

MASTERARBEIT

PROTOTYP ZUR KENNZAHLENBASIERTEN STEUERUNG VON AGILEN PROJEKTEN IN SOFTWAREUNTERNEHMEN MITTELS EARNED VALUE MANAGEMENT

ausgeführt am



Studiengang

Informationstechnologien und Wirtschaftsinformatik

Von: Markus Baumgartner
Personenkennzeichen: 1810320027

Graz, am 30. Juni 2022

.....
Unterschrift

EHRENWÖRTLICHE ERKLÄRUNG

Ich erkläre ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benützt und die benutzten Quellen wörtlich zitiert sowie inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

.....

Unterschrift

DANKSAGUNG

An dieser Stelle möchte ich mich bei all jenen Personen bedanken, welche mich bei der Erstellung dieser Masterarbeit unterstützt haben. Besonderer Dank gilt meinem Betreuer, Herrn DI Dr. Thomas Puchleitner, der durch sein konstruktives Feedback eine große Hilfe war. Darüber hinaus möchte ich mich bei meiner Lebensgefährtin Cynthia für die Unterstützung während des gesamten Studiums und auch für das mehrmalige Korrekturlesen der Arbeit bedanken. Großer Dank gilt auch meiner Mutter Ingrid, meinem Bruder Bernhard, meiner Schwägerin Susanne und meinen Freunden Armin und Benjamin, da sie stets Verständnis für meine eingeschränkte Verfügbarkeit hatten und motivierend zur Seite standen. Abschließend möchte ich allen Expertinnen und Experten meinen Dank aussprechen, welche sich die Zeit für die Interviews genommen und den Abschluss dieser Arbeit durch ihren fachkundigen Input möglich gemacht haben.

KURZFASSUNG

Agile Teams sind dafür verantwortlich, den Fortschritt von Entwicklungsleistungen zu berichten. Für Unternehmen in der Softwareentwicklungsbranche ist jedoch häufig auch die Verfolgung von Arbeitszeiten von Bedeutung. Neben oftmals regulatorischen Vorgaben zur Aufzeichnung von Dienstzeiten kann die Dokumentation von Arbeitszeiten auch Aufschluss über die Entwicklungskosten geben und zur Steuerung von Projekten genutzt werden. Mit dem Projektsteuerungsinstrument Earned Value Management können zur Laufzeit projektbezogene Kennzahlen kalkuliert werden, welche auf dem Fortschritt und den Kosten zum betrachteten Zeitpunkt basieren.

Die vorliegende Arbeit befasst sich mit der Untersuchung von Earned Value Management in agilen Softwareunternehmen. Dabei soll festgestellt werden, inwieweit das Instrument einen positiven Effekt auf die Steuerung von Projekten, welche mit agilen Methoden geführt werden, hat. Darüber hinaus sollen die Rahmenbedingungen für einen Einsatz ermittelt werden. Im theoretischen Teil dieser Arbeit wird mit der Beschreibung der agilen Grundsätze sowie von Vorgehensmodellen und Frameworks eine Einführung in die agile Softwareentwicklung gegeben. Darüber hinaus wird im Theorieteil das Thema Projektsteuerung behandelt und das Instrument Earned Value Management näher erläutert.

Die Vorgehensweise dieser Arbeit orientiert sich an das Design Science Research Paradigma, welches die Entwicklung und Evaluierung eines Artefaktes vorsehen. Basierend auf der theoretischen Auseinandersetzung mit facheinschlägiger Literatur wurde im empirischen Teil dieser Arbeit ein Artefakt in Form eines Prototyps erstellt, welcher die kennzahlenbasierte Steuerung von agilen Softwareprojekten mittels Earned Value Management ermöglicht. Zur Evaluierung des Prototyps wurden fünf Expertinnen und Experten aus unterschiedlichen Softwareunternehmen befragt. Aus den Interviews konnten Ergebnisse im Hinblick auf mögliche Anwendungsgebiete, Restriktionen und Erweiterungsmöglichkeiten abgeleitet werden. Die Ergebnisse haben gezeigt, dass Earned Value Management vor allem in Unternehmen mit geringer Teamanzahl und hoher Planungssicherheit eingesetzt werden kann. Kurzfristige Änderungen des Projektplans führen jedoch dazu, dass die Vorhersagen des Modells an Aussagekraft verlieren.

ABSTRACT

In the software development industry, agile teams typically need to report on the progress of a project. The often legally required recording of working hours can also provide information about development costs and can be used to control projects. Earned Value Management is a project management tool for calculating project-related key figures and creating forecasts at runtime. It is based on progress and cost-related data that teams collect in their daily business.

The aim of this work is to determine the extent to which the instrument has a positive effect on the management of projects that are run using agile methods. For this purpose, the project management tool Earned Value Management, and its application areas in agile environments are examined in more detail. The theoretical part of this thesis provides an introduction to agile software development with a description of the agile principles and selected process models. Furthermore, the theoretical part deals with the topic of project control with its essential tasks and reporting methods.

The procedure of this work is based on a design-scientific approach that aims at developing and evaluating an artefact. The artefact created in the context of this work is a prototype that uses the methods of Earned Value Management to enable the management of agile projects based on progress and cost-related data. Five experts from different software companies were interviewed to evaluate the prototype. From those interviews, areas of application, restrictions and expansion could be derived. The results have shown that Earned Value Management can be used primarily in companies with a small number of teams and a high level of planning security. Furthermore, short-term changes in the project plan cause the model's predictions to lose validity.

INHALTSVERZEICHNIS

1	EINLEITUNG.....	1
1.1	Problemstellung.....	1
1.2	Zielsetzung.....	2
1.3	Vorgehensweise, Methodik und Aufbau	2
2	EINFÜHRUNG IN DIE AGILE SOFTWAREENTWICKLUNG.....	4
2.1	Grundlagen	4
2.1.1	Definition von Agilität.....	4
2.1.2	Agile Werte	5
2.1.3	Agile Prinzipien	6
2.2	Vorgehensmodelle.....	6
2.2.1	Traditionelle Modelle	6
2.2.2	Agile Modelle	8
2.2.3	Hybride Modelle	10
2.3	Scrum.....	12
2.3.1	Entscheidung für Scrum.....	12
2.3.2	Definition.....	13
2.3.3	Rollen	14
2.3.4	Events	16
2.3.5	Artefakte.....	18
2.4	Zusammenfassung.....	19
3	KENNZAHLENBASIERTE STEUERUNG VON PROJEKTEN.....	21
3.1	Steuerung von Softwareprojekten.....	21
3.1.1	Planung	22
3.1.2	Schätzung.....	23
3.1.3	Budgetierung	26
3.1.4	Monitoring.....	27
3.1.5	Forecasts.....	29
3.2	Berichte, Metriken und Kennzahlen.....	31
3.2.1	Velocity.....	32

3.2.2	Burndown- und Burnup-Diagramme	33
3.2.3	Lead Time und Cycle Time.....	35
3.2.4	Cumulative Flow	36
3.2.5	Finanzielle Kennzahlen	37
3.3	Earned Value Management	38
3.3.1	Entscheidung für Earned Value Management.....	38
3.3.2	Definition und Abgrenzung	39
3.3.3	Einsatzgebiete	40
3.3.4	Kennzahlen.....	41
3.3.5	Earned Value Management in agilen Projekten	44
3.4	Zusammenfassung	46
4	EMPIRISCHER TEIL	47
4.1	Anwendung des Design Science Ansatzes	47
4.2	Umsetzung des Prototyps.....	50
4.2.1	Anforderungen	50
4.2.2	Basisdaten	51
4.2.3	Kalkulation	53
4.2.4	Darstellung	56
4.3	Evaluierung des Prototyps	60
4.3.1	Interviews mit Expertinnen und Experten.....	60
4.3.2	Aufbau der Interviews	61
4.3.3	Qualitative Auswertung	64
5	ABSCHLUSS DER ARBEIT.....	69
5.1	Zusammenfassung	69
5.2	Ergebnisse und Beantwortung der Forschungsfrage	70
5.3	Diskussion und Ausblick	72
	ABKÜRZUNGSVERZEICHNIS	74
	ABBILDUNGSVERZEICHNIS	75
	TABELLENVERZEICHNIS	76
	LITERATURVERZEICHNIS	77

1 EINLEITUNG

Im einleitenden Kapitel erfolgt zunächst eine Beschreibung der Problemstellung. Diese stellt zugleich die Motivation zur Erstellung dieser Arbeit dar. Im Anschluss daran erfolgt eine kurze Beschreibung der Zielsetzung. Im folgenden Abschnitt werden die Vorgehensweise und die Methodik, welche im Zuge dieser Arbeit gewählt wurden, beschrieben. Bevor auf den theoretischen Teil übergeleitet wird, erfolgt ein struktureller Überblick über den Aufbau der Arbeit anhand einer Darstellung der Kapitel und der darunterliegenden Abschnitte auf erster Ebene.

1.1 Problemstellung

Führungskräfte im agilen Projektmanagement sind einerseits für die Erreichung der Projektziele verantwortlich und müssen andererseits dafür sorgen, dass die Entwicklungsteams über die für den Erfolg erforderlichen Informationen und Ressourcen verfügen. (Nee, 2010)

Nach Hazzan und Dubinsky (2014) setzt eine effiziente Steuerung von Projekten voraus, dass Schätzungen und Vorhersagen zum Projekterfolg auf Basis von Daten aus der Vergangenheit gemacht werden können. Dies gestaltet sich schwierig, da sich durch kurze Feedbackzyklen und die Anpassung an neue Anforderungen der Umfang und die Komplexität eines Projektes ändern können.

Laut Popović, Atanasijevic und Atanasijevic (2019) stellt die Verfolgung und Kontrolle der Durchführung von agil geführten Softwareprojekten eine große Herausforderung dar. Je nach Status und Zustand eines Projektes sollen kurz- und langfristige Korrekturmaßnahmen abgeleitet werden können, um den schlussendlichen Projekterfolg sicherzustellen. Übliche Metriken und Berichte, welche bei der iterationsbasierten Projektentwicklung zum Einsatz kommen, können zur Überwachung von Projekten eingesetzt werden. Sie liefern jedoch keine Auskunft über den erwarteten Erfolg.

Sulaiman, Barton und Blackburn (2006) beschreiben mit Scrum ein agiles Projektmanagement-Framework, welches die Zusammenarbeit in agilen Entwicklungsteams fördert und die Maximierung des Kundennutzens sowie des wirtschaftlichen Erfolgs zum Ziel hat. Es definiert jedoch nicht, wie der Fortschritt und die Kosten verfolgt werden sollen, um eine Gegenüberstellung von geplanten mit den tatsächlichen Werten zu ermöglichen.

Earned Value Management (EVM) gilt im Programm- und Projektmanagement als eines der effektivsten Werkzeuge zur Messung der Projektperformance und Erstellung von fortschritts- und kostenbezogenen Forecasts. Es liefert Leistungskennzahlen, die auf dem Wert der

abgeschlossenen Arbeit und der angefallenen Kosten zu einem bestimmten Zeitpunkt basieren. (Project Management Institute, 2019)

Im Rahmen der vorliegenden Arbeit soll nun der Einsatz des Projektsteuerungsinstruments EVM in agilen Softwareprojekten untersucht werden. Dazu wurde folgende Forschungsfrage definiert:

Inwieweit kann das Earned Value Management als Werkzeug zur Steuerung von agilen Projekten in Softwareunternehmen eingesetzt werden?

Aus der Forschungsfrage wurde eine gerichtete Alternativhypothese (H1) abgeleitet. Die Forschungshypothese und die dazugehörige Nullhypothese (H0), welche dieser Arbeit zugrunde liegen, lauten wie folgt:

H1: Das Earned Value Management hat einen positiven Einfluss auf die Steuerung von agilen Projekten in Softwareunternehmen.

H0: Das Earned Value Management hat keinen positiven Einfluss auf die Steuerung von agilen Projekten in Softwareunternehmen.

1.2 Zielsetzung

Mit dieser Arbeit soll überprüft werden, inwieweit das Projektmanagement-Werkzeug EVM zur Steuerung von agilen Projekten in Softwareunternehmen eingesetzt werden kann und das unter Abschnitt 1.1 beschriebene Problem löst. Zur Erreichung des Ziels soll auf Basis einer umfangreichen Literaturrecherche ein Prototyp zur Kalkulation und Visualisierung von EVM-spezifischen Kennzahlen erstellt werden. Der Prototyp dient dabei zur Veranschaulichung der Funktionsweise. Durch Befragungen von Expertinnen und Experten soll festgestellt werden, welche Einsatzgebiete sich für das Werkzeug EVM in agilen Softwareprojekten ergeben. Im Zuge dessen sollen auch Limitationen, welche mit dem Einsatz des Werkzeuges verbunden sind, identifiziert werden.

1.3 Vorgehensweise, Methodik und Aufbau

Im theoretischen Teil dieser Arbeit erfolgt zu Beginn eine grundlegende Recherche von wissenschaftlicher Fachliteratur zur agilen Softwareentwicklung. Dabei werden eingangs die Themen Agilität sowie agile, traditionelle und hybride Vorgehensmodelle im Management von Projekten diskutiert. Im Anschluss daran wird das Framework Scrum näher beleuchtet. Nach der Einführung in die agile Softwareentwicklung folgt eine theoretische Auseinandersetzung mit der Steuerung von Projekten. Diese beinhaltet eine eingehende Beschreibung von wesentlichen Aufgaben der Projektsteuerung und die Erläuterung von gängigen Berichtsmethoden, Metriken und Kennzahlen. Im Anschluss daran wird das Projektsteuerungsinstrument EVM näher beschrieben. Neben der grundlegenden Definition werden auch der Zweck, die Einsatzgebiete sowie die wesentlichen Kennzahlen von EVM dargelegt und erklärt.

Der empirische Teil dieser Arbeit folgt dem Ansatz der Design Science. Dabei steht im Wesentlichen die Erstellung eines Prototyps zur kennzahlenbasierten Steuerung von agilen Projekten im Fokus. Aus der theoretischen Auseinandersetzung mit der relevanten Fachliteratur im ersten Kapitel dieser Arbeit werden die Anforderungen an den Prototyp abgeleitet. Im Anschluss an die Erläuterung der Anforderungen erfolgt die Dokumentation der Umsetzung. Für die Evaluierung des Prototyps zur Erläuterung und Darstellung des Instruments EVM werden Interviews mit Expertinnen und Experten durchgeführt. Die Ergebnisse der Befragungen werden im Anschluss transkribiert und mittels einer qualitativen Inhaltsanalyse nach Mayring (2015) analysiert. Durch die induktive Kategorienbildung sollen die Interviewergebnisse ausgewertet und Erkenntnisse im Hinblick auf den Einsatz von EVM in agilen Projekten gewonnen werden.

Im abschließenden Kapitel erfolgt eine Zusammenfassung des theoretischen und empirischen Teils dieser Arbeit. Mitunter werden darin die Ergebnisse und Erkenntnisse diskutiert und basierend darauf die Forschungsfrage dieser Arbeit beantwortet. Abschließend wird ein Ausblick auf mögliche Folgetätigkeiten basierend auf dieser Arbeit gegeben.

Die Abbildung 1-1 zeigt die Gliederung der einzelnen Kapitel und Abschnitte auf erster Ebene.

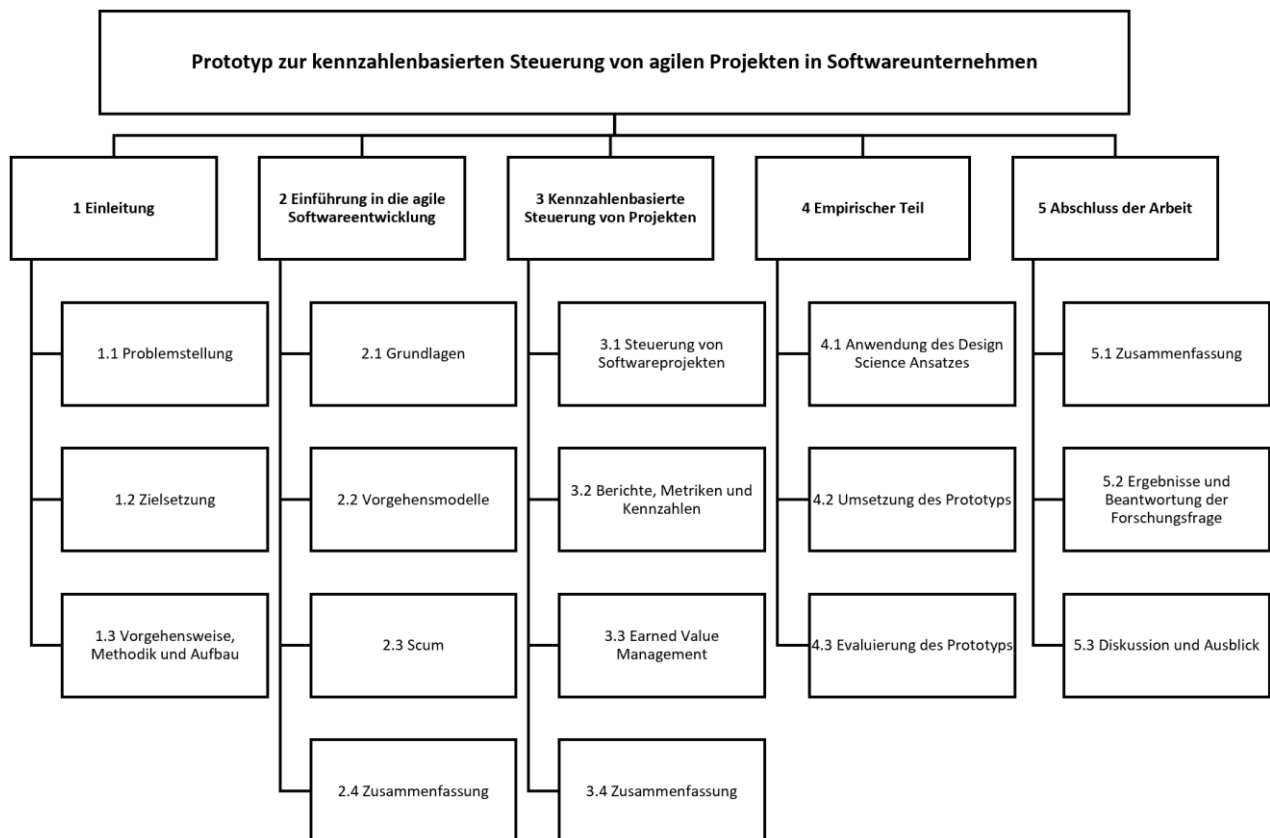


Abbildung 1-1: Aufbau der Arbeit

2 EINFÜHRUNG IN DIE AGILE SOFTWAREENTWICKLUNG

Die Nachfrage nach Agilität ist heute höher denn je und die Akzeptanz agiler Ansätze für das Management von Projekten hat stark zugenommen. Vor allem in der Softwareentwicklung haben sich agile Ansätze und Denkweisen stark etabliert und sind in vielen Unternehmen nicht mehr wegzudenken. Die Fähigkeit zur Anpassung sowie die Transparenz gelten als entscheidend für die Wertschöpfung von Softwareentwicklungsunternehmen. (Project Management Institute & Agile Alliance, 2017)

In diesem Kapitel wird zunächst der Begriff „Agilität“ definiert und dessen Werte und Prinzipien im Kontext der agilen Softwareentwicklung erläutert. Anschließend werden die unterschiedlichen Vorgehensmodelle im Projektmanagement beschrieben. Im Anschluss daran wird das Framework Scrum näher betrachtet und dessen Eigenschaften sowie darin vorkommende Artefakte beschrieben.

2.1 Grundlagen

Agile Prinzipien und Methoden genießen einen hervorragenden Ruf und sind inzwischen weit verbreitet. Rund um den Begriff „Agile“ gibt es eine erstaunlich große und inoffizielle PR-Maschine, die agile Vorgehensweisen fördert. Unter einer Vielzahl an Organisationen, welche agile Methoden anwenden und auch positiv darüber berichten, gehören auch bekannte Großunternehmen. Es gibt eine Vielzahl von Gründen für den Einsatz von agilen Vorgehensweisen. Zu den Hauptargumenten, um das Managementteam davon zu überzeugen, gehören die sehr kurzen Zeitspannen, nach welchen jeweils Ergebnisse geliefert werden können und die Flexibilität im Hinblick auf die einsetzbaren Methoden. (Cole & Scotcher, 2015)

2.1.1 Definition von Agilität

Agilität beschreibt die Fähigkeit, Änderungen in flexiblen Umfeldern herbeizuführen und erfolgreich auf sich ändernde Anforderungen zu reagieren. Es geht darum, die Umgebung, in der man sich befindet, zu verstehen und herauszufinden wie man sich an verändernde Umweltbedingungen anpassen kann, während man die ursprünglichen Ziele weiterverfolgt. Die zugrunde liegenden Werte und Prinzipien des Agilen Manifests sollen dabei eine grobe Anleitung bieten. (Agile Alliance, 2022a)

Laut Cobb (2011) ist Agilität nicht nur als Entwicklungsmethodik anzusehen, sondern vielmehr als Projektmanagementmethodik oder als Rahmenwerk für das Projektmanagement. „Agile“ hat sich im Laufe der Jahre stark weiterentwickelt, sodass eine breite Wissensbasis aufgebaut wurde und bedeutende Prozesse oder Frameworks wie Scrum davon abgeleitet werden konnten. Agile Methoden sind als Bausteine gedacht, die nach Bedarf kombiniert und erweitert werden, um eine vollständige Projektmethodik zu schaffen.

Nach Stellman und Greene (2021) liefert Agilität eine Reihe von Methoden, um Teams dabei zu helfen, effizienter zu arbeiten. Diese Methoden erstrecken sich über alle Bereiche des traditionellen Software-Engineerings, insbesondere dem Projektmanagement und beinhalten Praktiken, die möglichst einfach übernommen und angewandt werden können. Darüber hinaus kann Agilität auch als Denkweise angesehen werden, die den Austausch teaminterner Informationen fördert und Projektentscheidungen erleichtert. Anstatt eine Managementfachkraft einzusetzen, die oder der alle projektspezifischen Entscheidungen trifft, sollen die Entscheidungen in agilen Umfeldern gemeinsam als Team getroffen werden. Das Prinzip der Agilität setzt voraus, dass jedes Teammitglied gleichberechtigt ist und über denselben Wissensstand verfügt.

2.1.2 Agile Werte

Im Jahr 2001 wurde das Manifest für agile Softwareentwicklung nach einem Gipfeltreffen von 17 unabhängig denkenden Praktikern, größtenteils mit Programmierhintergrund, veröffentlicht. Zwar hatten viele Teilnehmer unterschiedliche Vorstellungen von der agilen Theorie, jedoch konnte man sich auf vier grundlegende Werte einigen. (Beck et al., 2001a)

„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** sind wichtiger als Prozesse und Werkzeuge
- **Funktionierende Software** ist wichtiger als umfassende Dokumentation
- **Zusammenarbeit mit dem Kunden** ist wichtiger als Vertragsverhandlung
- **Reagieren auf Veränderung** ist wichtiger 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.“ (Beck et al., 2001a, Abs. 1)

Moreira (2017) betont, dass besonders der abschließende Satz dabei hilft, die Absichten der Autoren zu verstehen. Es wird explizit betont, dass alle Aspekte wichtig sind und es aus agiler Sicht lediglich vor- und nachrangige Werte gibt. Im Zuge der Softwareentwicklung ist stets auf die Ausgewogenheit der agilen Grundwerte zu achten.

Cobb (2011) ergänzt, dass die Bedeutung der agilen Werte je nach Kontext individuell interpretiert werden können. Häufig besteht die fälschliche Annahme, dass die untergeordneten Werte gänzlich vermieden werden sollen. So kann es in Projekten unter Umständen also auch wichtig oder sogar notwendig sein, angemessene Dokumentation durchzuführen, Prozesse und Werkzeuge einzuführen oder Änderungen zu überwachen und nachzuverfolgen. Entscheidend ist stets, dass die Grundwerte und Prinzipien gemäß dem Projektumfeld interpretiert und angewandt werden.

