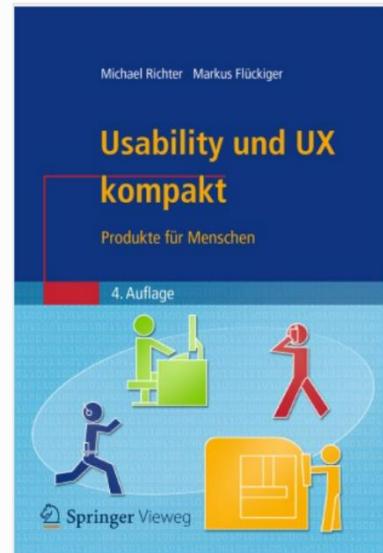


Informationstext UX-Designmethoden

Informationstext zu verschiedenen Methoden des UX-Designs.
Die Quelle der Texte ist das Buch Usability und UX kompakt.
Richter, Flückiger. S. 45ff. Springer Verlag, 4. Auflage
<https://www.springer.com/de/book/9783662498279>



Übersicht über wichtige nutzerzentrierte Methoden:

Methode	Zweck
Contextual Inquiry	Analyse der Benutzer und des Einsatzumfelds des neuen Systems
Personas und Szenarien	Modellieren der unterschiedlichen Benutzergruppen und der Anwendung aus Benutzersicht
Storyboards	Kommunizieren ausgewählter Abläufe mit dem neuen System
UX Prototyping	Entwickeln von Produktideen, Klären der Anforderungen, Konzipieren und Optimieren der Benutzeroberfläche
Use Cases und User Stories	Funktionale Anforderungen in die Entwicklung tragen
Usability Testing	Beurteilen des neuen Systems durch Benutzer
Fragebögen	Sammeln aussagekräftiger Zahlen zur Analyse von Benutzern und Kontext oder zur Beurteilung eines Systems oder Prototyps

Von Beobachtung und Befragung zu Bedürfnissen

Eine neue Software soll Versicherungsberater beim Erstellen von Angeboten unterstützen. Das Projektteam besucht die Berater, um die Arbeitsweise der Berater zu verstehen und eine passende Software zu erarbeiten. Besonders relevant ist das eigentliche Beratungsgespräch, da die Software genial Projektanfrage die Qualität der Beratung verbessern soll. Die Analysten nehmen an einigen Beratungsgesprächen teil. Sie beobachten das Gespräch und befragen zudem den Berater über den Verlauf. Die Beobachtung ermöglicht es den Analysten, wichtige Informationen aufzunehmen, die in einer reinen Befragung nicht auftauchen würden. Die Analysten zeichnen im Detail auf, welche Informationen tatsächlich relevant sind und wie der Berater daraus auf gegebene Versicherungsprodukte schließen. Die Analysten machen sich ebenfalls ein Bild von einigen Software-Anwendungen, die von den Beratern eingesetzt werden, beispielsweise um Berechnungen durchzuführen, Informationen abzulegen, Briefe zu schreiben etc. Diese detaillierten Informationen erlauben es dem Projektteam, ein zu den Aufgaben der Benutzer passendes System zu entwerfen.

Die psychologische Forschung bestätigt, dass Menschen Wissen, das sie in einer bestimmten Situation anwenden, nicht einfach abrufen können. In einem Interview ist solches implizites Wissen nur schwer erfassbar. Die Kombination von Beobachtung und Befragung ermöglicht einerseits, das wirkliche Geschehen im Detail zu erfassen, und andererseits, die Gründe und Zusammenhänge dahinter zu durchleuchten. Diese Informationen sind wertvoll, um den notwendigen Informationsgehalt, die Nutzerführung und die passenden Funktionen des geplanten Systems abzuleiten.

Fragestellung

Mit Contextual Inquiry sollen ausgewählte Fragestellungen beantwortet werden. Lohnenswerte Fragen zielen auf den Einsatz und das Umfeld heutiger Benutzer Produkte ab. Für die Weiterentwicklung eines Navigationsgeräts im Auto kann es beispielsweise aufschlussreich sein, wie die Rollenverteilung zwischen Beifahrer und Fahrer auf einer Urlaubreise ist.

Bevor ein Analyst mit Benutzern spricht, stellt sich das Projektteam die Frage, was in Bezug auf das neue System wichtig zu wissen ist.

Quelle: Usability und UX kompakt. Richter, Flückiger. S. 45ff.
Springer Verlag, 4. Auflage
<https://www.springer.com/de/book/9783662498279>

4.1 Facetten des Handelns: Contextual Inquiry

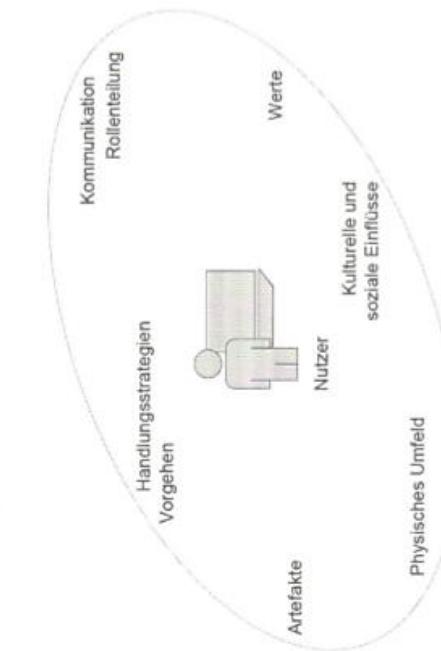
Es locken heiße Features, coole Technologien und verführerisches Spannendes. Sie brauchen nur zuzugreifen. Scheinbar ist die seit über dreißig Jahren aktive Software-Krise nur eine Frage der richtigen Technologie und heute dank Smartwatch, Big Data Analytics und Internet of Things gelöst (ersetzen Sie die Begriffe mit beliebigen Schlagworten der aktuellen IT-Marketing-Maschinerie).

Contextual Inquiry geht die Software-Krise nicht durch neue Technologien an, sondern durch ein fundiertes Verständnis der künftigen Benutzer, ihrer Tätigkeiten und Bedürfnisse. *Contextual Inquiry* lässt sich mit „Erhebung im Umfeld der Benutzer“ übersetzen. Der Analyst untersucht die Bedürfnisse der Anwender, indem er diese bei ihren Tätigkeiten beobachtet und sie dazu befragt.

Natürlich ergeben sich auch Fragestellungen, die mit Contextual Inquiry nicht beantwortet werden können: Beispielsweise wie viel ein Kunde für ein Produkt bezahlen würde (Marketing und Marktforschung), ob eine bestimmte Technologie geeignet ist (Entwicklung) oder wie ein Geschäftsprozess grundsätzlich ablaufen soll (Geschäftsprozessmodellierung).

Contextual Inquiry fokussiert auf die Tätigkeiten der späteren Benutzer und das Umfeld der Anwendung. Tab. 4.2 stellt fünf Sichten dar, die mit der Methode erfasst und dokumentiert werden können (vergleiche auch Abb. 4.1).

In kaum einem Umfeld ist der gesamte Nutzungskontext in nur einer Interviewrunde erfassbar. Die Erfahrung zeigt, dass die Fragestellungen zu Beginn breit und unscharf sind. Im Verlauf des Projektes kennt das Analyseteam den Kontext immer besser, und die Fragestellung wird konkreter und enger gefasst. Es lohnt sich deshalb, mehrere Iterationen durchzuführen. Die Resultate befriedigen jeweils die Fragestellung der nächsten Iteration.



Tab. 4.2 Fragestellungen, die in einer Contextual Inquiry untersucht werden können

Sicht	Fragestellung
Rollenteilung und Kommunikation	Typische Rollenverteilung Aufgaben und Verantwortlichkeiten Kommunikationsmittel!
Kommunikationszweck und Inhalte	Kommunikationszweck und Inhalte
Handlungsstrategien und Vorgehen	Vorteile und Probleme der Rollenverteilung Ausführung von Tätigkeiten Unterschiedliche Vorgehensweisen Stärken und Schwächen
Artefakte	Häufigkeit, Frequenz, Intensität und Dauer der Durchführung Ausnahmesituationen und Fehler, Spezialfälle Bei der Tätigkeit benutzte Dokumente, Formulare, Werkzeuge usw. Aufbau und Informationsgehalt Verwendungszweck Anpassung an individuelle Bedürfnisse Zweckentfremdete Verwendung
Kulturelle und soziale Einflüsse	Vorteile und Schwierigkeiten Personen, die Einfluss nehmen Wirkung von sozialem Druck, Macht ausübung Verhaltensregeln Ziele, Werte und Vorlieben Widersprüchliche Einflüsse Probleme und Chancen auf kultureller Ebene
Nutzer	Raumaufteilung, Arbeitsplatzgestaltung, Einrichtung Verfügbare Hilfsmittel Wege und Distanzen Einfluss auf Kommunikation Verbesserungspotenzial
Physisches Umfeld	

Im Kontext untersuchen

Mit einer ausgewählten Fragestellung stößt der Analyst gezielt in die Welt der Benutzer vor. Die Auswahl der Interviewpartner muss für gewöhnlich nicht repräsentativ im Sinne der Statistik sein, doch sie sollte ein breites Spektrum an Meinungen und Bedürfnissen abdecken. Es lohnt

Abb. 4.1 In einer Contextual Inquiry werden unterschiedliche Aspekte beim Einsatz bestehender Produkte und Systeme untersucht, um den Kontext der geplanten neuen Lösung detailliert zu beleuchten

sich, auf eine gewisse Streuung bezüglich Alter, Geschlecht, Position, Einsatzort, Erfahrung, Fachwissen, kulturellem Hintergrund und mehr zu achten.

Die Untersuchung findet vor Ort und während der Tätigkeit statt. Der Analyst beobachtet den Interviewpartner und stellt gezielt Fragen über das Beobachtete. Der Interviewpartner soll die eigene Handlungswweise reflektieren und so angewandtes Expertenwissen aufdecken.

Um dies zu erreichen, ist eine partnerschaftliche Haltung des Analysten notwendig. In gewissen Situationen kann dafür eine Hilfstechnik nützlich sein, bei welcher der Analyst die Rolle eines Auszubildenden einnimmt. Der Interviewpartner instruiert dabei den Beobachter, damit dieser die Aufgabe verstehen und nachvollziehen kann.

Der Analyst und der Interviewpartner diskutieren, ausgehend von einer gerade vorgeführten Tätigkeit, über Probleme, fachliche Zusammenhänge und Verbesserungsmöglichkeiten. Der Analyst sammelt alles, was im Interview diskutiert wird; ausgefüllte Formulare, Screenshots, Skizzen über interessante fachliche Zusammenhänge, Audioaufnahmen von Gesprächen und mehr.

Ungeplante oder seltene Tätigkeiten zu beobachten kann herausfordernd und zeitaufwändig sein. Solche Situationen lassen sich anhand der dabei entstandenen Artefakte rekonstruieren. Diese Methode eignet sich auch für lange dauernde Tätigkeiten. Ein alternativer Ansatz ist, die Interviewpartner zum Aufnehmen oder Protokollieren der Tätigkeiten zu motivieren („Tagebuchmethode“) und anschließend ein Interview basierend auf den Aufnahmen durchzuführen.

Bei neuartigen Produktentwicklungen können erste Entwürfe, Prototypen oder existierende vergleichbare Produkte in die Interviews einbezogen werden. Je näher die Erhebungssituation an die Realität der geplanten Anwendung heranreicht und je lebensreicher der Kontext dieser Anwendung vermittelt werden kann, desto wertvoller sind die Rückmeldungen der Benutzer.

Analysieren der gesammelten Daten

Die Analyse basiert auf den gesammelten Unterlagen, Notizen, Skizzen, Video- und Audioaufnahmen. Es ist von Vorteil, in einem Team aus Ana-

lysten und Entwicklern zu arbeiten: Analysten stellen sich andere Fragen und suchen nach anderen Lösungsansätzen als Entwickler. Entwickler lernen dabei insbesondere das Anwendungsumfeld kennen. Das Team extrahiert die folgenden Informationen:

- **Ziele und Bedürfnisse der befragten Personen sowie deren Probleme, Werte und Eigenheiten:** In Abschn. 4.2 zeigen wir, wie das Team dar- aus mittels Personas das Zielpublikum charakterisiert.
- **Aufgaben, Abläufe und Tätigkeiten:** Diese Informationen dienen als Grundlage für die Beschreibung der künftigen Abläufe mit dem neuen System.
- **Schwierigkeiten und erprobte Lösungsansätze der Benutzer mit heutigen Werkzeugen:** Diese Informationen schärfen den Blick für die wesentlichen Bedürfnisse der Benutzer und die dazu passenden Funktionen.
- **Begriffe und Informationen zum Fachbereich:** Ein genaues Verständnis der Objekte und Daten des Anwendungsbereits ist für die Gestaltung einer neuen Lösung unverzichtbar. In der Software-Entwicklung ist es geläufig, solche Zusammenhänge in einem Domänenmodell abzubilden. Eine Contextual Inquiry liefert dafür hervorragende Grundlagen. Das Team wird hier vor allem in gesammelten Formularen und Dokumenten sowie bestehenden Produkten und Hilfsmitteln fündig.

Abb. 4.2 zeigt eine effektive Art der Auswertung vieler Daten. Das Team hält relevante Beobachtungen und Erkenntnisse auf Kärtchen fest. In einem Auswertungsworkshop werden diese Kärtchen gemeinsam interpretiert, geordnet und daraus Schlüsse gezogen. Durch die Zusammenarbeit und den aktiven Austausch der Teilnehmer verdichten sich die Erkenntnisse in einem mehrstufigen Prozess zu Anforderungen, die es im Hinblick auf eine neue Lösung zu beachten gilt. Aufgrund der dabei entstehenden Gruppen thematisch zusammenhängender Kärtchen wird diese Technik als Affinitätsdiagramm bezeichnet.

[Beyer et al. 98] beschreiben mit **Contextual Design** eine weiterführende Methodik, um die Resultate aus einer Contextual Inquiry in grafischen Modellen festzuhalten. Die Modelle eignen sich gut zur Verdeutlichung der fünf Sichten des Kontexts und liefern wichtige Ansatzpunkte und Erkenntnisse für die Gestaltung einer neuen Lösung. Zur

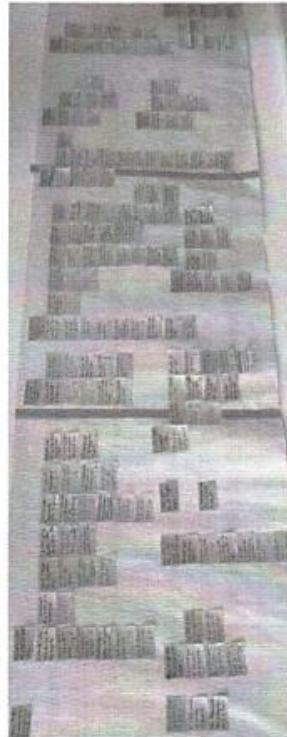


Abb. 4.2 Ein Affinitätsdiagramm führt zu einem aktiven Austausch der Teilnehmer bei der Analyse qualitativer Daten

Untersuchung von Störungsfällen kann beispielsweise die Kommunikation zwischen den beteiligten Personen in einem *Informationsflussmodell* aufgezeichnet werden. Abb. 4.3 zeigt einen Ausschnitt eines solchen Modells.

Geschäftsprozessmodellierung ergänzen

Contextual Inquiry ergänzt die Geschäftsprozessanalyse und -modellierung. Geschäftsprozesse geben die konsolidierten und standardisierten Abläufe in einem Unternehmen wieder. Sie modellieren hingegen nicht die Problemlösungsstrategien, die eine bestimmte Person anwendet, welche Abkürzungen und Optimierungen diese vornimmt und wie sie sich mit den Kollegen abspricht. Geschäftsprozesse sagen auch nichts über das konkrete physische und kulturelle Umfeld aus, in dem die Benutzung stattfindet. Diese Lücke schließt Contextual Inquiry und bringt die Aspekte der täglichen Arbeit und der konkreten Anwendung der Geschäftsprozesse in ein Projekt hinein.

Innovation aus dem realen Leben

Innovation entspringt unterschiedlichen Quellen: Neue und bessere Technologien ermöglichen neue Produkte und eröffnen neue Geschäftsfelder. Durch die Optimierung der Geschäftsprozesse werden neue Wege aufgezeigt und das bestehende Potenzial eines Unternehmens besser ausgenutzt. Contextual Inquiry erschließt eine weitere Quelle für Innovation: Die Methode zeigt verbreitete Muster, erprobte Lösungsansätze, Bedürfnisse und unlöste Probleme der Menschen auf – Faktoren, die Potenzial für wirklich nützliche Produkte bieten, welche die Benutzer direkt ansprechen.

