

ständigen und Verfeinern dieser Anforderungen in Anforderungen an das Produkt.

Anforderungskonfiguration

Eine Anforderungskonfiguration ist eine Zusammenstellung logisch zusammengehöriger Anforderungsversionen.

Anforderungsmanagement (CMMI)

Das Prozessgebiet „Anforderungsmanagement“ im CMMI beinhaltet die Verwaltung der vorhandenen Anforderungen bzw. Anforderungsartefakte.

Anforderungsmodell

Ein Anforderungsmodell ist in der Regel ein konzeptuelles Modell, das Anforderungen dokumentiert (Ziele, Szenarien oder lösungsorientierte Anforderungen).

Siehe: Konzeptuelles Modell

Anforderungsnachvollziehbarkeit

Siehe: Nachvollziehbarkeit

Anforderungsquelle

Eine Anforderungsquelle ist der Ursprung einer Anforderung.

Siehe: Stakeholder

Anforderungsspezifikation (Dokument)

Eine Anforderungsspezifikation ist ein Dokument, das spezifizierte Anforderungen enthält, d.h. Anforderungen, die definierten Spezifikationskriterien genügen.

Siehe: Spezifizieren von Anforderungen

Anforderungssteller (CMMI)

Anforderungssteller im CMMI sind die Stakeholder, die unmittelbar Anforderungen an das geplante System stellen.

Siehe: Stakeholder

Anforderungsversion

Eine Anforderungsversion ist ein definierter Änderungsstand eines Anforderungsartefakts.

Anwendungsfall

Deutscher Begriff für „Use Case“.

Siehe: Use Case

Applikations-Requirements-Engineering

Das Applikations-Requirements-Engineering ist eine Aktivität im Applikations-Engineering-Prozess in der Produktlinienentwicklung, in der Systemanforderungen und Rahmenbedingungen für ein zu realisierendes System

basierend auf wiederverwendbaren Domänenanforderungsartefakten sowie einem Domänenvariabilitätsmodell gewonnen und in einer Systemanforderungsspezifikation dokumentiert werden.

Siehe: Domänen-Requirements-Engineering

Attrappe

Deutscher Begriff für „Mock-Up“.

Siehe: Mock-Up

Attributierungsschema

Ein Attributierungsschema definiert die Attribute für einen Typ von Anforderungen. Für jedes Attribut legt das Attributierungsschema einen eindeutigen Attributnamen, die Attributsemantik, den Wertebereich und die Wertesemantik fest.

Siehe: Anforderungsattribut

Ausnahmeszenario

Ein Ausnahmeszenario definiert eine Interaktionsfolge, die ausgeführt wird, wenn in einem anderen Szenario (Haupt-, Alternativ- oder in einem weiteren Ausnahmeszenario) ein Ereignis eintritt, das die Erfüllung eines oder mehrerer mit dem Szenario assoziierter Ziele verhindert.

Siehe: Alternativszenario, Hauptszenario

B

Baseline

Englischer Begriff für Anforderungsbasislinie.

Siehe: Anforderungsbasislinie

Betrachtungsgegenstand

Ein Betrachtungsgegenstand ist eine Person oder ein materielles oder immaterielles Objekt, das bei der Systementwicklung berücksichtigt werden muss.

Siehe: Kontextaspekt

Bezeichner eines Anforderungsartefakts

Der Bezeichner eines Anforderungsartefakts ist ein Anforderungsattribut, das die Identifizierbarkeit des betrachteten Artefakts gewährleistet.

Siehe: Anforderungsattribut

C**Capability Maturity Model Integration (CMMI)**

Das Capability Maturity Model Integration ist ein Qualitätsmanagementmodell, das Vorschriften und Aktivitäten zur Sicherung und Verbesserung der Qualität von Prozessen und Produkten festlegt. Das CMMI definiert eine stufenförmige Darstellung sowie eine kontinuierliche Darstellung zur Bewertung von Organisationen hinsichtlich verschiedener Prozessgebiete.

Siehe: Prozessgebiet

Checkliste

Eine Checkliste fasst eine Anzahl von Fragen oder Aussagen zu einem Sachverhalt zusammen, die bei der Durchführung einer Entwicklungsaktivität beachtet werden sollen.

CMMI

Abkürzung für „Capability Maturity Model Integration“.

D**Darstellungstransformation**

Die Darstellungstransformation vollzieht sich bei der Kommunikation bzw. Dokumentation der Konzeptualisierung und gründet auf der Tatsache, dass abhängig von Wissen, persönlicher Prägung und Erfahrungen sowie der aktuellen Situation einer Person die gedanklichen Bilder unterschiedlich dokumentiert werden.

Siehe: Wahrnehmungstransformation

Datenfluss

Ein Datenfluss beschreibt den Transport von Daten von einem Systemteil zu einem anderen Systemteil.

(Quelle: [Yourdon 1989], Übersetzung des Autors)

Siehe: Datenflussdiagramm

Datenflussdiagramm (DFD)

Ein Datenflussdiagramm dokumentiert die Funktionalität eines Systems durch Funktionen (bzw. Prozesse), Datenflüsse und Datenspeicher. Datenflussdiagramme werden in der strukturierten Analyse eingesetzt.

Siehe: Strukturierte Analyse

Datenperspektive

Die Datenperspektive betrachtet die statische Struktur der Daten (d.h. Datentypen, deren Attribute sowie Beziehungen zwischen den Datentypen) und wird daher auch als Strukturperspektive bezeichnet.

Siehe: Funktionsperspektive, Verhaltensperspektive

Datenspeicher

Ein Datenspeicher beschreibt ruhende Datenpakete im System.

(Quelle: [Yourdon 1989], Übersetzung des Autors)

Siehe: Datenflussdiagramm

Deskriptives Szenario

Ein deskriptives Szenario verdeutlicht Abläufe und Interaktionen und wird erstellt, um Ziele und lösungsorientierte Anforderungen aufzudecken sowie zu konkretisieren.

