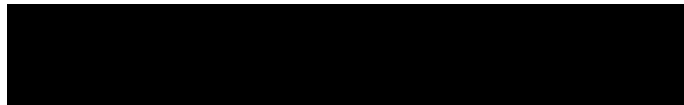


# Bachelorthesis

Vor- und Zuname  
Konstantin Lenz



Titel:  
„Management von agilen Projekten am Beispiel von Scrum“

Abgabedatum:  
27.03.2019

Betreuender Professor: Herr Prof. Dr. Werner Röhrs

Zweiter Prüfender: Herr Prof. Dr. Ralf Lenschow

Fakultät Wirtschaft und Soziales

Department Wirtschaft

**Studiengang:**

Marketing/Technische Betriebswirtschaftslehre

## Inhaltsverzeichnis

Inhaltsverzeichnis .....	2
Abbildungsverzeichnis .....	3
1 Einleitung.....	4
1.1 Hintergrundinformationen .....	4
1.2 Forschungsfrage und Ziel .....	5
1.3 Vorgehensweise .....	6
1.4 Kapitelbeschreibung .....	6
2 Klassisches und agiles Projektmanagement .....	7
2.1 Definitionen .....	7
2.2 Klassisches Projektmanagement.....	8
2.3 Agiles Projektmanagement .....	9
2.4 Scrum .....	11
2.4.1 Scrum-Entstehung .....	11
2.4.2 Was ist Scrum? .....	12
2.4.3 Scrum-Rollen.....	13
2.4.4 Scrum-Artefakte .....	17
2.4.5 Scrum-Ereignisse.....	18
2.4.6 Projektmanagement mit Scrum .....	21
3 Scrum in der Praxis.....	23
3.1 Scrum bei TablonautiX.....	23
3.2 Scrum bei Intel Corporation.....	26
3.3 Scrum bei Kaarten Carrousel .....	31
3.4 Scrum bei Swiss Property AG .....	35
4 Gesamtergebnisse .....	39
5 Fazit .....	42
Literaturverzeichnis .....	44

## Abbildungsverzeichnis

Abb. 1 Das Wasserfallmodell. In Anlehnung an Timinger 2017.....	9
Abb. 2 Der Sprint. In Anlehnung an Pichler 2009 .....	20
Abb. 3 Der Scrum-Flow. In Anlehnung an Pichler 2009. ....	22

# 1 Einleitung

## 1.1 Hintergrundinformationen

Unsere Zeit ist geprägt durch das Phänomen der immer komplexer werdenden Umwelt, in der von fast allen Unternehmen eine hohe Flexibilität und ein sehr gutes Reaktionsvermögen abverlangt werden.<sup>1</sup> Künstliche Intelligenz, lernende Systeme, hohe Vernetzung und Digitalisierung eröffnen den Unternehmen völlig neue Chancen. Produkte können günstiger und individualisierter angeboten, Prozesse optimiert und neue Märkte erschlossen werden. Doch mit der steigenden Zahl der vernetzten Märkte und Marktteilnehmer entstehen auch unvorhersehbare Abhängigkeiten, plötzliche Schwankungen und unberechenbare Risiken. Wie schnell und auf welche Art ein Unternehmen auf eine neue Situation reagiert, kann sich oft zu einer Überlebensfrage entwickeln. Projekte sind ein beliebtes Instrument, um auf Veränderungen zu reagieren und Ziele zu erreichen. Durch die Besetzung von Projektteams mit sorgfältig ausgewählten Experten sowie Bereitstellung weiterer benötigter Ressourcen, stellen die Projekte die beste Organisationsform zur Lösung komplexer Aufgaben innerhalb des vorab definierten Zeitintervalls dar.<sup>2</sup>

Die hohe Relevanz der Projekte und des Projektmanagements für die Unternehmenspraxis besteht schon länger. Die Ursprünge des „modernen“ Projektmanagements lassen sich bis in die 1950er Jahre rückverfolgen. Die Entwicklung der Rahmenbedingungen für Projekte wurde mit der Zeit immer dynamischer und begünstigte die Adoption neuer Ansätze zum Steuern von Projekten. So lagen dem klassischen Projektmanagement die Annahmen über relativ stabile Umgebung und Anforderungen zugrunde. Eine sehr detaillierte Planung von Anfang bis Ende stand im Fokus. Mit zunehmender Komplexität und Dynamik der Rahmenbedingungen wurde eine flexible und pragmatische Herangehensweise notwendig. Die existierenden, starren und plangetriebenen Methoden allein konnten den neuen Anforderungen nicht mehr gerecht werden. Also wurden sie durch agile Vorgehensweisen ergänzt oder substituiert.<sup>3</sup>

Von den vielfältigen agilen Methoden, die es inzwischen gibt, ist Scrum einer der am weitesten verbreiteten Ansätze.<sup>4</sup> Seinen Ursprung findet Scrum in der Softwareentwicklung.

---

<sup>1</sup> Vgl. Sharma, 2012, S. 2.

<sup>2</sup> Vgl. Gehr et al. 2018, S. 1.

<sup>3</sup> Vgl. Hagen: „Evolutionsphasen im Projektmanagement“, 2008 (online).

<sup>4</sup> Vgl. Kusay-Merkle, 2018, S. 18.

Nach dem sich das Internet in den 90er Jahren mit rasanter Geschwindigkeit entwickelte und ganz neue Möglichkeiten im IT-Bereich eröffnete, nahm auch die Komplexität der Projekte dort sehr stark zu.<sup>5</sup> Auf der Suche nach Möglichkeiten innerhalb der veränderten Umgebung angepasst arbeiten zu können, probierten die Softwareingenieure unter anderem auch agile Vorgehensweisen aus. Bei der Umsetzung neuer Methoden in die Praxis, hat sich Scrum am weitesten verbreitet und ist heute der De-facto-Standard in bestimmten Bereichen in der agilen Softwareentwicklung.<sup>6</sup>

Der Trend zu agilen Methoden beschränkt sich längst nicht nur auf IT-Bereiche. Die ansteigende Komplexität im Umfeld der Unternehmen nimmt globale Züge an und erfasst immer mehr Branchen. Bei vielen Unternehmen steht die Umstellung auf agile Arbeitsweise erst am Anfang oder läuft nur langsam an.<sup>7</sup> Ohne praktisches Wissen kann niemand einschätzen, was bei der Einführung gut gelingen und was sich als problematisch herausstellen wird. Selbst die Fachliteratur beschränkt sich größtenteils auf die Vermittlung theoretischer Kenntnisse, ohne dabei auf Praxisbeispiele zurückzugreifen. Allerdings existieren bereits einige Erfahrungsberichte und Fallstudien zu Unternehmen aus unterschiedlichen Branchen, die nach Scrum gearbeitet haben. Einige dieser praktischen Berichte werden in dieser Bachelorarbeit analysiert, um herauszufinden, wie gut die agile Methode Scrum in der Praxis funktioniert.

## 1.2 Forschungsfrage und Ziel

Im Kontext der Unterkapitel 1.1 lässt sich die Forschungsfrage wie folgt formulieren:

Inwieweit gelingt die Anwendung von Scrum zum Managen von Projekten in der Praxis?

Neben der Hauptfrage sollen auch einige Teilfragen geklärt werden:

- Wird Scrum als Ganzes angewendet oder werden nur bestimmte Elemente von Scrum übernommen?
- An welchen Stellen kommt es zu Schwierigkeiten bei der Umsetzung?
- Welche Verbesserungen beim Projektmanagement lassen sich nach der Einführung und Nutzung von Scrum feststellen?

---

<sup>5</sup> Vgl. Layton/Ostermiller, 2018, S. 39–40.

<sup>6</sup> Vgl. Roock/Wolf, 2018, S. 5–6.

<sup>7</sup> Vgl. Fischer, 2018, S. 81.

Das Ziel dieser Arbeit ist es, zu untersuchen, wie die Umsetzung von Scrum in der Praxis wirklich funktioniert, und damit eine Orientierungshilfe mit praktischem Bezug für alle potenziellen Scrumanwender anzubieten.

### 1.3 Vorgehensweise

Der Kern dieser Arbeit besteht aus einem theoretischen und einem praktischen Abschnitt. Im theoretischen Teil wird hauptsächlich die Fachliteratur zur Informationsbeschaffung herangezogen, um die klassischen und agilen Projektmanagementansätze zu beschreiben. Der Fokus liegt dabei auf agilen Ansätzen, die sehr ausführlich am Beispiel von Scrum erklärt werden. Die klassischen Methoden dagegen werden knapp und allgemein dargestellt.

Der praktische Teil konzentriert sich auf die in der Unternehmenspraxis angewandte Theorie. Dazu werden Erfahrungsberichte und Fallstudien zum Einsatz von Scrum in der Praxis analysiert und ausgewertet. Dabei sollen Erkenntnisse gewonnen werden, wie Scrum von Unternehmen angewandt wird, und mit welchen Ergebnissen dabei zu rechnen ist.

### 1.4 Kapitelbeschreibung

Das erste Kapitel soll der Einführung in das Thema dienen. Dort werden die Hintergründe des Projektmanagements allgemein und agiler Methoden insbesondere erklärt. Danach wird die Forschungsfrage formuliert und die Vorgehensweise beim Erstellen dieser Thesis erläutert. Mit dem zweiten Kapitel fängt der theoretische Teil dieser Arbeit an. Er startet mit einer kurzen Beschreibung von Ansätzen des klassischen Projektmanagements und Annahmen, die ihm zugrunde liegen. Danach folgt eine ausführliche Vorstellung agiler Vorgehensweisen am Beispiel von Scrum. Mit dem dritten Kapitel fängt der praktische Teil der Thesis an, in dem eine Analyse ausgewählter Erfahrungsberichte und Fallstudien aus der Praxis stattfinden wird. Dabei sollen Informationen über die Umsetzung von Scrum in der Praxis und seinen Auswirkungen gewonnen werden. Im vierten Kapitel werden die Ergebnisse der Analysen zusammengefasst und ausgewertet. Mit den Antworten auf Forschungsfragen und einer Schlussfolgerung im Fazit im fünften Kapitel wird diese Bachelorarbeit beendet.

## 2 Klassisches und agiles Projektmanagement

### 2.1 Definitionen

Vor dem eigentlichen Einstieg in das Thema sollen zunächst die grundlegenden Begriffe „Projekt“ und „Projektmanagement“ erläutert werden. Damit soll ein einheitliches Verständnis beider Termini in dieser Arbeit erreicht werden.

Die Projektdefinition des Deutschen Institut für Normierung (DIN) lautet: *„Vorhaben, das im Wesentlichen durch Einmaligkeit der Bedingungen in ihrer Gesamtheit gekennzeichnet ist“*.<sup>8</sup>

Die Einmaligkeit eines Vorhabens hat hier die zentrale Bedeutung. Erstens bedeutet sie eine immanente Neuartigkeit, die in der Erschaffung von etwas Neuem resultieren soll. Zweitens meint die Einmaligkeit einen festgelegten Zeitrahmen. Ein Projekt muss demnach jeweils einen geplanten Start- und Abschlusszeitpunkt haben. Und drittens bezieht sich die Einmaligkeit auf die zur Verfügung stehenden Ressourcen in Form von Budget, Mitarbeitern und Materialien. Diese unterliegen bestimmten Anforderungen und sind begrenzt. Einzelne Facetten der Einmaligkeit grenzen Projekte von Routinetätigkeiten ab.<sup>9</sup>

Projektmanagement wird durch das DIN wie folgt definiert: *„Gesamtheit von Führungsaufgaben, -organisation, -techniken und -mitteln für die Initiierung, Definition, Planung, Steuerung und den Abschluss von Projekten“*.<sup>10</sup> Das Projektmanagement ist Zielorientiert und seine wesentlichen Aufgaben sind Planung, Steuerung und Kontrolle eines Projektes. Bei der Planung werden unter anderem Projektziele definiert, Projektteams besetzt, Abläufe, Termine und das Budget festgelegt. Die Steuerung umfasst Funktionen wie Anleitung und Motivation von Mitarbeitern, Überwachung des Projektverlaufs, Korrekturmaßnahmen bei Planabweichungen und Projektkoordination. Bei der Kontrolle geht es um die Überprüfung der Wirksamkeit der geplanten Maßnahmen in Bezug auf alle Aspekte der Projektplanung.<sup>11</sup>

---

<sup>8</sup> Deutsches Institut für Normung, 2016, S. 155.

<sup>9</sup> Vgl. Wytrzens, 2017, S. 35–36.

<sup>10</sup> Deutsches Institut für Normung, 2016, S. 158.

<sup>11</sup> Vgl. Wytrzens, 2017, S. 25–26.

## 2.2 Klassisches Projektmanagement

Das klassische Projektmanagement wird in der Fachliteratur oft als Plangetrieben bezeichnet. Die Ursache hierzu ist in der detaillierten, systematischen und umfänglichen Planung der Projekte von Anfang bis Ende zu sehen. Diese soll auch, so der Denkansatz, die Projekte zum Erfolg führen.<sup>12</sup> Der Planung liegen oft folgende Annahmen zugrunde:

- Alle Anforderungen und Risiken lassen sich im Voraus vollständig beschreiben<sup>13</sup>
- Der Auftraggeber kann das Projektziel schon zu Beginn exakt definieren
- Seine Spezifikation reicht als die Basis für den Projektplan aus
- Etliche Änderungen im Verlauf des Projektes sind unerwünscht
- Änderungen der Spezifikation führen zu Änderungen der existierenden Pläne
- Komplexität des Projektes kann durch detaillierte Pläne abgemildert werden
- Projektanalyse am Projektende reicht aus um aus Fehlern zu lernen<sup>14</sup>

Nach Abschluss der Planung und Klärung aller Formalitäten, was aufgrund des Umfangs durchaus länger dauern kann, beinhaltet der Projektplan ein übergeordnetes Ziel. Es gilt das Projektergebnis in einer bestimmten Qualität, zu einem bestimmten Zeitpunkt und zu vereinbarten Kosten zu liefern.<sup>15</sup> Darauf folgt die Durchführung, wobei alle geplanten Arbeiten in der festgelegten Reihenfolge erledigt werden. Änderungen sind in diesem Abschnitt nicht vorgesehen und können zu Verzögerungen führen. In einem klassischen Projektumfeld findet die Umsetzung in der Regel in mehreren aufeinander folgenden Phasen statt.<sup>16</sup> Die Durchführung kann mithilfe verschiedener Managementmodelle erfolgen. Zu den bekanntesten Vorgehensweisen gehört das Wasserfallmodell. Für jede Phase dieses Modells muss vor Projektbeginn exakt definiert werden, was dort zu erreichen ist. Sind alle Aufgaben in einer Phase vollständig erledigt und das vorher definierte Ergebnis erreicht, darf die nachfolgende Phase starten (Abb. 1). Werden jedoch am Ende einer Phase Abweichungen zum vordefinierten Phasenergebnis festgestellt, darf die nachfolgende Phase nicht mehr starten. Zuerst müssen alle Abweichungen beseitigt werden, was meist einen Sprung in eine der früheren Phasen bedeutet.

---

<sup>12</sup> Vgl. Meyer/Reher, 2016, S. 36.

<sup>13</sup> Vgl. Preußig, 2018, S. 14.

<sup>14</sup> Vgl. Timinger, 2017, S. 165.

<sup>15</sup> Vgl. Kusay-Merkle, 2018, S. 48.

<sup>16</sup> Vgl. Preußig, 2018, S. 19.



