



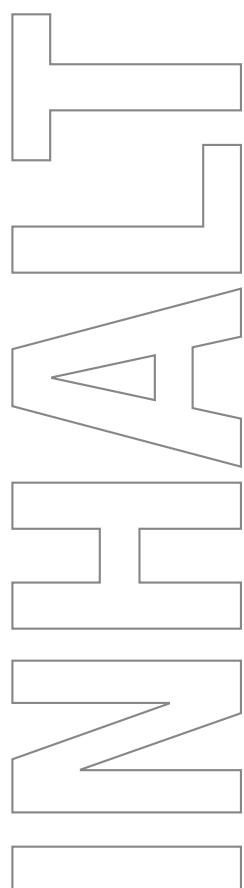
MODERNES (PROJEKT-) MANAGEMENT

SCRUM, KANBAN, MANAGEMENT 3.0 & CO.

DAS WHITEPAPER FÜR ENTSCHEIDER

INHALTSVERZEICHNIS

1. VORWORT	6
2. EINLEITUNG	7
3. „PLAN DRIVEN DEVELOPMENT“ VS. „VISION DRIVEN DEVELOPMENT“	8
4. AGILES VORGEHEN IST ITERATIV UND INKREMENTELL. WAS HEISST DAS?	9
4.1. Inkrement vs. Iteration verbildlicht: Die Mona Lisa	9
4.2. Was ist das kleinste shippable increment der Mona Lisa?	10
4.3. Was ist die einfachste Iteration des Inkrements, die shippable ist?	10
4.4. Woher weiß man, was Business Value hat?	11
5. AGILE ANFORDERUNGSANALYSE MIT IMPACT MAPPING	12
5.1. Was ist Impact Mapping und wofür wird es genutzt?	12
5.2. So funktioniert Impact Mapping	13
6. AGILES PROJEKTMANAGEMENT MIT SCRUM	15
6.1. Die Scrum-Rollen	17
6.1.1. Product Owner	17
6.1.2. Scrum Master	17
6.2. Scrum Meetings	18
6.2.1. Sprint Planning 1	18
6.2.2. Sprint Planning 2	18
6.2.3. Daily Scrum	18
6.2.4. Sprint Review	19
6.2.5. Retrospektive	19
6.3. Sprint	20
6.4. Die Scrum-Artefakte	21
6.4.1. Sprint Backlog	21
6.4.2. Product Backlog	21
6.4.3. Impediment Backlog	22
6.4.4. Burndown-Diagramm	23
6.5. Welche Vorteile bietet Scrum?	23
6.6. Die Kostenkalkulation in Scrum	24
6.7. Die größten Fehler im Scrum-Prozess	26
6.8. Übersicht der Vor- und Nachteile von Scrum	29
6.9. Zusammenfassung: Scrum	30





INHALTSVERZEICHNIS

7. AGILES PROJEKTMANAGEMENT MIT KANBAN	31
7.1. Zur Geschichte von Kanban	31
7.2. Kanban in der Softwareentwicklung	31
7.3. Kanban in Projekten	32
7.4. So funktioniert Kanban	32
7.5. Vorteile im Überblick	34
7.6. Konsens – Die Basis des Kanban-Change	34
7.7. Sieben Schritte für erfolgreiches Kanban	35
8. KANBAN VS. SCRUM	37
8.1. Vorgaben bei Kanban	38
8.2. Vorgaben bei Scrum	38
8.3. Scrum und Kanban im direkten Vergleich	39
8.4. Wann eignet sich Kanban und wann Scrum besonders	40
8.5. Von Wasserfall über Scrum zu Kanban	41
9. VERTRAGSGESTALTUNG BEI WEBPROJEKTEN	42
9.1. Klassisches oder agiles Projektmanagement	42
9.2. Probleme des klassischen „Wasserfalls“	42
9.3. Agile Projektmethode als Heilsbringer?	43
9.4. Was bedeutet Agilität rechtlich	44
9.5. Gretchenfrage: Werk- oder Dienstvertrag	44
9.6. Entscheidend ist der Vertragsinhalt	45
9.8. Zusammenfassung	46
10. TECHDIVISION BUBBLE BUDGETING	47
10.1. Bubble Budgeting	48
10.2. Wozu brauche ich Agile?	50
10.3. Wozu brauche ich Lean?	50
10.4. Alter Wein in neuen Schläuchen	50
11. WARUM EIN TRADITIONELLER AUSSCHREIBUNGSPROZESS VERALTET IST	51
12. AGILE METHODEN SIND ERFOLGREICH(ER) UND AUF DEM VORMARSCH	53
13. DER AGILE EVALUIERUNGSPROZESS	56
13.1. Klassisches Vorgehen vs. agiles Vorgehen	57
13.2. Die vier Phasen der agilen Software-Evaluierung	58
13.2.1. Discovery Phase	59
13.2.2. Translation-Phase (User-Stories/ Epics)	63
13.2.3. Auswahl- und Präsentationsphase	65
13.2.4. Implementierungsphase	68
13.3. Kosten für das Testprojekt	70



INHALTSVERZEICHNIS

14. FAZIT	71
15. AGILE ARBEITSWEISEN EIGNEN SICH NICHT NUR FÜR ENTWICKLUNG...	72
15.1. Problem der Usability Engineers	73
15.2. Problem des Entwicklungsteams	73
15.3. Lösungsansatz	74
15.4. 3 Praxistipps für eine erfolgreiche, interdisziplinäre Zusammenarbeit	75
15.5. Lösungsansatz	76
15.6. Ausblick	77
16. DAS DIGITALISIERUNGSMANAGEMENT-DILEMMA	78
16.1. Wir machen jetzt Scrum und sind damit modern und agil	79
16.2. Agilisierung in der Praxis – unser Ansatz und einige Erfahrungen	80
16.3. Ein kurzer historischer Abriss	81
17. THEORIE X UND THEORIE Y ALS AUSGANGSPUNKT	82
17.1. McGregor's kritische Entscheidung	83
18. FLOW	84
19. AGILE MANAGEMENT INNOVATIONS (AMIS) - HILFSMITTEL ZUR AGILISIERUNG	86
19.1. Die 26 AMIS im Überblick	86
20. DER TECHDIVISION AGILISIERUNGSPROZESS UND EINIGE LEARNINGS DARAUS	88
21. DIE ERGEBNISSE UNSERES AGILISIERUNGSWORKSHOPS	89
22. DIE RECHNUNG OHNE DEN WIRT	92
23. FAZIT	93
24. AUTOREN	94
25. TECHDIVISION GMBH	95
26. WAS WIR FÜR SIE TUN KÖNNEN...	96
27. DAS SAGEN KUNDEN ÜBER UNS...	97
28. DIE ERGEBNISSE UNSERES AGILISIERUNGSWORKSHOPS	98
29. DAMIT NEHMEN SIE FAHRT AUF	99
30. LESETIPPS	100
31. IMPRESSUM	101



VOLLER LESESTOFF...

Bei vorliegendem Dokument handelt es sich um einen Auszug unseres kostenlosen Whitepapers „Modernes (Projekt-)Management – Scrum, Kanban, Management 3.0 & Co.“

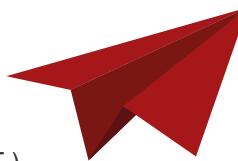
Auf rund 100 Seiten erläutern wir Ihnen ausführlich und praxisnah Wissenswertes zu modernem (agilem) Projektmanagement, das nicht nur für IT-Abteilungen relevant ist. Dabei gewähren wir auch einen Blick in unsere Agilisierungs- und Managementansätze und die damit einhergehenden Praxiserfahrungen.

Das Dokument kann unter nachfolgendem Link kostenlos heruntergeladen werden:
www.techdivision.com/projektmanagement-whitepaper

Hier geht's zur Vollversion



VORWORT



Das vorliegende Whitepaper möchte ich allen leidgeplagten (IT-) Projektverantwortlichen widmen, die sich mit den Tücken ständig ändernder Anforderungen bei (IT-)Projekten herumschlagen. Aufgrund von vorab – meist zu einem zu frühen Zeitpunkt – fest definierten Vorgaben werden Projekte meist unnötig in ein starres „Korsett“ gezwängt, das keinerlei Aktion, sondern nur noch Reaktion zulässt. In Zeiten sich extrem dynamisch ändernder Umgebungen kann so kaum mehr der nötige Freiraum für einen erfolgreichen Projektabschluss bestehen.

Darüber hinaus möchten wir mit dem vorliegenden Whitepaper alle Entscheidungsträger überzeugen und sie ermutigen, einen neuen Weg einzuschlagen: Pflichtenhefte und bis ins letzte Detail durchgeplante Konzepte sowie klassisches „Command & Control“ waren gestern. Lassen Sie Ihren Projekten und Mitarbeitern die Freiheit, sich dynamisch gemeinsam mit ihrem Umfeld zu entwickeln und erleben Sie die Vorteile des agilen Projektmanagementansatzes, der zu einer erfolgreichen Bewältigung von IT-Projekten beiträgt.

In diesem Sinne wünsche ich Ihnen die nötige Geduld, einen gewissen Weitblick und ein ganzes Stück Vertrauen in Ihre Projektverantwortlichen. Ich bin mir sicher, dass diese das in sie gesetzte Vertrauen und die damit mögliche Flexibilität nicht missbrauchen werden und stattdessen noch praxisorientiertere und auch bessere Endergebnisse abliefern werden.

Sollten Sie noch kein entsprechendes Basiswissen in modernem (Projekt-)Management besitzen oder Ihre bereits vorhandenen Kenntnisse nochmals auffrischen wollen, bieten wir Ihnen individuelle Workshops und Trainings mit entsprechendem Praxisbezug an. Sprechen Sie einfach mit uns! Gerne beraten wir Sie kostenlos und unverbindlich.



Ihr Josef Willkommer

EINLEITUNG

Wer kennt im IT-Umfeld nicht das altbekannte und auch oftmals „verdammte“ Projektmanagement anhand des sog. Wasserfall-Modells. Hierbei wird in einer mitunter recht umfangreichen Konzeptphase im Vorfeld zuerst ein Lastenheft erstellt, das die fachlichen Anforderungen an ein Projekt beschreibt. Die technische Übersetzung dieser Anforderungen erfolgt dann im sog. Pflichtenheft, das bei Bedarf noch um technische Feinkonzepte ergänzt wird. Jedoch vergehen nicht selten viele Wochen, ja sogar Monate, bis mit der eigentlichen Entwicklung anhand der so erstellten Dokumente begonnen werden kann. Dabei beinhaltet diese Vorgehensweise einen ganz entscheidenden Nachteil, der in der Vergangenheit aber dennoch immer wieder aufs Neue übersehen wurde: Ein nicht unwesentlicher Teil der Anforderungen ist zu Projektbeginn entweder noch nicht bekannt oder lässt sich im Vorfeld nur grob skizzieren und definieren, da aufgrund von zunehmender Komplexität im Webumfeld häufig erst im Projektverlauf alle technischen Anforderungen bekannt und evaluiert werden können. Zudem gibt es, meist aufgrund sich ändernder externer Faktoren oder Marktgegebenheiten, nahezu immer notwendige Anpassungen, die in den meisten Fällen zur Konsequenz haben, dass das Endergebnis häufig ganz anders aussieht bzw. aussehen soll, als dies zu Beginn der Konzeption geplant war.