Siehe: Erklärendes Szenario, Exploratives Szenario

DFD

Abkürzung für Datenflussdiagramm.

Direkte Beobachtung

Bei der direkten Beobachtung schaut der Beobachter den Stakeholdern bei der Durchführung einer bestimmten Aufgabe zu, analysiert ihre Tätigkeiten und stellt Verständnisfragen. (In Anlehnung an [Bray 2002])

Siehe: Ethnografische Beobachtung

Dokumentation (von Anforderungen)

Die Dokumentation von Anforderungen ist eine der drei Requirements-Engineering-Kernaktivitäten. Ihr Ziel ist es, alle Anforderungen gemäß den definierten Dokumentationsregeln zu dokumentieren bzw. gemäß den Spezifikationsregeln zu spezifizieren.

Siehe: Gewinnung, Übereinstimmung

Domänen-Requirements-Engineering

Das Domänen-Requirements-Engineering ist der Teilprozess der Produktlinienentwicklung, in dem die Gemeinsamkeiten und Unterschiede der Produkte definiert und die wiederverwendbaren Domänenanforderungsartefakte sowie das Domänenvariabilitätsmodell entwickelt werden.

Siehe: Applikations-Requirements-Engineering

E**Eindeutige Anforderung**

Eine Anforderung ist eindeutig, wenn die Dokumentation der Anforderung unter Berücksichtigung des jeweiligen Kontexts nur eine gültige Interpretation zulässt.

Siehe: Mehrdeutige Anforderung

Entwicklungsartefakt (CMMI)

Als Entwicklungsartefakte im CMMI werden sämtliche Artefakte bezeichnet, die im Lauf des Entwicklungsprozesses eines Systems erstellt werden.

Entwicklungsfacette

Die Entwicklungsfacette ist der Teilbereich des Systemkontexts, der alle Kontextaspekte umfasst, die den Entwicklungsprozess des Systems betreffen. Hierzu gehören z.B. Modellierungssprachen, computergestützte Entwicklungswerkzeuge, Qualitätsmanagementmodelle sowie Aspekte wie Vertrauenswürdigkeit, Sicherheit und Zuverlässigkeit von softwareintensiven Systemen.

Siehe: Gegenstandsfacette, Kontextaspekt, Nutzungsfacette, IT-Systemfacette

Erklärendes Szenario

Ein erklärendes Szenario wird zum Zweck der Erklärung von Interaktionen und Interaktionsfolgen erstellt. Es beinhaltet Argumente, Begründungen und alternative Sichtweisen der Stakeholder.

Siehe: Deskriptives Szenario, Exploratives Szenario

Essentielle Systemanalyse

Die essenzielle Systemanalyse [McMenamin und Palmer 1988] ist eine wichtige Weiterentwicklung der strukturierten Analyse, bei der zwischen der Essenz eines Systems und der Inkarnation eines Systems unterschieden wird.

Siehe: Essenz eines Systems, Inkarnation eines Systems

Essenz eines Systems

Die Essenz eines Systems ist die Menge der wahren Anforderungen an das System. Eine wahre Anforderung ist eine Eigenschaft oder eine Fähigkeit, die das System haben muss, unabhängig davon, wie das System später realisiert wird. (In Anlehnung an [McMenamin und Palmer 1984])

Siehe: Inkarnation eines Systems

Ethnografische Beobachtung

Bei der ethnografischen Beobachtung verbringt der Beobachter eine längere Zeit mit Stakeholdern, um aktiv ihre Arbeitsweisen und ihr Vorgehen zu erlernen und zu verstehen. (In Anlehnung an [Kotonya und Sommerville 1997])

Siehe: Direkte Beobachtung

Exploratives Interview

Das explorative Interview ist ein offenes Gespräch über einen Sachverhalt, das der Gewinnung von Informationen über die Ansichten des Interviewten in Bezug auf das geplante System dient. (In Anlehnung an [Oppenheim 1999])

Siehe: Interview, Standardisiertes Interview

Exploratives Szenario

Ein exploratives Szenario wird zum Zweck der Analyse und Bewertung alternativer Lösungsmöglichkeiten erstellt.

Siehe: Deskriptives Szenario, Erklärendes Szenario

Externe Validierung

Die externe Validierung ist eine Validierung durch Stakeholder, die nicht aus der entwickelnden Organisation bzw. Abteilung stammen.

F**Fähigkeitsgrad (CMMI)**

Der Fähigkeitsgrad eines Prozessgebiets in der kontinuierlichen Darstellung des CMMI beschreibt die Qualität einer Organisation bzw. ihrer Prozesse und Produkte mit Blick auf ein Prozessgebiet.

Feature

Ein Feature ist ein benutzererlebbares Merkmal eines Systems.

Fehlerwirkung (engl. *failure*)

Eine Fehlerwirkung ist eine Abweichung zwischen den erwarteten Ausgaben und den tatsächlich beobachteten Ausgaben bei der Ausführung eines Testobjekts.

Funktionale Anforderung

Eine funktionale Anforderung definiert eine vom System oder von einer Systemkomponente bereitzustellende Funktion bzw. Service.

Siehe: Nichtfunktionale Anforderung, Qualitätsanforderung

Funktionsgruppe

Eine Funktionsgruppe bündelt Systemanforderungen, die logisch zu einer Aufgabe des Systems gehören.

Funktionsperspektive

Die Funktionsperspektive betrachtet die Manipulation von Daten durch die Funktionen eines Systems, d.h. die Transformation von Eingaben in Ausgaben.

Siehe: Datenperspektive, Verhaltensperspektive

G

Gegenstandsfacette

Die Gegenstandsfacette ist der Teilbereich des Systemkontexts, der alle Gegenstände und Ereignisse umfasst, die ein System abbilden muss. Zudem beinhaltet die Gegenstandsfacette alle Aspekte, wie z.B. Gesetze und Standards, die die Abbildungsfunktion zwischen den Gegenständen und Ereignissen im Gegenstandsbereich und deren Abbildung im System beeinflussen.

