

Alles, was Sie über agile Projektentwicklung
mit Scrum wissen sollten

Version 1.0

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

Mit dem vorliegenden Artikel möchten wir die agile Projektentwicklung mit Scrum etwas näher beleuchten, wichtige Rollen und Tools sowie das Vorgehen in der Praxis kurz erläutern und auch auf das Thema Pricing und Kalkulation eingehen. Vorab möchten wir aber gleich noch mit einem Trugschluss aufräumen: Obwohl es bei Scrum keine ausufernde Konzeptphase gibt, bedeuten dies mitnichten, dass man planlos agiert. Genau das Gegenteil ist der Fall. Aber lesen Sie einfach selbst...

Während man beim klassischen Entwicklungsansatz nach dem sog. Wasserfall-Modell möglichst detaillierte Vorarbeiten leistet mit entsprechend genauen Arbeitsanweisungen, erhalten Scrum Teams entsprechend genaue Zielvorgaben, was die Zielerreichung anbelangt. Der Weg dorthin entwickelt sich jedoch innerhalb eines hochqualifizierten und interdisziplinären Scrum Teams während der Implementierung. Insofern kann auch der erste mögliche Trugschluss, Scrum sei „planlos“, widerlegt werden. Bei Scrum wird lediglich die genaue Art der Umsetzung nicht vorgegeben, sondern im Team erarbeitet, wobei hier zum einen der gruppenspezifische Prozess im Team vorteilhaft zum Tragen kommt und zum anderen natürlich auch laufende Erkenntnisse permanent in die Arbeit einfließen.

Der Scrum-Ansatz: Zerteilung komplexer und umfangreicher Entwicklung in kleine Teilprojekte, die nacheinander in sog. Sprints, die in der Regel zwei bis vier Wochen dauern, umgesetzt werden, und bei denen das Ziel die Auslieferung von funktionsfähigem und qualitativ hochwertigem Code darstellt.

Scrum akzeptiert, dass der Entwicklungsprozess nicht vorherzusehen ist.

Das oberste Ziel in einem Scrum-Projekt besteht darin, die bestmögliche Software unter Berücksichtigung der Kosten, der Funktionalität, der Zeit und der Qualität abzuliefern!

Dabei charakterisiert sich Scrum – insbesondere im direkten Vergleich mit klassischen Entwicklungsmethoden – durch die nachfolgenden drei Prinzipien:

- › **Transparenz:** Der Fortschritt und die Hindernisse eines Projektes werden täglich und für alle sichtbar festgehalten.
- › **Überprüfung:** In regelmäßigen Abständen werden Produktfunktionalitäten geliefert und beurteilt.
- › **Anpassung:** Die Anforderungen an das Produkt werden nicht ein und für alle Mal im Vorfeld festgelegt, sondern nach jeder Lieferung eines Teilprojektes neu bewertet und bei Bedarf angepasst.

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

Im Jahre 2001 wurden von einigen IT-Vordenkern zwölf Thesen zu Scrum aufgestellt, die zum besseren Verständnis beitragen, das mittlerweile berühmte „Agile Manifest“:

- › Unsere höchste Priorität ist es, den Kunden durch frühe und kontinuierliche Auslieferung wertvoller Software zufrieden zu stellen.
- › Heiße Anforderungsänderungen selbst spät in der Entwicklung sind willkommen. Agile Prozesse nutzen Veränderungen zum Wettbewerbsvorteil des Kunden.
- › Liefere funktionierende Software regelmäßig innerhalb weniger Wochen oder Monate und bevorzuge dabei die kürzere Zeitspanne.
- › Fachexperten und Entwickler müssen während des Projektes täglich zusammenarbeiten.
- › Errichte Projekte rund um motivierte Individuen. Gib ihnen das Umfeld und die Unterstützung, die sie benötigen und vertraue darauf, dass sie die Aufgabe erledigen.
- › Die effizienteste und effektivste Methode, Informationen an und innerhalb eines Entwicklungsteam zu übermitteln, ist im Gespräch von Angesicht zu Angesicht.
- › Funktionierende Software ist das wichtigste Fortschrittsmaß.
- › Agile Prozesse fördern nachhaltige Entwicklung. Die Auftraggeber, Entwickler und Benutzer sollten ein gleichmäßiges Tempo auf unbegrenzte Zeit halten können.
- › Ständiges Augenmerk auf technische Exzellenz und gutes Design fördert Agilität.
- › Einfachheit – die Kunst, die Menge nicht getaner Arbeit zu maximieren – ist essenziell.
- › Die besten Architekturen, Anforderungen und Entwürfe entstehen durch selbstorganisierte Teams.
- › In regelmäßigen Abständen reflektiert das Team, wie es effektiver werden kann und passt sein Verhalten entsprechend an.

