



Mittelstand 4.0
Kompetenzzentrum
Usability

Mittelstand-
Digital



Gefördert durch:
aufgrund eines Beschlusses
des Deutschen Bundestages

UUX und Agilität

Schwerpunktthema der Region Ost

Prof. Dr. Manfred Thüring

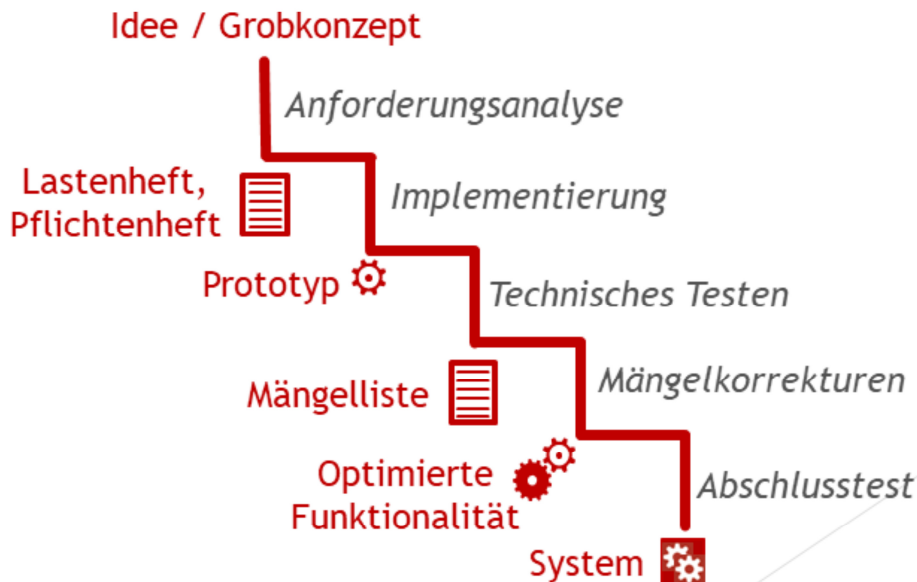
Eröffnungsveranstaltung des Kompetenzzentrums

Stuttgart

01.03.2018

Usability (Gebrauchstauglichkeit) und User Experience (Nutzererleben) werden in der Regel für ein System mit bestimmten Methoden untersucht. Wie gut dessen Usability ist und wie positiv sich das Erleben der Interaktion mit dem System letztlich gestaltet, wird davon bestimmt, inwieweit das System die Bedürfnisse seiner Nutzer befriedigt und die Anforderungen erfüllt, die diese an das System stellen. Damit dies gelingt, müssen nicht nur passende Methoden ausgewählt, sondern auch im Entwicklungsprozess eingesetzt werden. In welchem Maße dies geschieht bzw. möglich ist, hängt maßgeblich von der Methodologie ab, die der Systementwicklung zugrunde liegt.

Klassisches Wasserfallmodell



Häufig wird Software nach einem sequentiellen Modell entwickelt, dem sog. Wasserfallmodell, das hier vereinfacht dargestellt wird. Dabei wird meist eine initiale Produktidee zunächst in ein Grobkonzept überführt. Davon ausgehend werden Anforderungen an das Produkt, die ein Auftraggeber stellt, analysiert und in einem Lastenheft dokumentiert. Der Auftragnehmer beschreibt auf dieser Basis seine Leistungen und dokumentiert im Pflichtenheft, wie diese umgesetzt werden sollen. Anschließend erfolgt die Implementierung, wobei Prototypen entwickelt und technisch getestet werden, indem sie an entsprechenden Benchmarks gemessen werden. Mängel, die man dabei entdeckt, werden dokumentiert und idealerweise anschließend beseitigt. An dieser Stelle weicht das Modell strenggenommen von der Wasserfallstruktur ab, da es bei der Mangelbeseitigung zu mehreren Iterationen kommen kann. Zum Schluss wird das System einem finalen Test unterzogen, der gewährleisten soll, dass es keine größeren Mängel hat und in den Markt eingeführt werden kann.

