

IN DIESEM KAPITEL

Inwiefern uns Requirements Engineering hilft

Erläuterung der zentralen Begriffe

Aufgaben eines Requirements Engineers

Welche Zertifizierungen angeboten werden

Kapitel 1

Das ist Requirements Engineering

Für den Erfolg von Entwicklungsvorhaben – egal ob Projekt oder kontinuierliche Produktentwicklung – ist es entscheidend, sich erst mal klarzumachen, wozu das System überhaupt dienen soll und wie es dafür beschaffen sein muss. Das klingt erst mal selbstverständlich, wird aber trotzdem immer wieder vernachlässigt. Requirements Engineering ist die Disziplin, die sich damit beschäftigt, wann und wie diese Fragen zu beantworten sind.

In diesem Kapitel geht es darum, zu verstehen, was Requirements Engineering genau ist und warum es hilfreich sein kann. Hierzu bietet es sich an, als Erstes ein paar zentrale Begriffe zu klären und die Aufgaben besser zu kennen.

Warum uns Requirements Engineering weiterhelfen kann

Viele typische Projekte laufen so: Der Kunde – das kann auch ein interner Kunde sein, etwa eine Fachabteilung – beschreibt zu Beginn, wie er sich das neu- oder weiterzuentwickelnde System vorstellt. Bei Softwaresystemen geschieht das meist, indem er die gewünschten Bildschirmmasken beschreibt oder welche Informationen er einzugeben wünscht, um etwa Aufträge zu erfassen, und wie er diese Informationen beispielsweise in Reports aufbereitet sehen möchte. Diese Beschreibungen sind dabei schon sehr detailliert und es ist von »Knöpfen«, »Auswahlfeldern« und »Flags« die Rede.

Diese Beschreibungen der Wünsche der Kunden werden dann mit ein paar Erläuterungen an das Projektteam übergeben, welches das Ganze umsetzt. Zwischendurch wird es ein paar Rückfragen an den Kunden geben und gegen Ende wird es zeitlich eng, weshalb einige der

32 TEIL I Requirements Engineering verstehen

Kundenwünsche zusammengestrichen werden. Wenn am Ende das fertige System vorgestellt wird, geht dann der große Streit los, ob das Projektteam wirklich das umgesetzt hat, was der Kunde zu Beginn beschrieben hatte. Wie immer der Streit ausgeht, es ist schon klar, es braucht ein Anschlussprojekt, um das System so anzupassen, dass die Nutzer vernünftig damit arbeiten können.

Hier stellt sich die Frage: müssen Projekte so laufen? Natürlich nicht! Schauen wir uns an, was in dem beschriebenen Szenario alles falsch gelaufen ist:

- ✓ Niemand hat sich Gedanken gemacht, welchen Nutzen das System erzielen soll.
Oder – genauso schlecht – jeder hat sich seine eigenen Gedanken gemacht, aber es gibt kein gemeinsames Ziel.
- ✓ Die Kunden haben beschrieben, wie das System aussehen soll, aber nicht warum.
- ✓ Kunde und Projektteam arbeiten mehr nebeneinander her als miteinander. Jeder verlässt sich darauf, dass der andere schon weiß, was er tut, und konzentriert sich auf *seine* Aufgaben.

Durch das erstgenannte Problem gibt es beim Entwicklungsvorhaben keine klaren Schwerpunkte und keine Zielrichtung. Jeder wirft seine Wünsche in den Anforderungstopf, aus dem sich das Projektteam so lange bedient, bis der Topf leer oder die verfügbare Zeit abgelaufen ist – wobei Letzteres leider der Normalfall ist.



Am Beginn eines jeden Entwicklungsvorhaben sollte das Ziel klar herausgearbeitet und insbesondere durch den zu erwartenden Nutzen beschrieben werden.

Das zweite Problem offenbart, dass wir gewohnt sind, sehr schnell über Lösungen nachzudenken. Lösungen sind ja durchaus positiv, aber um zu wirklich guten Lösungen zu kommen, muss zuerst das Problem richtig verstanden werden.



Um die Anforderungen des Kunden zu verstehen, sollte immer die Frage im Vordergrund stehen, *was* das System leisten soll, um *welche Aufgaben* zu lösen. Es geht nicht darum, *wie* das System etwas leisten soll – im Sinne einer Lösung.

Wenn der Kunde also seine Wünsche in Form von Bildschirmmaskenbeschreibungen äußert, dann ist eben nicht klar, *was* das System leisten soll und wozu, sondern nur *wie* das System etwas machen soll.

Ein anderes Problem, das in dem beschriebenen Szenario zutage tritt: Der Kunde und das Projektteam haben es nicht als gemeinsame Aufgabe betrachtet, wirklich zu verstehen, was der Kunde braucht, um einen Nutzen zu haben, und was unter den gegebenen Rahmenbedingungen machbar ist.



Um die Anforderungen des Kunden zu verstehen, muss das Projektteam eng mit dem Kunden zusammenarbeiten. Es reicht nicht aus, wenn der Kunde seine Wünsche einfach aufschreibt und das Dokument dem Projektteam überreicht.