Quelle: <http://agilemanifesto.org/iso/de/principles.html>

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

Während in der klassischen Entwicklung häufig isoliert entwickelt wird und jeder Entwickler sein „eigenes Süppchen“ kocht, steht bei Scrum der Team-Gedanke im Vordergrund. So unterscheidet man innerhalb von Scrum drei klassische Rollen sowie drei dazugehörige Gruppen, die man jedoch auch aus anderen Entwicklungsansätzen kennt.

Auf der eine Seite steht das klassische Scrum-Team, das die nachfolgenden Rollen beinhaltet:

Product Owner: Diese Rolle kann als verlängerter Arm des Kunden gesehen werden. Der Product Owner gibt die Anforderungen und die strategische Marschrichtung inkl. Priorisierung von Anforderungen bzw. Tasks vor. Diese werden in Abstimmung mit dem Entwicklungsteam im sog. Product Backlog als sog. User Stories erfasst. Damit werden die gewünschten bzw. benötigte Funktionalitäten aus der Sicht des Users beschrieben. Der Product Owner überprüft am Ende eines Sprints auch die gelieferte Software im Hinblick auf Verwendbarkeit. Er alleine entscheidet über Features, Kosten und Timings (Auslieferung). Entscheidungen des Product Owners sind dabei verbindlich. Er ist für das Endergebnis und die wirtschaftlichen Aspekte verantwortlich. Normalerweise wird der Product Owner vom Auftragnehmer gestellt und stellt die Brücke zwischen Kunde und dem restlichen Scrum Team dar.

ScrumMaster: Er ist für den Erfolg eines Scrum-Projektes verantwortlich und für die Einführung der entsprechenden Scrum-Regeln. Er moderiert die anfallenden Meetings und kümmert sich darum, dass etwaige Hemmnisse im Scrum-Prozess beseitigt werden. Dazu zählen u.a. mangelnde Kommunikation oder Störungen von außen durch neue/geänderte Anforderungen etc. Der ScrumMaster arbeitet eng mit dem Entwicklungsteam zusammen, gehört aber nicht dazu, d.h. als Führungskraft gegenüber dem Entwicklungsteam übernimmt er ausschließlich administrative Aufgaben, ohne dabei selbst zu entwickeln und ohne dabei konkrete Arbeitsanweisungen geben zu dürfen. Im Prinzip kann er als eine Art „Nanny“ für die Entwickler gesehen werden, der dafür sorgt, dass diese sich wohl fühlen und „funktionieren“ können.

Entwicklungsteam: Wie der Name bereits vermuten lässt, ist das Entwicklungsteam für die Implementierung der vom Product Owner im Backlog geforderten Funktionalitäten zuständig. Eine Besonderheit bei Scrum besteht darin, dass das Entwicklungsteam zu Beginn des ersten Sprints im Rahmen des sog. Sprint Plannings gemeinsam die Anforderungen prüft und diskutiert und in der Folge dann eigenmächtig ein Commitment zur Umsetzung des ersten Sprints und der dort enthaltenen Anforderungen abgibt und sich verpflichtet, die vom Entwicklungsteam dort selbst definierten Anforderungen zu erreichen. Dabei werden die Anforderungen in Form von User Stories vom Team gemeinsam geschätzt und in sog. Tasks heruntergebrochen, die normalerweise max. einen Tag in Anspruch nehmen dürfen. Während der komplette Entwicklung steht dabei immer das Team im Vordergrund und nicht einzelnen Entwickler. Das Team gewinnt zusammen und muss sich auch zusammen arrangieren, sofern unerwartete Probleme auftauchen oder z.B. ein Entwickler aus-

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

oder abfällt. Wichtig dabei ist auch der interdisziplinäre Ansatz, über den Entwickler z.B. auch Testaufgaben übernehmen und so einen umfassenderen Einblick erhalten und sich gegenseitig ergänzen können. Je nach Projektgröße und Anforderungen besteht ein Entwicklungsteam dabei in der Regel aus 5-9 Entwicklern. Die Entwicklungsteams im Scrum organisieren sich dabei stets selbst. D.h. sie geben die detaillierten Umsetzungsmethoden gemeinsam im Team vor und überwachen sich auch gegenseitig.

Auf der anderen Seite stehen nachfolgenden drei Rollen, die man jedoch auch aus diversen anderen Ansätzen und Entwicklungsmethoden kennt:

Customer: Diese Rolle beschreibt – was auch nicht sonderlich überraschend ist – den Kunden. Er wird in den meisten Fällen vom Auftraggeber gestellt, könnte theoretisch aber auch durch den Auftragnehmer bereitgestellt werden. Der Customer sollte im engen Austausch mit dem Product Owner stehen, dessen Ziel darin besteht, den Customer mit der gelieferten Software zu begeistern.

User: Als User wird der spätere Anwender der Software verstanden. Dies kann zugleich auch der Customer sein, was jedoch nicht zwingend der Fall ist. User sollten im Scrum-Prozess insbesondere beim Sprint-Planning sowie bei Reviews dabei sein, um das gelieferte Ergebnis aus Anwendersicht beurteilen zu können.