Siehe: Entwicklungsfacette, IT-Systemfacette, Nutzungsfacette

Generische Praktik (CMMI)

Eine generische Praktik im CMMI unterstützt die Erfüllung eines generischen Ziels und beschreibt, auf welche Weise das Ziel erfüllt werden kann.

Generisches Ziel (CMMI)

Ein generisches Ziel im CMMI beschreibt die notwendigen Eigenschaften zur Institutionalisierung der Prozesse eines Prozessgebiets. Generische Ziele haben für mehrere Prozessgebiete denselben Wortlaut.

Gewinnung (von Anforderungen)

Die Gewinnung von Anforderungen ist eine der drei Requirements-Engineering-Kernaktivitäten. Sie verfolgt drei Teilziele:

- (1) Die Identifikation von Anforderungsquellen im Kontext des geplanten Systems
- (2) Die Gewinnung von existierenden Anforderungen
- (3) Die Entwicklung von innovativen Anforderungen

Siehe: Dokumentation, Übereinstimmung

Glossar

Ein Glossar legt die spezifische Bedeutung einer Menge von Fachbegriffen einer Domäne (d.h. eine Fachterminologie) fest. Neben den Begriffsdefinitionen kann ein Glossar Verweise zwischen verwandten Begriffen sowie Beispiele zur Erläuterung der Begriffe beinhalten.

Grobkonzept (CMMI)

Das Grobkonzept in der Terminologie des CMMI entspricht dem Lastenheft.

Siehe: Lastenheft

H

Hauptszenario

Ein Hauptszenario dokumentiert die Interaktionsfolge, die normalerweise ausgeführt wird, um eines oder mehrere mit dem Szenario assoziierte Ziele zu erfüllen.

Siehe: Alternativszenario, Ausnahme-szenario

Homonym

Ein Homonym ist ein Wort, das mehrere Bedeutungen besitzt. Beispiele für Homonyme sind „Bank“, „Tau“, „Steuer“.

Siehe: Lexikalische Mehrdeutigkeit, Synonym

I

Informationsmodell

Ein Informationsmodell ist ein abstraktes Abbild der Struktur von Informationen eines betrachteten Realitätsausschnitts. Informationsmodelle werden unter Verwendung einer konzeptuelle Modellierungssprache definiert.

Siehe: Modell

Inkarnation eines Systems

Die Inkarnation eines Systems ist die Gesamtheit aller Elemente (z.B. Personen, PCs, Software, Bleistifte, Telefone und CPUs), die dazu benutzt werden, die Essenz eines Systems zu realisieren. (In Anlehnung an [McMenamin und Palmer 1988])

Siehe: Essenz eines Systems

Inspektion

Eine Inspektion ist ein Prozess zur Qualitätsverbesserung und besteht aus den zwei

Hauptkomponenten „Produktverbesserung“ (Verbesserung des Dokuments selbst) und „Prozessverbesserung“ (Verbesserung des Prozesses der Dokumentenerstellung sowie des Inspektionsprozesses). (Quelle: [Gilb und Graham 1993], Übersetzung des Autors)

Instanzszenario

Ein Instanzszenario beschreibt konkrete Ausprägungen eines Szenario und somit die Ausprägung existierender oder imaginärer Interaktionen mit konkreten Ein- und Ausgaben zwischen konkreten Personen und/oder Systemen mit dem System.

Siehe: Typszenario

Interaktionsszenario

Ein Interaktionsszenario beschreibt Interaktionen zwischen dem System und den Systemnutzern (Personen und/oder Systeme) im Kontext des Systems. (In Anlehnung an [Pohl und Haumer 1997])

Siehe: Systeminternes Szenario, Kontextszenario

Interessenkonflikt (über Anforderungen)

Ein Interessenkonflikt liegt vor, wenn sich die Anforderungen und/oder die Ziele von Stakeholdern in Bezug auf das geplante System widersprechen.

Siehe: Sachkonflikt, Wertekonflikt

Interne Validierung

Eine interne Validierung ist eine Validierung, an der ausschließlich Stakeholder der entwickelnden Organisation bzw. Abteilung beteiligt sind.

Interview

Ein Interview ist ein Gespräch, bei dem ein Interviewer einen oder mehrere Interviewte über deren Anforderungen und Wünsche an das geplante System befragt.

Siehe: Exploratives Interview, Standardisiertes Interview

Ist-Analyse

Eine Ist-Analyse dient der Ermittlung von Fakten über existierende Systeme und Prozesse. Die gewonnenen Fakten werden konsolidiert und in einem Ist-Modell dokumentiert.

Siehe: Soll-Analyse

Ist-Modell

Ein Ist-Modell dokumentiert den Ist-Zustand eines Systems. Ist-Modelle bilden die aktuelle Situation ab und bieten eine Grundlage für die Analyse und Identifikation möglicher Verbesserungen eines Systems.

Siehe: Soll-Modell

IT-Systemfacette

Die IT-Systemfacette ist der Teilbereich des Systemkontexts, der alle Kontextaspekte umfasst, die die existierende oder zukünftige IT-Systemumgebung des geplanten Systems einschließlich der existierenden IT-Strategien betreffen.

Siehe: Entwicklungsfacette, Gegenstandsfacette, Nutzungsfacette

K

Konflikt (im Requirements Engineering)

Ein Konflikt im Requirements Engineering besteht, wenn die Bedürfnisse und/oder Wünsche verschiedener Stakeholder (oder Gruppen von Stakeholdern) an das geplante System sich widersprechen oder nicht alle Bedürfnisse und Wünsche berücksichtigt werden können.

Siehe: Übereinstimmung

Konkrete Syntax

Die konkrete Syntax einer Sprache umfasst Regeln zur Anordnung der Sprachkonstrukte einer Sprache sowie konkrete (meist grafische) Repräsentationen für die einzelnen Sprachkonstrukte.

