

Kontinuierliche Architekturbewertung durchführen

Bereich

Evaluation/Kontrolle

Aktivität

Machbarkeitsanalyse/
Interne Qualitätssicherung

Ziele

- Inkonsistenzen und offene Punkte in der Lösung identifizieren und deren frühzeitige Bearbeitung ermöglichen
- Architekturlösungen nachhaltig bewahren
- Konfidenz in die Eignung der entwickelten Architekturlösungen erreichen
- Wissensverteilung über Architekturkonzepte im Team fördern

schnell durchführbar ★★★☆☆

einfach durchführbar ★★★☆☆

agil einsetzbar ★★★☆☆

Motivation/Problemstellung

Ein typisches Merkmal von agilen Projekten ist der hohe Grad an Änderungen. In agiler Entwicklung werden Änderungen der Anforderungen und die stetige Anpassung des Systems darauf als essentieller Teil der Entwicklung angesehen. Dies ist jedoch auch eine Herausforderung. In einem hinreichend komplexen Entwicklungsprojekt sind die gegenseitigen Auswirkungen und Abhängigkeiten von Lösungskonzepten oft nur schwierig vollständig zu überblicken. Wird ein Lösungskonzept für neue oder geänderte Anforderungen entwickelt, kann es daher einerseits dazu kommen, dass Lösungskonzepte als geeignet eingestuft werden, dabei aber wesentliche Punkte übersehen werden, die die Erfüllung der bearbeiteten Anforderungen unmöglich machen. Andererseits kann ein grundsätzlich geeignetes Konzept solch große Auswirkungen auf andere Lösungskonzepte haben, dass die Erfüllung von wichtigen, querschnittlichen Anforderungen beeinträchtigt wird. Inkonsistenzen und ungeeignete Konzepte werden dann von Entwicklern häufig zufällig während der Implementierung aufgedeckt oder es kommt zu unerklärlichem Verhalten oder Fehlern. Das Ergebnis ist ein weniger wartbares System und zusätzlicher, ungeplanter Aufwand für Korrekturarbeiten.

Architekturbewertungen können helfen solche Inkonsistenzen in Architekturlösungen früher aufzudecken und mit den Auswirkungen umzugehen. Ansätze zur Architekturbewertung wie ATAM [1] oder RATE [2] können aufgrund der Eigenschaften von agiler Entwicklung häufig nicht unverändert eingesetzt werden. Eine stark fokussierte Architekturbewertung für die wichtigsten Anforderungen kann jedoch auch in zeitlich beschränkten Situationen gut durchgeführt werden. Werden solche Bewertungen wiederholt und regelmäßig durchgeführt, können sie einen großen Mehrwert für agile Projekte darstellen.

Kurzbeschreibung

Die Eignung eines Lösungskonzeptes für die Erfüllung von bestimmten Anforderungen soll geprüft werden. Zusätzlich zu diesen können noch weitere, wichtige Anforderungen ausgewählt werden, deren Erfüllung bei der Realisierung des neuen Konzeptes bewahrt bleiben soll. Ein Teammitglied wird als Assessor ausgewählt, der die Architekturbewertung durchführt. Dieser befragt ausgewählte Mitglieder des Teams über die Lösungen, die die gewählten Anforderungen erfüllen sollen. Dabei versucht er offene Punkte, Inkonsistenzen und Verbesserungspotentiale herauszufinden. Für diese werden Vorschläge für Stories erstellt, die diskutiert und bei Bedarf in das Backlog eingeplant werden.

