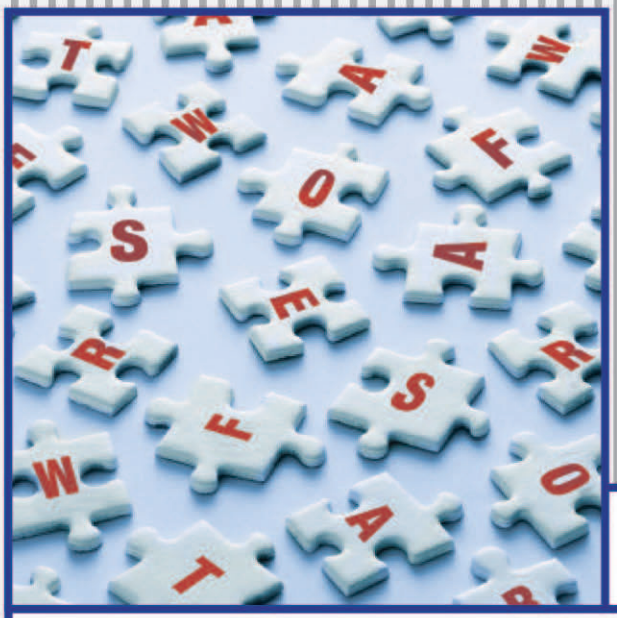
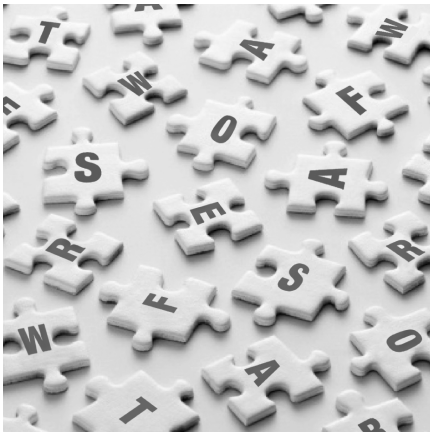
WWW.PEARSON-STUDIUM.DE
COMPANION WEBSITE



Ulrike Hammerschall
Gerd Beneken

Software Requirements

ALWAYS LEARNING
PEARSON



Ulrike Hammerschall
Gerd Beneken

Software Requirements

Higher Education
München • Harlow • Amsterdam • Madrid • Boston
San Francisco • Don Mills • Mexico City • Sydney
a part of Pearson plc worldwide

Beispiel 4.1 **Nachweis der Wartbarkeit**

Das Qualitätsmerkmal der „Wartbarkeit“ kann neben anderen beispielsweise durch das Teilmerkmal „Vollständigkeit der Dokumentation“ beschrieben werden. Eine mögliche Kennzahl zur Messung der Vollständigkeit, ist „der Prozentsatz vollständig dokumentierter Methoden im Source-Code“.

Dieser Wert lässt sich anhand des Source-Codes objektiv mithilfe von Werkzeugen und einer Metrik, die anhand eines Dokumentationsstandards die Vollständigkeit prüft, ermitteln. Als Ergebnis erhält man einen konkreten numerischen Wert, beispielsweise 90%, der besagt, dass 90% aller Methoden in diesem System hinsichtlich des Dokumentationsstandards vollständig dokumentiert sind.

Kennzahlen sind quantifizierbare Werte, die den Zustand des zu messenden Systems beschreiben. Eine Kennzahl drückt nur einen eng begrenzten Aspekt eines Qualitätsmerkmals aus. In gleicher Weise kann auch eine Metrik, die auf dieser Kennzahl basiert, nur eine sehr eingeschränkte Aussage über eine Systemeigenschaft machen. Beispielsweise reicht der Vollständigkeitsgrad der Dokumentation im Source-Code (siehe Kasten) keinesfalls aus, um eine belastbare Aussage über die Wartbarkeit des Systems zu erhalten. Man benötigt eine geeignete Auswahl an Metriken, die das Kriterium unter verschiedenen Gesichtspunkten prüfen.

Um das Ergebnis einer Metrik sinnvoll auswerten zu können, benötigt man zusätzlich ein objektives **Akzeptanzkriterium**, an dem das Ergebnis gemessen werden kann. Das Akzeptanzkriterium gibt vor, welcher Wert der Metrik für dieses konkrete System unter den genannten Bedingungen für eine Abnahme ausreicht.

