Ausgabe 2014 / 1

Erscheinungsart: ca. 4x jährlich in elektronischer Form

Rollen im agilen Requirements Engineering

In dieser Ausgabe:

- Was macht agiles Requirements Engineering aus?
- Beteiligte am Requirements Engineering
- Der Product Owner
- Das Entwicklungsteam
- Der Agile Master als Coach und Problemlöser
- Der Tester als Prüfer und Qualitätsberater
- Der Architekt als Berater für das Gesamtsystem
- Der Produktmanager als Dirigent mehrerer Teams

„Ich bin jetzt Product Owner für unser erstes Scrum Projekt. Da muss ich ja nichts mehr spezifizieren, oder?“

Agil bedeutet ganz und gar nicht, dass von nun an nichts mehr spezifiziert wird! Ganz im Gegenteil. In agilen Projekten wird sehr viel spezifiziert. Nur eben anders und zu anderen Zeitpunkten.

Agile Methoden, wie z.B. Scrum, ermöglichen es, flexibel auf sich ändernde Kundenwünsche einzugehen und jederzeit neue Themen aufgreifen und umsetzen zu können. Ohne Chaos. Ohne Stress. Und vor allem ohne Qualitätsverlust. Dafür arbeitet das Team das ganze Projekt über eng mit dem Kunden zusammen. So erlauben wir unseren Kunden jederzeit neue Funktion, Änderungen an den Masken, zusätzliche Daten an den Schnittstellen u.s.w. einzukippen. Wir alle lernen im Laufe des Projektes dazu. Das ist auch kein Problem, wenn unsere Methoden, vor allem in Requirements Engineering, entsprechend darauf eingerichtet sind.

Dafür gibt es ganz eigene Techniken zur Spezifikation von Anforderungen. Die klassische Spezifikation in Word wird in agilen Projekten durch schlanke Epics, User-Stories oder Use-Cases abgelöst. Hunderte Seiten dicke Word Dokumente und inkonsistente Excel Listen werden ersetzt durch ein einfach zu handhabendes Product Backlog.

Planung und Priorisierung sind in agilen Projekten sehr wichtig. Anstatt des ohnehin nie aktuellen Gantt-Projektplans gibt es ein Product- und Iteration Backlog. Der Projekt-Controllingbericht zur Verfolgung des Fortschritts der Umsetzung der Anforderungen wird abgelöst von Taskboards und Burndown Charts. Ziel ist es, mit möglichst einfachen Mitteln den Fortschritt im Team zu visualisieren und transparent zu machen.

Agile Methoden haben auch ganz eigene Rollen, die an der Erhebung, Analyse und Umsetzung der Anforderungen beteiligt sind. Jede dieser Rollen hat ihre eigenen Aufgaben und Verantwortlichkeiten und erfordert spezielle Fähigkeiten. Und gerade agile Methoden fordern von allen Beteiligten ein hohes Maß an Strukturiertheit und Disziplin!

Oberstes Ziel ist immer, den Wert der fertigen Software für den Kunden zu maximieren. Genau auf dieses Ziel hin ist das agile Requirements Engineering ausgerichtet.

Der Knowledge Letter ist ein periodisches Informationsmedium von Software Quality Lab und dessen Partnern mit den Schwerpunkten Projekt-, Prozess- und IT-Qualitäts-Management. Inhalt: fachliche Beiträge und Schwerpunktthemen, Vorstellung neuer Produkte und Leistungen, neue wissenschaftliche Erkenntnisse und andere Fachbeiträge aus unseren Themenbereichen. Aktuelle Fach- und Forschungsbeiträge sind willkommen.

Einsendungen: **info@software-quality-lab.com**

Weitere Infos zu diesem und anderen Themen: **www.software-quality-lab.com**

Was macht agiles Requirements Engineering aus?

