



□ Dr. Claudia Hirschmann

(E-Mail: Claudia.Hirschmann@siemens.com)
 hat sich nach einigen Jahren Softwareentwicklung dem Qualitätsmanagement in der Medizintechnik-Branche zugewandt. Heute ist sie Senior Consultant auf dem Gebiet Enterprise Processes Consulting in der Siemens AG Corporate Technology. Dr. Claudia Hirschmann ist EOQ Quality Management Technician, Certified Scrum Master und Certified Scrum Product Owner.

Agiles Anforderungsmanagement in der Medizintechnik

Für das Anforderungsmanagement ist in einer agilen Vorgehensweise in der Regel der Product Owner zuständig. In einigen Branchen gibt es neben den Kundenanforderungen eine Reihe von strengen gesetzlichen Vorgaben, die zusätzlich zu berücksichtigen sind.

Product Owner zwischen den Stühlen

Wissen, was der Kunde haben möchte und dies zum richtigen Zeitpunkt entwickelt zu haben, ist eine Kunst. Agile Methoden in der Entwicklung tragen wesentlich dazu bei, bei diesem Spagat erfolgreich zu sein. Für das Management der Kundenanforderungen ist ein Vorgehensmodell, das agile Methoden einsetzt, durchaus ein Fortschritt im Vergleich zu Phasenmodellen, die z.B. dem Wasserfall- oder V-Modell folgen.

Aber nicht alle Kunden bzw. Branchen sind gleich! Es gibt Bereiche, in denen Transparenz und detaillierte, dokumentierte Nachvollziehbarkeit extrem wichtig sind, was in dem dort geforderten Ausmaß im agilen Umfeld so nicht automatisch gegeben ist: Wo das Produkt direkten Einfluss auf Gesundheit und Leben hat, wie in der Branche der Medizinprodukte, muss zum Beispiel genau spezifiziert werden, was es wie können soll, welche Schutzmaßnahmen wie implementiert werden, usw. Nebenbei bemerkt – dies ist auch beruhigend zu wissen, wenn man sich z.B. unter ein Röntgengerät legen muss.

Um Medizinprodukte möglichst sicher und wirksam zu gestalten, wurde eine

Reihe von gesetzlichen und normativen Anforderungen an die Hersteller gestellt. Die Vorschriften beziehen sich auf alle Aktivitäten eines Unternehmens, also die Entwicklung, die Fertigung, ... und gleichermaßen auf die Spezifikationsphase der Geräte. Schließlich soll ja sichergestellt werden, dass nur effektive, effiziente und sichere Produkte auf den Markt kommen.

In einem agilen Entwicklungsvorgehen ist der Product Owner für die Identifikation und Festlegung der Anforderungen an das Produkt zuständig. Er sitzt zwischen den Stühlen eines auf der einen Seite schnellen, agilen und auf der anderen Seite regulatorisch korrekten Umgangs mit den sogenannten ‚Design Input Requirements‘ (Entwicklungsanforderungen, *siehe auch Erklärung im Kasten*).

Das Ohr am Kunden

Die Entwicklung bekommt die Anforderungen vom Product Owner übermittelt, der Spezialist für alles ist, was der Kunde wünscht, braucht und lieben wird. Dazu hat er sein Ohr jederzeit am Kunden, um schnell und flexibel auf dessen Bedürfnisse reagieren zu können.

Eine agile Vorgehensweise in der Entwicklung bietet für die Umsetzung der Ideen des Kunden ideale Voraussetzungen. Hier kann der Product Owner nach den relativ kurzen Entwicklungszyklen bereits lieferbare Ergebnisse sehen. Über die agilen Planungsmeetings hat er regelmäßige und relativ kurz getaktete Möglichkeiten, seine neuen Erkenntnisse aus der Welt des Kunden einzubringen, die dann je nach Machbarkeit schon innerhalb eines der nächsten Zyklen realisiert werden können.

Bei Medizinprodukten – insbesondere bei der Entwicklung von sicherheitsrelevanten Komponenten – ist es, wie bei allen Produktentwicklungen, wichtig, die Wünsche aus der Welt des Kunden unmissverständlich und gut zu verstehen, um sie adäquat umsetzen zu können und darüber hinaus dabei immer zu berücksichtigen, auf keinen Fall die Gesundheit von Beteiligten wie Patienten, Ärzten, MTAs, etc. zu gefährden.

