

User Experience in Kanban

Die UX-Karte ausspielen



Dominique Winter
Buhl Data Service GmbH
Am Siebertsweiher 3/5
57290 Neunkirchen
dwinter@buhl-data.com

Eva-Maria Schön
7P Solutions & Consulting AG
Calor-Emag-Straße 1
40878 Ratingen
eva-maria.schoen@7p-group.com

Jan Uhlenbrok
basecom GmbH & Co. KG
Hannoversche Straße 6-8
49084 Osnabrück
jan@usability3000.de

Jörg Thomaschewski
Hochschule Emden/Leer
Constantiaplatz 4
26723 Emden
joerg.thomaschewski@hs-emden-leer.de

Abstract

Die agile Projektmanagementmethode „Kanban“ zielt auf geringe Durchlaufzeiten ab und richtet damit den Fokus der Entwicklung auf das Moment der aktuellen Aufgabe. Um Produkte mit einer positiven User Experience zu entwickeln, muss bereits der Gestaltungsprozess auf die Bedürfnisse des Menschen ausgelegt sein. Dadurch werden agile Projekte vor die Herausforderung gestellt, sowohl die einzelnen Aufgaben als auch das gesamte Produkt zielgerichtet und nutzerzentriert zu realisieren. Dieser Beitrag soll zeigen, wie trotz des eingeschränkten Gesamtüberblicks dennoch eine Nutzerzentrierung bei der Entwicklung realisiert werden kann. Hierzu wird vor allem die Frage geklärt, wie sich Human-Centered Design Aktivitäten in Kanban integrieren lassen und an welchen Stellen im Softwareentwicklungsprozess Evaluationen der User Experience durchgeführt werden können.

Keywords:

/// Kanban
/// User Experience
/// Human Centered Design
/// Agile Softwareentwicklung

1. Einleitung

Agile Projekte zeichnen sich dadurch aus, dass während des Entwicklungsprozesses schnell auf Änderungen der Anforderungen reagiert werden kann. Während die agilen Modelle die Softwareimplementierung optimieren, gibt es bei der Integration der nutzerzentrierten Gestaltung noch Handlungsbedarf. Agile Vorgehensmodelle wie beispielsweise Extreme Programming, Scrum und Kanban geben weder Vorgaben für die Integration von Human Centered Design-Methoden noch Hinweise darauf, wie Anforderungen für die Erstellung des Product Backlogs erhoben und priorisiert werden. Erste Ansätze existieren für Scrum, beispielsweise eine vorgelagerte Visioning Phase (Beyer 2010), (Winter, Holt & Thomaschewski 2012), (Holt, Winter & Thomaschewski 2012) oder eine parallel zum Entwicklungssprint verlaufende Konzeptionsphase in gleicher Organisation (Obendorf, Gibbert, Petersen & Memmel 2010), (Sy 2007).

Abgeschlossene Entwicklungszeiträume, wie sie beispielsweise Scrum in Form von

Sprints vorsieht, fehlen bei Kanban. Es findet eine kontinuierliche Entwicklung statt. Während dieser kontinuierlichen Entwicklung betrachtet das Teammitglied nur seine eigene Aufgaben-Einheit im jeweiligen Fachgebiet. Eine wiederkehrende Zieldefinition über Aufgabengrenzen hinweg findet nicht statt. Somit kann die User Experience (UX) als Ergebnis vieler einzelner Eigenschaften eines Produkts ohne die Konzeption des Gesamtprodukts schwer berücksichtigt werden.

Kanban ist eine Projektmanagementmethode, die in IT-Organisationen für die agile Softwareentwicklung eingesetzt wird. Im Vergleich zu Scrum existieren bei Kanban weniger Vorgaben. Während bei Scrum die Rollen Product Owner, Scrum Master und Team-Mitglieder beschrieben sind, sind bei Kanban keine Rollen definiert (Kniberg & Skarin 2010). Bei Scrum sind zudem multifunktionale Teams vorgeschrieben, denn ein Scrum-Team muss alle notwendigen Fertigkeiten besitzen, um ein Produkt zu entwickeln (Gloger 2011). Dazu sind neben Programmierern unter anderem auch Konzepter, Business-Analysten und Designer erforderlich. Im Vergleich hierzu sind bei

Kanban multifunktionale und funktionale Teams, bestehend aus Experten, erlaubt (Kniberg & Skarin 2010). Da in vielen Unternehmen bereits funktionale Teams vorhanden sind, ist eine Einführung von Scrum im Hinblick auf die Anpassung der Unternehmensorganisation mit mehr Aufwand verbunden als eine Einführung von Kanban.