Siehe: Abstrakte Syntax

Kontext einer Anforderung

Der Kontext einer Anforderung besteht aus Kontextaspekten des Systemkontexts sowie aus anderen Anforderungen, die mit der betrachteten Anforderung in Beziehung stehen.

Kontextaspekt

Ein Kontextaspekt ist eine Anforderungsquelle, ein Betrachtungsgegenstand, eine Eigenschaft eines Betrachtungsgegenstands oder eine Beziehung zwischen Betrachtungsgegenständen.

Siehe: Anforderungsquelle, Betrachtungsgegenstand, Systemkontext

Kontextfacette

Eine Kontextfacette ist einer der vier folgenden Teilbereiche des Systemkontexts: Entwicklungsfacette, Gegenstandsfacette, IT-Systemfacette, Nutzungsfacette.

Siehe: Systemkontext

Kontextgrenze

Die Kontextgrenze separiert den relevanten Teil der Umgebung eines Systems von dem irrelevanten Teil, d.h. dem Teil, der keinen Einfluss auf die Systementwicklung hat und daher im Requirements Engineering nicht betrachtet wird.

Siehe: Systemgrenze, Systemkontext

Kontextszenario

Ein Kontextszenario beinhaltet neben den direkten Interaktionen zwischen dem System und den Systemnutzern weitere Kontextinformationen, die für das System und seine Verwendung wichtig sind. (In Anlehnung an [Pohl und Haumer 1997])

*Siehe: Interaktionsszenario,
Systeminternes Szenario*

Kontinuierliches Requirements Engineering

Das kontinuierliche Requirements Engineering beinhaltet die kontinuierliche Dokumentation, Gewinnung und Übereinstimmung von Anforderungsartefakten sowie die kontinuierliche Integration der relevanten Veränderungen in der Realität auf einer konzeptuellen Ebene. Es existiert in den Ausprägungen phasenübergreifendes kontinuierliches Requirements Engineering und projektübergreifendes kontinuierliches Requirements Engineering.

Siehe: Phasenübergreifendes Requirements Engineering, Projektübergreifendes Requirements Engineering

Konzeptuelle Modellierungssprache

Eine konzeptuelle Modellierungssprache wird durch eine abstrakte Syntax, durch die Semantik der Modellkonstrukte der Sprache sowie durch eine Notation für die Modellkonstrukte definiert.

Siehe: Abstrakte Syntax, Konkrete Syntax, Semantik

Konzeptuelles Modell

Ein konzeptuelles Modell ist ein Modell, das durch Verwendung einer konzeptuellen Modellierungssprache konstruiert wurde.

Kundenanforderung (CMMI)

Eine Kundenanforderung ist eine Anforderung, die vom Kunden gestellt wird.

Siehe: Produktanforderung

L**Lastenheft**

Das Lastenheft enthält eine Definition der Systemvision, eine Beschreibung der wesentlichen Systemziele (Funktionen und Qualitäten) und benennt wichtige Kontextaspekte (z.B. Rahmenbedingungen) der vier Kontextfacetten sowie ihrer Beziehungen zur Vision und zu den definierten Systemzielen.

Lexikalische Mehrdeutigkeit

Die lexikalische Mehrdeutigkeit drückt aus, dass eine Folge von Zeichen (z.B. ein Wort) mehrere (evtl. verwandte) Bedeutungen besitzt.

Siehe: Homonym, Synonym, Syntaktische Mehrdeutigkeit

Lösungsorientierte Anforderung

Lösungsorientierte Anforderungen sind Anforderungen an das geplante System. Sie werden typischerweise in den Perspektiven Daten/Struktur, Funktionen und Verhalten definiert und umfassen zudem die definierten Qualitätseigenschaften an das System.

Siehe: Anforderungsartefakt, Datenperspektive, Funktionsperspektive, Qualitätsanforderung, Verhaltensperspektive

M**Management (im Requirements Engineering)**

Das Management ist eine der beiden Querschnittsaktivitäten des Requirements Engineering mit den folgenden Aufgaben:

- (1) Beobachtung des Systemkontexts
- (2) Management der Requirements-Engineering-Aktivitäten
- (3) Management der Anforderungsartefakte

Siehe: Validierung

Management der Nachvollziehbarkeit

Das Management der Nachvollziehbarkeit von Anforderungen betrachtet die Aufzeichnung und Verwaltung von Nachvollziehbarkeitsinformationen.

Management von Anforderungskonfigurationen

Das Management von Anforderungskonfigurationen dient der Identifikation und Verfolgbarkeit der Anforderungskonfigurationen eines Systems und unterstützt die systematische Steuerung von Konfigurationsänderungen.

Management von Anforderungsversionen

Das Management von Anforderungsversionen betrachtet die Identifizierbarkeit, Evolution und Verfolgbarkeit von Anforderungsversionen über den Lebenszyklus hinweg.

Mehrdeutige Anforderung

Eine Anforderung ist mehrdeutig, wenn sie mehr als eine gültige Interpretation besitzt.

Siehe: Eindeutige Anforderung, Lexikalische Mehrdeutigkeit, Semantische Mehrdeutigkeit, Syntaktische Mehrdeutigkeit

Metamodell einer Modellierungssprache

Ein Metamodell definiert eine konzeptuelle Modellierungssprache, die für die Konstruktion von konzeptuellen Modellen verwendet wird.

Siehe: Konzeptuelles Modell

Missbrauchszenario

Ein Missbrauchszenario beschreibt eine unerwünschte Verwendung des Systems, d.h. eine Folge von Interaktionsschritten, die das System nicht gestatten soll.

Siehe: Negatives Szenario, Positives Szenario

Mock-Up

Ein Mock-Up ist ein plastisches Modell des geplanten Systems in Originalgröße und Farbe, jedoch ohne Funktionen.

