

“Die Wahrscheinlichkeit des Eintreffens ist 100%!“ – Risiken die Sie ohne formales Anforderungsmanagement eingehen

Robert Stevenson, HOOD Group, München, Deutschland

20. August 2009

Einführung

Jeder, der ein System oder Produkt erstellt, macht Anforderungsmanagement, selbst wenn er sich darüber nicht im Klaren ist. Zum Beispiel handelt es sich um eine Aktivität des Anforderungsmanagements, wenn Kunden darüber befragt werden, was sie über ein bestimmtes Produkt denken. Manche von uns verwenden formalisierte Prozeduren und Praktiken, die wir Requirements Management und Engineering (RM&E) nennen. Indem wir RM&E vom Unterbewussten ins Bewusste übertragen, machen wir die ersten Schritte um Risiken zu managen, die normalerweise während der Produkt- und Systementwicklung auftreten.

1 Ziele und Inhalt

Der vorliegende Artikel zielt darauf ab zu zeigen, wie Standardpraktiken des RM&E dazu verwendet werden können, erste Schritte in Richtung der Beherrschung von Risiken zu machen, die bei der Entstehung eines Produktes oder Systems auftreten. Er wendet sich im Wesentlichen an Risikomanager, die eine Einführung in RM&E benötigen. Natürlich sind Anforderungen nur eine Risikoquelle von vielen, die Risikomanager behandeln müssen. Manche Risikomanager sind überhaupt nicht direkt in den Werdegang eines Produktes oder Systems involviert. Und selbst solche, die es doch sind, müssen viele Risikoquellen in

Betracht ziehen, zum Beispiel Einflüsse der Umwelt oder menschlicher und finanzieller Ressourcen.

Dieser Artikel konzentriert sich auf die Schnittmenge der beiden Disziplinen Risiko- und Anforderungsmanagement, und zwar im Zusammenhang mit Produkt- und Systementstehungsprozessen. Genauer gesagt geht er speziell auf Risiken ein, die das Produkt selbst, und nicht die Entwicklungsaktivitäten betreffen, wie beispielsweise inhärente, zu niedrige Schätzungen. Siehe hierzu Abbildung 1.

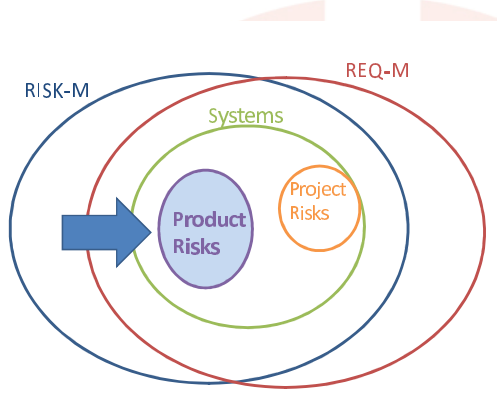


Abbildung 1: Fokus dieses Beitrages.

Ich beginne mit der Definition von Anforderungsmanagement. Anschließend zeige ich, wo die Risiken liegen, und welche Maßnahmen unternommen werden können, um sie zu vermeiden oder zu mindern. Erfahrene Anforderungsmanager können aus diesem Beitrag möglicherweise nicht mehr

lernen, als das sie sich über ihr Doppelleben als Risiko- und Anforderungsmanager klar werden...

2 Was ist Anforderungsmanagement?

Die Definition, die aus meiner Sicht die Sache am besten trifft, stammt von meinem Kollegen Colin Hood [Colin_Hood_1]. Ihre Kraft liegt in ihrer Einfachheit.

"Verstehen, was der Kunde haben möchte, und..."

„...sicherstellen, dass er genau das bekommt“.

Das hört sich kinderleicht an. Also wo liegt das Problem? Lassen Sie uns jedes der beiden Verben "verstehen" und "sicherstellen" genauer anschauen.

3 Verstehen

Risiko 1: Wir sprechen mit den falschen Personen.

Nehmen wir an, wir sprechen mit jemandem über seine Anforderungen an ein System oder Produkt. In diesem Fall müssen wir uns sicher darüber sein, dass wir mit dem "richtigen" "Kunden" sprechen. (Anmerkung: Hier

bedeutet "Kunde" auch einen internen Kunden innerhalb der Firma. Das trifft auch für die Bezeichnung "Lieferant" zu, die später in diesem Artikel aufgegriffen wird. Dieser kann nämlich auch ein interner Lieferant innerhalb einer Firma sein).