Im obigen Beispiel zur Vollständigkeit der Dokumentation (siehe Kasten) stellt sich beispielsweise die Frage, was das Ergebnis „90% aller Methoden sind vollständig dokumentiert“ tatsächlich bedeutet. Ist dies ein Zeichen für die Qualität des Systems? Ist dies gut, hervorragend, nicht so gut oder vielleicht sogar auf keinen Fall akzeptabel? Immerhin, so könnte man argumentieren, fehlen 10%. Andererseits sind 90% Dokumentation auf jeden Fall besser, als wenn es nur 50% wären? Ein Akzeptanzkriterium, welches mit der Anforderung vom Auftraggeber festgelegt wurde und beispielsweise besagt, dass mindestens 80% des Source-Codes dokumentiert sein muss, würde hier Klarheit schaffen.

Die Prüfung der Anforderungserfüllung erfolgt bei der Abnahme des Systems am Ende des Projekts. Bei Systemeigenschaften wird mithilfe der vorgegebenen Metriken geprüft, ob das System die zugeordneten Akzeptanzkriterien erfüllt. Dies kann in manchen Fällen durchaus aufwendig sein. Um beispielsweise den Nachweis zu erbringen, dass die Antwortzeit bei paralleler Nutzung von mehr als 500 Anwendern nie höher als 7 Sekunden ist, benötigt man eine entsprechende automatisierte Testumgebung, um die hohe Menge an parallelen Nutzern zu simulieren.

Beispiel 4.2 **Akzeptanzkriterien**

Eine Qualitätsanforderung, die ausdrückt, dass das System „performant“ sein soll, kann beispielsweise durch die Kennzahl „Antwortzeit“ messbar gemacht werden. Als Antwortzeit bezeichnet man die Reaktionszeit eines Systems auf eine Anfrage durch den Anwender. Das Akzeptanzkriterium kann nun wie folgt angegeben werden:

Die Antwortzeit des Systems beträgt weniger als 5 Sekunden bei allen Anfragen.

Nun reagiert ein System nicht immer gleich. Beispielsweise können abhängig von der Auslastung der CPU oder der Menge der Daten, die angefragt werden, starke Schwankungen bei der Antwortzeit auftreten. Entsprechend sollte das Akzeptanzkriterium angepasst werden (hier abhängig vom Lastverhalten):

Die durchschnittliche Antwortzeit bei geringer Auslastung (weniger als 50 parallele Nutzer) beträgt weniger als 3 Sekunden, die durchschnittliche Antwortzeit bei mittlerer Auslastung (zwischen 50 und 500 parallelen Nutzern) beträgt weniger als 5 Sekunden, bei hoher Auslastung (mehr als 500 parallele Nutzer) darf sie 7 Sekunden nicht überschreiten.

Spezifikationsschema für eigenschaftsorientierte Abnahmekriterien

Welche Metriken zur Abnahme einer nichtfunktionalen Anforderung zu prüfen sind, geben die Abnahmekriterien an. ▶ Abbildung 4.3 zeigt ein tabellarisches Schema zur Spezifikation von Abnahmekriterien für Anforderungen, die Eigenschaften beschreiben. Für jede Metrik muss ein eigenes Schema angelegt werden.

Metrik	Explizites Verfahren mit Kennzahl(en) zur Messung der Eigenschaft.
Akzeptanzkriterium	Angabe von Grenzwert(en) oder Grenzbereichen, die bei einer Messung eine Erfüllung der Eigenschaft nachweisen.
Operationalisierung	Optionale Angabe von Testszenarien oder einer Testumgebung, mit der die Erfüllung der Eigenschaft sinnvoll und mit vertretbarem Aufwand geprüft werden kann.

Abbildung 4.3: Spezifikationsschema für Abnahmekriterien von Eigenschaften.

Die Angabe einer Operationalisierung ist hilfreich, um sicherzustellen, dass die Ermittlung und Bewertung des Abnahmekriteriums durchführbar ist. Beispielsweise wäre für eine Anforderung „das System ist gegenüber unbefugten Zugriffen sicher“ eine Metrik „Anzahl der erfolgreichen Einbrüche in den nächsten 10 Jahren“ mit dem Akzeptanzkriterium „0“ ein nach obigen Regeln gültiges Abnahmekriterium Die Frage bleibt jedoch, wie man die Erfüllung des Kriteriums nachweisen möchte ...