Management: Die Aufgabe des Managements besteht darin, die Rahmenbedingungen für ein erfolgreiches Scrum-Projekt zu schaffen. Dazu zählt neben der Bereitstellung der notwendigen Ressourcen auch die Unterstützung des Scrum-Teams durch Beseitigung etwaiger externer Faktoren, die den Erfolg des Scrum-Projekts gefährden könnten.

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

Nachdem wir die Rollen in einem Scrum-Projekt definiert haben, wollen wir in der Folge einen kurzen Blick auf die Vorgehensweise und die entsprechenden Tools werfen, die wir im Anschluss im Detail erläutern:

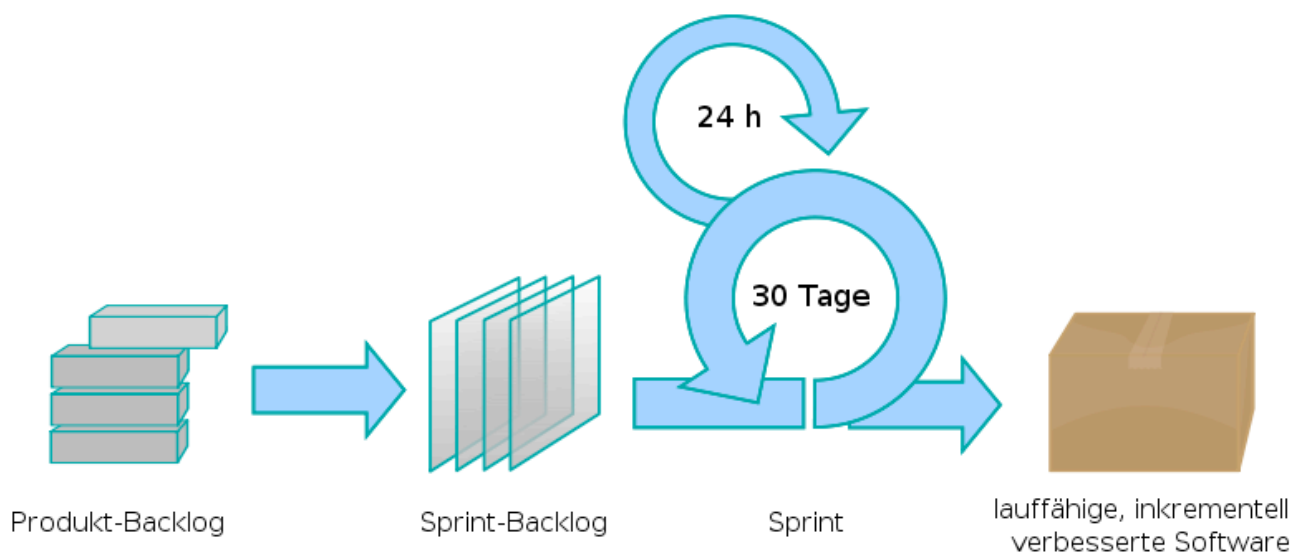


Abb.: Scrum-Prozess (Quelle: http://de.wikipedia.org/w/index.php?title=Datei:Scrum_process-de.svg&filetimestamp=20100701071608)

Product Backlog

Wie bereits erwähnt, werden die Anforderungen vom Product Owner im Product Backlog festgehalten. Unter dem Product Backlog versteht man eine priorisierte und rein nutzerorientierte Liste mit Anforderungen, die das zu entwickelnde Produkt berücksichtigen muss. Aufgrund der userorientierten Sichtweise spricht man daher auch von User Stories, die im Product Backlog eingetragen werden. Idealerweise erfolgt ein Eintrag in Form einer User Story als Antwort auf die Frage „Wer möchte was warum“ nach folgendem Muster:

„Als User möchte ich diese Funktionalität, damit ich folgenden Nutzen habe.“

Quelle: <http://borisgloger.com/2011/06/20/scrum-essentials-die-sieben-fragen-der-user-story/>

Dieses Backlog wird ausschließlich vom Product Owner gepflegt und laufend aktualisiert.

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

Sprint Planning Meeting 1

Im sog. Sprint Planning Meeting 1 werden diese Anforderungen dem Entwicklungsteam vom Product Owner vorgestellt. Einfach ausgedrückt wird damit dem Entwicklungsteam vorgestellt, was zu tun ist. Idealerweise ist während dieses Meetings auch ein User dabei, der dem Team – zusammen mit dem Product Owner – dann erklären kann, wie er sich die entsprechenden Funktionalitäten vorstellt. Das Entwicklungsteam hat hier Zeit, sich mit den Anforderungen vertraut zu machen, etwaige Fragen zu klären und die Anforderungen auch wirklich zu verstehen. Im Sprint Planning Meeting 1 werden am Ende auch die Abnahmekriterien für diesen Sprint definiert, die am Ende im sog. Sprint Review geprüft werden. Das Ziel eines jeden Sprints besteht in der Auslieferung gebrauchsfähiger und getesteter Software. Als zeitlicher Umfang für dieses Meeting können pro Sprint-Woche (in der Regel besteht ein Sprint aus 2-4 Wochen) 60 Minuten angesetzt werden. Die im Sprint Planning Meeting besprochenen Anforderungen werden im sog. Sprint-Backlog erfasst und dort überwacht.