Besonders Systeme, zum Beispiel ein Software-System oder ein System mit einer Mischung aus Software und Hardware, haben viele mögliche "Kunden". Finanzielle und zeitliche Einschränkungen der Projektbudgets hindern uns normalerweise daran, mit all diesen Personen zu kommunizieren, um ihre Anforderungen zu bekommen. Der Umstand, von wichtigen Rollen keine Anforderungen zu erheben, kann jedoch dazu führen, ganz einfach das "falsche" Produkt zu erstellen. Wir versäumen es vielleicht, einige sehr wichtige Anforderungen zu erfassen, weil wir nicht mit den "richtigen" Personen gesprochen haben.

Wie hoch ist die Wahrscheinlichkeit, dass dieser Fall eintritt? Aus meiner Erfahrung ist sie sehr hoch, tendiert gegen 100%, und muss deshalb einer näheren Betrachtung unterzogen werden.

Die klassische Methode um diese Situation zu vermeiden ist es, eine Checkliste für die sogenannten "Stakeholders" zu verwenden.

Siehe hierzu Abbildung 2.

Role	Name	Elicit	Notes
Pays for system	Mr Money	No	Has no idea
Uses system	U. Ser	Yes	
Tests system	M.O.T.	Yes	
Specifies system	Mr Wreck	Yes	
Is affected by system	P. Destrian	Yes	
Maintain the system	Mac Anick	Yes	

Abbildung 2: Beispiel für eine Stakeholder-Liste

Eine Liste von Rollen dient dabei als Checkliste. Für jede Rolle werden eine oder mehrere Personen identifiziert, die für die Erhebung der rollenspezifischen Anforderungen in Betracht gezogen werden. Manche Anforderungsmanager nehmen vielleicht das kalkulierte Risiko in Kauf, von

bestimmten Rollen keine Anforderungen zu erheben. Ein guter Manager wird die Gründe hierfür dokumentieren. Auch die Auswahl von Personen für jede Rolle birgt ein Risiko. Anforderungen von einem Systemnutzer zu erheben, der gerade erst in die Firma eingetreten ist, oder von jemandem, der sich gerade erst in eine neue industrielle Domäne einarbeitet, ist sicherlich nicht wirkungsvoll. Hier ist die Projekterfahrung eines erfahrenen Projektmitarbeiters gefragt, um die am besten geeigneten Personen für eine bestimmte Rolle auszusuchen. Aber aufpassen! Einige Projektverantwortliche kennen ihre "Kunden" nicht so gut wie sie es selbst meinen.

Risiko 2: Der Auftraggeber weiß nicht, was er will.

Manchmal hört man Entwickler eines Systems sich darüber beklagen, dass die Kunden nicht wissen, was sie wollen. Dies ist oft wahr, und das aus gutem Grund. Viele Menschen finden es leichter, sich eine bestimmte Lösung vorzustellen, um herauszufinden, was sie wirklich wollen. Nur wenige Menschen sind in der Lage präzise zu definieren, was sie brauchen, ohne ein ähnliches System oder eine ähnliches Problemfeld zu kennen. Unsere wirklichen wichtigen Anforderungen sind oft in unserem eigenen Unterbewusstsein versteckt, und wir brauchen einen Auslöser, der sie ans Licht bringt. Wenn wir versuchen unsere Kunden zu überzeugen, mit einem leeren Blatt Papier zu beginnen, ist es wahrscheinlich, dass wir nicht alle wirklich wichtigen Anforderungen bekommen.

Die Wahrscheinlichkeit für dieses Risiko ist 100%.

Wenn Sie nicht wissentlich Anforderungen managen ist es wahrscheinlich, dass diese Tatsache übersehen wird. Formales Anforderungsmanagement beinhaltet die Aktivität des "Erhebens". Das heißt, etwas ans Licht zu bringen, normalerweise eine Anforderung, ein Ziel oder eine Erwartung. Und damit kommt auch die Erkenntnis, dass wir ein wenig nach Anforderungen "graben" müssen, ähnlich wie ein Schatzgräber.

Es gibt viele sogenannte Erhebungstechniken, die verwendet werden können, um dies zu erreichen. Ein erfahrener und geschickter

Interviewer weiß, wie man die wichtigen Anforderungen aus dem Befragten "heraus kitzelt".