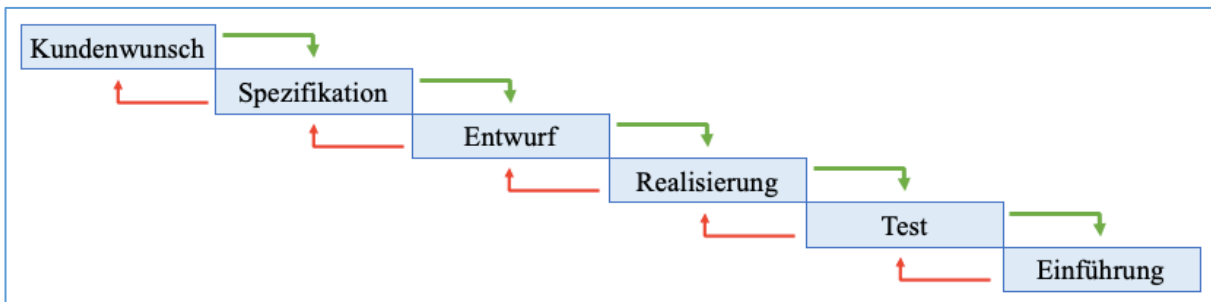


Abb. 1 Das Wasserfallmodell. In Anlehnung an Timinger 2017.

Sind alle Phasen durchlaufen und das Projekt abgeschlossen, kann sein Erfolg überprüft werden. Als erfolgreich gilt ein Projekt, wenn das Projektziel in verabredeten Umfang und Qualität, zu vereinbarten Kosten und geplantem Zeitpunkt erreicht wurde.<sup>17</sup> Die klare Struktur und eine einfache Definition des Projektfortschritts machten dieses Modell so beliebt.<sup>18</sup> Klassische Vorgehensmodelle können gut funktionieren, wenn das Projektumfeld stimmt und bestimmte Merkmale aufweist:

- Die Rahmenbedingungen bleiben stabil und unterliegen kaum Veränderungen
- Das Ergebnis und die Anforderungen sind vollständig klar
- Es gibt viele Erfahrungswerte aus ähnlichen Projekten
- Es werden keine oder nur wenige Änderungen im Projektverlauf erwartet
- Das Feedback des Auftraggebers spielt keine große Rolle im Projekt
- Das Problemfeld ist nicht komplex
- Die Lieferung von Zwischenergebnissen hat keine hohe Relevanz<sup>19</sup>

Je weniger dieser Anforderungen erfüllt werden, desto komplexer wird das Projektumfeld. Die plangetriebenen Methoden sind nicht für komplexe Bedingungen gedacht und können zum Verfehlen von Zielen führen. Komplexität ist eben der Grund, warum agile Methoden adoptiert werden und dementsprechend besser für solche Umgebungen geeignet sind.

## 2.3 Agiles Projektmanagement

Die stetig wachsende Komplexität der Umgebung, in der Projekte geplant und entwickelt werden, begünstigt die Suche nach neuen Vorgehensweisen und Adoption agiler Methoden.<sup>20</sup> Komplexe Prozesse sind von Dynamiken geprägt, deren Folgen oft nicht rückgängig gemacht

<sup>17</sup> Vgl. Kusay-Merkle, 2018, S. 48.

<sup>18</sup> Vgl. Timinger, 2017, S. 38–39.

<sup>19</sup> Vgl. Kusay-Merkle, 2018, S. 22.

<sup>20</sup> Vgl. Canditt, 2018, S. 1.

werden können. Eine komplexe Umgebung ist intransparent für die Entscheider, denn diese haben keine Möglichkeit das Gesamtkonstrukt zu überblicken. Handlungsprognosen können nicht abgegeben werden und Unerwartetes jederzeit auftreten.<sup>21</sup> Komplexe Umgebungen führen zu komplexen Projekten, und die lassen sich durch angepasstes agiles Vorgehen etwas einfacher steuern. Der Begriff „Agilität“ wurde bisher nicht abschließend definiert. Recht aussagekräftig erscheint die allgemeine Definition des Gabler Wirtschaftslexikon dazu: *„Agilität ist die Gewandtheit, Wendigkeit oder Beweglichkeit von Organisationen und Personen bzw. in Strukturen und Prozessen. Man reagiert flexibel auf unvorhergesehene Ereignisse und neue Anforderungen. Man ist, etwa in Bezug auf Veränderungen, nicht nur reaktiv, sondern auch proaktiv.“*<sup>22</sup>

Diese Definition beschreibt recht treffend die agile Projektsteuerung. Agile Vorgehensweisen gehen bewusst davon aus, dass ein Projektlebenszyklus zu Beginn eben nicht sehr genau vorhergesehen werden kann und Vieles noch unbekannt ist. Aus diesem Grund erfolgen die Planung und Umsetzung bei agilen Methoden immer iterativ, d.h. in kleinen Schritten. Diese Vorgehensweise erlaubt viele Anpassungen und Optimierungen im Projektverlauf und verleiht Projekten eine hohe Flexibilität.<sup>23</sup> Die kurzen Umsetzungsphasen führen jeweils zu einem Teilergebnis, das aktuellen Anforderungen entspricht und zeitnah dem Kunden zwecks Feedbacks vorgestellt wird. Der Kunde agiert als Ansprechpartner für das Projektteam und bleibt immer auf dem Laufenden. Neue Anforderungen und Verbesserungsvorschläge können somit jederzeit von beiden Seiten eingebracht werden. Frühere Anforderungen, die nicht mehr benötigt werden, können rechtzeitig entfernt werden. Dies vermindert den Arbeitsaufwand, verkürzt die Projektlaufzeit und verbessert die Qualität der Teilergebnisse, da nur die erforderlichen Projektarbeiten ausgeführt werden. Die Kommunikation zwischen dem Projektteam und dem Auftraggeber erfährt eine ganz neue Qualität. Der permanente Austausch senkt die Risiken und lässt gleichzeitig das Wissen des Projektteams ansteigen, da der Kunde immer besser verstanden wird.<sup>24</sup>

Die Basis für agile Vorgehensmodelle bildet das „Agile Manifest“. Es wurde 2001 von mehreren Softwareentwicklern verfasst, die bereits agile Praktiken anwendeten.<sup>25</sup> Den Manifestkern bilden vier Werte und seine Definition lautet:

---

<sup>21</sup> Vgl. Fischer-Korp, 2018, S. 91–92.

<sup>22</sup> Gabler Wirtschaftslexikon: „Agilität“, 2019 (online).

<sup>23</sup> Vgl. Timinger, 2017, S. 162–163.

<sup>24</sup> Vgl. Bause, 2018, S. 6.

<sup>25</sup> Vgl. Brandstätter, 2013, S. 6.

*„Wir erschließen bessere Wege, Software zu entwickeln, indem wir es selbst tun und anderen dabei helfen. Durch diese Tätigkeit haben wir diese Werte zu schätzen gelernt:*

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

*Das heißt, obwohl wir die Werte auf der rechten Seite wichtig finden, schätzen wir die Werte auf der linken Seite höher ein.“<sup>26</sup>*

Diese Werte kristallisierten sich aus praktischen Erfahrungen heraus. Die Softwareentwickler hatten erkannt, dass nur wenn alle Beteiligten (Entwickler, Kunden, Lieferanten) gemeinsam arbeiten und kooperieren, es gelingen kann, trotz hoher Komplexität echten Nutzen für alle Beteiligten zu stiften.<sup>27</sup> Die Kernwerte im Manifest gelten bis heute, aber beschränken sich längst nicht mehr auf reine Softwareentwicklungsprojekte. Folglich kann der Begriff „Software“ im Manifest in Abhängigkeit vom Projektgegenstand durch „Produkt“, „Ergebnis“ oder „Prozess“ ersetzt werden.<sup>28</sup>

Die Funktionsweise des agiles Projektmanagements mit seinen Bestandteilen und Besonderheiten wird im nächsten Kapitel am Beispiel von Scrum, einer der bekanntesten agilen Vorgehensweise, beschrieben.<sup>29</sup>

## 2.4 Scrum

### 2.4.1 Scrum-Entstehung

Der Begriff „Scrum“ wurde erstmals von den japanischen Wirtschaftswissenschaftlern Nonaka und Takeuchi in Verbindung mit der Entwicklung von etwas Neuem verwendet. Sie veröffentlichten 1986 den Artikel „The New New Product Development Game“, in dem sie von ihrem „Rugby-Ansatz“ berichteten. Die beiden Wissenschaftler gingen davon aus, dass einer der wichtigsten Erfolgsfaktoren bei der Entwicklung von neuen Produkten die Nähe der Teammitglieder zu einander ist. Beim Rugby gibt es eine ähnliche Situation, in der sehr viele Spieler eng zusammenstehen. Es entsteht eine Art Gedränge, das den englischen Begriff

---

<sup>26</sup> Beck et al.: „Manifest für Agile Softwareentwicklung“, 2001 (online).

<sup>27</sup> Vgl. Bartonitz et al., 2018, S. 4.

<sup>28</sup> Vgl. Timinger, 2017, S. 163.

<sup>29</sup> Vgl. Hanschke, 2017, S. 9.

Scrum trägt. Dabei agieren die Rugbyteams wie kleine, selbstorganisierte Einheiten, denen von außen nur eine grobe Richtung vorgegeben wird. Die Umsetzung bei der Zielerreichung bleibt den Teams frei überlassen und erfordert enge Zusammenarbeit. Genau diese Art von Zusammenarbeit sollte laut Nonaka und Takeuchi auch Projekte erfolgreich machen. Einige Jahre später nutzten die zwei Softwareingenieure, Ken Schwaber und Jeff Sutherland, genau diesen „Rugby-Ansatz“, um daraus ein Rahmenwerk für agile Softwareprojekte zu entwickeln. Mit Verweis auf Nonaka und Takeuchi nannten sie dieses Rahmenwerk SCRUM und dokumentierten es im Scrum-Guide.<sup>30</sup>

#### 2.4.2 Was ist Scrum?

Scrum ist ein agiles Projektmanagementmodell, das einen Rahmen für die Entwicklung von projektspezifischem Vorgehen bietet. Seine iterativ-inkrementelle Vorgehensweise eignet sich besonders zum Managen von komplexen Projekten. Komplexität wird durch kurze wiederkehrende Planungs- und Entwicklungszyklen sowie permanente Kundenfeedbacks aufgefangen und das Endergebnis entsteht schrittweise mit dem Fokus auf das Wesentliche. Mögliche Störungen und Risiken werden früher transparent und damit beherrschbarer. Insofern eignet sich Scrum zur Anwendung in komplexen Umgebungen, wo Überprüfen und Anpassen, regelmäßige Feedbacks sowie daraus gewonnenes Wissen zielführender sind als umfangreiche, detaillierte Planung mit anschließender Durchführung nach klassischen Vorgehensweisen.<sup>31,32</sup>

Wegen seines Ursprungs wird Scrum oft als eine agile Softwareentwicklungsmethode bezeichnet. Doch Boris Gloger, der von Ken Schwaber persönlich zum Scrum Master ausgebildet wurde, stellt fest, dass Scrum genauso gut für viele andere Anwendungen geeignet ist. Projektmanagement, Produktentwicklung, Produktmanagement und Organisationsentwicklung können hier beispielhaft genannt werden.<sup>33,34</sup>

Der theoretische Ansatz von Scrum basiert laut Sutherland und Schwaber auf der empirischen Prozesssteuerung, bei der Wissen aus Erfahrungen gewonnen wird und Entscheidungen auf Basis des Bekannten getroffen werden. Die iterative und inkrementelle Vorgehensweise erlaubt es Scrum, Prognosen zu optimieren und Risiken zu kontrollieren. Um die

---

<sup>30</sup> Vgl. Simschek/Kaiser, 2018, S. 29.

<sup>31</sup> Vgl. Dräther et al., 2013, S. 13–14.

<sup>32</sup> Vgl. Kusay-Merkle, 2018, S. 34–35.

<sup>33</sup> Vgl. Gloger, B. 2013, S. 15–19.

<sup>34</sup> Vgl. borisgloger.com: „Boris Gloger“, 2019 (online).

Funktionalität des empirischen Prozesses Scrum zu gewährleisten, müssen drei Voraussetzungen, die so genannten tragenden Säulen, gegeben sein:

**Transparenz** – erfordert, dass die wesentlichen Aspekte des Prozesses nach einem gemeinsamen Standard definiert werden, um ein gemeinsames Verständnis des Geschehens zu erzeugen. Die wichtigsten Aspekte des Prozesses sollen für diejenigen sichtbar sein, die das Ergebnis verantworten.

**Überprüfung** – die Scrum-Artefakte (Ergebnisse) und Fortschritte müssen regelmäßig in Bezug auf die Zielerreichung überprüft werden, um Abweichungen zu erkennen. Die Häufigkeit der Kontrollen soll so gewählt werden, dass die eigentliche Arbeit dadurch nicht behindert wird. Stattdessen sollen die Überprüfungen von kompetenten Prüfern gewissenhaft dort vorgenommen werden, wo die Arbeit verrichtet wird.

**Anpassung** – wird bei der Prüfung festgestellt, dass es nicht akzeptable Abweichungen gibt und das daraus resultierende Ergebnis ebenfalls nicht akzeptabel sein wird, müssen der Prozess und das betroffene Material angepasst werden. Die Anpassung hat schnell zu erfolgen, um weitere Abweichungen zu minimieren.<sup>35</sup>

Das Vorgehensmodell Scrum besteht aus drei Rollen, fünf Ereignissen, drei Artefakten und einigen wenigen Regeln, weswegen es einfach zu verstehen ist. Als agiles Rahmenwerk stellt Scrum den Menschen in den Mittelpunkt. Denn nur durch Interaktionen der Individuen führen agile Methoden zum Erfolg. Scrum ist ein leichtgewichtiges Rahmenwerk, aber keine Komplettlösung mit vordefinierten Verfahrensanweisungen. Scrum muss an die jeweilige Organisation angepasst werden und ist deswegen oft schwierig anzuwenden.<sup>36,37</sup>

### 2.4.3 Scrum-Rollen

Die zwingend erforderlichen Rollen im Scrum sind Product Owner, Scrum Master und das Team. Zusammen bilden sie das Scrum-Team. Eine Besonderheit von Scrum ist hierbei, dass es keine Projektleiterrolle gibt.

**PRODUCT OWNER** – trägt die wirtschaftliche Verantwortung für das Projekt. Er repräsentiert den Auftraggeber und die Stakeholder und erzeugt aus ihren Anforderungen eine Produktvision (mit Produkt ist hier und im weiteren Verlauf das Phasen- bzw.

---

<sup>35</sup> Vgl. Sutherland/Schwaber, 2017, S. 4–5.

<sup>36</sup> Vgl. Hanschke, 2017, S. 12.

<sup>37</sup> Vgl. Pichler, 2009, S. 1–2.