Die gesetzlichen und behördlichen Vorschriften, Bestimmungen, Standards sollen genau dies sicherstellen. Eine dieser Vorgaben ist, dass die Design Input Requirements (Entwicklungsanforderungen, *siehe auch Erklärung im Kasten*) dokumentiert werden müssen.

Design Input

In den USA wird für Medizinprodukte in der Quality System Regulation (QSR, 21 CFR Part 820) im Abschnitt ‚Design Controls‘ (Entwicklungslenkung) unter ‚Design Input‘ (Entwicklungsvorgaben) unter anderem gefordert, dass ‚Design Input Requirements‘ (Entwicklungsanforderungen) dokumentiert werden müssen. Laut QSR sind sie nicht nur zu dokumentieren, sondern auch einem Review zu unterziehen und freizugeben.

In der Norm für Medizingeräte-Software – Software-Lebenszyklus-Prozesse (IEC/ISO 62304) – wird ferner eine angemessene Dokumentation der Software Requirements inklusive Anforderungen, beispielsweise an Schnittstellen, Security und Usability Engineering für alle Software Safety-Klassen, gefordert. Risk Control Measures sind zumindest für die Software Safety-Klassen B und C mit aufzunehmen. Software der Klasse C als höchste Klasse wird schließlich bei Gefahr von potenziell eintretender schwerwiegender Verletzung oder Tod als dazu beitragende Komponente in möglichen Zusammenhang gebracht. Die Software Requirements sind in angemessener Weise zu analysieren und infolge der Analyse zu reevaluieren und zu aktualisieren.

Wie können all diese Anforderungen erfüllt werden, ohne die agile Entwicklung auszubremsen? Wie kann das ohne einen aufwendigen Dokumentations- und Review-Prozess zu Beginn jeder Iteration effektiv und effizient umgesetzt werden?

Eine Vorgehensweise aus dem traditionellen Wasserfall- oder V-Modell, in dem jeder Design Input mehr oder weniger schwerfällige Prozessschleifen durchlaufen muss, wird bei einem agilen Entwicklungsvorgehen in der Regel als weniger praktikabel angesehen. Herkömmliche Herangehensweisen scheinen das agile Vorgehen mit den relativ kurzen Iterationen von jeweils wenigen Wochen aufzuhalten.

Die Basis ...

Eine Grundlage der Planung und Entwicklung stellt in einer agilen Entwicklung der Product Backlog und der Releaseplan dar. Dabei wird im Releaseplan geplant, welche Product Backlog Items in welcher

Iteration implementiert werden sollen. Es ist wichtig und gute Praxis, einen Product Backlog und Releaseplan nachweisen zu können, die zum Beginn der ersten Iteration gültig waren, und ebenso freigegebene letzte Versionen, die die umgesetzten Design Input Requirements des freigegebenen Produkts dokumentieren.

Ein aktuell gehaltener Releaseplan erleichtert während des iterativen Vorgehens in der Entwicklung z. B. die Planung notwendiger Abstimmungen und Klärungen und die Erstellung einer angemessenen Architektur. Er hilft außerdem bei der Planung der Arbeiten an den Design Input Requirements, welche den Entwicklungsaktivitäten vorausgehen müssen.

Rein zeitlich gesehen, hat die Erfahrung aus Projekten gezeigt, dass die Spezifikationen der Product Backlog Items, die gemäß Releaseplan in einer Iteration implementiert werden sollen, als Daumenregel bereits zwei Iterations-Zeitscheiben vorher definiert werden sollten. Auf diese Weise kann die Entwicklung notwendige Vorarbeiten, wie z. B. eine Analyse der Auswirkungen auf die Architektur und der ggf. notwendigen Schutzmaßnahmen, rechtzeitig in ihre Iterationen einplanen und auch durchführen.

Ein Review dieser Spezifikationen der Backlog Items mit dem Architekten, einem „Chef-Entwickler“ und vorgeschriebenen

Funktionsträgern, wie dem Produktrisikoverantwortlichen etc., ist zum einen empfehlenswert, damit z. B. der Architekt eine tragfähige Architektur entwickeln kann und damit adäquate risikominimierende Maßnahmen analysiert, geplant und ergriffen werden können – spätestens, sobald sicherheitsrelevante Software ins Spiel kommt. Diese Arbeiten werden dann ebenfalls den Iterationen zugeordnet.

Zum anderen fordert die Norm IEC/ISO 62304 eine Klassifizierung der Software in Software Safety-Klassen A, B und C und stellt an sicherheitskritischere Software höhere Anforderungen – etwa an die Architektur, an das Detailed Design, die Risikoanalyse und an die Impact-Analyse. Diese Aktivitäten müssen in die Planung, die Schätzung der Aufwände und die Backlogs der entsprechenden Iterationen einfließen.