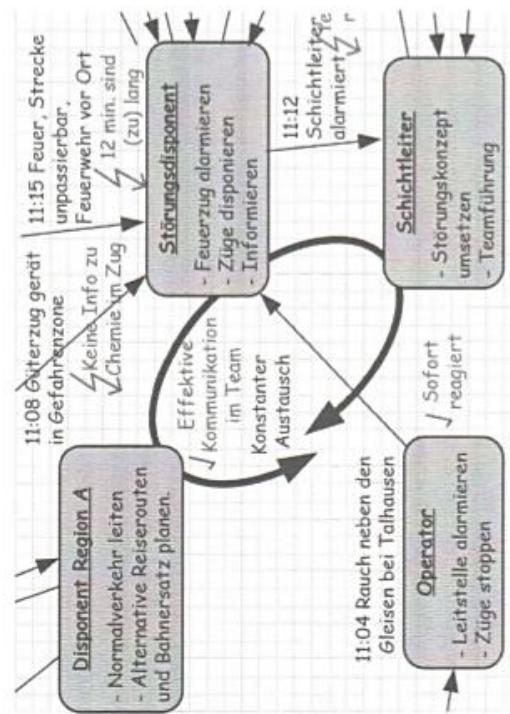


Abb. 4.3 Darstellung der Kommunikation während eines Störungsfalls in einem Informationsflussmodell (Flow Model). Es zeigt die komplexe Kommunikation zwischen den Beteiligten in allen Details und weist auf Stärken und Schwächen hin

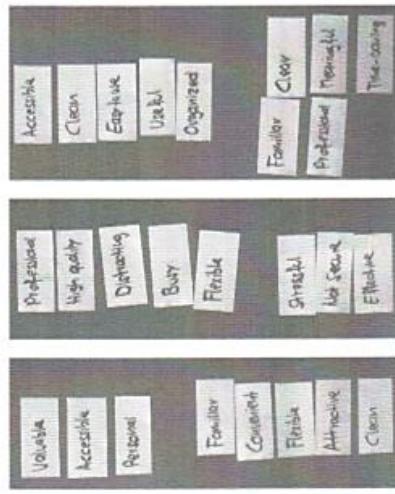


Abb. 4.4 Ein Produkt erzeugt bei seinen Benutzern unterschiedliche Reaktionen. Vorbereitete Antwortkarten bringen diese zum Vorschein. Zusammen mit den erklärenden Kommentaren der Interviewpartner lassen sich Nutzungsergebnisse nachvollziehen

Erfahrungswelt ergründen mit Response Cards

Beobachtungstechniken wie Contextual Inquiry erheben sichtbare Ereignisse. Einem trainierten Beobachter und Interviewer mag es durchaus gelingen, auch die emotionale Reaktion einer Person in Bezug auf eine Situation zu erfassen. Als Türöffner für eine Diskussion lässt sich die Technik der „Emotional Response Cards“ [Benedek et al. 02] verwenden, zum Beispiel im Anschluss an eine Beobachtungssituation. Der Teilnehmer wählt aus einem vordefinierten Set jene Karten aus, die am besten zum gerade Erlebten passen (siehe Abb. 4.4). In Kombination mit erklärenden Kommentaren lässt sich damit einfach und strukturiert seine emotionale Reaktion ergründen.

Anwendung im agilen Umfeld

- In der agilen Entwicklung wird angestrebt, nur so viel Analysearbeit durchzuführen, wie für die anstehenden Iterationen notwendig ist. Eine vorangehende Beobachtungs- und Analysephase mit Contextual Inquiry ist dann angebracht, wenn komplexe Tätigkeiten der Benutzer unterstützt werden sollen, die sich auch mit der neuen Software nicht stark verändern werden. Ansonsten kann es zielführender sein, hypo-

- thesengetrieben vorzugehen: Basierend auf kurzen Analysen und Annahmen über Benutzer und Umfeld werden Prototypen oder Teilprodukte der neuen Lösung entwickelt und diese in kurzen Entwicklungszyklen mit Benutzern validiert, verfeinert oder verworfen.
- Agile Teams sollten darauf achten, nicht zu schnell auf zukünftige Lösungen einzuschwenken, ohne die bestehenden Aufgaben und Handlungsstrategien der Benutzer wirklich verstanden zu haben.

Darauf sollten Sie achten

- Niemand wird gerne beobachtet, wenn nicht klar ist, wozu die Analyse dient. Nur eine Transparenz, offene, interessierte und partnerschaftliche Haltung führt zum Ziel.
- Behalten Sie während der Analyse die ausgewählte Fragestellung im Auge. So manches Projektteam hat sich schon in der Unzahl der Informationen in der Analyse verloren.
- Halten Sie auch fest, worauf Ihre Erkenntnisse zurückzuführen sind. Irgendwann kommt ein Auftraggeber und will wissen, weshalb ein bestimmtes Feature eingebaut wurde.
- Setzen Sie Untersuchungen im Kontext gezielt ein und wägen Sie den Nutzen gegenüber Prototyping bewusst ab: Ist es wichtig die Ist-Situation noch genauer zu verstehen oder geht es bereits darum, die Welt von morgen zu entwickeln?

Personas

Personas stellen prototypische Benutzer dar und verkörpern ihre unterschiedlichen Ziele, Verhaltensweisen und Eigenschaften, die im Hinblick auf das zu entwickelnde Produkt relevant sind. Die Methodik wurde durch den Interaktionsdesigner Alan Cooper eingeführt und publik gemacht [Cooper et al. 10]. Die Namensgebung leitet sich aus dem griechischen Theater der Antike ab. Die *Persona* war eine Maske, welche die Rolle der Schauspieler typisierte und gleichzeitig als Schallverstärker diente – ein treffender Begriff, wie wir finden. Personas charakterisieren Benutzer in bestimmten Rollen. Abb. 4.5 zeigt ein Beispiel einer Persona.

Personas werden aufgrund von Informationen über die zukünftigen Benutzer eines Systems erarbeitet. Dazu dienen beispielsweise Ergebnisse von Usability-Tests.

Quelle: Usability und UX kompakt. Richter, Flückiger. S. 45ff.

Springer Verlag, 4. Auflage

<https://www.springer.com/de/book/9783662498279>

4.2 Modellierte Realität: Personas und Szenarien

Eine Versicherungsgesellschaft entwickelt ein Softwaresystem, um Schadensmeldungen und die daraufführen geleisteten Entschädigungen zu verwahren. In der Spezifikation erscheint der Akteur „Sachbearbeiter“ und als einer der wesentlichen Anwendungsfälle „Schadensmeldung erfassen“. In einem Workshop, in dem ein erster Prototyp mit einigen Benutzern besprochen wird, verläuft die Diskussion etwas hitzig: Während ein Sachbearbeiter den Vorschlag ganz gut findet, da alle notwendigen Informationen übersichtlich dargestellt seien, stößt ein anderer über die vielen Eingabefelder und Abhängigkeiten. Es stellt sich heraus, dass der Erste das bestehende System täglich benutzt und viele Fälle erfasst, während der andere nur einige Male pro Monat Daten für wenige Spezialfälle eingibt und deshalb viele der Funktionen gar nicht benötigt. Der Prototyp erfüllt offenbar nur die Anforderungen des ersten Benutzers, nicht aber jene des zweiten.

Dieser Abschnitt befasst sich mit den Methoden *Personas* und *Szenarien*. Es handelt sich dabei um zwei Instrumente, um die unterschiedlichen Bedürfnisse der Benutzer zu modellieren und daraus passende Lösungen abzuleiten.

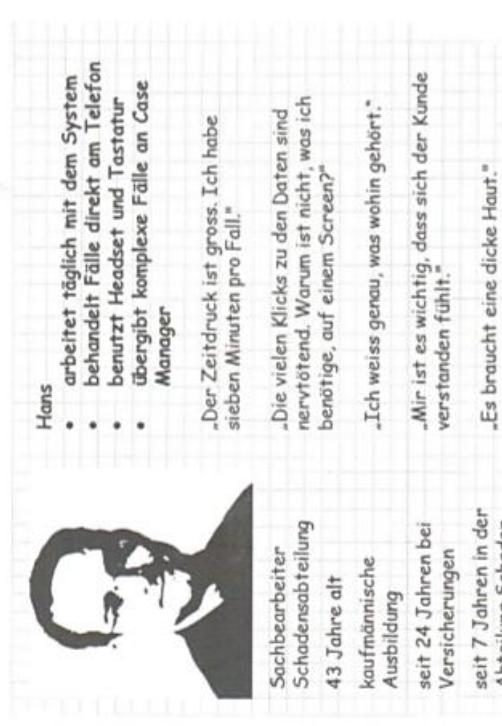


Abb. 4.5 Personas illustrieren wichtige Eigenschaften und Bedürfnisse der Benutzer im Hinblick auf ein geplantes neues Produkt

nisse aus Workshops mit Benutzern, Contextual Inquiries, Fragebögen oder Usability Walkthroughs mit bestehenden Systemen. Der Analyst entwirft Vorschläge und validiert diese mit den Beteiligten, oder die Personas werden in einem gemeinsamen Workshop erstellt. Eine Persona sollte schließlich die für das Produktdesign relevanten Eigenschaften der Benutzer widerspiegeln.

Im obigen Beispiel könnten zwei Personas, nennen wir sie Hans und Niklaus, für zwei unterschiedliche Benutzergruppen stehen. Hans und Niklaus unterscheiden sich deutlich in ihren Zielen und arbeiten ganz unterschiedlich mit der Applikation. Während Hans mehrere Fälle pro Tag bearbeitet und das heutige System sehr gut kennt, arbeitet Niklaus nur gelegentlich am System für einzelne, komplexere Fälle.

Was Sie über Ihre Benutzer festhalten sollten

Im Rahmen eines Projekts entstehen mehrere Personas, die jeweils einen typisierten Benutzer beschreiben. Eine Persona kann über folgende Eigenschaften Auskunft geben:

- Ziele der Benutzer,
- Beruf, Funktion, Verantwortlichkeiten und Aufgaben,
- Fachliche Ausbildung, Wissen und Fähigkeiten,
- Verhaltensmuster und Vorgehensweisen,
- Werte, Ängste, Sehnsüchte, Vorlieben,
- Allgemeine Computerkenntnisse,
- Kenntnisse über verwandte Produkte, Vorgängersysteme, Konkurrenzprodukte,
- Verbesserungspotenzial in der heutigen Situation,
- Erwartungen an eine neue Lösung.

Der erstellte Charakter soll einprägsam sein. Seine Eigenschaften sollen einfach verinnerlicht werden können. Dazu kann eine Persona mit zusätzlichen passenden Informationen zum Leben erweckt werden:

- Name, Alter, Geschlecht,
- Markige Charakterzüge,

- Bild, Skizze, Porträt,
- Passende Zitate aus Interviews,
- Ein Tag im Leben von ...

Insbesondere die weichen Kriterien wie Ziele, Werte und Ängste erscheinen einem Ingenieur im ersten Moment zwecklos. Die Handlungswweise von Menschen in bestimmten Situationen wird indessen stark von genau solchen Faktoren beeinflusst.

Fokussierung der neuen Lösung

Das Projektteam lässt den Blick über die Galerie der Charaktere schweifen. Ein Analyst meint: „Hans und Niklaus sind sehr verschieden. Müssen wir da nicht zwei User Interfaces erstellen?“ Die Diskussion ist eröffnet. Wie soll das Projektteam mit den unterschiedlichen Personas umgehen? Gibt es wichtige und weniger wichtige Benutzergruppen?

Personas lassen sich wie in Tab. 4.3 dargestellt klassifizieren. Die Einstufung der Personas Hans und Niklaus ist eine bewusste Entscheidung. Handelt es sich um zwei primäre Personas, dann wird das Projektteam zwei optimierte Benutzerschnittstellen entwerfen. Wäre Niklaus hingegen eine Non-Persona, dann würde bewusst keine Optimierung für diese Benutzergruppe erfolgen.

Tab. 4.3 Die Klassifikation der erarbeiteten Personas fordert die Fokussierung auf bestimmte Benutzergruppen und damit eine Priorisierung der Anforderungen bei der Gestaltung eines neuen Produkts

Typ	Bedeutung
Primäre Persona	Für deren Bedürfnisse und Anforderungen wird das Produkt optimiert und die Benutzerschnittstelle erstellt
Sekundäre Persona	Bedürfnisse sind größtenteils durch eine primäre Persona abgedeckt. Nur kleine Erweiterungen notwendig
Ergänzende Persona	Bedürfnisse sind vollständig durch eine primäre Persona abgedeckt
Non-Persona	Eine Persona, die vom Projektteam explizit nicht berücksichtigt wird

Denkanstoß

Denken Sie an ein spannendes Projekt zurück. Wurde über Benutzer diskutiert? Wurden gewisse Benutzerkreise ausgeklammert, bewusst oder unbewusst? Wie hätte die Verwendung von Personas dieses Projekt beeinflusst?

Szenarien

Anwendungsszenarien oder kurz **Szenarien** sind ein zentrales Element in jeder nutzerorientierten Entwicklung. Sie schlagen die Brücke zwischen den Anforderungen und dem Entwurf einer neuen Lösung.

Ein Szenario beschreibt in Form eines realistischen Beispiels, wie ein Benutzer mit dem geplanten System interagieren wird. In einfachen Sätzen oder mittels Kärtchen an einer Pinnwand wird ein konkreter Ablauf aus Benutzersicht im Anwendungskontext dargestellt. Dabei sollte, wie auch bei Personas, mehr auf inhaltlich richtige Aussagen als auf deren formale Korrektheit geachtet werden.

Abb. 4.6 zeigt ein kurzes Szenario zur Beschreibung einer neuen Versicherungsapplikation. Es beschreibt in wenigen Sätzen, wie die Aufnahme eines Schadensfalls in Zukunft mit dem neuen System ablaufen soll. Das Szenario spiegelt eine Reihe zusammengehöriger Anforderungen aus Benutzersicht wider:

Szenario 1: Aufnehmen eines Schadensfalls

Es ist 15:00 Uhr. Bei Hans klingelt das Telefon. Auf seinem neuen Laptop-Bildschirm erscheinen neben der Telefonnummer Name und weitere Angaben des anrufenden Kunden. Hans nimmt den Anruf entgegen und begrüßt den ungeduldigen Kunden, der eine kaputte Fensterscheibe melden möchte. Der Kunde hat die Nummer seiner Police vor sich und so kann Hans die passende vom Übersichtsbildschirm auswählen. Danach nimmt er den Schadensfall des Kunden auf.

Abb. 4.6 Ein Szenario illustriert die Anwendung einer geplanten neuen Lösung

- Automatische Anzeige des Namens und Angaben des Anrufers,
- Darstellung aller bestehenden Versicherungspolicen eines Kunden,
- Kurze Antwortzeiten des Systems.

Szenarien werden basierend auf den Anforderungen an ein neues System erstellt. Sie können iterativ entwickelt oder in Workshops zusammen mit Benutzern erarbeitet werden. Ein großer Vorteil von Szenarien ist ihre leichte Verständlichkeit. Sie können von verschiedenen Stellen wie Analysten, Benutzer und Entwicklung schon zu einem frühen Zeitpunkt überprüft, ergänzt oder korrigiert werden. Mit anderen Worten: Der Analyst *modelliert* mit Szenarien die Anforderungen an ein neues System. Folgende Eigenschaften zeichnen ein Szenario aus:

- Es wird für eine bestimmte Benutzergruppe entworfen, berücksichtigt ihre Eigenschaften und erfüllt ihre Bedürfnisse.
- Es stellt einen konkreten Fall aus der Anwendung dar.
- Es zeigt, wie die Benutzer das neue Produkt in ihrem realen Umfeld einsetzen werden.
- Es illustriert die für die Entwicklung der neuen Lösung relevanten Aspekte.
- Es beschränkt sich nicht auf den Schönwetterfall, sondern beschreibt auch exemplarisch wichtige Ausnahme- und Fehlersituationen.

Verwendung von Szenarien

Szenarien können zu unterschiedlichen Zeitpunkten in der Entwicklung einer neuen Lösung und für verschiedene Ziele eingesetzt werden:

- *Erhebung und Validierung von Anforderungen:* Die Reflektion am konkreten Beispiel erlaubt es Auftraggeber und Benutzern, Anforderungen in der konkreten Anwendungssituation zu vergewissern, zu überprüfen und zu ergänzen. Szenarien können als erste Prototypen eines neuen Systems betrachtet werden.
- *Spezifikation:* Szenarien illustrieren die Anwendung im realen Kontext und dienen als Ergänzung des Use-Case-Modells (siehe Abschn. 4.5). Sie vermitteln den Entwickler ein Verständnis der

- **Abläufe und Zusammenhänge.** In agilen Projekten sind Szenarien eine gute Quelle für die Erstellung von User Stories (vergleiche Abschn. 4.5). Szenarien bringen die konkrete Anwendung in die Diskussion ein.
- **User-Interface-Konzept:** Szenarien dienen dazu, die Abläufe der Benutzerschnittstelle zu beschreiben. Damit kann die Interaktion modelliert und mit Benutzern optimiert werden. Die technischen Anforderungen können von Entwicklern überprüft werden.
- **Usability-Testszenarien** (siehe Abschn. 4.7): Szenarien dienen als Basis für die Evaluation eines Systems oder eines Prototyps zusammen mit Benutzern.
- **Testszenerien:** Aus den erstellten Szenarien können Testszenerien für die Prüfung der entwickelten Software abgeleitet werden.
- **Schulung:** Szenarien dienen zur Schulung von Benutzern und als Basis für die Erstellung von Anleitungen.