Projektendergebnis gemeint). Daraus definiert er wichtige Produkteigenschaften und legt ihre Prioritäten fest. Es liegt in seiner Verantwortung, dass das Team immer an den wertvollsten Teilen des Projektes aus der Sicht der Organisation arbeitet. Durch kontinuierliche Priorisierung wird der Wert des Projektes gesteigert.<sup>38</sup> Die Hauptaufgaben des Product Owners sind:

**Zielvision entwickeln** – der Product Owner nutzt die von Stakeholdern kommunizierten Wünsche, Bedürfnisse, Funktionen und Eigenschaften, um eine Zielvision zu entwickeln. Diese beschreibt das Projektziel und wird dem Team noch vor Projektbeginn möglichst ausführlich erläutert. Hierbei ist wichtig, dass sich die Vision in Köpfen der Teammitglieder festsetzt und ein klares Ziel vor Augen erzeugt. Die Vision dient der Orientierung im Projektverlauf und legt den Grundstein für seinen Erfolg.<sup>39</sup>

**Product Backlog erstellen** – alle Anforderungen werden im Product Backlog dokumentiert und heißen dann Backlog Items. Der Product Owner erfasst die Anforderungen, entscheidet über ihre Bedeutung für das Projekt und priorisiert sie. Die Priorisierung der Items entscheidet über den Projektverlauf, da die wichtigsten Items zuerst umgesetzt werden.<sup>40</sup>

**Zusammenarbeit mit dem Team** – der Product Owner arbeitet während der gesamten Projektlaufzeit eng mit dem Team zusammen. Durch Detaillierung und Verfeinerung der Items hilft er dem Team, die Kundenbedürfnisse und Anforderungen sowie ihre Umsetzung im Projekt zu verstehen. Er steht dem Team permanent als Ansprechpartner zur Verfügung und hat immer Zeit für das Team.<sup>41</sup>

**Ergebnisse beurteilen** – iteratives Vorgehen führt zu regelmäßigen (Zwischen-) Ergebnissen im Projektverlauf. Der Product Owner inspiziert diese auf Übereinstimmung mit den Anforderungen und entscheidet, ob das aktuell Gelieferte akzeptiert wird oder nicht. Sein Feedback vergrößert das Teamwissen und bestätigt, ob der eingeschlagene Weg richtig ist.<sup>42</sup>

**Stakeholdermanagement** – der Product Owner ist die Stimme interner und externer Stakeholder. Er arbeitet eng mit allen Stakeholdern zusammen, sammelt ihre Anregungen und Informationen und verfeinert damit die Projektvision und den Product Backlog.<sup>43</sup>

---

<sup>38</sup> Vgl. Gloger, 2013, S. 78.

<sup>39</sup> Vgl. Röpstorff/Wiechmann, 2016, S. 75–76.

<sup>40</sup> Vgl. Gloger, 2013, S. 81–82.

<sup>41</sup> Vgl. Pichler, 2009, S. 11.

<sup>42</sup> Vgl. Wintersteiger, 2012, S. 33–34.

<sup>43</sup> Vgl. Rubin, 2014, S. 209.

**Releaseplanmanagement** – durch die Priorisierung der Backlog Items steuert der Product Owner den Projektverlauf. Er bestimmt, was zuerst erledigt und dem Auftraggeber zwecks Feedbacks vorgestellt wird. Mit Hilfe des Teams erfährt der Product Owner die geschätzte Bearbeitungszeit eines Items, bestimmt durch die Anordnung der Items den Liefertermin und generiert daraus den Releaseplan. Diesen Plan aktualisiert er ständig und informiert den Kunden oder seine Vorgesetzten über den Projektstatus anhand des Plans.<sup>44</sup>

**SCRUM MASTER** – unterstützt alle Beteiligten dabei, Scrum-Werte, -Prinzipien, und -Praktiken zu verstehen und anzunehmen. Er hilft dem Team und der Organisation, Scrum zu etablieren und sorgt dafür, dass die praktische Umsetzung möglichst nach den Scrum-Guide Regeln abläuft. Er unterstützt den Product Owner und das Team immer dort, wo Bedarf entsteht. Anderen Organisationmitgliedern bringt er die Scrum-Inhalte nahe und ist oft der Ausgangspunkt für den Veränderungsprozess in Richtung einer agilen Organisation.<sup>45</sup> Seine Rolle erfüllt der Scrum Master durch Übernahme bestimmter Aufgaben:

**Team unterstützen** – Scrum Master kümmert sich um die Bedürfnisse des Teams, damit dieses ungehindert am Projekt arbeiten und seine Performance steigern kann. Er löst organisationale Probleme des Teams und schützt es von äußeren Einflüssen, wie z.B. abziehen eines Teammitgliedes aus der laufenden Phase für eine „dringende Aufgabe“.

**Hindernisse beseitigen** – der Scrum Master beseitigt die Hindernisse, die im Projektverlauf zum Vorschein kommen. Dies können Störungen aller Art sein, die das Team daran hindern, seine Arbeit effektiv zu erledigen. Dazu muss der Scrum Master oft viel Überzeugungsarbeit leisten und beweisen, warum die Hindernisse direkt beseitigt werden müssen und welche Vorteile störungsfreies Arbeiten des Teams bringt.<sup>46</sup>

**Product Owner unterstützen** – der Scrum Master hilft dem Product Owner, die Scrum-Praktiken zu verinnerlichen und richtig einzusetzen. Dazu gehören die Betonung der Wichtigkeit der Teilnahme an allen Ereignissen, Unterstützung bei der Priorisierung des Product Backlogs und bei Bedarf wiederholte Ansprache der Aufgaben des Product Owners.<sup>47</sup>

**Scrum etablieren** – Scrum Master hilft, Scrum in der Organisation zu etablieren und fungiert dabei als Vorbild. Er fördert neue Denk- und Verhaltensweisen und spricht alte und unpassende Arbeitsgewohnheiten direkt an, um sie zu verändern. Um langfristige Erfolge zu

---

<sup>44</sup> Vgl. Gloger, 2013, S. 83.

<sup>45</sup> Vgl. Röpstorff/Wiechmann, 2016, S. 50.

<sup>46</sup> Vgl. ebenda, S. 56.

<sup>47</sup> Vgl. Gloger, 2013, S. 94.

erzielen, müssen neben dem Team auch andere Organisationsebenen von Scrum überzeugt werden. Der Scrum Master muss immer wieder die Notwendigkeit und die Vorteile von Scrum den Beteiligten vor Augen führen und versuchen ihre Denkweise zu verändern.<sup>48,49</sup>

**DAS TEAM** – ist crossfunktional (interdisziplinär) mit Experten aus verschiedenen Organisationsbereichen besetzt. Dies gewährleistet, dass das Team funktionsübergreifend und in der Lage ist, sämtliche im Phasenverlauf anfallenden Aufgaben zu erledigen und ein potenziell auslieferbares Teilprodukt zu erstellen. Das Team ist selbstorganisierend, trifft autonome Entscheidungen bezüglich der zu erledigenden Aufgaben und trägt gemeinsam die Verantwortung für die Qualität des Ergebnisses. Es darf die Menge, der vom Product Owner ausgewählten Items reduzieren, falls deren Umfang ansonsten die Qualität beeinträchtigen würde. Die empfohlene Größe des Teams liegt zwischen drei und neun Personen.<sup>50</sup> Die Aufgaben des Teams sind:

**Sprints durchführen** – während eines Sprints (eines der Ereignisse) erledigt das Team die eigentliche Arbeit, die in einer Phase anfällt. Das können Programmieren, Entwerfen, Zeichnen, Bauen, Integrieren und ähnliche Tätigkeiten sein. Am Ende eines jeden Sprints soll ein potenziell auslieferbares Teilergebnis entstehen.

**Untersuchen und Anpassen** – vor Beginn der täglichen Arbeit wird im so genannten Daily Scrum der Fortschritt des laufenden Sprints überprüft. Dabei werden Anpassungen und Verbesserungen für den aktuellen Tag vereinbart. Jedes Teammitglied ist verpflichtet, am Daily Scrum teilzunehmen, damit das ganze Team den Gesamtüberblick behalten und sein Ziel erfüllen kann.

**Product Backlog pflegen** – das Team unterstützt den Product Owner beim Pflegen des Backlogs. Dazu werden Tätigkeiten, wie Anlegen, Verfeinern, Schätzen und Priorisieren der Items erledigt. Das Helfen dem Product Owner verbessert die Kommunikation zwischen den beiden Rollen und verschafft dem Team einen guten Überblick über die Backlog Items.<sup>51</sup>

**Sprintplanung** – das Team beteiligt sich zusammen mit dem Product Owner und dem Scrum Master vor Beginn jedes Sprints an der Planung. Es hilft bei der Definition des Sprintziels und wählt Aufgaben aus, die zum Erreichen des Ziels umgesetzt werden müssen. Für jede Aufgabe gibt das Team eine Schätzung bezüglich des anfallenden Umfangs der Arbeiten ab.

---

<sup>48</sup> Vgl. Pichler, 2009, S. 20.

<sup>49</sup> Vgl. Rubin, 2014, S. 227.

<sup>50</sup> Vgl. Roock/Wolf, 2018, S. 20.

<sup>51</sup> Vgl. Rubin, 2014, S. 236–237.



Die Schätzung erfolgt oft anhand der so genannten Story Points oder mithilfe bestimmter Techniken, wie z.B. Planning Poker (jedes Teammitglied schätzt unabhängig, anschließend diskutiert das ganze Team bis nur zwei Schätzgrößen verbleiben, die Größere gilt dann). Die Genauigkeit der Schätzung ist wichtig für die Messung der Geschwindigkeit des Teams, die als Velocity bezeichnet wird. Velocity beschreibt die durchschnittliche Anzahl der innerhalb eines Sprints erledigten Items und erlaubt eine Prognose bezüglich der Fertigstellung einzelner Items. Velocitykenntnis ist wichtig für die Releaseplanung.<sup>52,53</sup>

**Produkte und Prozesse anpassen** – das komplette Team hat die Verpflichtung an zwei Meetings am Ende eines jeden Sprints teilzunehmen, Sprint Review und Sprint Retrospektive (beides Ereignisse). Dabei geht es um die Überprüfung und Anpassung der Ergebnisse, Zusammenarbeit und Prozesse der gerade abgeschlossenen Phase.<sup>54</sup>

#### 2.4.4 Scrum-Artefakte

Als Artefakte werden Zwischen- und Endresultate der Scrum-Aktivitäten bezeichnet, die im Projektverlauf entstehen. Sie dienen dazu, dem gesamten Scrum-Team eine möglichst hohe Transparenz über die wesentlichen Informationen und die Möglichkeit zur Überprüfung und Anpassung zu schaffen. Die drei Scrum-Artefakte heißen Product Backlog, Sprint Backlog und Inkrement.<sup>55</sup>

**PRODUCT BACKLOG** – ist die Sammlung aller Vorstellungen des Kunden, die das Projektergebnis betreffen. Sämtliche aktuell existierenden Anforderungen, Eigenschaften, Funktionen und Wünsche werden vom Product Owner hier als Backlog Items aufbewahrt. Oft werden die Items in Form von so genannten User Stories eingetragen (eine User Story beschreibt die Anforderung aus der Sicht des Anwenders, z.B.: ich drücke den Knopf und das Gerät startet). Die Besonderheit des Product Backlogs besteht darin, dass er weder zu Beginn noch im Verlauf des Projektes als vollständig betrachtet wird. Sein Inhalt wird während der gesamten Projektlaufzeit permanent ergänzt, aktualisiert und nach seinem Wert für das Projekt und den Kunden sortiert.<sup>56</sup>

---

<sup>52</sup> Vgl. ebenda, S. 237.

<sup>53</sup> Vgl. Wintersteiger, 2012, S. 146.

<sup>54</sup> Vgl. Rubin, 2014, S. 237.

<sup>55</sup> Vgl. Sutherland/Schwaber, 2017, S. 14–17.

<sup>56</sup> Vgl. Timinger, 2017, S. 169–173.

**SPRINT BACKLOG** – enthält User Stories in Form von einzelnen Aufgaben und wird durch das Team erstellt und gepflegt. Mit der Erfassung einzelner Aufgaben legt das Team fest, wie es vorgehen will, um die Ziele des aktuellen Sprints zu erreichen. Auch der Sprint Backlog erhebt zu keinem Zeitpunkt den Anspruch auf Vollständigkeit. Er verändert sich täglich und bietet immer einen guten Überblick über den Status des Sprints. Zwecks vereinfachter Visualisierung des Sprint Backlogs existieren zwei empfehlenswerte Methoden. Die Erste ist eine Pinnwand, auf der in vier Spalten die *User Stories*, *Aufgaben*, *Aufgaben in Arbeit* und *erledigte Aufgaben* erfasst werden. Die Zweite ist ein „Burndown-Chart“, auf dem die Anzahl der Aufgaben im Sprintverlauf abnimmt. Damit haben alle Teammitglieder immer den Sprint Backlog im Blick und die Transparenz über den Sprintstatus steigt bedeutend.<sup>57,58</sup>

**INKREMENT** – ist das Ergebnis der Arbeit des Teams während eines Sprints. Mit jedem Sprint entsteht eine Erweiterung oder Veränderung des Inkrements, wodurch sich sein Nutzen erhöht. Jedes Inkrement soll dem Kunden zwecks Feedbacks vorgeführt werden. Erkenntnisse aus dem Kundenfeedback fließen in die Planung des nachfolgenden Sprints ein. Ein Inkrement gilt als fertig, wenn es vordefinierten Kriterien entspricht. Die Summe dieser Kriterien wird im Fachjargon als „Done“ bezeichnet. Was genau „Done“ bedeutet, entscheidet das komplette Scrum-Team zusammen, um ein gemeinsames Verständnis darüber zu erzielen. Eine einheitliche Definition von „Done“, die für alle Scrum-Teams gelten würde, existiert nicht.<sup>59,60</sup>

Alle drei Artefakte lassen sich eindeutig den agilen Methoden zuordnen. Sie entstehen schrittweise, lassen jederzeit Änderungen zu, sorgen für hohe Transparenz, ermöglichen schnelle Korrekturen im Projektverlauf und benötigen regelmäßiges Kundenfeedback.

#### 2.4.5 Scrum-Ereignisse

Der Scrum-Guide schreibt fünf feste Aktivitäten vor, die als Ereignisse bezeichnet werden. Dazu gehören Sprint, Sprint Planning, Daily Scrum, Sprint Review und Sprint Retrospektive. Alle Ereignisse sind befristet und haben eine festgelegte maximale Dauer. Sie wiederholen sich immer wieder, bis das Projekt abgeschlossen ist.<sup>61</sup>

---

<sup>57</sup> Vgl. Pichler, 2009, S. 102–104.

<sup>58</sup> Vgl. Timinger, 2017, S. 174–176.

<sup>59</sup> Vgl. Sutherland/Schwaber, 2017, S. 17–18.

<sup>60</sup> Vgl. Roock/Wolf, 2018, S. 26–27.

<sup>61</sup> Vgl. Sutherland/Schwaber, 2017, S. 9–14.