2. Prozessablauf

Bei Kanban wird zur Visualisierung der Arbeitsaufgaben das sogenannte „Kanban-Board“ eingesetzt. Auf einer Art Tafel werden zunächst in Form von Spalten die einzelnen Aktivitäten (Implementierung, Testen, etc.) der Wertschöpfungskette benannt (Shalloway 2010), (Epping 2011). Die Anordnung der Spalten muss hierbei mit der Reihenfolge der Aktivitäten übereinstimmen, die zur korrekten Aufgabenerfüllung führen. Dies richtet sich nach den jeweiligen Prozessen des Unternehmens und kann sich daher von Fall zu Fall unterscheiden. Nicht jedes Unternehmen führt z.B. eine gesonderte, abschließende Qualitätssicherung durch, andere hingegen haben zwei Aktivitäten wie Oberflächen- und Performancetests.

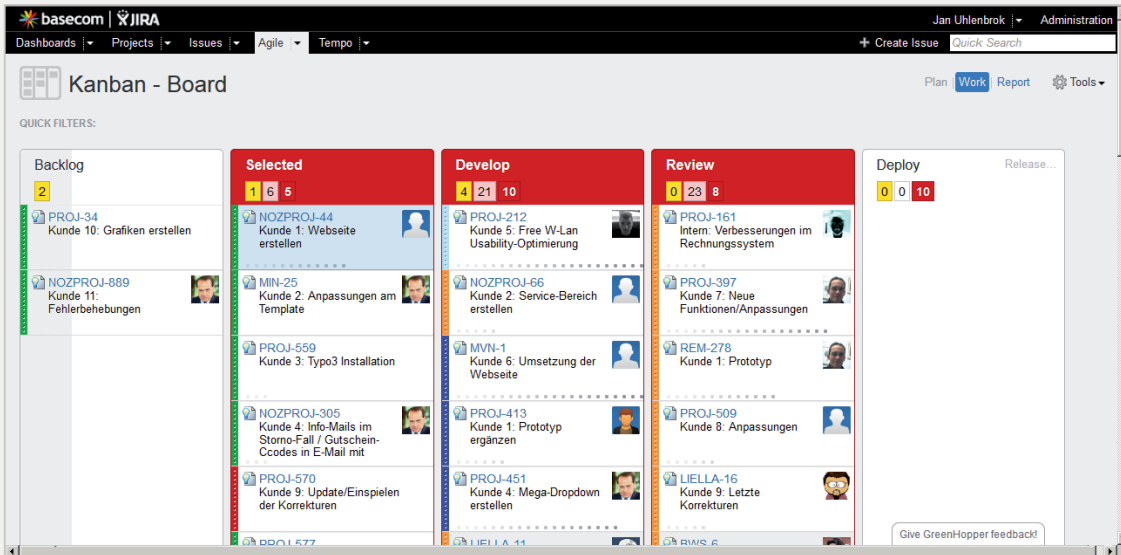


Abb. 1.
Beispielhaftes elektronisches Kanban-Board

Die Aufgaben selbst werden in Kanban auf gedruckten Karten (oder elektronisch als Tickets in einem Ticketsystem, [Abb. 1]) dargestellt. Jede Karte beinhaltet eine Beschreibung der zu erledigenden Aufgabe. Wurde in einer Spalte die Bearbeitung der Aufgabe abgeschlossen, so wandert die Karte weiter zur nächsten, bis schließlich die letzte Spalte erreicht wird.

Die Teammitglieder bearbeiten nur Aufgaben, die sich auf dem Board befinden. Das Ziel der Arbeit mit dem Kanban-Board ist, dass alle Aufgaben erfasst sind und sich im stetigen Fluss befinden. Hierzu dient die Limitierung der gleichzeitig platzierten Aufgaben pro Spalte, wodurch die laufende Arbeit einer Spalte beschränkt wird. Diese Limitierung wird als „Work in Progress“ bezeichnet und beschreibt die maximale Anzahl an gleichzeitig in Arbeit befindlichen Aufgaben (Epping 2011).