2.1.3 Agile Prinzipien

Das zuvor beschriebene Agile Manifest enthält neben den Grundwerten auch zwölf Prinzipien. Diese Prinzipien hinter dem agilen Manifest (Beck et al., 2001b) lauten wie folgt:

„Unsere höchste Priorität ist es, den Kunden durch frühe und kontinuierliche Auslieferung wertvoller Software zufrieden zu stellen.

Heiße Anforderungsänderungen selbst spät in der Entwicklung willkommen. Agile Prozesse nutzen Veränderungen zum Wettbewerbsvorteil des Kunden.

Liefere funktionierende Software regelmäßig innerhalb weniger Wochen oder Monate und bevorzuge dabei die kürzere Zeitspanne.

Fachexperten und Entwickler müssen während des Projektes täglich zusammenarbeiten.

Errichte Projekte rund um motivierte Individuen. Gib ihnen das Umfeld und die Unterstützung, die sie benötigen und vertraue darauf, dass sie die Aufgabe erledigen.

Die effizienteste und effektivste Methode, Informationen an und innerhalb eines Entwicklungsteams zu übermitteln, ist im Gespräch von Angesicht zu Angesicht.

Funktionierende Software ist das wichtigste Fortschrittsmaß.

Agile Prozesse fördern nachhaltige Entwicklung. Die Auftraggeber, Entwickler und Benutzer sollten ein gleichmäßiges Tempo auf unbegrenzte Zeit halten können.

Ständiges Augenmerk auf technische Exzellenz und gutes Design fördert Agilität.

Einfachheit - die Kunst, die Menge nicht getaner Arbeit zu maximieren - ist essenziell.

Die besten Architekturen, Anforderungen und Entwürfe entstehen durch selbstorganisierte Teams.

In regelmäßigen Abständen reflektiert das Team, wie es effektiver werden kann und passt sein Verhalten entsprechend an.“ (Beck et al., 2001a, Abs. 1)

2.2 Vorgehensmodelle

Im folgenden Abschnitt werden traditionelle und agile Vorgehensmodelle im Kontext des Projektmanagements für Softwareprojekte beschrieben. Da der Fokus der vorliegenden Arbeit auf agilen Praktiken liegt, werden zunächst traditionelle Vorgehensweisen beschrieben, um diese in weiterer Folge von agilen Modellen abzugrenzen.

2.2.1 Traditionelle Modelle

Nach Wysocki (2006) sehen traditionelle Modelle im Projektmanagement vor, dass der Umfang und die Anforderungen von Projekten bereits im Vorfeld klar definiert sind und vom umzusetzenden Entwicklungsteam verstanden werden. Aus diesem Grund ist eine intensive Projektplanung notwendig, welche die Entwicklung eines vollständigen Projektplans vorsieht. Aus

dem Projektplan soll ersichtlich werden, welche Aufgaben - für einen meist längeren zeitlichen Rahmen - geplant sind, und welche Ressourcen dafür nötig sind.

Eines der bekanntesten und verbreitetsten Projektmanagement-Vorgehensmodelle ist das Wasserfallmodell. Es basiert auf dem Prinzip, dass Arbeitspakete innerhalb von Projektphasen in sequenzieller Folge erscheinen und abgeschlossen werden. Diese vereinfachte Darstellung des Wasserfallansatzes setzt in gewisser Hinsicht also voraus, dass die Aufgaben bereits beim ersten Mal richtig gemacht werden sollen. Das Wasserfall-Modell ist zwar sehr einfach zu verstehen, jedoch stellt es das Projektmanagement vor einige Herausforderungen. Unter anderem sind eine erhebliche Voraussicht und exakte Planung der nötigen Arbeitsschritte und Ressourcen über Monate oder Jahre erforderlich. Diese Planung gestaltet sich bei Softwareprojekten besonders schwierig. Der unrealistische Planungshorizont kann zu Frustration, scheiternden Projekten und entgangenen Geschäften führen. (Pries & Quigley, 2011)

Kerzner (2017a) ergänzt, dass die zu Beginn erhobenen Anforderungen bei traditionellen Vorgehensmodellen die Grundlage für das Budget und die Zeitplanung bilden. Die Planung ist dabei meist umfangreich, zeitintensiv und kostspielig. Änderungen des Projektumfangs während der Umsetzung erfordern eine explizite Genehmigung und eine Anpassung des zuvor erstellten Projektplans. Solche Änderungen können zu Verzögerungen im Projekt führen, da der Genehmigungsprozess je nach Art und Größe der Änderung viel Zeit in Anspruch nehmen kann und eine rasche Einbeziehung der Kundinnen und Kunden nicht möglich ist.

McKenna (2016) bezeichnet das Wasserfallmodell ebenfalls als Reihe von Ereignissen, welche in strikter Reihenfolge nacheinander abgearbeitet werden müssen. Ein Fortschritt zur nächsten Phase ist nur nach Abschluss der jeweils davorliegenden Phase möglich. Dies wirkt sich bei Softwareprojekten vor allem negativ auf die Qualität aus, da der Code erst getestet wird, nachdem die gesamte Entwicklung abgeschlossen wurde. Da bei wasserfallbasierten Projekten die Einhaltung der Release-Deadline oberste Priorität hat, werden Verzögerungen in der Umsetzungsphase oftmals mit einer verkürzten Testphase ausgeglichen. Entgegen dem eigentlichen Prinzip werden somit Probleme und Fehler übersehen oder bewusst ignoriert.

Timinger (2021) bezeichnet das Wasserfallmodell und das darauf basierende V-Modell als planbasierte Vorgehensmodelle, da diese Vorgehensmodelle von klar definierten und planbaren Anforderungen ausgehen. Anders als bei agilen Methoden steht bei planbasierten Ansätzen der Leistungsumfang zu Beginn fest. Aufwand, Termine, Ressourcen und Kosten werden basierend auf dem vereinbarten Umfang geschätzt und können bei Bedarf adaptiert werden.

Winston W. Royce (1970) gilt als Erfinder des Wasserfallmodells und bezeichnet es in seiner reinsten Form als riskant. Er ergänzte, dass nur die Iterationen zwischen den einzelnen Phasen das Risiko verringern können.

In Abbildung 2-1 wird das Wasserfallmodell als Vorgehensmodell zum Management von Softwareentwicklungsprojekten nach Royce (1970) dargestellt.

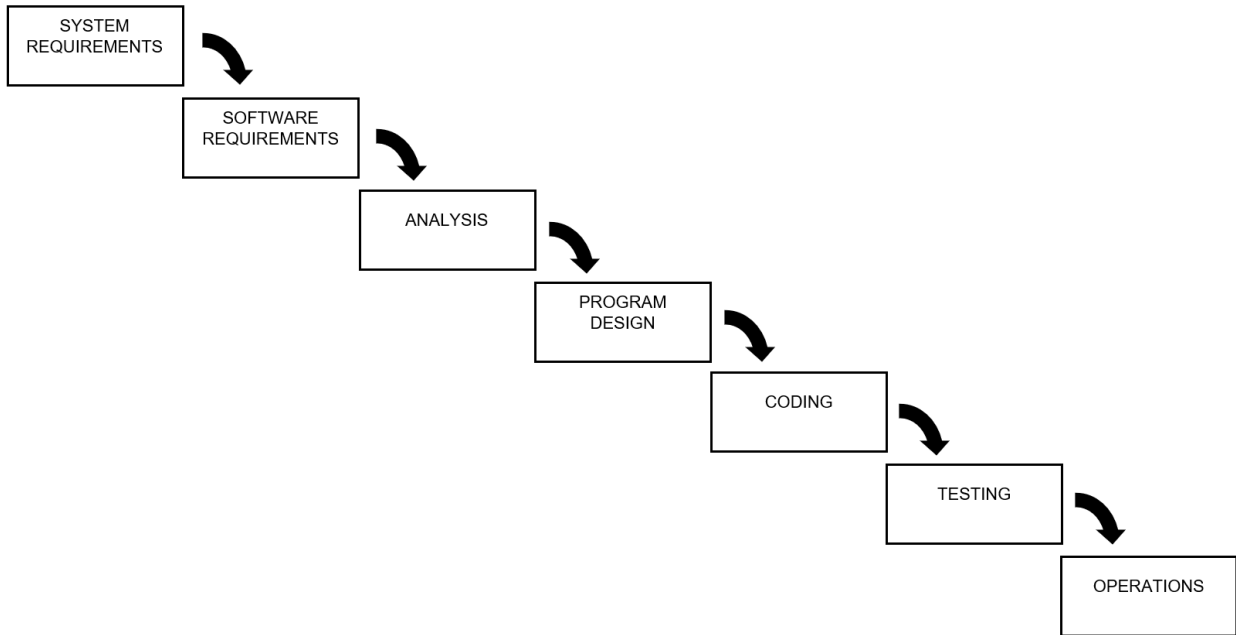


Abbildung 2-1: Wasserfallmodell nach Royce (1970)

2.2.2 Agile Modelle

Laut Coelho und Basu (2012) liefern traditionelle Vorgehensmodelle häufig keine zufriedenstellenden Antworten in Bezug auf Umfang, Zeitplan und Ressourcen eines Softwareprojekts. Ein wesentlicher Vorteil von agilen Vorgehensweisen, besteht darin, dass mehr Fokus auf die Lieferung von Ergebnissen an den Kunden gelegt und weniger Druck im Hinblick auf den Abschluss einzelner Aktivitäten erzeugt wird. Das übergeordnete Ziel von agilen Methoden ist die Beantwortung der Frage, welche Ergebnisse mit welchen Ressourcen in welchem Zeitrahmen einsatzfähig an Kundinnen und Kunden ausgeliefert werden können. Agile Ansätze helfen bei der Beantwortung dieser Fragen, da die Planung von Projekten auf verschiedenen Ebenen passiert und adaptiert werden kann. Darüber hinaus können Methoden eingesetzt werden, welche auf Basis historischer Daten Prognosen und Trends für den weiteren Verlauf eines Projektes ermöglichen.

Für Kerzner (2017a) ist auch das zusätzliche Vertrauen in Projektteams, um situationsbasierte Entscheidungen zu treffen, ein wichtiger Grund für den Erfolg des agilen Projektmanagements. Während traditionelle Projektmanagement-Methoden meist strikten Richtlinien folgen müssen, um einen wiederholbaren Projekterfolg zu erzielen, ermöglichen agile Ansätze die kundenspezifische Auswahl und Anpassung der Methodik. Dies ist vor allem für externe Kundinnen und Kunden wichtig, da diese die Anpassung der Methoden an deren Geschäftsmodell und ihre Art der Geschäftstätigkeit vorziehen.

Nach Wysocki (2006) wird im agilen Projektmanagement kein vollständiges Anforderungsdokument für den gesamten Projektumfang erstellt, da im Vorfeld nicht alle

Anforderungen für den gesamten Projektumfang bekannt sind. Infolgedessen wird lediglich für die bereits bekannten Anforderungen ein Plan erstellt, welcher in mehreren Iterationen abgearbeitet wird. Der mangelnde Ausblick hinsichtlich des Endergebnisses und die oftmalige Notwendigkeit zur Anpassung der Pläne führt zur Tatsache, dass agile Projekte meist einem höheren Risiko ausgesetzt sind als Projekte, die nach traditionellen Vorgehensweisen durchgeführt werden. Bei traditionellen Vorgehensweisen wird vor Projektstart üblicherweise bereits eine Reihe von Ergebnissen identifiziert und geplant. Die geplanten Artefakte bei agilen Projekten hängen allerdings jeweils vom zuletzt ausgelieferten Ergebnis ab. In Kombination mit sich ändernden Anforderungen kann das bedeuten, dass Ergebnisse in agilen Projekten stark verändert oder unter bestimmten Umständen gar nicht ausgeliefert werden. Agile Ansätze verfolgen also in der Regel auch das Ziel der minimalen Ressourcenverschwendung.

Agiles Projektmanagement fördert einen wertorientierten Ansatz, der es Entwicklungsteams ermöglicht, qualitativ hochwertige Arbeit mit hoher Priorität zu liefern. Während man bei der traditionellen Entwicklung häufig erst dann in die nächste Phase gelangt, wenn die vorherige Phase abgeschlossen ist, handelt es sich bei der agilen Projektentwicklung um einen fortlaufenden Prozess, bei dem die aktuelle Projektsituation die Basis für projektspezifische Entscheidungen bildet. Dies kann vor allem dann von Vorteil sein, wenn die Planung flexibel erfolgen kann und das Ergebnis wichtiger ist als die Dokumentation und Einhaltung von Prozessen. (Vanzant Stern, 2017)

Agile Vorgehensmodelle sehen vor, dass sowohl Design, Entwicklung und Tests als auch Reporting-spezifische Tätigkeiten der Planung, Beobachtung und Visualisierung nicht mehr, wie bei traditionellen Wasserfallmethoden, sequenziell ablaufen, sondern parallel erfolgen und inkrementell verfeinert werden. (Freedman, 2016)

Nach Moreira (2017) liegt das Hauptaugenmerk der agilen Vorgehensweise darin, den Kundinnen und Kunden so früh wie möglich einen Mehrwert zu liefern. Die Validierung und Beurteilung des gelieferten Mehrwertes passiert in iterativen Schritten durch die Kunden. Eine zentrale Aufgabe im agilen Projektmanagement liegt also in der Entscheidung, warum Funktionalitäten ausgeliefert werden und zu welchem Zeitpunkt das geschehen soll.

Laut Martinelli und Milošević (2016) scheinen Kundinnen und Kunden durch die frühe Auslieferung von Ergebnissen auch zu glauben, dass agile Prozesse den Abschluss ganzer Projekte beschleunigen. Da die Schnelligkeit in Bezug auf die Projektdurchlaufzeit kein zentraler Wert oder eine Charakteristik von agilen Methoden ist, ist dies nicht zutreffend. Vielmehr stehen die Bereitstellung nutzbarer Funktionen und die Realisierung von Geschäftswert zu einem möglichst frühen Zeitpunkt im Projektlebenszyklus im Vordergrund. Dies ist ein wesentlicher Vorteil gegenüber wasserfallbasierten Ansätzen und kann in agilen Projekten durch die Einführung kurzer Iterationen und die regelmäßige Auslieferung von Teilergebnissen erreicht werden.

Da die Auslieferung einer Software bei agiler Vorgehensweise in mehreren Zyklen passiert, ist die Testphase keine nachgelagerte Projektphase, sondern vielmehr ein Teil jeder Iteration, mit wesentlich höherem Stellenwert. Durch die Häufigkeit der Testvorgänge, versuchen

Entwicklungsteams die höchstmögliche Qualität jedes Inkrements und generell am Produkt sicherzustellen. (McKenna, 2016)

Die Tabelle 2-1 stellt die Projekterfolgsraten von agilen Projekten, welche mit wasserfallbasierten Methoden geführt werden, gegenüber. Eine Untersuchung ergab, dass IT-Projekte basierend auf agilen Vorgehensmodellen eine mehr als dreimal häufigere Erfolgsquote haben als Projekte, welche mit traditionellen Vorgehensweisen geführt werden. (The Standish Group International, 2020)

METHOD	SUCCESSFUL	CHALLENGED	FAILED
AGILE	42%	47%	11%
WATERFALL	13%	59%	28%

Tabelle 2-1: Erfolgsraten von IT-Projekten (The Standish Group International, 2020)

2.2.3 Hybride Modelle

Highsmith (2010) untergliedert den Einsatz von agilen Methoden und Werkzeugen innerhalb einer Organisation in vier Schichten. Während die Portfolio-Governance-Schicht die projektübergreifenden Eckpfeiler und Regeln vorgibt, werden im Projektmanagement spezielle Methoden zur Entwicklung von Produktvisionen und zur Planung von Releases angewandt. Im Fokus des Iterationsmanagements stehen die Planung und Ausführung einzelner Iterationen. Hierbei kann beispielsweise Scrum zum Einsatz kommen. In der vierten Ebene, welche zugleich die Grundlage einer effektiven Projektabwicklung bildet, wird entschieden, welche konkreten Praktiken in der Ausführung zum Einsatz kommen. Da keine einzelne Methode alle Ebenen abdecken kann, muss auf jeder Ebene individuell entschieden werden, welche Methoden zum Einsatz kommen. Durch die flexible Kombination auf den unterschiedlichen Ebenen lassen sich individuelle Frameworks erstellen. Die vier Schichten des Agile Enterprise Frameworks nach Highsmith (2010) werden in Abbildung 2-2 dargestellt.

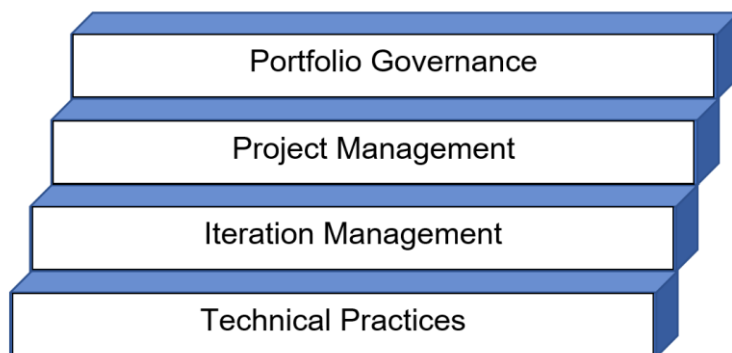


Abbildung 2-2: Das Agile Enterprise Framework nach Highsmith (2010)

Cobb (2011) ergänzt, dass agile Methoden auch mit traditionellen Projektmanagementansätzen kombiniert werden können, um ein Gleichgewicht zwischen Agilität und Kontrolle zu schaffen. Traditionelle Methoden und Werkzeuge werden oft als veraltet und nicht zeitgemäß angesehen. Je nach Komplexität, Risiko und weiteren Faktoren, wie gesetzlichen Vorgaben, gibt es aber nach wie vor gute Gründe, um traditionelle Methoden einzusetzen und mit agilen Praktiken zu kombinieren. Darüber hinaus besteht häufig noch der Irrglaube, dass Unternehmen nur vollständig nach traditionellen oder vollständig nach agilen Vorgehensmodellen arbeiten können. Es wird also angenommen, dass zwischen den beiden Extremen eine Lücke besteht und keine Mischform möglich ist. Tatsächlich gibt es eine Vielzahl an Möglichkeiten zur Kombination von traditionellen Methoden und agilen Praktiken. Softwareentwicklungsunternehmen müssen nicht zwingend eine vollständig traditionelle oder agile Haltung einnehmen. Vielmehr können nach und nach neue Praktiken und Ansätze erprobt werden, um den Grad der Agilität schrittweise zu erhöhen.

Die Abbildung 2-3 zeigt die Annahme einer fehlenden Mischform zwischen traditionellen und agilen Vorgehensmodellen und die tatsächliche schrittweise Steigerung der Agilität in Unternehmen nach Cobb (2011).

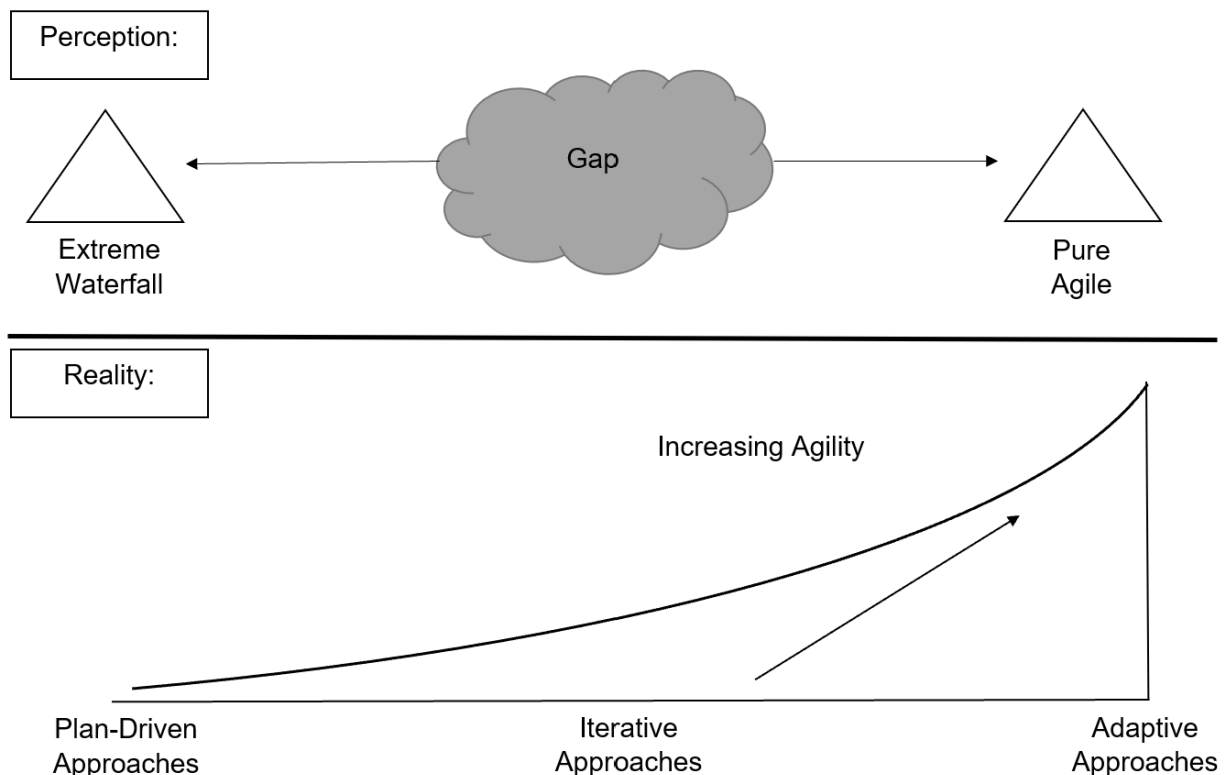


Abbildung 2-3: Schrittweise Steigerung der Agilität nach Cobb (2011)

Laut Kerzner (2017a) tendieren Unternehmen mit mangelnder Projektmanagementenerfahrung häufig zu formalisierten Ansätzen und traditionellem Vorgehen. Abweichungen vom Plan sollen häufig noch vermieden werden, damit Prozesse und Richtlinien eingehalten werden können. Erst

mit zunehmendem Reifegrad im Projektmanagement lässt sich erkennen, dass das Vertrauen in die Projektteams steigt und Unternehmen sukzessiv zu informelleren Methoden übergehen. Heutzutage werden die meisten Softwareprojekte mit einem Ansatz gemanagt, der weder agil noch starr ist. Zumeist handelt es sich um einen hybriden Ansatz mit einem gewissen Grad an Flexibilität und Informalität. Durch die variable Auslegung der Flexibilität bei jedem Projekt, werden solche Ansätze auch als Frameworks bezeichnet.

Auch das Project Management Institute und die Agile Alliance (2017) beschreiben die Möglichkeit zur Nutzung von hybriden Projektmanagementansätzen. Da das Ziel des Projektmanagements lautet, mit gegebenen Voraussetzungen einen möglichst hohen Geschäftswert zu erzielen, spielt der gewählte Ansatz zur Durchführung des Projektes keine Rolle. Unter Umständen kann es also Sinn machen, im selben Projekt eine Kombination aus agilen und traditionellen Ansätzen zu verwenden. Teams können also agile Ansätze wie kurze Iterationen, tägliche Statusmeetings und Retrospektiven einführen, während auf traditionelle Praktiken wie die Vorabschätzung und Fortschrittsverfolgung gesetzt wird.