**DER SPRINT** – ist die Phase, in der das Team die eigentlichen Projektarbeiten erledigt, d.h. die durch den Product Owner ausgewählten Kundenanforderungen (Items) in die Praxis umsetzt (Abb. 2). Die Länge der einzelnen Sprints beträgt zwischen einer und vier Wochen und hängt vom Projektumfang und benötigtem Feedbackintervall ab. Die maximale Sprintdauer ist auf 30 Tage begrenzt. In der Regel bleibt die Länge aller Sprints innerhalb eines Projektes konstant. Am Ende eines jeden Sprints soll ein fertiges Inkrement entstehen.<sup>62,63</sup>

**SPRINT-PLANNING** – jeder Sprint beginnt mit dem Sprint Planning, das üblicherweise aus zwei aufeinander folgenden Meetings (Planning I und Planning II) besteht. Im Planning I stellt der Product Owner ausgewählte User Stories vor und es wird das Sprintziel definiert. Im Planning II entscheidet das Team, wie die Umsetzung der Stories erfolgen wird. Für Planning I trifft sich das ganze Scrum-Team und versucht die Bedeutung und den Wert der ausgewählten User Stories möglichst genau zu klären. Im Anschluss an die Klärung überprüft das Team seine Schätzungen der Stories und aktualisiert diese bei Bedarf. Planning I endet, sobald Einigung über das Sprintziel und die Stories erzielt wurde. Anschließend startet Planning II. Dabei werden die im Planning I ausgewählten Stories in einzelne Aufgaben zerlegt, in den Sprint Backlog eingetragen und innerhalb des Teams verteilt. Dies wird vom Team selbst und bei Bedarf mit Unterstützung des Scrum Masters erledigt.<sup>64</sup>

**DAILY SCRUM** – ist eine Aktivität, die der Selbstorganisation des Teams dient. Es findet täglich statt, dauert nur 15 Minuten und sollte für Berichtserstattung und Tagesplanung genutzt werden. Das Team kann überprüfen, was bisher im Sprint erreicht wurde und welche Aufgaben noch offen sind. Der Sprintfortschritt wird für alle transparent und es können Anpassungen vorgenommen werden, um das Sprintziel möglichst effizient zu erreichen. Außerdem können eventuelle Hindernisse identifiziert werden.<sup>65</sup>

**SPRINT REVIEW** – ist ein Treffen des gesamten Scrum-Teams mit den Stakeholdern, das am Ende eines jeden Sprints stattfindet. Bei der Review wird das Ergebnis des Sprints mit seinen Anforderungen verglichen. Das Erreichen des Sprintziels ist dabei wichtiger als die Umsetzung aller Anforderungen. Mit der Review werden zwei wesentliche Ziele verfolgt, vorzuführen, was das Team seit Sprintbeginn erreicht hat und das Feedback der Stakeholder

---

<sup>62</sup> Vgl. Timinger, 2017, S. 179–180.

<sup>63</sup> Vgl. Rubin, 2014, S. 96.

<sup>64</sup> Vgl. Dräther et al., 2013, S. 75–79.

<sup>65</sup> Vgl. Wintersteiger, 2012, S. 121–122.

einholen. Die Rückmeldungen dienen der Überprüfung, ob die gewählte Richtung den Vorstellungen des Kunden entspricht und der Identifikation möglicher Verbesserungen für den nächsten Sprint. Gleichzeitig lernt das Team und erweitert sein Wissen.<sup>66</sup>

**SPRINT RETROSPEKTIVE** – findet am Ende eines jeden Sprints nach der Sprint Review statt. Während der Retrospektive reflektiert das Team über die Zusammenarbeit und Prozessabläufe des gerade beendeten Sprints. Der Scrum Master verantwortet den Scrum-Prozess und sollte daher bei diesem Meeting immer anwesend sein. Der zurückliegende Sprint wird in einige Einzelteile zerlegt, um diese nach einander zu inspizieren und Verbesserungspotenziale zu erkennen. Es wird geschaut, wo die Zusammenarbeit erfolgreich war und wo nicht, welche Hindernisse störten den Prozess, was konsumierte viel Zeit ohne einen Nutzen für den Sprint zu generieren und so weiter. Überprüfen und Anpassen wird angewandt und soll dem Team ermöglichen, die Prozessabläufe und Zusammenarbeit in kommenden Sprints zu verbessern.<sup>67</sup>

Alle Scrum-Ereignisse harmonisieren sehr gut miteinander und spiegeln voll und ganz ihre Zugehörigkeit zu agilen Vorgehensweisen wider. Beim Sprint Planning wird lediglich für den nächsten Sprint geplant. Der Sprint selbst wird dazu genutzt, nur die eindeutig bekannten und exakt definierten Anforderungen umzusetzen. Das Daily Scrum dient hoher Transparenz und Selbstorganisation des Teams. Und während der Sprint Review und Retrospektive werden das Projektgegenstand und der Scrum-Prozess überprüft und angepasst. Die Zusammenarbeit verbessert sich und das Lernen findet statt. Auch der Kunde spielt eine wichtige Rolle und wird laufend involviert und informiert.

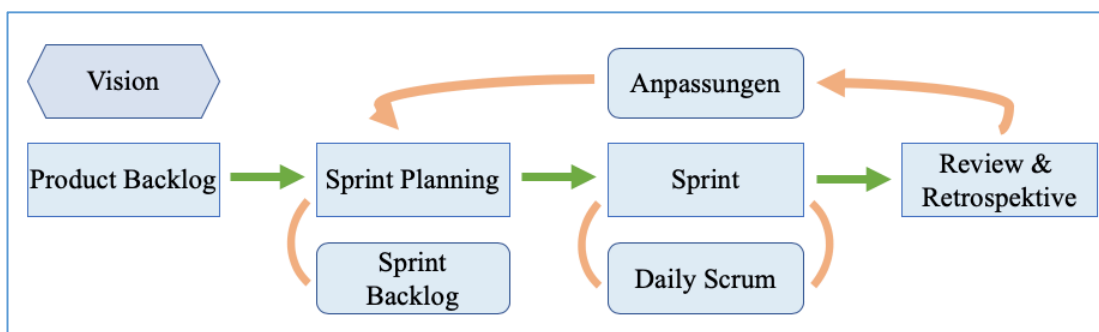


Abb. 2 Der Sprint. In Anlehnung an Pichler 2009

<sup>66</sup> Vgl. Röpstorff/Wiechmann, 2016, S. 254–256.

<sup>67</sup> Vgl. Dräther et al., 2013, S. 89–95.

## 2.4.6 Projektmanagement mit Scrum

Nach der Vorstellung einzelner Bestandteile von Scrum, darf der Scrum-Prozess als Ganzes natürlich auch nicht fehlen. Daher wird in diesem Kapitel der komplette Ablauf des von Scrum, der so genannte Scrum-Flow (Abb. 3), beschrieben. Der Prozess fängt mit einer Idee an, die der Auftraggeber dem Product Owner schildert. Dieser versucht während der ersten Gespräche mit dem Kunden möglichst viele Anforderungen maximal zu präzisieren und erzeugt daraus Backlog Items. Diese sortiert er nach ihrem Wert für das Projekt und liegt somit die Reihenfolge ihrer Umsetzung fest. Gleichzeitig wird ein interdisziplinäres Team aufgestellt, das sich aus Experten aus verschiedenen Unternehmensbereichen zusammensetzt, die in Abhängigkeit vom Projektziel ausgewählt werden. Ihre Zusammensetzung ermöglicht es Ihnen, im Idealfall alle Aufgaben im Projektverlauf ohne externe Unterstützung zu bewältigen. Beim ersten Sprint Planning stellt der Product Owner seine Zielvision vor und wählt die Items für den ersten Sprint aus. Dann wird das Sprintziel definiert. Das Team zerlegt die Items in einzelne Aufgaben, trägt diese in den Sprint Backlog ein und verteilt sie an die Teammitglieder. Der erste Sprint beginnt und dauert zwischen einer und vier Wochen. Im Sprintverlauf sorgt der Scrum Master dafür, dass keine Hindernisse die Arbeit des Teams stören, schützt das Team von äußeren Einflüssen und leistet Überzeugungsarbeit in der Organisation, um die agile Denkweise zu fördern. Das Team trifft sich jeden Tag zu Daily Scrums, überprüft den Sprintfortschritt und sortiert die Aufgaben neu, um das Sprintziel möglichst effizient zu erreichen. Am Ende des Sprints treffen sich der Kunde und das gesamte Scrum-Team zur ersten Sprint Review. Das Team stellt das erste Inkrement vor, der Kunde begutachtet die Ergebnisse und liefert dem Team sein Feedback. Aus der Rückmeldung gewinnt das Team neue Erkenntnisse, erweitert sein Wissen und generiert Verbesserungen für den kommenden Sprint. Der Product Owner gewinnt neue Ideen für seine Vision und aktualisiert den Product Backlog, auch neue Anforderungen nimmt er mit auf. Auf die Review folgt die Sprintretrospektive, in der die Zusammenarbeit der Teammitglieder sowie Prozessabläufe unter die Lupe genommen werden. Störungen und ihre Ursachen werden definiert und mögliche Verbesserungen für den nächsten Sprint eingeplant. Damit ist die erste Iteration, ein kompletter Sprint, abgeschlossen und das Scrum-Team trifft sich zum zweiten Sprint Planning. Die Sprints werden solange wiederholt bis das Projektziel, das sich aus einzelnen Inkrementen zusammensetzt, erreicht ist.<sup>68</sup>

---

<sup>68</sup> Vgl. Gloger, 2013, S. 9–11.

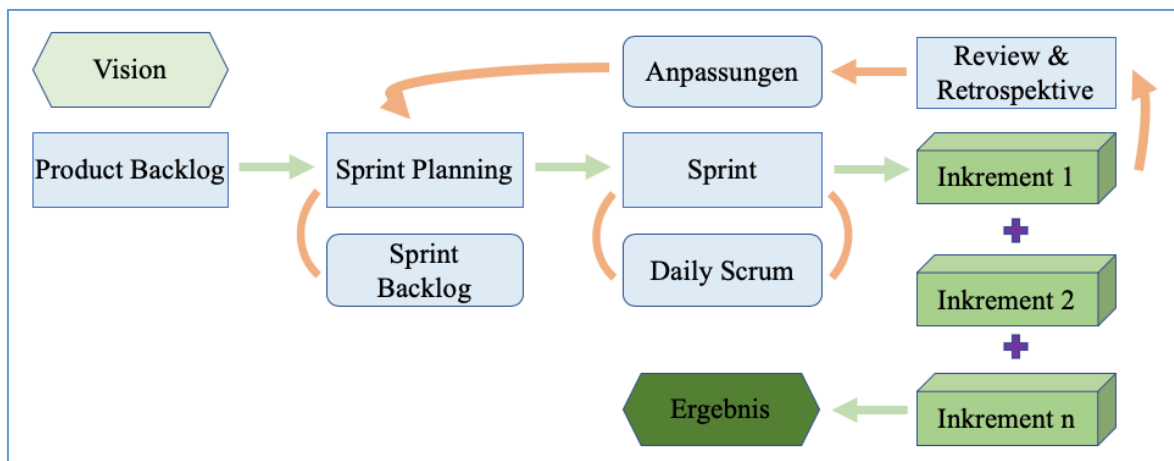


Abb. 3 Der Scrum-Flow. In Anlehnung an Pichler 2009.

Drei Rollen, fünf Ereignisse und drei Artefakte wirken auf bestimmte Art und Weise und nach wenigen klaren Regeln zusammen. Rollende Planung lässt jederzeit Änderungen und Implementierung neuer Anforderungen zu, kurze Entwicklungsphasen mit fertigen Teilergebnissen erlauben dem Auftraggeber seine Idee im Projektverlauf immer weiter zu präzisieren, permanente Teammeetings führen zu optimierten Prozessen und verbessern die Zusammenarbeit. Am Ende entsteht ein Ergebnis, das exakt den Kundenanforderungen entspricht und alle wichtigsten Eigenschaften besitzt. So oder so ähnlich dürfte die kurze Zusammenfassung des Scrum-Prozesses in der Theorie lauten. Die entscheidende Frage ist jedoch, ob die Umsetzung der Scrum-Praktiken in die Praxis auch wirklich so einfach gelingt und zu Verbesserungen führt. Dies wird im nächsten Kapitel anhand Erfahrungen aus der Scrum-Praxis untersucht.

### 3 Scrum in der Praxis

Zwecks Überprüfung der Anwendbarkeit und Wirksamkeit von Scrum in der Praxis werden in diesem Kapitel ausgewählte Erfahrungsberichte von vier Unternehmen analysiert, die bereits nach Scrum gearbeitet haben. Alle Praxisberichte wurden von unterschiedlichen Autoren erstellt und haben jeweils einen anderen Schwerpunkt, Umfang wie auch andere Darstellungsweisen. Um einerseits ihre Vergleichbarkeit zu gewährleisten und andererseits die Forschungsfragen beantworten zu können, wird sich die Analyse nur auf bestimmte Arten von Informationen in den Fallstudien konzentrieren. Erstens: ob alle Rollen, Artefakte und Ereignisse übernommen und gemäß dem Scrum-Guide oder in angepasster Form angewandt werden. Zweitens: welche Schwierigkeiten ergeben sich bei der Einführung und was verläuft ohne Probleme. Drittens: welche Verbesserungen können nach der Einführung von Scrum festgestellt werden.

#### 3.1 Scrum bei TablonautiX

Florian Hofmann, ein Softwareentwickler mit Erfahrung in agilen Vorgehensweisen und der Mitbegründer von TablonautiX GmbH, berichtet von seinen Erlebnissen mit Scrum in einem Blog-Artikel mit dem Titel: „Ist das noch Scrum oder kann das weg?“. Die Ausgangssituation: direkt nach der Gründung von TablonautiX wurde beschlossen agil zu arbeiten. Die Wahl fiel auf Scrum, da diese Methode unter den Softwareentwicklern sehr bekannt und weit verbreitet ist. Das Ziel des Projektes war es, eine schlanke Web-Applikation zur Datenanalyse zu entwickeln. Das Team von TablonautiX hatte auch klare Erwartungen an Scrum. Steigerung der Effizienz und Effektivität, Kostenminimierung, Erreichung einer hohen Flexibilität in Bezug auf interne und externe Stakeholder-Anforderungen und schnelle Lieferung von Ergebnissen trotz der Neuartigkeit der Software.

Florian Hofmann übernahm die Rolle des Product Owners und das Team legte los. Bereits nach kurzer Zeit wurden jedoch einige Abweichungen zu Scrum „by the book“ (Anwendung des Rahmenwerks 1:1 nach dem Scrum-Guide) festgestellt. Der Scrum-Prozess fing an sich zu verändern und dem Projektgeschäft anzupassen. Es ging soweit, dass das Team sich fragte, ob es überhaupt noch Scrum ist. Insbesondere vier Elemente von Scrum haben sich als problematisch erwiesen.

**Sprint Planning** – die Unterteilung in Planning I und Planning II mit einer Gesamtdauer von bis zu 8 Stunden wurde als viel zu lange empfunden. Sehr lange bedeutete gleichzeitig sehr

teuer und war damit nicht akzeptabel. Daher wurde Planning II abgeschafft und es gab nur noch Planning I. Dort stellte der Product Owner die User Stories vor und bei Fragen oder Uneinigkeit wurden diese diskutiert oder aus dem Sprint genommen. Statt des Planning II zerlegte jeder Entwickler seine User Story selbst in kleine Arbeitspakete. Zum Schätzen entwickelte das TablonautiX-Team ein eigenes System mit Bekleidungsgrößen von S bis XL. Zuerst schätzte jedes Teammitglied eine Aufgabe unabhängig. Dann wurden die Schätzungen verglichen, die Größere bekam den Zuschlag, oder es wurde weiter diskutiert. Auswirkungen der Anpassungen:

- Positiv – gesparte Zeit bei der Planung resultierte in mehr Zeit und Budget für Kunden und Projekt.
- Negativ – über die Umsetzung einzelner User Stories entschied jeder Entwickler selbst. Er konnte seine Kollegen um ihren Rat bitten, musste es aber nicht. Somit würden mögliche Hürden teilweise zu spät entdeckt.
- Negativ – die Schätzungen des Teams waren sehr ungenau. Dadurch konnten zu viele oder zu große User Stories im Sprint Backlog landen. Dies resultierte in verfehlten Sprintzielen und ungenauen Aussagen über die Geschwindigkeit des Teams.