Schlechte Anforderungen bremsen Innovationen aus

Legen die Anforderungen genau fest, wie das System sich verhalten soll und wie es auszusehen hat, bleibt dem Umsetzungsteam wenig Spielraum. Das bedeutet, dass dessen Kenntnisse und Kreativität für die Entwicklung guter Systeme nicht genutzt werden. Ob das System am Ende dem Kunden tatsächlich einen Nutzen bringt, kann das Umsetzungsteam schließlich nicht beurteilen, wenn es den Nutzen nicht kennt und nicht weiß, welche Aufgaben damit gelöst werden sollen.

Wenn die Anforderungen dagegen beschreiben, was das System im Einzelnen leisten soll und wie das übergeordnete Ziel aussieht, dann kann das Umsetzungsteam überlegen, wie nach ihrer Einschätzung das System beschaffen sein sollte, um den gewünschten Nutzen zu erbringen. Ein qualifiziertes Umsetzungsteam kennt die aktuellen technischen Möglichkeiten am besten. Selbstverständlich sollte es seine Umsetzungsideen erst mal mit dem Kunden diskutieren, bevor es sie umsetzt, um sicherzugehen, dass es einerseits die Probleme des Kunden richtig verstanden hat, und andererseits zu prüfen, ob die Lösungsideen auch passen.

Dieses Vorgehen eröffnet den Spielraum, um auch über ganz neue Lösungsideen nachzudenken, auf die der Kunde selbst nie gekommen wäre, und bildet die Basis, um innovative Ideen zu entwickeln. Also bremsen Sie als Requirements Engineer bitte keine innovativen Lösungen aus, indem Sie mit den Anforderungen bereits die Lösung für die Aufgaben und Probleme des Kunden vorgeben, sondern beschreiben Sie mit den Anforderungen, welche Aufgaben und Probleme er mit dem System lösen will.

Gutes Requirements Engineering trägt wesentlich dazu bei, all diese Probleme zu vermeiden, denn es bietet die Methoden und Techniken, um besser zu verstehen, was der Kunde wirklich braucht.



»Der Kunde hat immer Recht« heißt es. Aber nehmen Sie das nicht zu wortwörtlich, denn der Kunde ist sich häufig nicht im Klaren, was er wirklich braucht. Hinterfragen Sie also, was er als Anforderung beschreibt, und nutzen Sie die Methoden des Requirements Engineering, um herauszufinden, was er wirklich braucht.

Am Ende sollte das Requirements Engineering dazu beitragen, folgende Fragen zu beantworten:

- ✓ Was ist der erwartete Nutzen des Systems?
- ✓ Wer ist der Kunde? Wer sind die Nutzer?
- ✓ Welche Aufgaben wollen sie mit dem System lösen?
- ✓ Welche Fähigkeiten des Systems werden dafür benötigt?



Die Ausdrücke »Requirements Engineering« und »Anforderungsmanagement« beziehungsweise »Requirements Management« werden nicht immer einheitlich gebraucht. In diesem Buch ist Requirements Engineering der Oberbegriff und Requirements Management ein Teil davon. Dies entspricht einer im deutschsprachigen Raum verbreiteten Sichtweise, die auch vom International Requirements Engineering Board (IREB) geteilt wird.

Allerdings werden auch häufig Requirements Engineering und Anforderungsmanagement als unterschiedliche Disziplinen angesehen, denn in dieser Sichtweise umfasst das Requirements Engineering nur die Ermittlung und Dokumentation von Anforderungen, nicht aber das Organisieren.

Um die Verwirrung komplett zu machen: Es gibt auch die Lesart, dass Requirements Management der Oberbegriff ist, der alle Tätigkeiten rund um Anforderungen umfasst. Dieses Begriffsverständnis ist vor allem im angelsächsischen, aber eben auch im deutschsprachigen Raum anzutreffen.

Fragen Sie also im Zweifelsfalle einfach nach, was Ihr Gegenüber meint, wenn er einen dieser Begriffe benutzt.

Aufgaben im Requirements Engineering

Grob gesprochen bestehen die Aufgaben im Requirements Engineering aus vier Haupttätigkeiten:

- ✓ Anforderungen ermitteln,
- ✓ Anforderungen kommunizieren und festhalten,
- ✓ Anforderungen validieren,
- ✓ Anforderungen organisieren.

Beim ersten Punkt geht es darum, erst mal herauszubekommen, welche Anforderungen die Kunden, Nutzer und andere Stakeholder sich so vorstellen und dann gemeinsam mit ihnen zu erarbeiten, was sie tatsächlich brauchen, um den angestrebten Nutzen zu erreichen.

Beim zweiten Punkt geht es darum, die so erarbeiteten Anforderungen an diejenigen zu kommunizieren, die sie für ihre Arbeit benötigen, beispielsweise die Entwickler, Tester und Projektplaner. Ob wir das mit umfangreichen Anforderungsdokumenten bewerkstelligen, mit Anforderungsmodellen oder überwiegend mündlich, hängt von der Vorgehensweise ab; also ob wir klassisch oder agil arbeiten. In irgendeiner Form sollten die Anforderungen aber immer festgehalten werden, allein schon, um deren Umsetzung planen zu können, aber auch als Basis der Kommunikation.



