



Das agile Requirements Board

Ein Tool zur Unterstützung des agilen Requirements-Engineerings

Johannes Bergmann

Berater, Trainer, Eigentümer



Innovation Meets Quality

Über Software Quality Lab

LEISTUNGSPORTFOLIO



CONSULTING

- Management Consulting
- Prozesse und Vorgehensmodelle
- Teststrategie und -konzeption
- Requirements und Ausschreibungen
- Architektur und Modellierung
- Code Analyse und Metriken



OPERATIONAL SERVICES & TESTCENTER

- TestCenter
- Requirements Engineering
- Testmanagement und -spezifikation
- Softwareverifikation und -validierung
- Testautomatisierung und -reporting
- Reviews und Code Analyse



ACADEMY

- Requirements und Usability
- Architektur und Modellierung
- Testen und Automatisieren
- Agile Softwareentwicklung
- Projektabwicklung und Vorgehensmodelle
- Lehrgänge und Zertifizierungen



TOOL EXPERTISE

- Tool Evaluation Center
- Tool-Einführung und Pilotprojekte
- Tool-Schnittstellen und –Einbindung
- Managed Services und Lizenzen
- Softwareprozessautomatisierung
- Test Automation Frameworks
- Tool-Studien und Tool-Expertisen

Energie & Versorgung



Elektronik & Automation



Industrie & Engineering



Dienstleistungen & Handel



Medizin & Pharma



Software & IT



Finanz & Versicherung



Mobilität & Kommunikation



- Warum wir noch ein Tool brauchen
- Requirements Analyse in agilen Vorgehensweisen
- Requirements Board
- Übung

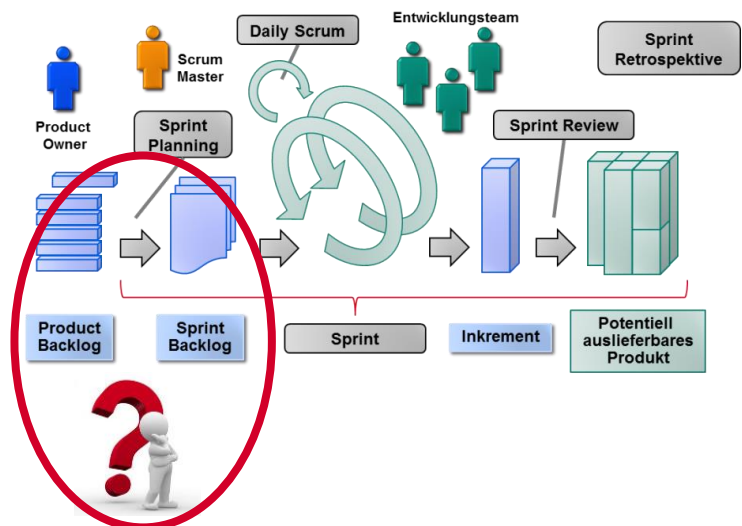
Warum wir noch ein Tool brauchen

Grenzen des „klassischen“ Story-Managements in Scrum

Wie wird RE in Scrum umgesetzt?

Fragen bei RE in Scrum:

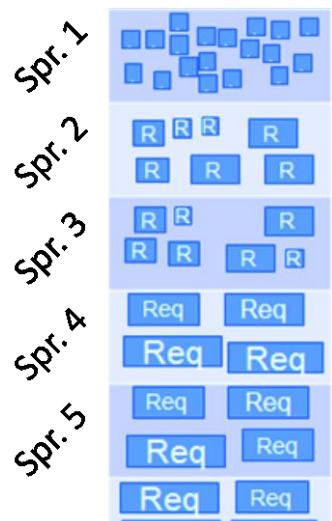
- Wie „entstehen“ Requirements (Epics, Stories, Features, etc.)?
- Wie erfolgt die Analyse, Klassifikation, etc.?
- Wie kommen REs vom Product Backlog in den Sprint Backlog?



„klassische“ Elemente des RE in Scrum

RE-Elemente in Scrum Product Backlog

- **Alle Requirements** an einer Stelle gesammelt
 - Auswahlbasis für Sprint Backlog
 - Umsetzung oft als einfache Liste (z.B. Excel, Word), in Taskmanagement-Tools (z.B. Jira) oder eigenes Backlog-Board mit Kärtchen
 - Definition und Anordnung erfolgt durch PO (ggf. in Abstimmung mit den Stakeholdern)
- Systematische Klassifizierung und Zusatzinfos werden tw. vernachlässigt





- = Eingangskriterium für den Sprint-Backlog:
- **Quality Gate für Requirements**
 - Kriterien, wann ein Backlog Item für die Umsetzung bereit ist.
 - **Sichert Mindestqualität**
 - **Zwischen Team und PO VOR dem Projektstart vereinbart**
 - **Laufend (z.B. in der Sprint-Retrospektive) reviewen und anpassen**

- Requirements (Stories) des Sprintbacklogs in der ersten Spalte
 - **Status der Umsetzung** in den restlichen Spalten
- Wie es zu der Auswahl der Requirements am Taskboard kommt, ist ev. unklar?