Moderne Software-Entwicklungsmethoden liefern Releases in regelmäßigen, kleinen Abständen, wobei jede Iteration die Erfüllung einiger Anforderungen oder Anwendungsfälle sicherstellt. Die iterativen Releases der Software sind eine Hilfe um die Sicht des Kunden noch mehr auf neue und existierende Anforderungen zu fokussieren. Unabhängig von Iterationen ist das Erstellen eines Prototyps des endgültigen Produktes ein sehr wirkungsvoller Weg, Träume, Erwartungen und wichtige Anforderungen des Kunden zu erfahren. Manche erfolgreiche Produktmanager können die Käufer sogar dahin bringen zu glauben, dass sie eine bestimmte Eigenschaft des Produktes unbedingt brauchen. Dies erreichen sie, indem sie schöne, verführerische Modelle mit neuen Eigenschaften kreieren lassen. Siehe hierzu Abbildung 3.



Abbildung 3: Beispiel eines Modells

(Quelle: Wikipedia)

Bei diesem Modell kann man die Proportionen des Gebäudes sehen, wie Schatten wirken und was man aus den Fenstern eventuell sehen würde. Vielleicht stellt der Kunde fest, dass er seine unausgesprochenen Pläne, Solarpanels auf dem Dach des kleinen Hauses im Vordergrund zu montieren, revidieren muss. (Die Aktivität des "Modellierens" kann auch Risiken vermeiden helfen, nicht nur für Gebäude, sondern auch für Software. Dies wird später diskutiert).

Risiko 3: Der Kunde schreibt eine Spezifikation.

Nehmen wir an, dass der Kunde wirklich nicht weiß was er will. Je nach Typ der Notationsart, die er verwendet, kann sich das, was er schreibt, von dem, was er meint, unterscheiden. In der Software-Industrie gibt es einige Notationsarten in Diagrammform, die es uns erlauben, unsere Anforderungen mit hoher Genauigkeit und Konsistenz zu formulieren. Beispiele dafür sind die "Unified Modelling Language" (UML) und die "Business Process Modelling Notation" (BPMN). Aber die meisten Kunden schreiben Anforderungen unter Verwendung von Varianten der sogenannten natürlichen Sprache. Dies ist die Sprache, mit der sich Menschen untereinander verständigen, z.B. Deutsch oder Englisch. Und hier beginnt das Problem. Wir Menschen verwenden diese natürlichen Sprachen nicht sehr präzise. Weiterhin verbleibt viel Information in unserem Kopf, die wir nicht mit einbeziehen, wenn wir Anforderungen niederschreiben. Sobald wir etwas niederschreiben, verdrehen wir das, was wir sagen wollen, unwissentlich. Wenn wir Prosa in einer Spezifikation verwenden, besteht somit eine große Gefahr, dass wir Anforderungen "zwischen den Zeilen" begraben.

Die Wahrscheinlichkeit, dass dies geschieht, ist 100%.

Was anschließend passiert, ist wie das Spiel "stille Post". Die Spezifikation wird einem Lieferanten gegeben, und er versucht etwas zu verstehen, was bereits teilweise unklar und nicht komplett ist. Somit besteht ein Risiko, dass das endgültige Produkt oder System ungeeignet für die Umgebung des Kunden sein wird. Solche Anforderungen, die der Kunde klar in seinem Kopf vorliegen hat, werden schlussendlich nicht erfüllt. Wenn wir einmal betrachten, wie viele Spezifikationen rund um die Welt in englischer Sprache geschrieben und gelesen werden, oft durch Menschen, die eine andere Muttersprache haben, so können wir davon ausgehen, dass dies ein weiteres Risiko zusätzlich zum "stille Post" Risiko ist, bzw. dieses noch verstärkt.

Die Wahrscheinlichkeit hierfür liegt bei 100%.

Wenn Sie unterbewusstes Anforderungsmanagement betreiben, kann es sein, dass Sie das Problem erkennen, aber nicht wissen wie Sie damit umgehen können.

Moderne Praktiken, um Anforderungen in natürlicher Sprache zu formulieren, helfen dieses Risiko zu reduzieren.