Wovon die Anforderungen handeln: das System

Wenn es darum ging, wovon die Anforderungen handeln, war bisher hier einfach von Systemen die Rede. Das ist natürlich ein recht allgemeiner Begriff, denn es gibt sehr unterschiedliche Arten von Systemen wie zum Beispiel

- ✓ Softwaresysteme,
- ✓ Hardwaresysteme,
- ✓ Softwareservices,
- ✓ verteilte Systeme,
- ✓ Gesamt- und Teilsysteme.

Diese Begriffe sind nicht gegeneinander abgegrenzt, vielmehr kann ein Gesamtsystem aus sämtlichen der genannten Arten von Systemen bestehen. Ein Mobiltelefon beispielsweise besteht aus Hardware, auf der unterschiedliche Software läuft, die wiederum auf unterschiedliche Services zurückgreifen, die auf mehreren Servern laufen, weshalb man das Ganze auch als verteiltes System betrachten kann.

In der agilen Welt ist auch häufig von *Produkten* die Rede. So können wir das Mobiltelefon als Kombination von Hard- und Software betrachten, was der Produktsicht des Geräteherstellers entspricht, während die Anbieter von Apps die Software für diese Geräte als Produkt verkaufen.

Der System-Begriff ist also sehr vielfältig. Da aber das Requirements Engineering auch sehr vielfältig ist und für alle der genannten Systeme, von ganzen Produkten bis hin zu den Teilsystemen, Anforderungen formuliert, wird hier einfach von Systemen die Rede sein; so hält es auch das International Requirements Engineering Board (IREB).

Da Anforderungen also für sehr unterschiedliche Systeme formuliert werden können, ist es wichtig, auch klar zu machen, um welches System es genau geht. Hierzu dienen Werkzeuge wie der Systemkontext, der im Abschnitt »Den Kontext des Systems verstehen« in Kapitel 8 vorgestellt wird.

Diese vier Grundaufgaben werden selten getrennt voneinander durchgeführt. Gerade die ersten beiden sind üblicherweise ineinander verwoben: Wenn wir zum Beispiel mit Stakeholdern über deren Anforderungen sprechen, ist es häufig hilfreich, uns etwas aufzuschreiben oder Abläufe in irgendeiner Form aufzuzeichnen. Das hilft uns in der Kommunikation, wenn wir gemeinsam mit dem Stakeholder darauf schauen, es hilft uns zu einem klareren Verständnis und es kann anschließend in eine Dokumentation einfließen.

36 TEIL I Requirements Engineering verstehen

(Und es ist jetzt völlig egal, ob Sie hier beispielsweise an User Stories oder etwa an UML-Aktivitätsdiagramme denken.)

Der dritte Punkt, validieren, dreht sich darum, sich bei den Stakeholdern, vor allem Kunden und Nutzern, zu versichern, dass die Anforderungen auch wirklich die richtigen sind. Sie müssen also für jede Anforderung folgende Fragen beantworten:

- ✓ Benötigt überhaupt jemand die Anforderung?
- ✓ Ist die Anforderung richtig beschrieben?
- ✓ Entspricht die Beschreibung der Anforderung den bestehenden Vorgaben?
- ✓ Passt die Anforderung zu den übrigen oder gibt es Widersprüche?
- ✓ Sind die Anforderungen vollständig?
- ✓ Sind die Anforderungen im gegebenen Rahmen überhaupt zu erfüllen?

Diese Fragen können Sie auf sehr unterschiedliche Weise beantworten: Sie können beispielsweise umfangreiche Anforderungsdokumente einem Review unterziehen oder eng mit dem Kunden zusammenarbeiten und ihm Zwischenergebnisse in Form von lauffähiger Software präsentieren. In diesem Fall wird das Validieren in das Ermitteln, Kommunizieren und Festhalten integriert.

Beim vierten Punkt, dem Organisieren der Anforderungen, geht es um so unterschiedliche Aufgaben wie

- ✓ Änderungen von Anforderungen,
- ✓ Priorisierung,
- ✓ Verfolgbarkeit (Traceability) und
- ✓ Versionierung.

Für all diese Punkte werden sehr unterschiedliche Techniken eingesetzt, die Sie in dem Teil »IV Requirements Management« kennenlernen werden.

Die Reihenfolge der eingangs genannten vier Haupttätigkeiten ist durch das Requirements Engineering nicht starr festgelegt, sondern hängt von der Vorgehensweise und den eingesetzten Methoden ab. Gerade bei agilen Ansätzen werden die Tätigkeiten parallel ausgeführt.

Wer das Requirements Engineering macht

Wenn wir an die Aufgaben des Requirements Engineering denken, stellt sich auch die Frage, wer sie übernehmen soll. Eine naheliegende Möglichkeit: jemand im Projekt ist dafür und nur dafür zuständig. Das heißt, wir haben einen Requirements Engineer. In der Praxis ist diese Rolle mittlerweile häufig anzutreffen, aber das bedeutet natürlich nicht, dass es nicht

auch anderen Möglichkeiten gibt und die Aufgaben von anderen Rollen miterledigt oder auf mehrere Rollen verteilt werden. Verbreitet ist zum Beispiel, dass die Entwickler oder die Projektleiter sich nebenbei um das Requirements Engineering kümmern.