Sprint Planning Meeting 2

Im anschließenden Sprint Planning Meeting 2 klärt das Entwicklerteam dann eigenverantwortlich, wie die zuvor vorgestellten Anforderungen umgesetzt werden. Dabei werden die Anforderungen in sog. Tasks zerlegt, die normalerweise nicht länger als einen Tag dauern sollen. Die Tasks werden dann am sog. Taskboard angebracht, wodurch ein sehr schneller Überblick zum Projekt, den dafür anstehenden Aufgaben und dem jeweiligen Status möglich wird. Auch für dieses Meeting sollten pro Sprint-Woche ca. 60 Minuten angesetzt werden. Das Ergebnis des Sprint Planning Meeting 2 wird im sog. Sprint Backlog festgehalten, bei dem jeder Task mit den nachfolgenden drei Ausprägungen versehen wird: „Task to Do“, „Work in Progress“, „Done“.

Sprint

Nach den beiden Sprint Plannings erfolgt die Umsetzung des ersten Sprints mit den zuvor definierten und abgestimmten Tasks. Hier werden grundsätzlich die im Rahmen des Sprint Plannings höher priorisierten Tasks zuerst erledigt. Ganz grundsätzlich wird dabei Task für Task vorgegangen, wodurch sichergestellt wird, dass die Anforderungen auch wirklich einzeln abgearbeitet und nacheinander fertig gestellt werden. Dadurch wird vermieden, dass noch nicht fertiggestellte Software ausgeliefert oder „zwischengeparkt“ wird. Während eines Sprints konzentriert sich das Entwicklungsteam ausschließlich auf die im Sprint definierten Anforderungen und Tasks. Die Aufgabe des ScrumMasters besteht darin, etwaige Störfeuer oder Hemmnisse zu beseitigen, so dass sich das Team ausschließlich auf die Fertigstellung und Auslieferung funktionsfähiger Software konzentrieren kann. Der ScrumMaster darf dabei keine Anweisungen an das Team erteilen, sondern lediglich an die Erreichung der zuvor definierten Ziele, zu denen sich das Team verpflichtet hat, erinnern. Das zentrale Ziel eines Sprints ist es, ein Stück potentiell auslieferbarer Software zu liefern. Selbst wenn aus Zeitgründen oder aus Gründen unerwarteter Komplexität bestimmte Aspekte in einem Sprint nicht (wie vorgesehen) realisiert werden können, endet der Sprint dennoch gemäß Zeitplan – und muss dabei zwingend ein in sich abgeschlossenes, funktionierendes Stück Arbeit produzieren. So

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

kann der Kunde nach jedem Sprint entscheiden, ob er das Teilstück des Gesamtprojekts evtl. schon produktiv einsetzen will – der Geschäftswert für den Kunden steht bei Scrum stets im Vordergrund.

Daily Scrum

Während des Sprints trifft sich das Entwicklungsteam täglich zum sog. Daily Scrum. Darunter versteht man ein zwingend auf maximal 15 Minuten definiertes Meeting, bei dem das Team den aktuellen Stand der Entwicklung sowie die aktuellen und die zuletzt bearbeiteten Tasks sowie die für heute anstehenden Tasks bespricht. Sollte sich herausstellen, dass einzelnen Tasks z.B. nicht innerhalb eines Tages erledigt werden können, können diese auch in kleinere Aufgaben heruntergebrochen werden. Sollten sich Fragen oder Probleme ergeben, die innerhalb der 15 Minuten nicht geklärt werden können, ist es die Aufgabe des ScrumMaster, sich um diese Punkte zu kümmern. Diese Punkte werden vom Scrum Master dann im sog. Impediment Backlog erfasst und weiter bearbeitet.

Burndown-Diagramm

Der aktuelle Entwicklungsstand wird idealerweise in einem sog. Burndown-Diagramm dargestellt, auf dem auf der x-Achse der Zeitverlauf und auf der y-Achse die Tasks angetragen werden. Eine Diagonale von links oben nach rechts unten stellt dabei den optimalen Projektverlauf dar. Je nach Abweichung der aktuellen Burndown-Linie von der Diagonale kann sehr schnell beurteilt werden, ob das Entwicklungsteam in Time ist oder aktuell Verzögerungen bestehen, die bis zum Ende aufgeholt werden müssen. Da alle Projektbeteiligten, also auch der Kunde (!!!), Zugriff auf dieses Burndown-Diagramm haben, wird höchstmögliche Transparenz während der Implementierung gewährleistet. Dadurch lässt sich bei etwaigen „Ausreißern“ auch frühzeitig gegensteuern bzw. geeigneten Gegenmaßnahmen ergreifen und das Projekt wieder auf „Schiene“ zu bringen.