Das erste, was wir machen müssen, ist den Prosa-Stil aufzugeben. Das ist eine Herausforderung für viele von uns, weil wir eine Menge Zeit damit verbracht haben zu lernen, wie man in Prosa schreibt. Im Gegensatz dazu werden Anforderungen aber in kurzen Sätzen geschrieben. Sie werden nummeriert um sie identifizierbar zu machen. Und eine gute Spezifikation wird klar unterscheiden, welcher Satz eine Anforderung ist, und welcher nur eine Kontextinformation oder eine Überschrift. Andere Techniken achten auf die Vermeidung von Adjektiven, die das Verständnis verschleiern und es schwierig machen zu beweisen, dass die Anforderung erfüllt wurde. Oder es werden Glossare von wichtigen Wörtern und deren Synonymen erstellt. Es gibt aber auch Methoden um Sätze klar zu strukturieren. All diese Techniken zielen darauf ab, Mehrdeutigkeiten zu entfernen und somit das Risiko zu vermindern, dass Dinge missverstanden werden. Eine Spezifikation, die vorher in Prosa geschrieben war, erscheint nun in tabellarischer Form wie in Abbildung 4 dargestellt.

ID	Type	Content	Priority	Comments
	Heading	1 Introduction		
	Information	Blah blah Blah		
	Heading	2 Requirements		
	Heading	2.1 Functional Requirements		
5	Requirement	The tea-drinker must be able to get a cup of tea 5 minutes after his request.	High	
6	Requirement	The tea-drinker must be able to select whether he wants milk	High	
7	Requirement	The tea-drinker must be able to select whether he wants sugar	Medium	Saccharine issue still open.
	Heading	2.1 Non-functional requirements		
9	Requirement	No user may be injured by use of the tea-maker	High	

Abbildung 4: Beispiel einer Spezifikation in tabellarischer Form.

Beachten Sie, dass diese Technik anderen hilft, um Klarheit über das Gesagte oder Geschriebene zu erhalten, aber sie kann uns

nur teilweise helfen, diejenigen Anforderungen zu entdecken, die noch im Kopf des Kunden stecken. Um diese Anforderungen ans Licht zu bringen, werden geeignete Erhebungstechniken benötigt, z.B. Interviews, Brainstorming, Modellierung etc.

Risiko 4: Der Kunde teilt uns die Lösung mit.

Wir Menschen vermischen oft mögliche Lösungen mit den eigentlichen Anforderungen. Ein Kollege, der in unserer Firma neu war, fragte uns, wo in der Nähe eine Drogerie sei. Aber seine wirkliche Anforderung war, einen Film zum Entwickeln zu bringen (das war noch vor der Zeit der digitalen Photographie). Als wir seine wirkliche Anforderung heraus gekitzelt hatten, schickten wir ihn zu einem Fotoladen gleich ein paar Häuser weiter. Wenn wir ihn nicht gefragt hätten, was hinter seiner Frage steckt, hätten wir ihn vielleicht in die falsche Richtung oder viel zu weit geschickt. Lösungen anstatt Anforderungen zu beschreiben erhöht das Risiko, dass Entwickler ungewollt in die falsche Richtung geschickt werden, und dass dann die "falsche" Lösung ausgewählt und gebaut wird.

Was ist die Wahrscheinlichkeit hierfür? Sie tendiert gegen 100%. Versuchen Sie einmal zu beobachten, wie oft Menschen Beschreibungen einer Lösung anstatt ihre eigentlichen Anforderungen vorbringen.

Wenn wir kein Bewusstsein für dieses Phänomen haben und kein formales Anforderungsmanagement durchführen, ist es extrem wahrscheinlich, dass wir nicht bemerken was das eigentliche Problem des Kunden ist und was lediglich eine von mehreren Lösungen darstellt. Solange wir Lösungen diskutieren kann es sein, dass wir nie das wirkliche Problem verstehen und dieses zu klären versuchen. Also: Hüten Sie sich vor Lösungen, die vorgeben, Anforderungen zu sein. Ein Weg, dieses Risiko zu umgehen, ist, hinter das wirkliche Problem zu kommen. Dies macht man, indem man den Anforderungsersteller fragt, warum er denn die Anforderung erfüllt haben möchte. Manchmal muss man mehrmals hintereinander nach dem "Warum" fragen. Man geht dabei Schritt für Schritt in der Gedankenkette zurück, um an die wirkliche Anforderung zu kommen.

Das Bewusstsein für die Trennung von Problem und Lösung zu erhöhen, ist eine der wichtigsten Ziele von RM&E-Trainingskursen. Eine gute Schulung kann das Risiko minimieren, aber dennoch ist viel Praxis notwendig.

