelnen Elemente auf den Bildschirmen. Der Styleguide stellt damit ein konsistentes visuelles Design auch bei einer späteren Erweiterung der Anwendung sicher.

Während der gesamten Implementierung stehen die UX-Experten für Fragen, Reviews und notwendige Anpassungen zur Verfügung. Dies gewährleistet die Realisierung der Softwarelösung entsprechend der im Vorfeld definierten Anforderungen.

## Evaluate-Phase – „Teste deine Resultate!“

Die Evaluation umspannt den gesamten UCD-Prozess. Durch die wiederkehrende Prüfung der Lösung kann die Nutzerorientierung im kompletten Entwicklungsprozess sichergestellt

werden: Probleme werden frühzeitig erkannt, wie beispielsweise geänderte oder neue Anforderungen, und die Usability der Anwendung wird verifiziert. Dadurch können notwendige Änderungen bereits in einem frühen Stadium erfolgen. Für die Evaluation sind verschiedene Methoden geeignet. Abbildung 7 zeigt eine Zusammenfassung von Methoden, die überwiegend in der Praxis Anwendung finden.

Aus der Abbildung geht hervor, dass sich die Ergebnisse der einzelnen Phasen sowohl mit Nutzern als auch mit Experten evaluieren lassen. Welche Methode in welcher Phase des UCD-Prozesses zum Einsatz kommt, hängt vom Ziel der jeweiligen Evaluation und den zur Verfügung stehenden Projektressourcen ab.

Vorteilhaft ist es, Stakeholder als Beobachter in die Evaluation einzubinden. In einem Usability-Test etwa können sie die Interaktion von Nutzern mit einem System live beobachten und erhalten direktes Feedback. Denn ein Workflow oder ein Design kann in ihren Augen logisch begründet sein, dem Nutzer jedoch vollkommen unverständlich erscheinen. Zusätzlich können Schwierigkeiten aufgedeckt werden. An die Tests schließt sich eine gemeinsame Auswertung der Beobachtungen an: Gefundene Probleme werden im Plenum geclustert, entsprechend ihres Schweregrades gekennzeichnet und es werden gemeinsam Lösungen erarbeitet. Im Anschluss daran werden die Zuständigkeit, eine Deadline für die Behebung und ein Kontrollmechanismus für die Überarbeitung definiert. Die Evaluation verhindert damit frühzeitig, dass der Nutzer aus dem Fokus der Entwicklung gerät und dient als Qualitätskontrolle hinsichtlich der Gebrauchstauglichkeit der Lösung.

## Fazit

User Experience überzeugt! Der Erfolg einer Softwarelösung am Markt bestimmt sich heute wesentlich durch die Akzeptanz, ja die Begeisterung der Anwender der Software. Im Artikel ist deutlich geworden, dass die Integration des UCD-Prozesses in den Entwicklungsprozess durchaus möglich ist, wenn sich alle Projektbeteiligten darauf einlassen. Im Ergebnis steigt der Return on Investment (ROI). Durch die Definition und Kalkulation von Kennzahlen, Kosten- oder Gewinn-Vergleichsrechnung oder Geld-Zeit-Wert-Relationen kann bereits bei relativ kleinen Änderungen erfasst werden, inwieweit sich eine Umstrukturierung des Entwicklungsprozesses hin zum UCD-Prozess auszahlt. Den Veränderungsprozess sollte ein UX-Experte begleiten, um das Umdenken hin zu Nutzerorientierung nachhaltig zu verankern.



Abb. 7: Evaluationsmethoden im Überblick



Diplom-Kommunikationspsychologin

**Anja Endmann** arbeitet als User Experience Researcher bei der itCampus GmbH in Leipzig, einem Tochterunternehmen der Software AG. Ihr Aufgabenspektrum umfasst die Bereiche User Research & Requirements Engineering sowie Evaluation von Softwarelösungen mit Experten und/oder realen Nutzern der Anwendung. Während des Studiums an der Hochschule Zittau/Görlitz und ihrer darauffolgenden Tätigkeit an der TU Chemnitz beschäftigte sie sich intensiv mit verschiedenen Methoden zur Erhebung von Anforderungen und Evaluation.  
E-Mail: a.endmann@itcampus.de