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

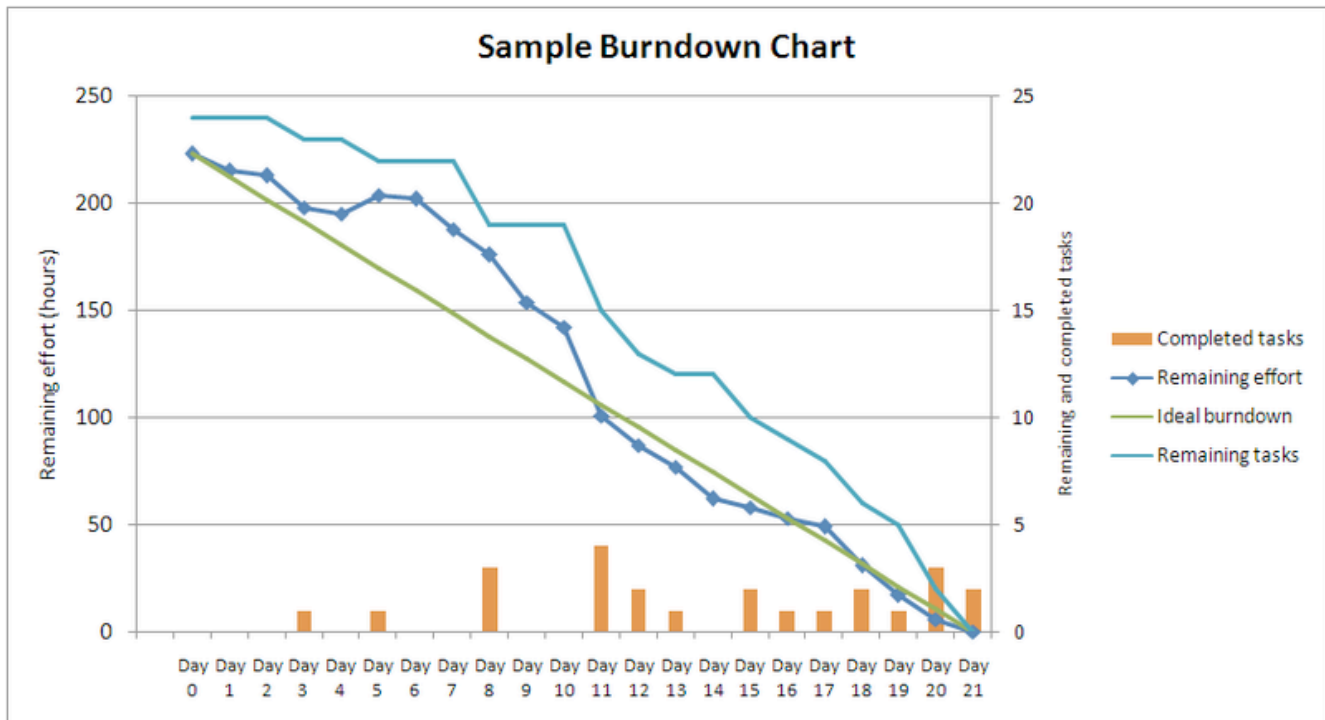


Abb.: Burndown Diagramm (Quelle: <http://en.wikipedia.org/wiki/File:SampleBurndownChart.png>)

Sprint Review

Am Ende eines Sprints erfolgt das sog. Sprint Review, bei dem das Entwicklungsteam dem Product Owner und vor allem auch dem User die aus den im Sprint Planning Meeting 1 definierten Anforderungen bzw. Tasks realisierte Softwarelösung vorstellt. Der Product Owner entscheidet dann anhand der vorab definierten Kriterien, ob das Ergebnis abgenommen werden kann oder nicht. Dabei besteht das Ziel in einem 100%-igen Zielerreichen. Sollten bestimmte Anforderungen nicht ganz erfüllt sein, werden diese vom Product Owner als neue bzw. nochmalige Tasks in das kommende Sprint Planning Meeting übernommen. Gleiches gilt für neue Ideen oder Anforderungen, die sich während des Sprint Reviews ergeben. In diesem Fall ist der ScrumMaster dafür zuständig, dass diese neuen Anforderungen an den Product Owner zur Übernahme ins Product Backlog weitergegeben werden.

