

UX4Agile

Integration von Best Practices des Usability- und User-Experience-Engineering in agile Entwicklungsprozesse

Steffen Hess, Dominik Rost

Fraunhofer IESE

Fraunhofer-Platz 1

67663 Kaiserslautern

{steffen.hess, dominik.rost}@iese.fraunhofer.de

Hartmut Schmitt

HK Business Solutions GmbH

Mellinweg 20

66280 Sulzbach

schmitt@hk-bs.de

Abstract

Agile Entwicklungsprozesse erfordern auch ein Umdenken bei der Arbeit von Usability Professionals. Der Workshop UX4Agile soll daher die Frage adressieren, wie Best Practices aus dem Usability- und User-Experience-(UUX-) Engineering möglichst einfach in agile Projekte integriert werden können. Darüber hinaus wird betrachtet, wie Ursachen für UUX-Mängel in agilen Projekten gezielt identifiziert werden können und durch existierende oder neu zu entwickelnde UUX-Praktiken gelöst werden können.

Das Forschungsprojekt PQ4Agile befasst sich mit den o.g. Fragestellungen aus Sicht von KMU der Softwarebranche. Im Workshop sollen erste Erkenntnisse des Projektes mit den Teilnehmern geteilt werden und mit deren eigenen Erfahrungen abgeglichen werden, um weitere praxisnahe Erkenntnisse zu sammeln.

Ergebnis des Workshops sind UUX-Best-Practices, die in agilen Projekten angewendet werden können, sowie ein Qualitätsmodell für UUX in agilen Projekten, das in eigenen Projekten angewendet werden kann.

Keywords

Agile, Qualitätsmodell, UX Best Practices

1. Einleitung

Agile Entwicklungsvorgehen haben heute eine weite Verbreitung erreicht: Viele Softwarefirmen und Agenturen erhoffen sich ein höheres Maß an Flexibilität, mehr Interaktion mit den Kunden und ein dynamischeres Reagieren auf geänderte Anforderungen. Kundennähe, Kundenfeedback und hohe Kundenzufriedenheit sind also zentrale Aspekte der agilen Entwicklung – ebenso wie des UUX Engineering.

Trotzdem stehen beide Ansätze scheinbar im Widerspruch zueinander: In agilen Projekten findet oftmals eine Fokussierung auf funktionale Anforderungen statt, so dass nichtfunktionale Aspekte wie Usability oder UX in den Hintergrund treten. Außerdem mag die in agilen Projekten übliche Entwicklung in kurzen Iterationszyklen nicht zu den zusammenhängenden Methoden und ganzheitlichen Betrachtungsweisen des UUX Engineering passen. Dadurch kommt es bei der Anwendung agiler Entwicklungsmethoden oft zu Schwierigkeiten oder UUX-Mängeln, die sich unmittelbar auf Kundenbeziehungen, Absatzchancen und Marktpositionen der Unternehmen auswirken. Im Forschungsprojekt PQ4Agile wird daher eine systematische Unterstützung für Anwender agiler Methoden entwickelt, die einen breiteren und erfolgreichen Einsatz dieser Methoden ermöglicht und die zu einer vorhersagbar hohen Qualität der entwickelten Produkte beiträgt.

Die Aktivitäten des Entwicklungsprozesses haben einen direkten Einfluss auf die Qualität der daraus resultierenden Produkte. Daher ist eine zentrale Kernidee des Projektes, agile Entwicklungsprozesse so durch etablierte Aktivitäten des Usability- und User-Experience-Engineerings anzureichern, dass hierdurch eine ganzheitliche Qualitätsbetrachtung und eine vorhersagbar hohe Produktqualität in agilen Entwicklungsprojekten möglich werden. Hierfür ist ein einheitliches Qualitätsverständnis notwendig, wofür das folgend beschriebene Qualitätsmodell geschaffen wurde.