Bereits vor etlichen Jahren wurde dieser Umstand zum Anlass genommen, um über neue Projektmanagementansätze nachzudenken, die mehr Flexibilität und am Ende auch mehr Sicherheit für alle Beteiligten bieten sollen: Das agile Projektmanagement war geboren und mit zunehmender Dynamik, insbesondere im Webumfeld, haben auch die Nachfrage und die Bedeutung dieser agilen PM-Ansätze in den letzten Jahren zu Recht signifikant zugenommen.

Quelle: <http://www.scrumalliance.org>

Dabei haben sich insbesondere Scrum und Kanban in jüngerer Zeit etabliert. Mit vorliegendem Dokument möchten wir diese beiden Ansätze kurz vorstellen, auf deren Besonderheiten, Vor- und Nachteile eingehen und einige Tipps und Tricks zum Einstieg in die Welt des agilen Projektmanagements vermitteln.

An dieser Stelle empfehlen wir, sich mit diesen Ansätzen intensiv auseinander zu setzen, da wir davon überzeugt sind, dass insbesondere Webprojekte aufgrund der weiter zunehmenden Dynamik und Komplexität zukünftig nur noch agil erfolgreich bewerkstelligt werden können.

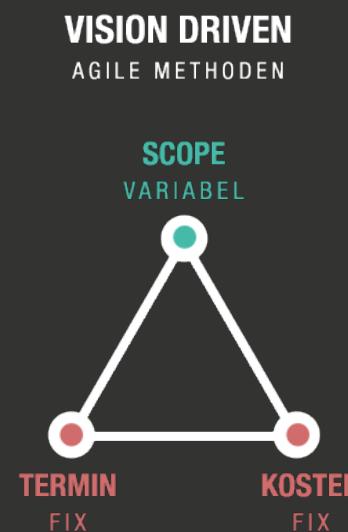
„PLAN DRIVEN DEVELOPMENT“ VS. „VISION DRIVEN DEVELOPMENT“

Beim klassischen Projektmanagement-Ansatz wird der Umfang der gesamten zu entwickelnden Lösung vorab festgelegt und genau definiert. Bei einer derartigen Planungs- und Spezifikationsphase – dem „Big Design Up-Front“ – stellt der Projektleiter bei Umsetzung eines Projekts oftmals fest, dass Zeit und Budget falsch kalkuliert sind oder dass das Projektteam die Bedürfnisse und Anforderungen des Kunden nicht zielführend bearbeiten konnte. Durch das „Plan-Driven-Development“ entstehen daher meist Stress, Unzufriedenheit und auch mangelnde Wirtschaftlichkeit. Nachfolgende Abbildung verdeutlicht den Unterschied zwischen dem klassischen Projektmanagement und agilen Methoden.

Quelle: <http://t3n.de/magazin/praxisbericht-scrum-kanban-scrumbuts-agiles-232822/>

Der agile Ansatz zeichnet sich vor allem dadurch aus, dass zu Beginn eines Projekts Zeit und Budget als Konstanten definiert werden. Zusammen mit dem Kunden werden dann Anforderungen ausgearbeitet, die sich innerhalb dieses Rahmens realisieren lassen. Die Rede ist dann vom sogenannten „Vision Driven Development“. Der Vorteil dieser Vorgehensweise liegt vor allem darin, dass der Kunde fortwährend den Verlauf des Projekts mitbestimmen kann und die einzelnen To-Do's von Iteration zu Iteration bestimmen kann (Scope Management). Das transparente Vorgehen hilft nicht nur den Entwicklern zielführend zu arbeiten. Auch der Kunde bekommt dadurch ein besseres Gefühl für das laufende Projekt. Durch die Value-Driven-Arbeitsweise kann das Team aus vorangegangenen Iterationen lernen und das Gelernte für die nächsten Schritte nutzen.

Quelle: <http://t3n.de/magazin/praxisbericht-scrum-kanban-scrumbuts-agiles-232822/>



AGILES VORGEHEN IST ITERATIV UND INKREMENTELL. WAS HEISST DAS?

Ein Inkrement ist ein fertiges Teilstück. Ergebnis einer Iteration ist dagegen eine neue (möglichst verbesserte) Version eines bestehenden Teilstücks oder auch des Ganzen.

Das fertige Stahlgerüst einer Brücke von Ufer zu Ufer, das man noch auskleiden muss, damit man die Brücke befahren kann, ist Ergebnis einer Iteration „Brücke bauen“.

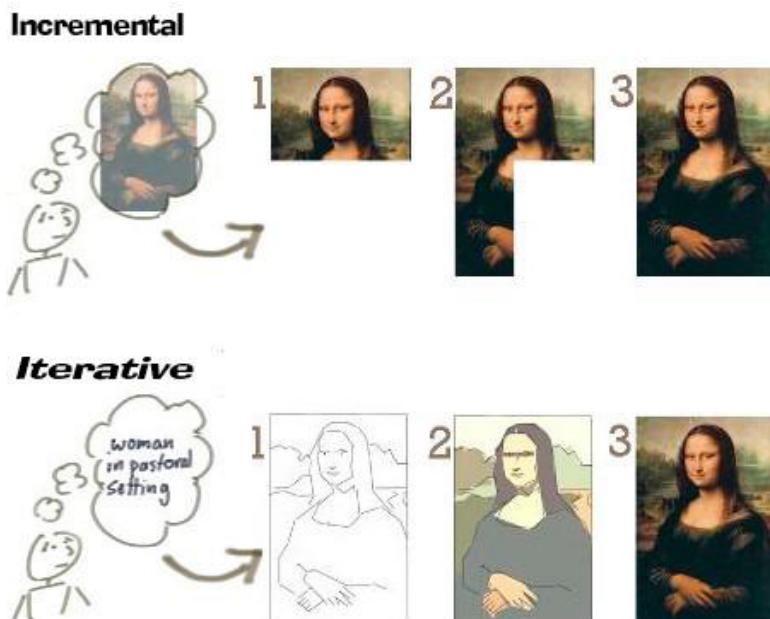
Die ganze Brücke ist da, aber eben erst einmal nur als Gerüst. Schwindelfreie können den Fluss so schon überqueren. Hinüberfahren kann man aber noch nicht. Die ersten 100 Meter fertige Brücke bis zum Pfeiler 2, mit allem Pipapo, fertig befahrbar, sind ein Inkrement der Brücke. Das Inkrement ist fertig, aber es ist noch keine Brücke, nur ein Teilstück. Niemand kommt über den Fluss, auch kein Fußgänger.

4.1. INKREMENT VS. ITERATION VERBILDLICHT: DIE MONA LISA

Bildlich zeigt die Mona Lisa, was gemeint ist. Das Beispiel ist von Jeff Patton, dem Erfinder der Story Maps für User Storys. Oben sehen wir inkrementelles Vorgehen, Stück für Stück wird das Bild in den Endzustand versetzt. Unten sehen wir iteratives Vorgehen, das gesamte Bild wird vom Start weg erstellt, erst einmal grob, dann wird es immer weiter verfeinert bzw. ausgemalt, bis das Gesamte fertig ist.

Quelle: http://www.agileproductdesign.com/blog/dont_know_what_i_want.html

Der Kern des agilen Vorgehens ist nun: Was ist das kleinste Inkrement, mit dem wir etwas Wertvolles liefern? Und was ist die einfachste Iteration, die wir dafür verwenden können?



4.2. WAS IST DAS KLEINSTE SHIPPABLE INCREMENT DER MONA LISA?

Vielleicht ihr Gesicht? Ja, die Kunstliebhaber würden etwas zahlen, um ein kleines Portrait (gemalt von da Vinci) von Mona Lisa zu bekommen, sagt der Gemälde-PO. Also nicht das gesamte Bild der Mona Lisa, sitzend vor einer Landschaft, sondern nur das Gesicht. Besser als nichts. Würden die Leute auch zahlen, um die linke obere Ecke des finalen Bilds zu besitzen? Weniger wahrscheinlich, sagt der Gemälde-PO. Fangen wir also mit dem Gesicht an, das scheint als erstes Inkrement den höchsten Business Value zu haben.

Unser Ziel ist dabei nicht, das große Ganze einfach nur in Teilstücke (Inkremeante) zu zerbrechen, die wir dann nach und nach liefern. Denn dann hat erst das Ganze, das sich am Schluss zusammensetzt, einen Wert.

Das ist zu spät. Vielmehr wollen wir Inkremeante wählen, die vermutlich (!) einen hohen Wert haben. Vielleicht kommt der Gemälde-PO und erzählt uns nach Präsentation des Inkrementes überrascht: „Portraits sind out! Kein Mensch will nur Gesichter! Hände! Das ist es. Wir brauchen die Hände dazu!“ Wir passen uns an den Markt an und sagen da Vinci: „Mach als nächstes die Hände.“ We increment to adapt.

Noch besser, als ein fertiges Inkrement abzuliefern, um zu sehen, ob es den Bedürfnissen der Nutzer entspricht, ist es aber, sich iterativ dem perfekten, fertigen Inkrement zu nähern. So merken wir noch schneller, ob Gesichter überhaupt gefragt sind.

4.3. WAS IST DIE EINFACHSTE ITERATION DES INKREMENTS, DIE SHIPPABLE IST?

Ein Gesicht schön zu malen, kostet Zeit und Geld, sagt da Vinci. Fangen wir deshalb mit einer ersten Iteration an: Eine Strichzeichnung des Gesichts?! Ist diese Strichzeichnung ein potentially shippable product? Ja, denn die Leute zahlen auch für eine Skizze von da Vinci, meint der PO. Wäre eine niedergeschriebene Absichtserklärung von da Vinci „Ich werde ein schönes Gesicht malen, es wird eine Frau zeigen, die lächelt, als habe sie sich einen kleinen Witz erzählt, ich sag's euch, das wird super“, wäre das ein potentially shippable product? Der Gemälde-PO geht davon aus, dass die Leute nicht viel für so etwas zahlen würden. Also nein. (Working software over comprehensive documentation)

Bei Scrum fallen die beiden „Rhythmen“ von Inkrementen und Iterationen übrigens zusammen (im Sprint). Das muss nicht so sein und ist in anderen agilen Methoden anders, z. B. im XP oder DSDM oder den Crystal Methods.

Noch mal zur Iteration: Ist eine Strichzeichnung, die einen Kreis, zwei Punkte (= Augen) und einen Halbkreis (= Lächeln) zeigt, ein akzeptables Ergebnis einer ersten Iteration? Der Gemälde-PO sagt nein, die Implementierungstiefe „Strichmännchen“ des Inkrementes „Gesicht“ der Mona Lisa würden die Kunstliebhaber mit höchster Wahrscheinlichkeit nicht akzeptieren. Ein bisschen mehr muss es schon sein.