Vor- und Nachteile

ABER:

Logisches Vorgehen
mit hoher Planungs-
sicherheit

Unflexibles Abarbeiten von Anforderungen

Zu spätes Feedback durch den Kunden

Aufwändige und teure Korrekturen

Lange Entwicklungszeiten

Später Markteintritt
und Qualitätsmängel

Hohes Misserfolgsrisiko

Das Wasserfallmodell hat den Vorteil, dass es ein systematisches Abarbeiten von Entwicklungsaufgaben in einer logischen Sequenz ermöglicht. Außerdem bietet es für den Auftragnehmer eine hohe Planungssicherheit, da die im Pflichtenheft dokumentierten Anforderungen meist Teil des Vertrages zwischen Auftragnehmer und Auftraggeber sind. Auf der anderen Seite ist es aber auch mit einer Reihe von Nachteilen verbunden, die das Risiko des Scheiterns eines IT-Projekts erhöhen.

(Miss-)Erfolg von IT-Projekten

Standish Group

- Seit 1994 jährliche Studie zum Erfolg von IT-Projekten
- Bis dato mehr als 40.000 Projekte analysiert.

Ergebnis 2015

- 29% erfolgreich
- 52% mit einem Teilerfolg
- 19% abgebrochen

Wichtigster Erfolgsfaktor

⇒ Einbeziehung von Kunden und Nutzern

Die Standish Group veröffentlicht seit 1994 die sog. Chaos Studie, in der mittlerweile der Erfolg bzw. Misserfolg von über 40.000 IT-Projekten analysiert wurde. Die Studie unterscheidet drei Typen von Projektergebnissen:

Typ 1 - Projekt erfolgreich abgeschlossen: Das Projekt wurde rechtzeitig, ohne Kostenüberschreitung und mit dem ursprünglich geforderten Funktionsumfang abgeschlossen.

Typ 2 - Projekt teilweise erfolgreich: Das Projekt wurde abgeschlossen, es kam jedoch zu Kosten- und/oder Zeitüberschreitungen oder es wurde nicht der vollständige geplante Funktionsumfang erreicht.

Typ 3 - Projekt nicht erfolgreich: Das Projekt wurde abgebrochen.

Im Jahr 2015 wurden laut Studie lediglich 29% von Softwareprojekten erfolgreich abgeschlossen (Typ 1), 52% erreichten lediglich einen Teilerfolg (Typ 2) und 19% wurden abgebrochen, ohne das gewünschte Ergebnis zu erzielen (Typ 3).

An erster Stelle für die Faktoren, die den Erfolg bzw. Misserfolg von IT-Projekten bestimmen, gaben 15,9 Prozent der verantwortlichen IT-Manager, die in der Chaos Studie befragt wurden, das „User Involvement“ an, also das Einbeziehen von Systemnutzern in den Entwicklungsprozess.

Gibt es Alternativen?

Ja, in Form agiler Entwicklungsmethoden

Iteration statt Wasserfall



- Kurze Zyklen aus Implementieren und Testen
- Frühes und begleitendes Kunden-Feedback
- Hohe Kundenorientierung durch dynamische Anpassung der Anforderungen

Kombinationsmöglichkeit mit Usability- und UX-Methoden!

Eine Alternative, die sich zunehmend in der Softwareentwicklung durchsetzt, besteht in der Anwendung agiler Entwicklungsmethoden. Sie zeichnen sich durch intensiveres Testen aus, bei dem Entwicklung und Überprüfung iterativ in kurzen Zyklen miteinander verschränkt werden. Dies geht einher mit einer stärkeren Einbindung des Kunden während der Entwicklung, verbunden mit der Möglichkeit, anfangs spezifizierte Anforderungen bei Bedarf zu verändern oder zu erweitern, um flexibel auf die Wünsche des Kunden eingehen zu können.

Agiles Manifest

Individuen und Interaktionen eher als Prozesse und Werkzeuge
Funktionierende Software eher als umfassende Dokumentation
Zusammenarbeit mit dem Kunden eher als Vertragsverhandlung
Reagieren auf Veränderung eher als das Befolgen eines Plans



Das heißt,

obwohl die Werte auf der rechten Seite wichtig sind,
werden die Werte auf der linken Seite stärker betont.

Die Grundprinzipien agiler Arbeit wurden bereits 2001 im sog. Agilen Manifest zusammengefasst. Sie besagen, dass bei der Systementwicklung eine neue Art der Priorisierung verfolgt werden sollte.