Der Requirements Engineer

Gibt es eine oder mehrere Personen, die sich ausschließlich um die Anforderungen kümmern und dafür auch ausgebildet sind, können diese sich ganz darauf konzentrieren, mit den einschlägigen Methoden die Anforderungen handwerklich sauber zu ermitteln und zu organisieren. Darüber hinaus wird in dieser Konstellation der Aufwand für das Requirements Engineering seltener unterschätzt.

Der Requirements Engineer übernimmt im Entwicklungsvorhaben eine Schnittstellenfunktion zwischen dem Kunden und den übrigen Rollen, welche die Anforderungen benötigen, wie die Entwickler, die Tester oder die Projektmanager. Der Requirements-Engineering-Spezialist agiert hier gewissermaßen als Anwalt des Kunden und vertritt dessen Bedürfnisse.

Dafür sollte er auch einige Fähigkeiten mitbringen:

- ✓ Kommunikationsfähigkeit,
- ✓ analytisches Denken,
- ✓ Abstraktionsvermögen,
- ✓ Einfühlungsvermögen,
- ✓ Moderationsfähigkeit,
- ✓ Teamfähigkeit,
- ✓ Konfliktlösungsfähigkeit.

Wer sonst noch das Requirements Engineering macht

Häufig wird das Requirements Engineering von anderen Rollen miterledigt, beispielsweise von

- ✓ den Entwicklern,
- ✓ dem Product Owner,
- ✓ dem Projektleiter,
- ✓ den zukünftigen Nutzern oder Kunden.

Das kann durchaus auch Vorteile haben, denn dadurch erfahren diejenigen, die mit den Anforderungen arbeiten müssen, was vor allem für die Entwickler gilt, von diesen Anforderungen aus erster Hand.

Praktisch bringt es aber häufig die Herausforderung mit sich, dass die Leute, die diese Rollen innehaben, den Schwerpunkt ihrer Arbeit in den rollenspezifischen Aufgaben sehen und daher für das Requirements Engineering über

- ✓ zu wenig Zeit und
- ✓ nicht das richtige Handwerkszeug verfügen.

Zu den spezifischen Herausforderungen der einzelnen Rollen können Sie im Abschnitt »Wer nimmt die Anforderungen auf?« in Kapitel 3 mehr erfahren.

Viele Arten von Anforderungen

Wenn Sie die Kunden nach ihren Anforderungen fragen, erklären diese entweder, was das System machen soll. In diesem Fall wird von funktionalen Anforderungen gesprochen. Oder sie erzählen – gerade wenn es darum geht, ein altes durch ein neues System zu ersetzen –, wie das System arbeiten soll, also schnell, zuverlässig, sicher und Ähnliches. Solche Arten von Anforderungen werden Nichtfunktionale oder Qualitätsanforderungen genannt.

Darüber hinaus wird häufig noch eine dritte Art von Anforderung unterschieden: die Rand- oder Rahmenbedingung. Allerdings geht es hier nicht nur um Anforderungen im eigentlichen Sinne, also etwas, was das Projektteam irgendwie umsetzen muss, das heißt eine Eigenschaft oder Fähigkeit des Systems. Bei den Randbedingungen geht es auch um Einschränkungen für die Realisierung des Systems wie: es soll eine bestimmte Datenbank verwendet werden oder das Ganze darf nicht teurer als eine bestimmte Summe werden.

Funktionale Anforderungen

Wenn Sie im Requirements Engineering beschreiben, was das System leisten soll, dann beschreiben Sie die funktionalen Anforderungen an das System. Funktionale Anforderungen können Sie als eine Abfolge von Schritten spezifizieren, die das System ausführen soll. Es gibt auch noch weitere Möglichkeiten, die in bestimmten Fällen auch vorzuziehen sind, aber um zu entscheiden, ob Sie es mit einer funktionalen Anforderung zu tun haben, überlegen Sie sich einfach, ob die Anforderungen als eine Abfolge von Schritten darstellbar sind.

Wenn Sie eine funktionale Anforderung beschreiben, dann werden Sie in der Regel nicht nur einen, sondern mehrere Abläufe zu einer Anforderung beschreiben, da es meist verschiedene Varianten gibt, wie das System etwas erledigen soll, und zudem auch die Fälle zu berücksichtigen sind, in denen etwas schief läuft. Neben den Ablaufschritten können Sie auch die Informationen nennen, die das System benötigt, sowie die Ergebnisse, die dabei herauskommen. Darüber hinaus kann es auch hilfreich sein, Voraussetzungen zu beschreiben.



Um funktionale Anforderungen zu verstehen und zu beschreiben, ist es hilfreich, wenn Sie den Blick vom System zum Benutzer wenden und aus dessen Sicht schildern, wie die Benutzer das System verwenden wollen. Techniken hierzu sind Anwendungsfälle und User Stories.