Diese Durchgängigkeit über den gesamten Entwicklungsprozess macht Szenarien zu einem äußerst effektiven Instrument in der Entwicklung interaktiver Systeme. Für eine weiterführende Vertiefung der szenarioorientierten Entwicklung möchten wir an dieser Stelle auf das Buch [Rosson et al. 02] verweisen.

Hintergrund: Die Macht des guten Beispiels

Ein Analyst achtet darauf, formal korrekt und präzise zu formulieren. Die Spezifikation eines neuen Systems darf letztendlich nur wenig Interpretationsspielraum zulassen. Um formal korrekt zu formulieren, muss zwangsläufig über verschiedene mögliche Fälle generalisiert werden. Die Gefahr dabei ist, dass die Realität auf der Strecke bleibt.

Der Sachbearbeiter nimmt eine Schadensmeldung eines Kunden auf ist eine formal korrekte Formulierung des Sachverhalts im obigen Beispiel, sie sagt allerdings wenig über die tatsächliche Situation aus. Natürlich wird im Beispiel nicht nur der Versicherungsmitarbeiter Hans mit dem neuen System arbeiten. Es sollen damit auch nicht nur Schadensmeldungen für kaputte Fensterscheiben erfasst werden. Eine aus Benutzersicht zentrale Anforderung ist im vorliegenden Fall, dass der Benutzer aufgrund der Kundenanfrage (eine kaputte Fensterscheibe) schnell und eindeutig (denn der Kunde wartet am Telefon) die richtige Versicherungspolice (z. B. eine Hausratversicherung) zuordnen kann. Diese Anforderung wird erst durch die Schilderung der konkreten Anwendungssituation ersichtlich und kommt in einer formal korrekten, generalisierten Darstellung nur schwer zum Ausdruck.

Ein Beispiel dagegen ist weder eindeutig noch vollständig. Interessanterweise ist das menschliche Gehirn hervorragend dafür geeignet, aus Beispielen Regeln abzuleiten. Mittels weniger, guter Beispiele kann ein Sachverhalt oft schneller, umfassender und manchmal sogar präziser dargestellt werden als mit einer formalen Spezifikation. Personas und Szenarien nutzen diese Tatsache aus. Indem sie wichtige, stimmige und realistische Beispiele wiedergeben, können sie die Anwendung eines geplanten Systems schon früh im Entwicklungsprozess relativ genau umreißen, ohne dass die Details präzisiert werden müssen.

Die Benutzerperspektive

Mit Personas und Szenarien kann das Projektteam die Perspektive der Benutzer einnehmen und aus deren Sicht diskutieren. In erster Linie soll damit das System oder Produkt entworfen und die Benutzerschnittstelle optimiert werden. Zum Beispiel könnte ein Szenario zeigen, in welcher Reihenfolge Hans Informationen sucht, liest und eingeht. Der Vergleich zu einem analogen Szenario mit Niklaus würde die Unterschiede im Vorgehen sichtbar machen. So kann für unterschiedliche Benutzergruppen die richtige Mischung zwischen unterstützenden, beschränkenden und flexiblen Funktionen definiert werden.

Personas und Szenarien dienen einem Projektteam auch zur Beurteilung von Konkurrenz- oder Vorgängerprodukten: Wie gut wird eine Person den im Szenario skizzierten Fall lösen? Dies gibt wertvolle Hinweise über Stärken und Schwächen anderer Lösungen.

Durch die Benutzerperspektive ändert sich die Diskussion. Die Teilnehmer werden zur *Perspektivenübernahme* (vergleiche Abschn. 1.2) ermuntert. Sie diskutieren aus der Sicht der Persona, statt aus einer aufgrund eigener Erfahrungen oder künstlerischen Vorstellungen gebildeten Individualsicht. Die Diskussion wird dadurch objektiver. Statt darüber zu streiten, ob der Benutzer dieses oder jenes Konzept verstände oder nicht, lässt sich die Frage untersuchen, welche Persona das Konzept kennt. Je fundierter die Daten, die zu den Personas führen, desto objektiver die Diskussion.

Anwendung im agilen Umfeld

- Personas sind in der agilen Produktentwicklung ein geeignetes Hilfsmittel, um eine gemeinsame Vorstellung der Zielgruppen zu vermitteln. Im Team-Raum als Poster aufgehängt lassen Personas die zukünftigen Benutzer präsent werden.
- *Ad-hoc Personas* verkörpern erste Hypothesen des Zielpublikums. In jeder Feedbackschleife der agilen Entwicklung kann das Team die Hypothesen aufgrund fokussierter Analysen prüfen, schärfen und erweitern.
- Szenarien nehmen in der agilen Entwicklung eine wichtige Stellung ein. Die im Backlog verfeinerten Einträge enthalten wenig Kontext. Szenarien zeigen im Zusammenhang auf, wie Anwender ihre Aufgaben mit dem neuen System erledigen werden. Sie ermöglichen den Entwicklern zu erkennen, wie sich ein Eintrag im Backlog auf die Nutzung auswirkt. Das Team kann das Produkt entlang der Szenarien auf Durchgängigkeit prüfen.

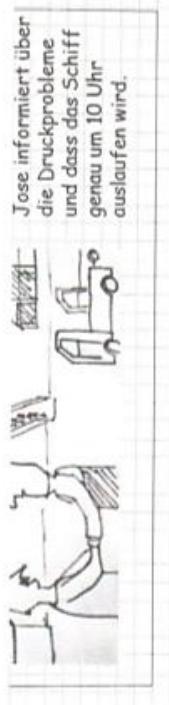
Darauf sollten Sie achten

- Wenn immer möglich sollten Personas aufgrund von Erkenntnissen über die (zukünftigen) Benutzer abgeleitet werden, beispielsweise aus Ergebnissen von Interviews, Contextual Inquiries, Beobachtungen oder Befragungen. Es besteht sonst die Gefahr, dass aufgrund falscher Vorstellungen an der Zielgruppe vorbeientwickelt wird.
- Es ist nicht immer möglich, mit Benutzern zu sprechen. Hier leisten Personas einen wertvollen Dienst, um Annahmen über die Benutzer aufzudecken, unterschiedliche Verständnisse zu diskutieren und ein gemeinsames Verständnis zu erreichen. In diesem Fall können sekundäre Informationsquellen genutzt werden, wie beispielsweise externe Fachexperten, Mitarbeiter aus dem Support Team und Leiter der Anwenderschulungen.
- Personas erlauben einem Projektteam, bewusst auf die relevanten Eigenschaften der Benutzer zu fokussieren. Sie sind somit Teil der Projektbegrenzung und ein wichtiges Mittel für die Planung nutzerorientierter Aktivitäten.

Die Anwendung visualisieren

Ein Storyboard zeigt mithilfe der Benutzerschnittstelle, wie ein System oder Produkt verwendet wird. Es stellt wichtige Aspekte der Anwendung bildlich dar und dient damit der Kommunikation zwischen allen Beteiligten.

Quelle: Usability und UX kompakt. Richter, Flückiger. S. 45ff.
Springer Verlag, 4. Auflage
<https://www.springer.com/de/book/9783662498279>



October 18, 4:10 am

Rudi schaut die Servicerapporte auf dem Tablet durch. Steffen, der Azubi, bedient das Tablet für Rudi.

Einige Teile müssen ersetzt werden und Rudi plant, dies heute zu tun.

Sitz-Daten	
Werk	Teile
Stoff: Halbton	Stoff: -
Farbe: Grau	Farbe: Grau
Material: Kunststoff	Material: -
Länge: 234,64	Länge: -
Fläche: 110,43	Fläche: -
Volumen: 25,04	Volumen: -
HM: 59423	HM: -
Spalte	Breite



October 18, 5:42 am

Rudi hat die Turbine gecheckt und will einen Komplettservice durchführen. José gestattet dies nicht, er befürchtet eine Verspätung. Schließlich setzt sich José durch und übernimmt die volle Verantwortung.



Abb. 4.7 Mit Hilfe eines Storyboards kann ein erster Eindruck der geplanten neuen Anwendung vermittelt werden

4.3 Die Vision erlebbar machen: Storyboards

In einer Fünfsekunde kann man eine Botschaft rund um die Welt senden. Aber es kann Jahre dauern, bis sie von der Aufenseite eines Menschenschädels nach innen dringt. (Charles F. Kettering)

Dieser Abschnitt stellt Storyboards vor, ein Mittel zur Kommunikation zwischen Auftraggebern, Fachvertretern, Benutzern und Entwicklern. Storyboards werden auch in anderen Gebieten, zum Beispiel in der Filmbranche, eingesetzt. Das Storyboard hilft dem Regisseur dabei, den Schauspielern und dem Filmteam den Aufbau des Films zu vermitteln. Es visualisiert Aspekte wie Perspektive, Beleuchtung, Gesichtsausdrücke, Kostüme und so weiter.

ten. Im Wesentlichen handelt es sich dabei um die Visualisierung eines Szenarios (vergleiche Abschn. 4.2).

Abhängig vom Kommunikationszweck kann ein Storyboard in unterschiedlichen Ausprägungen erstellt werden. Die Palette reicht von skizzartigen oder realistisch gestalteten Abfolgen der Benutzerschnittstelle (*User Interface Storyboard*) bis zu Bildergeschichten, die auch Kontext und handelnde Personen darstellen. Abb. 4.7 zeigt einen Ausschnitt aus einem Storyboard.

Storyboards kommen in Situationen zum Einsatz, wo Text alleine nicht ausreicht. Zwei wichtige Gründe sprechen für eine solche Visualisierung:

- In Bildern können Aspekte vermittelt werden, die mit Text nicht oder nur schwer auszudrücken sind, beispielsweise neuartige Konzepte, für die es noch keine Begriffe gibt.
- Mit der visuellen Umsetzung können Erlebnisse, die für die Anwendung von Bedeutung sind, besser in die Welt des Zielpublikums transporiert werden.

Ein Storyboard eignet sich deshalb, um folgende Gesichtspunkte aufzuzeigen:

- Dialogabläufe der Benutzerschnittstelle,
- Schwer verständliche Konzepte oder Sachverhalte,
- Wichtige Aspekte des Anwendungskontexts,
- Spezielle oder komplexe Umgebungen, in denen das System eingesetzt wird.

Eine Geschichte erzählen

Ein Storyboard erzählt die Geschichte, wie die Benutzer ein neues System nutzbringend einsetzen werden. Eine solche Geschichte vermittelt Vorschläge und Entscheidungen über Funktionsumfang, Gestaltung, internen Aufbau und mehr. Das Storyboard stellt eine implizite Frage an die Zuhörer: „Wir als Projektteam denken, dass diese Lösung eure Bedürfnisse erfüllt und so realisiert werden kann. Wo irren wir uns und wo

habt ihr Bedenken?“ Damit dies erreicht werden kann, sollten die folgenden Aspekte beachtet werden:

- Die Geschichte erzählt ein konkretes Fallbeispiel.
- Sie ist örtlich und zeitlich eingordnet.
- Sie erklärt die Zusammenhänge und stellt die kritischen Punkte deutlich dar.
- Der dargestellte Fall sollte kein Trivialfall sein.
- Die handelnden Personen werden charakterisiert.
- Die Geschichte begründet plausibel, warum die Personen so handeln.

Die Realitätsnähe und die Details der kritischen Punkte bieten Anlass zu interessanten Diskussionen, in denen Missverständnisse und Diskrepanzen aufgedeckt werden.

Was sollte ein Storyboard enthalten?

Ein Storyboard wird mit den Erkenntnissen im Verlauf eines Projekts präzisiert. Sind am Anfang nur erste Ideen oder verschiedene Varianten skizziert, so beinhaltet ein Storyboard später die bereits getroffenen Entscheidungen. Es enthält Aussagen zu folgenden Aspekten des geplanten Systems:

- Berücksichtigte und nicht berücksichtigte Bedürfnisse,
 - Änderungen der Geschäftsprozesse,
 - Neuerungen in der Arbeitsweise oder Tätigkeit,
 - Enthaltene bzw. ausgeklammerte Funktionen,
 - Den grundsätzlichen Aufbau der Benutzerschnittstelle,
 - Ausgewählte User-Interface-Details.

Diese Liste ist nicht vollständig. Abhängig vom Kommunikationszweck müssen auch nicht zu jedem Punkt Aussagen vorhanden sein.

Zielgerichtet kommunizieren

Storyboards können in verschiedenen Situationen und für unterschiedliche Zwecke eingesetzt werden:

- Zur Diskussion einer Idee oder einer ausgearbeiteten Lösung mit Benutzern und weiteren Stakeholdern.
- Um das korrekte Verständnis der Bedürfnisse und der fachlichen Zusammenhänge zu prüfen und Missverständnisse auszuräumen.
- Zur Diskussion von Vor- und Nachteilen verschiedener Varianten.
- Um über Neuerungen zu informieren und damit beispielsweise Akzeptanz für das neue Werkzeug zu erzeugen.
- Um neugierig auf das Neue zu machen.
- Um Führungskräfte darüber zu informieren, wie ihre Vision durch die neue Lösung Realität wird.
- Um Entwicklern die relevanten Anforderungen der Benutzung näherzubringen und zu zeigen, warum gewisse Entscheidungen getroffen wurden.
- Um Benutzern im Rahmen einer Aushildung einen Überblick über das System zu geben.
- Für Projektmarketing bei Auftraggebern, Geschäftsleitung und Benutzern.

Varianten und verwandte Techniken

Storyboards können in den unterschiedlichsten Varianten verwendet werden und der Kreativität sind wenig Grenzen gesetzt:

- Mit drei bis fünf handgezeichneten Bildern und wenigen Worten lässt sich eine Produktidee sehr anschaulich darstellen. Das Team kann Situationen für ein Storyboard auch selber nachstellen, fotografieren und so das Storyboard noch lebensechter gestalten.
- Aufwändiger aber auch eindrucksvoller sind Videoprototypen: Anstatt eine Bildergeschichte zu erstellen, dreht das Team ein kurzes, aussagekräftiges und oft auch unterhaltsames Video.
- Für Detaildiskussionen eignen sich User Interface Storyboards. Sie zeigen die detaillierte Interaktion mit dem geplanten Produkt entlang eines Szenarios. Das Team verwendet dazu Skizzen oder ausgereifte Darstellungen des vorgesehenen User Interfaces. Auch komplexe Abläufe lassen sich anhand solcher Storyboards bereits sehr anschaulich diskutieren.

Eine empfehlenswerte Lektüre ist „Sketching User Experiences“ [Buxton et al. 12].

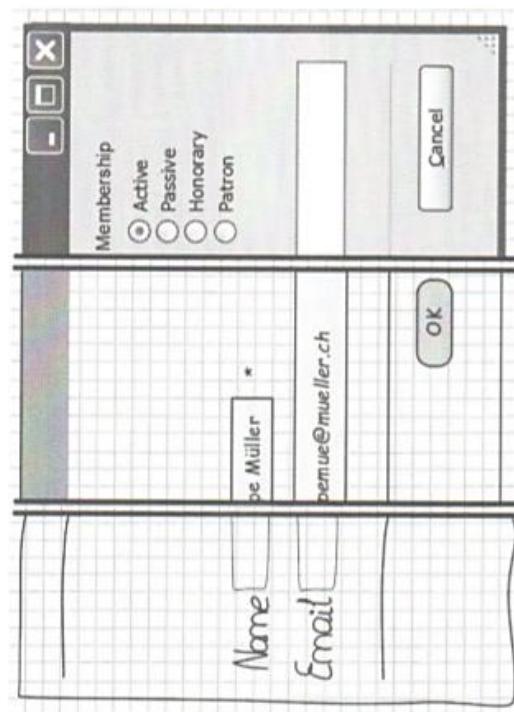
Die Realität in den Workshop holen

Storyboards sind ein ausgezeichnetes Mittel, um die Realität der Anwendung in einen Workshop einzubringen. Die Teilnehmer diskutieren basierend auf der erzählten Geschichte über Annahmen, reflektieren Unterschiede zur heutigen Situation und klären Missverständnisse an realen Beispielen.

Ein User Interface Storyboard kann als Grundlage für eine angeregte Diskussion mit Fachvertretern und Benutzern dienen. Diese können so die Konsequenzen für ihre Tätigkeit abschätzen und die beste Lösung wählen.

- *Funktionsumfang:* Welche der vorgesehenen Funktionen sollen im Prototyp gezeigt werden? Sind dies ausgewählte Ausschnitte, oder geht es darum, den gesamten Umfang darzustellen?
- *Funktionstiefe:* Wie detailliert sollen die einzelnen funktionalen Elemente wiedergegeben werden? Sollen beispielsweise mehrstufige Berechnungen nur ange deutet werden, oder sind die Zwischenschritte und ihre Resultate entscheidend?
- *Darstellungstreue:* Wie ähnlich soll der Prototyp dem Endprodukt in Bezug auf Ausschén der Benutzeroberfläche (Look&Feel) sein? Abb. 4.8 zeigt unterschiedliche Ausprägungen bezüglich Darstellungstreue.
- *Interaktivität:* Wie interaktiv soll der Prototyp sein? Braucht es laufähige Beispiele, um komplexe Abläufe wiederzugeben, oder genügen statische Darstellungen der Benutzerschnittstelle?