Siehe: Attrappe

Modell

Ein Modell ist ein abstrahierendes Abbild einer existierenden oder fiktiven Realität.

Siehe: Modellbasierte Anforderung

Modellbasierte Anforderung

Modellbasierte Anforderungen sind Anforderungen (Ziele, Szenarien oder lösungsorientierte Anforderungen), die in Anforderungsmodellen dokumentiert sind.

Siehe: Natürlichsprachliche Anforderung

N**Nachvollziehbarkeit (CMMI)**

Die Nachvollziehbarkeit im CMMI beschreibt, welcher Quelle eine Anforderung zuzuordnen ist und durch welche weiteren Anforderungen die betrachtete Anforderung verfeinert wurde.

Nachvollziehbarkeit (einer Anforderung)

Die Nachvollziehbarkeit einer Anforderung ist die Fähigkeit, den Lebenszyklus der Anforderung vom Ursprung der Anforderung über die verschiedenen Verfeinerungs- und Spezifikationschritte bis hin zur Berücksichtigung der Anforderung in nachgelagerten Entwicklungsartefakten verfolgen zu können. (In Anlehnung an [Gotel und Finkelstein, 1994])

Siehe: Anforderungsnachvollziehbarkeit

Natürlichsprachliche Anforderung

Eine natürlichsprachliche Anforderung ist eine Anforderung, die ausschließlich unter Verwendung der natürlichen Sprache dokumentiert ist.

Siehe: Modellbasierte Anforderung

Negatives Szenario

Ein negatives Szenario beschreibt eine Interaktionsfolge, die zur Nichterfüllung der mit dem Szenario assoziierten Ziele führt.

Siehe: Positives Szenario, Missbrauchsszenario

Nichtfunktionale Anforderung

Eine nichtfunktionale Anforderung ist eine unterspezifizierte funktionale Anforderung und/oder eine Qualitätsanforderung.

Siehe: Unterspezifizierte Anforderung, Qualitätsanforderung

Normsprache

Eine Normsprache ist eine eingeschränkte natürliche Sprache. Sie besitzt eine in Bezug auf eine spezifische Domäne eingeschränkte Grammatik (Syntax) und definiert eine Menge von Begriffen (Semantik), die unter

Verwendung der vorgegebenen Grammatik zur Konstruktion von Aussagen über die Domäne verwendet werden können.

Nutzungsfacette

Die Nutzungsfacette ist der Teilbereich des Systemkontexts, der alle Aspekte der Systemverwendung umfasst, wie bspw. die Nutzungsabläufe, Nutzungseinschränkungen oder Nutzungsbeeinflussungen.

Siehe: Entwicklungsfacette, Gegenstands-facette, IT-Systemfacette

P

Papierprototyp

Ein Papierprototyp beschreibt die grafische Oberfläche eines geplanten Systems in Form von Skizzen und Bildern.

Siehe: Prototyp, Mock-Up

Perspektivenbasiertes Lesen

Das perspektivenbasierte Lesen ist eine Lesetechnik, bei der Entwicklungsartefakte aus unterschiedlichen Perspektiven betrachtet werden, z.B. aus der Sicht eines Kunden, eines Softwarearchitekten oder Testers. Perspektivenbasiertes Lesen kann bspw. bei Inspektionen eingesetzt werden.

Pflichtenheft

Das Pflichtenheft verfeinert das Lastenheft und detailliert somit die im Lastenheft beschriebene Vision und die Systemziele (abstrakte Funktionen und Qualitäten) sowie ggf. definierte Rahmenbedingungen im Hinblick auf die angestrebte technische Umsetzung (Realisierung) des Systems.

Phasenübergreifendes Requirements Engineering

Das phasenübergreifende Requirements Engineering ist eine Ausprägung des kontinuierlichen Requirements Engineering. Es etabliert das Requirements Engineering als eine Querschnittsaktivität, die den gesamten Entwicklungsprozess eines Systems begleitet und eine konsistente und nachvollziehbare Erfassung und Verwaltung von Anforderungen gewährleistet.

Siehe: Kontinuierliches Requirements Engineering, Projektübergreifendes Requirements Engineering

Physikalisches Modell

Ein physikalisches Modell ist ein Modell, das eine Inkarnation eines in der Entwicklung befindlichen oder existierenden Systems darstellt.

Siehe: Inkarnation eines Systems

Positives Szenario

Ein positives Szenario beschreibt eine Interaktionsfolge, die zur Erfüllung der mit dem Szenario assoziierten Ziele führt.

Siehe: Missbrauchszenario, Negatives Szenario

Pragmatik

Die Pragmatik betrachtet die Wirkung von Modellen auf den Interpretierenden.

Siehe: Semantik

Pragmatische Qualität

Die pragmatische Qualität eines konzeptuellen Modells betrachtet den Umfang, in dem die Informationen über den Gegenstandsbereich, die im betrachteten Modell abgebildet sind, auch für den jeweiligen Verwendungszweck des Modells geeignet sind. Dies schließt auch die Verständlichkeit der Modelle für die Interpretierenden ein.

Pre-Traceability

Unter Pre-Traceability werden Nachvollziehbarkeitsbeziehungen von einer Anforderung zu Artefakten subsumiert, die der Anforderung im Projektverlauf vorgelagert sind, z.B. Anforderungsquellen.

Priorität einer Anforderung

Die Priorität einer Anforderung dokumentiert die Bedeutung der Anforderung bzgl. eines oder mehrerer ausgewählter Priorisierungskriterien. Die Priorität einer Anforderung kann dabei isoliert für jede Anforderung oder durch paarweise Vergleiche bestimmt werden.

Produktanforderung (CMMI)

Produktanforderungen sind in der Terminologie des CMMI Kundenanforderungen, die in der Sprache des Entwicklers formuliert wurden.