Durch die Visualisierung am Kanban-Board werden Engpässe schnell sichtbar. Sobald eine Aufgabe den Fluss der Aufgaben im Prozess blockiert, werden alle verfügbaren Ressourcen eingesetzt, um diese Blockade zu lösen (Anderson 2011). Die weiteren Regeln zum erfolgreichen Einsatz von Kanban finden sich bei Anderson (Anderson 2011), Leopold & Kaltenecker (Leopold & Kaltenecker 2012) und Patton (Patton 2009).

Die Begrenzung der gleichzeitig in einer Spalte befindlichen Arbeitsaufgaben führt zu einem weiteren Effekt: In jeder Prozesskette stellt eine der Spalten zwangsweise einen Engpass dar. Auch in einem Kanban-Board sorgt dieser „Flaschenhals“ dafür, dass nicht mehr Aufgaben weitergereicht werden können, als es diese eine limitierende Spalte zulässt (Leopold & Kaltenecker 2012). Wenn sich daher die Anzahl der eingehenden Aufgaben nach der limitierenden Spalte richtet, um damit den Aufgabenfluss konstant zu halten, so entstehen bei allen anderen Aktivitäten zeitliche Freiräume, die für aufgabenfremde Zwecke genutzt werden können (Anderson 2011). Beispielsweise muss die Entwicklung nicht mehr Ergebnisse liefern, als mit den geplanten Testressourcen anschließend getestet werden können. Die gewonnene Zeit kann in die Erhöhung von Qualität an Produkten, Projekten oder Abläufen investiert werden.

Im Gegensatz zu der in IT-Organisationen oftmals verwendeten Managementmethode Scrum werden weder gemeinsame Planungen noch größere Besprechungen zu inhaltlichen Themen eingeplant. Die Kanban-relevanten Besprechungen (z.B. Teamretrospektive oder Queue Replenishment Meeting) fokussieren die Optimierung des Prozesses (Leopold & Kaltenecker

2012). Inhaltliche Aspekte des Projektes bleiben dabei außen vor. Auch das tägliche Standup-Meeting, bei dem grob Aufgaben umrissen werden, wird auf aktuelle Aufgaben beschränkt. Dennoch ist Kanban ein flexibles Rahmenwerk und es bietet sich an, die aus Scrum bekannten Rituale auch in Kanban zu übernehmen. Somit kann der Kanban-Entwicklungsprozess auf die Bedürfnisse des Teams und des Produkts hin optimiert werden (Kniberg & Skarin 2010).

Durch die Fokussierung auf die aktuelle Aufgabe fehlt die Möglichkeit für den einzelnen, sich den für die User Experience wichtigen Überblick über das den Aufgaben zugehörige Projekt zu verschaffen. Denn selbst wenn einzelne Teil-Aufgaben auf User Experience getestet werden, so bedeutet dies nicht automatisch eine gute User Experience für die zu lösende Gesamt-Aufgabe. Hierbei ist vielmehr das Zusammenspiel der einzelnen Bestandteile zu betrachten, die miteinander interagieren (Poppendieck & Poppendieck 2003).

3. Integration der nutzerzentrierten Gestaltung in Kanban

Zur Integration nutzerzentrierter Gestaltungsansätze in den Kanban-Prozess



bieten sich neben einer Anpassung der in der Wertschöpfungskette befindlichen Aktivitäten (Release Evaluation, Erweiterung um weitere Kanban-Boards) vor allem die Einführung von UX-relevanten Artefakten (Persona-driven User Storys, Verwendung von Prototypen) und Ritualen (Überwinden von Teamgrenzen) an, damit die Anforderungen des Human Centered Design (HCD) (Deutsches Institut für Normung 2011) abgebildet werden können.

3.1. Release Evaluation

Da im Laufe der Prozessbewältigung immer mehr Aufgaben abgearbeitet werden, füllt sich nach und nach die letzte Spalte des Boards. Diese hat keine Begrenzung an Aufgabenzetteln. Um eine regelmäßige Evaluation bezüglich der nutzerbezogenen Eigenschaften eines Produkts durchzuführen, bietet es sich an, auch für die letzte Spalte eine Beschränkung einzuführen und beim Erreichen entsprechende UX-Evaluationen vorzunehmen. Die Anzahl der zu sammelnden Aufgaben hängt von der jeweiligen Teamgröße und den Releasezyklen ab. Beim Sammeln der Aufgaben gilt zu beachten, dass besonders neue oder veränderte Funktionen getestet werden, obwohl Bugs mit zu den zu sammelnden Aufgaben gehören. Aus diesen Evaluationen resultieren Verbesserungen, welche als neue Aufgaben im Kanban-Prozess einzuplanen sind. So entsteht ein iterativer Kontrollmechanismus, der langfristig die Einhaltung einer User Experience – Strategie ermöglicht und gesetzte Ziele verfolgbar macht.