Durch diese Prinzipien ergeben sich flexiblere Gestaltungsspielräume, so dass dynamische Veränderungen zeitnah berücksichtigt werden können – ein wesentlicher Vorteil gegenüber Unternehmen bzw. Abteilungen, die nach klassischen Ansätzen vorgehen. Außerdem wird deutlich, dass sich Arbeitsstrukturen und Teamzusammensetzungen diesem Prozess anpassen müssen, Individuen und Interaktionen bekommen hier einen besonders hohen Stellenwert. Teams und Teammitglieder erhalten dadurch deutlich mehr Verantwortung und Autonomie als bisher üblich. Inzwischen gibt es eine ganze Reihe verschiedener, agiler Ansätze. Einer der bekanntesten ist Scrum.

Agilität mit Scrum

Produktplanung

Product Backlog

- User Stories verantwortet durch Product Owner

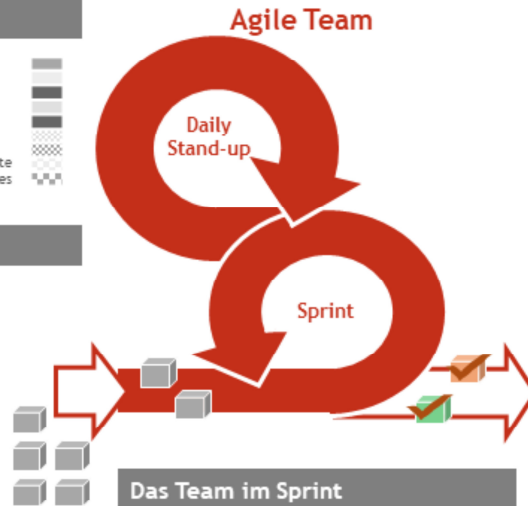
Priorisierte
User Stories

Meeting: Sprintplanung



Sprint Backlog

- verantwortet durch Scrum Team



Meeting: Daily Stand-up

15 Minuten, täglich

- Was habe ich gestern gemacht?
- Plan für heute?
- Welche Probleme/ Hürden?

Meetings: Review, Retrospective

Sprint Demo

Feedback

Sprint Review

Inhaltliche Prüfung

Sprint Retrospective

Introspektion mit Scrum Master

Scrum unterscheidet zunächst einmal verschiedene Rollen für die Personen, die an der Entwicklung teilnehmen. Der **Product Owner** ist für die Produktplanung und den reibungslosen Ablauf des Projekts verantwortlich. Der sog. **Scrum Master** achtet darauf, dass die Prinzipien der agilen Entwicklung eingehalten werden und fungiert als Moderator im Entwicklungsprozess. Das Scrum Team organisiert seine Arbeit autonom und ist für die eigentliche Implementierung zuständig.

Auch Scrum kommt nicht ohne **Planungen** und **Meetings** aus – beides ist deshalb zentraler Bestandteil des agilen Prozesses. In der **Produktplanung** werden Anforderungen in Form sog. **User Stories** aus der Perspektive des Nutzer formuliert und im sog. **Product Backlog** – meist auf einem einfachen, analogen White Board - gesammelt. Der Product Owner ist für die Pflege des Backlogs verantwortlich und **priorisiert** die Aufgaben, die für Abarbeitung der User Stories zu erledigen sind. Das Backlog ist **nicht statisch**, sondern kann durch den Product Owner dynamisch an die Kundenwünsche angepasst werden.

Aufgaben werden in sog. **Sprints** bearbeitet, aus denen sich der Projektverlauf zusammensetzt. Ein Sprint ist ein kurzes Arbeitsintervall von meist 2 Wochen Dauer, an dessen Ende eine **Erweiterung der Software** steht. Dieses „Software Increment“ muss voll funktionsfähig sein. Was dafür im einzelnen von wem zu tun ist, legt das Team in Eigenverantwortung bei der Planung des Sprints fest in einem Meeting fest, an dem alle Teammitglieder teilnehmen. Dabei werden unter Beachtung der Prioritäten User Stories aus dem Product Backlog ausgewählt und Aufgaben formuliert, die zur Umsetzung der Anforderungen bearbeitet werden müssen. Die Planungsergebnisse werden im sog. **Sprint Backlog** festgehalten.