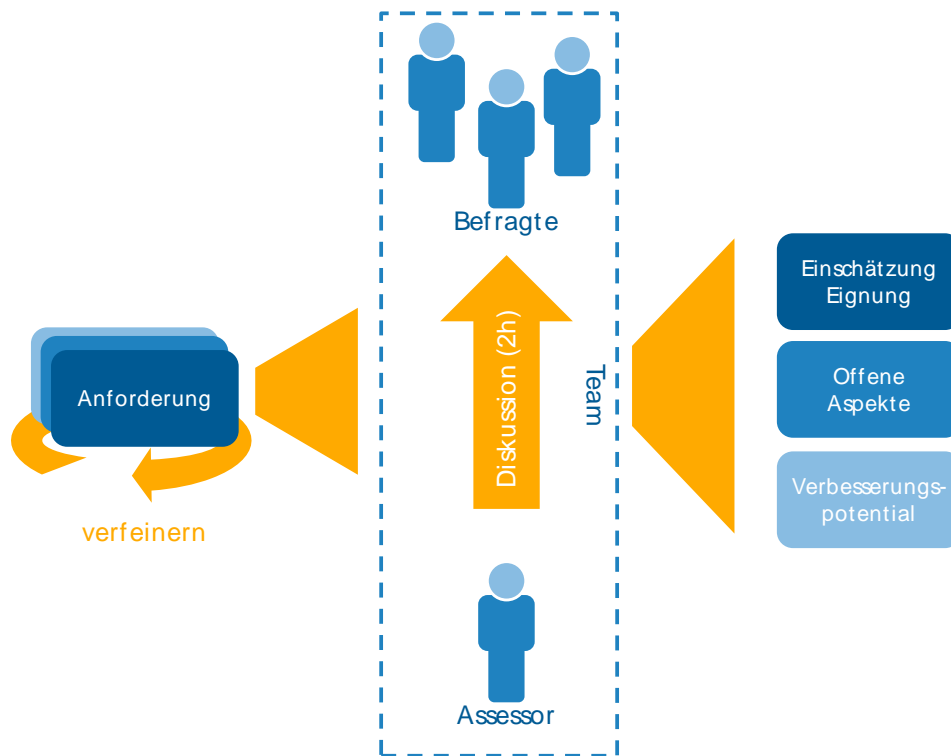


Abbildung 1:Kontinuierliche Architekturbewertung

Input

- Eine Menge von Anforderungen, für die das Konzept entwickelt wurde
- Zusätzliche wichtige oder querschnittliche Anforderungen

Output

- Eine Einschätzung über die Eignung des Systems hinsichtlich der analysierten Anforderungen
- Eine Menge von offenen Punkten und Verbesserungspotentialen für das System

Rahmenbedingungen

Ausführender

Ein Teammitglied als Assessor

Werkzeuge, Hilfsmittel

Vorkenntnisse/Erfahrungen

Guter Systemüberblick

Ort/Umgebung

Besprechungsraum mit Whiteboard

Weitere Teilnehmer

Teammitglieder, die an der Entwicklung der Lösungen beteiligt waren

Voraussichtliche Dauer

Etwa zwei Stunden pro
Bewertungsbesprechung plus etwa ein bis zwei Stunden für Vor- und Nachbereitung

Vorgehensweise

Vorbereitung

Lösungskonzepte werden für eine Menge von bestimmten Anforderungen häufig während eines Sprints in Vorbereitung für die Umsetzung im Folgesprint erarbeitet. Ein solches Lösungskonzept steht dann im Fokus der Bewertung. Der erste Vorbereitungsschritt dieser Best Practice ist die Auswahl von relevanten Anforderungen, gegen die das Lösungskonzept evaluiert werden soll. Dies sind zum einen immer die Anforderungen, für die das Konzept entwickelt wurde. Für diese Anforderungen soll die Eignung des Konzepts geprüft werden. Andererseits können zusätzlich besonders wichtige oder querschnittliche Anforderungen einbezogen werden. Dies sind häufig Anforderungen, deren Erfüllung in jedem Release nur schwer einzuschätzen ist, da man diesbezüglich kein direktes Feedback erhält. Dies ist besonders bei vielen Qualitätsanforderungen der Fall. Ob das System noch in jeder Situation die gewünschte Skalierbarkeit, Verfügbarkeit und Wartbarkeit aufweist, ist schwerer zu bestimmen als die korrekte Arbeitsweise einer Funktionalität. Aber auch Funktionalitäten, die nicht sehr häufig verwendet werden, können darunter fallen.

Während der Bewertung übernimmt ein Entwickler die Rolle des Assessors. Idealerweise hat dieser nicht direkt bei der Entwicklung und Realisierung des Lösungskonzeptes für die untersuchten Anforderungen mitgewirkt, um nicht voreingenommen zu sein. Einer oder mehrere Entwickler sind die Befragten.

Um die eigentliche Bewertung vorzubereiten, prüft der Assessor noch einmal die Anforderungen, gegen die bewertet werden soll. Diese sollten in einer Form vorliegen, die eine Bewertung der Erfüllung ermöglicht. Das heißt, statt „Das System soll leicht erweiterbar sein“, sollte eine Anforderung so konkret wie möglich als beispielsweise „Ein erfahrener Entwickler kann eine neue externe Datenquelle innerhalb eines Tages integrieren“ formuliert sein. Architekturszenarien [3] sind eine Möglichkeit um konkrete und prüfbare Qualitätsanforderungen zu erhalten. Liegen die Anforderungen nicht in einer entsprechenden Form vor, übernimmt der Assessor die Verfeinerung und Dokumentation der Anforderungen.

Durchführung

Nachdem konkrete Anforderungen verfügbar sind, kann die Bewertung durchgeführt werden. Dafür organisiert der Assessor ein Meeting und lädt dazu einen oder mehrere Entwickler ein, die zum bewerteten Lösungskonzept beigetragen haben. Während der Besprechung diskutiert der Assessor mit den Teammitgliedern jede Anforderung nacheinander. Das Team erklärt dem Assessor wie die Lösung die jeweiligen Anforderungen erfüllt. Ähnlich einem Cognitive Walkthrough aus dem Bereich User Experience spielt man dabei mental den gesamten Ablauf durch, mit dem das System die genannten Anforderungen erfüllt. Der Assessor fordert das Team heraus und versucht vergessene Aspekte, Inkonsistenzen und offene Punkte zu finden. Dabei macht sich der Assessor Notizen, insbesondere in Hinblick auf die gefundenen offenen Punkte. Dies wird für jede Anforderung wiederholt, bis alle Anforderungen besprochen wurden oder der geplante Zeitrahmen aufgebraucht ist.

