

Weshalb eine Requirements-Definition erst mit Risiko-Management vollständig ist

Andreas Birk¹, Heinrich Dreier²

¹ Software.Process.Management, andreas.birk@swpm.de

² 3er-consulting, hd@dreier-consult.de

Zusammenfassung. Für die Produktqualität und für die Produktivität der Entwicklung ist es von großer Bedeutung, wie gut Requirements mit Risiko-Management integriert sind. Gängige Prozessvorgaben und -anleitungen betrachten beide Bereiche allerdings weitgehend isoliert voneinander. Dieser Artikel motiviert und begründet, weshalb Requirements und Risiko-Management möglichst eng miteinander verzahnt werden sollten.

Requirements und Risiko-Management zusammen

Für fast jedes Software- und System-Produkt, und fast jedes Entwicklungsprojekt gilt: Es ist Risiken ausgesetzt und es kann Risiken für andere bewirken. Requirements legen die Grundlage für Produkte und Projekte. Folglich muss eine vollständige Requirements-Definition auch Risiko-Management beinhalten.

Für die Produktqualität und für die Produktivität der Entwicklung ist es von großer Bedeutung, wie gut Requirements mit Risiko-Management integriert sind. Allerdings betrachten gängige Prozessvorgaben die beiden Bereiche weitgehend isoliert voneinander. Beispielsweise führt dies dazu, dass bei Requirements-Änderungen die Risiko-Bewertung oft nicht aktualisiert wird. So entstehen zusätzliche Aufwände und die Qualität leidet.

Erst wenn Requirements und Risiko-Management eng miteinander integriert sind, können sie ihr volles Potenzial entfalten und viele Synergien bewirken.

Requirements und Risiken: Zwei Seiten der gleichen Medaille

Requirements, die Entwicklungsprojekte, in denen Requirements erstellt und bearbeitet werden, sowie die durch die Requirements spezifizierten Ergebnisse dieser Projekte stehen in mehrfacher Beziehung zu Risiken. Zwei Aspekte sind dabei besonders relevant (vgl. Abbildung 1):

(1) Requirements und Entwicklungsprojekte unterliegen Risiken, die ihren Erfolg beeinflussen. Requirements und Projekte sind also *Risiken ausgesetzt*.

(2) Requirements und Projekte, und die durch sie spezifizierten und erstellten Ergebnisse bewirken Risiken für externe Parteien, Objekte und Sachverhalte. Requirements und Projektergebnisse *bewirken Risiken*.

Der erste Aspekt, dass Requirements und Projekte Risiken ausgesetzt sind, wird in vielen Requirements-Methoden behandelt. Die Methoden greifen auf Techniken des Risiko-Managements zurück, um Risiken zu erfassen und zu bearbeiten. Ein Beispiel ist die *Volere*-Methode [1][2].

Der zweite Aspekt, dass Requirements und spezifizierte Systeme Risiken bewirken, wird in Prozessvorgaben ebenfalls häufig beachtet. Ein Beispiel ist der Prozess-Standard ISO/IEC 15504-5:2018 (SPICE) [3] und der abgeleitete Standard ASPICE aus der Automobilindustrie [4]. Sie fordern, dass das Requirements-Management die durch die spezifizierten Produkte bewirkten Risiken betrachten und behandeln muss. Ähnliches verlangen Branchenstandards und rechtliche Vorgaben etwa für funktionale Sicherheit, Medizintechnik und Datenschutz.

Risiko-Management: Eine Lücke im Requirements-Management

Während Prozessvorgaben wie SPICE fordern, dass Requirements-Management die bewirkten Risiken managen soll, mangelt es im Requirements-Management weitgehend an methodischen Anleitungen und Hilfestellungen, wie diese bewirkten Risiken behandelt werden können.

Einschlägige Requirements-Methoden wie *Volere* oder grundlegende Beiträge zum Risiko-Management für Requirements (z.B. [5][6]) behandeln fast

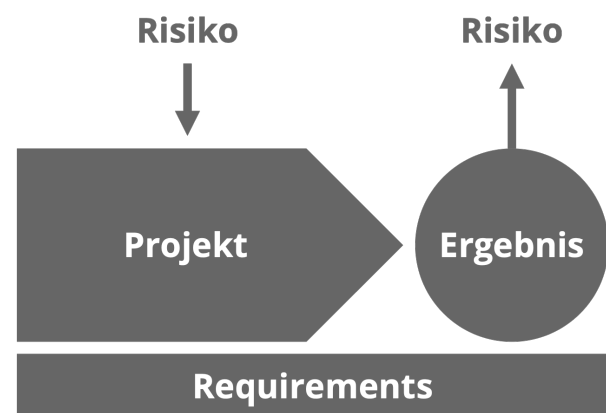


Abbildung 1: Risiken, denen ein Projekt ausgesetzt ist, und Risiken, die durch ein Projektergebnis bewirkt werden.

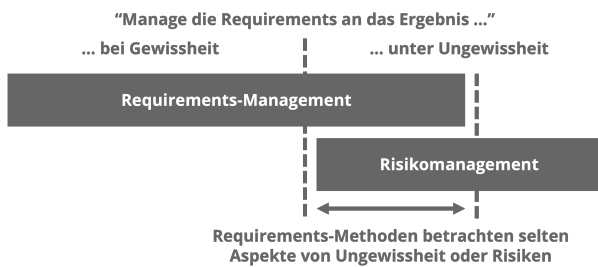


Abbildung 2: Requirements-Management und Risikomanagement überlappen sich, während Requirements-Methoden Lücken beim Umgang mit Ungewissheit aufweisen.

ausschließlich Risiken, denen das Requirements-Management ausgesetzt ist.