Siehe: Kundenanforderung

Projektübergreifendes Requirements Engineering

Das projektübergreifende Requirements Engineering ist eine Ausprägung des kontinuier-

lichen Requirements Engineering. Es ermöglicht zu einem beliebigen Zeitpunkt die von einem neuen System (bzw. System-Release) umzusetzenden Anforderungen aus den bekannten aktuellen Anforderungen zusammenzustellen. Rückmeldungen der einzelnen Entwicklungsprojekte werden kontinuierlich in die Anforderungsbasis integriert. Gleiches gilt für Veränderungen im Systemkontext.

Siehe: Kontinuierliches Requirements Engineering, Phasenübergreifendes Requirements Engineering

Prototyp

Ein Prototyp ist eine initiale Version eines Softwaresystems, die verwendet wird, um Konzepte zu demonstrieren, Entwürfe zu erproben und generell mehr über das Problem und dessen mögliche Lösungen herauszufinden. (Quelle: [Sommerville, 2004], Übersetzung des Autors)

Prozessgebiet (CMMI)

Prozessgebiete umfassen die Anforderungen an die Qualität für einen Themenbereich wie bspw. das Requirements Engineering, technische Lösungen, Projektmanagement oder die organisationsweite Innovation und Verbreitung. Um die Anforderungen in Bezug auf ein Prozessgebiet zu erfüllen, müssen generische und spezifische Ziele des Prozessgebiets erreicht werden.

Q

Qualitätsanforderung

Eine Qualitätsanforderung definiert eine qualitative Eigenschaft des Systems oder einzelner Funktionen des Systems.

Siehe: Funktionale Anforderung, Nichtfunktionale Anforderung

Qualitätsmanagementmodell

Ein Qualitätsmanagementmodell definiert die Vorschriften und Aktivitäten zur Sicherung und Verbesserung der Qualität von Prozessen und Produkten.

R

Rahmenbedingung (engl. *constraint*)

Eine Rahmenbedingung ist eine organisatorische oder technologische Anforderung, die

die Art und Weise einschränkt, wie ein Produkt entwickelt wird. (Quelle: [Robertson und Robertson, 2006], Übersetzung des Autors)

Reifegrad (CMMI)

Der Reifegrad (engl. *maturity level*) dient der Bewertung einer Organisation in der stufenförmigen Darstellung des CMMI. Das CMMI unterscheidet die Reifegrade initial, gemagt, definiert, quantitativ und optimierend. Eine Organisation erreicht einen Reifegrad genau dann, wenn sie die generischen und spezifischen Ziele aller Prozessgebiete des betrachteten Reifegrads erfüllt.

Siehe: Capability Maturity Model Integration, Fähigkeitsgrad

Release

Ein Release ist eine Anforderungskonfiguration, die zur Auslieferung eines Systems gebildet wird.

Siehe: Anforderungskonfiguration

Requirement

Englischer Begriff für „Anforderung“.

Siehe: Anforderung

Requirements Engineering

Das Requirements Engineering ist ein kooperativer, iterativer, inkrementeller Prozess, dessen Ziel es ist zu gewährleisten, dass

- (1) alle relevanten Anforderungen bekannt und in dem erforderlichen Detaillierungsgrad verstanden sind,
- (2) die involvierten Stakeholder eine ausreichende Übereinstimmung über die bekannten Anforderungen erzielen,
- (3) alle Anforderungen konform zu den Dokumentationsvorschriften dokumentiert bzw. konform zu den Spezifikationsvorschriften spezifiziert sind.

Es umfasst zudem die Querschnittsaktivitäten Management und Validierung.

S

Sachkonflikt (über Anforderungen)

Ein Sachkonflikt über die Anforderungen an das geplante System liegt vor, wenn Stakeholder falsch oder unvollständig über die Anforderungen an das geplante System informiert

werden und/oder sie die Bedeutung von Anforderungen unterschiedlich interpretieren.

Siehe: Interessenkonflikt, Wertekonflikt

Semantik

Die Semantik einer Sprache legt die Bedeutung der Sprachkonstrukte der Sprache sowie durch die Anordnung der Sprachkonstrukte gebildeten Ausdrücke fest.

Siehe: Syntax, Pragmatik

Semantische Mehrdeutigkeit

Semantische Mehrdeutigkeit liegt dann vor, wenn im gegebenen Kontext ein Satz verschiedenartig interpretiert werden kann, auch wenn keine lexikalische oder syntaktische Mehrdeutigkeit vorliegt.

Siehe: Lexikalische Mehrdeutigkeit, Syntaktische Mehrdeutigkeit

Semantische Qualität

Die semantische Qualität eines konzeptuellen Modells ist ein Maß für die Übereinstimmung der im Modell dokumentierten Informationen mit dem abgebildeten Betrachtungsgegenstand (Domäne). Sie wird im Wesentlichen durch die Vollständigkeit des Modells sowie durch die Validität des Modells (d.h. die Korrektheit der abgebildeten Gegenstände) bestimmt.

Software Requirements Specification (SRS)

Eine Software Requirements Specification ist ein Dokument, das die Anforderungen an die Software eines geplanten, softwareintensiven Systems enthält.

Siehe: System Requirements Specification

Softwaretest

Unter einem Softwaretest versteht man die systematische Ausführung einer Softwareeinheit (des Testobjekts) mit dem Ziel, Fehlerwirkungen aufzudecken. (In Anlehnung an [Spillner und Linz 2004; Myers 2001])

Soll-Analyse

Während der Soll-Analyse werden (basierend auf einem Ist-Modell) Verbesserungsmöglichkeiten für das geplante System identifiziert.

Siehe: Ist-Analyse

Soll-Modell

Das Soll-Modell dokumentiert Eigenschaften des geplanten Systems und definiert die Vorgaben für die Realisierung dieses Systems.