Im Fall dass nicht alle Anforderungen besprochen werden konnten, ist es am Assessor abzuwägen, ob die verbleibenden Anforderungen bei der nächsten Bewertung

besprochen werden sollten, oder ob das Risiko gering ist, wenn Anforderungen ausgelassen werden.

Nachbereitung/Auswertung

Als Nachbereitungsschritt konsolidiert der Assessor die Informationen, die er sich während der Besprechung notiert hat. Insbesondere für die offenen Punkte werden Vorschläge für Stories formuliert, die mit dem Product Owner und dem Team besprochen und priorisiert werden. Eine Kurzfassung der Bewertungsergebnisse kann den Teammitgliedern beim folgenden Stand-up (oder ähnlichen) Meeting mitgeteilt werden. Die ausführlichen Ergebnisse können beispielsweise während des nächsten Planungsmeetings (erster Teil) am Anfang der folgenden Iteration präsentiert werden, um etwaige Nacharbeiten zu koordinieren.

Gütekriterien/Empfehlungen

Für den Erfolg dieser Best Practice ist es essentiell die richtigen Teammitglieder für die teilnehmenden Rollen auszuwählen, insbesondere für den Assessor. Er benötigt eine sehr gute Übersicht über das Gesamtsystem und muss in der Lage sein, über die verschiedenen Aspekte nachzudenken, die die konzeptionellen Lösungen und das System insgesamt ausmachen. Dies ist eine Voraussetzung um offene Punkte in den diskutierten Lösungen finden zu können. Aus diesem Grund sollte dafür auch ein erfahrenes Teammitglied ausgewählt werden.

Die Interviewteilnehmer auf der anderen Seite sollten die relevanten Lösungen im Detail kennen und sollten zu deren Entwicklung beigetragen haben. Sie sollten bereit sein, offen über die Lösungen zu sprechen, insbesondere wenn Verbesserungspotential diskutiert wird. Wichtig dabei ist, dass beim Entdecken von Verbesserungspotential während der Diskussion nur verbesserungswürdige Aspekte besprochen und nicht direkt Lösungsideen ausgearbeitet werden. Dies würde für die Bewertung zu viel Zeit in Anspruch nehmen und sollte im Nachgang erfolgen.

Risiken

Die Durchführung dieser Best Practice ist nur sinnvoll, wenn die erzielten Ergebnisse (Risiken, offene Punkte, Verbesserungspotential) in der weiteren Entwicklung des Produktes verwendet werden. Wird dies nicht getan, wird nur in die Bewertung investiert, ohne später davon zu profitieren. Das heißt, für die Bearbeitung identifizierter Verbesserungspotentiale sollten explizit Stories erstellt und eingeplant werden. Die Zeit zur Vorbereitung und Durchführung der Bewertung muss als Teil der Entwicklung angesehen werden und damit als wertvolle Investition in die Qualität des Produktes. Ansonsten besteht die Gefahr, dass die Best Practice nicht sinnvoll durchgeführt wird und damit kein Mehrwert für das Projekt erzielt wird.

Einordnung in den agilen Referenzprozess

Mögliche Vorgänger

- Architektur\Architekturentscheidungen dokumentieren
- Architektur\Architekturelevante Anforderungen dokumentieren

Mögliche Nachfolger

- Testen\Reviews von Entwicklungsartefakten durchführen
- Architektur\Architekturlösungen im Team entwickeln

Mögliche Alternativen, verwandte Praktiken

- Unterstützung\Standardisierte Dokumentstrukturen verwenden
- Architektur\Feature-Entwicklung und interne Qualitätsaufgaben verzahnen

Einordnung in das PQ4Agile-Qualitätsmodell

Grundsätzlich alle Produktqualitäten auf die die entwickelten Architekturlösungen abzielen

Schlagworte

Architekturbewertung, kontinuierlich, Architekturkonzept

Weiterführende Informationen

Literatur

- [1] Len Bass; Paul Clements; Rick Kazman, "Software Architecture in Practice", Second Edition Addison Wesley Professional, April 9, 2003
- [2] Knodel, J.; Naab, M., "Software Architecture Evaluation in Practice: Retrospective on More Than 50 Architecture Evaluations in Industry," *Software Architecture (WICSA), 2014 IEEE/IFIP Conference on* , vol., no., pp.115,124, 7-11 April 2014
- [3] Rozanski, N., & Woods, E. (2005). *Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives*. Addison-Wesley Professional.