4.4. WOHER WEISS MAN, WAS BUSINESS VALUE HAT?

„Ist das immer so?“ fragen wir den Gemälde-PO.
Nein, sagt er. Es ist höchstwahrscheinlich bei Keith
Haring anders als bei da Vinci. Keith Haring ging in der
Implementierungstiefe über Strichmännchen selten
hinaus. Ein Strichmännchen von Keith Haring kaufen die
Leute wie warme Semmeln, auf die Art verdienen wir
Geld.

Der Business Value ist kontextspezifisch,
domänen spezifisch usw. Der Kunde muss uns sagen,
was für ihn Wert hat, wir können es (meistens) nicht
ohne ihn wissen. Oft weiß er es selber nicht, dann
müssen wir ihm helfen, herauszufinden, was für ihn
welchen Business Value hat und wie man das testen
kann. Impact Mapping ist eine der vielen Möglichkeiten,
die dem Kunden dabei helfen können.



AGILE ANFORDERUNGSANALYSE MIT IMPACT MAPPING

5.1. WAS IST IMPACT MAPPING UND WOFÜR WIRD ES GENUTZT?

Impact Mapping ist eine strategische Planungstechnik und Anforderungsanalyse. Es hilft Unternehmen dabei, das Ziel während einer Projektarbeit nicht aus den Augen zu verlieren und fördert ein zielgerichtetes Arbeiten. Konzepte wie Lean Startup oder Continuous Delivery gewinnen dabei immer mehr an Bedeutung, während der herkömmliche, auf Vollständigkeit und Korrektheit abzielende Begriff der „Anforderung“ immer mehr in den Hintergrund gerät (vgl. <http://btdays.de/2014se/sessions/continuous-learning-agile-anforderungsanalyse-mit-impact-mapping>). In diesem Zusammenhang wurde von Gojko Adzic das sogenannte Impact Mapping entwickelt, welches sich des Problems annimmt und den Spagat zwischen Plan und Experiment schafft (vgl. <http://de.slideshare.net/springify/software-that-matters-agile-anforderungsanalyse-mit-impact-mapping>).

Projekte haben eine voneinander abhängige, dynamische Beziehung zu Menschen, anderen Projekten sowie Organisationen und Gemeinschaften um sie herum.

Impact Mapping versteht sich dabei als kooperatives und kreatives Verfahren, das einen besseren Überblick in einem laufenden Projekt verschaffen soll (vgl. <http://de.slideshare.net/springify/software-that-matters-agile-anforderungsanalyse-mit-impact-mapping>).

„Impact mapping helps to reduce waste by preventing scope creep and over-engineered solutions. It provides focus for delivery by putting deliverables in the context of impacts they are supposed to achieve. It enhances collaboration by creating a big-picture view that business sponsors and delivery teams can use for better prioritisation and as a reference for more meaningful progress monitoring and reporting. Finally, it helps to ensure that the right business outcomes are achieved, or that unrealistic projects are stopped before they cost too much, by clearly communicating underlying assumptions and allowing teams to test them.“

Quelle: <http://impactmapping.org/about.php>

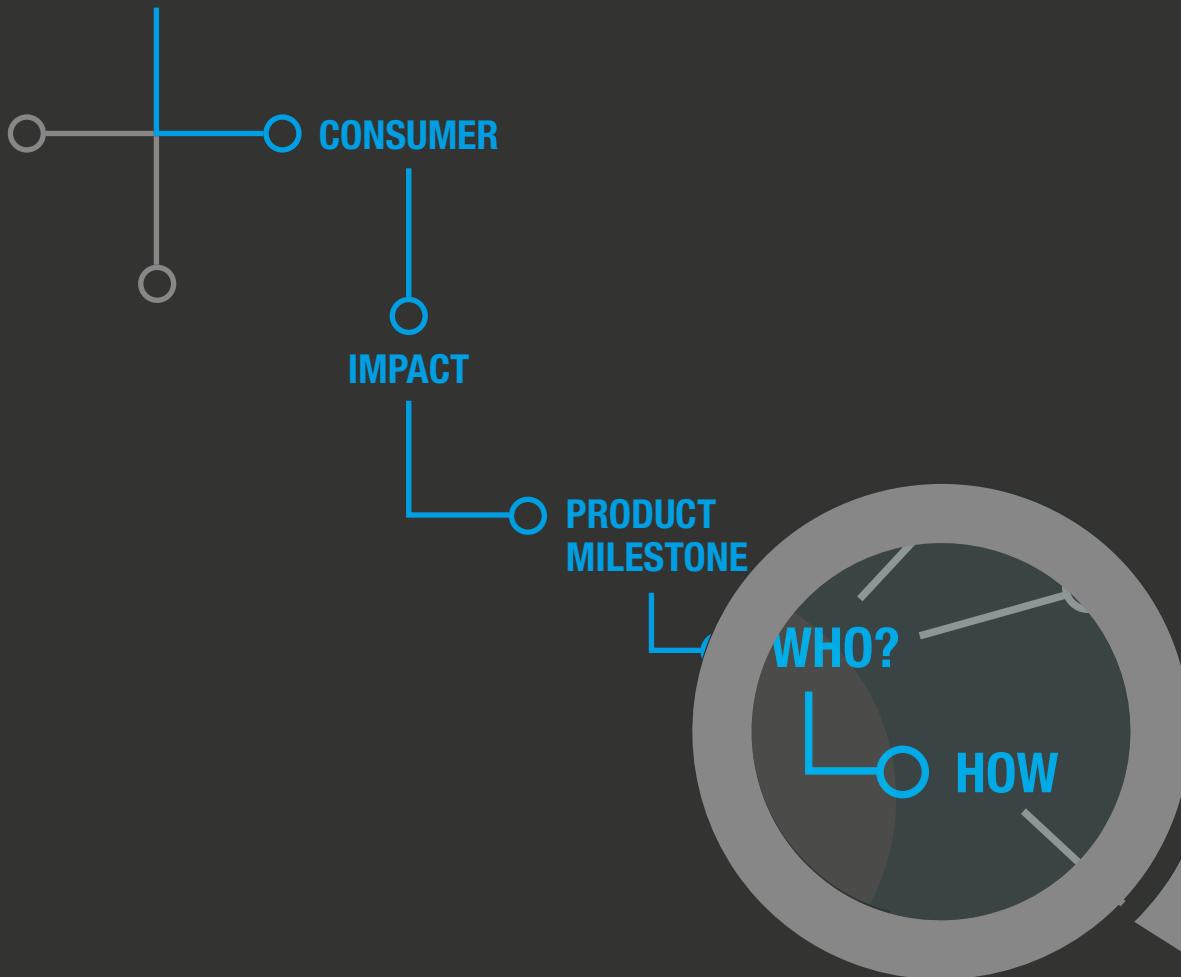
5.2. SO FUNKTIONIERT IMPACT MAPPING

Impact Mapping verbindet verschiedene Methoden miteinander und stellt Prozesse visuell dar. Die Methode dient somit zum Lösen komplexer Probleme und eignet sich vor allem auch für heterogene Gruppen. Vorkenntnisse werden kaum benötigt. Der Start beim Impact Mapping ist immer mit einem geschäftlichen Ziel verbunden. So kann der eigentliche Nutzen eines Projekts stärker fokussiert werden.

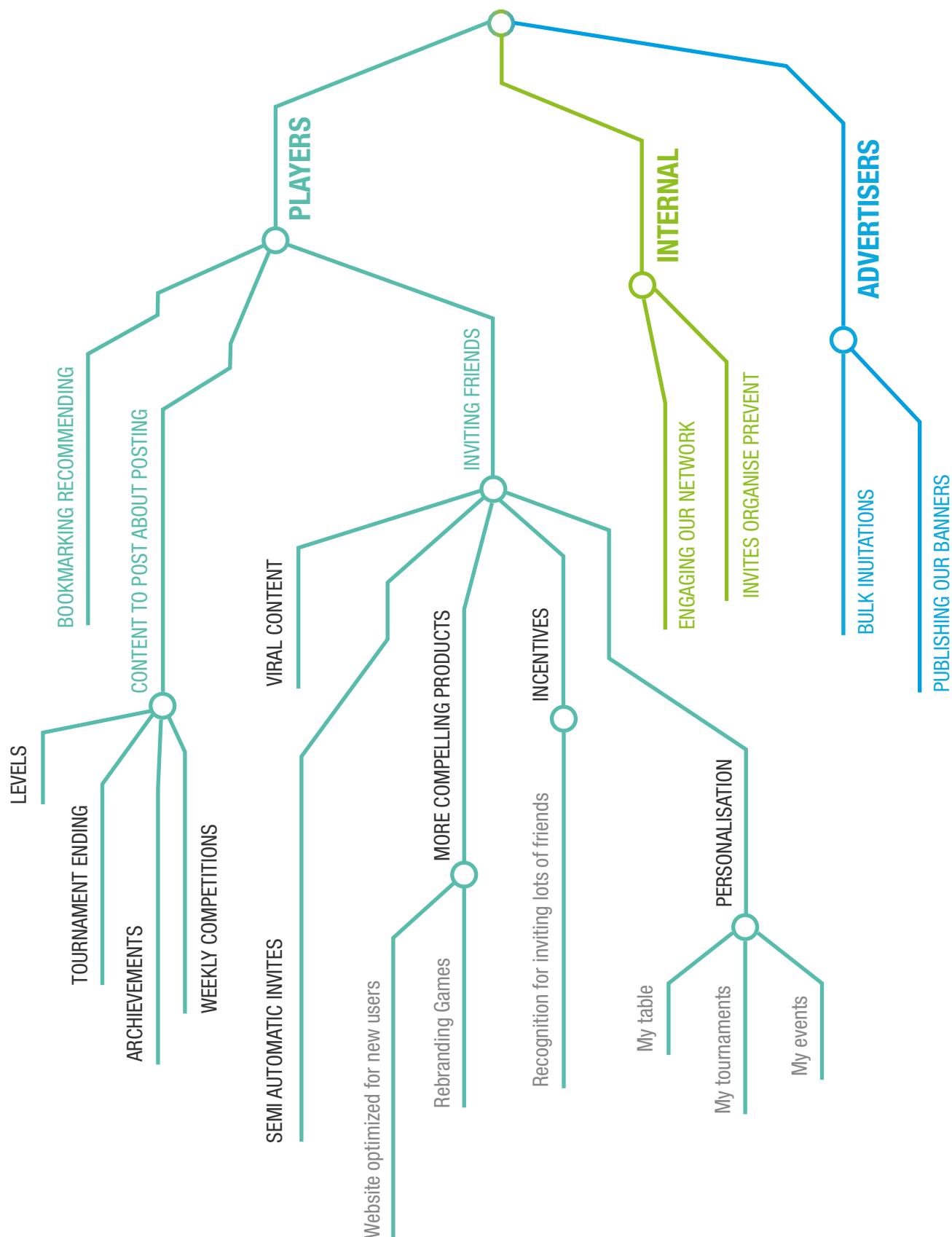
Impact Mapping bedeutet so viel wie „Auswirkungs-Zuordnung“ und besteht aus den Phasen „Vorbereitung“ (Preparation) und Zuordnung (Mapping). In der ersten Phase geht es vor allem darum, Ziele genau zu definieren, passende Messmethoden zu finden und den ersten Meilenstein festzulegen. In Phase zwei wird das Ganze dann in Form einer Mindmap-artigen Karte aufgearbeitet und visualisiert (vgl. <http://www.wolter.biz/tag/impact-mapping/>)

Beim Impact Mapping wird vor allem ein iteratives sowie agiles Arbeiten unterstützt, welches in vielen Unternehmen bereits zum Alltag gehört. Die graphische Herangehensweise fördert dabei die Entwicklung neuer Ideen (vgl. <http://www.wolter.biz/tag/impact-mapping/>).