Praxistipp: Testszenarien

Für manche Eigenschaftsanforderungen macht es Sinn, als Teil der Operationalisierung manuelle Testszenarien anzugeben, mit denen ihre Erfüllung nachgewiesen werden kann. Ein Szenario beschreibt die Schritte, die der Reihe nach zur Prüfung der Eigenschaft durchzuführen sind. Eine Methode, die szenariobasierte Prüfung von Qualitätsmerkmalen unterstützt, ist SAAM (*Software Architecture Analysis Method*) (Clements, Kazman und Klein).

SAAM wurde ursprünglich zur Bewertung von Software-Architekturen entwickelt. Die Vorgehensweise lässt sich jedoch einfach auf die Bewertung von Qualitätsmerkmalen eines Software-Systems übertragen. Eigenschaften, für die Testszenarien ein geeignetes Mittel zum Nachweis sind, sind beispielsweise die Modifizierbarkeit eines Systems (wie gut lassen sich Änderungen in das System einarbeiten?), die Portierbarkeit (lässt sich das System auf unterschiedlichen Plattformen installieren und anwenden?) oder die Benutzbarkeit (wie lange braucht ein ungeschulter Anwender, um sich in das System einzuarbeiten?).

Beispiele für Metriken

Die große Kunst, gute Abnahmekriterien zu definieren, liegt in der Wahl der richtigen Metriken. Metriken müssen eingehend daraufhin geprüft werden, ob sie für den gewünschten Zweck tatsächlich aussagekräftig genug sind. ▶ Tabelle 4.1 nennt beispielhaft Metriken zu einigen der Qualitätsmerkmale im SQuaRE-Standard (siehe Kapitel 3).

Qualitätsmerkmal	Beispiele für Metriken
Funktionelle Eignung	■ Erfüllungsgrad der Vorgaben zur Genauigkeit zu berechnender Ergebnisse.
Performanz und Effizienz	■ Durchschnittliche/maximale Antwortzeit: Zeitintervall zwischen dem Absenden einer Nachricht und dem Empfang der entsprechenden Antwort. ■ Durchschnittlicher Durchsatz: Menge an Daten/Transaktionen, die von einem System oder Teilsystem pro Zeiteinheit verarbeitet werden kann. ■ Maximal erlaubter Speicherverbrauch.
Kompatibilität	■ Anzahl der Verletzungen von Standardisierungsvorgaben in den angebotenen Schnittstellen (Struktur, Sprache, Datentypen).
Benutzbarkeit	■ Durchschnittlicher Zeitaufwand für Schulungen bzw. zur Verwendung des Systems ohne Schulung. ■ Durchschnittliche Anzahl von Anwenderaktionen zur Durchführung einer vom System bereitgestellten Funktionalität.
Zuverlässigkeit	■ Wahrscheinlichkeit dafür, dass das System zu einer gegebenen Zeit verfügbar ist (Verfügbarkeit = (Gesamtzeit – Gesamtausfallzeit) / Gesamtzeit). ■ Robustheit des Systems gegenüber Systemfehlern. ■ Reparaturzeit bei den wahrscheinlichsten Ausfällen (je kleiner die Reparaturzeit, desto höher die Verfügbarkeit).
Sicherheit	■ Anzahl erfolgreich abgewendeter vordefinierter Missbrauchsszenarien.

Tabelle 4.1: Metriken zur Messung von Qualitätsmerkmalen.

Qualitätsmerkmal	Beispiele für Metriken
Wartbarkeit	■ Vollständigkeit der Dokumentation. Anzahl (nicht) dokumentierter Klassen und Methoden. ■ Zahl der Verletzungen der Programmierrichtlinien. ■ Anzahl identifizierter redundanter Codestellen. ■ Anzahl direkter Methodenaufrufe und transitiver Abhängigkeiten im Code (*Coupling*). ■ Ähnlichkeit der Methoden zweier Klassen auf Grund der von ihnen verwendeten Variablen (*Cohesion*).
Portabilität	■ Anzahl an verwendeten nicht standardisierten Bibliotheken. ■ Anzahl nicht konfigurierbarer Systemeigenschaften im Code.

