

„100 SEITEN SPEZIFIKATION - UND WAS IST DIE KONSEQUENZ FÜR UNS?“

Anforderungen vorstellbar machen.

Die Überprüfung der Richtigkeit und Vollständigkeit von Anforderungen ist eine zentrale Aufgabe im Requirements Engineering. Für Business-Vertreter und Benutzer sind die oft umfangreichen Spezifikationen aber nicht sehr verständlich. Die Konsequenz sind spät entdeckte Fehler und aufwändige Änderungen. Wie aber soll Feedback von den Beteiligten eingeholt werden, wenn diese die Anforderungsdokumentation nicht nachvollziehen können? Dieser Artikel beschreibt einige einfache und in der Praxis bewährte Methoden, wie Anforderungen in konkreter, vorstellbarer Form wiedergegeben werden können, um sie mit Business Stakeholdern und Benutzern frühzeitig zu überprüfen.

Requirements Engineering gehört zu den wichtigsten Disziplinen moderner Software- und Produktentwicklung. Ohne das systematische Erarbeiten, Dokumentieren und Verwalten von Anforderungen der verschiedenen Interessensgruppen sind komplexe Projekte nicht in den Griff zu bekommen.

Ziel des Requirements Prozesses ist es, die Anforderungen und Rahmenbedingungen aufzunehmen, gegeneinander abzuwägen und im Verlauf des Projekts abzustimmen, so dass schließlich spezifizierte Vorgaben für die Entwicklung der neuen Lösung zur Verfügung stehen.

Wesentlich für den Projekterfolg ist eine von allen Interessensgruppen getragene und vereinbarte Zielvorstellung. Eine zentrale Aufgabe im Requirements Engineering ist deshalb die Überprüfung (Validierung) der aufgenommenen Anforderungen mit den Beteiligten:

- Wurden alle relevanten Anforderungen aus Geschäftssicht erfasst?
- Können die Benutzer mit der spezifizierten Funktionalität ihre Aufgaben effektiv und effizient erfüllen?
- Wurden die technischen Rahmenbedingungen für die neue Lösung berücksichtigt?

Das Aufnehmen, Dokumentieren und Validieren von Anforderungen ist ein iterativer Prozess. Dieser soll sicherstellen, dass die dokumentierten Anforderungen den tatsächlichen Bedürfnissen und Notwendigkeiten der verschiedenen Interessensgruppen entsprechen. Auf das Wesentliche reduziert stellt dies **Abbildung 1** dar:

Der Analyst steht dabei vor einer schwierigen Aufgabe. Formal korrekt dokumentierte, ausführliche Requirements sind als eindeutige, präzise Vorgaben für die Entwicklung notwendig. Für die Business-Vertreter und Benutzer dagegen sind die Spezifikationen mittels Use Cases, Ablaufdiagrammen und Modellen nicht sehr verständlich. Oft sind die Beschreibungen dermaßen abstrakt, dass deren tatsächliche Bedeutung für die Praxis kaum vorstellbar ist. Fehlerhafte Anforderungen werden dann erst spät im Entwicklungsprozess aufgedeckt, wenn die implementierte Anwendung erstmals sichtbar ist. Dies führt zu teuren Änderungen. Das Risiko ist zudem hoch, dass die entwickelte Lösung sich schlussendlich nicht in die

Geschäftsprozesse integriert und die Benutzer ihre Aufgaben nicht wie geplant oder gar nicht erfüllen können.