VISION



1M PLAYERS



Quelle: www.impactmapping.org

AGILES PROJEKTMANAGEMENT MIT SCRUM

Im vorliegenden Kapitel 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, bedeutet 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 im Vorfeld bereits detaillierte Anforderungen mit entsprechend genauen Arbeitsanweisungen für das gesamte Projekt zu leisten versucht, erhalten Scrum Teams ihre entsprechenden Zielvorgaben erst kurz vor der Implementierung und auch nur für das jeweilige Teilstück (Inkrement) der gesamt zu entwickelnden Software. Das hochqualifizierte und insgesamt interdisziplinäre Scrum Team bringt sich zudem in die Planung und konzeptionelle Weiterentwicklung der Software aktiv mit ein.

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 gruppendiffusiven Prozess im Team vorteilhaft zum Tragen kommt und zum anderen natürlich auch laufende Erkenntnisse permanent in die Arbeit einfließen.

Der Scrum-Ansatz: Zer teilung komplexer und umfangreicher Entwicklung in kleine Teilprojekte (Inkemente), die nacheinander in sog. Sprints (Iterationen), 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 dabei, dass der gesamte 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 für alle sichtbar festgehalten
- **ÜBERPRÜFEN:**
In regelmäßigen Abständen werden Produktfunktionalitäten geliefert und beurteilt.
- **ANPASSUNG:**
Die Anforderungen an das Produkt werden nicht ein für alle Mal im Vorfeld festgelegt, sondern nach jeder Lieferung eines Teilprojektes neu bewertet und bei Bedarf angepasst.

Im Jahre 2001 wurde von einigen IT-Vordenkern das mittlerweile berühmte „Agile Manifest“ erstellt, das auf den nachfolgenden zwölf Prinzipien basiert und die Grundlage für verschiedene agile Projektmanagementansätze bildet:

- › 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 Entwicklungsteams 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>



Das „Agile Manifest“ führt außerdem die u. s. Werte auf, von denen die jeweils links stehenden in der agilen Softwareentwicklung eine besonders wichtige, d. h. größere Rolle spielen als die rechts davon stehenden.

Quelle: <http://agilemanifesto.org>

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.



6.1. DIE SCRUM-ROLLEN

6.1.1. 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 als sog. User Stories in Abstimmung mit dem Entwicklungsteam im Product Backlog erfasst. Damit werden die gewünschten bzw. benötigten Funktionalitäten aus der Sicht des Users beschrieben.

Der Product Owner überprüft am Ende eines Sprints, ob die gelieferte Software die User Akzeptanz Kriterien erfüllt und verwendet werden kann. Entscheidungen des Product Owners sind verbindlich, denn er ist für das Endergebnis und die wirtschaftlichen Aspekte verantwortlich. Häufig wird der Product Owner vom Auftragnehmer gestellt und stellt die Brücke zwischen Kunde und dem restlichen Scrum Team dar.

6.1.2. SCRUM MASTER

Während der Product Owner für den Erfolg des Projekts zuständig ist, garantiert der Scrum Master für den Erfolg des Scrum-Prozesses. Dabei moderiert er 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. Der Scrum Master arbeitet eng mit dem Entwicklungsteam zusammen und übernimmt dabei ausschließlich administrative Aufgaben, ohne konkrete Arbeitsanweisungen geben zu dürfen. Dabei agiert der Scrum Master als „Coach“ innerhalb des Teams.

6.2. SCRUM MEETINGS

6.2.1. SPRINT PLANNING 1

Im sog. Sprint Planning 1 werden die Anforderungen dem Entwicklungsteam vom Product Owner vorgestellt. Einfach ausgedrückt erklärt der Product Owner dem Entwicklungsteam, was zu tun ist. Während dieses Meetings kann auch ein User, d. h. tatsächlicher Anwender der Software, dabei sein, der dem Team – zusammen mit dem Product Owner – 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 1 werden auch die Abnahmekriterien User Akzeptanz Kriterien für die einzelnen Stories definiert, die am Ende im sog. Sprint Review geprüft werden. Das Ziel eines jeden Sprints besteht in der Auslieferung gebrauchsfähiger und getester 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.

6.2.2. SPRINT PLANNING 2

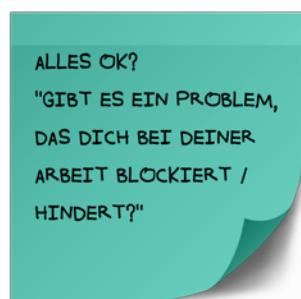
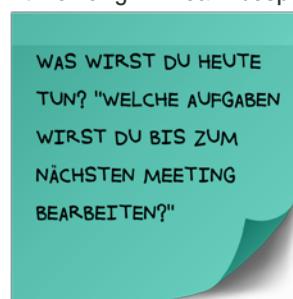
Im anschließenden Sprint Planning 2 klärt das Entwicklerteam dann eigenverantwortlich, wie die zuvor vorgestellten Anforderungen umgesetzt werden. Dabei werden die Anforderungen in sog. Tasks zerlegt, die normalerweise nicht länger als einen Tag dauern sollen. Die Tasks werden dann am sog. Taskboard angebracht,

wodurch ein sehr schneller Überblick über den aktuellen Sprint, die dafür anstehenden Aufgaben und den jeweiligen Status möglich wird. Auch für dieses Meeting sollten pro Sprint-Woche ca. 60 Minuten angesetzt werden.

6.2.3. 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. Konkret werden dabei die folgenden Fragestellungen im Team besprochen:



- › Sollte sich herausstellen, dass einzelne Tasks z. B. nicht innerhalb eines Tages erledigt werden können, können diese auch in kleinere Aufgaben herunter gebrochen werden. Sollten sich Fragen oder Probleme ergeben, die innerhalb der 15 Minuten nicht geklärt

werden können, ist es die Aufgabe des Scrum Masters, sich um diese Punkte zu kümmern. Diese Punkte werden vom Scrum Master dann im sog. Impediment Backlog erfasst und weiter bearbeitet. ›

6.2.4. 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 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 Erreichen des Ziels. Sollten bestimmte Anforderungen nicht ganz erfüllt sein, werden diese vom Product Owner als neue bzw. nochmalige User Story in das kommende Sprint Planning Meeting übernommen.

Gleiches gilt für neue Ideen oder Anforderungen, die sich während des Sprint Reviews ergeben.

6.2.5. RETROSPEKTIVE

Mit Abschluss des Sprint Reviews erfolgt ein ganz entscheidender Abschnitt im Rahmen eines Scrum-Projektes. Das Scrum-Team (Product Owner, Scrum Master, 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

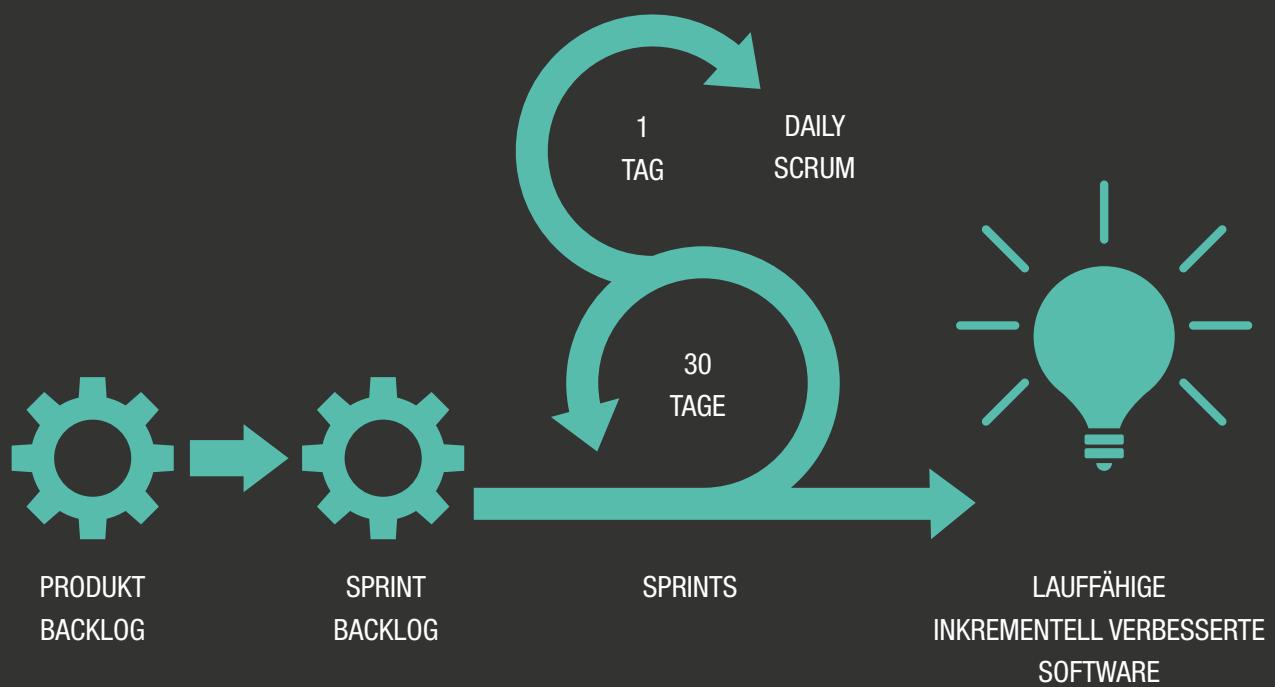
positive sowie negative Erkenntnisse werden notiert, mit dem Ziel, Verbesserungspotentiale für den neuen Sprint abzuleiten. Sofern es sich um Verbesserungen handelt, die das Team alleine betreffen, werden diese auch vom Team selbst gelöst. Etwaige andere Hemmnisse werden vom Scrum Master im sog. Impediment Backlog aufgenommen und vom ihm dann an den Product Owner zur Bewertung für den folgenden Sprint weitergegeben.

6.3. SPRINT

Im Rahmen eines Sprints, der eine Iteration von max. 4 Wochen darstellt, werden die im Sprint Planning besprochenen und am höchsten priorisierten Tasks (User Stories) erledigt. Während eines Sprints konzentriert sich das Entwicklungsteam also ausschließlich auf die in User Stories formulierten Anforderungen, auf die es sich im Sprint Planning committet hat.

Die Aufgabe des Scrum Masters besteht darin, etwaige Störfeuer oder Hemmnisse zu beseitigen, so dass sich das Team ausschließlich auf die Fertigstellung und Auslieferung funktionsfähiger Software konzentrieren kann. Der Scrum Master darf dabei keine Anweisungen an das Team erteilen, sondern unterstützt das Team lediglich dabei, das definierte Ziel zu erreichen.