Tabelle 4.1: Metriken zur Messung von Qualitätsmerkmalen. *(Forts.)*

Metriken werden idealerweise so gewählt, dass ihr Nachweis mithilfe von Werkzeugen erfolgen kann.

Methoden zur Ableitung von Metriken

Um sicherzustellen, dass ein System ein bestimmtes Qualitätsmerkmal erfüllt, benötigt man eine Menge von aussagekräftigen Kennzahlen und Metriken. Wie aber erhält man diese? Es ist ein gewisses Stück Erfahrung nötig, um hier die richtige Entscheidung zu treffen. Wählt man unwichtige Metriken oder vergisst wichtige, birgt dies gewisse Gefahren in sich.

- Fehlende Metriken werden naturgemäß auch nicht geprüft. Ihr Fehlen wiegt Entwickler und Anwender in der falschen Sicherheit, an alles gedacht zu haben.
- Die Wahl unwichtiger Metriken kann zu unnötigen Kosten führen, wenn versucht wird, sie dennoch mit großem Aufwand zu erfüllen.

In manchen Fällen benötigt man zum Nachweis einer Eigenschaft keine explizite Metrik. Es reicht eine einfache binäre Entscheidung, die besagt „Ja, die Eigenschaft ist erfüllt“ oder „Nein, die Eigenschaft ist nicht erfüllt“.

Methode **GQM**

Die GQM (*Goal Question Metric*) (Basili, Caldiera und Rombach) ist eine Methode, mit deren Hilfe Qualitätsmerkmale messbar gemacht werden. Mit ihrer Hilfe werden Ziele (beispielsweise das Erreichen eines Qualitätsmerkmals) über Fragen konkretisiert. Die Fragen helfen dabei, die Ziele möglichst exakt zu definieren. Zu den Fragen werden Metriken abgeleitet, die messbare Antworten zu den Fragen liefern.

Granularität der Abnahmekriterien

Die Granularität der Abnahmekriterien orientiert sich an der Granularität ihrer Anforderungen. Abnahmekriterien als Mittel zum Nachweis der Erfüllung können nur so detailliert sein, wie die Anforderungen, die sie näher spezifizieren. Generell gilt:

- Aus grobgranularen Anforderungen können nur grobgranulare Abnahmekriterien abgeleitet werden. Sie betreffen in der Regel den Nachweis der Kernfunktionalitäten in der Anforderung.
- Feingranulare Anforderungen, insbesondere Use-Case-Spezifikationen (siehe Kapitel 5 „Anforderungsspezifikation“) erlauben dagegen die Ableitung detaillierter Abnahmekriterien, die teilweise unmittelbar als Testfälle für die Abnahme übernommen werden können.

4.1.3 Verwaltungsinformation

Die dokumentierten Anforderungen sind die Treiber für die Durchführung des Entwicklungsprojekts. In allen Projektphasen werden sie nachgelesen, referenziert, diskutiert und geändert. Spezifikationen und Testfälle werden aus ihnen abgeleitet und das System muss sie implementieren. Dies ist nur möglich, wenn Anforderungen selbst als eigenständige Konzepte betrachtet werden, die über den gesamten Projektzeitraum und gegebenenfalls darüber hinaus kontinuierlich verwaltet werden.

Die Verwaltung erfolgt methodisch mithilfe von Metadaten, die Informationen zu Anforderungen liefern. ▶ Tabelle 4.2 listet einige der wichtigsten Metadaten. Die Liste kann jedoch nach Bedarf erweitert werden (siehe (Ebert), (Rupp und Pohl).