Wie aber soll Feedback von den Beteiligten eingeholt werden, wenn diese die Anforderungsdokumentation nicht nachvollziehen können? Um Anforderungen mit Business Stakeholdern und Benutzern frühzeitig validieren zu können, müssen diese in verständliche Form gebracht werden. Je konkreter und vorstellbarer die Anforderungen wiedergegeben werden, umso fundierter werden die Beteiligten diese prüfen und für ihre tägliche Praxis reflektieren können. Die folgenden Abschnitte beschreiben einige in der Praxis bewährte Methoden, um genau dies zu erreichen.

der autor



Michael Richter

[michael.richter@zuehlke.com]

ist Berater bei der Zühlke Engineering AG und Dozent an der HSR Hochschule für Technik Rapperswil für das Master of Advanced Studies in Human Computer Interaction Design. Er ist Autor des Buches „Usability Engineering kompakt“ [Ric07] und begleitet seit mehr als zehn Jahren Projekte in benutzer-orientierter Entwicklung.

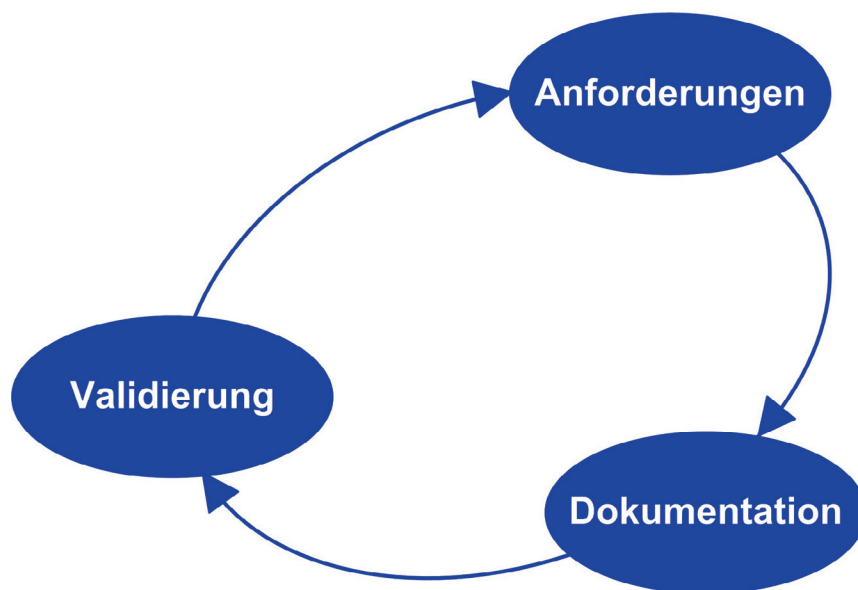


Abbildung 1: Requirements Engineering als iterativer Prozess

Personas repräsentieren Benutzeranforderungen

Personas stellen prototypische Benutzer dar und verkörpern deren Ziele und Verhaltensweisen. **Abbildung 2** zeigt ein (vereinfachtes) Beispiel einer solchen Persona.

Personas werden aufgrund von Informationen über die zukünftigen Benutzer eines Systems erarbeitet. Dazu dienen beispielsweise Ergebnisse aus Workshops mit Benutzern, Beobachtungen am Arbeitsplatz, Fragebögen oder Walkthroughs mit bestehenden Systemen. Im Rahmen eines Projekts entstehen meist mehrere Personas, die die Anforderungen unterschiedlicher Benutzergruppen repräsentieren. Die Methodik wurde vom Interaktionsdesigner Alan Cooper eingeführt und publik gemacht [Coo03].

Requirements Engineering ist ein zentrales Ausbildungsthema des Studiums zum Master of Advanced Studies in Human Computer Interaction Design (www.hcid.ch), einem Weiterbildungsangebot der HSR Hochschule für Technik Rapperswil (Fachhochschule Ostschweiz), der Universität Basel und der Hochschule für Gestaltung und Kunst Basel (Fachhochschule Nordwestschweiz). In dem 3-jährigen berufsbegleitenden Studiengang hat das 1. Jahr das Thema „Benutzerzentriertes Requirements Engineering“, im 2. Jahr steht „Interaction Design“ im Mittelpunkt, den Abschluss bildet die Master-Arbeit im 3. Jahr. Der Autor ist als Dozent aktiv an dieser Ausbildung beteiligt.