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 kann der Kunde nach jedem Sprint entscheiden, ob er das Teilstück des Gesamtprojekts evtl. schon produktiv einsetzen oder aber mehrere Teilstücke in ein Release bündeln und zu einem späteren Zeitpunkt online stellen möchte – der Geschäftswert für den Kunden steht bei Scrum also stets im Vordergrund.



6.4. DIE SCRUM-ARTEFAKTE

6.4.1. SPRINT BACKLOG

Das Sprint Backlog ist eine Liste von Aufgaben, die vom Scrum Team definiert und während eines Sprints abgearbeitet wird. Die Aufgaben werden dabei täglich überarbeitet und aktualisiert. Durch das Sprint Backlog wird ein Projekt transparenter und es wird ein Überblick geschaffen, welches Team-Mitglied an welcher Aufgabe arbeitet und welche Tasks noch umgesetzt werden müssen. In diesem Zusammenhang wird ein Taskboard eingesetzt. Quelle: <https://borisgloger.com/scrum/>

Wikipedia erklärt den Sprint Backlog wie folgt:

„Das Sprint Backlog dient zur Übersicht der für einen Sprint zu erledigenden Aufgaben. Zu diesem Zweck kann ein Taskboard eingesetzt werden. Es besteht aus vier Spalten. In der ersten Spalte („Stories“)

werden die User Stories aufgehängt, für die sich das Entwicklungsteam zu diesem Sprint verpflichtet hat (in der vom Product Owner priorisierten Reihenfolge). Die drei weiteren Spalten enthalten die Aufgaben oder Tasks, die sich aus den einzelnen User Stories ergeben (und die im Sprint Planning 2 festgelegt worden sind). Je nach Bearbeitungsstand sind die Tasks entweder offen („Tasks to Do“), in Bearbeitung („Work in Progress“) oder erledigt („Done“). Im Daily Scrum erklärt jedes Mitglied des Entwicklungsteams anhand des Taskboards, an welcher Aufgabe es am Vortag gearbeitet hat, und ob diese erledigt wurde. Tasks, die an einem Tag nicht beendet werden konnten, werden mit einem roten Punkt markiert. So können Hindernisse schnell identifiziert werden.“

6.4.2. 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. Idealerweise erfolgt ein Eintrag in Form einer User Story als Antwort auf die Frage „Wer möchte was warum?“ nach folgendem Muster:

„Als User x möchte ich Funktionalität y, damit ich Nutzen z habe.“

Quelle: <https://borisgloger.com/scrum/>

6.4.3. IMPEDIMENT BACKLOG

Beim Impediment Backlog handelt es sich um eine Liste aller Blockaden, die einer effektiven, produktiven Arbeit des Teams im Weg stehen. Dieses Backlog wird bei den Scrum-Artefakten meist übersehen. Boris Gloger gibt in seinem Blog „Tipps zum richtigen Impediment Backlog“, die von einem Scrum Master beachtet werden sollten.

Quelle: <https://borisgloger.com/scrum/>

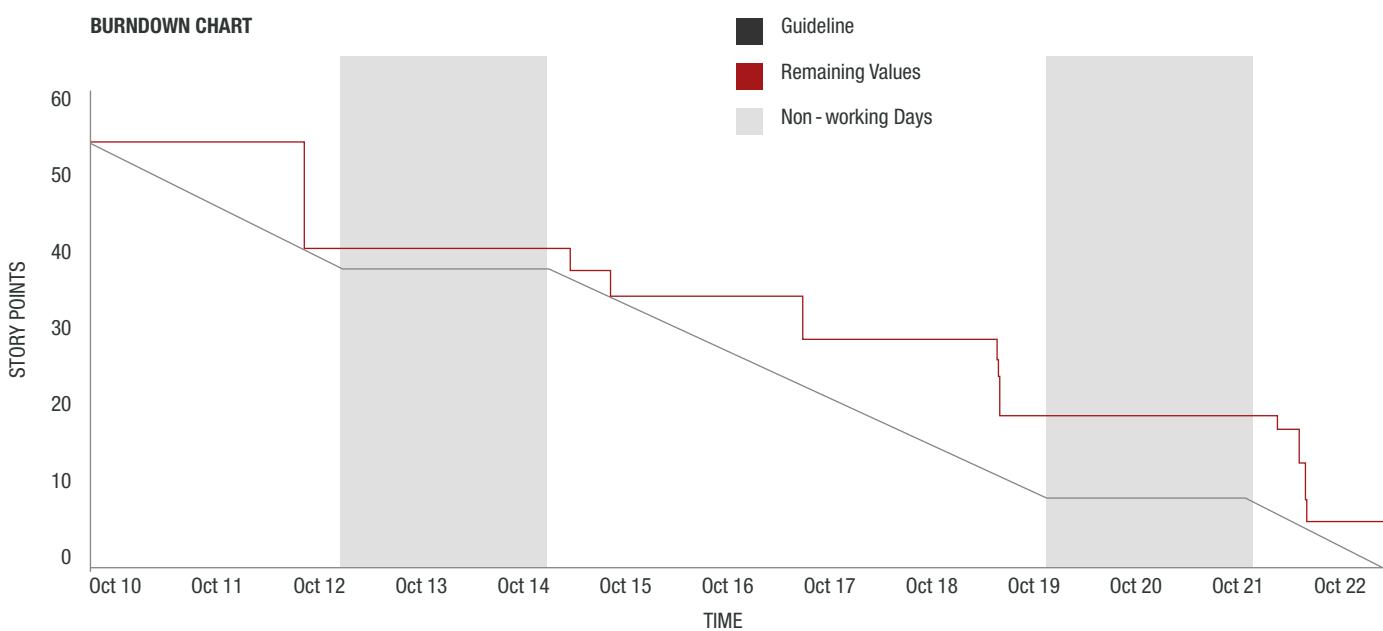
TIPPS ZUM RICHTIGEN IMPEDIMENT BACKLOG

1. Hänge es öffentlich auf. Am besten im Gang, so dass es alle sehen können.
2. Schreibe aktive Sätze. z. B. „Wir benötigen eine bessere Kaffeemaschine.“
3. Beim Formulieren der Sätze nicht beleidigen, aber auch nichts verschweigen.
4. Das Impediment Backlog sollte immer mindestens 10 Einträge vorweisen.
5. Denke daran, alle Bereiche der Verbesserung tatsächlich zu bearbeiten.
6. Mach es ordentlich: Es ist ein Ausdruck Eures Qualitätsbewusstseins, ob das Impediment Backlog ordentlich und gepflegt aussieht oder schlampig ist. Wir alle schließen von der Form auf den Inhalt: Unordentliche Charts = Schlampig programmierte Software.
7. Das Impediment Backlog muss sich verändern. Zeige durch Durchstreichen und Hinzufügen, dass auch DU als Scrum Master etwas tust.
8. Sage nicht: Der oder Die muss etwas tun, sondern DU als Scrum Master musst etwas tun.
9. Sprich das Backlog einmal in der Woche mit den anderen Scrum Mastern in deiner Organisation durch. Aktualisiere es mit den neuen Informationen.

6.4.4. 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 oder Storypoints eingetragen 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. es werden geeignete Gegenmaßnahmen ergriffen, um das Projekt wieder auf „Schiene“ zu bringen.



Quelle: Exemplarischer Burndown-Chart mit dem Projektmanagement-Tool JIRA / TechDivision

6.5. 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 Änderungen 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 das cross functional Team und den gruppodynamischen Prozess ergeben sich vielfach bessere Endergebnisse. Darüber hinaus arbeiten selbstorganisierte Teams effektiver.

Einer der größten Vorteile von Scrum liegt in der höchstmöglichen Transparenz. Alle Projektbeteiligten haben durch Zugriff auf Backlogs, auf Burndown-Diagramme mit dem Projektfortschritt, den angefallenen

Projektzeiten sowie etwaige Hemmnisse zu jedem Zeitpunkt vollständige Transparenz auf den aktuellen Entwicklungsstand. Hier kann jeweils sehr zeitnah gegengesteuert werden, wodurch böse Überraschungen ausbleiben.

Durch die permanente Diskussion in der Gruppe sowie die Reflexion 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.

6.6. DIE KOSTENKALKULATION IN SCRUM

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 Agenturleistungen sind dabei sehr preisgetrieben. Der Auftraggeber entscheidet sich daher möglicherweise für den vermeintlich günstigsten Anbieter. Machen wir uns hier aber nichts vor: Heutzutage hat niemand etwas

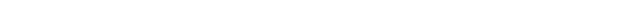
zu verschenken. Genauso wenig wie Sie Ihre Produkte oder Dienstleistungen für wenig Geld abgeben, wird dies eine Agentur tun. Insofern erlebt man es auch recht häufig, dass der vermeintlich günstigste Anbieter den Zuschlag bekommt, dieser in der Umsetzung dann aber eben aus Budgetgründen entweder nicht sauber arbeitet, ein halbfertiges Produkt abliefert 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!

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, sodass 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 und Involviering des Kunden in den Entwicklungsprozess hat dieser 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, dass das initiale Backlog mit auf hohem Abstraktionsniveau formulierten Anforderungen zusammen mit dem Kunden erstellt wird. Auf Basis der so erfassten Anforderungen kann eine erste Abschätzung der Aufwände erfolgen. Wir gehen dabei aufgrund der im Backlog erfassten Anforderungen von x Sprints á 2-4 Wochen mit x Personen aus. 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 wieder gehalten werden kann und das ohne – und dies ist sicherlich ganz entscheidend – Kompromisse bei der Qualität eingehen zu müssen!

6.7. DIE GRÖSSTEN FEHLER IM SCRUM-PROZESS

Innerhalb eines Scrum-Projekts zeigen Retrospektiven schnell auf, wie die Effizienz und damit Performance des Teams weiter gesteigert werden kann. Dabei besteht allerdings auch die Gefahr, dass eigene Modifikationen – sogenannte ScrumButs – an Scrum vorgenommen und wichtige Elemente ausgeblendet werden. Meist wurde dann die Quintessenz von Scrum nur bedingt verstanden.

BEISPIELE FÜR SCRUMBUTS SIND:

Wir nutzen Scrum, aber wir ...

... benötigen aufgrund unseres kleinen Teams keinen Scrum Master

... brauchen die Aufwandschätzung einzelner Aufgaben nicht

... verlängern einen Sprint, bis wir unser Ziel erreicht haben

... verzichten auf Retrospektiven

Quelle: <http://t3n.de/magazin/praxisbericht-scrum-kanban-scrumbuts-agiles-232822/>

SCUMBUTS =

an Scrum vorgenommene, eigene Modifikationen

TIPPS & TRICKS – SO VERMEIDEN SIE SCRUM-FEHLER

KEINE UMFASSENDE PLANUNG / VORBEREITUNG

Bei Scrum sind exzessive Vorbereitungsphasen nicht notwendig. Stattdessen sollte versucht werden, möglichst schnell mit der Implementierung zu starten und ein permanentes Feedback in den Sprint Reviews zur Weiterentwicklung und Verbesserung zu nutzen. Sogar das Product Backlog kann bei Bedarf nach Beginn des ersten Sprints erst noch erstellt werden.