Nach Boehm und Turner (2004) ist der Einsatz geeigneter Werkzeuge zur Verwaltung der Aufwände unter bestimmten Projektbedingungen unerlässlich. Agile Ansätze müssen in schnelllebigen Entwicklungsumgebungen durch ein angemessenes Maß an Planung und einen disziplinierten Prozess für die Erledigung der Arbeit unterstützt werden. Die Notwendigkeit von Disziplin in agilen Umfeldern wird wie folgt beschrieben:

„Agility without discipline is the unencumbered enthusiasm of a startup company before it has to turn a profit. Great companies have both (discipline and agility) in measures appropriate to their goals and environment.“ (Boehm & Turner, 2004, S. 2)

2.3 Scrum

In diesem Abschnitt wird das Framework Scrum näher beschrieben. Eingangs wird erläutert, weshalb Scrum im Rahmen dieser Arbeit gewählt wurde. Anschließend wird das Framework selbst definiert und erklärt, bevor näher auf die spezifischen Rollen, Events und Artefakte eingegangen wird.

2.3.1 Entscheidung für Scrum

Nach Jahrzehnten agiler Praxis gibt es inzwischen dutzende unterschiedliche Methoden und Vorgehensmodelle. Das bekannteste Modell ist das sogenannte „Scrum Framework“. Scrum unterstützt kurze Feedbackschleifen, verringert Risiken und ermöglicht rasch Kennzahlen zu ermitteln. (McKenna, 2016)

Laut Cole und Scotcher (2015) bietet Scrum eine Lösung, die nahezu in allen Organisationen Anwendung finden kann, da es sehr gut dokumentiert ist, einfache Prinzipien verfolgt und Teams in überschaubaren Größen gehalten werden. Sind Teams zu groß, können diese geteilt werden, und jedes Team bestimmt eine Vertreterin oder einen Vertreter, die oder der an einem täglichen

Treffen mit Vertreterinnen oder Vertretern der anderen Teams teilnimmt. Diese Technik zur Koordination mehrerer Scrum-Teams ist bekannt als Scrum of Scrums.

Zizlavsky und Fisher (2021) ergänzen, dass Scrum keine spezifischen Tools, Technologien oder Verfahren vorschreibt, sondern eine Möglichkeit für effizientes Zusammenarbeiten und Kommunizieren innerhalb eines Teams bietet. Im Vordergrund steht die Lieferung optimaler Ergebnisse, indem Hindernisse laufend überwacht und schnellstmöglich beseitigt werden.

2.3.2 Definition

Scrum ist die Methode, die in der Literatur am häufigsten erwähnt wird. Die Prozesse, die Terminologie und der Stil des Teammanagements sind eine genaue und transparente Demonstration der Funktionsweise und der damit verbundenen Philosophie. Das Wort „Scrum“ stammt aus dem Rugbysport und bezeichnet eine Spielsituation, in der sich die acht schwersten und körperlich stärksten Spieler jeder Seite versammeln und gemeinsam gegen ihre Gegner antreten. (Stark, 2014)

Aus fachlicher Sicht bezeichnen Schwaber und Sutherland (2020) Scrum als leichtgewichtiges und anpassbares Framework, welches nur wenige Teile in der Umsetzung voraussetzt. Es kann also flexibel eingesetzt, adaptiert und erweitert werden können. Die Flexibilität von Scrum wird wie folgt beschrieben:

„Various processes, techniques and methods can be employed within the framework. Scrum wraps around existing practices or renders them unnecessary. Scrum makes visible the relative efficacy of current management, environment, and work techniques, so that improvements can be made.“ (Schwaber & Sutherland, 2020, Abs. 2)

Auch Cole und Scotcher (2015) beschreiben, dass Scrum inzwischen mit einer Reihe anderer Frameworks koexistiert, es aber nach wie vor das beliebteste agile Framework ist. Dies liegt vor allem daran, dass es kompakt beschrieben wird, ohne Teams in lästigen Regeln, Prozessen und Dogmen zu ertränken. Darüber hinaus gewährt Scrum einen einfachen Einstieg in die agile Disziplin, da es klar definierte Rollen und Verantwortlichkeiten, Zeremonien und Artefakte hat, aber dennoch genug Flexibilität bei der Umsetzung zulässt.

Ein weiterer Vorteil nach Vanzant Stern (2017) ist, dass Scrum in beinahe jeder Art von Projekt eingesetzt werden kann, obwohl es ursprünglich nur für Softwareentwicklungsprojekte gedacht und eingesetzt wurde.

Das Project Management Institute und die Agile Alliance (2017) beschreiben Scrum als Prozess-Framework, das zur Produktentwicklung verwendet wird. Das Framework setzt sich primär aus Rollen, Ereignissen und Artefakten zusammen und wird durch individuell im Team vereinbarte Regeln zusammengehalten. Scrum verwendet einen iterativen Ansatz, um ein funktionierendes Produkt zu liefern. Es wird in Zeitspannen von einem Monat oder weniger mit konsistenten

Dauern ausgeführt, die als Sprints bezeichnet werden und in denen ein potenziell auslieferbares Produktinkrement produziert wird.

Die Abbildung 2-4 veranschaulicht das Scrum Framework auf einen Blick. Es beschreibt zudem den Zusammenhang der Scrum-spezifischen Rollen, Ereignisse und Artefakte. Eine genauere Beschreibung dieser Komponenten folgt in den nachfolgenden Abschnitten 2.3.2 bis 2.3.4.

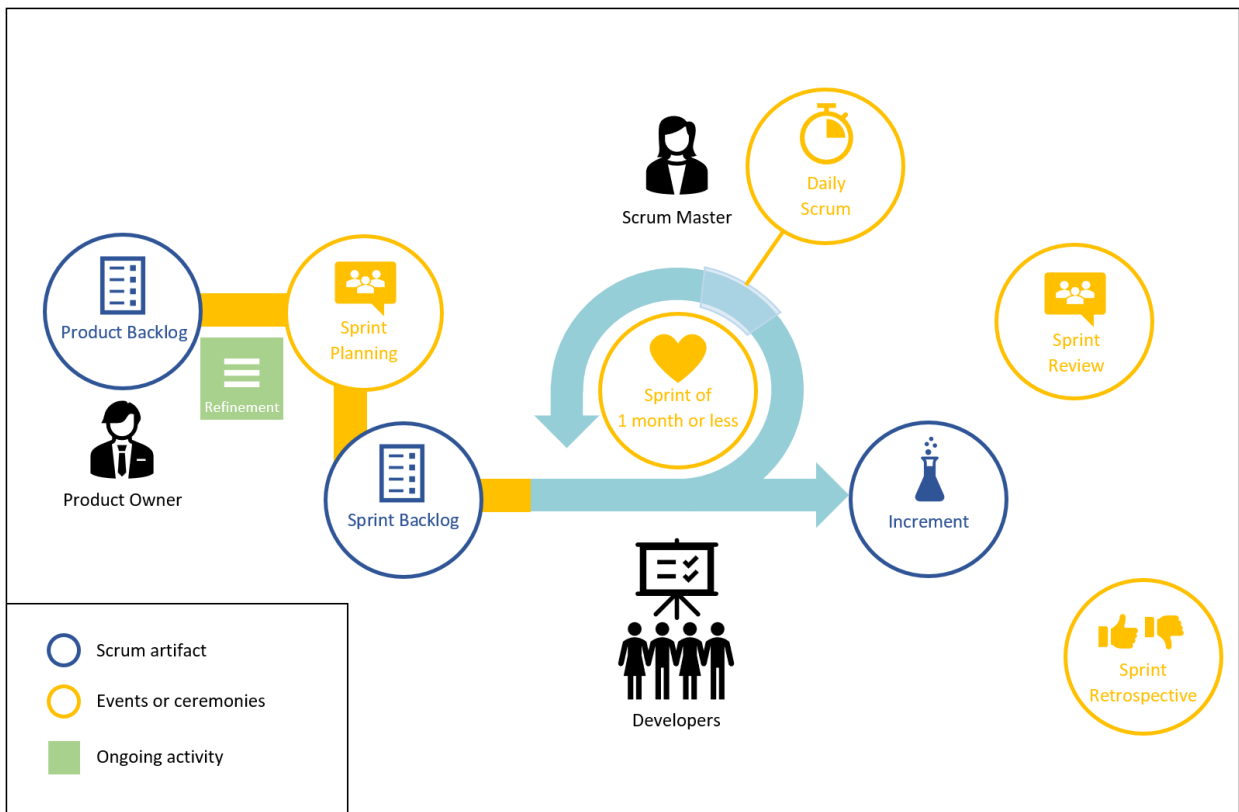


Abbildung 2-4: Das Scrum Framework (Scrum Alliance, 2022)

2.3.3 Rollen

In diesem Abschnitt werden die wesentlichen Rollen des Scrum Frameworks beschrieben.

Product Owner

Der Product Owner ist für die Ausrichtung des Produkts zuständig. Product Owner reihen und priorisieren die Arbeit basierend auf deren Geschäftswert ein. Product Owner arbeiten täglich mit ihren Teams zusammen, indem sie einerseits Feedback zu Geschehenem geben und andererseits die Richtung für die nächste Funktionalität festlegen. Der Product Owner arbeitet dafür mit Stakeholdern, Kunden und den Teams zusammen. Typischerweise haben Product Owner einen betriebswirtschaftlichen Hintergrund und bringen fundierte Fachkenntnisse in die

Entscheidungen ein. Agile Vorgehensweisen sehen vor, dass Product Owner das Backlog in Absprache mit dem Team erstellen. Das Backlog kann als Darstellung der geplanten und nötigen Aktivitäten angesehen werden. Ein entscheidender Faktor bei der Reihung ist neben der Priorität auch der Wert für den Kunden. (Project Management Institute & Agile Alliance, 2017)

Cline (2015) sieht in seiner Definition die zentrale Aufgabe eines Product Owners in der Vertretung der Interessen und Bedürfnisse aller involvierten Stakeholder. Sämtliche Anforderungen für Funktionalitäten müssen über den Product Owner eingekippt werden. Dieser arbeitet über sämtliche Phasen des Projektes eng mit dem Team zusammen.

Entwickler

Scrum Teams organisieren sich in der Regel selbst und bestehen aus Teammitgliedern, mit allen erforderlichen Fähigkeiten, um ein funktionierendes Produkt zu produzieren. In der Softwareentwicklung bestehen funktionsübergreifende Teams typischerweise aus Designern, Entwicklern, Testern und allen anderen erforderlichen Rollen. Diese Teams sollen Ergebnisse in kürzester Zeit, mit höherer Qualität und ohne externe Abhängigkeiten liefern können. (Project Management Institute & Agile Alliance, 2017)

Laut Scrum Guide von Schwaber und Sutherland (2020) verpflichten sich die Entwickler eines Scrum-Teams zur Erstellung verwendbarer Inkremente je Intervall. Die Fähigkeiten der Entwickler sind dabei oft breit gefächert und variieren je nach Anforderungen. Unabhängig von den Anforderungen sind Entwickler jedoch immer verantwortlich für die Erstellung eines Sprint-Plans (Sprint Backlog), die Einhaltung definierter Qualitätsmerkmale (Definition of Done), die laufende Adaptierung anhand des Sprintziels und den professionellen Umgang untereinander.

Scrum Master

Die Scrum Alliance (Scrum Alliance, 2022) beschreibt die zentrale Aufgabe des Scrum Masters in der Etablierung von Scrum innerhalb eines Teams oder in der gesamten Organisation. Sie tun dies, indem sie jedem helfen, Scrum-Theorie und -Praxis zu verstehen und zu verbessern. Zu den konkreten Aufgaben eines Scrum Masters gehört die Moderation von Meetings, der Schutz des Teams vor Fremdeinwirkungen, die antiautoritäre Führung des Teams und die Sicherstellung und kontinuierliche Verbesserung agiler Prinzipien in der gesamten Organisation.

Die Rolle des Scrum Masters ist umfangreich und erfordert ausgeprägte Führungsqualitäten. Da der Aufbau von Führungsqualitäten in Bezug auf Moderation, Coaching und Beseitigung von Hindernissen oftmals ein langer Prozess sein kann, laden viele Organisationen externe agile Coaches zur Unterstützung und zum Aufbau interner Ressourcen und Fähigkeiten ein. Auch wenn externe Coaches meist viel Erfahrung mitbringen, kann die mangelnde Beziehung zu den Teammitgliedern nur schwer kompensiert werden. Der Scrum Master sollte also als fixer Bestandteil eines Scrum Teams gesehen und intern besetzt werden. (Project Management Institute & Agile Alliance, 2017)

2.3.4 Events

Der folgende Abschnitt beschäftigt sich mit der Erläuterung der Scrum-spezifischen Events.

Sprint

Der Sprint ist das Herzstück des Scrum-Prozesses. Ein "Sprint" ist ein Zeitfenster von fester Länge, das in der Regel zwei bis vier Wochen dauert und in welchem die eigentliche Entwicklungsarbeit geleistet wird. Im traditionellen Projektmanagement basiert die Dauer einzelner Iterationen oder Projektphasen auf Schätzungen der darin geplanten Arbeitspakete. Durch die Planung in Form von Sprints wird die Länge der Zyklen vorgegeben und die darin zu erledigenden Aufgaben werden in kleinere Inkremente aufgeteilt. Dabei muss stets sichergestellt werden, dass jedes Inkrement in einem Sprint erledigt werden kann. Anstatt die Länge eines Sprints zu verlängern, um die zu erledigende Arbeit zu bewältigen, besteht die Herausforderung darin, herauszufinden, wie viele dieser kleinen Arbeitsschritte in einen einzigen Sprint mit fester Länge eingepasst werden können. (Cobb, 2015)

Die Agile Alliance (Agile Alliance, 2022b) ergänzt, dass Teams im Zuge von Sprints ein potenziell auslieferbares Produktinkrement produzieren. Darüber hinaus werden Sprints über folgende Merkmale definiert:

- Beibehaltung einer gleichbleibenden Dauer während der gesamten Entwicklungsarbeit
- Ein neuer Sprint folgt unmittelbar auf den Abschluss des vorherigen Sprints
- Start- und Enddatum des Sprints sind festgelegt

Sprint Planning

Das Sprint Planning-Meeting findet vor Beginn eines jeden Sprints statt. Es hat zwei Hauptziele und ist daher in der Regel auch in zwei Teile unterteilt. Das Ziel des ersten Teils besteht darin, dass der Product Owner und das Team aushandeln, welche Stories aus dem Product Backlog in den Sprint aufgenommen werden sollen. Das Product Backlog ist eine Auflistung der zu erledigenden Arbeit, typischerweise in Form sogenannter User Stories. Es sollte vom Product Owner vor dem Sprint Planning-Meeting aufbereitet werden. Die Entscheidung, welche User Stories im nächsten Sprint geplant werden, wird vom Product Owner gemeinsam mit dem Team getroffen. Als Grundlage für die Auswahl dienen Prioritäten und Aufwandsschätzungen der einzelnen Stories. Die identifizierten Stories bilden in weiterer Folge das jeweilige Sprint Backlog. Im zweiten Teil des Meetings werden die konkreten Aufgaben, die für die Umsetzung der User Stories erforderlich sind, definiert und verteilt. Wichtig ist, dass sich der Product Owner und das Team auf ein Gesamtziel einigen, das im Sprint erreicht werden soll. (Cobb, 2015)

Daily Scrum

Das Daily Scrum-Meeting dient laut Vanzant Stern (2017) einerseits dazu, um den aktuellen Fortschritt zu erheben, und andererseits um künftige Aufgaben zu koordinieren und etwaige Hindernisse, die den Fortschritt behindern könnten, zu identifizieren. Während des Meetings werden in der Regel folgende drei Fragen gestellt:

- Was haben Sie gestern erreicht?
- Was werden Sie heute tun?
- Gibt es irgendwelche Hindernisse, die Ihnen im Weg stehen?

Cobb (2015) ergänzt, dass das Daily Scrum-Meeting oft vor einem Scrum Board stattfindet, das den Fortschritt der Aufgaben im aktuellen Sprint anzeigt. Abbildung 2-5 zeigt ein Beispiel für ein Scrum-Board. Bei verteilten Teams wird anstelle eines physischen Scrum-Boards in der Regel ein agiles Projektmanagement-Tool verwendet. Das Daily Scrum-Meeting wird häufig auch für Besprechungen genutzt, welche den vorgesehenen Rahmen überschreiten. Es empfiehlt sich jedoch, den Fokus auf die wesentlichen Fragen zu begrenzen, die Länge des Daily Scrum-Meetings zu begrenzen und weitere Themen in separate Besprechungen auszulagern.

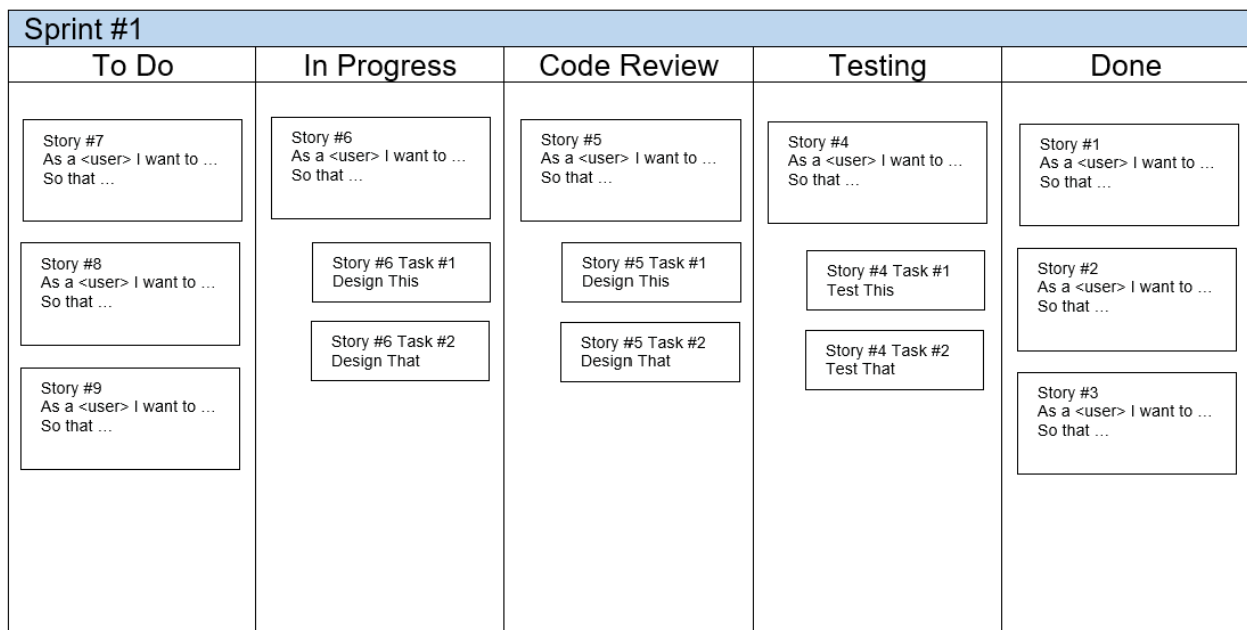


Abbildung 2-5: Scrum-Board nach Cobb (2015)

Sprint Review

Beim Sprint Review präsentiert das Team dem Product Owner und weiteren Stakeholdern die fertige Arbeit zur abschließenden Überprüfung und Abnahme. Der Zweck dieses Meetings ist es, das Ergebnis zu demonstrieren und zu besprechen. Sofern möglich, kann das Meeting dazu

verwendet werden, um einen kurzen Benutzerakzeptanztest durchzuführen. Es ist zu beachten, dass der Sprint Review nicht dazu gedacht ist, einen Statusbericht zu erstellen, sondern dass das Feedback des vergangenen Sprints in das Product Backlog für zukünftige Überlegungen aufgenommen wird. (Agile Alliance, 2022b)

Sprint Retrospektive

Die Sprint-Retrospektive ist eine Gelegenheit, am Ende des Sprints die gemachten Erfahrungen zu überprüfen und zu besprechen, wobei das Team zurückblickt, darüber nachdenkt, was gut und was nicht gut gelaufen ist. Da die Retrospektive am Ende eines jeden Sprints geschieht und die Sprints alle zwei bis vier Wochen stattfinden, können Probleme rasch identifiziert und Verbesserungen schneller abgeleitet werden als bei traditionellen Projekten. Ein agiler Entwicklungsprozess soll sich stark auf kontinuierliche Verbesserungen stützen, um den Prozess so weit wie möglich an das Projekt und die Umgebung anzupassen. (Cobb, 2015)

Nach Vanzant Stern (2017) können wie beim Daily Scrum gezielte Fragen gestellt werden, um eine Diskussion anzuregen und das nötige Feedback von den Teammitgliedern einzuholen. Diese Fragen könnten wie folgt lauten:

- Was ist während des Sprint-Zyklus gut gelaufen?
- Was ist während des Sprint-Zyklus schiefgelaufen?
- Was könnten wir anders machen, um uns zu verbessern?

2.3.5 Artefakte

Im abschließenden Teil dieses Abschnittes werden die grundlegenden Artefakte, welche in Scrum zum Einsatz kommen, erklärt.

Product Backlog

Sobald eine Vision eines Produkts vorhanden ist, muss das Projektteam genauer festlegen, was erforderlich ist. Ein zentrales Element jedes Projektes ist eine Art von Anforderungsliste, die im agilen Umfeld als „Product Backlog“ bezeichnet wird. Es ersetzt den traditionellen, detaillierten, anforderungsspezifischen Ansatz und hat die Form einer Einkaufsliste mit Ideen, die für das Produkt von Bedeutung sind. Die Elemente des Backlogs sind immer benutzerorientiert, auch wenn sie eine technische Ausrichtung haben. (Cole & Scotcher, 2015)

Sprint Backlog

Ein Sprint Backlog ist die Teilmenge des Product Backlogs, die ein Team während eines Sprints liefern will, um das Sprint-Ziel zu erreichen und Fortschritte in Richtung eines gewünschten

Ergebnisses zu machen. Das Sprint-Backlog besteht aus Product-Backlog-Elementen, die das Team während der Sprint-Planung mit seinem Product Owner vereinbart hat. Das Team ist Eigentümer des Sprint Backlogs und kann bestimmen, ob neue Elemente hinzugefügt oder bestehende Elemente entfernt werden. Dadurch kann sich das Team auf einen klaren Umfang für die Dauer des Sprints konzentrieren. Einige Teams können die Aufnahme eines neuen Product-Backlog-Elements erlauben, wenn es ein Product-Backlog-Element gleicher oder größerer Größe ersetzt, das bereits im Sprint-Backlog vorhanden ist. (Agile Alliance, 2022b)

Inkrement

Ein Inkrement im Kontext der Scrum-basierten Softwareentwicklung ist die Summe der nutzbaren Sprint-Backlog-Elemente, die von den Entwicklern in dem Sprint fertiggestellt wurden. Es stellt eine erkennbare, sichtbar verbesserte, funktionierende Version des Produkts dar. Inkremente müssen der sogenannten „Definition of Done“ entsprechen. Es muss also einem gemeinsamen Verständnis von „erledigt“ entsprechen. Die Definition of Done kann in verschiedenen Organisationen und Teams unterschiedlich sein und stellt sicher, dass ein vereinbarter Qualitätsstandard eingehalten wird. (Scrum Alliance, 2022)

2.4 Zusammenfassung

In diesem Kapitel wurde eingangs das Thema Agilität behandelt. Zu Beginn wurde der Begriff „Agilität“ erläutert, bevor kurz auf dessen Entstehung eingegangen wurde. Nach der Begriffsdefinition wurden die grundlegenden agilen Werte und Prinzipien dargestellt.

Die untergeordneten Werte werden laut Cobb (2011) häufig vernachlässigt oder gänzlich ignoriert, obwohl diese, abhängig von den Rahmenbedingungen im Projektmanagement, von Wichtigkeit sein können. Der Einsatz von Werkzeugen zur Projektsteuerung kann also auch in agilen Umfeldern legitim oder unter Umständen sogar notwendig sein.