In der klassischen Softwareentwicklung nach V-Modell oder Wasserfall werden die Requirements-Engineering-Aufwände sehr stark am Anfang konzentriert und in weiterer Folge noch ein gewisser Teil für das Änderungswesen aufgewendet. Insgesamt geht man von einem empfohlenen Aufwandsanteil von 15-20% des gesamten Projektaufwandes aus (in kritischen Projekten auch höher), um das Requirements Engineering auf einem angemessenem Niveau zu betreiben. In vielen klassischen Projekten wird diese Empfehlung leider ignoriert und der Aufwand für Requirements Engineering oft auf unter 5% gedrückt. Das ist eine der Hauptursachen vieler Projektmisserfolge.

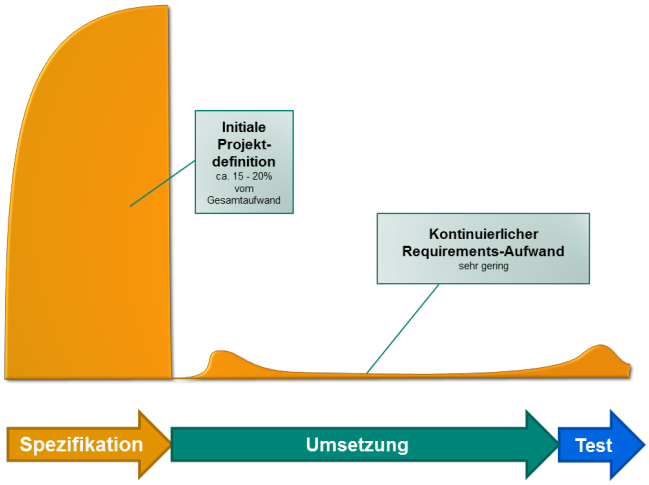

Abb. 1: Requirements-Aufwandsverteilung in klassischen Projekten

In agilen Projekten verteilt sich der Requirements-Aufwand in der Regel anders: Man geht von einer geringfügigen Spitze am Beginn des Projekts für Projektsetup und Definition der groben Anforderungen aus. Oft werden je nach Größe des Vorhabens ein bis drei Startiterationen angesetzt, die zu einem guten Teil für Anforderungsklärung verwendet werden. Um nicht in das klassische Schema „möglichst alles vorab spezifizieren“ zu kommen, sollte der initiale Aufwand für Requirements Engineering und Projekt-Setup bewusst niedrig gehalten werden und einen Anteil von ca. 5-10% des Gesamtaufwandes nicht übersteigen.

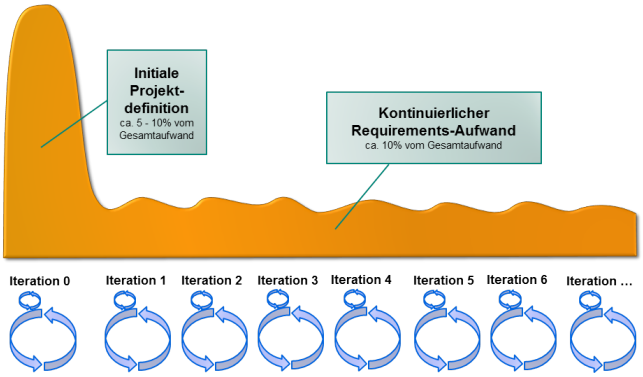

Abb. 2 Requirements-Aufwandsverteilung in agilen Projekten

Der restliche Requirements-Aufwand ist über alle folgenden Iterationen gesehen ziemlich gleichverteilt, da in jedem Sprint Definition, Klärung und Änderung von Anforderungen stattfinden. Hier ist sowohl die Analyse und Spezifikation der Anforderungen für den bzw. die kommenden Sprints gemeint, als auch die Requirements-Aufwände im Sprint Planning sowie die dann im laufenden Sprint selbst stattfindende Kommunikation zwischen Product Owner und Team zur Klärung der vielen auftauchenden Fragen. Es wird geschätzt, dass sich das gesamte Team ca. 10% seiner Zeit über die gesamte Projektlaufzeit mit der Analyse neuer und geänderter Anforderungen beschäftigt.