KEINE VERSTEIFUNG AUF TOOLS, DIE DEN SCRUM PROZESS VEREINFACHEN

Häufig wird versucht, im Vorfeld (bevor man Scrum wirklich verinnerlicht hat) entsprechende Softwaretools zu finden, die den Scrum-Prozess abbilden und vereinfachen. Zu Beginn reichen hier jedoch Stifte und Papier, da der Start damit genauso möglich und häufig gerade zu Beginn einfacher und schneller abbildbar ist.

DAILY SCRUM – KURZ UND KNACKIG!

Das tägliche Scrum Meeting darf nicht dazu dienen, größere Probleme oder Schwierigkeiten zu diskutieren und hier Lösungsansätze zu erarbeiten. Stattdessen sollte das Meeting möglichst kurz und „knackig“ gehalten werden.

Problemdiskussionen sollten dann im Anschluss mit den relevanten Personen separat geführt werden. Der Scrum Master überwacht dabei die Diskussion und sorgt dafür, dass das Ganze möglichst zielgerichtet abläuft.

SELBSTSTÄNDIGES ARBEITEN

Scrum Teams arbeiten selbst organisierend und „besorgen“ sich die Aufgaben eigenständig. Eine Zuweisung von Tasks ist nicht erforderlich und meist auch kontraproduktiv.

SCRUM MASTER ALS MITWIRKENDER

Der Scrum Master kann als Coach im Scrum Team bezeichnet werden, der dafür sorgt, dass der Scrum-Prozess möglichst reibungslos funktioniert und das Team sich vollständig auf die Implementierung konzentrieren kann. Eine aktive Mitarbeit im Projekt ist hier nicht förderlich und sollte in jedem Fall vermieden werden. Gleicher gilt für technische Vorgaben, die der Scrum Master unbedingt unterlassen sollte.

PRODUCT OWNER ALS KOMPETENTER ANSPRECHPARTNER

Als verlängerter Arm des Kunden muss der Product Owner in den Entwicklungsprozess und den aktuellen Stand der Entwicklung immer eingebunden und für das Team bei Fragen oder Unklarheiten auch jederzeit greifbar sein. Zudem sorgt er für die permanente Kommunikation mit dem Kunden.

KEINE VORGABEN „VON AUßEN“

Ausschließlich das Team entscheidet, wie viele Tasks in einem Sprint abgearbeitet werden. Vorgaben „von außen“ sollten hier unbedingt vermieden werden, um bestmögliche Qualität gewährleisten zu können.

KEINE ALLEINGÄNGE INNERHALB DES TEAMS

Scrum lebt vom Teamgedanken und der gemeinsamen Arbeit an einem Projekt. Die Teammitglieder sollten daher auf Alleingänge jeglicher Art verzichten und stattdessen ein möglichst umfassendes Wir-Gefühl entwickeln. Das Team gewinnt und verliert gemeinsam!

KLARE VERANTWORTLICHKEITEN

Der Product Owner ist für die Spezifikation der Anforderungen zuständig. Er beschreibt und „übersetzt“, was am Ende erstellt werden muss. Die technische Umsetzung und die technologischen Ansätze obliegen jedoch einzig und allein dem Team.

KEINE UNTERBRECHUNGEN INNERHALB EINES SPRINTS

Sofern während des Sprints besonders dringende Themen auftauchen, sollte nach Möglichkeit versucht werden, diese nach dem Sprint anzugehen und den Sprint möglichst unverändert zu Ende zu führen. In besonders dringenden Fällen, sollte statt einer Änderung innerhalb des Sprints der komplette Sprint abgebrochen werden.

TEAMMITGLIEDER ENTSCHEIDEN NICHT SELBST ÜBER ANFORDERUNGEN

Der Product Owner ist als Produktverantwortlicher für die Ausgestaltung des Produktes und das Endergebnis verantwortlich. Er entscheidet auch alleine über Anforderungen und Besonderheiten. Teammitglieder müssen sich bei Fragen im Vorfeld mit dem PO abstimmen und dürfen hier nicht eigenständig Entscheidungen treffen, die über technische Ansätze hinausgehen.

ZIELORIENTIERTES VORGEHEN

Zu Beginn eines Sprints werden entsprechende Sprintziele definiert, die zwingend eingehalten und während des gesamten Sprints überwacht werden müssen. Anhand von sog. Burndown-Charts kann der Projektverlauf täglich mitverfolgt werden. Am Ende eines Sprints muss eine funktionsfähige und getestete Software ausgeliefert werden, bei der im Zweifelsfall lieber ein Feature weggelassen wird. Die fertig gestellten Funktionen müssen jedoch möglichst fehlerfrei funktionieren, so dass damit ein „Go Live“ jederzeit möglich wäre.

KOMPRIMIERTE TEAMS

Da bei Scrum das Team im Vordergrund steht und hier der Teamgedanke über allem schwebt, sollte das Team auch möglichst in einem Raum sitzen um die Kommunikation und das Wir-Gefühl optimal zu unterstützen. Verteilte Scrum-Teams funktionieren zwar grundsätzlich auch, am meisten kann jedoch von komprimierten Teams profitiert werden.

ÄNDERUNGEN DES SCRUM TEAMS VERMEIDEN

Wie bereits mehrfach erwähnt, entfaltet Scrum seine Stärken im Team und mit laufender Zusammenarbeit der Team-Mitglieder. Insofern sollte während eines Scrum-Projektes eine Änderung am Scrum Team unbedingt vermieden werden. Insbesondere durch Kontinuität kann sich ein Scrum Team laufend verbessern und die Arbeit im Team permanent optimieren.

AUGENMERK AUF QUALITÄT

Eigenverantwortlichkeit sowie das Thema „Auslieferbare Software“ steht bei Scrum im Vordergrund. Dies bedeutet nichts anderes, als dass die gelieferten Ergebnisse während des Sprints entsprechend getestet und optimiert werden um ein bestmögliches und finales Ergebnis abzuliefern. Dies muss vom Team im Rahmen des Sprint-Plannings berücksichtigt werden, damit die implementierten Features auch noch ausreichend getestet werden können. Separate Test-Teams sind bei Scrum nicht vorgesehen, da die Ergebnisse dies nicht erforderlich machen.

6.8. ÜBERSICHT DER VOR- UND NACHTEILE VON SCRUM

VORTEILE

- › Höchstmögliche Flexibilität durch jederzeitige Änderung und Priorisierung von Anforderungen
- › Frühzeitige Ergebnisse & funktionsfähige Teilprojekte
- › Verbesserung in der Time to Market
- › Erhöhung der Kommunikation im Team und dadurch Verbesserungspotentiale
- › Größtmögliche Transparenz für alle Projektbeteiligten
- › Realistische und genaue Aufwandsschätzungen im Team
- › Frühzeitiges Erkennen von Problemen mit entsprechenden Handlungsmöglichkeiten
- › Verbesserung der Qualität durch konsequentes Teamwork
- › Fixe Termine

NACHTEILE

- › Erhöhter Kommunikations- und Abstimmungsaufwand
- › Ggf. Unklarheiten bei Zuständigkeiten von interdisziplinären Themen
- › Möglichkeiten zur Verzettelung im Detail bei unerfahrenen Teams
- › Verschlechterung der Effizienz durch fehlende Beseitigung bekannter Probleme aus den Reviews
- › Ggf. müssen Unternehmensstrukturen geändert werden, damit das Team selbstorganisiert arbeiten kann und die notwendige Unterstützung erhält.

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.

Die Grundlagen von Scrum und Kanban sind zwar einfach zu verstehen, eine disziplinierte Umsetzung ist jedoch meist nicht so einfach zu erzielen. In diesem Zusammenhang lohnt es sich meist, auf Scrum-Experten – bei TechDivision arbeiten u. a. vier Certified Scrum Professionals – zurückzugreifen, die Team und Kunden schulen und so eine reibungslose Projektabwicklung garantieren. Scrum ist dabei weitaus mehr als „nur“ ein Framework für Software- bzw. Produktentwicklung. Scrum ist vielmehr ein Paradigmenwechsel, der sowohl vom Kunden als auch vom Team ein Umdenken voraussetzt.

6.9. ZUSAMMENFASSUNG: SCRUM

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: <https://www.energizedwork.com/weblog/2007/05/fixed-price-contracts-dont-work>

Das nachfolgende Video zeigt in knapp 5 Minuten nochmals die wesentlichen Elemente sowie die Vorgehensweise bei Scrum.



„WAS IST SCRUM?“ VON BORIS GLOGER

5:36

> borisgloger consulting GmbH

[Watch the Video](#)

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 zu Buche 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 bestmöglich er und funktionierender Software, die die definierten Anforderungen erfüllen muss. Bei der Umsetzung bestehen hier mehr Freiheiten – was jedoch keinesfalls mit mangelnder Qualität gleichzusetzen ist.

AGILES PROJEKTMANAGEMENT MIT KANBAN

„Kanban ist eine Methode der Produktionsprozesssteuerung.“

Das Vorgehen orientiert sich ausschließlich am tatsächlichen Verbrauch von Materialien am Bereitstellungs- und Verbrauchszeitpunkt. Kanban ermöglicht ein Reduzieren der lokalen Bestände von Vorprodukten in und nahe der Produktion, die dort in Produkten der nächsten Integrationsstufe verbaut werden.“

Quelle: <http://de.wikipedia.org/wiki/Kanban>

7.1. ZUR GESCHICHTE VON KANBAN

Das Wort Kanban stammt aus dem Japanischen und bedeutet so viel wie „Karte“, „Tafel“ oder „Beleg“. Das Prinzip ist auch unter Hol-, Zurufprinzip oder Pull-Prinzip bekannt. Das ursprüngliche Kanban wurde bereits 1947 von Taichi Ohno entwickelt, der mit dem System auf die mangelnde Produktivität bei Toyota Motor Corporation im Vergleich zur amerikanischen Konkurrenz reagierte. Seine Idee war es dabei, die Produktion nach dem Supermarkt-Prinzip zu gestalten und zu organisieren: Sobald ein Verbraucher etwas aus dem Regal nimmt, wird dieses sofort wieder aufgefüllt.