Metadatum	Beschreibung
Identifikator	Projektweite eindeutige Identifikationsnummer für diese Anforderung. Das Benennungsschema für Identifikatoren ist frei wählbar, sollte innerhalb eines Projekts jedoch einheitlich gehandhabt werden.
Name	Kurzname der Anforderungen. Der Name ist üblicherweise nach dem Schema <Substantiv><Verb> wie beispielsweise „Reise buchen“ aufgebaut, bei Eigenschaften dagegen mit einem beschreibenden Hauptwort, beispielsweise „Performanz“.
Version	Aktuelle Version der Anforderung. Jede Anforderungsänderung sollte eine neue Versionsnummer nach sich ziehen. Mithilfe der Versionsnummer können zusammengehörige Konfigurationen an Anforderungen gebildet werden.
Priorität	Angabe der mit den Stakeholdern abgestimmten Priorität zur Realisierung der Anforderung. Das Schema ist frei wählbar. Im einfachsten Fall kann ein zweistufiges System gewählt werden mit den Werten „hoch“ und „niedrig“. Das Schema sollte innerhalb eines Projekts einheitlich gehandhabt werden.
Quelle	Quelle, von der die Anforderung erhoben wurde. Dies kann beispielsweise ein Stakeholder, ein Dokument oder das Altsystem sein. Diese Information ist wichtig, falls später im Projekt Rückfragen auftauchen sollten. Auch die Notwendigkeit der Anforderung kann damit jederzeit nachvollzogen werden.
Verantwortlich	Verfasser der Anforderungsdokumentation (i.A. nicht die Quelle). Der Verantwortliche für die Anforderung dient als Ansprechpartner bei Fragen zur Anforderung und ist verantwortlich für die Pflege der dokumentierten Anforderung.

Tabelle 4.2: Metadaten zur Verwaltung von Anforderungen.

Metadatum	Beschreibung
Änderungs-historie	Zeitliche Dokumentation von Änderungen bezogen auf die Anforderung selbst (nicht die Realisierung der Anforderung). Sie beginnt bei mit der Erstellung und Abstimmung der Anforderung. Spätere Änderungen an der Anforderung werden hier ebenfalls mit Änderungsverantwortlichem, Änderungsdatum und einer kurzen Beschreibung der Änderung protokolliert.
Offene Punkte	Punkte, die zur aktuellen Anforderungsversion noch zu klären sind. Es ist Aufgabe des Verantwortlichen die Klärung der offenen Punkte voranzutreiben und das Ergebnis gegebenenfalls in die Anforderungsdokumentation einzuarbeiten.
Quer-beziehungen	Anforderungen, zu denen inhaltliche Beziehungen bestehen. Dies können funktionale oder nichtfunktionale Anforderungen sein.
Stabilität	Angabe zur Wahrscheinlichkeit, dass sich die Anforderung noch ändern wird bzw. die Klärung offener Punkte, die vermutlich Änderungen nach sich ziehen werden.

Tabelle 4.2: Metadaten zur Verwaltung von Anforderungen. *(Forts.)*

Werkzeuge zur Anforderungsverwaltung erlauben die Festlegung von Metadaten und die Zuordnung von konkreten Werten zu den einzelnen Anforderungen. Der große Vorteil von Werkzeugen ist, dass sie automatisierte Operationen auf der Menge der Anforderungen erlauben, wie beispielsweise die Suche einer Anforderung nach einem Identifikator, die Sortierung aller Anforderungen nach Priorität oder die Suche nach allen Anforderungen zu einer spezifischen Quelle.

Steht kein Werkzeug zur Verfügung, empfiehlt sich eine einfache tabellarische Verwaltung der Anforderungen auf Dokumentenbasis. Die Verwendung von Dokumenten erfordert jedoch erhöhten Aufwand zur Pflege der Metadaten.

4.1.4 Lebenszyklus von Anforderungen

Anforderungen liefern für sich genommen keinen Mehrwert. Sie repräsentieren vielmehr den Wunsch eines Stakeholders für eine Funktionalität oder Eigenschaft, die ein System bereitstellen soll. Diese Idee wird Schritt für Schritt im Entwicklungsprozess konkretisiert und schließlich in Software umgesetzt. Eine Anforderung befindet sich daher zu jedem Zeitpunkt im Projekt in einem definierten Zustand bezüglich ihrer Umsetzung. Modelliert wird dies mithilfe eines Lebenszyklusmodells. Das Zustandsmodell in ▶ Abbildung 4.4 modelliert einen typischen Lebenszyklus für Anforderungen. Die Ellipsen entsprechen den Zuständen, in denen sich eine Anforderung befinden kann, die Pfeile markieren die erlaubten Zustandsübergänge.