3.2. Überwinden von Teamgrenzen

Aufgrund bestehender Unternehmensstrukturen bilden sich in vielen Organisationen funktionale „Silos“, die in sich geschlossene Gruppen von Experten beinhalten. Der Nachteil dieser funktionalen Silos besteht darin, dass sich Wissen an einer Stelle des Entwicklungsprozesses befindet und nicht ohne weiteres über die Teamgrenzen hinweg genutzt werden kann. Agile Entwicklungsprozesse leben

von der gemeinschaftlichen Zusammenarbeit verschiedener Disziplinen (Beck et al. 2001). Diese interdisziplinäre Zusammenarbeit innerhalb eines Projektes bietet den Vorteil, dass Ideen bereits in frühen Phasen des Softwareentwicklungsprozesses von unterschiedlichen Seiten betrachtet werden können (Gothelf & Seiden 2012). Beispielsweise kann ein UX-Experte bei der Konzeption eines neuen Features einen Frontend-Entwickler zur Beratung hinzuziehen. Der Frontend-Entwickler kann mit seinem Wissen die Ideen aus technologischer Sicht hinsichtlich Umsetzbarkeit und Aufwand bewerten. Darüber hinaus kann er Vorschläge zur Optimierung der Idee beisteuern.

Da sich die Summe der Aufgaben in einem Kanban-Board nach dem Engpass richten sollte, können gewonnene Freiräume eingesetzt werden, um die für eine gute User Experience erforderliche Kommunikation und Zusammenarbeit zu ermöglichen. Die Freiräume bieten Gelegenheit, Silos aufzubrechen und gemeinschaftlich zu arbeiten. Schließlich führt die gemeinschaftliche Weiterentwicklung der Idee zu einer besseren Lösung, die sich positiv auf die User Experience des zu entwickelnden Produktes auswirkt. Dies resultiert vor allem aus einem flexibleren Team, das schneller auf Nutzerfeedback reagieren (Klein 2013) und Kommunikationsschwierigkeiten durch das „Stille Post“-Prinzip verhindern kann (Brown 2012).

3.3. Verwendung von Persona-driven Use Storys

Zur Abgrenzung des Produktumfangs und zur Darstellung von Anforderungen werden in agilen Entwicklungsprozessen oftmals User Storys eingesetzt. Besonders in Verbindung mit Scrum (Wirdemann 2011) und Extreme Programming (Wells 1999) sind diese bekannt, können aber auch in Kanban eingesetzt werden. Eine User Story besteht aus drei Teilen (Cohn 2004). Zum einen handelt es sich hierbei um eine textuelle Beschreibung, die eine Definition der Anforderung darstellt und für die Projektplanung genutzt werden kann. Ein

weiterer Teil stellt die Diskussion um die User Story dar, die zu einem besseren Verständnis der eigentlichen Anforderung im Team beiträgt. Der dritte Teil besteht aus den Akzeptanzkriterien. Aus den Akzeptanzkriterien geht hervor, wann eine User Story abgenommen werden kann. Sogenannte Persona-driven User Storys (Holt, Winter & Thomaschewski 2012) können verwendet werden, wenn Personas entwickelt wurden. Diese speziellen User Storys stehen in direkter Verbindung mit einer speziellen Persona (Cooper 1999), (Pruitt & Adlin 2006), (Holt, Winter & Thomaschewski 2011). Sie bieten durch den Bezug auf eine konkrete Persona den Vorteil, dass der Nutzer nicht als anonymes Wesen angesehen wird. Werden Persona-driven User Storys in Kanban verwendet, werden die Personas nicht nur in der Konzeption genutzt, sondern explizit bis zu den Entwicklern transportiert. Sie können dadurch projektfassend verwendet werden.

3.4. Verwendung von Prototypen

Die Anzahl, Reihenfolge und Benennung der Spalten im Kanban-Prozess flexibel sind, ist es möglich, den Prozess durch neue oder andere Aktivitäten in Richtung Human Centered Design zu erweitern. So kann vor der Bearbeitung durch Softwareentwickler eine Umsetzungsspalte und eine Evaluationsspalte für Prototypen eingefügt werden. Dadurch werden diese Maßnahmen fest innerhalb des Prozesses vorgeschrieben und auf deren Einhaltung geachtet.