Das Kanban-Prinzip ist auch als eine Reaktion auf die Bedürfnisse der Kunden zu sehen, die neue Anforderungen an die Produktionsgeschwindigkeit, Lieferbereitschaft und Zuliefererbeziehungen haben. Das Kanban-Verfahren ersetzte dabei bisherige Produktionsverfahren und wurde von zahlreichen japanischen Unternehmen, darunter auch Toyota, eingeführt. In den 70er Jahren wurde das System auch in Deutschland und in den USA adaptiert.
vgl. [http://de.wikipedia.org/wiki/Kanban#Historische Entwicklung](http://de.wikipedia.org/wiki/Kanban#Historische_Entwicklung)

7.2. KANBAN IN DER SOFTWAREENTWICKLUNG

David J. Andersson überarbeitete die Grundidee von Kanban und passte diese vor allem hinsichtlich der Bedürfnisse in der Softwareentwicklung an. Ergebnis ist ein evolutionäres Vorgehen, bei dem kontinuierlich Arbeitsweisen verbessert und optimiert werden. In diesem Zusammenhang wird dabei immer von der gegenwärtigen Situation ausgegangen. Einen Kanban-Sollzustand gibt es dabei nicht. Kanban-Teams halten sich vielmehr an folgende grundlegende Prinzipien:

- › Visualisierung des Arbeitsflusses und der Arbeit
- › Limitierung des WIP (WIP = Work In Progress, in Ausführung befindliche Arbeit)
- › Steuerung und Messung des Arbeitsflusses
- › Prozessregeln explizit machen
- › Verbesserung durch bewährte Modelle und wissenschaftliche Methoden

Quelle: www.heise.de/developer/artikel/Software-Kanban-im-Einsatz-1235465.html

ZIEL IST ES DURCH KANBAN MECHANISMEN IM SYSTEM ZU IMPLEMENTIEREN, DIE LAUFENDE VERBESSERUNGEN UND VERÄNDERUNGEN ERLAUBEN. IM LAUFE DES PROZESSES KÖNNEN SICH SO DIE TEAMMITGLIEDER EINBRINGEN UND OPTIMIERUNGEN DES WORKFLOWS VORNEHMEN. DIESBEZÜGLICH MUSS ERST EINMAL DER KAIZEN-GEDanke IN DIE KÖPFE DER MENSCHEN GEBRACHT WERDEN. DABEI MÜSSEN VERÄNDERUNGEN NICHT AM MENSCHEN, SONDERN DURCH DEN MENSCHEN PASSIEREN.

7.3. KANBAN IN PROJEKTEN

Kanban und Scrum finden ihren Einsatz in unterschiedlichen Projekten. Bei Kanban handelt es sich im Gegensatz zu Scrum um eine Vorgehensweise, die einen kontinuierlichen Arbeitsfluss garantieren soll. In diesem Zusammenhang werden alle anstehenden Aufgaben und Abläufe mit Hilfe eines Boards, das in Zeilen und Spalten aufgeteilt ist, visualisiert. Auf dem Board finden sich die Aufgaben in Form von Tickets wider. Jede Spalte repräsentiert dabei einen Arbeitsschritt und das Projektteam kann jederzeit den Status einer Aufgabe einsehen. In diesem Zusammenhang gibt es ein Limit an parallel laufenden Aufgaben. Ziel dabei ist es, das Team nicht mit unterschiedlichen Aufgaben zu überhäufen, sondern ein konzentriertes Arbeiten an einem Task zu ermöglichen

(„Limit work in progress“). So ist ein gleichmäßiges Bearbeiten von Tickets ohne lange Wartezeiten oder Blockaden gewährleistet. Ein zentraler Aspekt ist dabei der Flow-Gedanke. Kanban visualisiert somit den aktuellen Prozess, ändert diesen aber zunächst nicht.

Kanban gilt als eine agile Methode, um Change Management evolutionär durchführen zu können. Dabei werden bestehende Prozesse schrittweise optimiert und verbessert. Durch die Änderung kleinerer Aspekte wird das Risiko für jede einzelne Maßnahme verringert. Ein Vorteil von Kanban ist auch der geringere Widerstand bei den Beteiligten. (vgl. <http://www.it-agile.de/wissen/methoden/kanban/>)

7.4. SO FUNKTIONIERT KANBAN

Eine wichtige Aufgabe von Kanban ist es, vorhandene Probleme und Arbeiten darzustellen und zu visualisieren. In diesem Zusammenhang spielt das sogenannte Kanban-Board eine wichtige Rolle, welches beispielsweise aus einem Whiteboard, Karteikarten oder Haftnotizen bestehen kann. Jede Aufgabe wird dabei durch eine Karte, etc., präsentiert. Der Vorteil dieser Vorgehensweise ist eine erhöhte Transparenz bei der Bearbeitung von Projekten.

Quelle: <https://management30.com/about/>

Quelle: <https://www.it-agile.de/wissen/einstieg-und-ueberblick/kanban/>

Kanban orientiert sich dabei an Lean- Prinzipien und basiert auf dem Pull-Prinzip (Hol-Prinzip), bei dem die anfallende Arbeit nicht von einem Vorgesetzten verteilt wird, sondern die Arbeiter bzw. Teammitglieder sich selbstständig ihre Arbeit holen. Diese wird auf sog. Kanban-Karten am Kanban-Board hinterlegt.

FAZIT

Die Praxis hat gezeigt, dass eine komplexe Welt, in der wir uns heute bewegen, nur noch über agile Methoden erfolgreich gemanaged werden kann. Diverse Studien belegen diese These auch recht eindrucksvoll.

Ob dies nun über Scrum, Kanban oder „exotischere“ bzw. eigens adaptierte Modelle erfolgt, hängt von unterschiedlichsten Faktoren ab und muss jedes Unternehmen anhand der jeweiligen Gegebenheiten für sich selbst herausfinden.

Was beim Thema Agilisierung jedoch niemals vergessen werden darf, sind die Mitarbeiter, die möglichst frühzeitig in den Prozess involviert und an die Hand genommen werden müssen. Eine Agilisierung und die damit verbundenen Vorteile im Bezug auf Flexibilität und Zeit funktionieren nur, wenn Entscheidungen auch dort kurzfristig getroffen werden können, wo sie gerade anstehen. Führungskräfte müssen daher lernen, los zu lassen, Verantwortung zu übertragen und etwaige Konsequenzen auch mitzugehen. Kaum ein Mitarbeiter wird bewusst falsche Entscheidungen treffen und die Praxis hat ja auch mehr als häufig gezeigt, dass selbst die erfahrensten und „besten“ Manager nicht vor Fehlentscheidungen gefeit sind....

Die Ideen hinter Management 3.0 dienen dazu, ein für agile Arbeits- und Managementmethoden passenderes Umfeld sowie eine zeitgemäßere Unternehmenskultur zu etablieren, über die Unternehmen auch ihre Attraktivität als Arbeitgeber (wieder) steigern können.

Wir freuen uns in jedem Fall jederzeit über Feedback, Fragen und Anregungen, aber auch über Erfahrungen und Learnings anderer Unternehmen.



AUTOREN

JOSEF WILLKOMMER, GESCHÄFTSFÜHRER

TECHDIVISION

Als Geschäftsführer von TechDivision beschäftigt sich Josef Willkommer seit vielen Jahren sehr intensiv mit Themen aus den Bereichen E-Commerce, Online Marketing und den dazu notwendigen, modernen Projektmanagement-Ansätzen. Darüber ist er als Chef-Redakteur des eStrategy-Magazins – einem quartalsweise erscheinenden, kostenlosen Online-Magazin mit Fokus auf E-Commerce und Online Marketing – auch journalistisch tätig und versucht sein Wissen in Form von Fachbeiträgen weiterzugeben. Auch auf diversen Fachkonferenzen trifft man ihn als Referent zu Themen rund um den elektronischen Handel.

SACHA STORZ, STANDORTLEITUNG MÜNCHEN / BUSINESS DEVELOPMENT

Sacha Storz ist ausgebildeter Psychologe und als Agile Evangelist bei TechDivision für die Einführung und Optimierung moderne Managementmethoden in der täglichen Praxis verantwortlich. Er verfügt hierzu über folgende Ausbildungen und Zertifizierungen:

- Certified Scrum Product Owner (CSPO)
- Certified Scrum Professional (CSP)
- Professional Scrum Master Level I
- Kanban Management Professional
- Certified Agile Requirements Specialist (cars)
- Licensed Management 3.0 Facilitator

Sacha Storz ist als Speaker auf unterschiedlichsten auch internationalen Veranstaltungen rund um Projektmanagement und New Work anzutreffen. Darüberhinaus gibt es Schulungen und Trainings zu diesen Themen.

DOMINIK HALLER, ONLINE MARKETING MANAGER / PROJEKTMANAGER

Der studierte Kommunikationswissenschaftler arbeitet als Online Marketing Manager bei der TechDivision GmbH und ist für das Content Management,

Partnermanagement, für die Eventorganisation sowie sämtliche Marketingaktivitäten online / offline zuständig. Darüber hinaus ist er als leitender Redakteur und Autor des eStrategy Magazins für Hintergrundrecherchen rund um das Thema E-Commerce, Online Marketing, E-Recht, etc. zuständig. Neben seiner beruflichen Tätigkeit bei der TechDivision GmbH engagiert er sich auch als Lehrbeauftragter an der Universität Salzburg.

EXTERNE AUTOREN

DR. MATTHIAS ORTHWEIN, LL.M. (BOSTON) / SKW SCHWARZ RECHTSANWÄLTE

Dr. Matthias Orthwein, LL.M. berät nationale und internationale Mandanten im IT-Recht, vor allem im Softwarerecht bspw. bei Lizenzrechtsauseinandersetzungen, beim Entwurf und der Verhandlung von Softwareerstellungs- und -kaufverträgen.

Als erfahrener Experte im IT Outsourcingrecht und im IT Projektrecht verhilft er Providern und Kunden zu einem erfolgreichen Projektabschluss und ist sowohl bei der Ausschreibung und Verhandlung von Verträgen als auch bei der Vertragsdurchführung und bei der Lösung eventueller Konflikte zwischen den Parteien tätig.

Der Schwerpunkt seiner Tätigkeit wird zudem durch die Beratung im nationalen und internationalen Datenschutzrecht sowie im e-Commerce abgerundet. Dr. Orthwein ist Lehrbeauftragter für IT und Datenschutzrecht an der Hochschule Rosenheim. Der Branchendienst „Legal 500 Germany“ zeichnet ihn auch 2017 wieder als „Häufig empfohlenen Anwalt für IT und Outsourcing“ aus.

TECHDIVISION GMBH

TURNING ONLINE PROJECTS INTO SUCCESS

25



EIN BLICK HINTER DIE KULISSEN

02:16

> TechDivision

Watch the Video

Der Webtechnologiedienstleister TechDivision gehört im deutschsprachigen Raum zu den führenden Adressen für anspruchsvolle eCommerce-Lösungen und Digitalisierung von Geschäftsprozessen auf Basis von Open Source Technologien. Das Leistungsspektrum von TechDivision reicht von Consultingleistungen über Konzept- und Designentwicklung sowie Implementierung und Betreuung bis hin zu Online-Marketing. Neben diversen mittelständischen Kunden vertrauen auch international agierende Unternehmen wie Allianz, Ritter-Sport, ZORO Tools, Salewa, FERRERO oder Cherry auf das Know-how und die Erfahrung von TechDivision. Aktuell verfügt TechDivision über zwei Standorte in Rosenheim/Kolbermoor und München und beschäftigt insgesamt mehr als 75 Mitarbeiter.

Mit dem eStrategy-Magazin veröffentlicht TechDivision zudem einmal im Quartal das führende Online-Magazin zum Thema eCommerce und Online-Marketing in der DACH-Region. Das Magazin kann kostenlos unter www.estrategy-magazin.de herunter geladen werden.

Mehr Informationen über TechDivision lesen Sie unter www.techdivision.com.



26

WAS WIR FÜR SIE TUN KÖNNEN...

Sie brauchen Unterstützung bei Ihrer Digitalen Transformation?

TechDivision verfügt sowohl im Bereich Technologie als auch

Personal/HR über fundierte und langjährige Expertise!