**Sprints** – Sprintziel-gefährdende Änderungen von Anforderungen im Sprintverlauf sind nicht erlaubt, TablonautiX wich hier jedoch teilweise ab. In dringenden Fällen wurden das Hinzufügen und Umsetzen von wichtigen Aufgaben und Anforderungen im laufenden Sprint zugelassen. Begründet wurde dies mit den ansonsten blockierten Stakeholdern oder Teammitgliedern, die nicht weiterarbeiten konnten. Ergebnisse der Abweichung:

- Positiv – die Handhabung eines Sprints wurde noch agiler, da auf Anforderungen der Stakeholder noch flexibler reagiert werden konnte. Auch einige Hindernisse konnten unverzüglich beseitigt werden und die Entwicklungsgeschwindigkeit erhöhte sich deutlich.
- Negativ – der Rhythmus eines Sprints wurde gestört und einige Sprints dauerten länger als geplant.
- Negativ – durch nachträglich hinzugefügte Aufgaben konnten nicht immer alle Sprint Backlog Einträge abgearbeitet werden und das eigentliche Sprintziel wurde verfehlt. Bei äußeren Einflüssen (z.B. Krankheiten) entstanden mehr Engpässe. Das Team reagierte darauf mit der Verlängerung des Sprints und milderte somit den entstandenen Druck etwas ab.



**Sprint Review** – soll immer zu einem festen Zeitpunkt stattfinden, was dem TablonautiX-Team nicht passte. Es wich von festen Terminen ab. Die noch vor dem Sprintende fertiggestellten Produktinkremente sollten unverzüglich dem Kunden vorgeführt werden, damit dieser sie testen und eventuell schon nutzen konnte. Ergebnisse der Abweichung:

- Positiv – die neueste Version stand dem Kunden noch schneller zur Verfügung.
- Positiv – verkürzte Lieferzeiten führten zu viel häufigerem Kundenkontakt.

Die wöchentlichen Treffen zwischen dem Product Owner und dem Kunden bezüglich der Funktionalität der neuen Version fanden weiterhin statt. Daran nahm jedoch nur noch ein Teammitglied teil. Die übrigen Teammitglieder konnten weiter am Projekt arbeiten, was positiv war. Negativ war dagegen, dass nur ein Teammitglied das ungefilterte Feedback des Kunden mitbekam.

**Scrum Master** – eine wichtige Aufgabe des Scrum Masters ist die Beseitigung von Hindernissen. Bei TablonautiX liefen alle Störungen beim Scrum Master auf und konnten von ihm nur nacheinander beseitigt werden. Dadurch verzögerte sich die Bearbeitung wichtiger Aufgaben. Also entschied sich das Team komplett gegen einen Scrum Master und verteilte seine Aufgabe auf alle Teammitglieder. Zur Einhaltung von Prozessen motivierten sich die Mitglieder gegenseitig und lösten selbst oder mithilfe des Product Owners die Konflikte. Ergebnisse der Abweichung:

- Positiv – das Team übernahm noch mehr Verantwortung und fühlte sich für den Prozess zuständig. Die Störungen stapelten sich nicht mehr.
- Negativ – zur Beseitigung der Hindernisse mussten ständig Teammitglieder abgestellt werden. Dies erforderte zusätzlichen Aufwand und gefährdete das Sprintziel.

Trotz wesentlicher Anpassungen bei vier genannten Elementen, ergaben sich deutliche Verbesserungen und TablonautiX entschied sich, sein „Scrum“ zu behalten:

- Durch verkürzte Planung blieb mehr Budget für die Entwicklung der Software übrig.
- Die im Sprint zugelassenen Änderungen ermöglichten eine wesentlich flexiblere Reaktion auf Kundenanforderungen.
- Durch modifizierte Sprint Reviews konnten fertige Produktinkremente mehrmals pro Sprint ausgeliefert werden und es gab öfter Feedback.
- Die Zusammenarbeit innerhalb des Teams verbesserte sich.

- Der fehlende Scrum Master führe zur stärkeren agilen Denkweise des Teams.<sup>69</sup>

**Auswertung:** TablonautiX versuchte alle Bestandteile von Scrum zu übernehmen und zumindest zu Beginn alle Rollen, Artefakte und Ereignisse originalgetreu umzusetzen. Im Projektverlauf wurden jedoch drei Ereignisse stark modifiziert und auf eine Rolle wurde komplett verzichtet. Im Ergebnis schaffte es TablonautiX, Scrum weder ganzheitlich noch entsprechend dem Scrum-Guide anzuwenden. Die entstandene Vorgehensweise basierte jedoch auf den Scrum-Werkzeugen und führte zu spürbaren Verbesserungen.

Schwierigkeiten ergaben sich besonders beim Durchführen von Ereignissen mit festen Zeiten und Terminen (Sprint, Sprint Planning und Sprint Review). Begrenzte Ressourcen Zeit und Budget, ungenaue Schätzungen der User Stories, Erledigung ungeplanter Aufgaben im Sprintverlauf und Nichteinhaltung von festen Terminen mit Kunden führten zu wesentlichen Veränderungen der drei Ereignisse. Auch mit der Rolle des Scrum Masters konnte sich das Team nicht anfreunden, da dieser als Flaschenhals und damit eine Art Hindernis wahrgenommen wurde.

Trotz misslungener Umsetzung gemäß dem Scrum-Guide, hat das Team von TablonautiX einige wichtige Verbesserungen nach der Einführung von Scrum festgestellt. Dabei fällt positiv die Übereinstimmung der wahrgenommenen Verbesserungen mit den Erwartungen zu Beginn des Projektes auf. Die Effizienz konnte gesteigert werden, Flexibilität in Bezug auf Anforderungen der Stakeholder hat sich ebenfalls erhöht und das Team hat es geschafft, die Produktinkremente schnell zu liefern. Die Entscheidung, diese Vorgehensweise beizubehalten, spricht für die Wirksamkeit der Scrum-Werkzeuge und ihre Vorteile.

### 3.2 Scrum bei Intel Corporation

Patrick Elwer, Chef-Ingenieur und verantwortlicher für agile Transformation bei Intel Corporation, berichtet in einer von Danube Technologies (Anbieter von Softwarelösungen und Beratung mit Fokus auf Scrum<sup>70</sup>) erstellten Fallstudie von seinen Erfahrungen bei der Umsetzung von Scrum bei Intel. Ausgangssituation: aufgrund langjähriger Erfahrungen in Konstruktion und Fertigung hatte sich bei Intel eine starke „Wassefallkultur“ entwickelt. Projekte wurden von funktionalen Teams durchgeführt, die auf dem Ergebnis des Vor-Teams

---

<sup>69</sup> Vgl. Hofmann: „Ist das noch Scrum oder kann das weg?“, 2016 (online).

<sup>70</sup> Vgl. Danube Technologies: „Danube Technologies | LinkedIn“, 2019 (online).

aufbauend jeweils ihren Teil erledigten. Dies führte unter anderem zu Überlastungen einiger Teams in späten Projektphasen und hohen Kosten. Agiles Vorgehen sollte derartige Probleme lösen.

Bei der Einführung wurde entschieden, Scrum außerhalb der Hauptprojekte in einer ruhigeren Phase auszuprobieren, in der die Testumgebungen für künftige Halbleiter erstellt wurden. Danube wurde beauftragt, Intel bei dem Scrum-Prozess zu begleiten. Sechs Teams mit einigen Sub-Teams gehörten zu der Pilotgruppe. Nach den Schulungen erklärten sich die Teams dazu bereit, die ersten drei Monate Scrum „by the book“ auszuprobieren, ohne dessen Effektivität zu hinterfragen oder den Prozess auf die Bedürfnisse von Intel anzupassen. Es wurden sieben Product Owner und nur ein Scrum Master (Patrick Elwer) bestimmt. Fünf Monate später gab es insgesamt vier Scrum Master, acht Teams und fünf weitere Teams sollten dazu kommen. Auch die ersten Schwierigkeiten wurden sichtbar. Die größte Herausforderung bestand in der Skalierung und Organisation der Zusammenarbeit aller Scrum-Teams. Nach weiteren Schulungen durch Danube, Erstellung eines Skalierungsplans sowie Ausprobieren und Adoptieren verschiedener Ansätze im Jahresverlauf, entstanden insgesamt 12 Scrum-Teams mit jeweils fünf bis neun Mitgliedern.

Nach Ablauf der drei-monatigen Pilotphase wurde entschieden, den Teams möglichst viele Freiheiten in der Selbstorganisation einzuräumen. Sie durften ausprobieren und adoptieren, was für sie am besten funktionierte. Freiwilligkeit und Selbstorganisation, aber auch die Transparenz wurden als entscheidende Faktoren, für die bis dahin erfolgte Transformation identifiziert. Zwecks höherer Transparenz wurde ein internes Wiki eingerichtet, wo alle Teams ihre besten Praktiken dokumentierten und festhielten, was gut funktionierte und was weniger. Daneben wurden einige Änderungen vorgenommen, um den Scrum-Prozess den Organisationsstrukturen und Anforderungen von Intel entsprechen zu lassen. So wurde die Rolle des Product Owners zum Manager für funktionale Teams umfunktioniert und einige zusätzliche Intel-spezifische Rollen definiert. Am Ende des ersten Jahres schien sich Scrum in der Organisation etabliert zu haben und sollte in Produktionsprojekten eingeführt werden.

Beim Start der Produktion erster Test-Halbleiter entschied Patrick Elwer, existierende Scrum-Prozesse zu überprüfen und war erstaunt. Ein Team kehrte zurück zum Wasserfallmodell, weitere Teams betrachteten ihre Arbeit mit Produktionsbeginn als erledigt und lösten sich komplett auf. Nur noch wenige Teams versuchten weiterhin nach Scrum zu arbeiten. Zwei-Wochen Sprints wurden in Tagessprints umgewandelt und die Ereignisse Planning, Daily Scrum, Review und Retrospektive schmolzen zu einem einzigen Meeting zusammen. Positiv

fiel in diesem Chaos die Anwendung einiger Kernpraktiken von Scrum auf. Priorisierung der Backlog Items nach Nutzen, Arbeiten im Rahmen des Backlogs, sowie Prozess- und Produktreview samt Verbesserungen fanden weiterhin statt, allerdings stark komprimiert. Nach einigen harten Wochen mit Fokus auf Fehlerbeseitigung und Anpassung, kehrten die verbliebenen Teams zu Zwei-Wochen Sprints zurück.

Die Produktion startete und es kam zu neuen Schwierigkeiten. Die weiterhin funktional organisierten Teams konnten viele Probleme in der Testumgebung von fertig produzierten Halbleitern nicht lösen. Auf Anraten der Berater wurde ein erstes crossfunktionales Pilotteam aufgesetzt, mit einem eindeutig positiven Ausgang. Das Team konnte viele Hindernisse selbst beseitigen, die Belastungen gingen stark zurück, Kommunikation und Wissenszuwachs innerhalb des Teams funktionierten reibungslos. Die Ergebnisse des Pilotteams überzeugten viele Manager und Scrum-Skeptiker bei Intel und begünstigten den Umstieg auf interdisziplinäre Teams. Am Ende des zweiten Jahres gab es bereits 18 Scrum-Teams und die Scrum-Prozesse funktionierten besser als je zuvor. Was bei der Adoption von Scrum gut funktionierte und den Prozess förderte, fasste Patrick Elwer wie folgt zusammen:

**Definition of „Done“** – in der Produktentwicklung werden Hardwareeinheiten auf Erfüllung von Kundenanforderungen getestet. Dazu bedarf es sehr detaillierter User Stories, die bestimmte Kriterien zur Kundenzufriedenheit erfüllen müssen. Die Erfüllung dieser Kriterien musste sowohl bei Verifizierung als auch Validierung bestätigt werden. Erst dann galt eine User Story als „Done“.

**Story Points** – gab es nur für User Stories mit dem Status „Done“. Dies zwang die Teams viel stärker auf Erfüllung aller Kriterien zu achten.

**Neun-Tage-Sprints** – mit Review und Retrospektive jeden zweiten Freitag. Damit lag jedes zweite Wochenende außerhalb des Sprints und die Teams konnten sich entspannen. Dies steigerte wesentlich die Lebensqualität und die Teammoral aller Scrum-Teams. Kurze Intervalle erlaubten bedarfsgerechte Anpassungen und die anforderungsbezogene Intensität reduzierte sich. Die Teams lieferten funktionierende Teilergebnisse jeden zweiten Freitag.

**Zentrale Scrum-Werkzeuge** – Erfassung bester Praktiken an einer Stelle mit Zugang für alle Teams und Erstellung eines Tools zum Managen von mehreren Scrum-Teams erleichterten agiles Vorgehen.

**Ein-Tag Aufgaben** – Definition kleinerer Aufgaben mit dem Ziel, diese innerhalb eines Tages zu erledigen. Unerledigte Aufgaben am Ende des Tages deuteten auf Hindernisse.

**Visualisierungen im Daily Scrum** – ermöglichten den Teams einen guten Überblick über den Status des Sprints und eventuelle Rückstände, über die sofort beraten wurde.

**Zwischenreview** – fertiggestellte Stories wurden vor der Sprint Review von dem Product Owner und einem Teammitglied inspiziert. Negative Überraschungen am Sprintende konnten so vermieden werden.

**Velocity** – spiegelte die tatsächliche Leistung der Teams und erlaubte regelmäßige Anpassung der Erwartungen und Planung entsprechend den Leistungen.

**Support der Führungsebene** – wurde als eines der wichtigsten Erfolgsfaktoren identifiziert. Mitarbeiter, die Scrum vorantrieben, wurden unterstützt und Scrum-Gegner gebremst.

**Verhaltensänderung** – die Teams lernten, dass konsequente Anwendung von Scrum zu besseren Scrum-Praktiken und Ergebnissen führte und passten Ihr Verhalten an.

Es gab auch ein Paar negative Aspekte, die Scrum hinderten. Diese wurden von Patrick Elwer wie folgt beschrieben:

**Product Owner und das Team** – zwecks Verbesserung der Kommunikation durften sich Product Owner als Teammitglieder an der täglichen Arbeit beteiligen. Dies führte oft zum Diktieren der Aufgaben durch den Product Owner und störte eine ehrliche Kommunikation innerhalb der Teams. Beim Umstieg auf interdisziplinäre Teams wurde diese Praxis nicht mehr zugelassen.

**Riesiger Product Backlog** – ein gemeinsamer und offener Product Backlog ermöglichte allen Personen mit Zugang, jederzeit neue Items hinzuzufügen. Die Teams füllten sich mit Anfragen bombardiert und einige Product Owner bestanden auf der Sperrung ihres Teils. Daher wurde für alle neuen Items ein separater Backlog eingerichtet, woraus die jeweiligen Product Owner die Items in den Product Backlog übertrugen.

Festgestellte Verbesserungen nach der Scrum-Einführung bei Intel:

- Konsistente Reduzierung des Produktentstehungszyklus um 66 Prozent.
- Plankonforme Leistung mit hoher Planungssicherheit und kaum Planabweichungen dank fester Sprintzyklen und aggressivem Steuern von Prioritäten und Umfang.
- Steigendes Engagement und prozesskonformes Verhalten der Mitarbeiter.