Risiko 5: Ein Lieferant liest die Kundenspezifikation

Das mag merkwürdig klingen.

Es sollte nämlich selbstverständlich sein, dass der Lieferant wirklich die Kundenspezifikation liest.

Aber während des Lesens wird der Lieferant auch Informationen auf dem Papier übersehen und einige von ihnen missverstehen oder verdrehen, bevor sie den Bewusstseinsbereich seines Gehirns erreichen.

Das Ergebnis ist sein bewusstes Verständnis, das nun teilweise inkomplett und inkorrekt ist. In vielen Entwicklungsprozessen ist der nächste Schritt für den Leser von Anforderungen, eine Lösung auszuwählen oder zu erarbeiten, und sie niederzuschreiben, um sie einem anderen geben zu können.

Aber nachdem sein Verständnis sicherlich nicht völlig komplett und korrekt ist, besteht ein Risiko, dass der Entwurf ebenso nicht komplett und korrekt sein wird.

Und wie wir weiter oben gelernt haben, wird er, wenn er beginnt etwas niederzuschreiben, unbewusst Informationen vergessen oder verdrehen. Das "stille Post"-Spiel setzt sich fort bis das Produkt fertig ist (Siehe Abbildung 5).

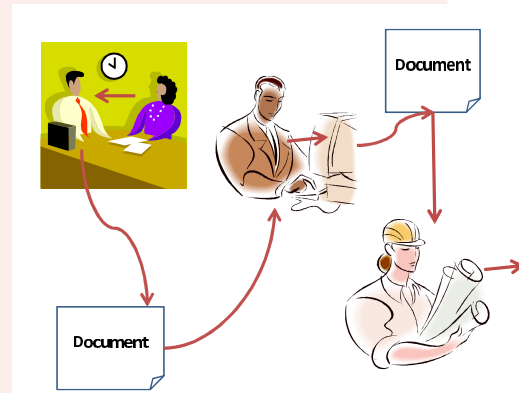


Abbildung 5: Risiken im Verlauf einer Kommunikations- / Spezifikationskette.

Die Wahrscheinlichkeit, dass dies geschieht, ist auch 100%.

Es gibt unterschiedlichste Methoden, die dem Lieferanten helfen können, die Anforderungen eines Kunden zu verstehen, um das Risiko eines Missverständnisses zu reduzieren.

Zwei dieser Methoden sind. (a) die in natürlicher Sprache geschriebenen Anforderungen anhand etablierter Qualitätskriterien zu analysieren, und (b) Modelle der System- oder Produktumgebung, in der das Produkt / System funktionieren soll, oder des vorgeschlagenen Systems / Produktes selbst erstellen. (Andere Maßnahmen, die hier nicht besprochen werden, beinhalten weitere Diskussion mit dem Kunden, oder Ableitung einer detaillierteren Spezifikation des Produktes, und das Veranlassen des Kunden, diese zu prüfen).

Obwohl wir immer noch das Verb "verstehen" ("...was der Kunde will") betrachten, sind wir jetzt ebenso auf dem Weg sicherzustellen, dass der Kunde bekommt was wir verstanden haben.

Lassen Sie uns (a) betrachten: "Analyse von Anforderungen in natürlicher Sprache".

Einige der oben erwähnten Maßnahmen können ebenso beim Analysieren von Anforderungen angewandt werden. Die erste Aufgabe besteht darin herauszufinden, wo genau in einer Kundenspezifikation die Anforderungen stehen, und dann die Kontextinformationen und Überschriften zu separieren. Wenn die Anforderungen in zusammen-hängenden Satzgruppen als Prosa geschrieben wurden, ist es sinnvoll sie in ihre grammatikalischen Einheiten, nämlich einzelne Sätze, aufzuteilen. Jeder Satz kann dann daraufhin analysiert werden, ob er eine oder mehrere Anforderungen enthält. Schlüsselverben und -substantive können isoliert und geklärt, synonyme Begriffe durch ein einziges Wort ersetzt werden. Sätze können so besser nach sogenannten "schwachen Wörtern" durchsucht werden (Adjektive, Komparative, etc.), die es normalerweise schwierig machen zu beweisen, dass eine Anforderung erfüllt wurde.

Oft fordere ich in meinen Trainingskursen die Teilnehmer dazu auf, eine Strategie vorzuschlagen, um die folgende Anforderung auf Erfüllung zu prüfen:

- "Das System muss intuitiv zu erlernen und leicht zu verwenden sein."