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

Retrospektive

Mit Abschluss des Sprint Reviews erfolgt ein ganz entscheidender Abschnitt im Rahmen eines Scrum-Projektes. Das Scrum-Team (Product Owner, ScrumMaster, Entwicklungsteam) trifft sich geschlossen zur sog. Retrospektive, bei der etwaige Probleme, Learnings und Verbesserungsmöglichkeiten für den nachfolgenden Sprint diskutiert werden. Im Prinzip wird hier der vorangegangene Sprint nochmals kritisch hinterfragt und positive sowie negative Erkenntnisse notiert, mit dem Ziel, Verbesserungspotentiale für den neuen Sprint abzuleiten. Sofern es sich um Verbesserungen handelt, die das Team alleine betreffen, werden diese auch vom Team selbst gelöst. Etwaige andere Hemmnisse werden vom ScrumMaster im sog. Impediment Backlog aufgenommen und vom ihm dann an den Product Owner zur Bewertung für den folgenden Sprint weitergegeben.

Welche Vorteile bietet Scrum?

Die Vorteile der agilen Softwareentwicklung mit Scrum sind vielfältig. Zum einen ist es so, dass bei Scrum-Projekten die Flexibilität deutlich zunimmt, da man ein Großprojekt in kleinere Teilprojekte (Sprints) herunterbricht und diese einzeln und nacheinander realisiert. Nach jedem Teilprojekt erfolgt ein Review. Die Erkenntnisse fließen dann wieder in die nachfolgenden Teilprojekte ein. Damit können auch Änderung am Markt oder neue Erkenntnisse und Ideen zum Produkt jederzeit aufgegriffen und in einem der kommenden Sprints berücksichtigt werden.

Ein ganz entscheidender Vorteil bei Scrum liegt auch im ausgeprägten Team-Gedanken. Anforderungen und Probleme werden im Team diskutiert und gelöst. Durch den gruppendynamischen Prozess ergeben sich vielfach bessere Endergebnisse. Darüber hinaus arbeiten selbstorganisierende Team effektiver.

Einer der größten Vorteile von Scrum liegt in der höchstmöglichen Transparenz. Alle Projektbeteiligten haben zu jedem Zeitpunkt vollständige Transparenz im Hinblick auf den aktuellen Entwicklungsstand (Zugriff auf Backlogs, auf Burndown Diagramme mit dem Projektfortschritt und den angefallenen Projektzeiten, auf die die aktuelle Testumgebung) sowie etwaige Hemmnisse. Hier kann jeweils sehr zeitnah gegengesteuert werden, wodurch böse Überraschungen ausbleiben.

Durch die permanente Diskussion in der Gruppe sowie die Reflektion am Ende eines jeden Sprints wird ein kontinuierlicher Verbesserungsprozess gewährleistet, der sich sehr positiv auf die fachlichen Ergebnisse und auch auf die wirtschaftlichen Belange auswirkt. Aufgrund der „häppchenweisen“ Realisierung arbeiten Entwickler konzentrierter und auch fokussierter. Durch das Arbeiten im Team und den dadurch gegebenen gegenseitigen Ansporn steigt die Motivation und die Arbeitsergebnisse werden besser.

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

Die Kostenkalkulation

Häufig wird vom Auftraggeber ein Fixpreis-Angebot gefordert, was in der Praxis ganz offensichtliche Mängel und Nachteile mit sich bringt: Zum einen lassen sich bestimmte Anforderungen und Features im Vorfeld häufig kaum vernünftig schätzen, da hier Erfahrungswerte und detaillierte Informationen fehlen. Zum anderen ist es bei dem überwiegenden Teil von IT-Projekten so, dass sich während der Implementierung mitunter weitreichende Änderungen ergeben. Entweder weil sich Anforderungen – z.B. aufgrund von geänderten Marktbedingungen – ändern oder weil sich herausstellt, dass die im Vorfeld geplanten Lösungsansätze – aus welchem Grund auch immer – sich in der Praxis so nicht umsetzen lassen.

Das zieht zwei Probleme nach sich, die mitunter recht schwergewichtig sind und die die scheinbare Sicherheit eines Fixpreisangebotes für den Auftraggeber in einem anderen Licht erscheinen lassen:

- › Ein vernünftig agierender Dienstleister kalkuliert – und dies muss er aus wirtschaftlichen Gesichtspunkten auch – Unsicherheiten bzw. Unwägbarkeiten aufgrund von fehlenden Detailinformationen und Erfahrungswerten in sein Angebot mit ein, was für den Auftraggeber bedeuten kann, dass er für die gewünschte Leistung deutlich zu viel bezahlt.
- › Gerade Agenturleistung ist mitunter sehr preisgetrieben. Der Auftraggeber entscheidet sich daher möglicherweise für den vermeintlich günstigsten Anbieter. Machen wir uns hier aber nichts vor: Heutzutage hat niemand etwas zu verschenken. Genauso wenig wie Sie Ihre Produkte oder Dienstleistungen für lau abgeben, wird dies eine Agentur tun. Insofern erlebt man es auch recht häufig, dass der vermeintlich günstigste Anbieter den Zuschlag bekommt, dieser in der Umsetzung dann aber eben aus Budgetgründen entweder nicht sauber arbeitet, ein halbfertiges Produkt abgeliefert und/oder am Ende eine Nachkalkulation fordert, um seine Kalkulation „sauber“ halten zu können. Dadurch kann das vermeintliche Schnäppchen schnell in einem Desaster enden. Im schlimmsten Fall geht es dann jedoch nicht mehr „nur“ ums Geld, sondern möglicherweise auch ums Image und etwaige Schäden.