- Verbesserte Moral, Kommunikation und Jobzufriedenheit durch permanentes Erreichen von Zielen und eingegangener Verpflichtungen dank velocity-basierter Planung.
- Hohe Transparenz durch Sichtbarkeit von Fehlern, Hindernissen, schlechten Werkzeugen und Praktiken.

In der Summe hat Scrum die Produktentwicklung bei Intel stark geprägt und positiv beeinflusst. Der Weg zur agilen Transformation war steinig, es gab viele Rückschläge und harte Lektionen. Durch viele Scrum-Unterstützer, hohes Mitarbeiterengagement und Support der Führungsebene, gelang es den Scrum-Teams trotz allem weiter zu machen und erfolgreich zu werden.<sup>71</sup>

**Auswertung:** Intels erste Scrum-Teams verpflichteten und bemühten sich in der Testphase Scrum „by the book“ auszuprobieren. Dabei gab es zunächst keine interdisziplinären Teams, was eine grobe Abweichung darstellt. Nach der Testphase wurde eine Rolle umfunktioniert und einige Neuen hinzugefügt. Vier Pflicht-Ereignisse verschmolzen zu einem. Erst mit der Zeit konnten interdisziplinäre Teams eingeführt und die Sprintlängen sowie weitere Ereignisse eingehalten werden. Im Endergebnis schaffte es Intel, Scrum als Ganzes und größtenteils gemäß dem Scrum-Guide anzuwenden. Die umfunktionierte Rolle des Product Owners ist eine wesentliche Abweichung, weshalb eine originalgetreue Umsetzung dieser Rolle nicht gelungen ist.

Schwierigkeiten gab es bei der Skalierung und Organisation der Arbeit von mehreren Scrum-Teams. Intel tat sich schwer, von funktionalen auf crossfunktionale Teams umzusteigen und mit vorgegeben Rollen auszukommen. Es dauerte, bis viele Organisationsmitglieder anfangen agil zu denken und die Nachhaltigkeit dahinter zu erkennen. Der Product Owner funktionierte als Teammanager, was zum Diktat der täglichen Aufgaben durch ihn führte und die Interaktionen störte. Es ist ein gutes Beispiel dafür, warum der Product Owner den Product Backlog managen soll und das Team autonom über die Aufgaben entscheiden. Das Festhalten an funktionalen Teams verhinderte den rechtzeitigen Umstieg auf interdisziplinäre Teams und verursachte auch Probleme in Phasen nach der Produktion. Die Einhaltung der Ereignisse und Sprintlängen stellte sich ebenfalls schwierig heraus und stabilisierte sich erst mit der Zeit.

Gut funktioniert hat dagegen die Zusammenarbeit mit externen Scrum-Beratern. Sie schulten das Personal, begleiteten Intel bei allen Scrum-Prozessen, zeigten die Notwendigkeit von

---

<sup>71</sup> Vgl. Elwer et al., 2008.

crossfunktionalen Teams und halfen auch beim Skalieren. Persönliches Engagement einzelner Mitarbeiter und stabile Unterstützung der Führungsebene förderten die Aufrechterhaltung der Scrum-Prozesse und ihre Weiterentwicklung. In der Summe gelang es Intel, Scrum in etwas angepasster Form in die Praxis umzusetzen und seine eigene projektspezifische Vorgehensweise innerhalb des Scrum-Rahmenwerks zu entwickeln.

### 3.3 Scrum bei Kaarten Carrousel

Christiaan Verwijs, zertifizierter Scrum Trainer, Scrum Master und Mitbegründer von „The Liberators“ (Beratung in Bezug auf Agilität und Lean), berichtet in seinem Blog über die Einführung von Scrum im Online-Marketingbereich bei Kaarten Carrousel (KC ist ein B2B-Anbieter von Online-Plattformen zum Designen von individuellen Postkarten, Einladungen etc.).<sup>72</sup> Die Ausgangssituation: das Softwareteam von KC stieg mithilfe von Christiaan Verwijs kurz von der Fallstudie erfolgreich auf Scrum um. Aufgrund guter Erfahrungen im eigenen Haus wurde entschieden, Scrum auch für Marketing-Aktivitäten einzuführen, um die dort dringend notwendigen Veränderungen einzuleiten. Im Marketingbereich arbeiteten viele gute Individualisten jeder für sich, sie waren kein Team und Zusammenarbeit oder Transparenz waren kaum vorhanden. Probleme bei der Zuweisung von Verantwortlichkeiten und Überlastungen standen an der Tagesordnung. Es gab keine Strategie oder Prioritäten und es wurde nach dem FiFo-Prinzip (was zuerst reinkommt, wird auch zuerst erledigt) gearbeitet. Zum Start wurde das vorhandene Personal auf zwei Teams aufgeteilt, die sich jeweils vollständig um eine Gruppe von fachlich verwandten Webseiten kümmerten (Konzept, Inhalte usw.) und in Ihren eigenen Teamräumen saßen. Alle Aktivitäten in Bezug auf eine Gruppe von Webseiten übernahm jeweils ein Team allein, was zur Eliminierung von fast allen Teamabhängigkeiten führte. Bei nötigen Änderungen der Basisplattform kooperierten beide Teams mit dem Softwareteam. Die Sprintlänge wurde auf zwei Wochen festgelegt, da die Softwareabteilung im gleichen Rhythmus arbeitete. Jedes Team bekam eine intensive, zweitägige Schulung. Zwei Mitgründer von KC wurden zu Product Ownern und jeweils ein Teammitglied zum Scrum Master, wobei über die Rollenbesetzung zusammen mit den Teams entschieden wurde. Während der ersten drei Sprints beteiligte sich Christiaan Verwijs noch sehr viel am Geschehen, zog sich jedoch mit dem Abschluss des dritten Sprints komplett zurück. Die Teams funktionierten bereits ganz gut und die Scrum Master waren herangereift,

---

<sup>72</sup> Vgl. Verwijs: „Professional Scrum Master“, 2019 (online).

um selbst zu übernehmen. Um größere Hürden wegzuräumen, wurde ein „Scrum Master Coffee“, ein Treffen des Beraters mit den drei Scrum Mastern im zwei-wöchigen Rhythmus, eingeführt. Dort wurden Störungen besprochen, die von den Teams selbst nicht beseitigt werden konnten. Zu Beginn ging es noch sehr viel um die Zusammenarbeit der drei Teams (Marketing und Softwareentwicklung), die für bestimmte Aufgaben oder den Wissensaustausch notwendig war. Nach fünf oder sechs Sprints wurden bereits erste Verbesserungen sichtbar:

**Scrum-Elemente** – alle Rollen, Artefakte und Ereignisse funktionierten richtig gut und wurden nicht weggelassen, übersprungen oder signifikant verändert. Ähnliche Resultate hatte bereits das Softwareteam von KC.

**Product Owner** – leistete gute Arbeit, um die Vision und Strategie innerhalb der Teams zu verankern. Nach kurzer Zeit reichte ein Product Owner für drei Teams.

**Hohe Transparenz** – machte sichtbar woran alle arbeiteten, sowohl innerhalb als auch zwischen den Teams. Dies wiederum verbesserte die Zusammenarbeit und ließ einzelne Teammitglieder öfter um Hilfe bitten oder Anderen Hilfestellung anbieten.

**Sprint Backlog** – veranschaulichte das gemeinsame Sprintziel und half den beiden Teams erst recht zu echten Teams zu werden. Dies wurde immer wieder als eines der wesentlichen Erfolgsfaktoren genannt.

**Sprints** – lieferten einen starken Antrieb, die klaren Sprintziele zu erreichen. Verfehlte Sprintziele führten zur gemeinsamen Enttäuschung und erreichte Ziele zur Freude. Die feste Sprintdauer ließ die Teams bereits nach kurzer Zeit ein gutes Gefühl über den Umfang Ihrer Leistung entwickeln und steigerte Ihre Ambitionen.

**Spritzyklen** – halfen den Teammitgliedern sich auf das Wesentliche zu konzentrieren und nicht durch unwichtige Nebentätigkeiten ablenken zu lassen. Daily Scrum mit visueller Darstellung des Fortschritts funktionierte richtig gut, um die Teams auf Kurs zu halten.

**User Stories** – halfen viele riesige und komplizierte Aufgabenblöcke in einzelne, kleine und klare Aufgaben zu zerlegen und sie einfacher zu erledigen. Dies steigerte Motivation und reduzierte Scheu vor großen und komplizierten Anforderungen, die früher aufgeschoben oder nicht umgesetzt wurden.

**Visualisierung** – hohe Mengen an Aufgaben ließen die Pinnwände sehr überladen aussehen, also stiegen die Teams auf eine spezielle Software zur Visualisierung des Sprint Backlogs mit



riesigen und permanent eingeschalteten Bildschirmen in Ihren Räumen um. Dies verschaffte einen noch besseren Überblick über den Sprintstatus.

Einiges funktionierte nicht wie erwartet oder wurde an eigene Bedürfnisse angepasst:

**Product Owner** – konzentrierte sich hauptsächlich an der Vermittlung der Vision und Einhaltung sowie Ausrichtung der Gesamtstrategie. Seine üblichen Aufgaben, wie Pflege des Backlogs, Priorisierung der Items und Kommunikation mit den Stakeholdern übernahmen die Teams. (Anmerkung: der Scrum-Guide erlaubt diese Praktik, die Verantwortung für den Product Backlog trägt dabei weiterhin der Product Owner).

**Sprint Planning** – dauerte zu Beginn recht lange und wurde deswegen als sehr anstrengend wahrgenommen. Im Projektverlauf fanden die Vorbereitung dafür etwas im Voraus statt, wodurch die Dauer der Sprintplanung auf nur eine Stunde reduziert werden konnte. (Anmerkung: Eine zu stark verringerte Dauer stellt eine deutliche Abweichung dar.)

**Product Backlog** – die Unterteilung der Product Backlog Items in sehr kurze und kleine Aufgaben resultierte in einem riesigen Sprint Backlog mit sehr vielen Aufgaben (50-60), was nach Christiaan Verwijs Erfahrung weit überdurchschnittlich war. Versuche die Teamgrößen oder Sprintdauer zu verändern, brachten keinen Erfolg und die großen Sprint Backlogs existieren bis heute.

**Aufgaben** – der Versuch ein Template für einzelne Aufgaben zu erstellen, schlug aufgrund großer Vielfalt der Aufgaben fehl. Stattdessen lernten die Teams ein greifbares Ziel für jede Aufgabe zu definieren.

**Schätzung** – der Aufgaben gestaltete sich aufgrund der großen Vielfalt ebenfalls recht schwierig. Es wurde ein Kompromiss gefunden, in dem Story-Punkte progressiv in Stunden umgerechnet wurden (1 Punkt = 1 Stunde, 2 Punkte = 2-4 Stunden usw.).

**Marketingaktivitäten** – beinhalten einige wiederkehrenden Aufgaben, wie z.B. alle zwei Wochen Mailings vorbereiten und verschicken, oder jede Woche Blogartikel schreiben. Diese Aufgaben verursachten einen großen Aufwand, da sie für jeden neuen Sprint wiederholt erstellt werden mussten. Nach vielen verworfenen Ansätzen entschieden die Teams zwecks Transparenz alles in Sprint Backlogs zu behalten und existierende Einträge erneut zu verwenden.

**In Sprint Reviews** – wurden nur größere Backlog Items überprüft, da die Prüfung vieler kleinen und detaillierten Aufgaben zu aufwendig war.

„Done“ – eine einheitliche Definition von „Done“ für alle Aufgaben konnte aufgrund der Vielfalt der Aufgaben im Sprint Backlog nicht gefunden werden. Stattdessen wurden „Done“-Kriterien für übergeordnete Items definiert.

Die Auswirkungen von Scrum bei Kaarten Carrousel sehen wie folgt aus:

- Das Marketingpersonal verwandelte sich aus Individualisten in gut funktionierende Teams.
- Die Teams arbeiteten nach festen Regeln und mit Fokus auf das Wesentliche.
- Die Zusammenarbeit wurde erreicht, weiterentwickelt und sowohl innerhalb als auch zwischen den Teams etabliert.
- Komplizierte Aufgaben wurden erledigt und nicht mehr aufgeschoben oder gestrichen.
- Es herrschten eine hohe Transparenz und Klarheit über alle Aktivitäten und Ziele.<sup>73</sup>

**Auswertung:** beide Marketingteams versuchten, von Anfang an Scrum „by the book“ anzuwenden. Sie nutzten alle Rollen, Artefakte und Ereignisse, schafften es, die Sprintlänge konstant zu halten und arbeiteten entlang des Rahmenwerks. Trotz einiger Anpassungen hat KC es geschafft, Scrum als Ganzes und größtenteils gemäß dem Scrum-Guide anzuwenden. Die verringerte Dauer eines Ereignisses stellt jedoch eine wesentliche Abweichung dar.

Die festgestellten Schwierigkeiten bei der Einführung scheinen zum größten Teil aus der großen Vielfalt der Aufgaben im Sprint Backlog zu resultieren. Der Umfang des Sprint Backlogs blieb groß, die Schätzungen waren unscharf, eine gemeinsame Definition von „Done“ fehlte. Die Dauer des Sprint Planning wurde reduziert und die Sprint Reviews beschränkten sich nur auf Größere statt alle Items. Auch konnte keine optimale Lösung für wiederkehrende Aktivitäten gefunden werden.

Positiv auf die Umsetzung haben sich leicht verständliche Sprintziele, feste Sprintdauer, Nutzung von User Stories und Ihre Aufspaltung in kleine, einfache Aufgaben sowie erzeugte Transparenz ausgewirkt. Auch in diesem Fall wurden die volle Unterstützung durch beide Gründer und die positive Einstellung der Mitarbeiter gegenüber Scrum als wichtige Faktoren für die erfolgreiche Umsetzung identifiziert. Beiden Marketingteams schafften es erfolgreich ein, an Ihre Bedürfnisse angepasstes, Vorgehen innerhalb des Scrum-Rahmenwerks zu entwickeln.

---

<sup>73</sup> Vgl. Verwijns: „Scrum for Marketing Teams: A Case Study“, 2016 (online).

Kritische Betrachtung: die überwiegend positive Berichtserstattung des Scrum-Beraters Christiaan Verwijs kann durch sein geschäftliches Interesse begründet sein. Ein Hinweis am Ende des Artikels lässt jedoch erkennen, dass diese Fallstudie ebenfalls auf Erfahrungen und Kommentaren einer Kaarten Carrousel Mitarbeiterin basiert. Daher darf von der Ausgewogenheit dieses Berichtes ausgegangen werden.