2. Qualitätsmodell

Das Qualitätsmodell umfasst alle relevanten Aspekte von Qualität im Softwareentwicklungsprozess. Die Qualität der entwickelten Produkte ist dabei von zentraler Bedeutung, kann aber nicht in Isolation betrachtet werden. Das Qualitätsmodell muss zusätzlich zu Produktqualitäten auch die Qualitäten von Entwicklungsprozessen umfassen, um die durch Veränderungen am Entwicklungsprozess verursachten Auswirkungen auf die Prozessqualitäten messen zu können. Darüber hinaus haben auch strukturelle Aspekte und deren Qualität, wie die vorhandene technische Infrastruktur oder die verfügbaren personellen Ressourcen einen Einfluss auf die resultierende Produktqualität. Diese müssen daher ebenso berücksichtigt werden.

Die erste Strukturierungsebene des Qualitätsmodells bilden die drei Qualitätsbereiche Produktqualität, Prozessqualität und Strukturqualität mit ihren einzelnen Qualitätsattributen. Für jeden dieser Bereiche dient ein bestehendes Qualitätsmodell als Basis: für den Bereich Produktqualität die ISO 25010 [1], für den Bereich Prozessqualität das Prozessqualitätsmodell von Kneuper [2] und für den Bereich Strukturqualität die ISO 9001 [3]. Diese Qualitätsmodelle definieren Qualitätsattribute der jeweiligen Bereiche auf zwei

Hierarchiestufen als Qualitätsmerkmale und Qualitätsteilmerkmale, die in Qualitätsmerkmalsgruppen zusammengefasst werden können. Im PQ4Agile-Qualitätsmodell werden diese übernommen und Verknüpfungspunkte und Überlappungen explizit gemacht. Das resultierende Modell deckt damit alle relevanten Bereiche in zusammenhängender Weise ab.

In einem konkreten Anwendungskontext können diese Bereiche nicht unabhängig voneinander betrachtet werden, sondern sind eng miteinander verbunden und bedingen sich gegenseitig. Abbildung 1 illustriert diese Zusammenhänge:

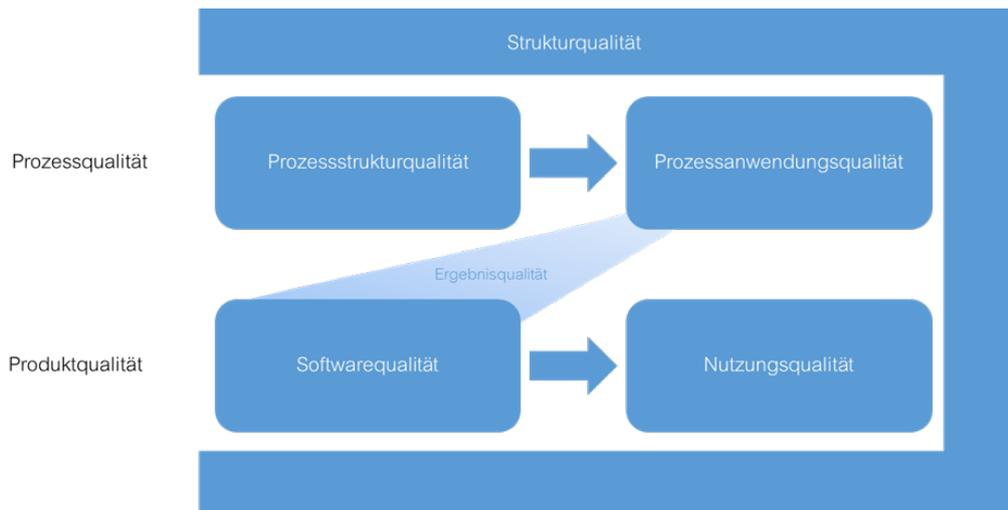


Abbildung 1: Qualitätsmodell

Bei der Prozessqualität kann zwischen Strukturqualität und Anwendungsqualität unterschieden werden. Dabei umfasst die Strukturqualität Attribute, die sich auf den Entwicklungsprozess als solchen beziehen, insbesondere auch auf den Prozess als geplantes Artefakt. Während der Anwendung bzw. Ausführung eines solchen Prozesses sind hingegen die Attribute der Anwendungsqualität von Relevanz. Dies bedeutet, dass die Qualitäten des angewendeten Prozesses maßgeblich von den Strukturqualitäten bestimmt sind, aber auch von anderen Faktoren beeinflusst werden können. Während der Ausführung eines Entwicklungsprozesses werden verschiedene Ergebnisse, wie beispielsweise Lastenhefte, Pflichtenhefte, Architekturdokumentationen, Testspezifikationen, Projektsteuerungsartefakte usw. produziert. Diese Ergebnisse des Entwicklungsprozesses besitzen jeweils eine spezifische Qualität. Diese Qualität bezieht sich auf das Attribut Ergebnisqualität der Prozessanwendungsqualität. Handelt es sich bei dem Ergebnis um das Endergebnis des Entwicklungsprozesses, also das Softwareprodukt selbst, entspricht die Ergebnisqualität der Produktqualität, die im Qualitätsbereich Produktqualität dargestellt ist. Darin wird analog zum Bereich Prozessqualität zwischen der Qualität des Produktes als solches und der Qualität bei der Nutzung unterschieden. Die Qualität der strukturellen Gegebenheiten bildet den dritten Bereich des Qualitätsmodells. Ein Entwicklungsprozess ist immer in einem strukturellen Umfeld eingebettet, wird also innerhalb einer gegebenen Infrastruktur, mit personellen Ressourcen und in einer gegebenen Arbeitsumgebung ausgeführt. Zu beachten ist, dass die Strukturqualität auch die Nutzungsqualität beeinflussen kann, z.B. durch die Qualität von

Dienstleistungen, die während des Betriebs einer Software erbracht werden. Die Qualität der gegebenen Struktur beeinflusst also direkt oder indirekt alle zuvor beschriebenen Qualitätsbereiche. Aus diesem Grund ist die Strukturqualität in Abbildung 1 als Umfeld der anderen Qualitätsbereiche dargestellt.

3. Agiler Referenzprozess

Unternehmen setzen agile Entwicklungsmethoden in vielen verschiedenen Varianten ein, kaum eine folgt vollständig dem Lehrbuch. Meist werden bei der Umstellung auf agiles Vorgehen verschiedene Techniken ausprobiert, verworfen, verändert und angepasst. So entsteht in den Unternehmen ein individueller Entwicklungsprozess basierend auf den Erfahrungen und spezialisiert auf die Gegebenheiten und Anforderungen des gegebenen Arbeitskontextes.

In PQ4Agile ist eine umfassende Praxisrelevanz von höchster Priorität. Um diese zu erreichen müssen UUX Best Practices entwickelt werden, die auf in Unternehmen gelebten Entwicklungsprozessen basieren. Dafür wurden die Entwicklungsprozesse der Industriepartner des PQ4Agile-Projektes erhoben und analysiert. Erhoben wurden dabei die durchgeführten Aktivitäten, beteiligte Rollen, durchgeführte Rituale/Meetings und eingesetzte Werkzeuge in den Bereichen Anforderungen, Planung und Design, Evaluation, Realisierung, Kontrolle und Projektplanung und Steuerung. Daraus wurde ein Referenzprozess entwickelt, der die Gemeinsamkeiten der verschiedenen Prozesse konsolidiert und auf die spezifischen Umsetzungen bei den Partnern verweist. Damit wird sichergestellt, dass bei der Entwicklung der UUX-Best-Practices eine hohe Verallgemeinerbarkeit gegeben ist, jedoch bei vorhandener Verbindung zu den Spezifika der einzelnen Unternehmen. Um zu vermeiden, dass der Referenzprozess essentielle Aktivitäten unberücksichtigt lässt, die aus bestimmten Gründen bei den Unternehmen nicht angewendet werden, wurde zusätzlich ein Vergleich zwischen dem Referenzprozess und Scrum [Ref Scrum Guide], Software Engineering Literatur [Ref Balzert] und CMMI [Ref CMMI] durchgeführt und der Referenzprozess wurde entsprechend ergänzt.