Im darauffolgenden Abschnitt wurden unterschiedliche Vorgehensmodelle im Projektmanagement erläutert. Zuerst wurden klassische Vorgehensmodelle wie das Wasserfallmodell nach Royce (1970) beschrieben und dessen Einsatzgebiete sowie Limitationen aufgezeigt. Im Kontrast dazu wurden auch agile Vorgehensmodelle erörtert. McKenna (2016) beschreibt, dass bei agilen Methoden die schrittweise Lieferung von Teilergebnissen in möglichst hoher Qualität im Vordergrund steht. Neben den agilen und traditionellen Modellen wird auch eine Kombination dieser beiden Ansätze beschrieben, die sogenannten „hybride“ Modelle.

Cobb (2011) beschreibt, dass stets ein Gleichgewicht zwischen Agilität und Kontrolle herrschen sollte und es eine Vielzahl an Möglichkeiten zur Kombination von traditionellen Methoden und agilen Praktiken gibt. Während häufig die Annahme besteht, dass Unternehmen entweder nur nach vollständig agilen oder traditionellen Ansätzen arbeiten, nutzen die meisten Unternehmen eine Mischform.

Im darauffolgenden Abschnitt wurde das Framework Scrum näher beschrieben. Zunächst wurde die Entscheidung für die Auswahl von Scrum erläutert. Im Anschluss daran wurde das Framework selbst, sowie die darin vorkommenden Rollen, Events und Artefakte näher beschrieben. Laut Cole und Scotcher (2015) ist Scrum das beliebteste agile Framework und kann aufgrund dessen Flexibilität in nahezu allen Organisationen eingesetzt werden.

3 KENNZAHLENBASIERTE STEUERUNG VON PROJEKTEN

Traditionell liegt der Schwerpunkt des Projektmanagements auf der Verfolgung von Kosten und des Zeitplans. Um eine zeitgemäße und erfolgreiche Steuerung von Projekten zu ermöglichen, muss ein Ansatz verfolgt werden, welcher das richtige Maß an Kontrolle und Agilität verbindet. Das passende Gleichgewicht hängt von einer Reihe von Faktoren ab. Ausschlaggebend dafür sind unter anderem die Unternehmenskultur, die Notwendigkeit von Controlling- und Reporting-Maßnahmen und der Grad der Ungewissheit von Anforderungen. (Cobb, 2011)

Laut Nikravan und Forman (2010) werden die Methoden zur Überwachung und Steuerung sowohl in traditionellen Projekten als auch in der Softwareproduktentwicklung ständig weiterentwickelt und erprobt. Agile Einflüsse wie der Einsatz kurzer Sprint-Zyklen finden immer häufiger Anwendung. Die iterative Vorgehensweise ermöglicht einerseits die schnellere Auslieferung von Ergebnissen, wodurch wiederum Projektrisiken und -unsicherheiten reduziert werden können. Während die Anforderungen und die Entwicklung von Produkten dynamisch bleiben, sind immer noch zusätzliche Werkzeuge erforderlich. Diese dienen einerseits zur Darstellung des Projektstatus gegenüber den Stakeholdern und liefern andererseits einen wertvollen Beitrag im Steuerungs- und Entscheidungsfindungsprozess.

Dieses Kapitel beschäftigt sich mit der kennzahlenbasierten Steuerung von agilen Projekten. Dabei werden zunächst wesentliche Aufgaben der Projektsteuerung erläutert. Im Anschluss daran werden bekannte Berichte, Metriken und Kennzahlen zur Steuerung beschrieben. Darüber hinaus wird das Projektsteuerungsinstrument EVM näher beschrieben.

3.1 Steuerung von Softwareprojekten

Der komplexe und vielschichtige Charakter der Softwareentwicklung macht das Management von Softwareprojekten zu einer anspruchsvollen Aufgabe. Der wettbewerbsorientierte Markt erfordert wirksame Mittel im Management. Ein Softwareentwicklungsprojekt sollte in Business Cases zerlegt werden. Dabei sollen die Gründe für die Investition und der erwartete Nutzen sowie die geplanten Kosten und Risiken dargelegt werden. Zu den wichtigsten Erfolgsfaktoren eines Softwareprojektes gehören auch ein effizientes Management und die Berücksichtigung zahlreicher Aspekte. Es muss sich auf Bereiche konzentrieren, die für den finanziellen Erfolg entscheidend sind. Dies inkludiert eine Analyse des Marktpotenzials, die effektive Nutzung von Ressourcen und Innovationsmöglichkeiten sowie die Entwicklung eines nachhaltigen Lernumfelds. (Trendowicz, 2013)

Nee (2010) ergänzt, dass einfache Statusdiagramme und tägliche Meetings wirksame Instrumente zur Berichterstattung sind. Diese sind für Stakeholder jedoch häufig undurchsichtig und beantworten nicht direkt die Fragen nach dem erbrachten Wert, den Kosten, den verbrauchten Ressourcen oder den Erwartungen. Die agile Theorie geht davon aus, dass die Kunden in Projekten eng eingebunden werden und stets über den Status eines Projektes

informiert sind. In der Praxis ist dies aber nur selten der Fall. Darüber hinaus verfügen Kunden oft nicht über die Fähigkeiten oder Sprache, um dieses Verständnis an alle Stakeholder weiterzugeben. Es ist daher von entscheidender Bedeutung, dass agile Managementfachkräfte Strategien entwickeln, um die Leistung des Teams im Projekt zu verfolgen und Stakeholdern den Fortschritt zu berichten. Moderne Managementfachkräfte nutzen projektspezifische Dashboards, um die wichtigsten Parameter in einfachen und leicht zu analysierenden Grafiken darzustellen.

3.1.1 Planung

Um ein Projekt effektiv steuern zu können, muss der Ist-Status dem Soll-Status gegenübergestellt werden. Während geplante Werte im Projektplan definiert werden, können Daten zum tatsächlichen Fortschritt meist aus Tools im Zusammenhang mit dem Projektmanagement abgefragt werden. Um einen gesamtheitlichen Zustand zu ermitteln, müssen technische und funktionale Qualitätskriterien erfüllt sein. Zur Erhebung müssen unter anderem die bisher angefallenen Kosten und der Umfang der geleisteten Arbeit bekannt sein. Bezüglich der Kosten dürfen allerdings nur jene für die tatsächlich erbrachte Arbeit berücksichtigt werden. Material, Investitionsgüter und Aufwände für Reisen oder Versicherungen sind zwar wichtig, dürfen bei der Erhebung und Bewertung des Projektstatus jedoch nicht berücksichtigt werden. (Lewis, 2011)

Das Project Management Institute (2019) ergänzt, dass eine wirksame Planung ein genaues Verständnis über die Erwartungen, Anforderungen, Beschränkungen, Umfang und Ergebnisse eines Projektes voraussetzt. In einer agilen oder hybriden Umgebung können dazu auch Burn-Charts und Velocity-Berechnungen gehören. Diese Berichtsmethoden werden im folgenden Abschnitt 3.2 genauer beschrieben. Die Planung umfasst auch die Entwicklung der verschiedenen Projektmanagementpläne und ist der Ort, an dem zunächst festgelegt wird, wie und wann das EVM für ein bestimmtes Projekt eingesetzt wird. Während des Planungsprozesses müssen die Mittel zur Bewertung des physischen Arbeitsfortschritts dokumentiert werden. Das Thema EVM wird im Abschnitt 3.3 näher behandelt.

Die agile Planung erfolgt inkrementell. Wie in Abbildung 3-1 ersichtlich, basiert der Planungsansatz in der Softwareentwicklung meist auf Iterationen und Releases. Dabei wird ein übergeordneter Plan entwickelt, der die grundlegenden Merkmale und Funktionen angibt, die je Release enthalten sein sollen. Innerhalb dieser gibt es zwei oder mehr Iterationen, wovon jede einen zusätzlichen Nutzen für die Stakeholder zum Ziel hat. Der Nutzen kann in Form von Funktionen, vermindertem Risiko, Experimenten oder anderen Möglichkeiten ausgedrückt werden. Die Planung für die Arbeit in künftigen Versionen wird mit einem niedrigen Detailgrad durchgeführt, da sich Gegebenheiten und Anforderungen basierend auf Feedback von früheren Versionen ändern könnten. (Project Management Institute, 2021)

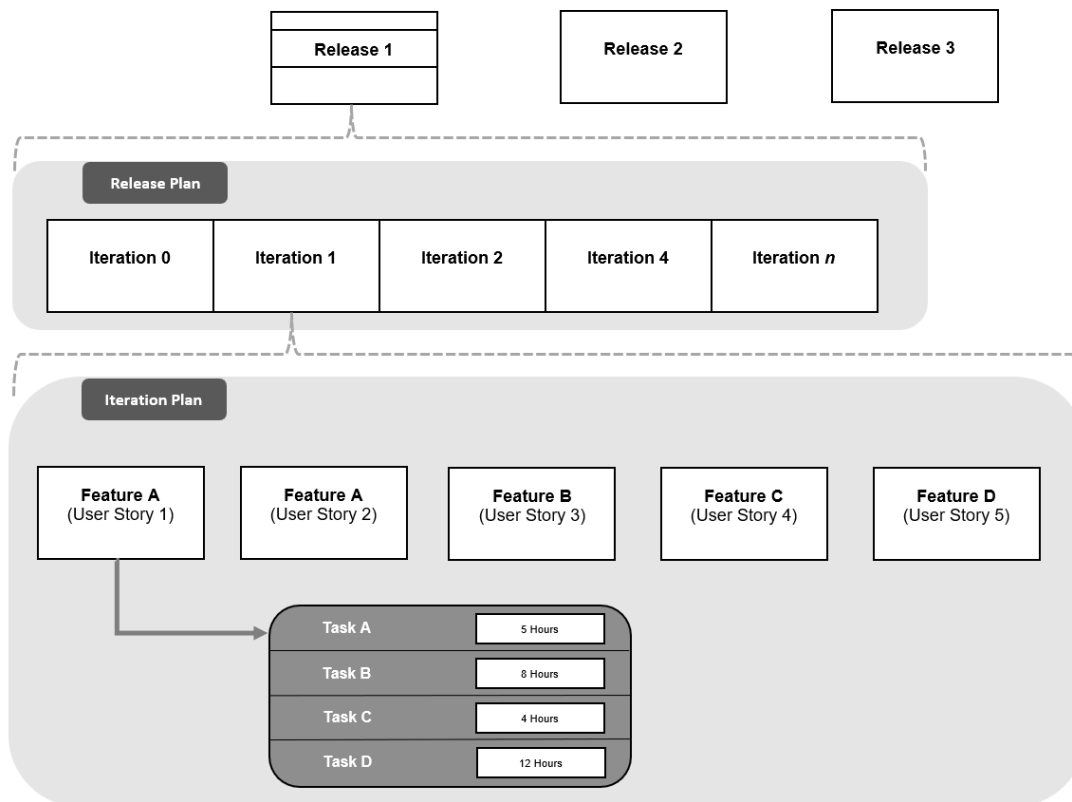


Abbildung 3-1: Planung auf Release- und Iterationsbasis (Project Management Institute, 2021)

Laut Stellman und Greene (2021) kann die Ermittlung der durchschnittlichen Geschwindigkeit eines Teams bei der Planung von Softwareentwicklungsprojekten helfen. Zwar ist es schwierig die Geschwindigkeit vor dem Start vorherzusagen, jedoch kann diese basierend auf Erfahrungen oder vergangenen Perioden ermittelt oder angenommen werden. Da Teams üblicherweise über viele Sprints hinweg mit einer konstanten Geschwindigkeit arbeiten, lassen sich ausgehend von Daten aus der Vergangenheit teamspezifische Vorhersagen erstellen.

3.1.2 Schätzung

Im Zusammenhang mit dem Aufwand bezieht sich die Schätzung typischerweise auf die Vorhersage des Aufwands, der erforderlich ist, um eine bestimmte Aktivität erfolgreich abzuschließen. Ein wirksames Projektmanagement erfordert eine zuverlässige Aufwands-, Kosten- und Termschätzung. Dabei benötigen die Managementfachkräfte eine Grundlage für die Entwicklung realistischer Pläne. Darüber hinaus benötigen sie eine überzeugende Basis für die Verhandlung von Projektbedingungen mit den Eignern und Kunden. Um realistische Pläne gegenüber den Anforderungen und Erwartungen von Beteiligten rechtfertigen zu können, sind möglichst genaue Schätzung notwendig. Unabhängig von dieser Notwendigkeit kalkulieren viele Softwareunternehmen immer noch mit unrealistischen Kosten und arbeiten mit knappen

Zeitplänen. Dies hat zur Folge, dass Projekte am Ende häufig hinter dem Zeit- und Kostenplan liegen oder gar nicht abgeschlossen werden können. (Trendowicz, 2013)

Schätzungen als Teil der Planung sind laut dem Project Management Institute (2021) eine quantitative Bewertung der wahrscheinlichen Menge oder des Ergebnisses. Sie werden in der Praxis für Variablen wie Kosten, Personalressourcen, Aufwand oder Dauer durchgeführt und können sich im Laufe des Projektes auf Grundlage der aktuellen Informationen und Umstände ändern. Das Project Management Institute (2021) nennt vier Aspekte, welche je nach Status und Fortschritt beeinflusst werden können:

1. Spannweite

Die Spannweite beschreibt die Unter- und Obergrenze einer Schätzung. Die Größe hängt stark vom Informationsgrad im Projekt ab. In frühen Phasen, wenn noch wenig Erfahrungen und Informationen über den Umfang, die Beteiligten, die Anforderungen oder die Risiken vorliegen, weisen Schätzungen eine große Spannweite auf.

2. Genauigkeit

Die Genauigkeit bezieht sich auf die Korrektheit einer Schätzung. Je geringer sie ist, desto größer ist die mögliche Spanne der Werte. Am Beginn von Projekten haben Schätzungen mangels Erfahrung üblicherweise eine geringere Genauigkeit als zu einem späteren Zeitpunkt.

3. Präzision

Die Präzision bezieht sich auf den Grad der Genauigkeit, welche mit der Schätzung verbunden ist. Zum Beispiel ist die Angabe einer Kalenderwoche als Zieltermin präziser als eine Schätzung auf Quartalsbasis. Die Abbildung 3-2 zeigt ein Beispiel für eine präzise Schätzung mit geringer Genauigkeit.

4. Zuversicht

Die Erfahrung in einem Projekt kann dazu beitragen, das erforderliche Maß an Vertrauen zu gewinnen, um valide Schätzungen zu erstellen. Bei neuen und sich ständig weiterentwickelnden Technologiekomponenten ist das Vertrauen eher gering, wodurch Schätzungen häufig ungenau sind.

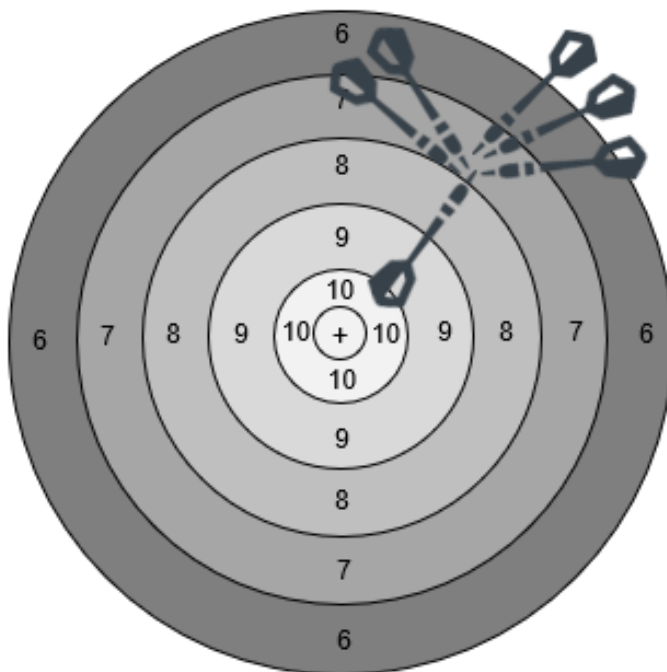


Abbildung 3-2: Geringe Genauigkeit und hohe Präzision (Project Management Institute, 2021)

Ein agiles Projekt wird nach Hill (2011) von einem Entwicklungsteam in einer Reihe von kurzen Iterationen durchgeführt. Die Dauer liegt in der Regel zwischen zwei Wochen und zwei Monaten. Ein wichtiger Aspekt bei der Schätzung von agilen Projekten ist die Ermittlung, welche und wie viele Aufgaben innerhalb einer Iteration abgeschlossen und ausgeliefert werden können. Dazu sind Methoden zur Bestimmung der Geschwindigkeit und den Durchsatz eines Teams erforderlich. Es wurden verschiedene Möglichkeiten entwickelt, um diese Größen in agilen Projekten zu messen und auszudrücken. Story Points können verwendet werden, um den Aufwand von Aufgaben in einem Vorhaben zu benennen. Die Entwicklung eines Zeitplanes setzt jedoch voraus, dass alle Stories für den betrachteten Zeitraum geschätzt wurden und dass die voraussichtliche Geschwindigkeit des Entwicklungsteams bekannt ist. Die Geschwindigkeit gibt dabei an, wie viele Story Points ein Team in einer Iteration ausliefert. An den Schätzungen und Zuweisungen der Story Points zu den Aufgaben sollten möglichst alle Entwickler mitwirken. Alternativ kann dies auch von einer kleineren Gruppe von Schlüsselentwicklern und Projektpersonal durchgeführt werden. Die Ermittlung der für ein Team erforderlichen Iterationen zur Umsetzung erfolgt durch eine Division der geschätzten Story Points mit der Team-Geschwindigkeit. Die Kosten können auf Grundlage des erforderlichen Aufwands und der Entwicklungskosten geschätzt und verfolgt werden. Wichtig ist dabei, dass die zu Beginn des Projektes erstellten Schätzungen nur als Größenordnungen angesehen werden, da Details zum Umfang und zur Komplexität möglicherweise noch nicht bekannt sind.

Dimitrov (2020) betont jedoch die Gefahr von versehentlichen Schätzfehlern mit Story Points in frühen Phasen des Vorhabens. Mangels Erfahrung in frühen Projektphasen können Aufgaben fälschlicherweise mit ähnlichen Aufgaben verglichen werden, ohne dabei alle Aspekte der

Komplexität betrachtet zu haben. So nennt er als Beispiel die fälschliche Annahme, dass ein Stein, der doppelt so groß aussieht wie ein anderer Stein, tatsächlich doppelt so groß ist. Aufgrund der Dreidimensionalität eines Steins ist das Volumen mathematisch gesehen jedoch achtmal so groß. Um dem Problem entgegenzuwirken, sollten Schätzungen nur für einen planbaren Zeitraum durchgeführt werden.

Rothman (2017) ergänzt, dass Stories häufig länger als einen Arbeitstag dauern können. In diesem Fall sollten Daten aus der Vergangenheit als Grundlage für die Schätzung verwendet werden. Dazu kann die Geschwindigkeit eines Teams über mehrere Iterationen hinweg gemessen oder eine Annahme getroffen werden. Unerfahrene Teams können bis zu sieben Iterationen benötigen, um die durchschnittliche Kapazität zu ermitteln.

Laut Cohn (2005) geht man häufig davon aus, dass die Geschwindigkeit eines Teams über mehrere Iterationen hinweg ähnlich bleibt. Zur Ermittlung der Schnelligkeit wird zumeist der gleitende Durchschnitt der Story Points von erledigten Aufgaben je Iteration herangezogen. Sofern ein Team noch nicht zusammengearbeitet hat oder neu im agilen Prozess ist, muss die Geschwindigkeit vorhergesagt werden. Zusammengefasst gibt es nach Cohn (2005) also folgende Methoden zur Schätzung der Geschwindigkeit bei der Release- oder Produktplanung:

- Berechnung eines gleitenden Durchschnitts auf Basis mehrerer vergangener Iterationen
- Annahme der Geschwindigkeit auf Basis der letzten Iteration
- Erstellen einer Prognose auf Basis von vergleichbaren Projekten

3.1.3 Budgetierung

Viele Projekte werden durchgeführt, um Einnahmen zu generieren. Andere Vorhaben hingegen dienen dazu, Kosten zu senken oder beides zu erreichen. Wenn der monetäre Wert und dessen Kosten abgeschätzt werden, können diese Informationen für Priorisierungen und Management-Tätigkeiten verwendet werden. Die Vorhersage des finanziellen Werts eines Objekts liegt meist in der Verantwortung des Product Owners, der häufig auch auf das Geschäftswissen und Prognosen zurückgreifen muss. (Cohn, 2005)

Laut Rothman (2017) ist die Kalkulation des Budgets eines agilen Projektes simpel, da die Personalressourcen in der Regel während der gesamten Durchlaufzeit gebunden sind. Es errechnet sich für Softwareprojekte aus optionalen Investitionskosten und den teamspezifischen Personalkosten über die gesamte Dauer. Sollten Personen das Team während des Vorhabens verlassen oder ihm beitreten, entstehen zusätzliche Kosten durch Verzögerungen und Ineffizienzen. Je wichtiger der finanzielle Aspekt in einem Vorhaben ist, desto wichtiger ist auch die Teamstabilität. Bei Bedarf können bereits im Vorfeld Schätzungen für die gesamte Laufzeit oder für bestimmte Abschnitte vorgenommen werden. Dabei sollte jedoch beachtet werden, dass sich Anforderungen auch während der Projektlaufzeit ändern können, und der Umfang des geplanten Produkts variiert.

Das Budget ergibt sich aus den aggregierten Schätzungen einzelner Aktivitäten. Um widerzuspiegeln, wann die Kosten anfallen, werden diese häufig über den Zeitplan verteilt. Auf diese Weise können Projektmanagementfachkräfte die zur Verfügung stehenden Mittel in einem bestimmten Zeitraum mit den geplanten Tätigkeiten in Einklang bringen. Wenn es in einer Periode beschränkte Ressourcen gibt, muss die Planung der Arbeit möglicherweise adaptiert werden. (Project Management Institute, 2021)

Lewis (2011) ergänzt, dass das Budget berechnet werden kann, indem der geschätzte Aufwand eines Projektes mit dem Kostensatz des Teams multipliziert wird. Übersteigt das erforderliche Budget den Betrag, der ausgegeben werden soll, können Komponenten identifiziert und entfernt werden, um die Kosten zu reduzieren. Jedoch ist die Ungenauigkeit von Schätzungen häufig ein Anlass dafür, dass Teams, welche mit agilen Softwareentwicklungsmethoden arbeiten, gänzlich auf die Erstellung von Budgets verzichten. Diese fehlende Bereitschaft kann eine Ursache dafür sein, dass die Kosten während eines Vorhabens außer Kontrolle geraten. Bei der Schätzung der Projektkosten spielt die Genauigkeit eine untergeordnete Rolle. Vielmehr soll es dazu dienen, ein finanzielles Ziel zu erstellen, welches im Laufe des Projektes verfolgt werden kann.

3.1.4 Monitoring

Das Scheitern von IT-Projekten, insbesondere von Softwareprojekten, ist häufig auf schlechtes Management zurückzuführen. Ein Abschluss kann finanziell unrentabel sein, selbst wenn alle inhaltlichen Ziele erreicht wurden. Umgekehrt können Vorhaben, auch wenn bestimmte Ziele nicht erreicht werden, wertvolle Informationen und Erfahrungen liefern. Um die Misserfolgsrate zu senken, sollen projektspezifische Daten erhoben und Metriken abgeleitet werden. (Marsanu, 2010)

Aus diesem Grund soll nach Stellman und Greene (2021) während der Durchführung eines Vorhabens eine gezielte Überwachung durchgeführt werden. Monitoring ist ein wichtiges Instrument, um Informationen mit Teammitgliedern zu teilen. Es gibt viele Dinge, die ein Softwareteam überwachen und messen kann. Die Wahl der zu überwachenden Parameter hängt davon ab, welche Fragen beantwortet und welche Probleme gelöst werden sollen. Das Ziel des Monitorings ist es, Teams dabei zu unterstützen, sich ein besseres Bild vom Projekt und dessen Status zu machen.