LEISTUNGSSPEKTRUM

TECHNOLOGIE

Entwicklung von individuellen Webapplikationen auf Basis modernster Open Source Technologie. Dadurch können wir größtmögliche Flexibilität, Wirtschaftlichkeit und Zukunftsfähigkeit gewährleisten:

- eCommerce Plattformen (b2c & b2b) einschließlich Omnichannel-Lösungen
- Individuelle Webapplikationen inkl. Single Page Apps
- Corporate Web, Intranet, Extranet
- Technologie-Audits und Evaluierungen

PERSONAL UND PROZESSE

- Workshops und Trainings zu agilen Managementmethoden (Scrum, Kanban, Management 3.0)
- Vorträge zum Thema Digitale- & Agile Transformation
- Consulting und Begleitung bei entsprechenden Change-Prozessen

DAS SAGEN KUNDEN ÜBER UNS:

27



„Der Workshop mit Stefan Willkommer (Geschäftsführer) und Sacha Storz (Management 3.0 Trainer / Agile Evangelist) der TechDivision war interessant und inspirierend. Unser Austausch über agiles Management und agile Transformation hat uns zum einen einen spannenden Einblick in die dynamische Organisation einer 70-Personen-Agentur gegeben. Die Feedbacks, Anregungen und kritischen Fragen der beiden wiederum haben dazu geführt, Aspekte unserer Change-Initiative, Organisationsstruktur und Unternehmenskultur neu zu betrachten. Ein sehr lohnenswerter Termin, der mit Sicherheit Folgetermine nach sich ziehen wird!“

Margerete Voll, Projektleiterin Allianz in Führung



„Wir sind sehr froh bei unserem digitalen Transformationsprozess einen erfahrenen Partner wie TechDivision an unserer Seite zu haben, der im Vorfeld bei der Auswahl entsprechender Technologien und Prozesse beratend zur Seite stand und uns in der Folge auch bei der Umsetzung und Weiterentwicklung unterstützt! An dieser Stelle schon mal vielen Dank für die bisher geleistete, sehr gute Arbeit!“

Michael Schmid, Leitung Marketing & PR, Cherry GmbH

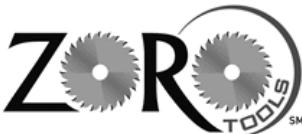


„Wie viele mittelständische Unternehmen beschäftigen uns auch wir inzwischen intensiver mit der fortschreitenden Digitalisierung. Bei der Unmenge an Möglichkeiten und Optionen war bzw. ist es für uns jedoch nicht einfach, die richtigen Ansätze zu finden. Hierbei begleitet uns TechDivision bereits seit geraumer Zeit sehr erfolgreich. An dieser Stelle vielen Dank für ihre bisherige Arbeit und ihr Engagement. Wir sind gespannt, wie sich unsere Ansätze weiterhin entwickeln.“

Holger Wagner. Leitung Marketing, Erich Jaeger GmbH + Co. KG

EIN AUSZUG UNSERER REFERENZEN

28

	EUROTOURS INTERNATIONAL	beyerdynamic	
CHERRY ♀	Allianz III		SapientNitro ™
	FERRERO		FLASCHENPOST <i>Online zu jedem Wein</i>
	RÖHM driven by technology		
KATHREIN Antennen · Electronic			
			

Alle abgebildeten Logos und genannten Warenzeichen sowie Markennamen sind Eigentum der entsprechenden Unternehmen.

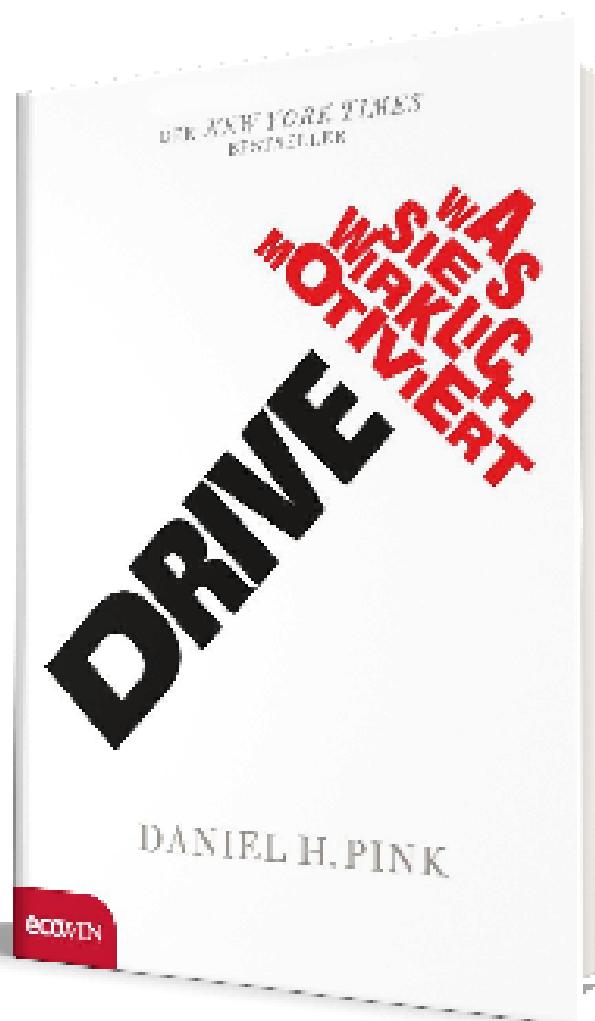
29

DAMIT NEHMEN SIE FAHRT AUF....

„DRIVE“ VON DANIEL PINK

Sind wir nicht alle davon überzeugt, dass wir am besten mit äußereren Anreizen wie Geld und Prestige oder durch „Zuckerbrot und Peitsche“ zu motivieren sind? „Alles falsch“, sagt Daniel H. Pink in seinem provokanten und zukunftsweisenden Buch. Das Geheimnis unseres persönlichen Erfolges ist das zutiefst menschliche Bedürfnis, unser Leben selbst zu bestimmen, zu lernen, Neues zu erschaffen und damit unsere Lebensqualität und unsere Welt zu verbessern. Daniel H. Pink enthüllt die Widersprüche zwischen dem, was die Wissenschaft weiß, und dem, was die Wirtschaft tut – und wie genau dies jeden Aspekt unseres Lebens beeinflusst. Er demonstriert, dass das Prinzip von Bestrafung und Belohnung exakt der falsche Weg ist, um Menschen für die Herausforderungen von Heute zu motivieren, egal ob in Beruf oder Privatebenen. In „Drive“ untersucht er die drei Elemente der wirklichen Motivation – Selbstbestimmung, Perfektionierung und Sinnerfüllung – und bietet kluge sowie überraschende Techniken an, um diese in die Tat umzusetzen. Daniel H. Pink stellt uns Menschen vor, die diese neuen Ansätze zur Motivation bereits erfolgreich in ihr Leben integriert haben und uns damit entschlossen einen außergewöhnlichen Weg in die Zukunft zeigen.

Jetzt bei Amazon bestellen



THE SURPRISING TRUTH ABOUT WHAT MOTIVATES US

10:47

> Daniel Pink

Watch the Video

LESETIPPS

- Semler, Ricardo:
„MAVERICK: THE SUCCESS STORY BEHIND THE WORLD'S MOST UNUSUAL WORKPLACE“
[Jetzt bei Amazon bestellen](#)
- Pink, Daniel H.:
„DRIVE: THE SURPRISING TRUTH ABOUT WHAT MOTIVATES US“
[Jetzt bei Amazon bestellen](#)
- Appelo Jurgen:
„MANAGEMENT 3.0 - LEADING AGILE DEVELOPERS, DEVELOPING AGILE LEADERS“
[Jetzt bei Amazon bestellen](#)
- Hsieh Tony:
„DELIVERING HAPPINESS: A PATH TO PROFITS, PASSION AND PURPOSE“
[Jetzt bei Amazon bestellen](#)
- Ries Eric:
„THE LEAN STARTUP: HOW CONSTANT INNOVATION CREATES RADICALLY SUCCESSFUL BUSINESSES“
[Jetzt bei Amazon bestellen](#)
- Hanisch Roland:
„DAS ENDE DES PROJEKTMANAGEMENTS“
[Jetzt bei Amazon bestellen](#)
- Soll Donald, Eisenhard Kathleen M.:
„SIMPLE RULES - EINFACHE REGELN FÜR KOMPLEXE SITUATIONEN“
[Jetzt bei Amazon bestellen](#)
- IT-Agile:
„AGILE MANAGEMENT INNOVATIONS – A PRIMER“
[Jetzt PDF downloaden](#)
- Valve, 2012:
„HANDBOOK FOR NEW EMPLOYEES“ (PDF)
[Jetzt PDF downloaden](#)
- Wagepoint:
„THE EPIC GUIDE TO EMPLOYEE MANAGEMENT“
[Jetzt zu wagepoint.com gehen](#)

IMPRESSIONUM

TechDivision GmbH

Spinnereiinsel 3a

83059 Kolbermoor

Tel. +49 8031 / 221055-0

Fax +49 8031 / 221055-22

info@techdivision.com



HERAUSGEBER:

TechDivision GmbH

QUELLENVERZEICHNIS:

- <http://de.wikipedia.org/wiki/Kanban>
- http://de.wikipedia.org/wiki/Kanban#Historische_Entwicklung
- www.heise.de/developer/artikel/Software-Kanban-im-Einsatz-1235465.html
- <http://www.it-agile.de/wissen/methoden/kanban/>
- www.heise.de/developer/artikel/Kanban-richtig-einfuehren-1344554.html?artikelseite=2
- www.heise.de/developer/artikel/Kanban-richtig-einfuehren-1344554.html?artikelseite=3
- www.t3n.de/magazin/praxisbericht-scrum-kanban-scrumbuts-agiles-232822/2/
- www.infoq.com/resource/news/2010/01/kanban-scrum-minibook/en/resources/KanbanAndScrum-German.pdf
- <https://jaxenter.de/agile-im-konzern-kann-das-funktionieren-41026>
- https://www.gpm-ipma.de/fileadmin/user_upload/Know-How/studien/Studie_Agiles-PM_web.pdf
- <https://management30.com/>
- Douglas McGregor, „The Human Side of Enterprise“, 1960
- <http://agiletrail.com/2012/11/08/agile-management-innovations-a-primer>
- Rocketboy by RichVintage, istockphotos.com, S.11.
- Josef Willkommer by Florian Sydekum, TechDivision GmbH, S.19
- Superboy by Sandra Hascher, TechDivision GmbH, S.23
- Spinnereiinsel by Wilhelm Bauer, TechDivision GmbH, S.35

„TURNING ONLINE PROJECTS INTO SUCCESS“

Sie haben Fragen? Wir stehen Ihnen telefonisch und per Mail gerne zur Verfügung und freuen uns auf eine gemeinsame und erfolgreiche Zusammenarbeit!

TechDivision GmbH

Spinnereiinsel 3a
83059 Kolbermoor

Balanstr. 73, Haus 8, 3 OG
81541 München

Tel +49 8031 2210 55-0
Fax +49 8031 2210 55-22

info@techdivision.com
www.techdivision.com