Abbildung 2 illustriert den entwickelten Referenzprozess. Dieser umfasst ausschließlich Aktivitäten, da die erhobenen Rollen, Meetings und Werkzeuge eine zu hohe Spezifität aufweisen, um eine Verallgemeinerbarkeit zu erreichen. Die vorhandenen Aktivitäten werden in die zuvor erwähnten Bereiche unterteilt. Folgend werden diese Bereiche und die zugehörigen Aktivitäten näher erläutert:

- *Anforderungen*: Die Verwaltung und Bearbeitung von Anforderungen an das Softwareprodukt.
- *(Kunden-)Anforderungen erheben*: Die Erfassung von Anforderungen an das Softwareprodukt. Dabei handelt es sich meist um Anforderungen des Kunden, die beispielsweise in gemeinsamen Workshops erhoben werden. Ebenso fällt darunter aber auch die Erhebung von Anforderungen aus anderen Quellen, wie beispielsweise aus rechtlichen Bestimmungen.

- *(Kunden-)Anforderungen / Anforderungsänderungen bearbeiten:* Nach der initialen Erfassung werden Anforderungen konsolidiert, aufbereitet, strukturiert und dokumentiert. Dies trifft sowohl für neue Anforderungen wie für geänderte Anforderungen zu.
- *(Kunden-)Anforderungen einarbeiten:* Herunterbrechen von Anforderungen in Einzelaufgaben und Einplanen in die Entwicklung, beispielsweise die Übernahme in das Projektbacklog.
- *Planung und Design:* Die Erstellung von konzeptionellen Lösungen für die gegebenen Anforderungen. Darunter sind sowohl Architektur- als auch User-Experience-Aktivitäten zu verstehen.
 - *Lösungskonzept entwerfen:* Die Erstellung von fachlichen bzw. technischen Lösungskonzepten und Treffen von Architekturentscheidungen.
 - *Oberflächenentwurf erstellen:* Die Erstellung von Visual Designs und Interaktionskonzepten.
 - *Softwareinkrement entwerfen:* Die detaillierte Planung der Implementierung auf Basis der Lösungskonzepte.
 - *Änderungskonzept entwerfen:* Analog zum Lösungskonzept wird hier die Erstellung von technischen Lösungskonzepten verstanden. Zusätzlich ist dabei aber die bestehende Konzeption und Umsetzung zu beachten und im Lösungskonzept muss dafür eine adäquate Migrationsstrategie entwickelt werden.
- *Evaluation:* Die Prüfung der entwickelten Lösungskonzepte hinsichtlich Ihrer Eignung zur Erfüllung der Anforderung vor der eigentlichen Umsetzung. Die Art der eingesetzten Aktivitäten ist nach der zu erreichenden Konfidenz zu wählen.
 - *Prototypen erstellen:* Die Erstellung von (Wegwerf-)Prototypen.
 - *Reviews durchführen:* Durchführung von Reviews der Lösungskonzepte durch eine dritte Person.
 - *Walkthroughs durchführen:* Die Durchführung von Walkthroughs, oft gemeinsam mit dem Kunden.
 - *Machbarkeitsanalysen durchführen:* Die Durchführung von Machbarkeitsanalysen verschiedener Art.
- *Realisierung:* Die Umsetzung der Lösungskonzepte und Durchführung verschiedener Implementierungstätigkeiten zur Entwicklung des Softwareprodukts.
 - *Softwareinkrement implementieren:* Realisierung eines Inkrements des Softwareproduktes durch Schreiben von Quellcode.
 - *Fehler beheben:* Die Suche und Behebung von Fehlern oder Qualitätsmängeln im Quellcode.
 - *Softwareinkrement ausliefern:* Bauen und Bündeln von Softwareinkrementen sowie deren Auslieferung, inklusive Installation beim Kunden.
 - *Dokumentation erstellen:* Zusammenstellung von Informationen und Verfassen von Nutzerdokumentation für die Arbeit mit einem Softwareinkrement.
 - *Daten migrieren:* Migration von Datenbeständen beispielsweise im Fall eines Versionsprunges oder der Ablösung eines Altsystems.
- *Kontrolle*