Quelle: Usability und UX kompakt. Richter, Flückiger. S. 45ff.
 Springer Verlag, 4. Auflage
<https://www.springer.com/de/book/9783662498279>



4.4 Kritzein für Fortgeschritten: UX Prototyping

User Experience Prototyping (UX Prototyping) wird eingesetzt, um Produkte und Aspekte der Benutzerschnittstelle zu entwerfen, zu evaluieren und zu verbessern, noch bevor ein lauffähiges System vorhanden ist. Weil dabei oft einfache Werkzeuge wie beispielsweise Papier und Bleistift zum Einsatz kommen, spricht man auch von **Lo-Fi Prototyping** (von englisch *Low Fidelity*: geringe Wiedergabebreite).

Dimensionen eines Prototyps

Abhängig vom Ziel, das verfolgt wird, können unterschiedliche Arten von Prototypen zum Einsatz kommen. Um den geplanten Prototyp näher zu charakterisieren, lassen sich die folgenden Dimensionen unterscheiden:

Abb. 4.8 Unterschiedliche Ausprägungen von Prototypen bezüglich Darstellungstreue. *Links:* einfache Handskizze; *Mitte:* Drahtmodell (Wireframe); *Rechts:* endgültiges Look&Feel

- **Datengehalt:** Sollen reale Daten zum Einsatz kommen, genügen realistische Beispiele oder gar Platzhalter für Zeichnungen und dargestellte Informationen? Wie relevant ist die dargestellte Menge an Informationen?
- **Technische Reife:** Wie viel der endgültigen User-Interface-Technologie soll im Prototyp verwendet werden? Muss der Prototyp mit der Entwicklungsumgebung der Zielpлатform entwickelt werden, oder sind einfache Zeichnungswerzeuge ausreichend?

Jeder Prototyp stellt einen Kompromiss zwischen notwendigem Aufwand und Zweck dar. Bevor Sie mit UX Prototyping loslegen, sollten Sie sich deshalb im Klaren sein, welche Fragestellungen Sie verfolgen. Daraus lässt sich ableiten, welche Art von Prototyp geeignet ist. Die folgenden Abschnitte beinhalten einige typische Verwendungszwecke.

Produktideen entwickeln und Anforderungen schärfen

Das Team skizziert aufgrund der mittels Contextual Inquiry vor Ort gesammelten Informationen erste Entwürfe der Benutzerschnittstelle mit Papier und Bleistift. Bestehende Formulare und Applikationen geben Auskunft über Begriffe und Daten. Die beobachteten Abläufe zeigen, in welcher Reihenfolge die Benutzer diese Informationen verwenden. Die Entwürfe werden zu einer ersten Simulation der Benutzerschnittstelle zusammengestellt und mit den Benutzern ausprobiert.

Man spricht bei Attrappen wie im obigen Beispiel auch von **Mock-ups**. Mit Mock-ups können Benutzer bereits konkrete Fälle durchspielen und diskutieren. Dabei geht es nicht darum, das System zu entwerfen, sondern Lösungsideen mit einem einfachen Hilfsmittel erfahrbar zu machen. Abb. 4.9 zeigt ein Beispiel eines solchen Mock-ups. Das Projektteam deckt dabei Bedürfnisse der Benutzer auf, klärt fachliche Missverständnisse und vertieft die Anforderungen:

- Notwendiger Informationsgehalt,
- Passende Funktionen und Abläufe,
- Einbettung in die Geschäftsprozesse,
- Datenaustausch mit anderen Systemen und Applikationen,

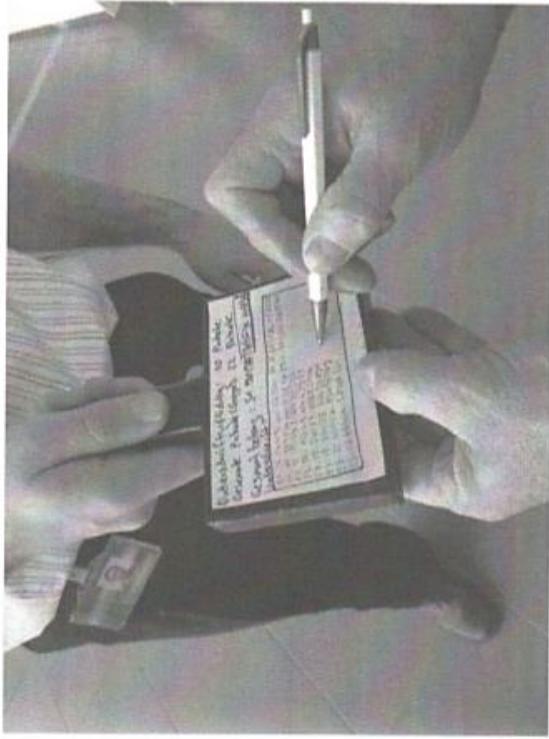


Abb. 4.9 Eine einfache Attrappe („Mock-up“) eines geplanten neuen mobilen Smartphones für die Paketzustellung macht das Gerät bereits in einer frühen Konzeptphase erlebbar

- Darstellung von Tabellen, Grafiken, Funktionen usw.
- Wichtige Details der Benutzerschnittstelle.

Die Benutzerschnittstelle konzipieren

Eine der zentralen Aufgaben eines nutzerorientierten Vorgehens ist es, ein für die Aufgaben der Benutzer und das angestrebte Nutzungserlebnis geeignetes **User-Interface-Konzept** zu erarbeiten. Dabei geht es darum, die Grundsätze der Benutzerschnittstelle festzulegen. Wie bewegen sich die Benutzer durch Menüs und Dialoge? Wie wird die Information strukturiert und dargestellt? Muss das System für spezielle Technologien, zum Beispiel für die Bedienung per Touchscreen, optimiert werden?

- Als Ausgangslage dienen die erarbeiteten Personas und Szenarien und die vereinbarten Anforderungen. Einfache Prototypen helfen dem User Interface Designer, die Szenarien durchzuspielen und ein geeignetes Konzept auszuarbeiten. Das User-Interface-Konzept sollte schließlich folgende Aspekte beinhalten:
 - Grundsätzlicher Aufbau und Screen-Layout,
 - Anzeige- und Eingabegeräte,
 - Aufteilung und Struktur von Informationen,
 - Verwendung und Verhalten von Fenstern,
 - Wichtige Bedienelemente,
 - Navigation mittels Menüs, Schaltflächen und Links,
 - Prüfung von Eingaben und Anzeige von Fehlermeldungen,
 - Konzepte für das Speichern von Informationen und Zuständen,
 - Rückgängig machen und erneut ausführen,
 - Interaktionsprinzipien wie Gesten, direkte Manipulation, Drag&Drop oder Kontextmenüs.
- Dabei muss auch die Technologie der Ziell платформ berücksichtigt werden, beispielsweise Eingabe- und Ausgabemedien, Betriebssystem, Bildschirmgrößen und Auflösung.

- Erlaubt die Benutzerschnittstelle flüssiges Arbeiten?
 - Gibt es Hürden oder Stolpersteine für Personen, die das System zum ersten Mal benutzen?
 - Ist die Navigation effizient?
 - Finden Benutzer die gewünschte Information?
 - Werden Warnmeldungen bemerkt und richtig interpretiert?
 - Passt die Benutzerschnittstelle zu den Details der Arbeitsabläufe und Tätigkeiten?
- Kritische Funktionen sollten mit einer realistischen Menge echter Daten hinterlegt sein. Damit können Benutzer ausgewählte Fälle durchspielen und die Benutzerschnittstelle auf ihre Tauglichkeit beurteilen.

Für gutes Aussehen sorgen

Moderne Produkte und Benutzeroberflächen überzeugen auch durch ein ästhetisches Design. Designer arbeiten mit Prototypen, um sich verschiedene Varianten vor Augen zu führen und die Details der Gestaltung auszuarbeiten. Ein zentraler Punkt dabei ist, Funktionalität und Ästhetik zu verbinden. User Interface Designer werden sich deshalb zu folgenden Punkten Gedanken machen:

Die Benutzerschnittstelle optimieren

Gerade wenn hohe Anforderungen an Effizienz, Verständlichkeit oder an die Qualität der Arbeitsergebnisse gestellt werden, gewinnen Details der Benutzerschnittstelle große Bedeutung. Schon eine unglücklich formalisierte Bezeichnung kann verhindern, dass Benutzer einen Automaten richtig benutzen können. Eine immer wieder zu bestätigende Warnmeldung wird Vielbenutzer zur Weißglut treiben, und eine umständliche Navigation verlangsamt die Arbeit.

Ein User Interface Designer sollte deshalb kritische Ausschnitte der Benutzerschnittstelle mittels Prototypen umsetzen und mit Benutzern evaluieren (vergleiche Abschn. 4.7). Im Folgenden einige wesentliche Fragestellungen:

- Emotionale Wirkung des Produktes,
- Bildsprache und Symbolik,
- Farben und Kontraste,
- Textgestaltung und Schriften,
- Anordnung und Abstände der Elemente,
- Animationen und flüssige Übergänge,
- Form, Wirkung und Haptik physischer Bedienelemente.

Für solche Beobachtungen ist im Regelfall ein Werkzeug notwendig, das eine grafische Gestaltung der Bedienelemente ermöglicht. Zahlreiche Grafikprogramme bieten die häufigsten Bedienelemente moderner Benutzeroberflächen als vordefinierte Schablonen an und erlauben gleichzeitig, schnell und detailliert zu gestalten.

Das User Interface spezifizieren

Mit einem User-Interface-Prototyp können viele Aspekte der Benutzerschnittstelle auf anschauliche Weise festgehalten werden. Gerade in agilen Entwicklungsteams, die eine leichtgewichtige Dokumentation bevorzugen, kann das Projektteam viele Ergebnisse einer Diskussion mit UI-Skizzen für die spätere Entwicklung festhalten. Im Rahmen der Spezifikation kann ein Prototyp für folgende Zwecke eingesetzt werden:

- Illustration des Funktionsumfangs,
- Verdeutlichung der Funktionsweise,
- Spezifizieren der User-Interface-Elemente,
- Aufzeigen der Navigation und Interaktion,
- Visualisierung der geplanten Lieferung,
- Abschätzung des Realisierungsaufwands durch die Entwickler.

In der Regel sind im Rahmen der User-Interface-Spezifikation die Darstellung des Funktionsumfangs und für ausgewählte Aspekte auch eine realistische Funktionstiefe gefordert. Eine gewisse Interaktivität ist ebenfalls hilfreich, da so verschiedene Zustände der Benutzerschnittstelle visualisiert werden können.

Tab. 4.4 fasst die beschriebenen Verwendungszwecke und die dafür notwendigen Dimensionen für die Erstellung von Prototypen zusammen.

Paper Prototyping

Leider sind wir Erwachsenen der Ansicht, dass Zeichnen nur etwas für Kinder oder Künstler ist, jedoch keine seriöse und ernsthafte Tätigkeit. In diesem Abschnitt möchten wir in aller Deutlichkeit darstellen: Zeichnen ist eine Notwendigkeit! Der Entwurf eines ersten Prototyps mit Papier und Bleistift hat verschiedene Stärken:

- Praktisch alle Personen können damit umgehen.
- Einfache Skizzen sind schnell erstellt und angepasst.

Tab. 4.4 UX-Prototypen können je nach Verwendungszweck unterschiedliche Dimensionen des geplanten Produkts und der Benutzerschnittstelle darstellen

Zweck	Dimensionen
Produktidee entwickeln	Funktionsumfang; Kernfunktionen eriketbar machen Geringe Funktionstiefe Technische Reife und Darstellungstreue sind eher kontraproduktiv
Anforderungen schärfen	Funktionsumfang mit realistischen Daten darstellen
Benutzerschnittstelle konzipieren	Dimensionen Ausgewählte Funktionen im Detail
Benutzerschnittstelle optimieren	Teilweise interaktiv Hohe Darstellungstreue Interaktiv für ausgewählte Funktionen Oft real Daten notwendig Oft hohe technische Reife notwendig
Für gutes Aussehen sorgen	Mittlere Darstellungstreue Ausgewählte Funktionen im Detail
User Interface spezifizieren	Funktionsumfang und -tiefe sind mittel bis hoch Mittlere Interaktivität Hohe Darstellungstreue

- Es wird weniger Zeit für Details aufgewendet als mit Grafikprogrammen.
- Es sind keine technischen Hilfsmittel notwendig.
- Auch nicht standardisierte Bedienelemente sind schnell skizziert.
- Mehrere Personen können gemeinsam arbeiten.
- Ein Papier-Prototyp zu zerknüllen und wegzuwerfen, fällt leicht.

Aus diesen Gründen arbeitet beispielweise ein Projektteam bei der Bedürfnisanalyse effizienter mit Papier und Bleistift als mit elektronischen Werkzeugen. Papier-Prototypen sind auch gut geeignet, um im Rahmen von Interviews und Workshops aufkommende Ideen zu visualisieren.

Ein weiterer wichtiger Punkt ist, dass Mock-ups auf Papier und elektronische Prototypen nicht die gleiche Wirkung haben. Papier-Prototypen signalisieren durch ihre Skizzhaftigkeit, dass noch viel offen ist und auch über Grundsätzliches diskutiert werden kann. Entsprechend lässt sich gezielter über Abläufe und den konzeptionellen Aufbau diskutieren.

tieren. Bei einem endgültig ausschöpfenden Prototyp gehen hinzu gezeichnete Personen eher davon aus, dass das Grobkonzept bereits feststeht und nur noch an den Details gefeilt werden soll. Eine umfassende Übersicht zum Thema bietet das Buch *Paper Prototyping* [Snyder 03].

Iteratives Vorgehen

UX Prototyping ist ein iterativer Prozess. Auf das Wesentliche reduziert stellt dies Abb. 4.10 dar. Das Projektteam erstellt aufgrund der Anforderungen einen ersten Prototyp des geplanten Produkts oder Systems. Dieser hilft bei der weiteren Optimierung des Produkts und Präzisierung der Anforderungen, beispielsweise im Rahmen von Usability-Tests, Walkthroughs (mehr dazu in Abschn. 4.7), Workshops oder Reviews. Im Folgenden eine vereinfachte Schritt-für-Schritt-Anleitung aus der Praxis:

- Schritt 1: Die genaue Fragestellung für die anstehende Iteration festlegen, z. B. prüfen einer Produktidee, testen einer Design-Hypothese, vertriefen eines schwierigen Aspekts oder analysieren eines Produktziklos.
- Schritt 2: Eine konkrete Anwendungssituation definieren, für welche das Produkt eine Lösung bietet und die der Beantwortung der Fragestellung dient.
- Schritt 3: Geeignete Benutzertypen festlegen und deren Ziele beschreiben (z. B. in Form von Personas): ihre Beweggründe, ihre Werte und ihre Träume.

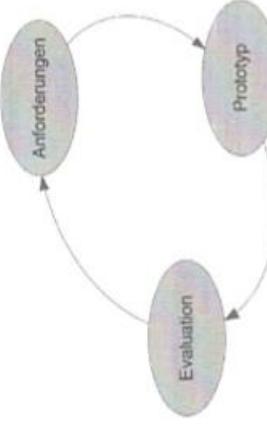


Abb. 4.10 Prototypen dienen der Evaluation von Anforderungen in einem iterativen Vorgehen

Tab. 4.5 Übersicht verschiedener Arten von Prototyping Tools

Tool	Verwendung
Papier & Bleistift, Whiteboard, Folien	Diese Mittel eignen sich besonders in Workshops und Interviews, sowie für explorative Skizzen
Office-Anwendungen <i>Microsoft PowerPoint, Apple Keynote</i>	Mit wenig Aufwand können damit erste Interaktionen erstellt werden
Bildbearbeitungsprogramme <i>Adobe Photoshop</i>	Volle Kontrolle über die grafische Gestaltung
Grafikprogramme <i>OmniGraffle, Microsoft Visio, Adobe Fireworks</i>	Bieten vordefinierte Schablonen der gängigen Bedienelemente. Damit lassen sich schnell gut und echt ausschende Mock-ups erstellen
UI-Prototyping-Werkzeuge <i>Aware RP, Balsamiq Mockups, Just-in-mind Prototyper</i> und viele mehr	Speziell für Prototyping entwickelt, unterstützen diese Werkzeuge den Designprozess von der interaktiven Skizze bis zur Umsetzung auf der Zielpлатform
Multimediarwerzeuge <i>Adobe Director, Adobe Flash</i>	Insbesondere für interaktive Prototypen mit hohen Design-Anforderungen oder Animationen
Programmierwerkzeuge, HTML-Editorn, Entwicklungswerzeuge: <i>Microsoft Expression Blend, Adobe Flex</i>	Interaktive Prototypen mit größeren Datenmengen und komplexeren Verhalten

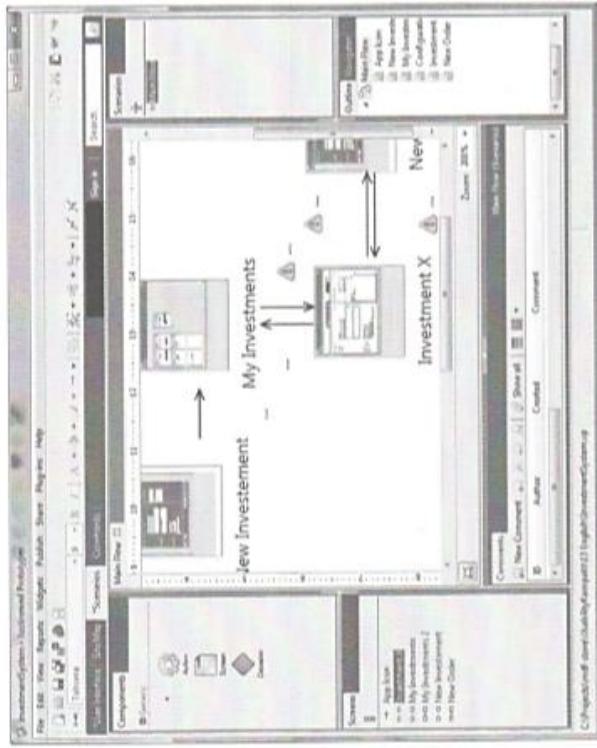


Abb. 4.11 Mit einem Prototyping Tool lässt sich ein einfacher interaktiver Prototyp auch ohne Programmierkenntnisse zusammenstellen

- Um zielführende Lösungsansätze zu identifizieren und Feedback der Benutzer zu erhalten, ist es hilfreich, wenn das Team in den ersten Iterationen verschiedene Lösungsvarianten als UX-Prototypen umsetzt und ein gemeinsames Verständnis im Team bezüglich UI-Konzept, Navigation, Informationsarchitektur, Eingabemodalitäten und Layout aufbaut.