### 3.4 Scrum bei Swiss Property AG

Auf der Creative Construction Conference 2016 (CCC ist eine kreative Baukonferenz für Forscher und Praktiker aus dem Baugewerbe<sup>74</sup>) wurde eine Fallstudie über die Umsetzung von Scrum in einem Bauprojekt der Swiss Property AG vorgestellt. Die fünf Autoren begleiteten acht Wochen lang ein Scrum-Projektteam und stellten Ihre Ergebnisse in einem Konferenzpapier vor. Die Ausgangssituation: die Erkenntnis über sich ständig verändernde Bedingungen im Umfeld von Bauprojekten hat sich bei vielen Bauunternehmen durchgesetzt. Dem gegenüber stehen seit Jahrzehnten unveränderte Projektmanagementmodelle und verhindern eine effiziente und zeitgemäße Steuerung von Bauprojekten. Immer mehr Bauunternehmen nutzen daher die Erfahrungen aus anderen Branchen und setzen auf neue Methoden. Swiss Property AG entschied sich dazu, Scrum in einem laufenden Projekt auszuprobieren. Das Ziel war der Bau von drei Mehrfamilienhäusern mit einer Gesamtfläche von 2100m<sup>2</sup>, verteilt auf elf Wohnungen und eine Gewerbefläche. Das Projekt bestand insgesamt aus sechs übergeordneten Phasen: strategische Planung, Vorstudie, Projekt, Bauausschreibung, Umsetzung und Verwaltung. Da die erste Phase bereits abgeschlossen war, wurde entschieden Scrum in den Phasen zwei und drei (Vorstudie und Projekt) zu implementieren. Die zweite Phase beinhaltet Projektdefinition, eine Machbarkeitsstudie und die Auswahl des am besten geeigneten Projektes. Die dritte Phase beinhaltet die Optimierungen des Konzepts, der Wirtschaftlichkeit, des Projektes und der Kosten. Am Ende der dritten Phase sollten alle notwendigen Arbeiten zur Beantragung der Baugenehmigung erledigt sein.

Bei der Einführung wurde entschieden alle Scrum-Elemente zu übernehmen und sie möglichst unverändert zu lassen. Es wurden alle drei Rollen besetzt, im Projektverlauf alle Ereignisse durchgeführt und alle Artefakte verwendet. Ein Product Owner und ein Scrum Master wurden bestimmt und das Entwicklungsteam mit Experten aus verschiedenen Bereichen des

---

<sup>74</sup> Vgl. Skibniewski et al.: „Creative Construction Conference 2016: An invitation“, 2016 (online).

Baugewerbes besetzt. Das siebenköpfige Expertenteam deckte mehrere wichtige Fachbereiche des Bauwesens ab. Es gab drei Architekten, einen Bauphysiker, einen Bauingenieur, einen Kostenschätzer und einen Innenarchitekten.

Das Scrum-Team wurde angewiesen möglichst an allen Ereignissen teilzunehmen. Aufgrund fehlender Erfahrung im agilen Projektmanagement, wurde für die Erreichung der Baugenehmigung zunächst ein sehr optimistischer Zeitraum von vier Wochen festgelegt. Nach den ersten zwei Wochen erkannte das Scrum-Team, dass die geplanten vier Wochen keinesfalls ausreichen werden. Daher wurde entschieden, die Dauer der Phasen zwei und drei von 4 auf insgesamt 15 Wochen auszuweiten. Damit sollte sichergestellt werden, dass die Baugenehmigung mit einer minimalen Anzahl von Beschränkungen und Einwänden seitens der Behörden erteilt werden kann. Die erste Sprintlänge betrug fünf Arbeitstage. Am Montag fand das Sprint Planning statt, Dienstag bis Donnerstag wurden die geplanten Aufgaben erledigt und dabei jeden Tag Daily Scrums abgehalten. Am Freitag fanden abschließend die Sprint Review und Retrospektive statt. Nach vier Wochen erkannte das Scrum-Team, dass die fünftägige Sprintlänge viel zu knapp bemessen wurde, um alle notwendigen Aufgaben zu erledigen und zusätzlich alle Scrum-Ereignisse durchzuführen. Die neue Sprintlänge wurde daher auf zwei Wochen festgelegt. Die Erstellung und Pflege des Product Backlogs wurden durch den Scrum Master und einen Architekten aus dem Entwicklungsteam durchgeführt, da der Product Owner bereits mit anderen Aufgaben stark ausgelastet war. Um die befürchtete Verzögerung bei der Erstellung der Product Backlog Items zu vermeiden, entschied sich das Scrum-Team zu diesem Schritt. Der Sprintfortschritt und anfallende Aufgaben wurden mithilfe einer Pinnwand als auch einer speziellen Software visuell dargestellt. Der Product Owner und die Autoren des Konferenzpapiers konnten somit immer einen Blick auf die aktuelle Situation werfen, auch wenn sie sich nicht im Raum des Entwicklungsteams befanden. Die Schätzung der Items fanden während der fünf ersten Sprints ausschließlich auf Basis der Erfahrungswerte der Teammitglieder statt und folgten keinem System. Ab dem fünften Sprint nutzte das Team „Planning Poker“ um eine höhere Schätzungsgenauigkeit zu erreichen. Alle Daily Scrums beschränkten sich hauptsächlich auf die Beantwortung von drei Kernfragen durch jedes Teammitglied. Was habe ich gestern erledigt? Was erledige ich heute? Und was hindert mich daran, meine Aufgaben zu erledigen? Im Anschluss bewerteten alle Teammitglieder das aktuelle Daily Scrum mit einem extra dafür entwickelten Formular. Die nachfolgend aufgeführten Gesamtergebnisse der Studie bestehen aus drei Komponenten. Erstens: aus den im Projektverlauf gesammelten Aufzeichnungen der Autoren. Zweitens: aus

Interviews, die am Ende der acht Wochen mit sieben Teammitgliedern und dem Scrum Master durchgeführt wurden. Drittens: aus der kritischen Auswertung der Product Backlog Items.

**Scrum-Einführung** – die sieben Teammitglieder und der Scrum Master hatten nur wenig Kenntnisse über Scrum vor der Einführung und bezeichneten Ihre Eindrücke von der Umsetzung zunächst nur als durchschnittlich gut. Sie verstanden jedoch die Relevanz und Notwendigkeit von Scrum für Ihr Projekt.

**Positive Wahrnehmung** – nach fünf Wochen war das Scrum-Team überzeugt, dass Scrum deutlich effizienter als Ihre alten Ansätze und Methoden ist. Das Team stellte eine höhere Transparenz, bessere Kommunikation und Zusammenarbeit der Teammitglieder, einen besseren Informationsfluss und schnellere Projektfortschritte fest.

**Daily Scrums** – verbesserten sich mit der Zeit und stifteten immer mehr Nutzen. Dies wurde hauptsächlich durch die permanente Beantwortung der drei Kernfragen mit einhergehenden Verbesserungen begründet. Das Team gewann ein besseres Verständnis über den gesamten Prozess, berichtete über erledigte Aufgaben und plante effizient die anstehenden Arbeiten. Gleichzeitig konnten auch die Hindernisse besser identifiziert und genannt werden. Die Länge der Daily Scrums wurde als angemessen wahrgenommen.

**Das Team** – die vielen gemeinsamen Meetings ermöglichten den Teammitgliedern, die Sichtweise Ihrer Kollegen zu verstehen, und warum bestimmte Aufgaben auf bestimmte Art und Weise erledigt wurden. Dies führte wiederum zum Wissenszuwachs der Teammitglieder in den für sie unbekanntem Fachgebieten und förderte die crossfunktionale Teamorganisation. Besonders nützlich erwies sich die visuelle Darstellung der Projektfortschrittes mithilfe der Pinnwand, die jederzeit einen Blick auf alle wichtigen Informationen ermöglichte.

**Fehlende Schulung und Projektleiter** – problematisch wurden die fehlenden Kenntnisse bei der Einführung bezeichnet und einige Teammitglieder fanden es sehr ungewöhnlich, ohne einen Projektleiter zu arbeiten. Der Aufwand für die Abstimmungen innerhalb des Teams war sehr hoch, da es aufgrund des fehlenden Projektleiters auch die Verantwortung trug.

**Product Backlog Items** – wurden in größere Aufgaben aufgeteilt, die nicht innerhalb eines Tages erledigt werden konnten. Daher gab es nur wenige Teammitglieder, die in Daily Scrums oft über vollständig erledigte Aufgaben berichten konnten.

**Dauer und Effektivität einiger Ereignisse** – die Dauer von Sprint Planning und Sprint Retrospektive wurde als etwas zu lang wahrgenommen und die der Sprint Review als zu kurz. Die Effektivität von Planning und Review bekamen leicht überdurchschnittliche Bewertungen, und die Retrospektive wurde nur durchschnittlich bewertet.

Zu den Auswirkungen von Scrum bei Swiss Property AG gehören:

- Verbesserte Zusammenarbeit und Kommunikation des Teams.
- Eine höhere Transparenz und besserer Informationsfluss in Bezug auf den Projektstatus.
- Effizientere Planung.
- Wesentliche Beschleunigung der Projektfortschritte.<sup>75</sup>

**Auswertung:** das Scrum-Team von Swiss Property setzte von Anfang an das Scrum-Rahmenwerk als Ganzes und gemäß dem Scrum-Guide um. Den Beteiligten gelang es alle Rollen, Artefakte und Ereignisse ohne wesentliche Anpassungen zu übernehmen.

Nicht optimal verlief die Umsetzung der Rolle des Product Owners aufgrund seiner Auslastung. Einige Teammitglieder mussten zwangsweise einen Teil seiner Aufgaben übernehmen, wobei diese Praktik im Scrum-Guide erlaubt ist. Die beklagten fehlenden Scrum-Kenntnisse resultierten in der Zerlegung der Items in größere Aufgaben, die selten innerhalb eines Tages erledigt wurden. Das Fehlen eines Projektleiters und die damit verbundene Übertragung der Verantwortung auf alle Teammitglieder führten zu Unsicherheiten und aufwendigen Abstimmungen innerhalb des Teams. Die nicht immer optimale Dauer einiger Ereignisse ist dagegen nicht ungewöhnlich für die ersten Sprints.

Die tadellos funktionierenden Daily Scrums beschränkten sich hauptsächlich auf die Beantwortung der Kernfragen und verliefen optimal in Bezug auf Ihre Dauer und erzeugten Nutzen. Die Visualisierung des Projektfortschritts war sehr wirkungsvoll und verhalf dem Team das Wesentliche im Blick zu behalten. Auch die Teilnahme des ganzen Scrum-Team an allen Ereignissen verlief problemlos und ermöglichte ein besseres Verständnis sowie Wissenszuwachs innerhalb des Teams. Das Swiss-Property-Team schaffte es als einziges, Scrum entsprechend dem Scrum-Guide und ohne Anpassungen anzuwenden.

---

<sup>75</sup> Vgl. Streule et al., 2016.

## 4 Gesamtergebnisse

Nach der Analyse der einzelnen Fallstudien sollen ihre Ergebnisse zusammengeführt werden. Damit werden die Gemeinsamkeiten und Unterschiede in der Anwendung von Scrum sowie bei seinen Auswirkungen auf das Projektmanagement quer über alle Erfahrungsberichte sichtbar. Die Erkenntnisse daraus erlauben eine Schlussfolgerung über die praktische Anwendbarkeit von Scrum und Beantwortung der Forschungsfragen.

### *Die Rollen:*

**Product Owner** – diese Rolle fand ihre Anwendung in allen untersuchten Fällen, aber nur bei drei Unternehmen entsprechend dem Scrum-Guide. Die Übernahme einiger Aufgaben des Product Owners durch Teammitglieder bei Swiss Property und Kaarten Carrousel gehört zu erlaubten Praktiken. Bei Intel dagegen wurde diese Rolle wesentlich umfunktioniert und verursachte zeitweilig Störungen in den Sprints.

**Scrum Master** – kam im Endergebnis nur in drei Fällen vor. TablonautiX verzichtete auf diese Rolle, da sie als Ursache für Hindernisse wahrgenommen wurde. Damit blieb es das einzige Unternehmen ohne einen Scrum Master.

**Das Team** – gab es verständlicherweise in allen Fällen und zwar in der empfohlenen Teamgröße. Bis auf Intel setzten alle Unternehmen von Anfang an auf crossfunktionale Teams. Intels verzögerter Umstieg führte mehrmals zu Störungen und verhinderte die frühere Entfaltung positiver Auswirkungen. Schlussendlich kamen crossfunktionale Teams in allen Fällen zum Einsatz und erwiesen sich als sehr vorteilhaft.

### *Die Artefakte:*

**Product Backlog** – wurde in allen Fallstudien gemäß dem Scrum-Guide verwendet. Bei Intel wurden lediglich sein Umfang und Kontrollierbarkeit bemängelt, was durch die Fallspezifik begründet war und relativ zeitnah gelöst werden konnte.

**Sprint Backlog** – fand in allen vier Fällen seine entsprechende Anwendung. Mit Ausnahme der umfangspezifischen Besonderheit bei Kaarten Carrousel, wirkte sich dieses Artefakt äußerst positiv und förderlich auf die Scrum-Prozesse aus. Besonders die visuelle Darstellung einzelner Aufgaben und des Fortschrittes erwiesen sich in allen Fällen sehr förderlich.

**Produktinkrement** – wurde eher indirekt und im Zusammenhang mit dem Sprintziel oder der Review erwähnt, jedoch immer im positiven Kontext. In allen Fällen wurden Teilergebnisse geliefert, aber eine detaillierte Beschreibung dazu fehlte in allen untersuchten Fallstudien.

#### ***Die Ereignisse:***

**Sprint Planning** – fand in allen Fallstudien statt. Bei TablonautiX erfuhr dieses Ereignis eine starke Modifizierung und bei Kaarten Carrousel eine wesentliche Reduzierung seiner Dauer. In der Summe wurde Sprint Planning nur in zwei von vier Fällen entsprechend dem Scrum-Guide angewandt.

**Sprint** – als die eigentliche Arbeitsphase war selbstverständlich in allen Fallstudien präsent. Die Ermittlung der passenden Sprintlänge brauchte in manchen Fällen etwas Zeit, das ist aber nicht ungewöhnlich. TablonautiX ließ Änderungen während der Sprints zu, passte ihre Dauer an und verfehlte teilweise die Sprintziele. Im Ergebnis wurde dieses Ereignis daher nur in drei Fällen entsprechend dem Scrum-Guide umgesetzt.

**Daily Scrum** – fand ebenfalls in allen untersuchten Fällen statt. Dieses Ereignis wurde als besonders nutzenstiftend für die Aktualisierung des Sprintfortschritts und eine effizientere Planung beschrieben.

**Sprint Review** – wurde zwar in allen Fallstudien durchgeführt, gestaltete sich allerdings nicht überall einfach. TablonautiX führte bedarfsabhängige Zwischenreviews ein und ließ nur noch ein Teammitglied an der eigentlichen Review mit dem Auftraggeber teilnehmen. Damit schaffte es TablonautiX nicht, Sprint Review gemäß dem Scrum-Guide zu nutzen. Leichte, jedoch zulässige Anpassungen gab es in zwei Fällen. Intel führte Zwischen-Reviews vor den eigentlichen Reviews ein, hielt sich aber an feste Termine und Kaarten Carrousel konzentrierte sich in der Review auf größere Items. In der Summe schafften es drei von vier Unternehmen die Sprint Review entsprechend dem Scrum-Guide anzuwenden.

**Sprint Retrospektive** – ist ein Ereignis, das in allen untersuchten Fällen durchgeführt, aber nicht näher kommuniziert wurde. Da die Retrospektive in keinem negativen Kontext erwähnt wurde, darf von einer relativ einfachen Adoption dieses Ereignisses in der Praxis ausgegangen werden.