Nichtfunktionale Anforderungen

Kunden und Nutzer haben nicht nur Erwartungen, was das System leisten soll, sondern auch darüber, *wie* es etwas leisten soll. Dazu gehören unter anderem solche Fragen:

- ✓ Wie schnell soll es eine bestimmte Aufgabe erledigen?
- ✓ Wie sicher soll es hinsichtlich Missbrauch sein?
- ✓ Wie einfach soll die Erledigung einer Aufgabe für die Nutzer sein?
- ✓ Wie einfach soll es für die Entwickler sein, neue Anforderungen in das System zu integrieren?
- ✓ Wie aufwendig darf es für die Entwickler sein, das System auf ein anderes Betriebssystem zu portieren?

Das Problem mit den nichtfunktionalen Anforderungen: Die Kunden und späteren Anwender sind sich häufig selbst nicht darüber im Klaren, welche Erwartungen sie eigentlich haben. Die werden ihnen erst dann klar, wenn sie mit dem System arbeiten und es sich nicht so verhält, wie sie es sich wünschen. Oder wenn die Integration neuer Anforderungen viel komplizierter und teurer wird, als sie es sich vorstellen. Einfacher ist es daher bezüglich der nichtfunktionalen Anforderungen, wenn es darum geht, ein bestehendes durch ein neues System zu ersetzen. Dann gibt es konkrete Erwartungen, die sich daraus ableiten, was in dem Altsystem nicht funktioniert.



Nichtfunktionale Anforderungen werden häufig übersehen und nur unvollständig erkannt. Viele dieser Anforderungen können die Entwickler aber nur umsetzen, indem sie das System entsprechend strukturieren, also die Systemarchitektur passend gestalten. Dies nachträglich zu machen, ist sehr aufwendig und führt daher zu unbefriedigenden Kompromissen. Warten Sie also nicht ab, bis die Nutzer am fertigen System merken, welche nichtfunktionalen Anforderungen sie eigentlich haben, sondern kümmern Sie sich frühzeitig darum.

Um Ihnen einen Überblick darüber zu geben, was möglicherweise für Ihr System an nichtfunktionale Anforderungen infrage kommt, seien in Abbildung 1.1 die Kategorien des einschlägigen ISO-Standards 25010 zur Softwarequalität genannt.

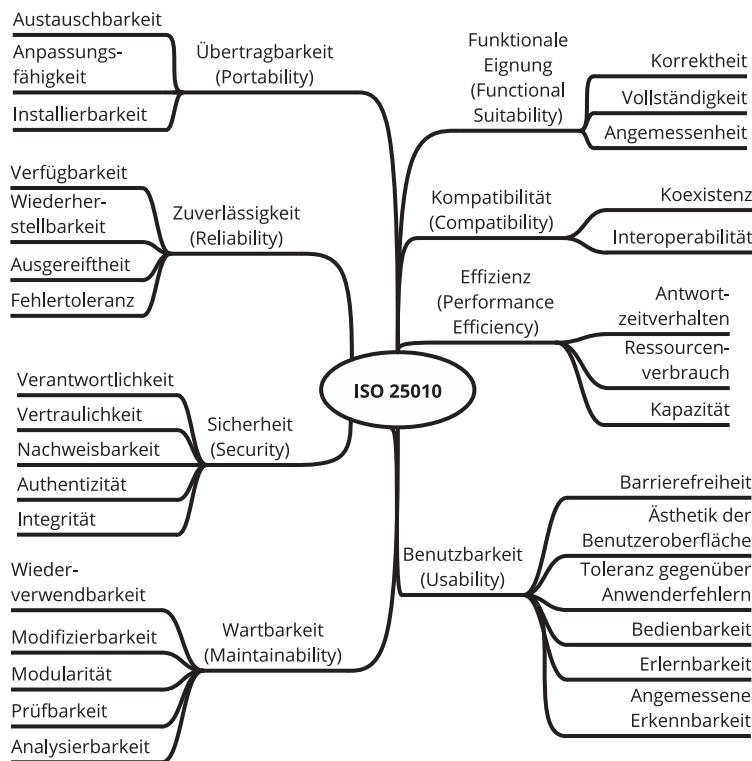


Abbildung 1.1: Kategorien nichtfunktionaler Anforderungen nach ISO 25010

Randbedingungen

Wenn Sie den Auftrag erhalten, ein System zu entwickeln, heißt das selten, dass jetzt alles erlaubt ist, um die gegebenen Anforderungen umzusetzen. Bestimmt gibt es Erwartungen, was das System höchstens kosten darf oder bis wann das Ganze in Betrieb gehen soll. Häufig wird Ihnen auch die Vorgehensweise vorgegeben, also ob zum Beispiel agil oder eher klassisch zu entwickeln ist.

Durch Vorgaben dieser Art sind Sie nicht mehr ganz so frei, was die Umsetzung der Anforderungen betrifft, denn alle Varianten, die teurer werden oder länger brauchen, scheiden schon mal aus. Die Vorgaben schränken also die Vielfalt der möglichen Lösungen stark ein. Genau genommen nicht nur der Lösungen, sondern auch der Anforderungen, denn Anforderungen, zu denen es keine Lösungen gibt, die zu den Vorgaben passen, lassen sich natürlich auch nicht umsetzen.