Zeiterfassungssystem		Sprint 5			
Stories	Nicht begonnen	In Arbeit		fertig (Done)	
<p>Story 1: „Als Mitarbeiter möchte ich meine Tages- und Wochenzeiten erfassen, damit ich gegenüber dem Arbeitgeber meine Arbeitszeiten nachweisen kann.“</p>	<p>Task 1.0: 2h Systemanforderungen für Personalisierung des Front-Ends für Abwesenheitszeiten erheben</p>	<p>Task 1.1: 1h Systemanforderungen und Feature-Definition programmieren</p>	<p>Task 1.2: 4h Datenmodell für Zeiterfassung erstellen</p>	<p>Task 1.3: 4h Frontend-Entwicklungsarbeiten für die Darstellung von den erfassten Arbeitszeiten und dem Abwesenheitsstatus erstellen</p>	<p>Task 1.4: 2h Frontend-Entwicklungsarbeiten für die Darstellung von den erfassten Arbeitszeiten und dem Abwesenheitsstatus erstellen</p>
<p>Story 2: „Als Mitarbeiter möchte ich die Tages- und Wochenzeiten in einem Programm visualisieren können, damit die Projektleitung und die Projektschichtung darauf erstellen kann.“</p>	<p>Task 1.5: 2h Systemanforderungen mit UI-System analysieren</p>	<p>Task 1.6: 4h Datenmodell erstellen und für die Zeiterfassung berücksichtigen</p>		<p>Task 1.7: 2h Frontend-Entwicklungsarbeiten für die Darstellung von den erfassten Arbeitszeiten und dem Abwesenheitsstatus erstellen</p>	
<p>Story 3: „Als Anwender möchte ich ...“</p>		<p>Task 1.8: 1h Systemanforderungen und Feature-Definition programmieren</p>		<p>Task 1.9: 4h Datenmodell für Zeiterfassung erstellen</p>	<p>Task 1.10: 4h Frontend-Entwicklungsarbeiten für die Darstellung von den erfassten Arbeitszeiten und dem Abwesenheitsstatus erstellen</p>
<p>Story 4: ...</p>	<p>Task 1.11: 2h Frontend-Entwicklungsarbeiten für die Darstellung von den erfassten Arbeitszeiten und dem Abwesenheitsstatus erstellen</p>	<p>Task 1.12: 1h Systemanforderungen mit UI-System analysieren</p>			



Innovation Meets Quality

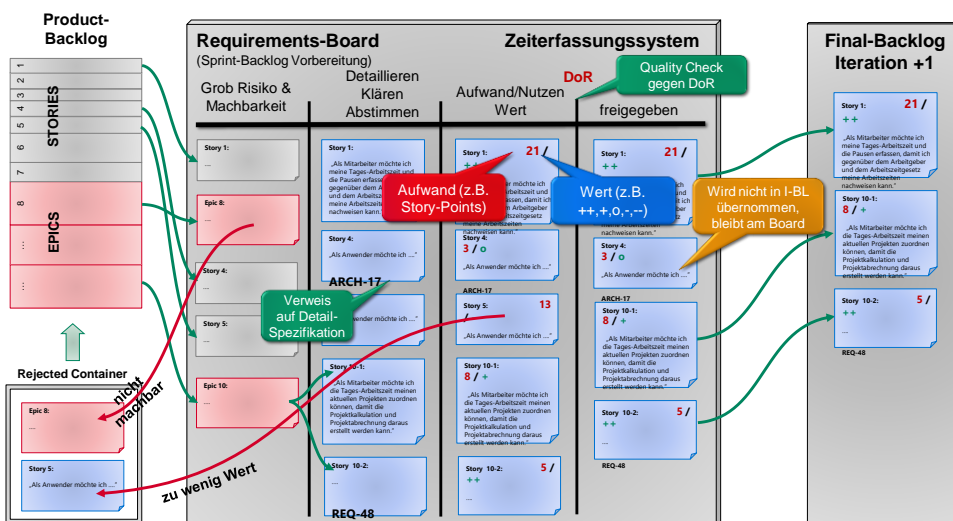
Requirements Board

zur Unterstützung der Analyse und Klassifizierung



Innovation Meets Quality

Requirements Board – inkl. RE-Stages





- Systematische Requirements-Analyse auf dem Weg eines REQ vom Product Backlog in den Sprint Backlog ist sinnvoll (z.B. für besseres Verständnis und Einschätzung)
- Passendes Requirements-Management Tool ist dazu notwendig (nur eine Liste ist zu wenig)
- „Agiles RE-Board“ kann Taskboard sinnvoll ergänzen und das RE in agilen Projekten unterstützen.

- **Requirements-Engineering für die agile Softwareentwicklung**

dpunkt.verlag

- auch als Seminar verfügbar!





Klein-
gruppen-
Zeit bis
14:00

- Grober unsortierter Product-Backlog ist als Beispiel vorhanden (ggf. auch selbst was überlegen), RE-Board und kleine Story-Cards sind vorhanden
 - 3-4er Gruppen bilden / zusammenarbeiten
 - Ggf. auch unterschiedliche Sichten einnehmen:
 - PO definiert Nutzen, möchte möglichst viel umgesetzt bekommen;
 - Entwickler/Architekten schätzen Machbarkeit und Entwicklungs-Aufwand ein
 - Qualitätssicherung achtet auf Qualität und Testbarkeit
- Beschäftigen Sie sich mit dem RE-Board, Diskutieren Sie untereinander, Probieren Sie es ev. auch aus mit den kleinen Storycards
- Ab 14 Uhr: Diskussion bzw. F&A im Plenum

Software Quality Lab
INNOVATION MEETS QUALITY

Academy | Consulting | Operational Services | Tool Expertise