Darauf sollten Sie achten

- UX Prototyping ist eine iterative Tätigkeit. Investieren Sie nicht zu viel Zeit in die Perfektionierung, bevor Sie Feedback einholen.
- Es lohnt sich, ein Werkzeug zu wählen, mit dem ohne großen Aufwand verschiedene Varianten ausgearbeitet und Änderungen umgesetzt werden können. Häufig wird zu früh auf Programmierwerkzeuge für die Erstellung von Prototypen gesetzt.
- Bei Geräten sollten Prototypen die Software in Kombination mit der Hardware zeigen. Anzahl, Größe und Anordnung von Bedienelementen beeinflussen das Bedienkonzept maßgeblich.
- Effizientes Prototyping erfordert die Kombination verschiedener Kompetenzen (z. B. Domänenwissen, Gestaltungskompetenz und Entwicklungskompetenz). Je besser es einem Team gelingt, diese Fähigkeiten zu kombinieren, desto erfolgreicher wird die Lösung sein.
- Erst die Anwendung im realen Umfeld zeigt, wie praxistauglich ein Produkt wirklich ist. Mit UX Prototyping kann diese Realität teilweise erzeugt werden, um Feedback zu erhalten. Wägen Sie ab: Welche Fragestellungen lassen sich mittels Prototyping beantworten und für welche benötigen Sie eine erste Produktivversion? Je komplexer und kritischer die Anwendung, desto sollten mit Prototypen vor der Implementierung weitere Erkenntnisse gesammelt werden. Je komplexer und neuer das Umfeld, umso wichtiger ist das Feedback der Benutzer aus dem realen Einsatz.

Anwendung im agilen Umfeld

- Im agilen Umfeld dienen Prototypen als leichtgewichtige Spezifikation, zur Klärung von Anforderungen vor der Iteration, zur Konzipierung und Optimierung des User Interfaces und um eine ästhetische Gestaltung zu erreichen. Wireframes dienen zur Ergänzung der User Stories und um Details der Benutzerschnittstelle für die Entwicklung festzuhalten. Dieser Schritt findet typischerweise vor jeder Iteration statt.

4.5 In die Entwicklung tragen: Use Cases und User Stories

Anwendungsfälle (*Use Cases*) stammen aus dem Software Engineering und sind ein verbreitetes Instrument zur Spezifikation technischer Systeme. Im agilen Umfeld (vergleiche Abschn. 3.1 „Hintergrund: agile Software-Entwicklung“) haben sich **User Stories** anstelle der Use-Case-Modellierung etabliert.

In der Usability- und UX-Literatur findet sich zum Thema Use Cases und User Stories eher wenig. Dies mag überraschen, da mittels dieser Techniken das Verhalten eines Systems aus Benutzersicht dargestellt wird. Die Abläufe, die ein Benutzer später mit dem System erlebt, werden zu einem großen Teil von den spezifizierten Anwendungsfällen oder User Stories bestimmt. Deren Entwurf hat für die User Experience eines Systems daher eine zentrale Bedeutung. Auch wenn Use Cases und User Stories keine nutzerorientierten Methoden im engeren Sinn darstellen, haben wir uns aus oben genannten Gründen entschieden, sie in unsere Sammlung der wichtigsten Methoden aufzunehmen.

Use Cases

Use Cases beschreiben die (geplanten) Funktionen eines Systems und damit dessen Verhalten gegenüber der Außenwelt. Eine große Stärke von Use Cases ist, dass diese die Funktionsvielfalt aus Benutzersicht in zusammengehörige Einheiten aufbrechen und Schritt für Schritt näher definieren. Ein weiterer Vorteil von Use Cases ist die Beschreibung in natürlichsprachlicher Form, die im Verlauf der Entwicklung für alle Beteiligten verständlich bleibt.

Um eine bestimmte Funktion darzustellen, verwendet das Projektteam sogenannte **Akteure (Actors)**, die mit dem System in Interaktion treten. Akteure verkörpern dabei die Rollen von Benutzern oder anderen Systemen. Der Anwendungsfall selbst beschreibt den funktionalen Ablauf mit dem System aus Sicht des Akteurs. Soll beispielsweise eine Anwendung zur Buchbestellung im Internet beschrieben werden, dann wäre der Kunde, der das Buch bestellt, ein Akteur, die Bestellung eines Buches ein Anwendungsfall, die Verfolgung der Bestellung ein zweiter, die Be-

wertung eines Buches ein dritter usw. Dabei sollte ein Anwendungsfall immer eine aus Sicht des Akteurs abschließende Handlung umfassen. Die Akteure und Anwendungsfälle eines Systems können in einem *Use-Case-Modell* repräsentiert werden. Das Projektteam modelliert damit im Rahmen der Anforderungsanalyse die Funktionsweise des Systems. Das System selbst wird dabei zunächst von außen als Blackbox betrachtet, d. h. es wird noch nicht beschrieben, wie das Verhalten zu stande kommt. Die einzelnen Interaktionsschritte jedes Anwendungsfalls werden in einem zweiten Schritt beschrieben und dienen zur Spezifikation für die Entwicklung. Die Anwendungsfälle verkörpern somit das funktionale Verhalten eines Systems. Das Use-Case-Modell lässt sich in Form eines *Use-Case-Diagramms* auch grafisch darstellen. Ein solches Use-Case-Diagramm dient als Übersicht über die Funktionen des Systems und seiner Schnittstellen zur Außenwelt. Abb. 4.12 zeigt ein Beispiel eines Use-Case-Diagramms in der UML-Notation.



Abb. 4.13 Die Use-Case-Spezifikation zeigt, wie ein Akteur mit dem System interagiert

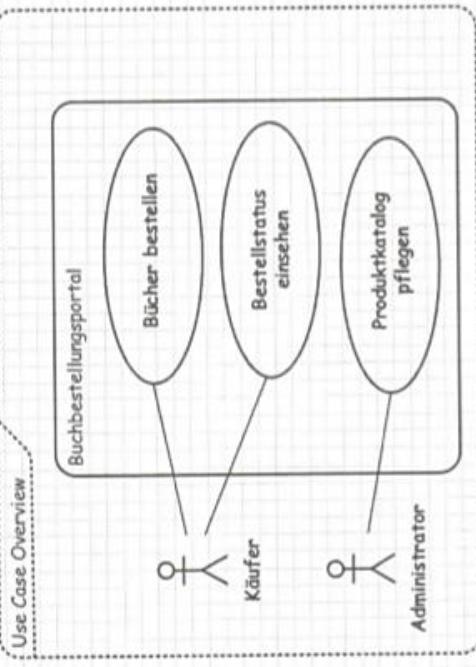


Abb. 4.12 Ein Use-Case-Diagramm eignet sich als Funktionsübersicht einer geplanten Anwendung

In der *Use-Case-Spezifikation* wird ein Anwendungsfall detailliert beschrieben (siehe Abb. 4.13). Die Schritte in der Interaktion zwischen Akteur und System werden aufgeführt. Neben dem „Schönwetterfall“ berücksichtigt das Projektteam hier auch alternative Abläufe und Fehlerfälle. Bei der Ausformulierung eines Anwendungsfalls sollten noch keine technischen Details beschrieben werden. Ein häufiger Fehler ist, dass in Use-Case-Spezifikationen Details der Benutzeroberfläche vorweggenommen werden. Dies ist aus zweierlei Gründen nicht sinnvoll: Der beschriebene Ablauf kann unter Umständen mit einem alternativen Interaktionskonzept besser umgesetzt werden als zunächst angenommen. Weiter erschwert eine zu starke Detailtiefe die Pflege und Anpassung der Anwendungsfälle bei Änderungen. Es ist deshalb weitaus besser, neben der Use-Case-Spezifikation einen User-Interface-Prototyp oder ein Sto-

ryboard zu erstellen, das die Abläufe mit der konkreten Benutzerschnittstelle verdeutlicht. Mittels eines solchen Prototyps kann ein Projektteam die Entwürfe mit Auftraggebern und Benutzern verifizieren und vervollständigen. Ergänzend können auch Ablaufdiagramme für die Darstellung der einzelnen Schritte verwendet werden.

Die Spezifikation muss schließlich von zwei ganz unterschiedlichen Parteien verstanden werden können: dem Auftraggeber und den Entwicklern. Anwendungsfälle sollten deshalb sowohl formal korrekt als auch verständlich sein. Eine praktische Hilfestellung für die Formulierung guter und verständlicher Anwendungsfälle bietet [Cockburn 03].

Hintergrund: Funktionale und nicht-funktionale Anforderungen

Im Anforderungsmanagement wird zwischen funktionalen und nicht-funktionalen Anforderungen unterschieden:

Funktionale Anforderungen betreffen, wie der Name sagt, jene Aspekte, die mit dem Funktionsangebot des geplanten Systems zusammenhängen. Ein Buchbestellsystem kann beispielsweise Funktionen zum Suchen, Bestellen und Bewerten eines Buches anbieten. Die Abläufe, die für das Bestellen eines Buches notwendig sind, können etwa mit Use Cases oder User Stories näher beschrieben werden.

Nicht-funktionale Anforderungen umfassen sämtliche geforderten Qualitätsaspekte und Rahmenbedingungen. Im Beispiel muss das Buchbestellsystem bestimmten Anforderungen bezüglich Verfügbarkeit, Antwortzeiten, Ausfallsicherheit usw. genügen. Nicht-funktionale Anforderungen spielen ebenfalls eine wesentliche Rolle für das Nutzerlebnis eines Systems und haben Auswirkungen auf die verwendete Technologie und Software-Architektur. Auch bezüglich Usability können nicht-funktionalen Aspekte festgelegt werden, etwa im Hinblick auf die geforderte Effizienz für die Benutzer. In unserem Beispiel: Die Suche und Bestellung eines bestimmten Buches soll für 90 % der Benutzer in weniger als fünf Minuten möglich sein.

Die Überlegung, technische Details in Anwendungsfällen noch weitgehend auszuklammern, führte zum Konzept der **Essential Use Cases** [Constantine et al. 99]. Dabei wird versucht, in der Beschreibung von Anwendungsfällen nur die Interaktion mit dem System als solche auszudrücken und von deren technischen Umsetzung konsequent zu trennen. Damit soll vermieden werden, dass zu früh auf eine – eventuell nicht optimale – technische Lösung eingeschwankt wird. Das Beispiel in Tab. 4.6 zeigt einen einfachen Essential Use Case für ein Bücherbestellsystem.

Tab. 4.6 Ein Essential Use Case zeigt den Ablauf der vorgesehenen Benutzerinteraktion ohne Informationen zur technischen Umsetzung

Essential Use Case: ein Buch einer Bestellung zufügen	Absicht des Benutzers	Verantwortlichkeit des Systems
Führt Suche aus		Präsentiert Suchdialog
[Optional] informiert sich über Buch		Präsentiert passende Bücher
Fügt Buch zur Bestellung hinzu		Präsentiert Details zum Buch
		Bestätigt Abschluss

User Stories

User Stories haben sich im agilen Umfeld (vergleiche Abschn. 3.1 „Hintergrund: agile Software-Entwicklung“) anstelle der ausführlichen Use-Case-Spezifikationen etabliert. Die beiden Techniken muten im ersten Augenblick sehr ähnlich an. Wie ein Use Case beschreibt auch eine User Story eine Funktion aus Sicht der Benutzer. Der Titel eines Anwendungsfalls mit dem Akteur „Käufer“ lautet beispielsweise „Bücher bestellen“. Eine User Story im selben Kontext könnte lauten: „Als unangemeldeter Benutzer kann ich ein Buch in den Warenkorb legen, um dieses später zu bestellen“.

Tatsächlich gründen die Unterschiede der beiden Techniken im Verwendungszweck. Use Cases werden eingesetzt, um das Verhalten des künftigen Systems zu vereinbaren und festzuhalten. Basiert auf diesen Anforderungen soll das System entwickelt und getestet werden. Typischerweise stehen formelle Vorlagen für die Dokumentation der Use Cases zur Verfügung.

User Stories dagegen halten lediglich eine Kommunikationsabsicht fest. Wenn das Projekt soweit ist, soll die User Story mit den richtigen Personen detailliert diskutiert und vereinbart werden. Allen Beteiligten soll klar sein, was für welchen Zweck implementiert wird, wie es getestet und abgenommen wird. User Stories können zu Beginn grobgranalartig sein, werden im Verlauf des Projekts jedoch in neue, feinere aufgebro-



Abb. 4.14 Eine User Story steht für eine Kommunikationsabsicht. Vor der Umsetzung wird die geplante Funktion noch im Detail diskutiert und falls notwendig, die User Story aufgeteilt

Bestellstatus einsehen

Als angemeldeter Benutzer kann ich den Status meiner Bestellung einsehen, so dass ich verstehe, wann die Produkte verschickt werden.

Akzeptanzkriterien:

- Bestellungen des angemeldeten Benutzers angezeigt (Datum und Status)
- Mehrere Bestellungen auswählbar
- Button «Details» oberhalb der Liste
- Ausgewählte Bestellungen expandieren in der Liste.
- Details zeigen die Produkte, auch bereits gelieferte Produkte.
- Pro Produkt auch das Produktbild
- ...

chen. Abb. 4.14 zeigt ein Beispiel einer User Story mit bereits vereinbarten Akzeptanzkriterien.

Das Ziel ist, dass mehrere User Stories in einer Iteration vollständig implementiert werden können. User Stories bedingen die direkte und zeitnahe Kommunikation zwischen den Beteiligten. Das Team hält Details entsprechend mit Stichworten, Skizzen der Benutzerschnittstelle, einfachen Szenarien und so weiter fest. Agile Teams verwenden auch Personen bei der Formulierung von User Stories und schlagen so die Brücke zwischen Benutzeranforderungen und Entwicklung. Für weiterführende Informationen zu User Stories siehe auch [Widemann 11] oder [Cohn 04].

Woher kommen die Akteure?

Akteure und Anwendungsfälle werden typischerweise in *Use Case Workshops* mit Auftraggebern, Fachstellen und Benutzern erarbeitet. Die Schwierigkeit dabei ist, dass Akteure keine realen Benutzer, sondern Rollen darstellen, die mit dem System interagieren. Ein häufiger Fehler ist, dass die Akteure so weit verallgemeinert werden, dass der Bezug zur

Realität verloren geht. Dies kann dazu führen, dass wichtige Bedürfnisse vergessen oder Anforderungen unterschiedlicher Benutzergruppen in einem einzigen Akteur vereinigt werden. In der Folge werden Anwendungsfälle konstruiert, die eine Vermischung von Funktionen beinhalten und besser klar getrennt würden; mit dem Resultat, dass die so spezifizierten Abläufe für die tatsächlichen Benutzer in der realen Anwendung später nicht geeignet sind.

Besser ist es, ein Use-Case-Modell aufgrund der Ergebnisse aus der Analyse von Benutzern und Kontext zu erstellen. Contextual Inquiry, Interviews oder vergleichbare Methoden, mit denen die Verantwortlichkeiten und Tätigkeiten der Benutzer aufgenommen werden, sind eine wichtige Voraussetzung dafür.

User Story Mapping

Eine *User Story Map* stellt im agilen Sinne einen Backlog dar, in welchen User Stories priorisiert, weiter verfeinert und mit zusätzlichen Informationen versehen werden können (zum Beispiel geschätzter Aufwand, Stand der Umsetzung usw.). Die Methode wurde von Jeff Patton entwickelt und wird in seinem Buch [Patton 14] beschrieben.