Der Lebenszyklus einer Anforderung beginnt damit, dass sie von jemandem initial in einem Dokument oder in einem Werkzeug angelegt wird. Die Anforderung ist damit im Zustand *angelegt*. Die Anforderung muss zu Beginn noch nicht vollständig ausgearbeitet sein. Dies sollte jedoch erfolgen, damit die Anforderung in den Zustand *vorgelegt* übergehen kann. Eine verantwortliche Rolle – beispielsweise der Produktmanager – entscheidet nun über ihre Realisierung. Als Ergebnis der Entscheidung kann die Anforderung *akzep-*

tiert, *abgelehnt* oder *zurückgestellt* werden. Abgelehnte Anforderungen werden nicht realisiert, zurückgestellte Anforderungen werden für spätere Versionen vorgemerkt, akzeptierte Anforderungen werden im Projektplan für das aktuelle Projekt eingeplant.

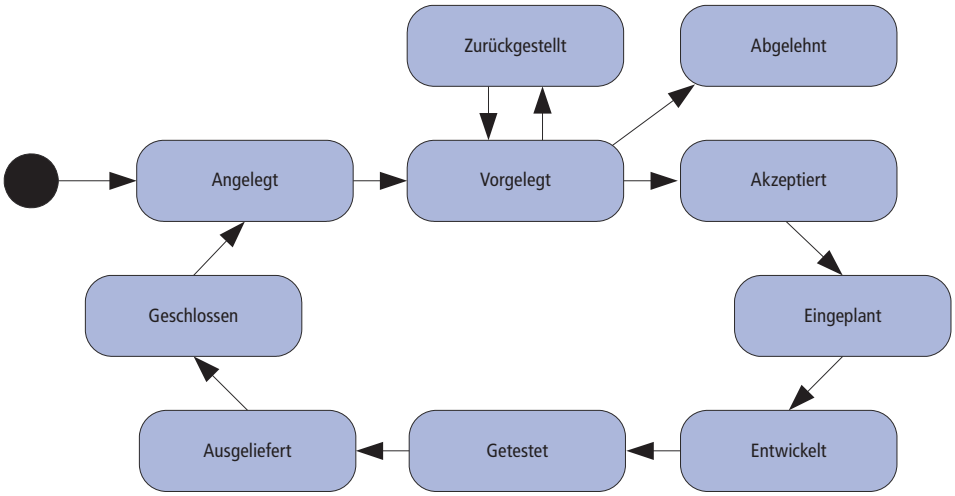


Abbildung 4.4: Typisches Lebenszyklusmodell für Anforderungen.

Eine akzeptierte Anforderung wird entwickelt, getestet und schließlich als Teil des Systems ausgeliefert. Die Anforderung geht damit in den finalen Zustand *abgeschlossen*. Treten bei der Nutzung der Funktionalität Änderungswünsche auf, kann die zugehörige Anforderung wieder geöffnet werden. Der Lebenszyklus beginnt von vorne. Der Lebenszyklus einer Anforderung kann je nach Organisation und Systemtyp unterschiedlich gestaltet sein. Jede Organisation sollte ihr individuelles Lebenszyklusmodell definieren.

Die Verwaltung des Lebenszyklus erfolgt über ein Metadatum *Status*, das den aktuellen Zustand einer Anforderung im Lebenszyklus nachbildet. Die Menge der gültigen Werte für den Status ergeben sich aus den Zuständen des Lebenszyklusmodells. Die Verwaltung des Lebenszyklus ist Teil der Anforderungsverwaltung und wird in Kapitel 8, Anforderungsmanagement, vertieft.

Fallstudie

Eva – Anforderungsdokumentation

Für das Evaluierungssystem Eva wird beispielhaft die Dokumentation einer funktionalen und einer nichtfunktionalen Anforderung vorgestellt. Das hier verwendete Schema berücksichtigt alle Aspekte, die in den vorangegangenen Abschnitten eingeführt wurden. Die Metadaten erlauben eine effektive Verwaltung der Anforderung, Änderungen an der Anforderung werden in der Änderungshistorie dokumentiert.

Funktionale Anforderung

ID	FA_01
Name	Hochschulweiten Evaluationsbogen anlegen
Version	1.1
Motivation	Um hochschulweit eine einheitliche Gestaltung der Evaluationen zu erreichen, ist die Vorgabe einer einheitlichen Vorlage durch das Qualitätsmanagement notwendig.
Beschreibung	Das Evaluierungssystem muss den QM dabei unterstützen, eine neue Vorlage für einen Evaluationsbogen zu erstellen. Der QM kann alternativ einen komplett neuen Bogen erstellen, die Kopie eines archivierten Bogens verwenden oder eine zuvor gespeicherte Draft-Version weiter bearbeiten.
Querbezüge	NFA_01 Benutzbarkeit
Status	Vorgelegt
Quelle	Frau A. Meier (Qualitätsmanagerin der Hochschule)
Priorität	Hoch
Verantwortlich	Frau R. Luxemburg (RL)