Solche Einschränkungen für mögliche Anforderungen und Lösungen werden *Randbedingungen* oder *Rahmenbedingungen* genannt.

Die genannten Beispiele für Randbedingungen schränken den Raum der Möglichkeiten in organisatorischer Hinsicht ein. Daneben gibt es aber auch technische Randbedingungen,

wenn etwa vorgegeben wird, dass eine bestimmte Programmiersprache, eine bestimmte Datenbank oder ein bestimmtes Framework zu verwenden ist.



Manches steht schon zu Beginn eines Projekts als Randbedingung fest, zum Beispiel welches Datenbanksystem erlaubt ist. Möglicherweise darf das Projektteam dies auch nach eigener Einschätzung frei entscheiden, was die Anforderungen am besten erfüllt. Je nachdem spielt die Festlegung auf ein Datenbanksystem also die Rolle einer Randbedingung oder eines Lösungsansatzes.

Der Unterschied liegt darin, ob die Festlegung schon vor dem Projekt erfolgt, und daher aus Gründen, die außerhalb des Projekts zu suchen sind; hier etwa: wir haben bereits dieses Datenbanksystem bei uns und kennen uns daher damit schon gut aus. Im anderen Fall erfolgt die Festlegung erst auf dem Weg zur Lösung und ergibt sich aus den Anforderungen.

Daneben können die Randbedingungen auch echte Anforderungen enthalten, zum Beispiel Barrierefreiheit. Diese Anforderung kann sowohl die Rolle einer nichtfunktionalen Anforderung spielen als auch die einer Randbedingung, beispielsweise wenn dies aus einer verbindlichen Unternehmensvorgabe folgt.



Man kann sich darüber streiten, ob Randbedingungen überhaupt Anforderungen sind, aber zumindest das International Requirements Engineering Board (IREB) sieht es so. Mehr zu dieser Frage im Kasten »Sind Randbedingungen Anforderungen?« in Kapitel 11.

Abstraktionsstufen von Anforderungen

Eine ganz andere Art der Unterscheidung von Anforderungen geht von einer Hierarchie der Perspektiven aus, die von der geschäftlichen Sicht über die Stakeholder- und die Nutzer- bis hin zur Systemsicht geht. So etwas findet sich beispielsweise in dem ISO-Standard 29148 »Systems and software engineering – Life cycle processes – Requirements engineering«. Dort werden folgende Abstraktionsstufen von Anforderungen definiert.

- ✓ **Business Requirements:** Geschäftliche Anforderungen, die eine Organisation mit dem System erfüllen will. Verortet das zu entwickelnde System (oder die Systeme) im Geschäftsmodell und beschreibt den Nutzen aus geschäftlicher Sicht.
- ✓ **Stakeholder Requirements:** Anforderungen an das System aus Sicht der Stakeholder. Hier wird die gewünschte Unterstützung der Geschäftsprozesse durch das System konkretisiert.
- ✓ **User Requirements:** Beschreiben die vorgesehenen Nutzungen des Systems in ihrem Nutzungskontext, aber auch die zugehörigen nichtfunktionalen Anforderungen. Die User Requirements sind Teil der Stakeholder Requirements.
- ✓ **System Requirements:** Definiert den Systemkontext, die funktionalen und nichtfunktionalen Anforderungen für das gesamte System, einschließlich der Verifikation.

- ✓ **Software Requirements:** Funktionen und nichtfunktionalen Anforderungen für ein Softwaresystem, einschließlich der Verifikation. Ist die Software Teil eines größeren Systems, werden die Rolle der Software und die Schnittstellen zu dem System beschrieben.

Das International Requirements Engineering Board (IREB) nimmt diese Unterscheidung der ISO auf und fügt ihr noch eine weitere Kategorie zu:

- ✓ **Domain Requirements:** Anforderungen an die organisatorische oder technische Umgebung des Systems.

Diese sehr feingranulare Unterscheidung von Anforderungen schlägt sich in der ISO 29148 auch in der Dokumentation von Anforderungen nieder, die eine Business Requirements Specification, eine Stakeholder Requirements Specification, eine System Requirements Specification und eine Software Requirements Specification vorsieht. Mehr zu den Arten der Dokumentation können Sie im Abschnitt »Gliederung von Anforderungsdokumenten« in Kapitel 13 nachlesen.

Dies ist natürlich mit einem beträchtlichen Aufwand verbunden und in der Praxis selten anzutreffen. Allenfalls bei sehr komplexen Systemen wie Flugzeugen kann das in dieser Ausführlichkeit angemessen sein.



Die Unterscheidung von funktionalen und nichtfunktionalen Anforderungen sowie Randbedingungen auf der einen Seite und die hier vorgestellten Abstraktionsstufen schließen sich nicht gegenseitig aus, sondern verlaufen gewissermaßen quer zueinander. In den verschiedenen hierarchischen Kategorien finden sich sowohl funktionale als auch nichtfunktionale Anforderungen.