An dieser Stelle müssen die Adjektive "leicht" und "intuitiv" definiert werden. Für wen sollte das System leicht und intuitiv sein? Wie werden diese Personen die Leichtigkeit und Intuitivität messen? Wie können diese Eigenschaften bewiesen und getestet werden? Wenn ein Lieferant dies nicht klarstellt und auf Festpreisbasis arbeitet, könnte er ein großes Risiko eingehen.

Das HOOD Produkt "Desire"© [Desire_2] durchsucht Dokumente und hilft dem Leser, solche problematischen Anforderungen zu identifizieren.

Wenn wir an dieser Stelle kein formales Anforderungsmanagement durchführen, werden wir möglicherweise einige dieser Analysetechniken intuitiv anwenden. Mit einer bewussten Anwendung kann man oft bessere Ergebnisse erzielen.

Bezüglich des Punktes (b) "Modelle erstellen..." lassen Sie uns die natürliche Sprache verlassen und zu einigen der Notationen zurückkommen, die wir weiter oben erwähnt haben. Es gibt verschiedene technische Notationen, die es erlauben, Diagramme zu erzeugen, welche die Umwelt, in der das System zu funktionieren hat, oder das System selbst repräsentieren können. Notationen dieser Art benutzen Symbole und Formen, sowie Regeln, um sie kombinieren zu können. Diese Regeln stellen hohe Qualität der Anforderungen dadurch sicher, dass sie helfen, fehlende Informationen und Widersprüche zu identifizieren.

Eine Sammlung solcher Diagramme wird auch Modell genannt. Die Tätigkeit des Modellierens kann dabei helfen, fehlende Anforderungen in einem bereits erhobenen und dokumentierten Satz von Anforderungen ans Licht zu bringen und mögliche Widersprüche und Lücken herauszustellen.

4 Sicherstellen

Risiko 6: Anforderungen gehen unterwegs verloren.

Lassen Sie uns nun das zweite Verb aus Colin Hoods Definition etwas näher unter die Lupe nehmen (...sicherstellen, dass er genau das bekommt"). Wenn Sie keine Möglichkeit haben um verfolgen zu können, was mit einer Anforderung geschehen ist, besteht die Gefahr, dass sie vergessen wird oder in dem Berg von Information verloren geht, mit dem wir es zu tun haben. Die Wahrscheinlichkeit, dass dies geschieht, ist sehr hoch.

Informelles und unbewusstes Anforderungsmanagement kann diese Situation nur für einige wenige Anforderungen und einfache Systeme meistern.

Vergessen wir nicht, dass große Systeme in Phasen entwickelt und gebaut werden, wobei man sich am Anfang auf die Anforderungen des Kunden konzentriert. In einem weiteren Schritt werden diese zu Systemanforderungen verfeinert, die dann wiederum weiter herunter gebrochen werden in Anforderungen an Subsysteme. Es besteht dabei das Risiko, dass eine Anforderung ganz einfach vergessen wird, unabhängig davon, ob sie unterwegs missverstanden oder verfälscht wird. Je größer das System, umso größer ist die Wahrscheinlichkeit, dass dieses Risiko auftritt.

Bei einer großen Anzahl von Anforderungen wird normalerweise Unterstützung durch ein Softwarewerkzeug benötigt.

Erinnern Sie sich noch an das „stille Post“-Spiel? Wäre es nicht wünschenswert, wenn man den Pfad einer Anforderung nachverfolgen könnte, nachdem sie durch jemand niedergeschrieben wird, durch eine weitere Person gelesen, dann wieder verfeinert umformuliert, und schließlich jemandem in der Kette weitergegeben wird. Wir könnten sehen, wo das Original verloren gegangen ist oder missverstanden wurde. Um diese Nachverfolgung sicherstellen zu können ist es notwendig, der Anforderung eine Identitätsnummer zu geben, die als Referenz in allen weiteren Arbeitsschritten verwendet werden kann. Dies ist normalerweise eine alphanumerische Reihe von Schriftzeichen (String). Und dann müssen wir in der Lage

sein, diese Identitätsnummer zu verfolgen, wo immer sie erscheint. Es gibt einige Softwareprogramme, die dieses Vorgehen unterstützen, eine Anforderung zu verwalten und individuell zu verfolgen.