Bei der Durchführung eines Sprints arbeiten die Teammitglieder die Aufgaben selbstständig ab. Der Fortschritt wird täglich in einem Meeting von maximal 15 Minuten (z.B. **Daily Stand-ups**) untereinander kommuniziert. Hier geht es nicht um Leistungskontrolle, sondern darum, das Team zu synchronisieren, also alle auf den gleichen Kenntnisstand zu bringen. Anhand der Anzahl der erledigten Tasks mit ihrem jeweiligen Aufwand, kann die Entwicklungsgeschwindigkeit (die sog. **Velocity**) des Teams bestimmt werden. Sie ermöglicht eine unmittelbare Rückmeldung über den aktuellen Stand und hilft, eventuelle Verzögerungen oder Probleme früh zu erkennen, um geeignete Maßnahmen ergreifen zu können.

Am Ende eines Sprints werden im **Review-Meeting** die implementierten Funktionen (das Inkrement) inhaltlich geprüft und nach erfolgreichem Test wird die Erfüllung der bearbeiteten Aufgaben im Product Backlog dokumentiert. Da auch der Prozess und die Interaktionen untereinander im Fokus stehen, wird die Zusammenarbeit in der sogenannten **Retrospective** ausgewertet. Hier geht es primär darum, Bedingungen für eine gute und effiziente Zusammenarbeit mittelfristig zu verbessern und dadurch Mitarbeiterzufriedenheit und Produktivität zu steigern.

Vorteile von Agilität



Frühes Feedback



Reduziertes Risiko



Faster Time to Market



Flexibilität



Qualität



Transparenz



Skalierbarkeit

... aber wie können Nutzer in die Entwicklung einbezogen werden?

Agilität bietet zahlreiche Vorteile:

Frühes Feedback: Dadurch dass nach jedem Sprint eine lauffähige Erweiterung der Software vorliegt, kann diese dem Kunden gezeigt werden, um schon früh – und idealerweise regelmäßig - Feedback einzuholen. Notwendige Änderungen oder Erweiterungen von Anforderungen werden dadurch schnell erkannt, so dass sich Verbesserungen unkompliziert und mit vergleichbar geringen Folgekosten vornehmen lassen.

Reduziertes Risiko: Es werden sowohl technische Risiken, funktionale und inhaltliche Risiken (durch frühes Kunden-Feedback) als auch organisationale Risiken (durch einen für alle transparenten Prozess) reduziert.

Schnellere Markteinführung bzw. Auslieferung: Durch die iterative Entwicklung kann sehr viel schneller auf sich verändernde Anforderungen eingegangen werden. Auch Anpassungen oder individuelle Kundenleistungen im Aufbau und der Funktionalität von Software können deutlich effizienter und schneller realisiert werden, wodurch sich die Entwicklungszeit verkürzen lässt.

Flexibilität: Dadurch dass vor allem die Anforderungen detailliert in User Stories spezifiziert werden und unmittelbar in der Implementierung umgesetzt werden, eröffnen sich größere Gestaltungsspielräume als sonst üblich.

Qualität: Qualität kann durch funktionale Tests und User Acceptance Tests früh überprüft werden.

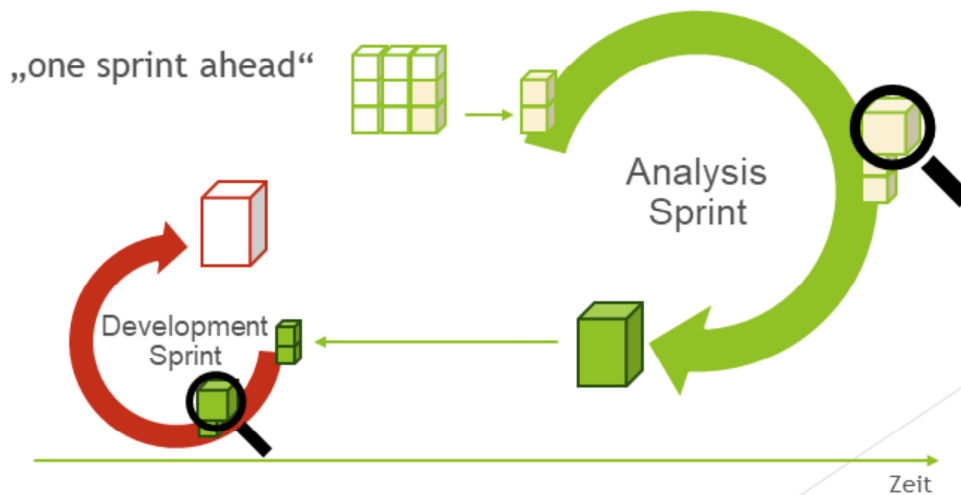
Transparenz: Sowohl der aktuelle Sprint Status und der Inhalt der Backlogs, als auch Verantwortlichkeiten und Prozessherausforderungen werden an alle kommuniziert. Durch diese Transparenz können Fortschritts- und Akzeptanzmaße bestimmt werden (KPIs), die den Prozess und das Produkt objektiv überprüfbar machen.