Laut Davis (2015) gibt es keine allgemeingültige Metrik, die eine Aussage darüber liefert, ob agile Teams bestmöglich arbeiten. Das Sammeln und Analysieren von Daten dient vielmehr dazu, um mehr über die Leistung von Softwareentwicklungsteams zu erfahren und gleichzeitig als Möglichkeit herauszufinden, ob bereits getroffene Maßnahmen wirksam waren. Eine Metrik kann aus einer einzelnen Datenquelle stammen oder eine Kombination von Daten mehrerer Quellen sein. Die Anwendung in einer Feedbackschleife parallel zur Entwicklung ermöglicht die Ableitung gezielter Anpassungen für Teams und verbessert die Kommunikation im Unternehmen. Die Schritte der Feedback-Schleife nach Davis (2015) sind in Abbildung 3-3 abgebildet und lauten:

1. **Sammeln:** Sammeln von Daten zum aktuellen Status
2. **Messen:** Analysieren der Daten und Ableiten von Maßnahmen
3. **Reagieren:** Anwendung der Maßnahmen
4. **Wiederholen:** Beobachtung der betroffenen Metriken

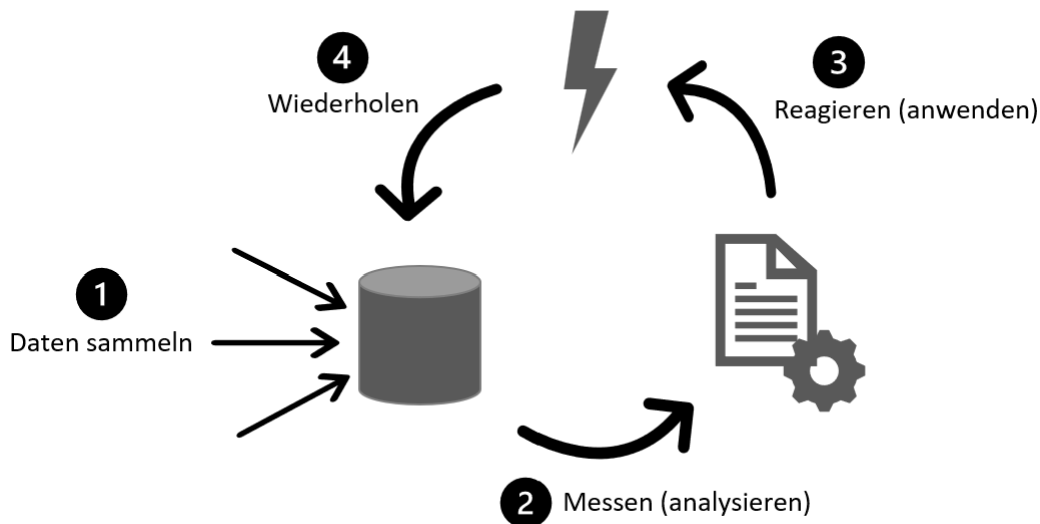


Abbildung 3-3: Anwendung von Metriken in einer Feedbackschleife (Davis, 2015)

Nach Nee (2010) besteht eine der Hauptstärken traditioneller Projektmanagementmethoden darin, dass sie ein robustes, erprobtes Instrumentarium an Kennzahlen und Berichtstechniken bereitstellen. Diese ermöglichen Stakeholdern einen Einblick in den Fortschritt und in die Herausforderungen der Entwicklungsarbeit. Auch im agilen Projektmanagement steht eine breite Palette von Metriken zur Verfügung, die verschiedene Arten von Fortschritten auf den Iterations- und Release-Ebenen agiler Projekte aufzeigen. Da agile Projektmanagementfachkräfte keine langfristigen, mehrmonatigen Projektpläne mit weit in die Zukunft reichenden Prognosen entwickeln, sind viele bekannte und nützliche Kennzahlen nicht verfügbar oder in Verwendung. Agile Teams verwenden andere Metriken wie Burndown-Diagramme, um den Projektstatus darzustellen. Ein erweitertes Verständnis für die agile Methodik ermöglicht es Führungskräften zu bestimmen, wie und wann sie Metriken am effektivsten einsetzen, um den Zustand eines Projektes abzufragen und darzustellen.

Cline (2015) bestätigt, dass Metriken sowohl in traditionellen als auch in agilen Projekten eingesetzt werden können, um den Fortschritt des Projektes in Bezug auf Kosten und Zeitplan zu überwachen. Wenn sich das Vorhaben von seinen Planwerten entfernt, sind Korrekturmaßnahmen erforderlich. Je früher die Projektmanagementfachkraft diese Abweichungen erkennt, desto geringer sind die Maßnahmen und umso schneller wird das Projekt wieder ins Lot kommen. Werden notwendige Gegenmaßnahmen zu spät ergriffen, muss das Vorhaben möglicherweise gestoppt werden, um den finanziellen Schaden zu begrenzen.

3.1.5 Forecasts

Zumeist möchten Personen, oft Führungskräfte, wissen, wann ein Projekt abgeschlossen sein könnte. Darüber hinaus ist auch die laufende Ermittlung der verursachten Kosten relevant, damit parallel weitere Ausgaben geplant werden können. Aus diesem Grund ist es wichtig, einen Status liefern zu können, welcher Antworten auf die projektspezifischen Fragen geben kann. (Rothman, 2017)

Im Manifest für agile Softwareentwicklung geht es laut Dimitrov (2020) im Wesentlichen um das Erstellen von Software. Dennoch sind die Grundsätze auch in einem breiteren und allgemeinen Kontext recht gut anwendbar. Das erste Postulat des Manifests lautet, dass Menschen und Interaktionen über Prozessen und Werkzeugen stehen. Indem Menschen und ihre Bedürfnisse mittels Tools und Forecasts unterstützt werden, bleibt man dem Kern der Aussage und einer besseren Arbeitsweise treu.

Laut dem Project Management Institute (2021) werden Forecasts primär dazu verwendet, um die aktuellen Projektpläne zu evaluieren und gegebenenfalls anzupassen, falls dies notwendig erscheinen sollte. Es wird zwischen folgenden Arten unterschieden:

- **Qualitativ:** Forecasts auf Basis von Expertenschätzungen
- **Kausal:** Forecasts auf Basis eines Zustands oder zukünftigen Ereignisses
- **Quantitativ:** Forecasts auf Basis von Informationen aus der Vergangenheit

Durch moderne Softwareentwicklungstechniken wird die Zuverlässigkeit von Entwicklungsteams beträchtlich gesteigert. Dies ermöglicht neue Methoden für projektspezifische Forecasts. Bereits mit geringem Investment zur Erhebung von Projektdaten können einfache und kostengünstige Vorhersagen in Softwareprojekten erstellt werden. Dies verbessert das Projektumfeld für alle Beteiligten, da sich Projektteams auf die Abwicklung des Projektes konzentrieren können, während das Management einen aussagekräftigen Ausblick im Rahmen eines effektiven Projektcontrollings bekommt. (Dimitrov, 2020)

Metriken geben einen Überblick über eine Vielzahl von Aspekten wie Ressourcenverbrauch, Kosten, Leistung und Risiken. Auf der Grundlage historischer Daten über Projekte, die in der Vergangenheit durchgeführt wurden, lässt sich ein System erstellen, das Informationen über die Chancen für den erfolgreichen Abschluss liefert. Ebenfalls kann der Status von Entwicklungen, die unter ähnlichen Bedingungen durchgeführt wurden, erhoben werden, um Maßnahmen auf der Grundlage der gewonnenen Erkenntnisse abzuleiten. (Marsanu, 2010)

Forecasts werden nicht erstellt, um zu beweisen, dass etwas richtig oder falsch ist, oder um Aufzeichnungen zu führen und Verantwortliche für etwaige Abweichungen zu finden. Der Zweck besteht darin, wichtige Entscheidungen zu einem möglichst frühen Zeitpunkt im Projektverlauf treffen zu können. Als Basis dafür muss die Bereitschaft bestehen und notwendige Rahmenbedingungen geschaffen werden, um Entscheidungen ableiten zu können, welche die Parameter während der Umsetzungsphase ändern können. Die Ermittlung der Geschwindigkeit

in Form von Story Points ist von großer Bedeutung bei der Erstellung projektspezifischer Forecasts, da sie einen Anhaltspunkt für das Entwicklungstempo eines Teams gibt. Ein nachhaltiges und konstantes Tempo gilt als Grundvoraussetzung für die Erstellung von Forecasts. (Dimitrov, 2020)

Lewis (2011) nennt mehrere Möglichkeiten, um Endergebnisse von Projekten vorherzusagen. Neben der Planung auf der Grundlage der bisherigen Erkenntnisse, kann die Berechnung der vorhergesagten Ergebnisse auf EVM Daten basieren. Die gebräuchlichste und am meisten akzeptierte Methode ist die Verwendung des kumulativen Kostenentwicklungsindex zum Zeitpunkt der Fertigstellung.

Dimitrov (2020) beschreibt, dass bei agilen Projekten mit schätzbarem Umfang bereits in frühen Phasen Fortschrittsmessungen durchgeführt werden können. Während eine erste Messung bereits Auskunft über den erwarteten und tatsächlichen Fortschritt gibt, kann durch weitere Durchführungen, jeweils am Ende einer Iteration, bereits ein Trend für den weiteren Verlauf erkennbar werden. Zwar sind Vorhersagen in früheren Projektstadien tendenziell ungenau, jedoch dienen sie primär dazu, frühzeitige und effektive Entscheidungen zu treffen. Durch zusätzliche Messungen kann die Genauigkeit von Forecasts zwar erhöht werden, jedoch nimmt die Nützlichkeit einer Vorhersage mit zunehmender Zeit ab. In der Abbildung 3-4 wird dargestellt, dass durch frühzeitige Messungen mehr Zeit für effektive Entscheidungen bleibt.

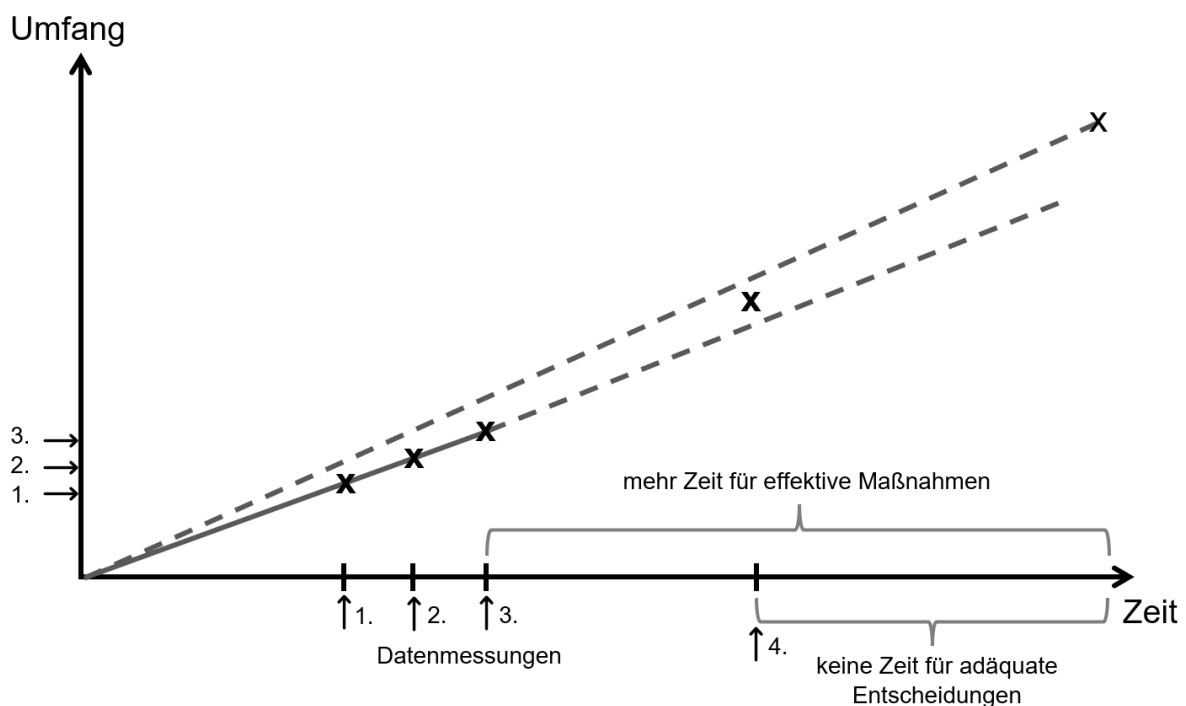


Abbildung 3-4: Vorhersagen in agilen Projekten (Dimitrov, 2020)

3.2 Berichte, Metriken und Kennzahlen

Nach Davis (2015) stellen Metriken im Kontext der Softwareentwicklung jene Daten dar, die während des Lebenszyklus eines Produkts abgerufen werden können. Sie spiegeln im Wesentlichen die Leistung von Softwareentwicklungsteams wider und können aus einer einzigen Datenquelle stammen oder eine Kombination von Daten aus mehreren Datenquellen sein. Jeder verfolgbare Datenpunkt in einem Projekt kann schließlich zu einer Metrik werden, welche ein wertvolles Werkzeug zur Verbesserung der Kommunikation und der Zusammenarbeit sein kann. Auf Basis dieser Daten können in weiterer Folge Leistungsindikatoren und Trends abgeleitet werden, die auf dem Verhalten von Teams basieren. Durch diesen Zusammenhang lässt sich nachvollziehen, wie sich teaminterne Veränderungen und getroffene Maßnahmen auf die für das Team wichtigen Indikatoren auswirken.

Laut Jones (2008) gibt es für Softwaremessungen eine breite Palette an Anwendungsgebieten. Er beschreibt das Ziel der Anwendung von Messungen wie folgt:

„The goal of applied software measurement is to give software managers and professionals a set of useful, tangible data points for sizing, estimating, managing, and controlling software projects with rigor and precision.” (Jones, 2008, S. 2–3)

Nach Cole und Scotcher (2015) wurden Metriken in der Vergangenheit häufig als Management-Kontrollinstrument benutzt, um Druck aufzubauen und den Output von Teams zu erhöhen. Aus diesem Grund war der Begriff im Kontext der traditionellen Softwareentwicklung negativ behaftet und schlecht akzeptiert. Agile Methoden und Frameworks haben die Implementierung von Metriken jedoch revolutioniert. Gemeinsam mit Kennzahlen gehören sie dem Team und werden von ihm verwendet, um die Arbeit zu erledigen. Aufgrund dieser Schwerpunktverlagerung werden Leistungsindikatoren besser angenommen, da Entwicklungsteams auch von dieser Art von Information profitieren können. Insbesondere können Methoden wie Scrum und Kanban ohne bestimmte Kennzahlen nicht richtig funktionieren. Agile Metriken und Kennzahlen sind also nach wie vor von großem Nutzen für das gesamte Unternehmen, werden aber nicht mehr primär als Instrument zur Kontrolle und zum Erzwingen zusätzlicher Entwicklungsleistung angesehen. Dadurch, dass das Team für diese Messgrößen verantwortlich ist, sind sie wesentlich zuverlässiger. Besonders deutlich wird dies bei der Messung des Outputs. Beispielsweise wird die Information über die Geschwindigkeit genutzt, um vorherzusagen, was es innerhalb der vorgegebenen Zeitvorgaben liefern kann. Die Vorhersage ist zuverlässiger, weil sie auf Erfahrungen beruht und nicht willkürlich angenommen wird. Die Geschwindigkeit kann auch verwendet werden, um zu berechnen, welcher Wert in der nächsten Iteration geliefert werden kann. Dies ist möglich da bekannt ist, wie viel das Team in diesem Zeitraum kostet, und welcher Umfang im Durchschnitt geliefert wird.

Die Entscheidung, was und wie oft gemessen werden soll, hängt laut dem Project Management Institute (2021) von der Natur des geplanten Projektes ab. Während produktbezogene Messgrößen für die zu entwickelnde Software individuell erstellt und verwendet werden können,

kommen häufig auch Metriken zum Einsatz, die sich auf den Zeitplan und das Budget beziehen. Diese werden meist durch organisatorische Standards bestimmt und beziehen sich auf zuvor geplante Zeitpläne und Budgets, mit welchen die tatsächlichen Ergebnisse verglichen werden.

Nach Dimitrov (2020) trägt der gezielte Einsatz von Metriken zur Verbesserung der Vorhersagbarkeit in Projekten bei. Zugleich wird die Wahrscheinlichkeit, dass ein Forecast dem tatsächlichen Ergebnis entspricht, erhöht. Die dadurch resultierende Steigerung der Produktivität kann als wirksame Projektsteuerung betrachtet werden. Andererseits kann das Erzwingen von messbaren Prozessen in agilen Umfeldern auch kontraproduktiv sein und Teams bei der Umsetzung von Projekten einschränken.

Nachfolgend werden Berichte, Metriken und Kennzahlen, welche in der Literatur im Zusammenhang mit Softwareprojekten häufig diskutiert werden, beschrieben.

3.2.1 Velocity

Laut Hill (2011) wird für die Erstellung von Schätzungen in agilen Projekten vorausgesetzt, dass im Vorhinein bekannt ist, wie schnell das Entwicklungsteam liefern kann. Die Geschwindigkeit wird auch als Team Velocity oder Velocity bezeichnet und für gewöhnlich in Story Points je Iteration angegeben. Sie muss mit dem Umfang des Projektes abgestimmt sein, damit Stories einzelnen Projektiterationen zugeordnet werden können und um den Zeitplan und die Kosten für das Projekt abschätzen zu können. Üblicherweise wird die Velocity auf Basis der bisherigen Projektleistung innerhalb derselben Organisation und für dasselbe Entwicklungsteam ermittelt. Zur Bestimmung anhand eines vergangenen Projektes ermittelt man die durchschnittliche Anzahl von Story Points, die von einer Projektiteration geliefert werden.

Cohn (2005) ergänzt, dass die Velocity nach einer Alles-oder-Nichts-Regel berechnet werden sollte. Wenn eine Story fertiggestellt ist, zählt das Team bei der Berechnung die gesamte Schätzung. Wird eine Story während einer Iteration nur teilweise fertiggestellt, darf sie bei der Ermittlung nicht berücksichtigt werden.

Wenn Teams eine konstante Geschwindigkeit erreicht haben, kann diese nach Rothman (2017) als Grundlage für Schätzungen in agilen Projekten verwendet werden. Eine konstante Velocity wird jedoch oft erst nach mehreren Iterationen erreicht. Sie soll als Kapazität für ein Team betrachtet werden, und Auskunft über die Anzahl von Story Points geben, welche in einer Iteration durchschnittlich geliefert werden konnten. Verschiedene Umstände können jedoch auch dazu führen, dass die Velocity variiert. Üblicherweise dann, wenn das Team neue Bereiche oder Technologien kennenlernt, sich die Stories in ihrer Größe und Komplexität stark unterscheiden, oder wenn die Iterationsdauer verändert wird.

Laut Lester (2017) neigen Teams dazu, über mehrere Sprints hinweg ein konstantes Tempo zu entwickeln, womit die Planung künftiger Iterationen ermöglicht wird. Jedoch gibt es keine Garantie dafür, dass im folgenden Sprint der erwartete Wert, basierend auf vergangenen Iterationen, geliefert wird. Es kann immer wieder vorkommen, dass Teams bestimmte Stories falsch einschätzen oder unerwartete Probleme auftreten. Darüber hinaus müssen Abwesenheiten wie

Urlaube, Krankenstände oder generelle personelle Veränderungen berücksichtigt werden, welche Einfluss auf das Projekt haben.

Davis (2015) beschreibt, dass Teams in der Lage sein sollten, jene Arbeit zu erledigen, welche vor Beginn einer Iteration geschätzt wurde. Die Velocity soll also jenem Umfang entsprechen, zu welchem sich ein Team während der Iterationsplanung verpflichtet hat. Inwieweit man den iterationsbasierten Schätzungen nachkommt, kann in einem Velocity-Diagramm abgebildet werden, welches die geschätzten und tatsächlich erledigten Story Points je Iteration darstellt. Aus dem Grad der Abweichung können verschiedene Eigenschaften des betroffenen Entwicklungsteams abgeleitet werden. Das Velocity-Diagramm gibt also Aufschluss darüber, wie gut ein Team die Arbeit einschätzen kann, wie konsequent es seine Arbeit erledigt und wie zuverlässig es bezüglich des festgelegten Aufwandes ist.

3.2.2 Burndown- und Burnup-Diagramme

Viele Teams, die nach agilen Vorgehensweisen wie Scrum arbeiten, nutzen Burndown-Diagramme, um den Fortschritt für einen bestimmten zeitlichen Abschnitt zu messen. Als Basis wird in der Regel eine Ideallinie hinzugefügt, welche den geplanten Fortschritt über den ausgewählten Zeitraum, beispielsweise einem Sprint, widerspiegeln soll. Im Wesentlichen soll die tatsächlich vollbrachte Arbeit zum gegebenen Zeitpunkt und damit der Fortschritt im Vergleich zum Plan veranschaulicht werden. Am Ende der jeweiligen Iteration soll die ausstehende Arbeit optimalerweise gleich Null sein. (Rothman, 2017)

Laut Cohn (2005) können teamspezifische Burndown-Diagramme auch auf Release-Ebene angewandt werden. In diesem Fall wird die Anzahl der Story Points oder idealen Tage veranschaulicht, die innerhalb des Releases zu Beginn jeder Iteration verbleiben. Dabei ist festzuhalten, dass der Burndown eines Teams nie vollkommen konstant ist. Er schwankt aufgrund von unterschiedlichen Faktoren wie adaptierten Schätzungen, unklaren Anforderungen, oder Änderungen des Umfangs. Es kann sogar vorkommen, dass ein Burndown-Diagramm einen sogenannten „Burnup“ während einer Iteration zeigt. Dies kann vorkommen, wenn die verbleibende Arbeit unterschätzt wurde oder sich der Umfang der ausstehenden Tätigkeiten erhöht hat. Der Schlüssel zur Interpretation eines Release-Burndown-Diagramms liegt darin, zu verstehen, dass es den Fortschritt des Teams zeigt, abzüglich der Arbeit, die dem Release hinzugefügt wurde.

Bei iterationsbasierten Projekten können Burndown-Diagramme verwendet werden, um zu sehen, wohin sich das Projekt im Laufe der Zeit entwickelt. Die Abbildung 3-6 zeigt ein Beispiel für ein Burndown-Diagramm, in dem ein Team plante 37 Story Points zu liefern und nach dem zehnten Tag noch sieben Story Points übrig hat. Die gepunktete Burndown-Linie stellt den geplanten Fortschritt dar, während die durchgehende Linie den tatsächlichen Fortschritt widerspiegelt. (Project Management Institute & Agile Alliance, 2017)

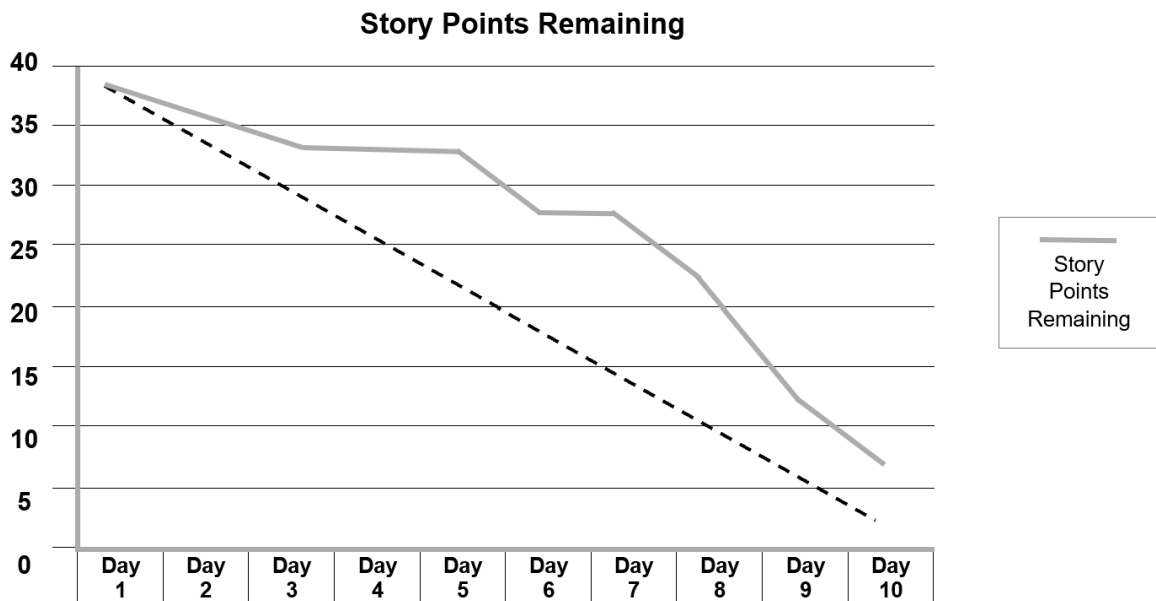


Abbildung 3-5: Burndown-Diagramm (Project Management Institute & Agile Alliance, 2017)

Rothman (2017) empfiehlt die Verwendung von Burnup-Diagrammen, wenn das Team in einem iterationsbasierten agilen Ansatz arbeitet. Sie sollen den Teams dabei helfen, die Velocity und den Fertigstellungsgrad zu ermitteln und darzustellen. Während Burndown-Diagramme dazu verwendet werden, um die verbleibenden Aufwände darzustellen, zeigen Burnup-Diagramme die bereits erledigte Arbeit auf. Die Abbildung 3-7 zeigt ein Burnup-Diagramm, welches die erledigte Arbeit zum jeweiligen Zeitpunkt im Vergleich zur Idealline darstellt.