- *Interne Qualitätssicherung durchführen (Verifikation)*: Durchführung von internen Qualitätssicherungsaktivitäten wie die Erarbeitung von Testfällen sowie deren Ausführung.
- *Externe Qualitätssicherung durchführen (Validierung)*: Durchführung von externen Qualitätssicherungsaktivitäten wie Akzeptanztests oder Abnahmen mit dem Kunden.
- *Projektplanung und -Steuerung*: Querschnittliche Aktivitäten zur Projektplanung und -Steuerung, die somit nicht zu eigentlichen Entwicklungstätigkeiten zählen. Sie sind dennoch hier aufgeführt, da verschiedene Rollen während der Entwicklung häufig auch mit solchen Aktivitäten beschäftigt sind.
 - *Projekt planen*: Aktivitäten zur Planung von Projekten wie der Erarbeitung von Zeitplänen und Risikoabschätzungen.
 - *Projekt koordinieren*: Die Koordination von Projekten während der Durchführung. Darunter fallen auch Controllingaktivitäten.
 - *Aufwand schätzen*: Die Schätzung von benötigten Zeit- und Ressourceneinsätzen zur Erreichung eines Projekt[teil]ziels.
 - *Aufgaben zuweisen*: Die Zuweisung von Entwicklungsaufgaben zu konkreten Personen, die sie durchführen sollen.

Mit Ausnahme des Bereichs Projektplanung und -Steuerung werden die beschriebenen Aktivitäten während des Entwicklungsprozesses iterativ ausgeführt. Ein Neudurchlauf wird durch das Auftreten eines bestimmten Auslösers initiiert. Dies kann beispielsweise das Auftreten neuer oder geänderter Kundenanforderungen, Umpriorisierungen, Änderungen von Projektfaktoren oder auch einfach der Start eines neuen Sprints sein. Diese wiederholte Ausführung ist in Abbildung 2 mit den grauen Pfeilen illustriert.