**Positive Auswirkungen** – obwohl es zwischen den einzelnen Fallstudien deutliche Unterschiede in Bezug auf Branchen und Anwendungen gab, konnten neben individuellen auch einige übereinstimmende Verbesserungen nach der Scrum-Einführung ermittelt werden:



- Die grundsätzliche Wahrnehmung des Scrum-Rahmenwerks war in allen Fällen positiv. Alle genannten Unternehmen entschieden sich dafür, ihre neuen Vorgehensweisen zu behalten.
- In allen Fällen verbesserten sich die Zusammenarbeit und Kommunikation in den Teams.
- In drei Fällen gab es schnellere Projektfortschritte, entweder in Form von regelmäßigen Lieferungen von Teilergebnissen oder in verkürzter Abwicklung der Projekte.
- Ebenfalls in drei Fällen erhöhte sich Transparenz der Projekte.
- In zwei Fällen wurde eine höhere Effizienz im Projektverlauf festgestellt.
- In Einzelfällen ergaben sich verbesserte Zusammenarbeit zwischen den Teams, deutlich flexiblere Reaktionen auf Anforderungen, weniger Planabweichungen, zeitnahe Erledigung komplizierter Aufgaben, zielgerichtetes Arbeiten und prozesskonformes Verhalten der Teammitglieder.

Grundsätzlich konnten in allen Fällen Verbesserungen bei der Projektabwicklung festgestellt werden. Die Zusammenfassung und Auflistung dieser Ergebnisse beantwortet die Teilfrage bezüglich der positiven Auswirkungen, die sich nach der Einführung und Nutzung von Scrum ergeben.

Zu den positiven Aspekten gehören neben den Vorteilen auch einige allgemeine Faktoren, die sich als besonders förderlich bei der Scrum-Einführung erwiesen haben. Dazu gehören Unterstützung der agilen Methoden durch die Führungsebene, persönliches Engagement von einzelnen Mitarbeitern mit einer offenen Haltung gegenüber Veränderungen und das Beauftragen von externen Scrum-Beratern.

Im Gegensatz zu mehrfach ermittelten gemeinsamen Vorteilen wurde lediglich ein vergleichsweise schwierig umzusetzendes Scrum-Element ermittelt. Sprint Planning konnte in nur zwei Fällen entsprechend dem Scrum-Guide angewandt werden. In zwei weiteren Fällen kam es zu deutlichen Abweichungen. Von den übrigen einmalig aufgetretenen Problemen, sind gleich Drei auf eine einzige Fallstudie zurückzuführen, in der die Umsetzung nach dem Scrum-Guide gar nicht gelungen ist. Die restlichen Schwierigkeiten tragen teilweise einen vorübergehenden, nicht nachhaltigen Charakter oder sind durch die Fallspezifik begründet.

## 5 Fazit

Das Ziel dieser Arbeit war es, zu untersuchen, wie gut das Scrum-Rahmenwerk in der Praxis funktioniert. Auf Basis der gewonnenen Erkenntnisse, lässt sich die Forschungsfrage bezüglich der Anwendbarkeit von Scrum zum Managen von Projekten wie folgt beantworten: die agile Methode Scrum funktioniert in der Praxis größtenteils und führt zu wesentlichen Verbesserungen bei der Projektabwicklung. Ihre Anwendung ist nicht einfach und gelingt nicht immer im Einklang mit dem Scrum-Guide. Projektspezifische Anpassungen sind möglich und sinnvoll und mindern nicht die positiven Auswirkungen. Auch wenn nicht alle Scrum-Elemente adoptiert werden, können sich spürbare Verbesserungen ergeben. Die aus der teilweisen Anwendung entstehenden Vorgehensweisen sind agil und berücksichtigen die veränderten Bedingungen des Projektumfeldes.

Die Teilfrage, ob Scrum als Ganzes oder nur teilweise umgesetzt wird, lässt sich wie folgt beantworten: in den meisten Fällen kommt tatsächlich das ganze Scrum-Rahmenwerk zum Einsatz. Die vollständige Adoption von Scrum in der Praxis gelang in drei von vier untersuchten Fällen. In zwei davon mit jeweils einem angepassten Element und im Dritten vollständig entsprechend dem Scrum-Guide. In der vierten Fallstudie erfuhren einige Scrum-Elemente gravierende Modifizierungen oder wurden abgeschafft, so dass das Ergebnis zwar als agil und auf Scrum basierend, jedoch nicht mehr als Scrum bezeichnet werden kann. Das Überraschende dabei ist, dass ausgerechnet bei der Softwareentwicklung die Umsetzung nach dem Scrum-Guide nicht funktioniert hat. Und das, obwohl das Scrum-Rahmenwerk eigens für die Softwareprojekte entwickelt wurde.

Die Anzahl der ermittelten Problemen bei der Adoption von Scrum ist in Relation zu den Vorteilen eher gering. Die Antwort auf die Teilfrage bezüglich der Schwierigkeiten bei der Umsetzung lässt sich daher wie folgt beantworten: die Ergebnisse der Untersuchung zeigen, dass besonders die Rollen *Product Owner* und *Scrum Master* sowie Ereignisse *Sprint*, *Sprint Planning* und *Sprint Review* in der praktischen Umsetzung mit einem höheren Aufwand verbunden waren. Davon kann nur beim Ereignis *Sprint Planning* von einer erhöhten Komplexität ausgegangen werden. Alle anderen Schwierigkeiten trugen eher einen fallspezifischen und einmaligen Charakter. Alle übrigen Rollen, Artefakte und Ereignisse ließen sich dagegen verhältnismäßig einfach für die Projektsteuerung adoptieren.

Die allgemeine Beschreibung von Scrum erwähnt den Menschen im Mittelpunkt und Erfolg durch Interaktionen von Individuen. In allen untersuchten Fällen wurden deutliche

Verbesserungen sowohl bei der Zusammenarbeit speziell als auch beim Projektmanagement allgemein festgestellt. Diese Ergebnisse bestätigen den „Menschen im Mittelpunkt-Ansatz“ und seine Wirksamkeit. Die präsentierten Fallstudien zeigen außerdem, dass Scrum für unterschiedlichste Anwendungen geeignet ist: Entwicklung von Soft- und Hardware, Online-Marketing oder Baubranche, überall kann Scrum adoptiert werden. Wie gut die agile Transformation gelingt, hängt von vielen Faktoren ab. Die Organisationsgröße und -struktur, etablierte Prozesse, Projektumfang und Vorkenntnisse haben einen direkten Einfluss auf die Umsetzung. So kämpfen riesige Konzerne mit ihrer Größe, existierenden Strukturen und Skalierungseffekten. Während junge und schnell wachsende Unternehmen von der Vielfalt der Aufgaben erdrückt werden und ihre Arbeit erst vernünftig organisieren und strukturieren müssen. Dementsprechend sieht auch die Adoption des Scrum-Rahmenwerks jeweils anders aus. Die Ergebnisse dieser Arbeit führen zu einem überwiegend positiven Gesamteindruck von der Anwendbarkeit und Wirksamkeit von Scrum in der Praxis. Dabei soll nicht unerwähnt bleiben, dass die gewonnenen Erkenntnisse lediglich auf vier Fallstudien basieren und deswegen in ihrer Aussagekraft eingeschränkt sind. Nichtsdestotrotz bieten diese Ergebnisse einen praxisnahen Blick auf das Scrum-Rahmenwerk und können zur Orientierung bei der Scrum-Anwendung herangezogen werden.

Scrum, wie auch andere agile Methoden, soll die Durchführung von Projekten in komplexen Umgebungen erleichtern und ein anforderungsgerechtes Projektmanagement ermöglichen. Komplexität im Projektumfeld resultiert aus der Entwicklung bestimmter Trends. Es darf davon ausgegangen werden, dass diese Trends sich in der Zukunft noch weiter verstärken und auf immer mehr Branchen ausbreiten werden. Daneben werden auch die Projekte und damit das Projektmanagement eine weiterhin wichtige Rolle in der unternehmerischen Praxis spielen. Somit ergibt sich, dass einerseits die Komplexität der Projektumgebungen weiter steigen wird, andererseits das Projektmanagement aus dem Unternehmensalltag nicht mehr wegzudenken ist. Dieser Sachverhalt verdeutlicht die Wichtigkeit und Relevanz agiler Vorgehensweisen für die Zukunft. Die Anwendung agiler Methoden würde viel einfacher gelingen, wenn involvierte Akteure auf vorhandenes Erfahrungswissen zurückgreifen könnten. Der Aufbau dieses Wissens würde unter anderem durch weitere und vor allem umfangreichere sowie repräsentative Studien aus der agilen Praxis wesentlich begünstigt werden. Davon könnten dann viele Unternehmen weltweit profitieren und ihre Projekte trotz komplexer Umgebungen erfolgreich zum Abschluss führen.

## Literaturverzeichnis

**Bartonitz, Martin et al.:** Agile Verwaltung: Wie der Öffentliche Dienst aus der Gegenwart die Zukunft entwickeln kann (Hrsg.: Wagner, Ludger), Berlin (Springer Gabler), 2018.

**Bause, Katharina:** Vorgehensweisen, Möglichkeiten und Risiken in der Umstellung zum agilen Projektmanagement, 2018. (verfügbar unter: <https://www.sosy-lab.org/Teaching/2017-WS-JurPM/>).

**Beck, Kent et al.:** „Manifest für Agile Softwareentwicklung“, 2001  
<https://agilemanifesto.org/iso/de/manifesto.html>, (01. Feb. 2019).

**borisgloger.com:** „Boris Gloger“, 2019 <https://www.borisgloger.com/team/boris-gloger/> (11. Feb. 2019).

**Brandstätter, Jonathan:** Agile IT-Projekte erfolgreich gestalten: Risikomanagement als Ergänzung zu Scrum, Wiesbaden (Springer Fachmedien Wiesbaden), 2013.

**Canditt, Sabine:** Komplexität begegnen: Nutzen Sie agile Ansätze auch außerhalb der IT, in: ProjektMagazin (16/2018).

**Danube Technologies:** „Danube Technologies | LinkedIn“, 2019  
<https://de.linkedin.com/company/danube-technologies> (17. Feb. 2019).

**Deutsches Institut für Normung:** Projektmanagement. Netzplantechnik und Projektmanagementsysteme, Stand der abgedruckten Normen: September 2016 Berlin, Wien, Zürich (Beuth), 2016.

**Dräther, Rolf; Koschek, Holger; Sahling, Carsten:** Scrum: kurz & gut, Beijing, Cambridge, Farnham, Köln, Sebastopol, Tokyo (O'Reilly), 2013.

**Elwer, Patrick et al.:** Agile Project Development at Intel: A Scrum Odyssey, 2008. (verfügbar unter: <http://www.scrumcasestudies.com/intel/>).

**Fischer, Benjamin:** Rugby für das Büro, in: Frankfurter Allgemeine Zeitung vom 07./08. April 2018, Frankfurt.

**Fischer-Korp, Christa:** Erfolgreiche Change-Prozesse im öffentlichen Bereich: Strategien, Methoden und Tools, Wiesbaden (Springer Gabler), 2018.

**Gabler Wirtschaftslexikon:** „Agilität“, 2019 <https://wirtschaftslexikon.gabler.de/definition/agilitaet-99882> (01. Feb. 2019).

**Gehr, Simone et al.:** Systemische Werkzeuge für erfolgreiches Projektmanagement: Konzepte, Methoden, Fallbeispiele, Wiesbaden (Springer Fachmedien Wiesbaden), 2018.

- Gloger, Boris:** Scrum: Produkte zuverlässig und schnell entwickeln, München (Hanser), 2013.
- Hagen, Stefan:** „Evolutionsphasen im Projektmanagement“, 2008 [https://pm-blog.com/2008/04/26/evolutionsphasen\\_projektmanagement/](https://pm-blog.com/2008/04/26/evolutionsphasen_projektmanagement/) (29. Dez. 2018).
- Hanschke, Inge:** Agile in der Unternehmenspraxis: Fallstricke erkennen und vermeiden, Potenziale heben, Wiesbaden (Springer Vieweg), 2017.
- Hofmann, Florian:** „Ist das noch Scrum oder kann das weg?“, 2016 <https://www.produktbezogen.de/ist-das-noch-scrum-oder-kann-das-weg/> (04. Jan. 2019).
- Kusay-Merkle, Ursula:** Agiles Projektmanagement im Berufsalltag: Für mittlere und kleine Projekte, Berlin, Heidelberg (Springer Berlin Heidelberg), 2018.
- Layton, Mark C.; Ostermiller, Steven J.:** Agiles Projektmanagement für Dummies, Weinheim (Wiley-VCH), 2018.
- Meyer, Helga; Reher, Heinz-Josef:** Projektmanagement: Von der Definition über die Projektplanung zum erfolgreichen Abschluss, Wiesbaden (Springer Gabler), 2016.
- Pichler, Roman:** Scrum – agiles Projektmanagement erfolgreich einsetzen, Heidelberg (dpunkt.verlag), 2009.
- Preußig, Jörg:** Agiles Projektmanagement: Agilität und Scrum im klassischen Projektumfeld, Freiburg, München, Stuttgart (Haufe Gruppe), 2018.
- Roock, Stefan; Wolf, Henning (2018):** Scrum – verstehen und erfolgreich einsetzen, Heidelberg (dpunkt.verlag), 2018.
- Röpstorff, Sven; Wiechmann, Robert:** Scrum in der Praxis: Erfahrungen, Problemfelder und Erfolgsfaktoren, Heidelberg: (dpunkt.verlag), 2016.
- Rubin, Kenneth S.:** Essential Scrum: umfassendes Scrum-Wissen aus der Praxis, Heidelberg, Hamburg (MITP Verlags GmbH), 2014.
- Sharma, Jacqueline:** Scrum und das Standardmodell wirksamen Managements nach Malik: Eine Synthese systemischen Managements, Stuttgart (Ibidem-Verlag), 2012.
- Simschek, Roman; Kaiser, Fabian:** SCRUM: das Erfolgsphänomen einfach erklärt, Konstanz, München (UVK Verlagsgesellschaft mbH), 2018.
- Skibniewski, Mirosław; Hajdu, Miklós; Mályusz, Levente:** „Creative Construction Conference 2016: An Invitation“, 2016 <http://2016.creative-construction-conference.com/> (25. Feb. 2019).

**Streule, Thomas et al.:** Implementation of Scrum in the Construction Industry, in: Procedia Engineering, 164 (2016), S. 269–276. (verfügbar unter: <https://linkinghub.elsevier.com/retrieve/pii/S1877705816339601>).

**Sutherland, Ken; Schwaber, Jeff:** Der Scrum-Guide, 2017. (verfügbar unter: <https://www.scrumguides.org/download.html>).

**Timinger, Holger:** Modernes Projektmanagement: mit traditionellem, agilem und hybridem Vorgehen zum Erfolg, Weinheim (Wiley), 2017.

**Verwijs, Christiaan:** „Scrum for Marketing Teams: A Case Study“, 2016 <https://medium.com/the-liberators/scrum-for-marketing-teams-a-case-study-b650b31d9ab8>, (14. Feb. 2019).

**Verwijs, Christiaan:** „Professional Scrum Master“, 2019 <https://www.eventbrite.co.uk/e/43392216287?aff=efbneb>, (14. Feb. 2019).

**Wintersteiger, Andreas:** Scrum: Schnelleinstieg, Frankfurt am Main (entwickler.press), 2012.

**Wytrzens, Hans Karl:** Projektmanagement: Der erfolgreiche Einstieg, Wien (Facultas), 2017.

### **Eidesstaatliche Erklärung**

Ich versichere, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quelle kenntlich gemacht.

Ich erkläre mich damit

- einverstanden,
- nicht einverstanden

dass ein Exemplar meiner Bachelorthesis in die Bibliothek des Fachbereichs aufgenommen wird; Rechte Dritter werden dadurch nicht verletzt.

Hamburg, den .....(Unterschrift der/des Studierenden)