In einer User Story Map werden User Stories – in der Regel in Form von Kärtchen – an eine Wand gehängt und in zwei Dimensionen platziert (siehe Abb. 4.15). Typischerweise wird die Nutzung des Produkts als grober Ablauf in Schritten von links nach rechts dargestellt, von oben nach unten die Priorität der einzelnen User Stories. Die gesamte Story Map repräsentiert dabei die zu entwickelnden Funktionen des Produkts und dient als Basis für die Release-Planung. Die Stärke einer User Story Map liegt darin, dass die einzelnen User Stories auf diese Weise zueinander in Bezug gesetzt werden und die gesamte Anwendung jederzeit sichtbar bleibt.

Ähnlich zu einem Anwendungsszenario (siehe Abschn. 4.2) stellt eine User Story Map die Nutzung eines Systems oder Produkts ebenfalls als zeitlichen Ablauf dar. User Story Mapping schlägt so eine nützliche Brücke zwischen UX und Entwicklung.

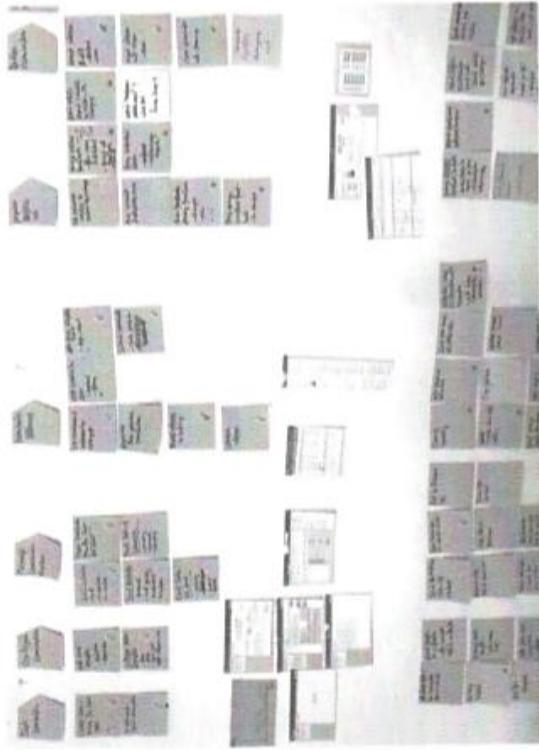


Abb. 4.15 Eine User Story Map zeigt die User Stories in Zusammenhang mit dem Gesamtablauf

Use Cases oder Szenarien?

Akteure und Anwendungsfälle sind konzeptionell verwandt mit Personas und Szenarien (siehe Abschn. 4.2). Beides sind Methoden zur Modellierung und Beschreibung der vorgesehenen Interaktionen mit dem System. Dennoch gibt es einige wesentliche Unterschiede (vergleiche Tab. 4.7):

- Akteure definieren Rollen, die mit dem System interagieren. Personas hingegen charakterisieren prototypische Benutzer und fokussieren auf die Eigenschaften unterschiedlicher Benutzergruppen.
- Anwendungsfälle halten einen bestimmten Teil des Systemverhaltens für die Entwicklung fest. Sie generalisieren über die verschiedenen Möglichkeiten der Nutzung. Szenarien dagegen beschreiben konkrete Beispiele der Systemnutzung und illustrieren die Anwendung im realen Kontext.

Tab. 4.7 Gegenüberstellung von Personas und Szenarien mit den Elementen der Use-Case-Technik (Akteure und Anwendungsfälle)

Methode	Format	Ziel
Akteur	Rollenbeschreibung	Gruppierung zusammengehöriger funktionaler Abläufe
Persona	Prototypischer Benutzer	Charakterisierung unterschiedlicher Benutzergruppen
Anwendungsfall	Beschreibung des Systemverhaltens	Spezifikation des funktionalen Verhaltens für die Entwicklung
Szenario	Konkretes Beispiel der Systemnutzung	Beschreibung der Anwendung im realen Kontext

Realistische Personas und Szenarien, die aufgrund von Benutzeranalysen erstellt wurden, bilden eine ausgezeichnete Grundlage für den Entwurf eines Use-Case-Modells oder die Erstellung einer User Story Map. Im Verlauf des Projekts detailliert und komplettiert das Team das gesamte funktionale Verhalten des Systems.

Denkanstoß

Stellen Sie sich ein System zum Bestellen von Büchern vor. Denken Sie sich den Akteur „Besteller“ und den Use Case „Buch bestellen“. Nun stellen Sie sich Ruth vor. Sie ist Mitarbeiterin einer wissenschaftlichen Bibliothek und ordert am Montag 219 Neuerscheinungen über dieses Bestellsystem. Wie unterscheidet sich ein solches Bestellsystem von jenem, das Lara verwendet, um den neusten Roman ihres Lieblingsautors zu bestellen?

Ein Analyst, der eine Spezifikation für ein neues System schreibt, untersucht verschiedene einzelne Situationen und generalisiert die erhaltenen Informationen zu einem abstrakten Modell. Haben Leser der Spezifikation allerdings nur dieses abstrakte Modell zur Verfügung, können sie die konkreten Einzelfälle und deren Unterschiede nicht herleiten. Denn diese Informationen sind in der Abstraktion nicht mehr enthalten. Szenarien hingegen sind hervorragende Instrumente, um Zusammenhänge oder Abläufe aufzuzeigen (vergleiche Abschn. 4.2 „Hintergrund: Die Macht

des guten Beispiels“). In den meisten Fällen ist es bereits hilfreich, wenige stellvertretende Beispiele – etwa einen typischen Arbeitsablauf eines typischen Benutzers – möglichst realitätsnah in Form eines Szenarios wiederzugeben.

Anwendung im agilen Umfeld

- Es werden nur jene Anforderungen im Detail ausgearbeitet, die in kommenden Iterationen realisiert werden. User Stories unterstützen diesen Prozess. Funktionen, deren Umsetzung erst in späteren Iterationen geplant ist, werden zunächst nur grob aufgenommen und erst dann ausgearbeitet und spezifiziert, wenn sie wirklich entwickelt werden sollen.
- User Stories alleine eignen sich hingegen nicht besonders gut dazu, das Wissen über Anforderungen langfristig aufzubewahren, sofern dies notwendig oder angebracht ist. Die Dokumentation mittels Use Cases, das Festhalten von Entscheidungen, Szenarien mit ergänzenden Kontextinformationen usw. sind dazu besser geeignet.

Checkliste für den Einsatz von Use Cases und User Stories

- Sind die erstellten Akteure in der Realität wiederzufinden? Wurden sie auf Basis der Analyse von Benutzern und Kontext erstellt? Wurden verschiedene Benutzergruppen berücksichtigt?
- Umfassen die Anwendungsfälle oder User Stories abgeschlossene Handlungsalüfe aus Benutzersicht? Entsprachen diese den erarbeiteten Szenarien und tatsächlichen Tätigkeiten der Benutzer?
- Wurde in den Use-Case-Spezifikationen oder User Stories an Alternativabläufe und das Verhalten bei Fehlerfällen gedacht? Auch diese gilt es in einem nutzerorientierten Systemdesign zu berücksichtigen.
- Wurden die Anwendungsfälle sowie alternativen Abläufe mit Angaben zu Dauer und Häufigkeit des Vorkommens ergänzt?

Quelle: Usability und UX kompakt. Richter, Flückiger. S. 45ff.

Springer Verlag, 4. Auflage

<https://www.springer.com/de/book/9783662498279>

- Beinhalten die Beschreibungen keine technischen Details, die besser in einem User-Interface-Konzept, Storyboard oder Prototyp abgebildet würden?
- Ist bei jedem Schritt im Use Case klar, welche Informationen der Benutzer für seine Aufgabe benötigt?

4.7 Auf dem Prüfstand: Usability Testing

Vermutlich haben Sie schon davon gehört, dass Software und Produkte in sogenannten *Usability Labs* geprüft werden können. Vielleicht hatten Sie sogar schon selbst dazu Gelegenheit, einem *Usability-Test* als Zuschauer oder Testperson beizuwohnen. Obwohl man Usability-Labore nun seit gut 25 Jahren auch im deutschen Sprachraum findet, ist deren Bekanntheitsgrad noch immer relativ klein, und die Wenigsten wissen, was sich bei einem solchen Test genau abspielt. Wir wollen deshalb den typischen Ablauf eines formalen Usability-Tests aufzeigen, um danach auch *kurz* auf andere, weniger formale Methoden einzugehen.

Der formale Usability-Test

Zunächst müssen Auftraggeber und Testleiter das Ziel des Usability-Tests klären. Fachleute unterscheiden zwischen *formativer* Evaluation, die eine Verbesserung des geprüften Systems zum Ziel hat, und *sum-*

maiver Evaluation, die ein Produkt im Sinne einer Qualitätskontrolle zusammenfassend prüft.

Als Vorbereitung für eine Usability-Testserie stellen Testleiter und Auftraggeber die Aufgaben zusammen, die von den Testpersonen mit der zu prüfenden Applikation bearbeitet werden sollen. Um ein gewisses Maß an Vergleichbarkeit zu erreichen, sind diese Aufgaben für jede Testperson dieselben. Man spricht deshalb auch von *Standardaufgaben*. Die Qualität der Ergebnisse eines Usability-Tests hängt wesentlich von der Ausarbeitung dieser Aufgaben ab. In Abschn. 2.4 wurde aufgezeigt, dass das Nutzungserlebnis mit einer neuen Lösung davon abhängt, wie gut die Abläufe für die Benutzer unterstützt werden. Die Erarbeitung relevanter und aus Benutzersicht realistischer Aufgaben sollte deshalb mit großer Sorgfalt durchgeführt werden. Wurden bereits Anwendungsszenarien erstellt, dienen diese als Grundlage (siehe Abschn. 4.2). Ein guter Testleiter wird auf die Einhaltung folgender Kriterien für die Standardaufgaben achten, die Sie als Auftraggeber kontrollieren sollten:

- Die Aufgabenstellung ist ein aus Benutzersicht realistisches Szenario und könnte sich tatsächlich so abspielen.
- Es wird ein Ziel der Anwendung aus Benutzersicht formuliert, keine technische Anleitung zur Erfüllung dieses Ziels. Zum Beispiel: „Sie suchen nach einer passenden Farbe für die Frühlingskleider Ihrer kleinen Tochter“ ist besser als „setzen Sie die Filterkriterien auf gelb und rosa“.
- Die Aufgaben stellen für die Testpersonen einen mittleren Schwierigkeitsgrad dar. Sie sollten lösbar, jedoch nicht zu trivial sein.
- Begriffe und Bezeichnungen, die in der Applikation vorkommen, sind zu vermeiden. Zum Beispiel: „Sie überweisen den Betrag von ...“ ist neutraler als „gehen Sie im Menü auf Einzahlungen“.

Neben den Aufgaben für den Test muss auch das zu prüfende System selbst bzw. der Prototyp entlang der beabsichtigten Anwendungsszenarien vorbereitet werden. Dies kann zum Beispiel bedeuten, dass bestimmte Systemzustände oder Ausgaben abgebildet werden müssen, um der Testperson den Eindruck eines möglichst realistischen, lauffähigen Systems zu vermitteln.

Für eine Usability-Testserie sollten Testpersonen eingeladen werden, die möglichst aus der Benutzergruppe der zu prüfenden Applikation stammen, d. h. tatsächlich zu den späteren Benutzern gehören oder gehören können. Während dies bei Consumer-Produkten oder Internet-Anwendungen vielleicht auch die Sekretärin oder die Mitarbeiter aus dem Nachbarbüro sein können, ist es für spezialisierte Anwendungen unerlässlich, die entsprechenden Fachleute zu rekrutieren.

Die notwendige Anzahl Testpersonen hängt im Wesentlichen von den Zielen des Tests ab. In der Regel genügen fünf bis sieben Testpersonen, um die wichtigsten Anwendungsszenarien einer Applikation anhand eines Prototyps zu prüfen und gezielte Verbesserungsmaßnahmen einzuleiten. Sollen dagegen mit großer Sicherheit sämtliche Stolpersteine vor der Einführung einer Anwendung ausgeräumt werden, sind weitere Testserien notwendig (mehr dazu im Abschnitt „Wie viele Testpersonen sind notwendig?“).

In manchen Fällen kann es sinnvoll sein, dass die Testpersonen eine kurze Schulung oder Einführung in die Applikation erhalten, zum Beispiel, wenn es sich um Expertenbenutzer handelt, die nach einer Einarbeitsphase eine hocheffiziente Applikation bedienen sollen. Handelt es sich dagegen um Gelegenheitsbenutzer, etwa für eine Consumer-Anwendung, sollte die Applikation nicht zuerst erklärt werden. Der Testleiter führt jede Testperson in die Ziele und den Ablauf des Usability-Tests ein, und es werden Spielregeln vereinbart:

- Die Testperson darf den Test jederzeit unterbrechen bzw. abbrechen.
- Die Testperson wird meist gebeten, laut zu denken, d. h. ihre Handlungen für die Beobachter zu kommentieren.
- Sollte die Testperson mit einer Aufgabe nicht mehr weiterkommen, kann sie selbstständig zur nächsten Aufgabe weitergehen.
- Die Beobachter melden sich nur, wenn wirklich notwendig. Sie sollten den Testverlauf möglichst nicht beeinflussen. Eine Konversation mit dem Benutzer erfolgt ausschließlich über kontrollierte Bedingungen, zum Beispiel über eine Gegensprechanlage.

Während der eigentlichen Testphase arbeitet die Testperson in einem speziell eingerichteten Testraum mit der zu prüfenden Applikation gemäß

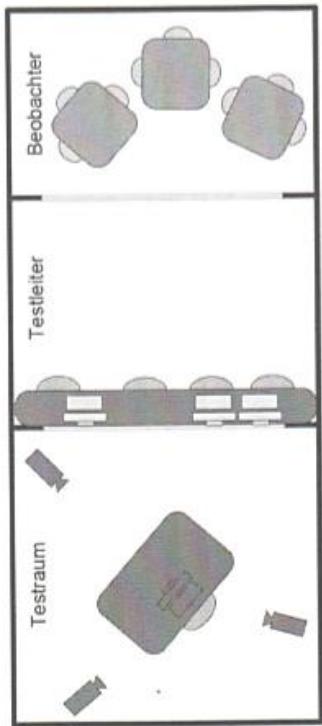


Abb. 4.17 Schematische Darstellung mit den wichtigsten Elementen eines Usability Labs. Die Benutzer bearbeiten im Testraum Aufgaben mit dem neuen Produkt. Die Beobachter protokollieren Schwachstellen. Die Interaktion mit der Benutzerschnittstelle wird zur späteren Analyse auf Video aufgezeichnet

den Aufgaben. Abb. 4.17 zeigt den Aufbau eines typischen Usability Labs.

Als Beobachter sollten Vertreter des Auftraggebers, Entwickler und Usability/UX-Experten teilnehmen. Testleiter und Beobachter verfolgen die Aufgaben aus einem separaten Raum, der häufig durch eine Glasscheibe abgetrennt ist. Der Bildschirm bzw. das zu prüfende Produkt sowie häufig auch das Gesicht der Testperson und die Situation als Ganzes werden auf Video aufgezeichnet. Abb. 4.18 zeigt eine solche Sicht.

Die Beobachter protokollieren unklare und problematische Situationen oder Fehler, die sich in der Anwendung mit der Applikation ergeben. Die Testphase sollte nicht länger als etwa eine Stunde dauern.

In einer Nachbesprechung analysieren die Beobachter die entsprechenden Stellen im Video nochmals mit der Testperson. Dabei werden bereits erste Möglichkeiten für Verbesserungsmaßnahmen besprochen. In jedem Fall sollte die Testperson dazu Gelegenheit haben, das Erlebte in einem kurzen Interview oder frei zu kommentieren. Oft ergeben sich gerade aus eher informellen Situationen wertvolle Hinweise.

Die Probleme in der Anwendung, Feststellungen und Verbesserungsvorschläge werden in einem Testbericht festgehalten. Die Resultate hängen einerseits von der Erfahrung des Testleiters und der Beobachter ab

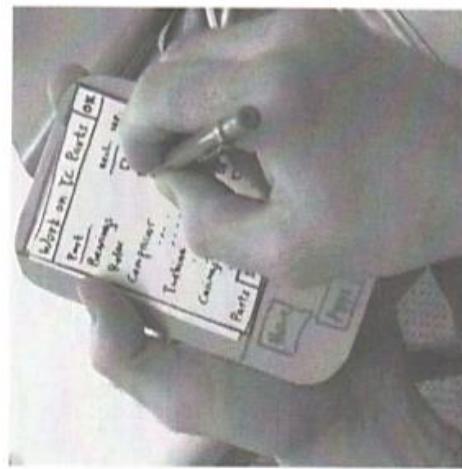


Abb. 4.18 In Usability-Tests können Prototypen eingesetzt werden, um Schwierigkeiten für die Benutzer frühzeitig zu erkennen, versteckte Anforderungen zu entdecken und das User Interface zu verbessern

und andererseits von der Zeit, die für die Ausarbeitung von Verbesserungsvorschlägen zur Verfügung steht. Es ist deshalb lohnenswert, das Ziel der Usability-Tests sowie den Umfang der Vorschläge vorher mit dem Auftraggeber genau zu klären und zu vereinbaren. Ein guter Testbericht sollte folgende formale Kriterien erfüllen:

- Gute wie schlechte Ergebnisse werden aufgeführt.
- Für jede Schwachstelle bzw. Verbesserungsmaßnahme wird ein Schweregrad angegeben.
- Die Schwachstellen sind mit Screenshots illustriert.
- Die Standardaufgaben sowie eine Beschreibung der Testpersonen werden angefügt.
- Es wird klar zwischen beobachteten Problemen und persönlichen Meinungen und Vorschlägen unterschieden.