Aus unserer Sicht und aufgrund unserer 15-jährigen Erfahrung im Bereich Webentwicklung können wir mit nahezu 100%-iger Sicherheit feststellen, dass man bei Softwareprojekten seriöser- und richtigerweise im Vorfeld eigentlich nur Schätzungen abgeben kann, da hier einfach zu viele Unwägbarkeiten mitspielen, die nicht kalkulierbar sind. Insofern bietet ein Fixpreis-Angebot nur auf den allerersten Blick die vermeintliche Sicherheit für den Auftraggeber. Wie bereits ausgeführt, wird er die tatsächlichen Kosten – und diese werden sehr häufig höher ausfallen als das erste Angebot – anderweitig bezahlen, entweder unmittelbar in Form von Change Requests oder über Umwege, also etwa über Nachbesserungsmaßnahmen durch einen anderen Dienstleister aufgrund mangelnder Qualität bzw. nicht fertiggestellter Software. Darüber sollte man sich als Auftraggeber im Klaren sein: Die Rechnung wird kommen – so oder so!

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

Fairerweise muss man hier auch ganz klar erwähnen, dass es unter wirtschaftlichen Gesichtspunkten kaum möglich sein wird, bei einem komplexeren IT-Projekt ein Lasten- und Pflichtenheft in der Qualität zu erstellen, dass es als richtige Basis für ein Fixpreisangebot verwendet werden kann. Der Aufwand hierfür wird am Markt kaum bezahlt werden, wodurch sich dann einmal mehr die Frage stellt, ob ein günstigeres „Alibi-Lasten- und Pflichtenheft“ dann überhaupt Sinn macht.

Scrum ist insofern der für alle Beteiligten ehrlichste Ansatz. Ein Großprojekt wird in kleinere Teilprojekte zerlegt und dann einzeln geschätzt. Es wird auch nur das abgerechnet, was tatsächlich an Aufwand entstanden ist. Durch die vollständige Transparenz hat der Auftraggeber höchstmögliche Sicherheit und kann auch jederzeit reagieren. D.h. er kann bei sich abzeichnenden Budgetüberschreitungen jederzeit – auch in Abstimmung mit dem Scrum-Team – gegensteuern und im Notfall natürlich auch abbrechen. Hier hat er auch immer den Vorteil, dass bis dahin entwickelte Software oder Teile davon weiterverwendet werden können, da das Ziel von Scrum in der Bereitstellung von funktionsfähiger Software bzw. Softwareteilen besteht. Das bedeutet: Transparenz und Bezahlung nur für die tatsächlich erbrachte Leistung! Aus unserer Sicht ist dies der deutlich bessere und auf lange Sicht gesehen auch für alle Beteiligten der wirtschaftlichste Ansatz.

In der Praxis hat sich bewährt, das Product Backlog im Rahmen eines bezahlten Workshops zusammen mit dem Kunden zu erstellen. Auf Basis der so erfassten Anforderungen kann eine erste Abschätzung der Aufwände („Wir gehen aufgrund der im Backlog erfassten Anforderungen von x Sprints á 2-4 Wochen mit x Personen aus.“) erfolgen. Durch diese Kalkulation kann der Projektumfang in einem sehr frühen Stadium bereits umrissen werden. Hier besteht für den Auftraggeber dann auch die Möglichkeit, die so ermittelten Werte zu deckeln oder einen niedrigeren Wert als Höchstgrenze für die Implementierung anzusetzen. Bei der Umsetzung wird dies dann entsprechend berücksichtigt. Grundsätzlich werden in der Folge nur die tatsächlich angefallenen Aufwände abgerechnet, wobei durch den jederzeitigen Zugriff auf die Projektmanagementtools diese Aufwände täglich eingesehen und überwacht werden können. Dadurch werden böse Überraschungen vermieden!

Insofern lässt sich mit Scrum – auch wenn dies nicht der eigentlichen Idee von Scrum entspricht – ein IT-Projekt auch mit einem vorab fixierten Budget realisieren. In diesem Fall wird – sofern das Budget für die gewünschten Features nicht ausreichen sollte – in Abstimmung mit dem Kunden die eine oder andere Funktionalität gestrichen oder angepasst, wodurch das definierte Projektbudget jedoch wieder gehalten werden kann ohne – und dies ist sicherlich ganz entscheidend – Kompromisse bei der Qualität eingehen zu müssen!