Aus Entwicklungssicht kann das Review damit als Vorsondierung für das Grooming dienen. Grooming meint hier eine Arbeits-sitzung aus dem Scrum-Umfeld, die während der Iteration stattfindet und in der in erster Linie der für die nächsten 1-2 Iterationen relevante Backlog in kleinere Portionen heruntergebrochen und damit das Iterations-Planungsmeeting vorbereitet wird.

Insgesamt ist bei den Reviews darauf zu achten, dass mindestens zum Beginn der ersten Iteration die Spezifikationen der in

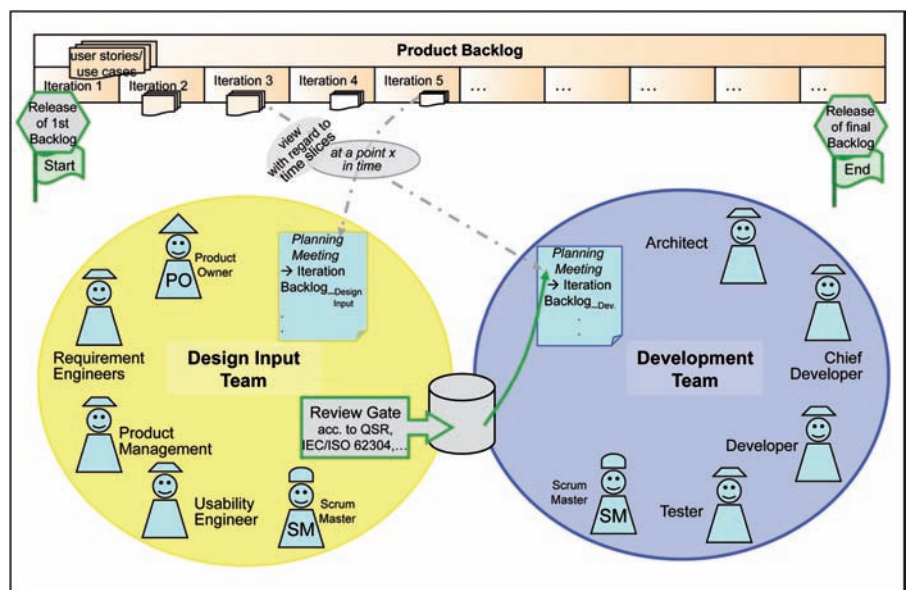


Abb. 1: Review der Spezifikationen der Design Input Requirements von Autor, Architekt, „Chef-Entwickler“ und vorgeschriebenen Funktionsträgern wie Produktrisikoverantwortlichen, etc. unter Beachtung von QSR, IEC/ISO 62304, ... In dieser Aufstellung ist zu bemerken, dass je ein Scrum-Master für das Design Input-Team und das Development-Team unterwegs sind, die zudem als Bestandteil der Teams angesehen werden.

der ersten Iteration umzusetzenden Backlog Items als freigegeben nachweisbar sind. Dies kann beispielsweise in einer vorbereitenden Iteration 0 durchgeführt werden.

Ferner ist am Ende, i. d. R. vor der letzten Implementierungsphase, eine Spezifikation aller Requirements, die im freigegebenen Produkt umgesetzt sein werden, dokumentiert freizugeben. Bei den Freigaben ist auf die Einhaltung der jeweils vorgeschriebenen Review-Teilnehmer, wie Autor, Architekt, „Chef-Entwickler“, Tester, Produktrisikoverantwortlicher, ... zu achten.

Und wer kümmert sich um die Design Input-Anforderungen, deren Dokumentation, die Reviews, ... ?

Agile Teams

In der Regel wird bei agilen Vorgehensweisen viel Augenmerk auf das Entwicklungsteam gelenkt, das die Anforderungen umsetzt, die der Product Owner vom Kunden mitbringt. Das ist gut so, wenn man an die täglichen Aufgaben in der Entwicklung denkt, angefangen von der Entwicklungsumgebung über den Build-Prozess bis zur Testumgebung.

Weniger Gewicht wird üblicherweise auf die Vorgehensweise gelegt, wie der Product Owner zu den Design Input Requirements kommt. Auch hier können sich agile Mittel bewähren, vor allem in einem Umfeld, wo der Product Owner eine ganze Reihe von Aufgaben, wie Spezifikationen inkl. Usability, usw. definieren und dokumentieren sowie Kundenkontakte pflegen, ... zu bewältigen hat, hinter welchen zudem normalerweise zwecks Umsetzung ein nicht zu kleines Team steht.