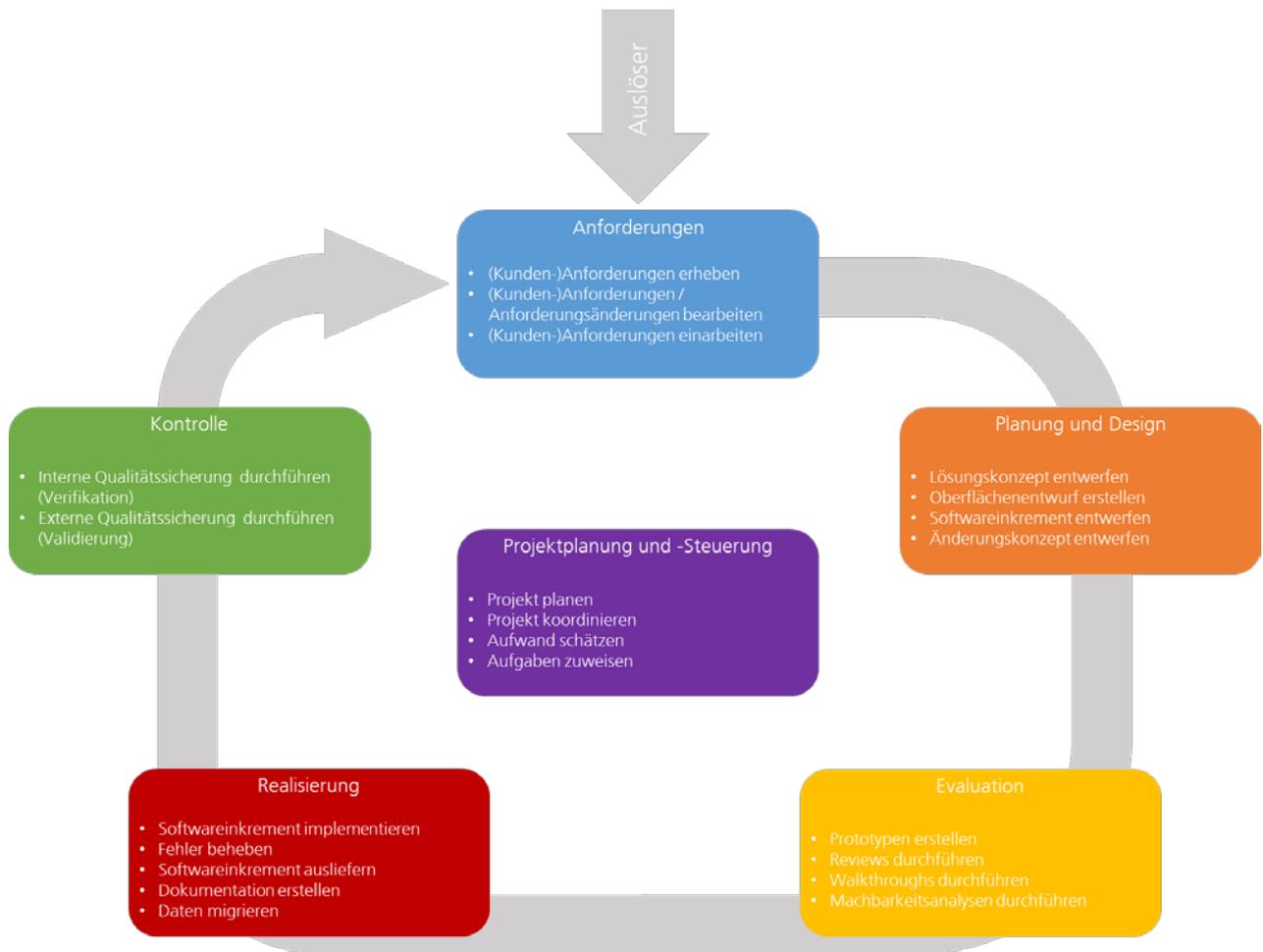


Abbildung 2: Agiler Referenzprozess

4. UUX-Best-Practices

Unternehmen weisen häufig in Ihren Entwicklungsprozessen spezifische Stärken auf, auch begünstigt durch den Einsatz agiler Entwicklungstechniken. Die schnelle Reaktion auf sich ändernde Anforderungen ist ein Beispiel dafür. Solche Stärken müssen bewahrt werden, wenn Änderungen am Entwicklungsprozess, wie bei der Etablierung von UUX-Best-Practices vorgenommen werden. Hierzu wurden in einer initialen Analysephase bei den im Projekt beteiligten KMU bereits folgende „Best-Practices“ identifiziert. Diese Darstellung gibt eine Übersicht über bereits eingesetzte pragmatische Lösungen, stellt keinesfalls eine Empfehlung dar. Ziel des Projektes ist es aber, eine oder mehrere Best Practices detailliert auszugestalten, so dass diese auch von anderen KMU verwendet werden können.

User Interface Circle

Für die Gestaltung des User Interface verantwortliche, querschnittlich agierende Gruppe. Der User Interface Circle dient hierbei den agilen Teams als Dienstleister. Es gibt so eine übergeordnete Instanz, die über mehrere agile Projektteams hinweg eine Konsistenz der User Experience sicherstellt. Die Verantwortlichkeiten dieser Gruppe sind Entwurf von Oberflächen, Erstellen und Pflege von Styleguides und User Experience im Allgemeinen.