Das *Common Industry Format (CIF)* für Usability-Testberichte des American National Standards Institute (ANSI) wurde 2006 ein internationaler Standard (ISO/IEC 25062, 2006). Obwohl das Format ursprünglich für summativen Usability-Studien entwickelt wurde, kann es als guter Startpunkt für Ihren eigenen Testbericht dienen.

Mehr Informationen über Usability-Tests finden sich im „Handbook of Usability Testing“ (Rubin et al. 2008).

Stärken und Schwächen des Usability-Tests

Lange Zeit galt der formale Usability-Test als die Königsmethode im Usability Engineering, und tatsächlich hat die Methode auch einige Stärken:

- Unter Laborbedingungen können Schwachstellen der Benutzeroberfläche eindeutig nachgewiesen werden.
- Methodische Gütekriterien wie Objektivität, Reliabilität und Validität können weitgehend eingehalten werden (siehe auch „Hintergrund: Methodische Gütekriterien“).
- Die Beobachtungssituation in einem Usability Lab ist optimal, die Beobachter können sich ein gutes Bild über die Stärken und Schwächen einer neuen Lösung machen.
- Die Beteiligten werden dazu gezwungen, einen Benutzer zuzusehen, ohne in das Geschehen einzutreten. Jeder Usability-Testleiter wird Ihnen versichern, dass die Scheibe zwischen Test- und Beobachtungsraum vor allem dazu da ist, die Beobachter davon abzuhalten, dem Benutzer zu helfen.
- Schwierigkeiten in der Anwendung werden schnell deutlich.
- Die Methode ist für alle Beteiligten gut sichtbar. Die Bedeutung des Einbeziehens der Benutzer wird unmittelbar ersichtlich.

Die Schwächen der Methode sind die folgenden:

- Eine formale Usability-Testserie ist aufwändig. Es muss genügend Zeit für die Ausarbeitung der Aufgaben, die Vorbereitung des zu prüfenden Systems, die Rekrutierung der Testpersonen, die Durchführung der Testreihe in einem Usability Lab und nicht zuletzt die Erarbeitung von Verbesserungsvorschlägen und das Verfassen eines Testberichts eingesetzt werden.
- Die Methode kommt im Allgemeinen relativ spät zum Einsatz. Prototypen müssen so weit vorbereitet sein, dass die Testpersonen selbstständig damit arbeiten können.

- Es besteht die Gefahr, dass der Testbericht mit wertvollen Resultaten nach dem Test in der Schublade des Auftraggebers verschwindet, weil Kenntnisse oder Mittel für Verbesserungsmaßnahmen fehlen. Es ist deshalb notwendig, dass Testleiter, Auftraggeber und Entwickler die Schwachstellen zusammen besprechen und priorisieren. Es kann unter Umständen sogar sinnvoll sein, auf einen ausführlichen Bericht zu verzichten und die Zeit direkt in Lösungsvorschläge zu investieren.

Hintergrund: Methodische Gütekriterien

In der Testtheorie verwendet man eine Reihe von Gütekriterien, anhand derer Aussagen über die Qualität eines Messinstrumenten in einer bestimmten Untersuchung gemacht werden. Die Kenntnis dieser Gütekriterien ist auch für die Anwendung neutralorientierter Methoden von Relevanz, besonders wenn es um die Evaluation von Systemen geht. Als Hauptgütekriterien werden unterschieden:

- **Objektivität:** Die Resultate der Untersuchung sollen unabhängig von den Rahmenbedingungen sein. Ein wichtiger Aspekt ist dabei die Unabhängigkeit von der durchführenden Person, die als *Testleiterunabhängigkeit* bezeichnet wird.
- **Reliabilität:** Bei wiederholter Anwendung unter gleichen Bedingungen soll das Instrument zu gleichen Resultaten kommen.
- **Validität:** Das Instrument soll das messen, was es vorgibt, in unserem Fall also die Usability oder User Experience eines Systems und nichts anderes. Dies kann zum Teil durch die Ausschaltung ungewollter Einflüsse erreicht werden (*interne Validität*). Von Bedeutung ist indessen auch die Generalisierbarkeit der Ergebnisse (*externe Validität*). Sie kann durch eine Erhöhung der Anzahl befragter Personen und eine Überprüfung mit anderen Methoden gesteigert werden.

Alternativen zum formalen Usability-Test

Der **Usability Walkthrough** ist eine Alternative zum formalen Usability-Test. Statt eine Testperson unter kontrollierten Bedingungen in einem separaten Testraum allein arbeiten zu lassen, begleitet der Testleiter den Benutzer und moderiert den Testablauf. Wie beim formalen Test bearbeitet der Benutzer realistische Aufgaben mit dem zu prüfenden System, aber der Moderator hat die Möglichkeit, direkt einzutreten, Fragen zu stellen und bestimmte Abläufe mit dem Benutzer durchzugehen. Dies kann, muss aber nicht in einem Usability Lab durchgeführt werden. Diese Methode eignet sich besonders gut, um früh im Prozess noch unfertige

Prototypen zu evaluieren, ohne dass schon ein lauffähiges System vorhanden wäre. Es ist selbstredend, dass die Testleiterunabhängigkeit bei dieser Spielart nicht mehr gegeben ist und der Moderator sehr genau wissen muss, wie er den Benutzer anleiten kann, ohne ihm zu stark zu beeinflussen.

Als **mobiles Usability Lab** wird eine mobile Einrichtung zur Durchführung von Usability-Tests vor Ort anstatt in einem Labor bezeichnet. Viele Testleiter verwenden heute Programme, die eine Aufzeichnung der Benutzerinteraktion auf dem Testcomputer selbst auf Harddisk sowie das Streamen der Aufnahmen über das Internet – und damit die Beobachtung über beliebig viele andere entfernte PCs – erlauben. Eine häufig zu diesem Zweck eingesetzte Usability Testing Software ist Morae® von Techsmith. Ausgestattet mit einem Laptop und einer Webcam lassen sich so Usability-Testserien ohne die Kosten eines Usability Labs durchführen. Nutzen und Resultate hängen allerdings stark von der Erfahrung des Testleiters ab. Mobile Usability-Tests sind vor allem dann angebracht, wenn bei den Benutzern vor Ort getestet werden soll, zum Beispiel an speziellen Arbeitsplätzen, oder wenn die Umgebung einen maßgeblichen Einfluss hat, beispielsweise bei öffentlichen Terminals wie Ticketautomaten in Bahnhöfen oder Bankautomaten im Freien. Fachleute sprechen dann auch von *Usability-Feldtests*. Auf ähnliche Weise werden **entfernte Usability-Tests** möglich, z. B. mit Beobachtern in anderen Ländern und Zeitzonen (*Remote Usability Testing*).

Um die aufwändige Rekrutierung geeigneter Testpersonen zu vermeiden und um schnell und einfach Feedback von möglichen Benutzern zu erhalten, können Prototypen und Systeme in Entwicklung an frequentierten Orten aufgestellt werden, z. B. in Eingangshallen oder Fußgängerzonen. Interessierte Passanten können auf diese Weise **kurze Usability-Tests** absolvieren. Diese Methode wurde unter dem Namen **Hallway Testing** bekannt.

Wie viele Testpersonen sind notwendig?

Stellen Sie sich vor, Sie wären ein Ladenbesitzer. Gerade haben Sie beobachtet, wie eine ältere Dame im Eingangsbereich über eine hervorstehende Türschwelle stolpert. Kurz darauf geschieht einem jungen Mann dasselbe. Würden Sie die Schwelle ersetzen oder warten, bis weitere Personen straucheln?

Kritische Usability-Probleme können ebenso deutlich sein, wenn sie im Rahmen eines Usability-Tests auftreten. Wenn zwei oder drei Testpersonen an derselben Stelle der Anwendung Mühe bekunden, ist es offensichtlich, dass ein Stolperstein vorliegt, den es auszuräumen gilt. Es ist schlichtweg nicht notwendig, eine vierte oder fünfte Testperson dabei zu beobachten. Mit anderen Worten: Um mittels Usability-Tests Verbesserungen der Benutzerschnittstelle zu erreichen, genügen qualitative Aussagen. Es sind keine quantitativen Untersuchungen mit vielen Benutzern erforderlich (siehe auch Abschn. 4.8 „Hintergrund: quantitative und qualitative Methoden“). Trotzdem stellt sich die Frage, wie viele Testpersonen sinnvoll und notwendig sind, um genügend Sicherheit zu erhalten, dass kritische Stellen auch aufgedeckt werden. Im Folgenden finden Sie einige Anhaltspunkte:

- Iteratives Prototyping, Verbesserung und Anpassung auf die Benutzerbedürfnisse, qualitative Aussagen: 4–6 Benutzer pro Iteration. Systeme mit hohen Anforderungen an die Nutzungsqualität: 7–15 Benutzer. Risikominimierung bei kritischem Systemen: nicht unter 15 Testpersonen.
- Qualitätskontrolle vor Einführung eines Systems, quantitative Aussagen: abhängig von Systemumfang und Anforderungen an die Nutzungsqualität: nicht unter 10 Testpersonen.

Anwendung im agilen Umfeld

- Kurze Iterationen erlauben eine konstante Überprüfung des Nutzungserlebnisses mit schlanken Usability-Tests oder Walkthroughs. Die Evaluation erfolgt kontinuierlich für die anstehenden Fragestellungen. Umfangreichere Testreihen dienen zur gezielten Optimierung

Weitere Möglichkeiten zum Testen der Benutzerfreundlichkeit (Usability)

können sein:

- Benutzung von Onlinefragebögen
- Automatisierte Tests zur Nachverfolgung von Mausklicks und Navigationspfaden
- Blickverlaufsmessungen (Eye-Tracking)
- Auswertung von Logfiles (Web Controlling), die zeigen, wer was wie lange angeschaut hat

besonders herausfordernder Aspekte und werden möglichst frühzeitig in die Entwicklung eingeplant.

- Durch die Aufteilung größerer Produktentwicklungen in mehrere Releases können ausgewählte Anwender bereits frühe Versionen des Produkts einsetzen und Erfahrungen sammeln. Somit können bereits Erkenntnisse aus dem Kontext für die Weiterentwicklung gewonnen werden, beispielsweise mit Beobachtungsmethoden.
- Hat ein Produkt eine gewisse Reife erreicht, streben agile Teams kürzere Release-Zyklen an. In den kürzeren Zyklen können verschiedene Varianten des Produkts mit dem Zielpublikum geprüft werden. Damit lässt sich zum Beispiel feststellen, ob ein neues Feature die gewünschte Wirkung erzielt und das Team somit erfolgreich war.

In Kürze

Methode	Usability Testing
Resultate	Bisher unbekannte Anforderungen aufgedeckt Schwachstellen identifiziert, dokumentiert und priorisiert Verbesserungsmaßnahmen erarbeitet
Vorgehen	Testen mit Benutzern anhand von Standardaufgaben
Aufwand	Formaler Usability-Test im Labor mit 5–7 Benutzern (Vorbereitung, Durchführung und Auswertung): Testleiter 10 PT, Auftraggeber 5–10 PT, Testpersonen total 3–5 PT
Beteiligte	Usability Walkthrough mit 5–7 Benutzern: Testleiter 5 PT, Auftraggeber 3–5 PT, Testpersonen total 2–3 PT
Plnung	Auftraggeber, Entwickler, Benutzer, Testleiter, Beobachter Nach der Erstellung eines ersten Prototyps Am Ende jeder Iteration Vor der Einführung

Sozialwissenschaften, wo statistisch auswertbare Daten vieler Personen gefragt sind, wie etwa bei der Erhebung von Einstellungen, Meinungen und Erfahrungen, oder wenn die Ergebnisse mit einer größeren Gesamtheit verglichen werden sollen, etwa bei psychologischen Studien.

Benutzerbefragungen mittels Fragebögen sind eine wichtige Methode, um Antworten von einer größeren Anzahl Personen zu erhalten. Wenn Sie schon selber einen Fragebogen entworfen haben, zum Beispiel, um Meinungen von Mitarbeitern oder Kunden abzufragen, dann wissen Sie, dass dies gar nicht so einfach ist. Die Zuverlässigkeit der Aussagen hängt von der Qualität des Fragebogens, der Auswahl der befragten Personen und der richtigen Durchführung der Befragung ab.

Quelle: Usability und UX kompakt. Richter, Flückiger. S. 45ff.
Springer Verlag, 4. Auflage
<https://www.springer.com/de/book/9783662498279>

Einsatz von Benutzerbefragungen

Erinnern Sie sich an die in Abschn. 3.4 vorgestellten Aufgabenbereiche eines nutzerorientierten Vorgehens? Benutzerbefragungen dienen zwei Aufgaben: Sie können sowohl zur Analyse von Benutzern und Kontext als auch zur Beurteilung eines Systems (Evaluation) eingesetzt werden. Benutzerbefragungen stellen eine Ergänzung zu den bereits vorgestellten Methoden dar. Der Analyst erreicht damit im Vergleich zu Contextual Inquiry oder Usability-Tests eine größere Anzahl Personen. Fragebögen dienen ihm zur standardisierten Erfassung und Auszählung der Antworten.

Abhängig vom Ziel und der Fragestellung können verschiedene Instrumente eingesetzt werden. Die Palette reicht von einfachen, selbst erstellten Fragebögen bis zum Einsatz methodisch geprüfter Usability- und UX-Standardfragebögen.

Für die Durchführung einer Benutzerbefragung kommen verschiedene Befragungsformen in Betracht: Ein Fragebogen kann beispielsweise schriftlich verschickt, online angeboten oder per Telefonbefragung ausgefüllt werden.

Der Einsatz von Fragebögen erfolgt oft mit dem Ziel, für die gesamte Benutzergruppe repräsentative Aussagen zu erhalten. Da solche Untersuchungen aufzählbare Werte abzielen, sprechen Fachleute auch von *quantitativen*, im Gegensatz zu *qualitativen* Studien. Für sie gilt eine Reihe methodischer Besonderheiten, die beachtet werden müssen. Da dies

4.8 Zahlenmaterial: Fragebögen

Beziehen Sie regelmäßig die neusten Studien bekannter Analysten und Marktforschungsinstitute? Für die Erhebung des zu Grunde liegenden Zahlenmaterials werden häufig Fragebögen benutzt. Die Methodik für die Erstellung und den Einsatz von Fragebögen hat ihre Wurzeln in den

Tab. 4.9 Vergleich zwischen <i>quantitativen</i> und <i>qualitativen</i> Forschungsansätzen	
Quantitative Forschung	Qualitative Forschung
Viele Teilnehmer	Wenige Teilnehmer
Repräsentative Stichprobe	Typische Vertreter
Hypothesen prüfen	Hypothesen bilden
Standardisiert	Flexibel, explorativ
Zahlenmäßige Ausprägungen	Hintergründe, Zusammenhänge
Geschlossene Fragen	Offene Fragen
Statistische Analyse	Inhaltsanalyse
Einfache Auswertung	Aufwändige Auswertung
Beispiele: statistische Befragung, standardisiertes Telefon-Interview	Beispiele: Contextual Inquiry, Stakeholder Interview, Fokusgruppe

mit entsprechendem Aufwand verbunden ist, sollte genau geprüft werden, ob die vorliegende Fragestellung überhaupt mit einer quantitativen Befragung beantwortet werden kann.

Hintergrund: quantitative und qualitative Methoden

Die empirische Forschung unterscheidet zwei grundsätzliche Ansätze zur Erkenntnisgewinnung:

- *Quantitative* Studien haben zum Ziel, zahlenmäßige Ausprägungen möglichst genau zu beschreiben. Dabei wird üblicherweise eine repräsentative Stichprobe befragt und die erhobenen Daten auf die Grundgesamtheit verallgemeinert. In der Regel wird vorher eine Hypothese festgelegt, die anhand der Ergebnisse überprüft werden soll. Quantitative Methoden laufen typischerweise stark standardisiert ab, um möglichst gleiche Voraussetzungen für die Aussagen der Befragten zu schaffen.
- *Qualitative* Erhebungen zielen darauf ab, Hintergründe, Zusammenhänge und Ursachen festzustellen. Dabei wird auf die subjektiven Aussagen der Befragten Wert gelegt. Der Ablauf dieser Methoden ist im Vergleich zu quantitativen Verfahren flexibel, offen und explorativ. Oft werden während der Durchführung neue Hypothesen generiert, um diese in einer nächsten Iteration weiterzuverfolgen. Aus qualitativen Daten kann man keine Mengenangaben ableiten.