Gerade in der Medizinprodukte-Branche, in der eine Reihe der Produkte als Kleinserie entwickelt und gefertigt wird (beispielsweise Röntgengeräte), kann in der Regel der Product Owner nicht alle Arbeiten rund um Kundenkontakte, Spezifikationen, Usability Engineering, usw. allein ausführen. Um diese Themen kümmern sich in der Realität mehrere Kolleginnen und Kollegen.

Nichtsdestotrotz soll es für das Entwicklungsteam nur einen Product Owner als Ansprechpartner für sämtliche Fragen zu „Product Owner-Themen“ geben! *Ein* Ansprechpartner gibt dem Entwicklungsteam eine gewisse Sicherheit, da Stimme

und Anforderungen des Kunden, einschließlich Design Input-Anforderungen, nur aus einer Richtung kommen und somit leichter verarbeitet werden können. Das ist eine Herausforderung für den Product Owner und die Gruppe, die sich um die Design Input Requirements kümmert, die es zu meistern gilt.

Ein Weg, diesem Auftrag gerecht zu werden, ist, für die Aktivitäten ‚rund um den Design Input aus Kundensicht‘ gleichsam eine agile Crew aufzustellen. Mit den verschiedenen Arbeitsmitteln eines agilen Teams kann der Product Owner stets tagesaktuell informiert sein, um die Fragen der Entwicklung beantworten zu können, und die Planung der Design Input Spezifikationen kann relativ einfach passend zu dem zeitlichen Rahmen der Iterationen geplant werden.

Zu der Aufstellung eines agilen Teams gehören unter anderem die Einführung agiler Elemente, wie das Daily Scrum, das Iterations-Planungsmeeting sowie Team-Retrospektiven, – was für die Kolleginnen und Kollegen, die an den Design Input Requirements arbeiten, eine gewisse Umstellung bedeuten mag:

- Über das tägliche, kurze „Daily Scrum“ Meeting hat der Product Owner neben dem aktuellen Informationsaustausch im Design Input-Team die Möglichkeit, eventuell aufgetretenen Klärungsbedarf des Entwicklungsteams zeitnah in das Design Input- Team einzubringen und zu lösen.
- Einen Scrum-Master als Prozessunterstützer zu benennen, ist auch für das agile Design Input-Team empfehlenswert. Seine Aufgaben sind in erster Linie: bei der Umsetzung der neuen Vorgehensweise zu helfen, Daily Scrum, Iterations-Planungsmeetings und Team-Retrospektiven einzurichten, Hindernisse aus dem Weg zu räumen und das Team bei den Problemen der Arbeitsumgebung zu unterstützen. Da immer wieder Hindernisse auftreten, die sowohl das Entwicklungsteam als auch das Design Input-Team betreffen und die gemeinsam besser als jeweils separat gelöst werden können, sollten sich die beiden Scrum-Master regelmäßig treffen und bei der Beseitigung solcher Hindernisse unterstützen. Dabei denke man nur an diverse Tools, die von beiden Teams parallel bedient werden.

■ Um in den Iterations-Zeitintervallen bestmöglich arbeiten zu können, ist es ratsam, wenn das agile Design Input-Team gleichfalls in Iterationen arbeitet und die Spezifikationen somit zu den richtigen Zeitpunkten für das „Entwicklungsteam“ bereitstehen. In dem Iterations-Planungsmeeting wird analog der Iteration Backlog aus dem Product Backlog gemäß Releaseplan und Priorisierung nach Risiko, Kundennutzen, etc. ermittelt. Zur Vorbereitung empfiehlt es sich, ein kurzes team-übergreifendes Grooming mit dem Product Owner, einem „Chief Requirements Engineer“, dem Architekten und dem „Chief-Entwickler“ durchzuführen und den Product Backlog zugleich als Planungs- und Kommunikationsmedium gut zu pflegen.

- Das Iterations-Review-Meeting des Design Input-Teams wäre dann z. B. von dem Architekten und einem „Chief-Entwickler“ abzunehmen. Auf Basis der bis dahin bereits durchgeführten Reviews der Spezifikationen kann allerdings i. d. R. das Review-Meeting des Design Input-Teams auf eine Abklärung des Status der Design Input-Spezifikationen abgekürzt werden.
- Die Team-Retrospektive sollte auch für das agile Design Input-Team nicht fehlen. Sie hilft, Erfahrungen mit der möglicherweise ungewohnten Vorgehensweise auszutauschen, aus aufgetretenen Herausforderungen zu lernen, neue Lösungswege zu diskutieren und zur Anwendung zu bringen.