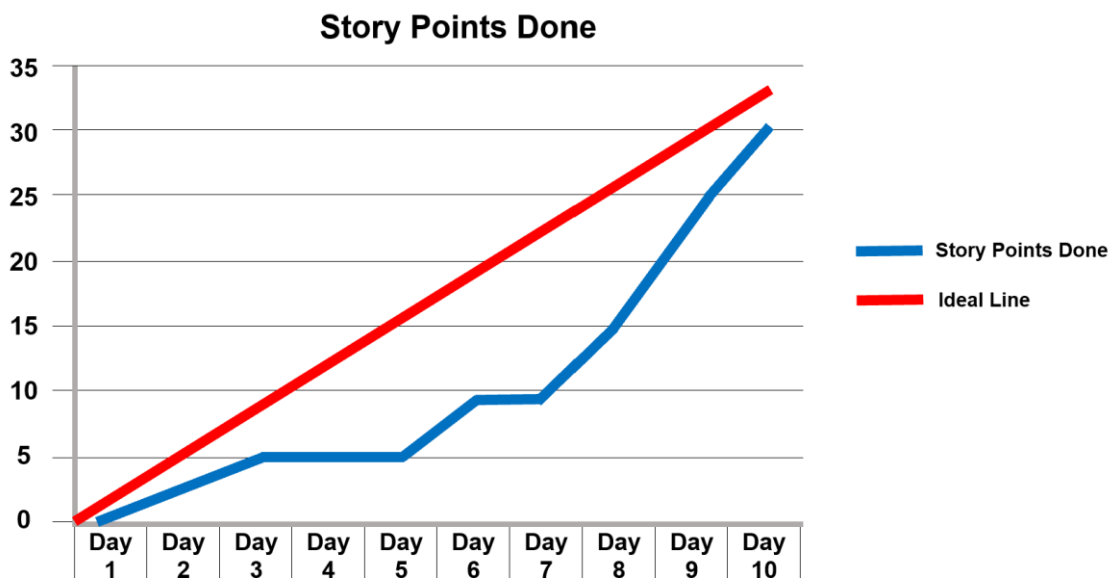


Abbildung 3-6: Burnup-Diagramm (Rothman, 2017)

McKenna (2016) beschreibt, dass Burndown-Diagramme dazu verwendet werden, um den verbleibenden Aufwand in einem Sprint oder Release darzustellen. Es unterstützt primär bei der Vorhersage, ob das Team alle Aufgaben, die es sich für die aktuelle Iteration vorgenommen hat, abschließen wird. Burnup-Diagramme sollen hingegen aufzeigen, was bis zu einem bestimmten Zeitpunkt im Projekt geliefert wurde. Der Burnup auf Release-Ebene funktioniert nach dem gleichen Prinzip. Im Diagramm sollen jedoch nur Story Points von bereits fertiggestellten Aufgaben dargestellt werden. Das Release-Burnup-Diagramm stellt somit den Umfang von vollständig erledigten Aufgaben dar und wie viele Story Points noch im Release-Backlog verbleiben.

3.2.3 Lead Time und Cycle Time

Laut Stellman und Greene (2021) messen viele Teams die sogenannte Lead Time für Stories innerhalb eines Projektes oder einer Version. Dabei handelt es sich um eine Messung der durchschnittlichen Zeit, die zwischen der Erstellung der Anforderung und der Auslieferung des Ergebnisses vergeht. Zur Ermittlung muss das Startdatum notiert werden, wenn eine Anfrage erstellt wird. Das Enddatum entspricht dem Datum der Freigabe einer neuen Version der Software, welche diese Anforderung enthält. Die Lead Time entspricht also jener Zeit, die zwischen dem Start- und dem Enddatum einer Aufgabe verstreicht. Zur Bestimmung wird die Summe aller gemessenen Lead Times durch die Anzahl der Anforderungen im betrachteten Zeitraum geteilt. Es gibt jedoch auch noch andere Möglichkeiten zur Berechnung. So können beispielsweise umfangreichere Stories stärker gewichtet in die Kalkulation einfließen.

Nach Davis (2015) legen vor allem Teams, welche nach Kanban praktizieren, großen Wert auf die Ermittlung der Lead Time, da diese üblicherweise großen Wert auf den Durchsatz legen. Von Scrum-Teams wird sie häufig ergänzend zu anderen Metriken ermittelt. Die Velocity beschreibt das Verhältnis des geplanten zum tatsächlich erledigten Umfang in einer bestimmten Zeitspanne. Die Lead Time gibt hingegen Auskunft darüber, wie lange es dauert, einzelne Stories oder ganze Releases zu erledigen. Sie ist eine wertvolle Metrik und ein guter Indikator dafür, um Ineffizienzen im Gesamtprozess aufzuzeigen.

Rothman (2017) beschreibt die Lead Time einer Story als jene Zeit, welche zwischen der Erstellung des entsprechenden Backlog-Elements im Projekt und der Nutzung durch den Kunden vergeht. Dies inkludiert auch die etwaige Zeit zum Testen von Funktionen in mehrstufigen Test-Umgebungen. Die Cycle Time hingegen beschreibt die Zeitspanne, beginnend vom Zeitpunkt, an dem eine Story aktiv vom Team bearbeitet wird, bis zu dem Zeitpunkt, an dem sie erledigt wird. Sie kann Teams dabei helfen, den durchschnittlichen Durchsatz zu erheben und etwaige Engpässe zu erkennen. Darüber hinaus ist die Cycle Time nützlich für den Product Owner, da unerwartet hohe Zeiten darauf hindeuten können, dass Stories zu groß sind.

Laut dem Project Management Institute und der Agile Alliance (2017) kommt die Cycle Time vor allem dann zum Einsatz, wenn Teams von externen Personen oder Gruppen abhängig sind. Mit ihr kann ermittelt werden, wie lange es dauert, bis das Team die Arbeit abgeschlossen hat. Die

Lead Time hingegen inkludiert zusätzlich die Zeit vor der Umsetzung und jene zwischen Erledigung und Kundennutzung. In Abbildung 3-8 werden die jeweiligen Zeitspannen für Lead Time und Cycle Time anhand eines Task-Boards veranschaulicht.

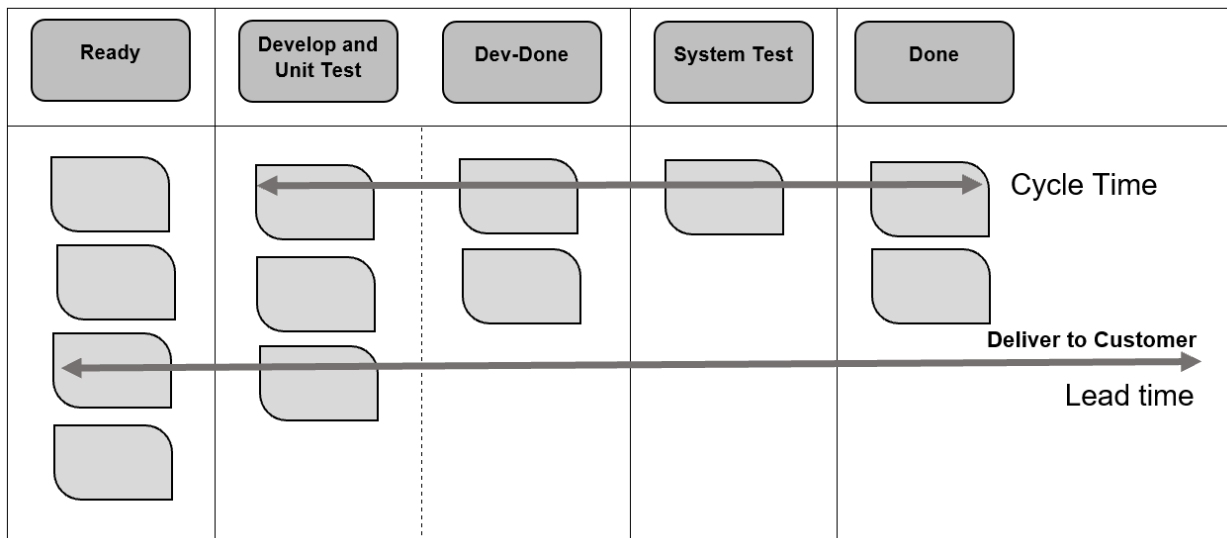


Abbildung 3-7: Lead und Cycle Time (Project Management Institute & Agile Alliance, 2017)

3.2.4 Cumulative Flow

Herkömmliche Instrumente zeigen den Fortschritt, die geplante Arbeit oder die Geschwindigkeit eines Teams auf, jedoch nicht, in welchem Status sich die Aufgaben in einem Projekt befinden. Genau zu diesem Zweck kommt das Cumulative Flow-Diagramm zum Einsatz. Es zeigt die kumulierte Anzahl der Aufgaben je Projektstatus entlang der Zeitachse eines Projektes. Betrachtet man lediglich eine einzige Iteration, sollte es am Ende ausschließlich erledigte Aufgaben darstellen, da sich Entwicklungsteams dazu verpflichten, die geplanten Stories am Ende einer Iteration fertigzustellen. Häufig plant der Product Owner aber im Voraus, um dem Team einen Ausblick auf die anstehende Arbeit zu geben. (Rothman, 2017)

Das Project Management Institute und die Agile Alliance (2017) ergänzen, dass das laufende Hinzufügen von Aufgaben in einem Projekt zu Problemen in Entwicklungsteams führen kann. Wenn Teams mehr Arbeit in Bearbeitung haben, als sie abschließen können, wird die Bereitstellung von Funktionen verzögert. Erhöhte Lead und Cycle Times führen letztendlich häufig zu zusätzlichem Druck, um in derselben Zeit mehr Funktionen auszuliefern.

Müller (2021) bezeichnet das Cumulative Flow-Diagramm als gebräuchliches Instrument im Reporting agiler Projekte. Es stellt die Anzahl der Arbeitseinheiten je Prozessschritt dar, wobei diese nicht zwingend in Story Points angegeben werden muss. Der Cumulative Flow soll in Scrum-Teams als Ergänzung zum Burndown-Diagramm zum Einsatz kommen und bei der

Planung oder bei Retrospektiven dabei helfen, Engpässe zu erkennen und Optimierungsmaßnahmen abzuleiten.

Nach A. T. Pham und Pham (2013) soll das Cumulative Flow-Diagramm vor jedem täglichen Meeting aktualisiert werden, um den Fortschritt des Teams aufzuzeigen. Das Diagramm zeigt, ob sich die Durchlaufzeit verbessert oder der Durchsatz steigt.

Nach Davis (2015) muss ein plötzlicher Anstieg von unerledigten Elementen nicht zwingend darauf hindeuten, dass ein Team Schwierigkeiten hat, die offene Arbeit abzuschließen. Dies kann auch passieren, wenn die Aufgaben zu groß sind und das Team beginnt, diese in kleinere Einheiten zu zerlegen. Nützlich ist das Cumulative Flow-Diagramm vor allem bei der Identifikation von Engpässen, indem visuell dargestellt wird, wann die Anzahl der Aufgaben in einem bestimmten Prozessschritt zugenommen hat.

3.2.5 Finanzielle Kennzahlen

Nach Hartmann und Dymond (2006) wird Unternehmen durch den Einsatz agiler Methoden und regelmäßigen Lieferungen von Softwareprodukten eine rasche und regelmäßige Erwirtschaftung von Erträgen ermöglicht. Die Anwendung wertbasierter Metriken soll dabei helfen, intelligente Investitionen zu tätigen und den Output von Teams quantifizierbar zu machen. Im Kontext der Softwareentwicklung wird für marktreife Produkte oder Inkremente ein Wert definiert. Je schneller Ergebnisse geliefert werden können, desto schneller wird ein Wert für die Kundinnen und Kunden geschaffen. Er kann entweder am Ende einer Iteration oder zum Zeitpunkt der Freigabe einer Produktversion generiert werden. In letzterem Fall müssen die Teilwerte aller Iterationen innerhalb einer Version summiert werden. Obwohl es üblicherweise unterschiedliche Kennzahlen zum Geschäftswert gibt, empfiehlt es sich innerhalb einer Organisation auf eine Schlüsselkennzahl zu fokussieren. Beispiele für gängige Kennzahlen sind der Return on Investment (ROI), die Net Present Value (NPV) oder die Internal Rate of Return (IRR).

Der ROI spiegelt die Kapitalrendite wider und ist ein Maß für den monetären Erfolg eines Projektes. Die Kennzahl ergibt sich aus dem Verhältnis zwischen dem erwirtschafteten und dem für die Herstellung des Produkts investierten Betrag. Durch iterative Ansätze können Produkte in mehreren Iterationen ausgeliefert werden. Somit können bereits während der Entwicklung des Produkts Einnahmen erzielt werden. Wenn Entwicklungen bereits während des Projektes in Rechnung gestellt werden, anstatt auf das Ende der Projektlaufzeit zu warten, werden getätigte Investitionen und Vorleistungen zur Erstellung des Produktes viel früher gedeckt. (Pries & Quigley, 2011)

Die NPV entspricht dem sogenannten Nettogegenwartswert. Zur Ermittlung und Interpretation ist ein Verständnis für den Zeitwert eines Geldes von wesentlicher Bedeutung. Die Berechnung erfolgt, indem alle zukünftig erwarteten Cashflows auf die heutigen Werte abgezinst und zusammengefasst werden. Dies stellt eine Schätzung dessen dar, was ein Projekt voraussichtlich zum Nettogewinn der Organisation beitragen wird. (Moran, 2016)

Die IRR wird oft als interner Zinsfuß bezeichnet und beschreibt den Zinsfaktor, wenn der Nettogegenwartswert Null entspricht. Anstatt eines Geldbetrages wird also ein Zinssatz errechnet, den ein Projekt mindestens erreichen soll. Da Unternehmen häufig Schwellenwerte für die Berechnung des internen Zinssatzes festlegen, wird von der Anwendung abgeraten, wenn Projekte mit erheblichen Ressourcenunterschieden verglichen werden sollen. (Pries & Quigley, 2011)

3.3 Earned Value Management

Laut Lester werden agile Projekte zunehmend auf Basis von Daten gesteuert, um sicherzustellen, dass die Projektziele erreicht und auf die Anforderungen der Kunden abgestimmt werden. Der Projektfortschritt wird dabei häufig in Echtzeit verfolgt und mittels Kundenfeedback verifiziert, wobei funktionierende Software oder ganze Lösungen die Basis darstellen. Zudem gewinnen die Aspekte Governance und Compliance im Zuge des agilen Projektmanagements zunehmend an Bedeutung. Damit soll die Verbesserung von Prozessen sowie die Einhaltung von Geschäftszielen und Vorschriften erleichtert werden. (Lester, 2017)

In diesem Abschnitt wird zunächst ein Überblick über das EVM als Instrument des Projektcontrollings gegeben und die Begründung für die Auswahl dieses Werkzeuges beschrieben. Darüber hinaus wird die wesentliche Terminologie sowie der Zweck erläutert, bevor darin vorkommende Kennzahlen und deren Kalkulation veranschaulicht werden. Im Anschluss darauf wird näher auf den Einsatz von EVM in agilen Projekten eingegangen, um die Unterschiede zur traditionellen Variante darzustellen.

3.3.1 Entscheidung für Earned Value Management

Nach Wanner (2019) gibt es die Projektsteuerung bereits seit mehreren Jahrhunderten in unterschiedlichsten Ausprägungen. Der Großteil der Methoden ist jedoch für moderne Projekte nicht mehr geeignet, wodurch der Bedarf entstand, zeitgerechte Steuerungsmethoden zu entwickeln. Aus diesem Grund wurde das EVM vom amerikanischen Verteidigungsministerium, welches bis heute noch als Schrittmacher des modernen Projektmanagements gilt, entwickelt. EVM hat sich inzwischen zu einer Best Practice in der Projektsteuerung entwickelt, die neben der effektiven Überwachung von Projekten auch objektive Werte zum Projektfortschritt und Frühwarnsignale durch Trends und statistische Vorhersagen liefert.

Laut Popović et al. (2019) verfolgt der agile Ansatz im Projektmanagement das Ziel der Maximierung des ROI durch frühzeitige und kontinuierliche Wertschöpfung. Aus den agilen Werten und Grundprinzipien geht jedoch nicht hervor, auf welche Weise die Kosten eines Projektes verwaltet und nachverfolgt werden sollen, um den tatsächlichen ROI zu ermitteln. Die üblichen iterationsspezifischen Metriken wie Burndown- und Burnup-Diagramme liefern diese Art von Daten nicht. Eine mögliche Lösung besteht jedoch darin, die sogenannte EVM Methode auf Basis der Velocity, dem geschätzten Backlog und den Entwicklungskosten zu implementieren.

Herkömmliche agile Metriken, wie unter Abschnitt 3.2 beschrieben, können zwar wertvoll bei der Erhebung des Projektfortschritts sein, sie liefern jedoch nicht genügend Information zur finanziellen Lage eines Projektes. Mit agilen Praktiken zur Projektsteuerung können in der Regel keine Aussagen über die erwarteten Kosten bei Fertigstellung eines Releases getroffen werden. Darüber hinaus fehlen häufig finanzielle Kennzahlen zur Unterstützung des Unternehmens beim Kostenmanagement, um beispielsweise den ROI oder andere Finanzinformationen zu erheben. Das Berichtswesen, welches üblicherweise in Scrum und anderen agilen Ansätzen verwendet wird, bietet auch keine Schätzungen oder Prognosen des Kostenverlaufs oder zu den Kosten zur Fertigstellung eines Releases, wodurch Entscheidungen wie das Ändern von Anforderungen erschwert werden. Mit dem Einsatz von EVM können die fehlenden Informationen bezüglich der Kosten ermittelt werden. Es kann nicht nur in Projekten mit traditionellen Vorgehensweisen eingesetzt werden, sondern auch ergänzend zu gewöhnlichen Berichten in agilen Projekten. (Project Management Institute, 2019)

3.3.2 Definition und Abgrenzung

Nach Kim, Wells und Duffey (2003) ist EVM eine Projektmanagementmethode zur Steuerung von Projekten, die auf der Messung der Arbeitsleistung basiert. Es ist ein System, das die endgültigen Kosten und die Dauer des Projektes vorhersagt, indem es die geleistete Arbeit mit der geplanten vergleicht. EVM kann gleichzeitig als Warnwerkzeug zur Früherkennung von Problemen während der Projektlaufzeit eingesetzt werden.

Laut dem Project Management Institute (2019) können durch die Verknüpfung der Plankosten mit den tatsächlichen Kosten und dem Arbeitsfortschritt zuverlässige Aussagen über die zukünftige Leistung von Projekten erstellt werden. Zeitnahes Feedback auf Grundlage des Projektzustandes ist entscheidend für das erfolgreiche Management von Projekten. Als Voraussetzung für den Einsatz von EVM sind eine sorgfältige Planung des Projektumfangs und die Überwachung des Fortschritts über den gesamten Projektlebenszyklus notwendig. Als wesentliche Parameter gelten zum jeweiligen Zeitpunkt einer Messung der geplante Fortschritt, der tatsächliche Fortschritt und die bisherigen Kosten.

Wanner (2019) beschreibt das EVM als Monitoring-Werkzeug, welches die Ermittlung von Leistungsindikatoren und die Erstellung von projektspezifischen Forecasts ermöglicht. Im Fokus steht dabei die Anwendung von Managementkennzahlen zur Bestimmung des Status, zur Erstellung von Prognosen und zur Definition von Korrekturmaßnahmen. EVM wird häufig synonym mit den Begriffen *Earned Value Analyse (EVA)* und *Earned Value Management System (EVMS)* verwendet. Während die EVA nur einen Teil der Methodik im EVM beschreibt, ist unter EVMS das gesamte System von integrierten Prozessen zu verstehen, welches Normen und Kriterien für das EVM implementiert. Die Begriffe und deren Hierarchie sind in der Abbildung 3-10 dargestellt.

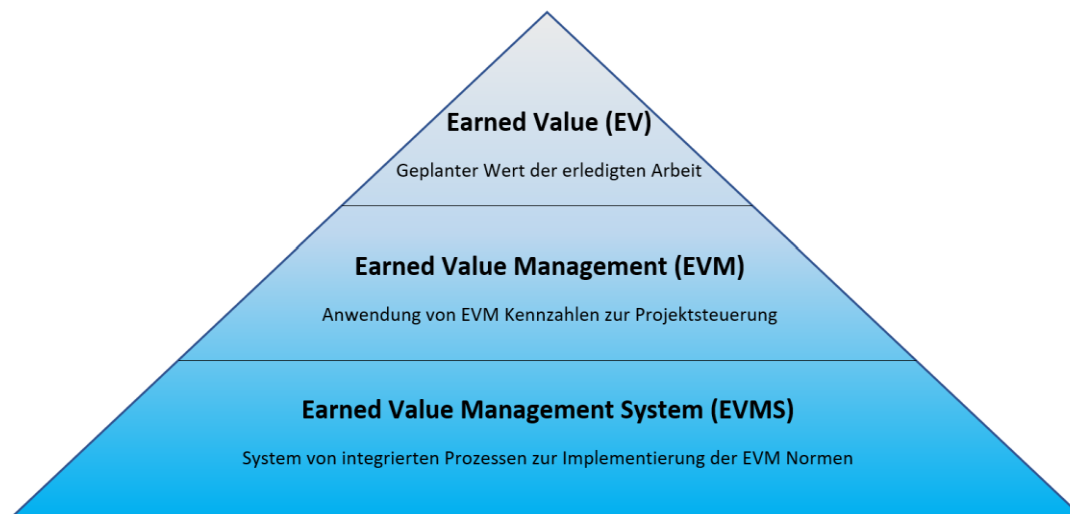


Abbildung 3-8: Hierarchie der EVM Terminologie (Wanner, 2019)

3.3.3 Einsatzgebiete

Der Umfang und Zeitplan sind zwar wesentliche Dimensionen in einem Projekt, um jedoch wichtige Entscheidungen treffen zu können, muss auch die finanzielle Lage analysiert werden. Dies bietet nicht nur zusätzliche Möglichkeiten zur Steuerung, sondern auch eine fundierte Einschätzung der längerfristigen Rentabilität. Teams können beispielsweise den gewünschten Funktionsumfang zum vereinbarten Termin liefern, jedoch zu höheren Kosten als geplant. Somit sollten die geplanten und tatsächlichen Kosten zusammen mit weiteren Kennzahlen dargestellt werden. Die Metriken zur Projektleistung lässt sich über verschiedene Indizes abbilden und in einem Diagramm darstellen. (Dimitrov, 2020)

Popović et al. (2019) ergänzen, dass die Verfolgung und Steuerung von Softwareprojekten mit agilen Vorgehensweisen eine große Herausforderung darstellt. Zum effektiven Management eines Projektes gehört die Überwachung des Status unterschiedlicher Kennzahlen. Abhängig vom jeweiligen Zustand sollen korrigierende Maßnahmen abgeleitet werden können, um das Projekt wieder auf Kurs zu bringen. Dies kann durch den gezielten Einsatz des Steuerungsinstruments EVM erreicht werden, indem auf Sprint- oder Release-Ebene zusätzliche Transparenz zum Fortschritt und zur Performance geschaffen wird.