Tab. 4.9 stellt die beiden Ansätze gegenüber.

In keinem Fall darf ein Fragebogen als eine einfache Sammlung von Fragen verstanden werden, die man mal eben so verschiekt. Die Durchfüh-

rung einer Befragung setzt voraus, dass sich der Ersteller genau überlegt, welche Fragestellung beantwortet werden soll, wie die Untersuchung durchgeführt wird und wie der Fragebogen dafür aufgebaut sein muss. Methodiker sprechen auch vom *Untersuchungsdesign* und der *Fragebogenkonstruktion*.

Planung einer Benutzerbefragung

Zur Festlegung eines geeigneten Untersuchungsdesigns gehören einige zentrale Überlegungen:

- Welche Fragen oder Hypothesen sollen beantwortet werden? Werden reine Fakten erhoben, soll ein System beurteilt oder ein Vergleich durchgeführt werden?
- Wie ist der zeitliche Ablauf der Untersuchung? Wird nur einmal erhoben, werden die Aussagen verschiedener Gruppen verglichen oder werden die gleichen Benutzer in Abständen mehrmals befragt?
- Wie erfolgt die Auswahl der zu befragenden Benutzer: als zufällige Stichprobe oder nach bestimmten Kriterien?
- Wie viele Personen müssen befragt werden, um statistisch genügend gesicherte Aussagen machen zu können?
- Mit welchen Instrumenten wird die Untersuchung durchgeführt? Wird ein bestehender Fragebogen eingesetzt oder ein eigener erstellt?

Fragebogenkonstruktion

Konstruktion und Einsatz eines Fragebogens müssen methodisch korrekt erfolgen, um aussagekräftige Ergebnisse zu erhalten. Dies gilt auch für einfache Umfragen mit wenigen Fragen. Bevor die Erstellung eines eigenen Fragebogens ins Auge gefasst wird, sollte deshalb abgeklärt werden, ob nicht Standardfragebögen für die Fragestellung verfügbar sind.

Für die Konstruktion eigener Fragebögen zu statistisch zuverlässigen Aussagen lohnt sich in jedem Fall das Hinzuziehen eines Fragebogen-Experten.

Die folgenden methodischen Aspekte sind sowohl für die Erstellung eines eigenen Fragebogens als auch bei der Auswahl eines Standardfragebogens relevant:

- Soll mit offenen oder geschlossenen Fragen gearbeitet werden? Bei offenen Fragen können die Benutzer freie Antworten formulieren, bei geschlossenen wird aus vorgegebenen Antworten ausgewählt. Offene Fragen erlauben es, eine Frage in der Breite auszuleuchten; sie sind allerdings in der Auswertung aufwändiger als geschlossene Fragen. Offene Fragen sind deshalb für qualitative Untersuchungen besser geeignet, während geschlossene Fragen eher für quantitative Studien verwendet werden.
 - Kommen Skalen (z. B. Werte von 1–7) zum Einsatz? Was bedeuten die Skalen (z. B. einverstanden – nicht einverstanden, Note 1–6)? Dies ist von Bedeutung im Hinblick auf die Auswertung und Interpretation der erhobenen Werte.
 - Wie erfolgt die Instruktion zum Ausfüllen des Fragebogens?
 - Sind alle Fragen für die Zielgruppe verständlich? Ein Fragebogen sollte immer an einer Versuchsstichprobe getestet werden.
 - Wie lange dauert das Ausfüllen des Fragebogens? Die Abbruchquote steigt, und die Antwortqualität sinkt mit der Länge eines Fragebogens. Diese Effekte sind bei Online-Befragungen noch ausgeprägter als bei schriftlichen Fragebögen.
- Für die methodischen Details zur Durchführung von Fragebogen-Untersuchungen sei an dieser Stelle auf die einschlägige Fachliteratur verwiesen, z. B. [Bortz et al. 15].

Analyse von Benutzern und Kontext

Fragebögen können zur Analyse der Benutzer und des Kontexts der Anwendung eingesetzt werden und damit zur Klärung der Anforderungen an eine neue Lösung beitragen. Da es in der Anforderungsanalyse meist darum geht, ein neues Gebiet in seiner Breite auszuloten, Details der täglichen Aufgaben zu erfassen oder Ursachen und Zusammenhänge zu erkunden, ist dies zunächst einmal die Domäne qualitativer Methoden.

Ergänzend zur Durchführung von Contextual Inquiry, Interviews oder Beobachtungen kann es indessen erwünscht sein, bestimmte Aspekte mit einem größeren Benutzerkreis abzuklären oder Aussagen gezielt zu erhärten. Eine sinnvolle Anwendung wäre beispielsweise, die Benutzer zunächst mit offenen Fragen nach Vorlieben und Problemen mit bestehenden Systemen zu fragen und diese in einem zweiten Schritt mittels einer Beurteilungsskala bewerten zu lassen.

Fragebögen können auch eingesetzt werden, um Fakten über Benutzer und Anwendung zu erheben, die im Hinblick auf die Anforderungen an die neue Lösung relevant sind:

- Alter, Geschlecht, Aushildung und Erfahrung der Benutzer,
- Rollen, Aufgaben, Tätigkeiten,
- Häufigkeit, zeitliche und örtliche Verteilung der Anwendung,
- Vorhandene technische Ausstattung, z. B. Betriebssystem, Bildschirmgröße und -auflösung, Browersversion, vorhandene Applikationen usw.

In Marktbefragungen für neue Produkte wird manchmal auf diese Weise versucht, Auskunft von den zukünftigen Benutzern über die erwünschten Funktionen oder die Notwendigkeit bestimmter Features zu erhalten. Die Möglichkeiten solcher Befragungen sind allerdings begrenzt. Können sich die befragten Personen beispielsweise nicht ganz genau vorstellen, worum es bei der neuen Lösung geht, sind die Antworten irreführend oder wertlos. Auch die Schwierigkeiten, die sich aus den gewünschten Funktionen ergeben, können die Benutzer kaum abschätzen. Sie entstehen erst in der Anwendungssituation.

Beurteilung von Usability und UX: Standardfragebögen

Standardfragebögen sind vorgefertigte und geprüfte Instrumente zur Usability- und UX-Beurteilung für den Einsatz in Produktentwicklungen oder Studien. Sie können mit relativ wenig Aufwand zur Evaluation lauffähiger Prototypen, zur Identifikation von Schwachstellen oder als

Unterstützt die Software die Erfüllung Ihrer Arbeitsaufgaben, ohne Sie als Benutzer unnötig zu belasten?						
						-
Die Software	-	-	-	+/-	+	++
ist kompliziert zu bedienen	●	●	●	●	●	●
bietet nicht alle Funktionen, um die anfallenden Aufgaben effizient zu bewältigen	●	●	●	●	●	●
bietet alle Funktionen, um die anfallenden Aufgaben effizient zu bewältigen	●	●	●	●	●	●

Abb. 4.19 Der ISONORM 9241/110 ist ein Standardfragebogen zur Beurteilung der Usability einer Anwendung durch deren Benutzer. Der Fragebogen basiert auf den sieben Dialogkriterien der ISO Norm 9241-110. Die Abbildung zeigt einen Ausschnitt des Kriteriums „Aufgabenangemessenheit“

Qualitätskontrolle bei produktiven Systemen und interaktiven Produkten eingesetzt werden.

Die Beurteilung durch die Benutzer erfolgt dabei nach bestimmten Kriterien. Verbreitete Standardfragebögen zur Beurteilung der Usability von Software sind der IsoMetrics [Willumeit et al. 96], der 75 Fragen (Items) umfasst, und der ISONORM 9241/110 [Prümper et al. 97] mit insgesamt 35 Fragen. Beide Fragebögen verwenden die Dialogkriterien der ISO-Norm 9241-110 (siehe auch Abschn. 4.6).

Der ISONORM 9241/110 wurde auch im Einsatz als Online-Version validiert [Richter 99] und stellt damit ein einfaches Mittel zur Beurteilung von Internet-Anwendungen dar. Abb. 4.19 zeigt einen Ausschnitt dieser Online-Version.

Die Bearbeitung des Fragebogens dauert etwa 20 Minuten. Die Auswertung erfolgt im Sinne einer Gesamtbewertung der Usability über die Durchschnittswerte aller Fragen oder auf Ebene der einzelnen ISO-Kriterien. So lassen sich beispielsweise erste Hinweise auf Schwachpunkte finden. Tab. 4.10 zeigt ein Beispiel einer Auswertung mit Mittelwert \bar{x} und Standardabweichung s (Werte 1–7).

Auch zur Beurteilung der User Experience wurden standardisierte Fragebögen entwickelt, wie z. B. der *User Experience Questionnaire* (UEQ) [Lugwitz et al. 08]. Der UEQ erfasst die sechs Dimensionen Attraktivität, Durchschaubarkeit, Verlässlichkeit, Effizienz, Steuerbarkeit

und Originalität. Der *AttrakDiff* [Hassenzahl et al. 03] ist ein weiterer häufig eingesetzter Fragebogen zur Beurteilung des Nutzungserlebnisses, das der Benutzer bei der Bedienung interaktiver Produkte erfährt. Der Fragebogen unterscheidet die Aspekte pragmatische Qualität, hedonische Qualität und Attraktivität.

Ein wichtiger Aspekt bei standardisierten Fragebögen ist die Vergleichbarkeit der Daten. Dadurch, dass jede Person dieselben Urteils-skalen ausfüllt, können die Ergebnisse verdichtet, statistisch ausgewertet und die Aussagen miteinander verglichen werden. Standardfragebögen sind deshalb ein geeignetes Mittel, um Vergleiche anzustellen, zum Beispiel wenn die Nutzungsqualität zweier verschiedener Prototypen, die Aussagen unterschiedlicher Benutzergruppen oder die Beurteilung eines Systems zu verschiedenen Zeitpunkten untersucht werden soll.

Für die Interpretation der Resultate ist es wichtig zu wissen, dass bei

der Beurteilung mittels Fragebögen folgende generelle *Urteilsfehler* auftreten können:

- Bei der Beurteilung besteht die Gefahr, dass die Beurteiler nicht zwischen den einzelnen Kriterien differenzieren, sondern sich von ihrem Gesamteindruck des Produkts beeinflussen lassen (*Halo-Effekt*).
- Das zu beurteilende Objekt kann systematisch zu niedrig oder zu hoch eingestuft werden (*Milde-Härtefehler*). Die Gründe dafür können vielfältig sein und beispielsweise mit Ablehnungen oder Vorlieben zu tun haben.

Tab. 4.10 Auswertung einer Benutzerbefragung mit dem Fragebogen ISONORM 9241/110

ISO-Kriterium	\bar{x}	s
Aufgabenangemessenheit	4,76	0,88
Selbstbeschreibungsfähigkeit	5,20	0,87
Steuerbarkeit	4,64	1,05
Erwartungskonformität	4,95	0,96
Fehlertoleranz	4,76	0,97
Individualisierbarkeit	3,76	1,22
Lernförderlichkeit	5,48	1,06
ISO-Gesamtbewertung	4,80	0,75

- Es besteht eine gewisse Tendenz, alle Kriterien im mittleren Bereich einer Urteilsskala einzustufen (*zentrale Tendenz*). Ist dieser Effekt ausgeprägt, kann dies ein Hinweis auf eine mangelnde Kenntnis des zu beurteilenden Objekts sein.

Kann man Nutzungsqualität messen?

In der Praxis taucht hin und wieder die Frage auf, ob und wie man die Nutzungsqualität von Systemen oder Produkten messen kann. Damit ließen sich unterschiedliche Produkte miteinander vergleichen oder Qualitätsprüfungen durchführen. Nicht zuletzt wäre eine hohe, mit anerkannten Instrumenten gemessene Nutzungsqualität auch ein gutes Verkaufargument. Nicht selten werden Vergleiche publiziert, die auf Ergebnissen aus Usability-Tests oder Expertenmeinungen basieren. Genügt das? Wie müsste ein solches Instrument aussehen?

Halten wir uns nochmals die Definition aus Abschn. 2.2 vor Augen: Ein Mensch-Computer-System setzt sich zusammen aus dem Benutzer, dem Werkzeug (System), der Aufgabe und dem Umfeld. Eine ganzheitliche Bewertung der Nutzungsqualität ist nur unter Berücksichtigung aller vier Komponenten möglich, sollte also mit tatsächlichen Benutzern in deren Umfeld stattfinden. Weiter muss eine genügend hohe Anzahl Fälle vorhanden sein, um statistisch signifikante Aussagen machen zu können. Ein gutes Messinstrument muss außerdem den testtheoretischen Gütekriterien genügen (siehe auch Abschn. 4.7, „Hintergrund: Methodische Gütekriterien“). Und schließlich muss für die Interpretation genügend Vergleichsmaterial verfügbar sein, um überhaupt eine Aussage über die Höhe der gemessenen Werte machen zu können („das Produkt erreicht den Wert 6,5“).

Qualitative Evaluationsmethoden wie Usability-Tests oder Experten-Reviews liefern wichtige Ergebnisse für die Entwicklung und interessante Vergleiche, was in den meisten Fällen auch genügt. Für eine *Messung* der Nutzungsqualität sind sie hingegen nicht geeignet. Entsprechende quantitative Untersuchungen mit Benutzern in ihrem realen Umfeld wären vorstellbar, sind aber zum einen sehr aufwändig und zum anderen nicht über verschiedene Systeme vergleichbar.

Ein vielversprechender Ansatz ist der Einsatz von Standardfragebögen, die als Evaluationsinstrument die oben genannten Anforderungen weitgehend erfüllen. Die Beurteilung eines Systems findet aus Sicht der Benutzer im Anwendungskontext statt und erfolgt einheitlich über dieselben Kriterien. Die Fragebögen wurden von ihren Autoren bezüglich Gütekriterien überprüft und optimiert. Mit einer Normierung streben die Autoren zudem an, eine Vergleichsbasis für verschiedene Anwendungen zu schaffen. So wäre es zumindest theoretisch möglich, auch unterschiedliche Produkte durch ihre Benutzer beurteilen zu lassen und die so erhaltenen Werte zu vergleichen.

Über den Nutzen eines Vergleichs unterschiedlicher Systeme oder Produkte lässt sich streiten. Eine interessante Anwendung für die Praxis ist indes sicher folgende: Mittels regelmäßiger Benutzerbefragungen lässt sich im Sinne einer Qualitätskontrolle die Nutzungsqualität eines Systems über die Zeit überprüfen. Dies ist beispielsweise bei Internet-Anwendungen mithilfe einer regelmäßig geschalteten Online-Befragung ohne großen Aufwand möglich. Sollten sich die Werte (z. B. nach einem Redesign) negativ verändert, ist Handlungsbedarf angesagt. Einige Anbieter nutzen diese Möglichkeit und überprüfen die Nutzungsqualität ihrer Produkte regelmäßig mittels aufgeschalteter Online-Fragebögen.

Checkliste zum Einsatz von Fragebögen

- Ist für die vorhandene Fragestellung eine quantitative Untersuchung notwendig und geeignet, oder sind vielmehr qualitative Methoden besser (z. B. Contextual Inquiry, Interviews, Usability-Tests mit Prototypen)?
- Dienen Zahlenwerte und statistische Angaben zur Beantwortung der Fragestellung?
- Können die Benutzer die Fragen aus ihrem Wissen oder ihrer Erfahrung beantworten?
- Lässt sich die Fragestellung so in einen Fragebogen verpacken, dass sie verständlich bleibt und in kurzer Zeit beantwortet werden kann? Wie groß ist die Gefahr von Falschantworten?
- Ist eine statistische Auswertung der Antworten möglich, sinnvoll und bezüglich Aufwand vertretbar?

- Muss ein Fragebogen selbst entwickelt werden oder ist ein geeigneter Standardfragebogen verfügbar?
- Wie kann der Fragebogen in das Produkt oder in die Anwendung integriert werden, so dass eine große Anzahl Nutzer ihre Bewertung abgibt und damit die Nutzungsqualität laufend überwacht werden kann?
- Wie kann sichergestellt werden, dass die – unter Umständen unkontrollierte – Auswahl der Teilnehmer keine ungewollten Einflüsse auf das Resultat hat?

In Kürze

Methoden Resultate	Benutzerbefragung mittels Fragebögen Quantitative Daten, statistische Aussagen zur Usability und User Experience eines Systems, als Ergänzung in der Anforderungsanalyse oder zur Qualitäts sicherung
Vorgehen	Einsatz eines Standardfragebogens oder Erstellung eines zugeschnittenen Fragebogens, Durchführung der Befragung und Auswertung der Antworten
Aufwand	Erstellen und Auswerten eines kleineren qualitativen Fragebogens: 5–10 PT Durchführen und Auswerten einer Untersuchung mit Standardfragebogen: 10–20 PT
Beteiligte Planung	Erstellen und Auswerten einer quantitativen Fragebogen-Studie: nicht unter 30 PT Auftraggeber, Fragebogen-Experte Analyse: Im Anschluss an Contextual Inquiry Evaluationsinstrument: Im Rahmen eines Pilottests oder zur ständigen Qualitäts sicherung