Der Vorteil beim Einsatz von Prototypen ist die Möglichkeit, bei den Projektbeteiligten ein umfassendes Bild vom zu entwickelnden Produkt zu erzeugen (Beyer 2010). Prototypen zeigen komplexe Zusammenhänge einzelner Anforderungen und verdeutlichen, wie diese im Kontext des Produkts zu verstehen sind. Des Weiteren können komplexe Arbeitsabläufe der Nutzer visualisiert werden. Ein weiterer Vorteil von Prototypen ist, dass bereits vor der Entwicklung Nutzerfeedback eingeholt werden kann. Es besteht die Möglichkeit, erste Usability-Tests an

Prototypen durchzuführen und somit wichtige Erkenntnisse für die Optimierung des zu entwickelnden Produktes zu sammeln (Hamborg, Klassen & Volger 2009). Des Weiteren kann bereits in frühen Phasen des Entwicklungsprozesses überprüft werden, ob das konzeptuelle Modell des Produktes mit den Annahmen über das mentale Modell der Nutzer übereinstimmt. Diese Übereinstimmung ist grundlegend für eine intuitive User Experience (Weinschenk 2011).

3.5. Erweiterung um weitere Kanban-Boards

Eine Integration eines HCD-Prozesses ist innerhalb eines einzelnen Kanban-Boards schwerlich umzusetzen, da dieses häufig auf die reine Umsetzung der Programmierfähigkeit und der Abnahme- und Auslieferungstätigkeiten beschränkt ist. Daher ist eine Erweiterung des Kanban-Prozesses nach „vorne“ am sinnvollsten, denn dadurch werden Konzeptionsarbeiten in gleicher Art und Weise organisiert wie die Realisierung durch die Programmierung und der Prozess der Wertschöpfungskette wird vereinheitlicht. Um unterschiedliche Teams abzubilden (z.B. Konzeptionsteam und Programmiererteam) können auch zwei oder mehr Boards nebeneinander genutzt werden. Ergebnisse der Konzeption wandern dann vom Konzeptionsboard zum Entwicklungsboard und können anschließend durch ein Administrations- oder Technikteam ausgeliefert werden. [Abb. 2]

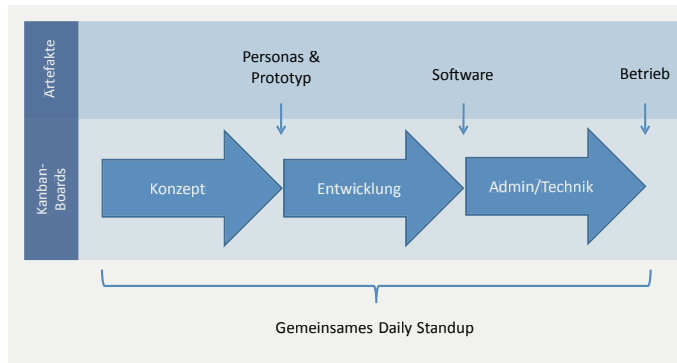


Abb. 2. Teamübergreifender Kanban-Prozess

Dies ermöglicht auch Engstellen zwischen den Teams aufzudecken, wenn die Realisierung nicht schnell genug vorankommt, um die fertigen Konzepte zu entwickeln oder umgekehrt.

Obwohl beim Einsatz von mehreren Kanban-Boards die Teams getrennt werden können, müssen Aufgaben gemeinsam geschätzt und auf technische Parameter wie Realisierbarkeit, Aufwand, etc. abgestimmt werden. Dies sollte durch Meetings einer teamübergreifenden Gruppe aus allen Kanban-Teams erfolgen. Die Ergebnisse der Beurteilung sind dann die Grundlage für die Einplanung auf dem Entwicklungsboard.

Es ist sinnvoll, dass alle Teams gemeinsam an den Daily Standup-Meetings teilnehmen, so dass das Entwicklungsteam über zukünftige Aufgaben informiert ist und das Konzeptionsteam aktuelle Probleme der Realisierung kennt. Dies fördert den Austausch zwischen den beteiligten Teams. Des

Weiteren muss die langfristige Entwicklung für alle Teams zugänglich gemacht werden (Epping 2011). Ziel ist es, teamübergreifend eine langfristige Planung zu kommunizieren und somit eine Gesamtübersicht für alle Projektbeteiligten zu schaffen.