Skalierbarkeit: Die Arbeit erfolgt in kleinen Teams, die eine hohe Selbstorganisation aufweisen. In den Teams erfolgt die Koordination und gegenseitig Synchronisation ohne großen Mehrwand. Häufig ineffizientes Micro-Management im bidirektionalen Austausch (z.B. Vorgesetzter und Mitarbeiter) wird reduziert.

Bei der Betrachtung der Vorteile wird deutlich, dass der wichtigste **Erfolgsfaktor**, den die Studie der Stanley Group nennt - das „User Involvement“ – explizit berücksichtigt wird. Allerdings wirft dies die Frage auf, wie Nutzer so einbezogen werden können, dass sich UUX-Methoden einsetzen lassen, ohne dass dies die fristgerechte Beendigung eines Sprints verhindert.

Agilität und UUX-Evaluation

... sich ausschließende Alternativen oder vereinbare Optionen ?



Auf den ersten Blick erscheinen Scrum und UUX-Evaluationen keine „natürlichen Freunde“ zu sein. Ist z.B. ein Usability Test Teil eines Sprints, so kostet er in jedem Fall Zeit und Ressourcen, die der eigentlichen Implementierung verloren gehen. Deckt der Test sogar Mängel auf, die während des Sprints behoben werden müssen, kann dies zu Verzögerungen führen und den Sprinterfolg gefährden.

Auf den zweiten Blick ändert sich das Bild allerdings. Das Erkennen von Usability-Mängeln ermöglicht die Beseitigung von Defiziten, die für den Nutzer von elementarer Bedeutung sein können. Da agile Ansätze wie Scrum vor allem auf eine Erhöhung der Kundenzufriedenheit abzielen, leisten entsprechende Tests also einen wesentlichen Beitrag zur Erreichung dieses Ziels.

Gibt es also Wege beides miteinander zu vereinbaren? Eine Lösungsmöglichkeit bietet die Methode des **„one sprint ahead“**. Hier findet in einem Analyse-Sprint die Konzeption einer Funktionalität nach dem User Centered Design Prozess statt, so dass am Ende ein vorgetesteter (Teil-)Prototyp vorliegt, welcher anschließend im Entwicklungs-Sprint implementiert wird. Dessen Abnahme und Prüfung sollte dann nicht durch ein technisches Testen, sondern wiederum durch eine nutzerzentrierte Evaluation erfolgen.

Heute: Analoges Arbeiten mit Scrum

Derzeit werden die Scrum Backlogs meist auf analogen Boards angelegt und gepflegt.



Nachteile:

- Präsenz vor Ort erforderlich
- Transparenz über Team- und Unternehmensgrenzen hinweg eingeschränkt
- Medienbrüche zu digitalen Daten und Dokumenten

Heutzutage findet agiles Arbeiten in vielen Unternehmen noch mittels analoger Task Boards statt. Dies ist mit einigen Nachteilen verbunden.

Der Zugriff auf die Boards und die Teilnahme an entsprechenden Meetings erfordert die Präsenz vor Ort, was für mobile Mitarbeiter/innen und Telearbeiter/innen erforderlich hinderlich ist. Desweiteren wird dadurch die Transparenz des Arbeitsstandes über Team- und Unternehmensgrenzen hinweg stark eingeschränkt. Und zu guter Letzt entstehen Medienbrüche zu digitalen Daten und Dokumenten.