Wireframes

Wireframes, Papierprototypen oder auch klickbare Prototypen, die mit digitalen Tools erstellt wurden sind weit verbreitet um Designkonzepte zu evaluieren und auch Anforderungen zu kommunizieren.

Nutzer- bzw. Kunden-Workshops

Je nach Projektsetting werden regelmäßig Nutzer- bzw. Kunden-Workshops durchgeführt. Diese werden entweder zur Anforderungsanalyse oder Konzeptevaluation durchgeführt. Ergebnisse werden möglichst direkt in Form von Tickets oder User Stories ins Backlog eingepflegt.

Usability Bug-Reports

Usability Bug Reports sind vom Kunden entdeckte Usability Schwachstellen und werden in Form von Tickets ins Backlog des jeweiligen Projektes eingepflegt.

Styleguides

Für die Konzeption des User Interfaces werden Interaction Styleguides und Visual Styleguides verwendet. Beim Interaction Styleguide werden die Rahmenbedingungen für die Interaktionsgestaltung vorgegeben (z.B. „Ein Benutzer muss mit max. 5 Interaktionsschritten eine Aufgabe im System abschließen können“). Der Visual Styleguide befasst sich wie gewöhnlich bekannt eher mit grafischen Rahmenbedingungen (z.B. Farben, Icons).

Persona

Personas werden eingesetzt, um im gesamten agilen Team eine Sensibilisierung für die Benutzergruppe zu erreichen und in Diskussionen konkrete Ansatzpunkte für eine Entscheidungshilfe zu haben. Sie eignen sich hervorragend als leichtgewichtiges Tool und sind durch explizites Anpinnen an Wänden oder Flipchart omnipräsent.

Usability und User Experience Coaching

Grundsätzlich wird versucht, das Bewusstsein für Usability und UX durch Coaching Aktivitäten zu stärken. Grundkenntnisse und ein gesteigertes Qualitätsbewusstsein in dieser Richtung werden durch ständige Kommunikation und Schulung bei allen Teammitgliedern gestärkt.

UX Product Philosophy

In der UX Product Philosophy werden gewünschte UX Anforderungen möglichst verständlich dokumentiert und in einer Matrix dargestellt. Diese ist individuell für das jeweilige Projekt und kann auch zur Evaluation innerhalb des Entwicklerteams genutzt werden.

6. Zusammenfassung und Ausblick

Dieser Artikel gibt einen ersten Überblick über die bereits erzielten Ergebnisse im Projekt PQ4Agile. Diese Ergebnisse sollen im Rahmen des während der UPA 2014 statt findenden Workshops unterstützt werden. So sollen vor allem weitere UUX Best Practices gesammelt werden und dem existierenden Qualitätsmodell zugeordnet werden.

7. Literatur

Balzert, H. (2011). Lehrbuch der Softwaretechnik: Entwurf, Implementierung, Installation und Betrieb. Spektrum Akademischer Verlag; Auflage: 3.

CMMI Product Team. (2010). CMMI for Development, Version 1.3. Software Engineering Institute. Software Engineering Institute.

<http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=9661>

[1] ISO – International Organization for Standardization / IEC – International Electrotechnical Commission (2011): Software-Engineering - Qualitätskriterien und Bewertung von Softwareprodukten (SQuaRE) - Qualitätsmodell und Leitlinien

[2] Ralf Kneuper (2011): Was ist eigentlich Prozessqualität? In: Hans-Ulrich Heiß, Peter Pepper, Holger Schlingloff, Jörg Schneider (Hrsg.): INFORMATIK 2011 – Informatik schafft Communities. Gesellschaft für Informatik, Bonn

[3] ISO – International Organization for Standardization (2008): Qualitätsmanagementsysteme - Anforderungen (ISO 9001:2008)

J. Ferreira, H. Sharp, und H. Robinson, "Agile Development and User Experience Design Integration as an Ongoing Achievement in Practice," in 2012 Agile Conference, 2012, pp. 11–20.