Eine Persona sollte 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



Jakob S.

- Sachbearbeiter Schadensabteilung
- 38 Jahre alt
- kaufmännische Ausbildung
- seit 15 Jahren im Versicherungsgeschäft
- seit 7 Jahren in der Schadensabteilung
- arbeitet täglich mit dem heutigen System

*„Ich will ein System, wo ich alles auf einen Blick sehe.“
„Ich brauche kein Programm, das mir Vorschläge macht. Ich weiss genau, was wohin gehört.“*

Abbildung 2: Personas geben Benutzeranforderungen wieder.

Der Analyst entwirft Vorschläge und validiert diese mit den Beteiligten, oder die Personas werden in einem gemeinsamen Workshop erstellt. Die Beschreibungen geben letztendlich die für die Projektziele relevanten Eigenschaften der Benutzer wieder. Personas bilden eine hervorragende Basis für das weitere Vorgehen, indem sie die impliziten Annahmen des Projektteams explizit machen und objektive Kenntnisse über die zukünftigen Benutzer anschaulich darstellen.

Szenarien konkretisieren Prozesse

Anwendungsszenarien oder kurz *Szenarien* 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. Mittels einfacher Sätze 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. **Abbildung 3** zeigt ein Beispiel.

Das Szenario im obigen Beispiel beschreibt in wenigen Sätzen, wie die Aufnahme eines Schadensfalls in Zukunft mit einer neuen Versicherungsapplikation ablaufen soll. Es gibt eine Reihe von zusammenge-

hörigen Anforderungen aus Benutzersicht wieder:

- 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, zum Beispiel zusammen mit Benutzern, erarbeitet werden. Ein großer Vorteil von Szenarien ist ihre leichte Verständlichkeit. Sie können von verschiedenen Stellen wie Auftraggeber, Benutzer und Entwicklung schon zu einem frühen Zeitpunkt überprüft, ergänzt oder korrigiert werden.

Die Durchgängigkeit über den gesamten Entwicklungsprozess macht Szenarien zu einem äußerst effektiven Instrument in der Entwicklung von interaktiven Systemen:

- Die Reflektion am konkreten Beispiel erlaubt es Auftraggebern und Benutzern, Anforderungen in der konkreten Anwendungssituation zu vergegenwärtigen, zu überprüfen und zu ergänzen. Szenarien können als erste Prototypen eines neuen Systems betrachtet werden.

Szenario 1: Aufnehmen eines Schadensfalls

- Es ist 15:00. Bei Jakob S. klingelt das Telefon.
- Auf seinem Bildschirm erscheinen neben der Telefonnummer Name und weitere Angaben des anrufenden Kunden.
- Jakob nimmt den Anruf entgegen und begrüßt den ungeduldigen Kunden, der eine zerbrochene Fensterscheibe melden möchte.
- Da für diesen Kunden verschiedene Versicherungsverträge bestehen, wählt Jakob die entsprechende Police aus der Übersicht aus.
- Danach nimmt Jakob den Schadensfall des Kunden auf.
- ...

Abbildung 3: Szenarien beschreiben einen typischen Ablauf mit der vorgesehenen neuen Lösung.

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 konkrete 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 Benutzer Jakob S. mit dem neuen System arbeiten. Es sollen damit auch nicht nur Schadensmeldungen für kaputte Fensterscheiben erfasst werden. Eine zentrale Anforderung ist im vorliegenden Fall, dass der Benutzer aufgrund der Kundenanfrage (eine zerbrochene 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, um 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.