Digitalisierte Agilität als zukünftige Alternative

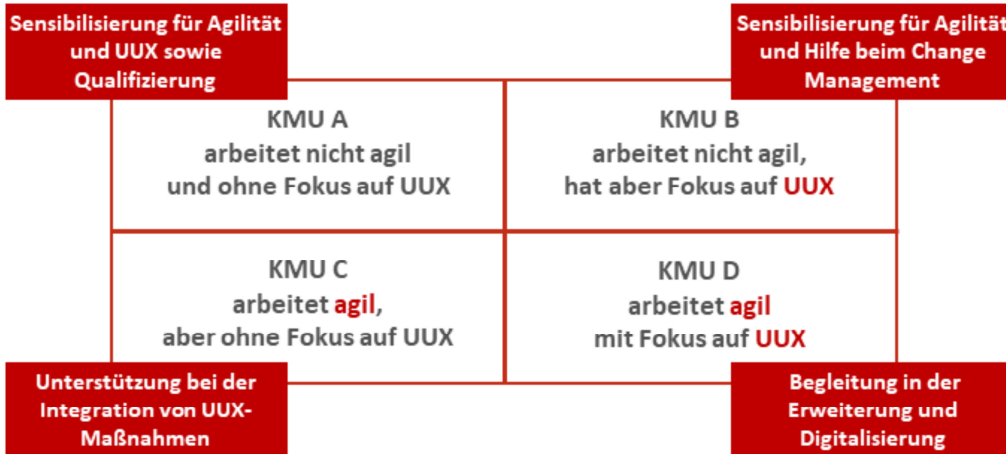


Abhilfe verspricht die Umstellung auf digitale Arbeitsmittel, die eine Vernetzung ermöglichen, verschiedene Endgeräte einschließen und in den agilen Prozessen eingesetzt werden können. Eine solche „**digitalisierte Agilität**“ ermöglicht:

- die Vernetzung von Mitarbeiter(inne)n verschiedener Abteilungen bzw. Teams
- Telearbeit
- die Einbindung mobiler Mitarbeiter/innen und verschiedener Endgeräte
- die Partizipation von Lieferanten, Kunden, Dienstleistern und Beratern

... und somit die Erhöhung von Transparenz und Partizipation insgesamt.

Zielgruppen in der Region Ost



Fazit: Gute Usability und positives Nutzererleben sind wichtige Qualitätsmerkmale eines Systems. Um beides zu gewährleisten, müssen Kunden und zukünftige Systemnutzer möglichst frühzeitig und kontinuierlich in die Systementwicklung einbezogen werden. Dies wiederum erfordert den Einsatz von UUX-Methoden und damit entsprechende Kenntnisse bei dem entwickelnden Unternehmen. Agile Methodologien, wie z.B. Scrum, bieten praktikable Möglichkeit zur Durchführung von UUX-Evaluationen, wie etwa durch den Ansatz des „one sprint ahead“.

Um die Vorteile zu nutzen, die sowohl UUX-Methoden als auch agile Entwicklungsmethoden bieten, sollen Unternehmen in der Region für beide Themenfelder sensibilisiert und bei Bedarf qualifiziert werden. Darüber hinaus gilt, es die Potenziale digitaler Tools und Arbeitsmittel für agiles Arbeiten zu erschließen.

Vor diesem Hintergrund ergeben sich vier Zielgruppen von entwickelnden Unternehmen, an die sich das Angebot des Kompetenzzentrums in der Region Ost richtet.






Unser Leistungsangebot für die 4 Zielgruppen ...

... zu den Themen UUX, Agilität und Digitalisierung:

- Trainings und Workshops
- Arbeitsunterlagen zum Selbststudium
- Demonstrationen und Ausprobieren
- Unterstützung beim Change Management
- Pilotprojekte

Region Ost: Projektpartner und Schwerpunkte



-  Technische Universität Berlin
 - Wissenstransfer
 - Demonstrator „Digitalisierte Agilität“
-  Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences
 - Analyse agiler Arbeit
 - Best Practice Beispiele
-  movingimage
 - Umsetzungsprojekt
-  useTree
 - Pilotprojekte
-  Berlin Partner für Wirtschaft und Technologie
 - Öffentlichkeitsarbeit
 - Vernetzung

Die Arbeiten in der Region Ost werden von der Technische Universität Berlin (Fachgebiet Kognitionspsychologie und Kognitive Ergonomie) koordiniert.

Quellen, Tipps und Links

- <https://www.knowledgetrain.co.uk/resources/practice/top-agile-books>
- <http://www.agilemanifesto.org/>
- <http://www.scrumguides.org/>
- <https://manifesto.co.uk/10-top-resources-agile-beginners/>
- <http://www.agile-ux.com/2009/01/18/12-best-practices-to-integrate-ux-into-agile-process/>
- <https://www.infoq.com/articles/standish-chaos-2015>