Leider wird bei der Entstehung eines Systems oder einer Software eine ganze Reihe von sogenannten Softwarewerkzeugen gebraucht, wobei diese unterschiedliche Unterstützungsaufgaben an verschiedenen Stellen des relativ komplexen Entwicklungsprozesses leisten. Um die Spuren der Flüsterer in dem Spiel "stille Post" den gesamten Weg hinunter zum Produkt verfolgen zu können, muss man Daten verfolgen, die in verschiedenen Datenbanken gespeichert sind und mit unterschiedlichen Softwarewerkzeugen verarbeitet werden. Produkte, welche diese unterschiedlichen Werkzeuge verbinden können, kommen immer stärker auf den Markt.

Auch findet der Prozess der Verfeinerung von Anforderungen, der abstrakte Ideen in präzise Spezifikationen umsetzt, normalerweise in unseren Köpfen statt. Wir treffen Annahmen und wählen eine von vielen möglichen Lösungen, wobei wir uns gänzlich unbewusst sind, dass wir das tun. Mit der Unterstützung von RM&E-Softwarewerkzeugen können wir Gedankenketten darstellen, die eine Anforderung von einer zur nächsten Stufe verfeinert haben. Wenn wir diese Information aus unserem Kopf auf Papier bringen, könnte jemand anderer unsere Entscheidungsprozesse nachprüfen und so die Qualität des endgültigen Produktes verbessern. Siehe hierzu Abbildung 6.

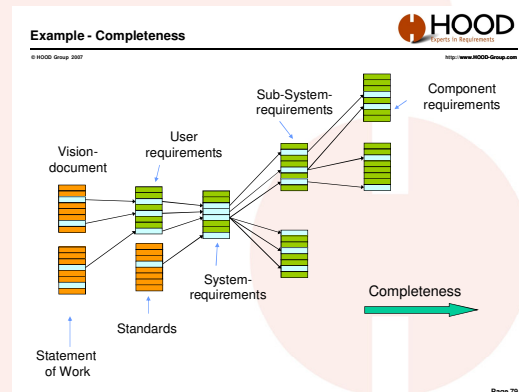


Abbildung 6: Verknüpfung von Anforderungen über mehrere Stufen, ermöglicht durch Werkzeuge

Risiko 7: Das System wird anhand veralteter Anforderungen getestet.

Ich betone, dass ich hier angenommen habe, dass das System getestet wird. Ich hoffe es ist jedem klar, dass es ein hohes Risiko darstellt, ein System nicht zu testen. Ein ähnlicher Artikel wie der vorliegende könnte heißen: *"Das Risiko, kein formales Testmanagement durchzuführen"*. Ich beende diesen Artikel nicht, ohne die äußerst wichtige Verbindung zwischen Anforderungen und Test anzusprechen.

Wir können und sollten ein System testen um sicherzustellen, dass es nicht zusammenbricht oder ungewollten Schaden anrichtet. Dadurch können wir beweisen, dass wir das System *korrekt* gebaut haben. Aber ist es das *"richtige"* System? Haben wir das *richtige* System in Bezug auf die Anforderungen gebaut? Um diese Frage zu beantworten, müssen wir uns fragen; "Richtig für **welche** Anforderungen?" Idealerweise sollten wir die Kundenanforderungen und auch die Anforderungen der restlichen wichtigen Stakeholder als Testgrundlage nehmen. Dafür ist es absolut notwendig, sich für die Befragung bestimmter Stakeholder entschieden zu haben, ihre Anforderungen ans Licht zu bringen, daraus einen Satz von Anforderungen schriftlich zusammenzustellen, und schließlich zusammen mit den Stakeholdern zu priorisieren.

Wir dürfen dabei aber nicht vergessen, dass Anforderungen sich während der Entwicklung des Systems ändern, und auch mit neuen Schwerpunkten priorisiert werden. Meist werden diese Änderungen nicht niedergeschrieben. Oder sie werden unter vielen e-Mails begraben. Somit passiert es, dass das endgültige System lediglich gegen die Originalanforderungen getestet werden kann, die ja nun veraltet sind. Eine große Anzahl von Tests wird deshalb nicht durchgeführt werden.

Die Chancen, dass so etwas vorkommt, liegen zwischen 50 und 100%.