In Abbildung 3 ist die Erweiterung der Wertschöpfungskette um Konzeptionsboard und Technikboard dargestellt. Nachdem eine Anforderung das Konzeptionsboard durchlaufen und die Spalte „Done“ erreicht hat, wird diese zu einer oder mehreren Aufgaben im Entwicklungsboard. Sobald die Aufgaben die Spalte „Done“ des Entwicklungsboards erreicht haben, können diese zu einem weiterführenden Technikboard verschoben werden, damit beispielsweise die umgesetzten Produktbestandteile auf verschiedenen Systemen installiert werden. [Abb. 3]

Abschließend gilt zu beachten, dass nach dem gesamten Prozess ein Abgleich

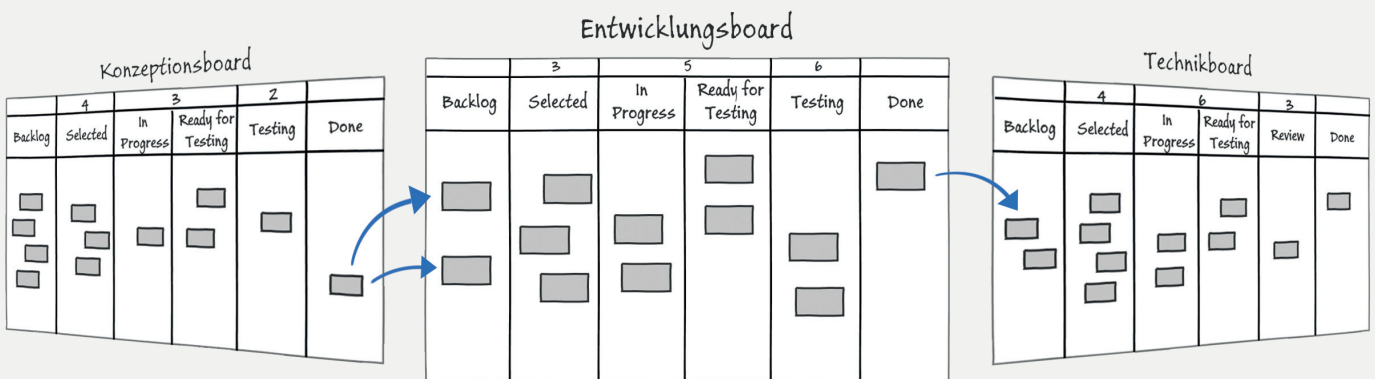


Abb. 3. Beispielhafte Kombination mehrerer Kanban-Boards



zwischen den Planungen des Konzeptionsboards und der abgeschlossenen Entwicklung vorgenommen werden muss. Dies dient der Zielkontrolle und sichert eine gerichtete Produktentwicklung.

4. Zusammenfassung

Agile Prozesse können und müssen mit einigen Erweiterungen im Hinblick auf die nutzerzentrierte Gestaltung optimiert werden. Bei Scrum bietet sich hierfür eine vorgelagerte Visioning Phase an, in welcher die Vision des Produktes konkretisiert wird. Ebenso kann eine parallel zum Entwicklungssprint verlaufende Konzeptionsphase genutzt werden. Parallel kann bei der Projektmanagementmethode Kanban ein weiteres Kanban-Board für die Konzeption eingeführt werden. Dieses Board wird gleichzeitig mit dem in der Entwicklung eingesetzten Entwicklungsboard verwendet. Der Einsatz eines Konzeptionsboards bietet den Vorteil, dass konzeptionelle Aufgaben, die für das Design einer positiven User Experience wichtig sind, auf gleiche Art und Weise organisiert und im Fortschritt kommuniziert werden können wie die Entwicklungsaufgaben. Somit werden konzeptionelle Aufgaben im Entwicklungsprozess etabliert und erhalten den gleichen Stellenwert. Des Weiteren werden die Grenzen zwischen verschiedenen Teams wie beispielsweise Konzeption und Entwicklung aufgebrochen, was zu positiven Effekten analog denen eines multifunktionalen Teams führt. Dies wird nicht nur durch ein gemeinsames Daily Standup-Meeting erreicht, sondern insbesondere durch ein gleichgewichtiges Einbeziehen aller am Prozess beteiligten Teammitglieder.