Das EVM wurde laut Vanhoucke (2009) zur Überwachung und Steuerung des Basisprojektplans unter Berücksichtigung des Umfangs, der Liefertermine und des Budgets eingeführt. Es handelt sich um ein stark verbreitetes und allgemein anerkanntes Managementinstrument, welches die Berechnung von Kosten- und Terminabweichungen ermöglicht. Auf deren Basis können wiederum Leistungsindizes und Forecasts bezüglich der Kosten und Termine im Projekt erstellt werden. Die errechneten Indizes liefern zudem frühzeitig Hinweise auf den Status, um den Bedarf von etwaigen Korrekturmaßnahmen aufzuzeigen.

Rozenes, Vitner und Spraggett (2006) ergänzen, dass das am häufigsten genutzte multidimensionale Kontrollsystem zur Erhebung des Projektstatus bezüglich des Fortschritts und der Kosten die EVA ist. Als einziger Nachteil wird die fehlende Integration mit weiteren Dimensionen wie Technologie, Qualität und Design genannt.

Für Martinelli und Milošević (2016) liegt der Hauptzweck des EVM Ansatzes in der Vorhersage von projektspezifischen Endergebnissen während der Laufzeit. Mit dieser Methode lässt sich bereits ab einem Fertigstellungsgrad von 15 Prozent eine solide und statistisch verlässliche Prognose erstellen, sofern bei prognostizierten Abweichungen erforderliche Korrekturmaßnahmen ergriffen werden. EVM ist darüber hinaus ein Instrument, welches für jedes Projekt, unabhängig von der Branche und der Größe, eingesetzt werden kann. Angesichts der Menge an Ressourcen, die bei großen Vorhaben eingesetzt werden, lässt sich eine umfassende EVA in der Regel auch leicht rechtfertigen.

Laut dem Project Management Institute (2019) hat sich EVM in den letzten Jahrzehnten als bewährte Praxis im Projektcontrolling etabliert. Der Einsatz hilft bei der Bereitstellung von Frühwarnsignalen und ermöglicht durch rechtzeitiges Setzen von Korrekturmaßnahmen die Erzielung früher Erfolge. Darüber hinaus unterstützt es bei der Erreichung von Kosten- und Terminzielen und trägt zur Verbesserung der Kommunikation durch Einbindung von Stakeholdern bei. Die errechneten Zahlen aus der EVM Kalkulation werden üblicherweise in einen finanziellen Wert umgerechnet, während die Darstellung des Fortschritts meist mit Hilfe von Burn-Diagrammen erfolgt.

Lewis (2011) ergänzt, dass zur frühzeitigen Identifikation von Problemen nicht nur die Information notwendig ist, wie viel Aufwand gesamt in ein Projekt geflossen ist. Vielmehr wird dies erst ermöglicht, wenn der entstandene Aufwand dem Fortschritt gegenübergestellt werden kann. EVM wurde von Praktikern des Projektmanagements als Methode zur Messung des Fortschritts sowie zur Erstellung von Forecasts entwickelt. Es gilt als das beste bisher entwickelte System für diese Zwecke.

3.3.4 Kennzahlen

In diesem Abschnitt werden die Kennzahlen, welche im Rahmen des EVM zur Anwendung kommen, aufgelistet und beschrieben.

Laut Zirkler, Nobach, Hofmann und Behrens (2019) basiert das EVM auf drei Grundwerten. Diese Werte müssen bekannt sein und analysiert werden, um in weiterer Folge Abweichungen, Trends und Forecasts errechnen und visualisieren zu können.

Planned Value

Carroll (2017) bezeichnet den Planned Value (PV) als Schlüsselement, welches die Grundlage vom EVM bildet. Er ist zeitbasiert und stellt die kumulativen Plankosten auf Basis des geplanten Fortschritts zum jeweiligen Statusdatum im Projekt dar. Der PV am Ende des Projektes wird auch

als Budget at Completion (BAC) genannt. Das BAC stellt den Wert für den geplanten Umfang dar.

Das Project Management Institute (2017) bezeichnet den PV als Summe der budgetierten Kosten der geplanten Arbeit im betrachteten Berichtszeitraum.

Earned Value

Der Earned Value (EV) ist nach Pries und Quigley (2011) jener Wert, den das Projekt zu einem Zeitpunkt, gemessen am Fortschritt, erwirtschaftet hat. Der EV stellt also den kumulierten Wert der erledigten Arbeit im Verhältnis zum PV dar.

Actual Cost

Die Actual Cost (AC) stellt die tatsächlichen Kosten für die geleistete Arbeit dar. Der Betrag basiert auf Aufzeichnungen von Aufwänden, die innerhalb eines bestimmten Zeitraumes angefallen sind. (Kerzner, 2017b)

Kerzner (2017b) definiert die EVM-spezifischen Varianzen als Abweichungen von geplanten Werten bezüglich Zeitplan, Fortschritt oder Kosten. Sie werden üblicherweise während des Projektverlaufs überwacht und bei Überschreitung vordefinierter Schwellwerte gemeldet. Abweichungen vom Plan soll mit gezielten Korrekturmaßnahmen entgegengewirkt werden.

Schedule Variance

Die Schedule Variance (SV) ist der Betrag, um den das Projekt zu einem bestimmten Zeitpunkt vor oder hinter dem geplanten Liefertermin liegt. Sie stellt die Differenz zwischen dem EV und dem PV dar. (Project Management Institute, 2019)

Cost Variance

Die Cost Variance (CV) stellt die arithmetische Differenz zwischen dem EV und den AC zum betrachteten Zeitpunkt dar. (Lester, 2017)

Variance at Completion

Die Variance at Completion (VAC) stellt die Abweichung zwischen den geplanten und den prognostizierten Gesamtkosten zum Ende des betrachteten Projektes dar. Sie gilt als Prognose des Budgetdefizits oder -überschusses nach dem Projektabschluss. (Wanner, 2019)

Schedule Performance Index

Der Schedule Performance Index (SPI) bezeichnet die zeitliche Effizienz eines Projektes und errechnet sich aus Verhältnis vom EV zum PV. (Project Management Institute, 2019)

Cost Performance Index

Der Cost Performance Index (CPI) ist das Gegenstück zum SPI und beschreibt ein Maß für die Kosteneffizienz. Zur Berechnung wird der EV den AC gegenübergestellt. (Project Management Institute, 2019)

Laut Popović et al. (2019) zeigt der CPI wie effizient das Budget, verglichen mit der geplanten Verwendung, ausgegeben wird.

Anbari (2003) nennt neun verschiedene Szenarien um den Status eines Projektes im Hinblick auf dessen Fortschritt und Budget anhand der zeitbezogenen Kennzahlen SPI und SV sowie der kostenbasierten Kennzahlen CPI und CV zu interpretieren. Diese Szenarien sind in der Tabelle 3-1 abgebildet.

Leistungskennzahl	SV & SPI			
		>0 & >1	=0 & =1	<0 & <1
CV & CPI	>0 & >1	unter Zeitplan und unter Budget	gemäß Zeitplan und unter Budget	über Zeitplan und unter Budget
	=0 & =1	unter Zeitplan und gemäß Budget	gemäß Zeitplan und gemäß Budget	über Zeitplan und gemäß Budget
	<0 & <1	unter Zeitplan und über Budget	gemäß Zeitplan und über Budget	über Zeitplan und über Budget

Tabelle 3-1: Bedeutung der Leistungskennzahlen im EVM (Anbari, 2003)

Estimate to Completion

Die Kennzahl Estimate to Completion (ETC) beschreibt die geschätzten Kosten der Aufwände, die erforderlich sind, um den Rest des Projektes abzuschließen. (Carroll, 2017)

Estimate at Completion

Estimate at Completion (EAC) ist die Kennzahl zur Vorhersage der erwarteten Gesamtkosten für die Fertigstellung aller Arbeiten. Es stellt die Summe der tatsächlichen Kosten bis zum Statuszeitpunkt und der geschätzten Fertigstellungskosten (ETC) dar. (Project Management Institute, 2019)

To Complete Performance Index

Der To Complete Performance Index (TCPI) beschreibt den notwendigen CPI, um die verbleibende Arbeit innerhalb des geplanten Gesamtbudgets abzuschließen. Der TCPI sollte also der angestrebte CPI für die restliche Projektdauer sein. (Wanner, 2019)

Laut dem Project Management Institute (2019) errechnet sich der TCPI aus dem Verhältnis der Kosten für die Fertigstellung der ausstehenden Arbeit (ETC) zum verfügbaren Budget (BAC).

Wanner (2019) unterteilt die einzelnen Kennzahlen in vier Gruppen. Die Kennzahlen PV, EV, AC und BAC bilden die Basisdaten und dienen als Grundlage zur Kalkulation von Abweichungen, Leistungsindizes und Vorhersagen. Die Abbildung 3-11 stellt eine Übersicht der Zusammenhänge der Kennzahlen im EVM und die Formeln zur Kalkulation dar.

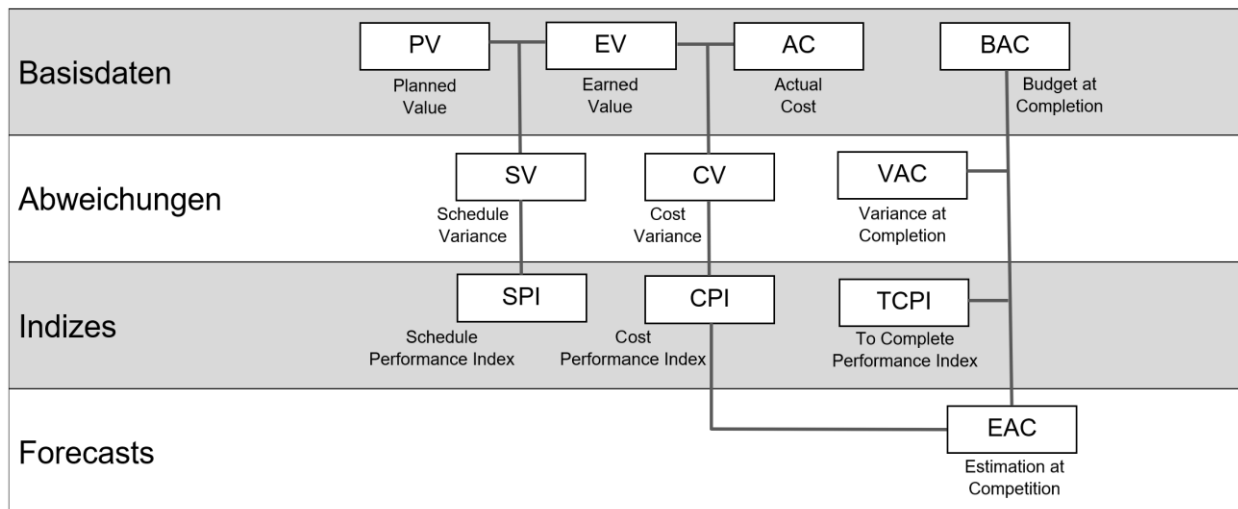


Abbildung 3-9: Zusammenhang der EVM Kennzahlen nach Wanner (2019)

3.3.5 Earned Value Management in agilen Projekten

Nach Lysenko und Fediai (2021) überschreitet die Mehrheit von IT-Projekten die ursprünglich veranschlagten Kosten. Trotz des offensichtlichen Bedarfs zur Kostenkontrolle wird diesem vor allem bei Projekten, welche mit agilen Methoden wie Scrum durchgeführt werden, wenig Aufmerksamkeit geschenkt. Dies beruht häufig auf Zweifel bezüglich der mangelnden Kompatibilität von Instrumenten wie EVM mit agilen Ansätzen. Zudem besteht häufig auch die fälschliche Annahme, dass eine vollständige Definition und Schätzung der Anforderungen sowie eine strikte Planung vorausgesetzt wird. Im Framework Scrum wird jedoch die Notwendigkeit der inkrementellen, mehrstufigen Planung betont. Näher betrachtet kann ein Sprint mit dessen fixen Attributen der Dauer und dem Backlog sowie dem übergeordneten Ziel, ein wertvolles Produkt zu liefern, als Teilprojekt gesehen werden. Lassen sich zudem die Kosten für die geplanten Aufgaben in einem Sprint messen, lässt sich EVM auf Basis dieser Metriken auch in agilen Umfeldern etablieren. Bei der Erstellung von Forecasts können EVM-spezifische Berechnungen unter Zuhilfenahme von Tabellenkalkulationstools durchgeführt werden. EVM kann also auch für Scrum-Teams ein wertvolles Steuerungsinstrument sein.

Nee (2010) zufolge geht man bei traditionellen Ansätzen davon aus, dass der gesamte Umfang eines Projektes feststeht. Darüber hinaus ist der Ablauf meist vorhersehbar und der künftige Fortschritt lässt sich aus dem bisher geleisteten Aufwand prognostizieren. Bei agilen Projekten hingegen sind viele der zugrunde liegenden Annahmen aus dem traditionellen Projektmanagement nicht anwendbar. Der Planungshorizont ist in der Regel wesentlich kürzer und der Umfang kann sich während des Projektverlaufs kurzfristig ändern, wodurch es mit zunehmender Dauer immer schwieriger wird, die Plankosten zu ermitteln. Aus diesem Grund sollten EVM-basierte Prognosen nicht auf die gesamte Dauer, sondern vielmehr auf einzelne Iterationen oder Releases angewandt werden. Solange sich das Team auf die Entwicklung konzentrieren kann, und nicht von Statusberichten ablenken lässt, können traditionelle Steuerungsinstrumente auch in agilen Umfeldern effektiv sein.

Moran (2016) ergänzt, dass agiles EVM auch die Zusammenführung von Berichten aus mehreren Projekten ermöglicht. Die Velocity ist ein teamspezifisches Maß und kann aus vielen verschiedenen Gründen schwanken. Sie kann also nicht einfach übernommen werden, um den teamübergreifenden Fortschritt darzustellen. Um einen entsprechenden Überblick zu erhalten, besteht jedoch die Möglichkeit, den EV je Team zu berechnen und zu aggregieren. EVM kann also auch als effektives Prognoseinstrument auf Release-, Projekt- und Programmebene eingesetzt werden.

Laut Nikravan und Forman (2010) sind Burndown- und Burnup-Diagramme wesentliche Techniken zur Überwachung der Leistung in agilen Projekten und entsprechen den grundlegenden EVM-Messkonzepten. Das Release-Burnup-Diagramm ähnelt dem EVM-Diagramm, da es einerseits den geplanten Fortschritt, analog zum PV, darstellt. Darüber hinaus wird auch der tatsächliche Fortschritt über mehrere Iterationen hinweg aufgezeigt. Die einzige Metrik, welche im Release-Burnup-Diagramm im Vergleich zu EVM-Diagrammen fehlt, sind die tatsächlichen Kosten. Dass die tatsächlichen Kosten in agilen Teams häufig nicht erfasst werden, liegt oft daran, dass diese mit typischen agilen Metriken nur schwer messbar sind. Zur Ermittlung kostenspezifischer Kennzahlen, gibt es trotz deren Wichtigkeit für die Stakeholder, nur selten Empfehlungen und Anleitungen zur Messung.

Für Fairley (2009) ist die Protokollierung von Aufwand und Zeit in der Abwicklung von Projekten von wesentlicher Bedeutung. Die korrekte und verlässliche Zeiterfassung ermöglicht die Erhebung der Kosten und liegt in der Verantwortung jedes Teammitglieds. Leiterinnen und Leiter eines Teams oder Projektes sind für die Vollständigkeit der Daten und deren zeitnahe Erfassung zuständig.

Laut dem Project Management Institute (2019) werden die tatsächlichen Kosten in einem Projekt üblicherweise aus den Arbeitsstunden abgeleitet, welche für den betrachteten Zeitraum im Zeiterfassungssystem der Organisation erfasst wurden.

3.4 Zusammenfassung

In diesem Kapitel erfolgte zu Beginn eine theoretische Auseinandersetzung mit der Steuerung von Projekten. Dabei wurden eingangs Aspekte wie die Planung oder die Schätzung des Projektumfangs behandelt. Darüber hinaus wurden auch wesentliche Aufgaben wie die Budgetierung, das Monitoring und das Erstellen von Vorhersagen im Kontext von Projekten beschrieben.

Laut Trendowicz (2013) ist es im Projektmanagement notwendig auf die effektive und effiziente Nutzung von Projektressourcen zu achten. Dabei müssen stets zahlreiche Bereiche berücksichtigt und beobachtet werden, um den finanziellen Erfolg eines Projektes sicherzustellen.

Im darauffolgenden Abschnitt wurden gängige Berichte, Metriken und Kennzahlen im Zusammenhang mit der Überwachung und Steuerung von agilen Projekten näher beschrieben. Im Zuge dessen erfolgte auch eine kurze Erläuterung von finanziellen Kennzahlen, welche im Projektmanagement häufig zum Einsatz kommen.

Im abschließenden Kapitel wurde das Projektsteuerungsinstrument EVM näher beschrieben. Nach der Begründung für die Auswahl dieses Werkzeuges erfolgte die Definition des Instruments sowie eine Beschreibung der wesentlichen Anwendungsgebiete. Darüber hinaus wurden die Kennzahlen des EVMs beschrieben und der Einsatz in agilen Umfeldern diskutiert.

Nikravan und Forman (2010) beschreiben die Wichtigkeit von gängigen Berichtsmethoden wie Burndown- und Burnup-Diagrammen. Allerdings werden die tatsächlichen Kosten in agilen Projekten häufig nicht verfolgt. Diese fehlende Komponente kann mit dem Einsatz von EVM kompensiert werden, da die Überwachung des Fortschritts mit Burn-Diagrammen den grundlegenden Messkonzepten von EVM entspricht.

4 EMPIRISCHER TEIL

Dieses Kapitel beschäftigt sich mit dem empirischen Teil dieser Arbeit. Zunächst wird der Design Science Ansatz beschrieben, an welchen sich die Umsetzung des empirischen Teils anlehnt.

Im Anschluss daran erfolgt die Beschreibung der Entwicklung des Prototyps. Dabei wurden zunächst die Anforderungen aus der theoretischen Auseinandersetzung mit EVM abgeleitet. Darüber hinaus wurden die Basisdaten beschrieben, welche für den Einsatz des Prototyps erforderlich sind. Zur besseren Veranschaulichung wurden beispielhafte Daten herangezogen. Danach wurden die Kalkulationen zur Ermittlung der EVM Kennzahlen dargestellt, welche zur Erfüllung der zuvor beschriebenen Anforderungen notwendig sind. Im Anschluss daran werden der Prototyp und die damit ermittelten Kennzahlen dargestellt. Zum einfacheren Verständnis wird dies unter Zuhilfenahme von Beispieldaten durchgeführt.

Nach der Umsetzung und Beschreibung des Prototyps werden die Ergebnisse der Forschung validiert und bewertet. Die Evaluierung der Forschungsergebnisse erfolgt im Rahmen von Interviews mit Expertinnen und Experten, welche im Programm- oder Projektmanagement von Softwareunternehmen tätig sind. Mit der Durchführung einer qualitativen Inhaltsanalyse nach Mayring (2015) soll die Einsatzfähigkeit von EVM in unterschiedlichen Softwareunternehmen ermittelt werden.

4.1 Anwendung des Design Science Ansatzes

Hevner, March, Park und Ram (2004) unterscheiden in der Wirtschaftsinformatik zwischen zwei grundlegenden Forschungsparadigmen. In der verhaltenswissenschaftlichen Forschung wird versucht menschliches oder organisatorisches Verhalten zu analysieren. Im Gegensatz dazu werden mit designwissenschaftlichen Ansätzen wie dem Design Science Ansatz neue und innovative Artefakte erstellt oder erweitert.

Design Science Research ist laut vom Brocke, Hevner und Maedche (2020) ein Paradigma zur Problemlösung, mit welchem technologische und wissenschaftliche Wissensbasen durch die Schaffung von Artefakten erweitert werden. Dies erfolgt durch die Lösung von Problemen aus der realen Welt und die Verbesserung der Umgebung, in der Artefakte instanziiert werden.

Wieringa (2014) definiert ein Artefakt als etwas von Menschen Erschaffenes, um einen praktischen Zweck bei der Erstellung, Wartung oder Verwendung eines Informationssystems zu erfüllen. Dabei können Artefakte unterschiedlichste Gestalten annehmen. Als Beispiele können Algorithmen, Methoden, Techniken und Konzepte genannt werden.

In dieser Arbeit entspricht das zu entwickelnde Artefakt einem Prototyp zur kennzahlenbasierten Steuerung von agilen Softwareprojekten mittels EVM. Die Umsetzung dieses Prototyps wird im folgenden Abschnitt 4.2 näher erläutert.

Johannesson und Perjons (2014) ergänzen, dass der Hauptgrund für die Anwendung von Design Science Ansätzen in Alltagsproblemen liegt, welche mit Artefakten gelöst werden sollen. Im Fokus soll dabei die Unterstützung von Menschen in der Praxis stehen, indem Lösungen für bestehende Probleme in der Praxis bereitgestellt werden. Die Praxis stellt dabei den jeweiligen Kontext dar, in welchem das Problem beobachtet wurde. Artefakte existieren also nicht isoliert, sondern sind immer in einem größeren Kontext zu betrachten. Die Beziehungen zwischen Menschen, Problemen, Artefakten und der Praxis sind in Abbildung 4-1 zusammengefasst.

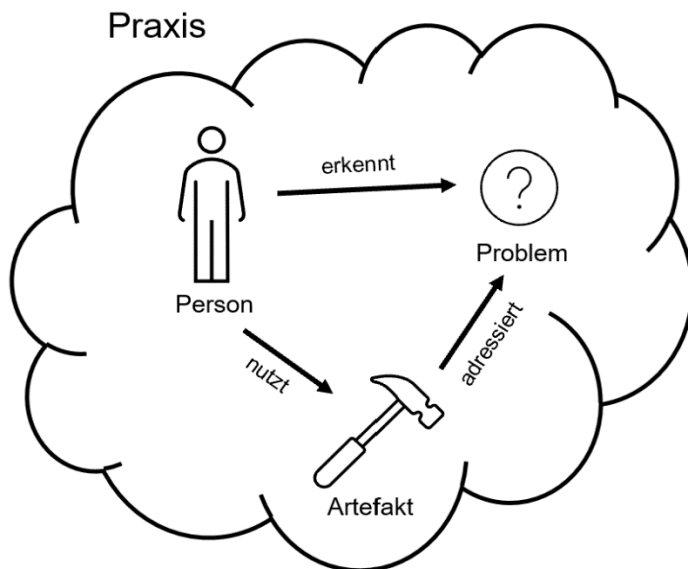


Abbildung 4-1: Zusammenhänge in der Design Science (Johannesson & Perjons, 2014)

Nach Hevner et al. (2004) besteht das grundlegende Prinzip der designwissenschaftlichen Forschung darin, dass Verständnis und Wissen für ein Problem bei der Erstellung und Anwendung eines Artefaktes erworben werden. In Bezug auf die Anwendung des Design Science Ansatzes wurden sieben Leitgedanken abgeleitet. Diese sind in der Tabelle 4-2 abgebildet und soll Forscherinnen und Forschern dabei helfen, die Anforderungen an eine effektive designwissenschaftliche Forschung besser zu verstehen.