Buchauszug: *Usability Engineering kompakt - Benutzbare Software gezielt entwickeln*. [Ric07]

- Szenarien illustrieren die Anwendung im realen Kontext und dienen als Ergänzung der Use Cases. Sie vermitteln den Entwicklern ein Verständnis der Abläufe und Zusammenhänge.
- Szenarien dienen dazu, die Abläufe der Benutzerschnittstelle zu beschreiben. Damit kann die Interaktion modelliert und mit Benutzern optimiert werden. Die technischen Anforderungen wiederum können von den Entwicklern überprüft werden.
- Realistische Personas und Szenarien bilden eine gute Grundlage für den Entwurf von Use Cases. Im Verlauf des Projekts detailliert und komplettiert der Analyst die Anwendungsfälle, so dass das gesamte funktionale Verhalten des Systems, inklusive der Anbindung externer Systeme, spezifiziert wird.

Für eine weiterführende Vertiefung der szenariobasierten Entwicklung sei an dieser Stelle auf [Ros02] verwiesen.

User Interface Storyboards visualisieren die Anwendung

User Interface Storyboards visualisieren die Anwendung aus Benutzersicht, indem sie einen ersten Eindruck über die Abläufe der

Benutzerschnittstelle im vorgesehenen Kontext geben. Mit Storyboards hat der Analyst die Möglichkeit, Lösungsideen anschaulich darzustellen, um beispielsweise Feedback von Benutzern und Führungskräften zu erhalten.

Ein Storyboard zeigt mithilfe der Benutzerschnittstelle, wie ein System oder Produkt verwendet wird. Es stellt wichtige

Work on TC Parts		OK
Part	exch.	rep.
Bearings	☐	
Rotor	☐	
Compressor	☐	☒
Turbine	☐	☒
Casing	☐	☐
Parts	Spares	Remarks

Abbildung 4: User Interface Storyboards visualisieren Abläufe.

Aspekte der Anwendung bildlich dar und dient damit der Kommunikation zwischen allen Beteiligten.

Der Analyst setzt Storyboards in Situationen ein, in denen 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 transportiert werden.

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 Ausbildung einen Überblick über das System zu geben.
- Für Projektmarketing bei Auftraggebern, Geschäftsleitung und Benutzern.

Storyboards sind ein ausgezeichnetes Mittel, um die Realität der Anwendung in einen Workshop einzubringen. Die Teilnehmer diskutieren über Annahmen, reflektieren Unterschiede zur heutigen Situation und klären Missverständnisse an realen Beispielen. Es ist immer wieder verblüffend, wie Personen auf abstrakter Ebene aneinander vorbei diskutieren und welche Missverständnisse sich klären,

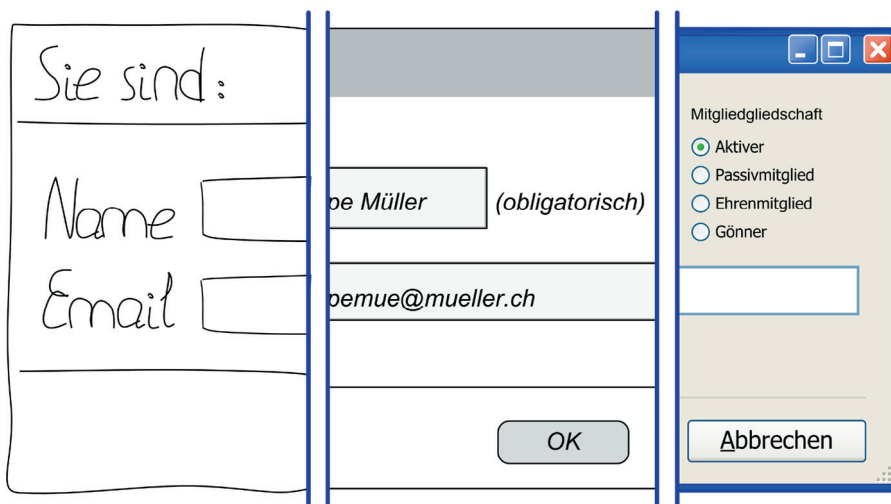


Abbildung 5: Unterschiedliche Ausprägungen eines User Interface Prototypen (Mock-up): Einfache Papierskizze, Drahtmodell (Wireframe) und endgültiges Look&Feel.

sobald jemand eine erste Skizze einer Benutzerschnittstelle auf ein Blatt Papier kritzelt. Die Visualisierung von Anforderungen durch User-Interface-Entwürfe hilft, unterschiedliche Verständnisse zu vermeiden.