Literatur

1. Anderson, D. (2011). Kanban: Evolutionäres Change Management für IT-Organisationen, 1. Auflage, Heidelberg, Neckar: dpunkt.
2. Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R., Mellor, S., Schwaber, K., Sutherland, J. & Thomas, D. (2001). Manifesto for Agile Software Development.
3. Beyer, H. (2010). User-centered agile methods, San Francisco, Calif: Morgan & Claypool.
4. Brown, D. (2012). Agile user experience design: A practitioner's guide to making it work, San Francisco, Calif: Morgan Kaufmann.
5. Cohn, M. (2004). User stories applied: For agile software development, Boston, Mass: Addison-Wesley.
6. Cooper, A. (1999). The inmates are running the asylum, Indianapolis, Ind: Sams.
7. Deutsches Institut für Normung (2011). Ergonomie der Mensch-System-Interaktion – Teil 210: Prozess zur Gestaltung gebrauchstauglicher interaktiver Systeme.
8. Epping, T. (2011). Kanban für die Softwareentwicklung, Berlin, Heidelberg: Springer Berlin Heidelberg.
9. Gloger, B. (2011). Scrum: Produkte zuverlässig und schnell entwickeln, 3. Auflage, München: Hanser, Carl.
10. Gothelf, J. & Seiden, J. (2012). Lean UX: Applying lean principles to improve user experience, 1. Auflage, Sebastopol, CA: O'Reilly Media, Inc.
11. Hamborg, K.-C., Klassen, A. & Volger, M. (2009). Zur Gestaltung und Effektivität von Prototypen im Usability-Engineering, In Mensch & Computer.
12. Holt, E.-M., Winter, D. & Thomaschewski, J. (2011). Personas als Werkzeug in modernen Softwareprojekten: Die Humanisierung des Anwenders, In Usability Professionals 2011, H. Brau, A. Lehmann, K. Petrovic und M.C. Schroeder (Hrsg.), Stuttgart.
13. Holt, E.-M., Winter, D. & Thomaschewski, J. (2012). Von der Idee zum Prototypen: Werkzeuge für die agile Welt, In Usability Professionals 2012, H. Brau, A. Lehmann, K. Petrovic und M.C. Schroeder (Hrsg.), Stuttgart.
14. Klein, L. (2013). UX for lean startups: Faster, smarter user experience research and design, Sebastopol, CA: O'Reilly Media, Inc.
15. Kniberg, H. & Skarin, M. (2010). Kanban and Scrum: Making the most of both, [S.l.]: C4Media, Inc.
16. Leopold, K. & Kaltenecker, S. (2012). Kanban in der IT: Eine Kultur der kontinuierlichen Verbesserung schaffen, München: Hanser.
17. Obendorf, H., Gibbert, R., Petersen, I. & Memmel, T. (2010). Agile UX – Wege zur agilen nutzerzentrierten Entwicklung, In Brau H., Diefenbach S., Göring K., Peissner M., Petrovic K. (Hrsg.): Usability Professionals 2010.
18. Patton, J. (2009). Kanban Development Oversimplified, http://www.agileproductdesign.com/blog/2009/kanban_over_simplified.html, abgerufen am 16.03.2013.
19. Poppendieck, M. & Poppendieck, T. (2003). Lean software development: An agile toolkit, Boston, Mass: Addison-Wesley.
20. Pruitt, J. & Adlin, T. (2006). The persona lifecycle: Keeping people in mind throughout product design, Amsterdam ;, Boston: Elsevier; Morgan Kaufmann Publishers, an imprint of Elsevier.
21. Shalloway, A. (2010). Common Myths of Kanban, <http://www.netobjectives.com/blogs/common-myths-kanban>, abgerufen am 27.03.2013.
22. Sy, D. (2007). Adapting usability investigations for agile user-centered design. In: Journal of usability studies, Vol. 2, Nr. 3, S. 112–132.
23. Weinschenk, S. (2011). 100 things every designer needs to know about people, Berkeley, CA: New Riders.
24. Wells, D. (1999), User Stories, <http://www.extremeprogramming.org/rules/userstories.html>, abgerufen am 27.07.2013
25. Winter, D., Holt, E.-M. & Thomaschewski, J. (2012). Persona driven agile development: Build up a vision with personas, sketches and persona driven user stories. In: Proceedings of the 7th Conference on Information Systems and Technologies (CISTI).
26. Wirdemann, R. (2011). Scrum mit User Stories, 2. Auflage, München: Hanser, Carl.