Änderungshistorie

Datum	Beschreibung	Verantw.
1.3.2013	Anforderung initial erstellt	RL
12.4.2013	Anpassung von archivierten EB wurde ergänzt.	RL

Abnahmekriterien

Abnahmekriterium 1

Ausgangssituation	Der QM ist am System angemeldet.
Ereignis	Der QM erstellt mithilfe des Systems einen neuen Evaluationsbogen.
Akzeptanzkriterium	Der Bogen ist angelegt und editierbar.

Abnahmekriterium 2

Ausgangssituation	Der QM ist am System angemeldet. Ein Evaluationsbogen liegt im Archiv vor.
Ereignis	Der QM erzeugt einen neuen Evaluationsbogen auf der Basis eines archivierten Bogens.
Akzeptanzkriterium	Ein neuer Bogen ist angelegt und editierbar. Der ursprüngliche Bogen ist unverändert archiviert.

Abnahmekriterium 3

Ausgangssituation	Der QM ist am System angemeldet. Ein Evaluationsbogen liegt als Draft-Version vor.
Ereignis	Der QM öffnet die Draft-Version.
Akzeptanzkriterium	Der Bogen ist angelegt und editierbar.

Stabilität	Änderungen sind in größerem Maß nicht zu erwarten.
Offene Punkte	Zu klären ist, welche Art von Fragen auf dem Evaluationsbogen zugelassen werden.

Nichtfunktionale Anforderung

ID	NFA_01
Name	Benutzbarkeit
Version	1.0
Motivation	Das System wird von vielen verschiedenen Personen in der Hochschule genutzt, die teilweise aus nicht technischen Bereichen kommen.
Beschreibung	Das System muss für alle Anwendergruppen einfach zu bedienen sein. Für alle Anwender mit Ausnahme der Studierenden muss zur Bedienung des Systems eine etwa einstündige Einführung ausreichen, um das System bedienen zu können. Studierende müssen das System intuitiv anwenden können.
Querbezüge	
Status	Akzeptiert
Quelle	Herr S. Müller (Dekan der Fakultät 20)
Priorität	Hoch
Verantwortlich	Frau I. Schmidt (IS)

Änderungshistorie

Datum	Beschreibung	Verantw.
1.3.2013	Anforderung initial erstellt	IS

Abnahmekriterien

Abnahmekriterium 1

Metrik	Für Studiendekane und Dozenten: Dauer einer Einführung bis zur Fähigkeit, mit dem System adäquat zu arbeiten.
Akzeptanzkriterium	max. 2 Stunden
Operationalisierung	Durchführung einer Einführung mit abschließender Evaluation der Teilnehmer zur Einschätzung ihrer erworbenen Kenntnisse.

Abnahmekriterium 2

Metrik	Für Studierende: Dauer bis zum Verständnis für den Umgang mit der Anwendung.
Akzeptanzkriterium	Unmittelbar
Operationalisierung	Befragung von ausgewählten Studierenden nach Durchführung einer Bewertung.

Offene Punkte: -

4.1.5 Qualitätskriterien für Anforderungen

Qualitätskriterien von Anforderungen wurden bereits an mehreren Stellen im Buch erwähnt. Was aber ist genau darunter zu verstehen? Was unterscheidet eine qualitativ gute Anforderung von einer schlechten? Eine Antwort auf diese Frage liefert der IEEE[1]-Standard 830-1998 (IEEE 830-1998). Der Standard definiert acht Qualitätskriterien für Anforderungen. Die Kriterien beziehen sich teilweise auf einzelne Anforderungen, teil-

1 IEEE: Institute of Electrical and Electronics Engineers

weise auf die Gesamtheit aller Anforderungen. Dies ist notwendig, da Qualitätsmängel wie Widersprüche, inkonsistente Beschreibungen und Lücken häufig erst im Kontext mehrerer Anforderungen auffallen. ▶ Tabelle 4.3 listet die Qualitätskriterien nach IEEE 830 (frei übersetzt nach (IEEE 830-1998)).