Siehe: Ist-Modell

Spezifizieren von Anforderungen

Das Spezifizieren von Anforderungen ist eine spezifische Ausprägungsform des Dokumentierens von Anforderungen. Ziel des Spezifizierens ist es, eine Anforderungsspezifikation zu erstellen, die den vorgegebenen Spezifikationsvorschriften für Anforderungen genügt.

Siehe: Dokumentation

Spezifische Praktik (CMMI)

Spezifische Praktiken im CMMI unterstützen die Erfüllung spezifischer Ziele und erläutern, auf welche Weise ein spezifisches Ziel erfüllt werden kann.

Siehe: Generische Praktik

Spezifisches Ziel (CMMI)

Spezifische Ziele im CMMI sind Ziele, die einem spezifischen Prozessgebiet zugeordnet sind.

Siehe: Generisches Ziel

SRS

Abkürzung für „Software Requirements Specification“.

Stakeholder (im Requirements Engineering)

Ein Stakeholder ist eine Person oder eine Organisation, die ein potenzielles Interesse an dem zukünftigen System hat und somit in der Regel auch Anforderungen an das System stellt. Eine Person kann dabei die Interessen von mehreren Personen oder Organisationen vertreten, d.h. mehrere Rollen einnehmen. (In Anlehnung an [Robertson und Robertson 2006])

Standardisiertes Interview

Im standardisierten Interview stellt der Interviewer einem Interviewten vorbereitete Fragen in einer vorher festgelegten Reihenfolge. (In Anlehnung an [Oppenheim 1999])

Siehe: Exploratives Interview, Interview

Status (CMMI)

Der Status einer Anforderung in der Terminologie des CMMI gibt Auskunft darüber, ob eine Anforderung stabil ist oder Änderungen zu erwarten sind.

Strukturierte Analyse

Die strukturierte Analyse ist eine Methode zur Systemanalyse, in deren Zentrum die Funktionen des zu entwickelnden Systems stehen. Die Funktionen des Systems werden durch hierar-

chische Datenflussdiagramme beschrieben, die statische Struktur der Daten durch ein Datenlexikon und das Verhalten elementarer Funktionen durch Minispezifikationen.

Siehe: Essenzielle Systemanalyse, Datenflussdiagramm

Synonym

Ein Synonym ist ein Wort, das die gleiche Bedeutung besitzt wie ein anderes Wort. Beispiele für Synonyme sind „Streichholz“/„Zündholz“ oder „Auto/Kraftfahrzeug“.

Siehe: Homonym, Lexikalische Mehrdeutigkeit

Syntaktische Mehrdeutigkeit

Syntaktische Mehrdeutigkeit liegt dann vor, wenn einer Wortfolge mehr als eine grammatische Struktur zugeordnet werden kann und die Wortfolge für die verschiedenen Strukturen auch verschiedene Bedeutungen aufweist. (Quelle: [Hirst 1987])

Siehe: Lexikalische Mehrdeutigkeit

Syntaktische Qualität

Die syntaktische Qualität eines konzeptuellen Modells betrachtet den Umfang, in dem das Modell den syntaktischen Vorgaben der Modellierungssprache entspricht.

Siehe: Syntax

Syntaktisches Anforderungsmuster

Ein syntaktisches Anforderungsmuster dokumentiert wiederverwendbares Wissen über die Formulierung natürlichsprachlicher Anforderungen, indem es syntaktische Strukturen für die Dokumentation von natürlichsprachlichen Anforderungen vorgibt.

Syntax

Die Syntax einer Sprache legt die Menge der Sprachkonstrukte der Sprache fest und definiert Regeln zur Kombination dieser Sprachkonstrukte.

Siehe: Abstrakte Syntax, Konkrete Syntax, Pragmatik, Semantik

SysRS

Abkürzung für „System Requirements Specification“.

System Requirements Specification (SysRS)

Die System Requirements Specification ist ein Dokument, das die Anforderungen an ein

durch Hardware und Software zu realisierendes System enthält.

Siehe: Software Requirements Specification

Systemanalyse

Gegenstand der Systemanalyse ist die Untersuchung existierender Systeme oder Prozesse mit dem Ziel, Anforderungen an ein geplantes System zu entwickeln.

Siehe: Essentielle Systemanalyse

Systemgrenze

Die Systemgrenze separiert das geplante System von seiner Umgebung. Sie grenzt das System von den Teilen der Umgebung ab, die durch den Entwicklungsprozess nicht verändert werden können.

Siehe: Kontextgrenze, Systemkontext

Systeminternes Szenario

Ein systeminternes Szenario betrachtet systeminterne Interaktionen in Form von Interaktionsfolgen zwischen Systembestandteilen. (In Anlehnung an [Pohl und Haumer 1997])

Siehe: Interaktionsszenario, Kontextszenario

Systemkontext

Der Systemkontext ist der Teil der Umgebung eines Systems, der für die Definition und das Verständnis der Anforderungen an das System relevant ist. Der Systemkontext besteht aus den vier Kontextfacetten: Gegenstandsfacette, IT-Systemfacette, Nutzungsfacette und Entwicklungsfacette.

Siehe: Kontext einer Anforderung, Kontextaspekt, Kontextfacette, Kontextgrenze, Systemgrenze

Szenario

Ein Szenario beschreibt ein konkretes Beispiel für die Erfüllung bzw. Nichterfüllung eines oder mehrerer Ziele. Es konkretisiert dadurch eines oder mehrere Ziele. Ein Szenario enthält typischerweise eine Folge von Interaktionsschritten und setzt diese in Bezug zum Systemkontext.

Siehe: Ziel

Szenario (CMMI)

Ein Szenario im CMMI ist eine Folge von Ereignissen, die bei der Nutzung eines Systems auftreten kann. Szenarien werden im CMMI

genutzt, um Bedürfnisse der Stakeholder explizit zu machen.