P. Hodgetts, "Experiences integrating sophisticated user experience design practices into agile processes," in Agile Development Conference (ADC'05), pp. 235–242.

M. Isomursu, A. Sirotkin, P. Voltti, und M. Halonen, "User Experience Design Goes Agile in Lean Transformation -- A Case Study," in 2012 Agile Conference, 2012, pp. 1–10.

[MAIER et al. 2014] Andreas Maier, Hartmut Schmitt, Dominik Rost (2014): PQ4Agile-Qualitätsmodell. Verfügbar unter: <http://www.pq4agile.de/qualitaetsmodell> [25.06.2014]

T. Memmel, H. Brau und D. Zimmermann, Agile nutzerzentrierte Softwareentwicklung mit leichtgewichtigen Usability Methoden – Mythos oder strategischer Erfolgsfaktor? In: Usability Professionals Tagungsband 2007, S. 223-227.

Schwaber, K., & Sutherland, J. (2009). Scrum Guide. Scrum Alliance, 19(6), 21. doi:10.1053/j.jrn.2009.08.012

M. Singh, "U-SCRUM: An Agile Methodology for Promoting Usability," in Agile 2008 Conference, 2008, pp. 555–560.

J. Verner und M. A. Babar, "Software quality and agile methods," in Proceedings of the 28th Annual International Computer Software and Applications Conference, 2004. COMPSAC 2004., pp. 520–525.

8. Autoren



Steffen Hess studierte Wirtschaftsingenieurwesen mit Fachrichtung Informatik an der TU Kaiserslautern. Bereits während seines Studiums arbeitete er als Usability Tester und User Researcher beim Fraunhofer IESE in Kaiserslautern. Seit 2009 ist er dort als wissenschaftlicher Mitarbeiter in den Bereichen Requirements Engineering und User Experience mit den Themenschwerpunkten Interaktionsdesign, UX Prototyping und Mobile Geschäftsanwendungen aktiv. Er verfügt über Erfahrung in der Konzeption, Entwicklung und Bewertung von zahlreichen Anwendungen und Apps in verschiedenen Branchen sowie aus Forschungsprojekten im Mobile Software Engineering. Heute ist Steffen Hess am Fraunhofer IESE als Teamleiter für User Experience und Mobile Software Engineering tätig.



Dominik Rost studierte Informatik und Software Engineering an der Hochschule Mannheim. Seit 2009 ist er wissenschaftlicher Mitarbeiter beim Fraunhofer Institut für Experimentelles Software Engineering (IESE) in Kaiserslautern und arbeitet im Bereich Softwarearchitekturen für Informationssysteme. Der Schwerpunkt seiner Tätigkeit ist die Entwicklung praktisch anwendbarer Methoden und Werkzeuge für die Verbesserung von Architekturarbeit in Entwicklungsprojekten. Auch wenn er in allen Bereichen der Softwarearchitektur tätig ist, spezialisiert er sich auf die Dokumentation und Modellierung von Softwarearchitekturen. Neben der angewandten Forschung führt er Beratungsprojekte mit Industriekunden aller Branchen durch, um deren Produktarchitekturen zu evaluieren, zu entwickeln, zu dokumentieren, sowie um die Fähigkeiten der Partner im Bereich Softwarearchitektur zu verbessern.



Hartmut Schmitt – Koordinator Forschungsprojekte bei der HK Business Solutions GmbH; seit 2006 Verbundvorhaben im Umfeld Mensch-Computer-Interaktion, Usability/User Experience und Software-Engineering, u.a. als Projektkoordinator in mehreren BMBF-geförderten Verbundvorhaben. Hartmut Schmitt ist Mitglied der Gesellschaft für Informatik und engagiert sich hier im Arbeitskreis „Use Cases in Forschung und industrieller Praxis“ der Fachgruppe Requirements Engineering.