Gerade hinsichtlich einer möglichst effizienten und angemessenen Arbeitsweise bei Review und Dokumentation der Spezifikationen leistet die stete Prozessverbesserung in den Retrospektiven wertvolle Dienste.

Gemeinsam effizient

Ja, im Medizinprodukteumfeld gibt es einige Erfordernisse, zum Beispiel an Dokumentation, Reviews, Freigaben von gesetzlicher und normativer Seite, die gerade im Blick auf den Umgang mit den Anforderungsspezifikationen in einer agilen Entwicklung als schwierig durchführbar erscheinen. Der Weg aus dem Dilemma, einerseits schnell und agil, andererseits aber doch tragfähig zu agieren, ist nicht immer einfach.

Der gezeigte Ansatz erfordert eine gewisse Umstellung der herkömmlichen Arbeits-

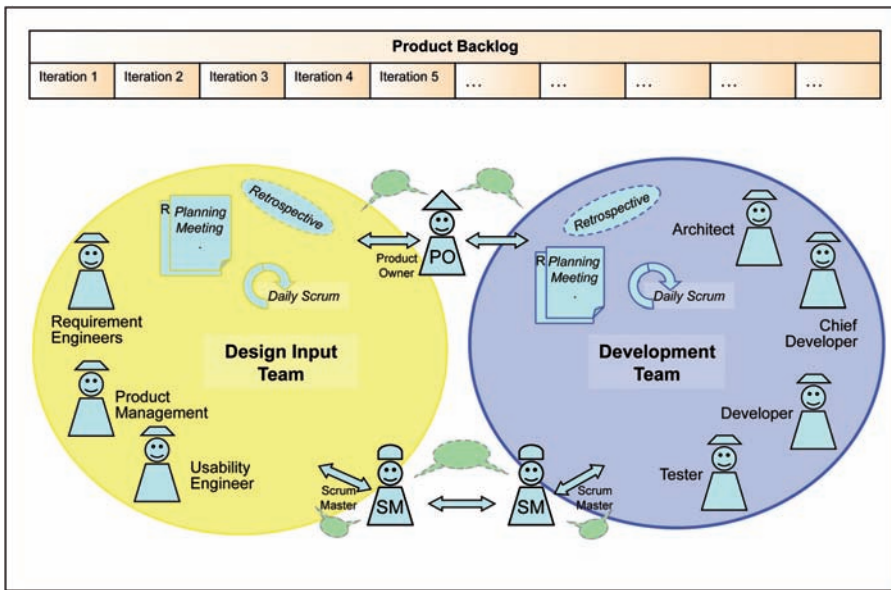


Abb. 2: Design Input-Team und Development-Team

weise und mag auf den ersten Blick mühsam oder bedenklich klingen. Die Vorteile, die sich unter anderem durch die aufgebauten, regelmäßigen Kommunikationswege

zwischen dem Entwicklungs- und dem Design Input-Team und innerhalb der Teams einstellen, werden allerdings schnell offensichtlich, sodass sich die Mühe lohnt.

Veränderungen erfordern in der Regel anfangs etwas Geduld und oft ist ein Stück Coaching hilfreich, insbesondere, um eine gute Balance zwischen beiden Welten – der agilen und der regulatorischen – zu erreichen. Ideal ist es, wenn man einen „Bergführer“ hat, der sich kompetent im Gelände – sowohl im regulatorischen als auch im agilen auskennt. Er kann einen gut zum angestrebten Ziel bringen und dabei helfen, die Vorteile möglicher Methoden optimal zu nutzen.

Zudem erleichtert die wechselseitige Unterstützung aller Stakeholder und insbesondere der beiden agilen Teams im Sinne einer fairen Partnerschaft bedeutend den Weg hin zu einem für den Kunden attraktiven Produkt.

Das Design Input-Team hat das Ohr am Kunden und übermittelt seine Erkenntnisse an das Entwicklungsteam, das daraus innovative, attraktive Lösungen entwickelt. Beide arbeiten abgestimmt und optimal verzahnt zusammen und können so unter Beachtung der gesetzlichen Vorschriften effizient und effektiv Ihre Innovationen auf den Markt bringen. ■