Auch in agilen Vorgehensweisen kann man also davon ausgehen, dass man durch die Anfangsiterationen und die kontinuierlichen Aufwände für die Requirements-Themen über die gesamte Projektlaufzeit auf einen Aufwand von ca. 15-20% kommt. Das ist in etwa gleich oder etwas geringer, als in klassischem Vorgehen empfohlen. Der wesentliche Unterschied ist, dass die Aufwände anders verteilt sind und genau dort anfallen, wo sie den größten Nutzen haben.

Beteiligte am Requirements Engineering

Ein wesentlicher Erfolgsfaktor für gut laufende Projekte ist die intensive Einbindung und Kommunikation mit den Fachleuten und Nichttechnikern im Requirements Engineering während des gesamten Projekts. Immer sind unterschiedliche Personen im Fachbereich des Kunden und im Team an der Erhebung, Spezifikation und Ausarbeitung von Anforderungen beteiligt. Abbildung 3 gibt einen Überblick über die beteiligten Rollen im Kunden- bzw. Fachbereich und im agilen Team.

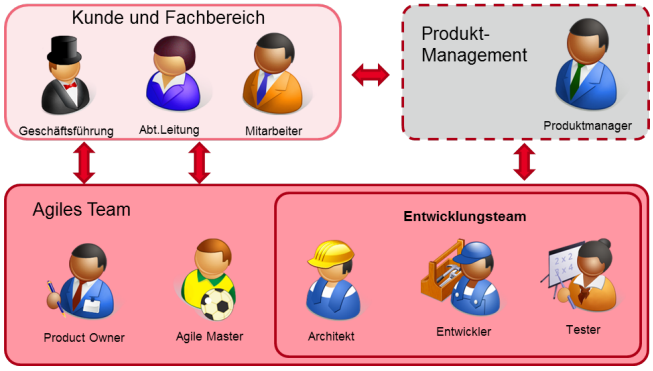

Abb. 3 Unterschiedliche Rollen im Fachbereich und im Entwicklungsteam

Der Product Owner als Stellvertreter des Kunden

Der Product Owner (PO) ist für die Anforderungen im Product Backlog und deren Priorisierung verantwortlich. Er schreibt die Anforderungen und trifft als deren „Besitzer“ alle inhaltlichen Entscheidungen. Er sorgt dafür, dass ständig

Vollständiger Knowledge Letter Zugang

Wir freuen uns, dass Sie an diesem Thema Interesse haben und den Knowledge Letter von Software Quality Lab bis hierher gelesen haben.

Dieser Knowledge Letter ist eine Vorschau (gekürzte Version des gesamten Artikels).

Wenn Sie den ungekürzten Knowledge Letter lesen möchten, registrieren Sie sich bitte unter
http://www.software-quality-lab.com/download/knowledge-letter/anfrage-knowledge-letter/

Sie erhalten nach der Registrierung vollen Zugang zu allen bisherigen Knowledge Lettern von Software Quality Lab und erhalten automatisch künftige Knowledge Letter per E-Mail.

Software Quality Days — Die größte Konferenz zum Thema „Software Qualität“ in Europa!

Besuchen Sie die Top-Konferenz mit allen Infos rund um Software Qualität.

Beste Qualität der Vorträge und Tutorials sowie eine Mischung aus praktischen und wissenschaftlichen Beiträgen machen die Software Quality Days zum Top-Event.

In den 3 praktischen Tracks werden anwendungsorientierte Vorträge präsentiert. Der wissenschaftliche Track zeigt Beiträge mit hohem Innovationsgrad und praktischer Anwendbarkeit, basierend auf Forschungs-ergebnissen. Im Solution Provider Forum präsentieren Aussteller ihre neuesten Tools mit Praxis-Beispielen.

Nähere Infos unter

www.software-quality-days.com