Möglichkeiten der Zertifizierung

Wenn Sie Ihr Wissen im Requirements Engineering nachweisen wollen, können Sie eines der folgenden Zertifikate erwerben:

- ✓ *Certified Professional for Requirements Engineering (CPRE)* vom International Requirements Engineering Board (IREB), dies gibt es für Einsteiger, Fortgeschrittene und Experten.
- ✓ *Certified Business Analysis Professional (CBAP)* und weitere Zertifikate des International Institute of Business Analyses (IIBA)
- ✓ *PMI Professional in Business Analysis (PMI-PBA)* vom Projektmanagement Institute (PMI)

Im Folgenden werden diese genauer beschrieben.

Zertifikate des IREB

Im deutschsprachigen Raum sind die Zertifizierungen des International Requirements Engineering Board (IREB) am verbreitetsten, insbesondere das Grundlagenzertifikat *Certified Professional for Requirements Engineering – Foundation Level* (CPRE FL).

Den inhaltliche Anspruch, den das IREB mit dem Foundation Level verbindet, besteht darin, die grundlegenden Techniken des Requirements Engineering unabhängig von der Vorgehensweise nachzuweisen; also unabhängig davon, ob Sie klassisch wasserfallmäßig oder agil vorgehen. Der Schwerpunkt liegt dabei eher auf den klassischen Techniken, aber es werden dort auch spezifisch agile Methoden behandelt. Vieles können Sie auch in beiden Vorgehensweisen einsetzen.



Alles, was Sie an Wissen über Requirements Engineering benötigen, um das Zertifikat CPRE FL zu erwerben, finden Sie hier im Buch.

Für das Foundation Level müssen Sie keine Voraussetzungen erfüllen und lediglich eine Multiple-Choice-Prüfung ablegen, die Sie in zahlreichen Sprachen durchführen können, unter anderem in Deutsch und Englisch.

Neben dem Foundation Level bietet das IREB aber auch unterschiedliche Zertifikate im so-nannten Advanced- beziehungsweise Expert-Level an.

Mit den vier Varianten des Advanced-Zertifikats können Sie folgende Themen vertiefen:

- ✓ **Requirements Elicitation:** Methoden und Techniken, um Anforderungen zu ermitteln.
- ✓ **Requirements Modeling:** Grafische Beschreibung funktionaler Anforderungen.
- ✓ **Requirements Management:** Methoden und Techniken, um Anforderungen zu verwalten.
- ✓ **RE@Agile:** Die Einbettung klassischer und agiler Techniken des Requirements Engineering in ein agiles Vorgehen.

Die Hürde für die Advanced-Zertifikate ist naturgemäß etwas höher als beim Foundation Level: Zum einen müssen Sie bereits über das Foundation-Zertifikat verfügen. Zum anderen erwartet das IREB neben dem Multiple-Choice-Test auch noch eine Hausarbeit von Ihnen.

Richtig schwierig wird es dann beim Expert-Level: Als Voraussetzung verlangt das IREB von Ihnen drei Advanced-Zertifikate (oder vergleichbare Nachweise), drei Jahre Praxis im Requirements Engineering sowie Erfahrungen als Trainer oder Coach. In einer schriftlichen Bewerbung müssen Sie belegen, dass Sie die genannten Voraussetzungen erfüllen. Anschließend ist hier eine Hausarbeit fällig, die Sie später in einer mündlichen Prüfung vorstellen und verteidigen. In der mündlichen Prüfung werden Sie aber auch noch allgemein zum Requirements Engineering befragt – und das alles auf Englisch.

44 TEIL I Requirements Engineering verstehen

Außerhalb der Foundation-Advanced-Expert-Einordnung bietet das IREB auch noch ein Zertifikat speziell für das agile Requirements Engineering an, das sogenannte RE@Agile-Primer-Zertifikat. Hier geht es um einschlägige agile Requirements Engineering-Techniken wie User Stories oder Product Backlog und inwieweit das mit den eher klassischen Techniken kombiniert werden kann. Vom Umfang des Stoffes und der Prüfung ist dieses Zertifikat noch unterhalb des Foundation Levels einzuordnen.

Überraschenderweise ist das RE@Agile-Primer-Zertifikat nicht die Voraussetzung für das RE@Agile-Advanced-Zertifikat – hierfür benötigen Sie das Foundation-Zertifikat.

Hier noch mal der Überblick über alle Zertifikate des International Requirements Engineering Board (IREB):

- ✓ **Foundation Level:** Hier müssen Sie nachweisen, dass Sie die grundlegenden Techniken des Requirements Engineering kennen und einige davon anwenden können.
- ✓ **Advanced Level:** Um Ihre fortgeschrittenen Kenntnisse nachzuweisen, bietet das IREB vier Zertifikate an: *Elicitation, Requirements Modeling, Requirements Management* und *RE@Agile*.
- ✓ **Expert Level:** Damit können Sie Ihr Expertenwissen auf höchstem Niveau nachweisen.
- ✓ **RE@Agile Primer:** Hier geht es um die Grundlagen des Requirements Engineerings im agilen Vorgehen.

Praktisch für Sie: die Zertifikate des IREB haben eine unbegrenzte Dauer und müssen nicht erneuert werden.