Mock-ups decken versteckte Anforderungen auf

Um das Verhalten des Systems weiter zu verfeinern und die Ablaufschritte aus Benutzersicht in Aktion darzustellen, werden mit User Interface Prototypen erste Dialogschritte mit möglichst einfachen Mitteln umgesetzt und von ausgewählten Benutzern und weiteren Stakeholdern geprüft. Man spricht bei solchen Attrappen der Benutzerschnittstelle auch von *Mock-ups*.

Mit Mock-ups können bereits konkrete Fälle durchgespielt und diskutiert werden. Dabei geht es nicht darum, das System zu entwerfen, sondern die Anforderungen mit einem einfachen Hilfsmittel erfahrbar zu machen. Der Analyst deckt dabei neben fachlichen Missverständnissen weitere Bedürfnisse auf:

- Notwendiger Informationsgehalt
- Funktionalität und Abläufe
- Datenaustausch mit anderen Systemen und Applikationen
- Darstellung von Tabellen, Grafiken, Funktionen usw.
- Wichtige Details der Benutzerschnittstelle

Im Rahmen der Spezifikation können Prototypen für folgende Zwecke eingesetzt werden:

- Illustrieren des Funktionsumfangs
- Verdeutlichung der Funktionsweise
- Spezifizieren der User-Interface-Elemente
- Aufzeigen der Navigation und Interaktion
- Visualisieren der geplanten Lieferung
- Abschätzen des Realisierungsaufwands durch die Entwickler

Die erstellten Prototypen vermitteln den Beteiligten erstmals einen Eindruck der laufenden Anwendung und dienen als einfach verständliche und gemeinsame Sprache

zwischen Benutzern, Auftraggebern und Entwicklern. Die in Use Cases textuell oder mit grafischen Modellen formulierten Anforderungen werden dadurch greifbar und vorstellbar. Dank der erhöhten Realitätsnähe kommen bisher unentdeckte Anforderungen zum Vorschein.

Fazit

Anforderungen von der Business- oder Kundenseite, die vom Analysten erarbeitet, analysiert und in Spezifikationen übersetzt werden, stellen die Weichen für die Entwicklung eines neuen Systems.

Um die aufgenommenen Anforderungen zu validieren, müssen Benutzer, Business-Einheiten, Fachstellen und Entwickler ein gemeinsames Verständnis erreichen. Dies ist allein mit abstrakten Dokumentationen nur schwer möglich.

Die Erarbeitung von konkreten, vorstellbaren Ergebnissen ist für die Validierung von Requirements notwendig: Personas, Szenarien, Storyboards und Mock-ups dienen als gemeinsame Sprache der verschiedenen Stellen und erlauben eine effiziente Zusammenarbeit.

Es sollte ein iterativer Prozess aufgesetzt werden, in dem Anforderungen und Spezifikationen visualisiert, mit Benutzern und Geschäftseinheiten überprüft und falls notwendig angepasst werden können. ■

Literatur

[Coo03] Cooper, A. & Reimann, R., *About Face 2.0: The Essentials of Interaction Design*, Wiley, Indianapolis, 2003

[Ros07] Rosson, M. B. & Carroll, J. M., *Usability Engineering: Scenario-Based Development of Human Computer Interaction*, Morgan Kaufmann, San Francisco, 2002

[Ric07] Richter, M. & Flückiger, M., *Usability Engineering kompakt - Benutzbare Software gezielt entwickeln*, Spektrum Akademischer Verlag, Heidelberg, 2007