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

Zusammenfassung

Simon Baker hat vor längerer Zeit zum Thema Festpreisprojekte folgendes geschrieben:

„To say how much something will cost, you need to know in great detail exactly what needs to be built so that your estimates for the work can be accurate. But you can't define everything you want, in detail and up front, and get it exactly right so there will be no changes in the future. And no estimate is ever accurate (it wouldn't be an estimate if it was). [...] Don't base the contract with your vendor on conformance to a detailed requirements specification. If you do, you're saying all your good ideas happen at the start of a project and you're effectively betting all your money on a hole-in-one...“ (Quelle: <http://www.energizedwork.com/weblog/2007/05/fixed-price-contracts-dont-work.html>)

Aus unserer Sicht bleibt für Auftraggeber nur die Empfehlung, sich beim nächsten Projekt einmal auf das Thema Scrum einzulassen. Auch ein Festpreisangebot schützt den Auftraggeber nicht vor etwaigen Risiken und damit verbundenen Mehraufwänden, die dann vielleicht zwar nicht sofort zur Kasse schlagen, zu einem späteren Zeitpunkt aber in nahezu allen Fällen zum tragen kommen. Bei Scrum steht nicht ein fixer Preis für vorab genauestens definierte Features im Vordergrund, sondern die termingerechte Auslieferung bestmöglicher und funktionierender Software, die definierte Anforderungen erfüllen muss. Bei der Umsetzung bestehen hier mehr Freiheiten – was jedoch keinesfalls mit mangelnder Qualität gleichzusetzen ist.

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

Als Zusammenfassung sehen Sie nachfolgend nochmals die Vor- und Nachteile von Scrum in der Übersicht, wobei aus unserer Sicht die positiven Aspekte deutlich überwiegen:

Vorteile	Nachteile
› Höchstmögliche Flexibilität durch jederzeitige Änderung und Priorisierung von Anforderungen	› Erhöhter Kommunikations- und Abstimmungsaufwand
› Frühzeitige Ergebnisse & funktionsfähige Teilprojekte	› Ggf. Unklarheiten bei Zuständigkeiten von interdisziplinären Themen
› Verbesserung in der Time to Market	› Möglichkeit zur Verzettelung im Detail bei unerfahrenen Teams
› Erhöhung der Kommunikation im Team und dadurch Verbesserungspotentiale	› Verschlechterung der Effizienz durch fehlende Beseitigung bekannter Probleme aus den Reviews
› Größtmögliche Transparenz für alle Projektbeteiligten	› Ggf. müssen Unternehmensstrukturen geändert werden, damit das Team selbstorganisiert arbeiten kann und die notwendige Unterstützung erhält
› Realistische und genaue Aufwandsschätzungen im Team	
› Frühzeitiges Erkennen von Problemen mit entsprechenden Handlungsmöglichkeiten	
› Verbesserung der Qualität durch konsequentes Teamwork	
› Fixe Termine	

Agile Projektentwicklung mit Scrum

weiterkommen mit TechDivision

Was aus unserer Sicht noch besonders herauszustellen ist: Auch mit Scrum ist man vor Sackgassen in einem IT-Projekt nicht geschützt, man erkennt diese in der Regel jedoch deutlich schneller und kann dementsprechend frühzeitig reagieren. Insofern eignet sich Scrum immer dann ganz besonders gut, wenn bereits zu Beginn des Projektes bei einer Vielzahl von Punkten noch Klärungsbedarf besteht, gewisse Unsicherheiten vorhanden sind bzw. bereits im Vorfeld zu erkennen ist, dass es hier noch zu diversen Änderungen und Anpassungen im Projektverlauf kommen kann bzw. wird.

Sollten Sie noch Fragen zur agilen Projektentwicklung mit Scrum haben, stehen wir Ihnen natürlich jederzeit gerne zur Verfügung.

TechDivision GmbH
Spinnereiinsel 3a
83059 Kolbermoor

Tel. +49 8031 221055-0
Fax +49 8031 221055-22

Email: info@techdivision.com
Web: www.techdivision.com

Über TechDivision

Als etablierte Internetagentur unterstützt TechDivision seit 1997 namhafte nationale und internationale Kunden bei der ganzheitlichen Planung, Konzeption und Umsetzung von webbasierten Technologien. Der Fokus liegt in der Realisierung von E-Commerce-Shop-Lösungen basierend auf Magento und der Entwicklung von Unternehmenswebseiten und Intranetlösungen mit dem Content Management System TYPO3.

TechDivision ist „Magento Gold Partner“ der ersten Stunden und kann als erfahrener Magento-Dienstleister zwischenzeitlich weit mehr als 30.000 Stunden Magento-Projekterfahrung vorweisen. Als „TYPO3 Association Gold Member“ verfügt TechDivision im Bereich TYPO3, Web-Entwicklung (PHP, JAVA, XHTML, CSS, Flash) und Webdesign ebenfalls über umfangreiches und langjähriges Know-How. Zahlreiche namhafte Kunden vertrauen inzwischen auf das Know-how „made in Kolbermoor“.

Wir entwickeln unsere Projekte selbst mit Scrum und können Ihnen daher unsere Erfahrungen in diesem Whitepaper weitergeben.