T

Terminator/Schnittstelle (engl. terminator)

Terminatoren beschreiben externe Objekte (Personen, Personengruppen, Organisationen oder Systeme), die mit dem System kommunizieren, die aber durch den Entwicklungsprozess nicht verändert werden können. (In Anlehnung an [Yourdon 1989])

Siehe: Datenflussdiagramm

Testfall

Ein Testfall umfasst die für die Durchführung eines Tests notwendigen Vorbedingungen, die Menge der Eingaben und erwarteten Ausgaben, Prüfanweisungen sowie die erwarteten Nachbedingungen. (In Anlehnung an [Spillner und Linz 2004])

Testfallszenario

Ein Testfallszenario ist ein Typszenario, das Typen von Eingaben, erwartete Ausgaben und die Interaktionen zwischen Typen von Akteuren spezifiziert, die am Test beteiligt sind.

Siehe: Typszenario

Typ-A-Szenario

Siehe: Systeminternes Szenario

Typ-B-Szenario

Siehe: Interaktionsszenario

Typ-C-Szenario

Siehe: Kontextszenario

Typszenario

Ein Typszenario abstrahiert über eine konkrete Ausprägung von Akteuren und Ein- und Ausgaben eines Szenarios, d.h. Typszenarien beschreiben die Interaktionen zwischen dem System durch Typen von Akteuren und Typen von Ein- und Ausgaben.

U

Übereinstimmung (über Anforderungen)

Die Übereinstimmung ist eine der drei Requirements-Engineering-Kernaktivitäten. Sie verfolgt vier Teilziele: (1) Identifikation von Konflikten; (2) Analyse der Konfliktursache; (3) Auflösung der Konflikte durch geeignete

Strategien; (4) Dokumentation der Konfliktauflösung einschließlich der Begründungen.

Siehe: Dokumentation, Gewinnung

Und-Oder-Baum (Und-Oder-Zielbaum)

Ein Und-Oder-Baum ist ein Baum, dessen Knoten Ziele und dessen Kanten Dekompositionsbeziehungen zwischen Zielen dokumentieren. Ein Knoten kann hierbei höchstens eine eingehende Kante besitzen.

Siehe: Und-Oder-Graph

Und-Oder-Graph (Und-Oder-Ziel-Graph)

Ein Und-Oder-Graph ist ein gerichteter azyklischer Graph, dessen Knoten Ziele und dessen Kanten Dekompositionsbeziehungen zwischen Zielen repräsentieren. Im Gegensatz zu den Knoten eines Und-Oder-Baums können die Knoten eines Und-Oder-Graphen mehr als eine eingehende Kante besitzen.

Siehe: Und-Oder-Baum

Unterspezifizierte Anforderung

Eine unterspezifizierte Anforderung ist vage definiert und lässt dadurch mehrere Interpretationen zu. Da unterspezifizierte Anforderungen nicht objektiv überprüft werden können, müssen sie durch funktionale Anforderungen und ggf. Qualitätsanforderungen konkretisiert werden.

Siehe: Funktionale Anforderung, Qualitätsanforderung

Use Case

Ein Use Case dokumentiert Aktionsfolgen (Szenarien) einschließlich Alternativ- und Ausnahmesequenzen, die ein System oder eine Systemkomponente bei einer Interaktion mit externen Objekten ausführt, um einen Mehrwert zu erbringen. (In Anlehnung an [Rumbaugh et al. 2005])

Siehe: Alternativszenario, Ausnahmeszenario, Hauptszenario

Use-Case-Szenario

Ein Use-Case-Szenario ist eine gültige Interaktionsfolge, die sich aus den für den Use Case definierten Haupt-, Alternativ- und Ausnahmeszenarien ergibt und zu einer definierten Terminierung des Use Case führt. Terminierung bedeutet dabei, dass das Use-Case-Szenario entweder zu der Erfüllung der mit dem Use Case assoziierten Ziele oder zu einem definierten Abbruch führt.

V

Validierung (im Requirements Engineering)

Die Validierung ist eine Querschnittsaktivität des Requirements Engineering. Ihre Aufgabe ist die Überprüfung der Eingaben, der Aktivitätsdurchführung und der Ausgaben (Anforderungsartefakte) der Requirements-Engineering-Kernaktivitäten auf die Erfüllung der definierten Qualitätskriterien. Die Validierung erfolgt unter Einbeziehung der relevanten Stakeholder, weiterer Anforderungsquellen (Standards, Gesetze etc.) sowie ggf. externer Personen.

Siehe: Management

Verhaltensperspektive

Die Verhaltensperspektive beschreibt das Verhalten des Systems in Form von erlaubten Zuständen, Zustandsänderungen und erzeugten Ausgaben.

Siehe: Datenperspektive, Funktionsperspektive

Vision eines Systems (kurz: Vision)

Die Vision eines Systems beschreibt eine signifikante gewünschte Veränderung der aktuellen Realität.

W

Wahrnehmungstransformation

Bei der Wahrnehmungstransformation wird im Zug der Konstruktion eines gedanklichen Abbilds der beobachteten Realität dieses Abbild unbewusst transformiert.

Siehe: Darstellungstransformation

Wertekonflikt (über Anforderungen)

Ein Wertekonflikt über die Anforderungen an das geplante System liegt vor, wenn verschiedene Stakeholder die Anforderungen an das geplante System bzgl. bestimmter Bewertungskriterien (z.B. Wichtigkeit) unterschiedlich bewerten oder ihnen eine unterschiedliche Bedeutung beimessen.

Siehe: Interessenkonflikt, Sachkonflikt

Z

Ziel

Ein Ziel ist die intentionale Beschreibung eines charakteristischen Merkmals des zu entwickelnden Systems bzw. des zugehörigen Entwicklungsprozesses.

Zielmodell

Ein Zielmodell ist ein konzeptuelles Modell, das Ziele, deren Dekomposition in Teilziele sowie ggf. weitere Abhängigkeiten zwischen (Teil-)Zielen dokumentiert.

Siehe: Und-Oder-Baum, Und-Oder-Graph, konzeptuelles Modell