Es besteht also eine weit verbreitete methodische Lücke im Requirements-Management für den Umgang mit Ungewissheit im Zusammenhang mit Requirements. Abbildung 2 illustriert dies: Requirements können sowohl sicher absehbare Sachverhalte betreffen als auch ungewisse.

Beispiele für Ungewissheit bei Requirements sind die Machbarkeit technologisch neuer Produktfeatures, die Umsetzbarkeit aufwändiger Features abhängig von ökonomischen Rahmenbedingungen, sowie die durch das spezifizierte Produkt bewirkten Risiken (funktionale Sicherheit, Umweltschäden, Datenschutz etc.).

Praktiken für integriertes Requirements- und Risikomanagement

In der Praxis des Requirements-Managements trifft man durchaus tragfähige Vorgehensweisen an, die Requirements mit Risiko-Management integrieren. Sie können als Startpunkt für die methodische Integration in Prozessvorgaben dienen:

Requirements-Risiken durch Attribute kennzeichnen: Insbesondere wenn Requirement mit spezialisierten Requirements-Tools erfasst und verwaltet werden, kann man für jedes Requirement leicht Attribute mit Risikobewertungen mitführen. Solche Bewertungen können unterschiedlichste Sachverhalte ausdrücken, und damit verschiedene Aspekte des Risiko-Managements unterstützen: Technologische Umsetzungsrisiken, ökonomische Umsetzungsrisiken oder Risiken, die von einer spezifizierten Funktionalität ausgehen (z.B. Gefährdungen von Nutzern).

Risiko-Management auf Requirements beziehen: Prozessvorgaben können für jede Projektstufe (z.B. Quality Gates) fordern, dass Risiko-Management durchgeführt wird. Dies kann sich sowohl auf die Betrachtung von Umsetzungs- und Projekt-Risiken beziehen, als auch auf Risiken, die durch die spezifizierten Features bewirkt werden.

FMEA-Funktionen mit Requirements verbinden: FMEA (*Failure Mode and Effects Analysis*, vgl. [7]) ist eine Qualitätssicherungsmethode, die im Grunde von Requirements ausgeht und durch sie bewirkte Risiken untersucht und behandelt. Dabei setzt sie aber die

erforderlichen Requirements-Informationen neu auf und behandelt auch die Risiken unabhängig von möglicherweise schon bestehenden Requirements-Spezifikationen. In organisationspezifischen Prozessvorgaben können Requirements-Management und FMEA stärker zusammengeführt werden. Tool-Brücken zwischen spezialisierten Requirements- und FMEA-Tools ermöglichen es, die betreffenden Workflows effektiv zu integrieren und effizienter zu gestalten.

Argumente für die Integration von Requirements- und Risikomanagement

Die Fallbeispiele zeigen: Requirements-Management und Risiko-Management können in der Praxis wirkungsvoll integriert werden. Als Nutzen einer solchen Verbindung dürfen Unternehmen erwarten:

- Frühere Absicherung der Requirements-Qualität durch frühzeitige Risiko-Analyse, die ebenso als Beitrag zu Requirements-Analyse und -Validierung wirken kann
- Erhöhung der Produktqualität durch bessere Risiko-Absicherung; etwa aus der systematischen Risiko-Behandlung heraus, die auch zu neuen Requirements und zusätzlichen Tests führen kann
- Verringerte *Cost of Failure*, durch bessere Qualität und höhere Risiko-Resilienz der Produkte
- Erhöhte Produktivität und verringerte Kosten, durch Vermeidung später Änderungen (etwa Änderungen wegen vernachlässigter Risiken) und durch Synergien zwischen Requirements, Risiko-Management und Qualitätssicherung

Damit solche Nutzeneffekte breit verfügbar werden, muss die oben beschriebene methodische Lücke geschlossen werden: Risiko-Management muss fester Teil des Requirements-Managements werden. Dann kann sowohl das Requirements-Management als auch das Risiko-Management noch weiter an Akzeptanz gewinnen, und die Methoden werden noch tiefer in der Unternehmenspraxis verankert.

Literaturverweise

- [1] S. Robertson and J. Robertson, *Mastering the requirements process: Getting requirements right*, 3rd Ed. Upper Saddle River, NJ: Addison-Wesley, 2012.
- [2] *Volere Website*. <https://www.volere.org>
- [3] ISO/IEC, *Information technology — Process assessment — Part 5: An exemplar software life cycle process assessment model*, ISO/IEC 15504-5:2012(E), 2012.
- [4] Qualitäts Management Center im Verband der Automobilindustrie (VDA QMC), *Automotive SPICE®*. <https://vda-qmc.de/software-prozesse/automotive-spice>
- [5] B. W. Boehm and T. DeMarco, *Software risk management*, IEEE Software, vol. 14(3), pp. 17–19, 1997.
- [6] T. DeMarco and T. Lister, *Risk management during requirements*, IEEE Software, vol. 20(5), pp. 99–101, 2003.
- [7] *FMEA – Wikipedia*. <https://de.wikipedia.org/wiki/FMEA>