Eine Verständigung darüber, dass die dokumentierten Anforderungen die geforderten Qualitätskriterien erfüllen sollten, ist normalerweise kein Problem. Die Frage bleibt jedoch, wie man die Eigenschaften tatsächlich umsetzt. Wie stellt man beispielsweise sicher, dass eine Anforderung korrekt oder eindeutig ist? Auch wenn es nicht die eine Methode gibt, die absolut sicherstellt, dass eine Anforderung alle Qualitätskriterien erfüllt, so gibt es doch eine Reihe von Maßnahmen, die dabei helfen, diesem Ziel möglichst nahe zu kommen. Viele der Maßnahmen wurden bereits in anderem Kontext eingeführt. Die dritte Spalte in Tabelle 4.3 nennt beispielhaft Maßnahmen, die eine Erfüllung der geforderten Qualitätskriterien in der Praxis unterstützen können. Die Liste der genannten Maßnahmen ist bei Weitem nicht vollständig, soll jedoch zeigen, dass die zum Teil abstrakt klingenden Qualitätseigenschaften aus dem Standard mit relativ einfachen und gängigen Mitteln sichergestellt werden können.

Kriterium	Beschreibung	Methoden zur Umsetzung
Correct (Korrekt)	Eine Anforderungsspezifikation ist korrekt genau dann, wenn jede Anforderungen, die dokumentiert ist, eine Anforderung ist, die von einem Stakeholder benötigt wird.	Gut durchgeführter Prozess zur Anforderungsermittlung und Bewertung, Durchführung von Reviews.
Unambiguous (Eindeutig)	Eine Anforderungsspezifikation ist eindeutig genau dann, wenn jede Anforderung darin genau eine Interpretation zulässt.	Einheitliche Regeln zur Anforderungsformulierung. Anlegen eines Glossars. (siehe Abschnitt 1.4.4).
Complete (Vollständig)	Eine Anforderungsspezifikation ist vollständig genau dann, wenn sie folgende Elemente enthält: - Alle signifikanten Anforderungen die sich auf Funktionalität, Performanz, Design-Constraints, Eigenschaften oder externe Schnittstellen beziehen. - Eine Definition aller Ergebnisse des Systems auf alle möglichen Arten von Eingaben. Dies betrifft das Verhalten im Fall gültiger wie ungültiger Eingaben. - Eine vollständige Referenzierung aller Abbildungen, Tabellen und Diagramme und eine vollständige Definition aller verwendeten Begriffe und Maßeinheiten.	Verwendung von Vorlagen für Anforderungsdokumente und einzelne Anforderungen. Durchführung von Reviews.
Consistent (Konsistent)	Eine Anforderungsspezifikation ist konsistent genau dann, wenn keine der beschriebenen Anforderungen zueinander im Widerspruch stehen.	Durchführung von Reviews. Verwendung von (semi-) formalen grafischen Modellierungsmethoden.

Tabelle 4.3: Qualitätskriterien Anforderungsspezifikationen nach IEEE 320.

Copyright

Daten, Texte, Design und Grafiken dieses eBooks, sowie die eventuell angebotenen eBook-Zusatzdaten sind urheberrechtlich geschützt. Dieses eBook stellen wir lediglich als **persönliche Einzelplatz-Lizenz** zur Verfügung!

Jede andere Verwendung dieses eBooks oder zugehöriger Materialien und Informationen, einschließlich

- der Reproduktion,
- der Weitergabe,
- des Weitervertriebs,
- der Platzierung im Internet, in Intranets, in Extranets,
- der Veränderung,
- des Weiterverkaufs und
- der Veröffentlichung

bedarf der **schriftlichen Genehmigung** des Verlags. Insbesondere ist die Entfernung oder Änderung des vom Verlag vergebenen Passwortschutzes ausdrücklich untersagt!
Bei Fragen zu diesem Thema wenden Sie sich bitte an: info@pearson.de

Zusatzdaten

Möglicherweise liegt dem gedruckten Buch eine CD-ROM mit Zusatzdaten bei. Die Zurverfügungstellung dieser Daten auf unseren Websites ist eine freiwillige Leistung des Verlags. **Der Rechtsweg ist ausgeschlossen.**

Hinweis

Dieses und viele weitere eBooks können Sie rund um die Uhr und legal auf unserer Website herunterladen:

http://ebooks.pearson.de