Während unserer HOOD Trainingskurse stellen wir diese Situation nach. Am Ende der Übung frage ich die Teilnehmer, wer diese Situation kennt. Nahezu alle kennen sie

(Diejenigen, welche sie nicht kennen, sind in der Regel noch am Anfang ihrer beruflichen Laufbahn und haben somit nicht die notwendige Erfahrung, um dies zu beurteilen.). Manche Software-Entwicklungsteams testen ihre Software gegen das Bild des Systems, das sie in ihrem Kopf haben. Unglücklicherweise kann dieses Bild aufgrund der "stillen Post" verzerrt, nicht komplett, ungenau oder ganz einfach veraltet sein.

Es gibt verschiedene Wege um dieses Risiko zu umgehen. Eine dieser Methoden ist sehr einfach: man muss nur sicherstellen, dass man einen Satz von Anforderungen niederschreibt und diese abgestimmten Anforderungen laufend aktualisiert, wobei man sich wieder das Einverständnis der involvierten Stakeholder holen muss. Eine andere Technik zögert die detaillierte Beschreibung von Testfällen so lange hinaus, bis die Anforderungen eine hohe Stabilität erreicht haben. Die Testfälle sollten auf dem Satz von Anforderungen basieren, der mit den Stakeholdern abgestimmt wurde. Man bezieht sich hierbei auf die Identitätsnummern, von denen man annimmt, dass sie erfüllt werden, wenn das System diesen Testfall erfolgreich absolviert hat.

Wenn Sie Werkzeuge haben um dies zu tun, können Testfälle zu Anforderungen verlinkt werden. Moderne Anforderungsmanagement-Werkzeuge können Sie alarmieren, wenn etwas am anderen Ende des Links geändert wurde, so dass man die Testfälle oder die Anforderungen entsprechend anpassen kann.

5 Zusammenfassung

Risikomanager, die an der Entstehung eines Systems (Hardware und Software) direkt beteiligt sind, sollten sich darüber bewusst werden, dass die größten Risiken für ein Produkt entstehen, wenn man versucht zu verstehen und zu interpretieren, was der Kunde möchte.

Das Risiko, das dann damit verbunden ist, das "richtige" Produkt zu liefern, hängt stark von der Reife des Entwicklungsprozesses ab, der angewandt wird. Wenn Sie versuchen die Anforderungen zu verfolgen, werden Sie früher oder später bemerken, dass Sie Ihre Entwicklungsprozesse und Ihre Software-

werkzeug-Landschaft verbessern müssen. Siehe hierzu Tabelle 1.

Die HOOD Group ist immer bereit, Hilfe zur Selbsthilfe ("Helping you to help yourself") zu leisten. Es ist ihr Motto.

6 Referenzen

[Colin_Hood_1] HOOD Group Training Material / Colin HOOD / 2001

[Desire_2] <<http://www.hood-group.com/en/products/desire/>>

HOOD Group / 2007

7 Über den Autor

Robert Stevenson ist Senior Consultant in der HOOD Group.

Nach erfolgreichen Arbeiten bei einem Marktführer für ein Anforderungsmanagement-Werkzeug wechselte er im Jahre 2002 zur HOOD Group.

Sein Fachgebiet ist die Einführung von Anforderungsmanagement-Methoden und -Werkzeugen in betrieblichen Informationsverarbeitungsprojekten. Herr Stevenson hat mehr als 25 Jahre Erfahrung in der Informationsverarbeitungsindustrie, sowohl in England als auch in Deutschland.

Aktivität	Risiko	Gegenmaßnahme
Verstehen	Wir reden mit den verkehrten Personen	Stakeholder Management
Verstehen	Der Kunde weiß nicht was er will	Prototyping Modellieren Iterative Releases
Verstehen	Der Kunde schreibt eine Spezifikation	Qualitätskriterien für Anforderungen in natürlicher Sprache Modellieren
Verstehen	Der Kunde teilt uns eine Lösung mit	Trennung der Lösungen von den Problemstellungen
Verstehen/ sicherstellen	Der Lieferant liest (missversteht) die Kundenspezifikation	Analyse der Anforderungsformulierung Modellierung
Sicherstellen	Die Anforderung geht auf dem Weg verloren	Nachverfolgbarkeit sicherstellen
Sicherstellen	Das System wird anhand veralteter Anforderungen getestet	Laufende Aktualisierung der Anforderungen Managen von Änderungen durch Links von Tests zu Anforderungen

Tabelle 1 Zusammenfassung