Zertifikate des IIBA

Vom inhaltlichen Anspruch sehr umfassend sind die Zertifikate des International Institute of Business Analyses (IIBA). Dies liegt auch daran, dass das Konzept der Business Analysis in der Sicht des IIBA umfassender ist als das des Requirements Engineering: Während dieses erst mit dem Beginn eines Projekts einsetzt und mit diesem auch beendet wird, geht es bei der Business Analysis auch darum, wie zuvor überhaupt der Bedarf festgestellt und anschließend der Nutzen überprüft wird.

Das IIBA bietet ebenfalls ein dreistufiges Zertifizierungsverfahren an:

- ✓ **Entry Certificate in Business Analysis (ECBA):** Wie der Name schon sagt, das Einsteigerzertifikat vom IIBA, mit dem Sie Ihr Basiswissen in den vom IIBA vorgesehenen Bereichen der Business Analysis nachweisen.
- ✓ **Certificate of Capability in Business Analysis (CCBA):** Hiermit weisen Sie nach, dass Sie in der Lage sind, die Methoden der Business Analysis selbstständig anzuwenden.

- ✓ **Certified Business Analysis Professional (CBAP):** Durch dieses Zertifikat können Sie zeigen, dass Sie ein tiefes Verständnis der Methoden des Business Analysis haben, das Sie selbstständig auf unterschiedlichste Situationen anwenden können.

Entsprechend dem recht hohen Anspruch des IIBA benötigen Sie auch ein umfangreiches Wissen an Business-Analysis- beziehungsweise Requirements-Engineering-Methoden, um die Prüfung zum Zertifikat zu bestehen. Darüber hinaus müssen Sie auch noch einige formale Voraussetzungen erfüllen:

- ✓ Sie müssen für die letzten vier Jahren mindestens einundzwanzig Stunden, für den CPAB sogar 35 Stunden Weiterbildung in der Business Analysis nachweisen,
- ✓ Sie haben umfangreiche praktische Erfahrungen in der Business Analysis gesammelt (nur CCBA und CBAP) und
- ✓ Sie können Führungskräfte oder Kunden als Referenzen angeben (nur CCBA und CBAP).

Außerdem gibt es noch ein Zertifikat für die Business Analysis im agilen Umfeld, die **Agile Analysis Certification (AAC)**. Hierfür müssen Sie keine formalen Voraussetzungen erfüllen, aber es werden zwei bis drei Jahre einschlägige Erfahrungen erwartet. Von daher entspricht dies in etwa dem Niveau des CCBA.

Das IIBA hat seinen Hauptsitz in Kanada, ist aber weltweit verbreitet und auch in Deutschland, Österreich und der Schweiz vertreten. Entsprechend der Herkunft des IIBA finden die Prüfungen auf Englisch statt. Die Zertifikate sind global gesehen bekannter als die des IREB, aber im deutschsprachigen Raum hat das IREB beim Bekanntheitsgrad klar die Nase vorn.

Die fortgeschrittenen Zertifikate des IIBA, also CCBA und CBAP, müssen Sie regelmäßig erneuern, indem Sie nachweisen, dass Sie sich fortwährend weiterbilden, beispielsweise durch den Besuch von Seminaren oder Konferenzen.

PMI Professional in Business Analysis (PMI-PBA)

Eine weitere Möglichkeit der Zertifizierung bietet das Projekt Management Institut (PMI) an. Das PMI ist in den USA beheimatet, aber weltweit verbreitet. Bekannt ist es vor allem für sein Zertifikat »Project Management Professional (PMP)«, das zweifellos zu den führenden im Bereich Projektmanagement gehört. Mit der Zeit hat das PMI weitere Zertifizierungsmöglichkeiten geschaffen, darunter auch **PMI Professional in Business Analysis (PMI-PBA)**. Mehrere Stufen der Zertifizierung sind hier – anders als beim IREB oder IIBA – nicht vorgesehen, es geht hier recht anspruchsvoll los. Der PMI-PBA ist praktisch das Konkurrenzprodukt des PMI zum CBAP des IIBA.

Entsprechend der Herkunft vom Projekt Management Institut wird hier die Business Analysis vor allem als Teil eines Projekts betrachtet und kommt damit der IREB-Sicht näher als der IIBA-Sicht, welche die Business Analysis auch projektübergreifend betrachtet.

46 TEIL I Requirements Engineering verstehen

Wie das IIBA formuliert auch das PMI ein paar Voraussetzungen, die erfüllt sein müssen, bevor Sie überhaupt zur Prüfung zugelassen werden:

- ✓ Sie müssen belegen, dass Sie in den letzten vier Jahren Trainings zur Business Analysis im Umfang von mindestens fünfunddreißig Stunden besucht haben, und
- ✓ Sie haben bereits umfangreiche praktische Erfahrungen in der Business Analysis gesammelt.

Auch die Prüfung zum PMI-PBA findet auf Englisch statt und auch hier müssen Sie das Zertifikat regelmäßig erneuern, indem Sie nachweisen, dass Sie sich fortwährend weiterbilden, beispielsweise durch den Besuch von Seminar oder Konferenzen.