Leitlinie	Beschreibung
Artefaktgestaltung	Design Science sieht die Erstellung eines Artefaktes vor, welches einen praktischen Nutzen erzeugt.
Problemrelevanz	Das Ziel von Design Science lautet zur Lösung eines existierenden und aktuellen Problems beizutragen.

Designevaluierung	Der Nutzen eines Artefaktes soll mittels angemessenen Bewertungsmethoden evaluiert werden.
Forschungsbeitrag	Design Science erfordert neben der Lösung des Problems einen allgemeingültigen Beitrag zur Forschung.
Forschungsstrenge	Bei der Erstellung und Evaluierung des Designartefaktes sollen rigorose Methoden angewandt werden.
Optimierung	Die Lösung des Problems soll inkrementell entwickelt und laufend optimiert werden.
Kommunikation	Das Ergebnis soll dem Fachpublikum wirkungsvoll präsentiert werden.

Tabelle 4-1: Sieben Leitlinien des Design Science Ansatzes (Hevner et al., 2004)

Laut Benner-Wickner, Kneuper und Schlömer (2020) hat sich der Design Science Ansatz vor allem in technischen Disziplinen als geeignete Forschungsmethode etabliert. Aufgaben zur Lösung praxisrelevanter Probleme können durch die Erstellung von Modellen, Methoden oder eines Informationssystems durchgeführt werden. Zur Durchführung einer designwissenschaftlichen Forschung können viele verschiedene Variationen angewandt werden. Eine für Abschlussarbeiten besonders empfehlenswerte Variante ist jene von Österle et al. (2010).

Nach Österle et al. (2010) basiert das Vorgehen in der designwissenschaftlichen Forschung im Wesentlichen auf einem vierstufigen Prozess, wobei die einzelnen Phasen jeweils in mehreren Iterationen durchlaufen werden können. Die vier Phasen des Design Science Prozesses nach Österle et al. (2010) lauten:

1. Analyse

In der Analysephase werden zunächst der Kontext und das darin zu lösende Problem erläutert. Darüber hinaus werden die Forschungsziele spezifiziert und der aktuelle Stand der Wirtschaft und Wissenschaft analysiert. Auf dieser Basis kann das Artefakt definiert und die nötigen Anforderungen abgeleitet werden.

5. Design

Artefakte sollen mit allgemein anerkannten Methoden entwickelt werden. Die Erstellung eines Artefaktes soll dabei die Rahmenbedingungen und Anforderungen aus der vorgelagerten Analysephase berücksichtigen.

6. Evaluierung

Die Ergebnisse der Forschung sollen mit wissenschaftlicher Strenge validiert und bewertet werden. Als Basis für die Evaluierung eines Artefaktes dienen die zuvor abgeleiteten Anforderungen und Ziele.

7. Diffusion

Nach der erfolgreichen Erstellung eines Artefaktes sollen die Erkenntnisse und Ergebnisse der Forschung einem breiteren Publikum zugeführt werden. Dies kann durch die Veröffentlichung von wissenschaftlichen Arbeiten oder durch gezielte Anwendung in der Praxis geschehen.

4.2 Umsetzung des Prototyps

Der folgende Abschnitt beschäftigt sich mit der Umsetzung des Prototyps, welcher zur Veranschaulichung des Projektsteuerungsinstruments EVM angewandt wird. Für die Umsetzung des Prototyps wurde das Tabellenkalkulationstool Excel vom Hersteller Microsoft (2022) eingesetzt, mit welchem Berechnungen durchgeführt sowie Grafiken und Diagramme erstellt werden können.

4.2.1 Anforderungen

Die Anforderungen an den Prototyp leiten sich aus den Einsatzgebieten und Beschreibungen der Kennzahlen des EVM ab, welche in den Abschnitten 3.3.3 und 3.3.4 näher beschrieben wurden. Die nachfolgende Tabelle 4-2 fasst die Anforderungen an den umzusetzenden Prototyp zusammen.

#	Beschreibung
A01	Wert für die geplante Arbeit im betrachteten Zeitraum
A02	Wert der erledigten Arbeit im betrachteten Zeitraum
A03	Kosten für die geleistete Arbeit im betrachteten Zeitraum
A04	Abweichung zwischen erledigter und geplanter Arbeit
A05	Abweichung zwischen Kosten und Wert der geplanten Arbeit im betrachteten Zeitraum
A06	Verhältnis zwischen erreichtem und geplantem Wert im betrachteten Zeitraum
A07	Verhältnis zwischen Kosten und geplantem Budget im betrachteten Zeitraum
A08	Erwartete Restkosten zur Fertigstellung des Vorhabens
A09	Erwarteter Wert des Fortschritts zum Ende des betrachteten Vorhabens
A10	Erwartete Gesamtkosten zum Ende des betrachteten Vorhabens

A11	Abweichung zwischen erwartetem und geplantem Wert des Fortschritts
A12	Abweichung zwischen erwarteten Kosten und erwartetem Wert des Fortschritts
A13	Erforderlicher Kostenentwicklungsindex zur planmäßigen Umsetzung der verbleibenden Arbeit

Tabelle 4-2: Anforderungen an den Prototyp

4.2.2 Basisdaten

In diesem Abschnitt werden zunächst die für den Prototyp erforderlichen Basisdaten beschrieben. Zur Veranschaulichung werden für die team- und sprintspezifischen Basisdaten exemplarische Daten definiert, welche in weiterer Folge für die Kalkulation und Darstellung des Prototyps verwendet werden. Dabei ist festzuhalten, dass sämtliche Daten frei erfunden sind und keinerlei Bezug zu einem Team oder Projekt aus der Realität darstellen.

Das Team besteht aus acht Mitarbeiterinnen und Mitarbeitern, welche laut Dienstvertrag jeweils zu 40 Stunden pro Woche angestellt sind. Sämtliche Teammitglieder sollen ausschließlich an dem Beispielprojekt arbeiten, damit die Kapazität, abgesehen von Abwesenheiten und Ineffizienzen, vollständig dem Projekt gewidmet werden kann. Die beispielhaften Teamdaten werden in der Tabelle 4-3 dargestellt.

Mitarbeiter/-in	Vertragsstunden (pro Woche)	Gesamtkosten (pro Jahr)
Mitarbeiter/-in 1	40 h	120.000,00 €
Mitarbeiter/-in 2	40 h	95.000,00 €
Mitarbeiter/-in 3	40 h	110.000,00 €
Mitarbeiter/-in 4	40 h	100.000,00 €
Mitarbeiter/-in 5	40 h	90.000,00 €
Mitarbeiter/-in 6	40 h	115.000,00 €
Mitarbeiter/-in 7	40 h	95.000,00 €
Mitarbeiter/-in 8	40 h	100.000,00 €

Tabelle 4-3: Beispiel-Teamdaten

Aus den Kosten- und Kapazitätsdaten des Beispiel-Teams lassen sich die Summen für die jährlichen Teamkosten sowie die jährliche Kapazität bilden. Unter der Annahme, dass eine Iteration zwei Wochen entspricht, können die durchschnittlichen Kosten sowie die maximale Kapazität für eine Iteration ermittelt werden. Dies ermöglicht auch die Errechnung der durchschnittlichen Kosten für eine Arbeitsstunde. Für die durchschnittliche Velocity des Teams

werden 35 Story Points pro Iteration angenommen. Eine Iteration über zwei Wochen entspricht durchschnittlich zehn Werktagen. Daraus lassen sich die durchschnittlichen Kosten für einen Story Point ermitteln. Die Aufstellung der teamspezifischen Kosten je Stunde und Kosten je Story Point können aus der Tabelle 4-4 entnommen werden.

Teamkosten pro Jahr	825.000,00 €
Teamkosten pro Iteration	31.730,77 €
Teamkapazität pro Iteration	640 h
Kosten pro Stunde	49,58 €
Team Velocity	35
Kosten pro Story Point	906,59 €

Tabelle 4-4: Ermittlung der Kosten

Zur Kalkulation der EVM Kennzahlen müssen laut Zirkler et al. (2019) neben den teamspezifischen Daten auch jene über den geplanten und tatsächlich erreichten Umfang sowie den dafür eingesetzten Kosten bekannt sein.

Die Tabelle 4-5 stellt die Informationen über den geplanten und tatsächlich erreichten Umfang je Sprint dar. Darüber hinaus wird der geleistete Aufwand des Teams für die jeweilige Periode aufgelistet, wodurch die eingesetzten Kosten ermittelt werden können.

Die Werte für den erreichten Umfang sowie den dafür benötigten Aufwand je Sprint basieren wiederum auf einer Annahme und sollen ein realistisches Szenario darstellen. Einerseits wird aufgrund von Abwesenheiten und Ineffizienzen im Team davon ausgegangen, dass nicht die volle Teamkapazität für das Projekt eingesetzt werden kann. Andererseits wird auch angenommen, dass der geplante Umfang aus unterschiedlichen Gründen nicht immer erreicht werden kann.

Sprint	Geplante Story Points	Geplante Story Points (kumuliert)	Erreichte Story Points	Erreichte Story Points (kumuliert)	Aufwand	Aufwand (kumuliert)
1	33	33	20	20	480 h	480,00 h
2	25	58	18	37	360 h	840,00 h
3	26	84	23	61	385 h	1.225,00 h
4	24	108	26	87	380 h	1.605,00 h
5	26	134	29	116	400 h	2.005,00 h
6	34	168	39	155	525 h	2.530,00 h

7	32	200	35	190	565 h	3.095,00 h
8	30	230	35	225	550 h	3.645,00 h

Tabelle 4-5: Beispiel-Sprintdaten

4.2.3 Kalkulation

Auf den zuvor angenommenen Basisdaten können nun die Kalkulationen zur Ermittlung der EVM Kennzahlen durchgeführt werden. Die Tabelle 4-6 stellt die Anforderungen aus Abschnitt 4.2.1 den Kennzahlen und die dafür benötigte mathematische Formel gegenüber.

#	Kennzahl	Formel
A01	Planned Value Budget at Completion	$PV = \text{Kosten pro Story Point} * \text{Geplante Story Points (kumuliert)}$ $BAC = \text{Kosten pro Story Point} * \sum \text{Geplante Story Points}$
A02	Earned Value	$EV = \text{Kosten pro Story Point} * \text{Erreichte Story Points (kumuliert)}$
A03	Actual Cost	$AC = \text{Kosten pro Stunde} * \text{Aufwand (kumuliert)}$
A04	Schedule Variance Schedule Variance in Story Points Schedule Variance in Tagen	$SV = EV - PV$ $SV_{sp} = \frac{SV}{\text{Kosten pro Story Point}}$ $SV_t = \frac{SV_{sp} * \text{Werktage pro Sprint}}{\text{Team Velocity}}$
A05	Cost Variance	$CV = EV - AC$
A06	Schedule Performance Index	$SPI = \frac{EV}{PV}$
A07	Cost Performance Index	$CPI = \frac{EV}{AC}$
A08	Estimate to Completion	$ETC = \frac{BAC}{CPI}$
A09	Schedule Estimate at Completion	$SEAC = BAC * SPI$
A10	Cost Estimate at Completion	$CEAC = \frac{SEAC}{CPI}$
A11	Schedule Variance at Completion Schedule Variance at Completion in Story Points Schedule Variance at Completion in Tagen	$SVAC = SEAC - BAC$ $SVAC_{sp} = \frac{VAC}{\text{Kosten pro Story Point}}$ $SVAC_t = \frac{VAC_{sp} * \text{Werktage pro Sprint}}{\text{Team Velocity}}$

A12	Schedule Variance at Completion	$CVAC = SEAC - CEAC$
A13	To Complete Performance Index	$TCPI = \frac{BAC - EV}{BAC - AC}$

Tabelle 4-6: Kalkulation der EVM Kennzahlen

Unter Zuhilfenahme der Formeln kann nun die Kalkulation der EVM Kennzahlen durchgeführt werden. Die Berechnung erfolgt auf Basis der zuvor unter Abschnitt 4.2.2 genannten Team- und Sprintdaten sowie der Auswahl des betrachteten Zeitraumes über das Eingabefeld „Sprint“. Die Abbildung 4-3 stellt die beispielhafte EVM Kalkulation nach Abschluss des dritten Sprints dar.

Neben den errechneten Kennzahlen werden zudem bedingte Formatierungen für die jeweiligen Werte angewandt. Bei Unterschreitung der geplanten Kosten oder positiver Abweichung des Fortschritts wird ein grünes Symbol angezeigt. Im Gegensatz dazu wird bei Überschreitung der Kosten oder negativer Abweichung zwischen tatsächlichem und geplantem Fortschritt ein rotes Warnsymbol eingeblendet. Im vorliegenden Beispiel wurde eine Toleranz von zehn Prozent definiert. Für negative Abweichungen innerhalb dieser Toleranzgrenze wird ein neutrales (gelbes) Symbol dargestellt.

	Team		
	Velocity		35
	Kosten pro Stunde		49,58 €
	Kosten pro Story Point		906,59 €
	Umfang		
	Sprint		3
	Geplante Story Points		84
	Erreichte Story Points		60
	Protokollierter Aufwand (Stunden)		1.225
	EVM Kalkulation		
Basisdaten	Budget at Completion		208.516,48 €
	Planned Value		76.153,85 €
	Earned Value		55.030,22 €
	Actual Cost		60.734,68 €
Abweichungen	Schedule Variance	▼	-21.123,63 €
	Schedule Variance (Story Points)	▼	-24
	Schedule Variance (Tage)		-6,86
	Cost Variance	▼	-5.704,46 €
Indizes	Schedule Performance Index	▼	0,72
	Cost Performance Index	■	0,91
	To Complete Performance Index	■	1,04
Forecasts	Schedule Estimate at Completion	▼	150.677,98 €
	Schedule Variance at Completion	▼	-57.838,50 €
	Schedule Variance at Completion (Story Points)	▼	-64
	Schedule Variance at Completion (Tage)		-18,29
	Cost Estimate to Completion		105.562,65 €
	Cost Estimate at Completion	▼	166.297,33 €
	Cost Variance at Completion	▼	-15.619,34 €
	Toleranz für neutralen Status		10,00%

Abbildung 4-2: EVM Kalkulation nach Abschluss von Sprint 3

Die Kennzahlen des EVM werden auf Basis ihrer Eigenschaften und Funktionalitäten in vier Kategorien unterteilt. Aus der folgenden Aufstellung und Beschreibung dieser Kategorien wird ersichtlich, welche Anforderungen welchen Kategorien zugewiesen werden.

1. Basisdaten (A01, A02, A03)

Diese Kategorie beinhaltet die grundlegenden EVM Kennzahlen und bietet einen Überblick über den geplanten und erreichten Fortschritt sowie die kumulierten Kosten zum Ende des Sprints im ausgewählten Zeitraum.

2. Abweichungen (A04, A05)

In dieser Kategorie werden die kumulierten Abweichungen des Fortschritts und der bisherigen Kosten zum Ende des ausgewählten Sprints dargestellt. Die

Fortschrittsabweichung wird in Form der Differenz zwischen geplantem und erreichtem Wert sowie in Story Points und Werktagen angegeben.

3. Leistungsindikatoren (A06, A07, A13)

In dieser Kategorie werden die Leistungsindikatoren aus den EVM Kalkulationen dargestellt. Während der SPI und der CPI das Verhältnis zwischen geplantem und tatsächlichem Fortschritt beziehungsweise der Kosten darstellen, stellt der TCPI den notwendigen CPI für die restliche Dauer des Projektes dar, um innerhalb der geplanten Kosten zu bleiben.

4. Forecasts (A08, A09, A10, A11, A12)

Die Kennzahlen dieser Kategorie stellen Prognosen für den weiteren Verlauf des Vorhabens auf Basis der bisherigen Daten dar. Ähnlich wie bei den Abweichungen werden die voraussichtlichen Abweichungen des Fortschritts und der Kosten zum geplanten Ende des Vorhabens prognostiziert. Fortschrittsbezogene Abweichungen werden wiederum zusätzlich in Story Points und Werktagen angegeben.

4.2.4 Darstellung

Die zuvor berechneten Kennzahlen werden zusätzlich zur tabellarischen Aufstellung auch in Form eines dynamischen Liniendiagramms dargestellt. Das Diagramm basiert auf den Basisdaten sowie der Auswahl des Sprints, welcher den bisher absolvierten Zeitraum definiert. Die Abbildung 4-4 zeigt das EVM-Diagramm analog zu den zuvor ermittelten Kennzahlen zum Ende des dritten Sprints. Die Kennzahlen der aus Abschnitt 4.2.3 beschriebenen Kategorien, Basisdaten, Abweichungen und Forecasts wurden zur besseren Veranschaulichung eingezeichnet.

In den Abbildungen 4-5 bis Abbildung 4-12 werden die acht EVM-Diagramme zum Ende jedes Sprints während des Vorhabens dargestellt. Aus dem Verlauf lässt sich die Abhängigkeit der Forecasts vom bisherigen Projektverlauf erkennen.

Earned Value Management

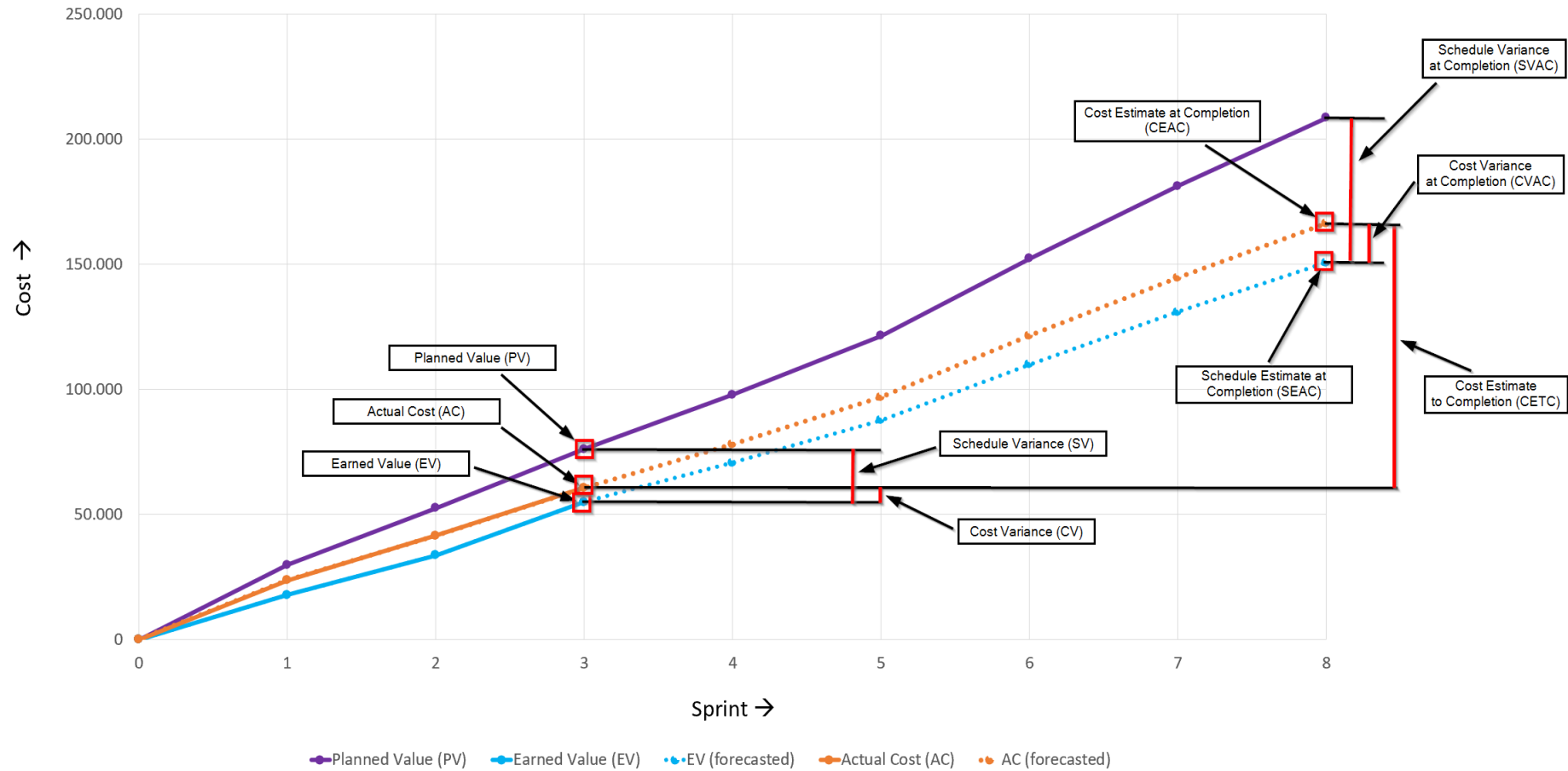


Abbildung 4-3: Kennzahlen im EVM-Diagramm

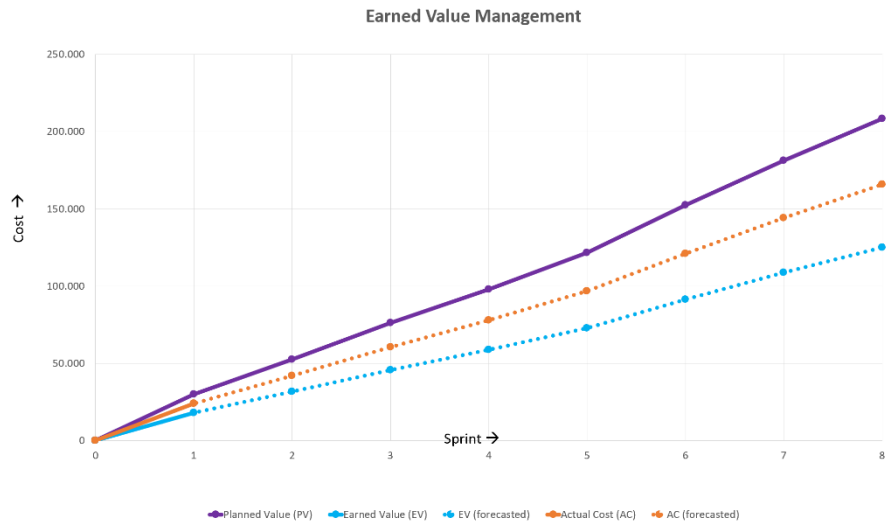


Abbildung 4-4: EVM-Diagramm (Sprint 1)

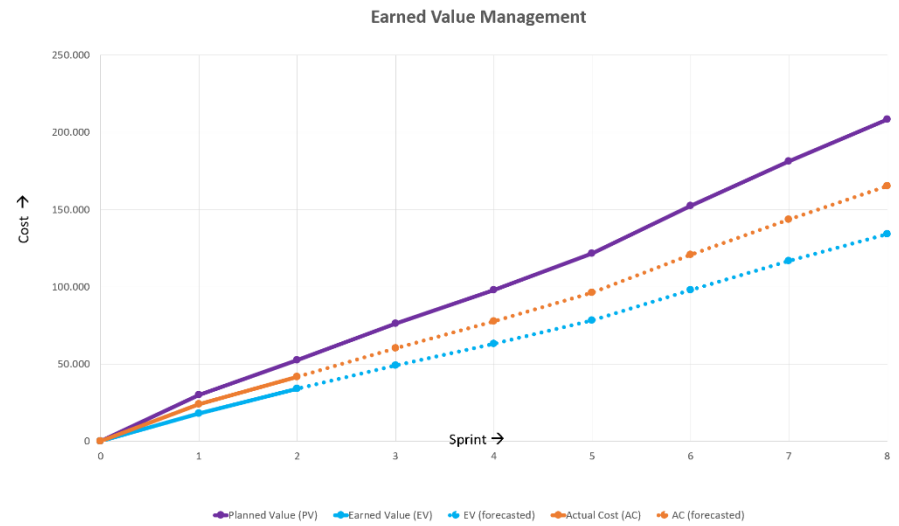


Abbildung 4-5: EVM-Diagramm (Sprint 2)

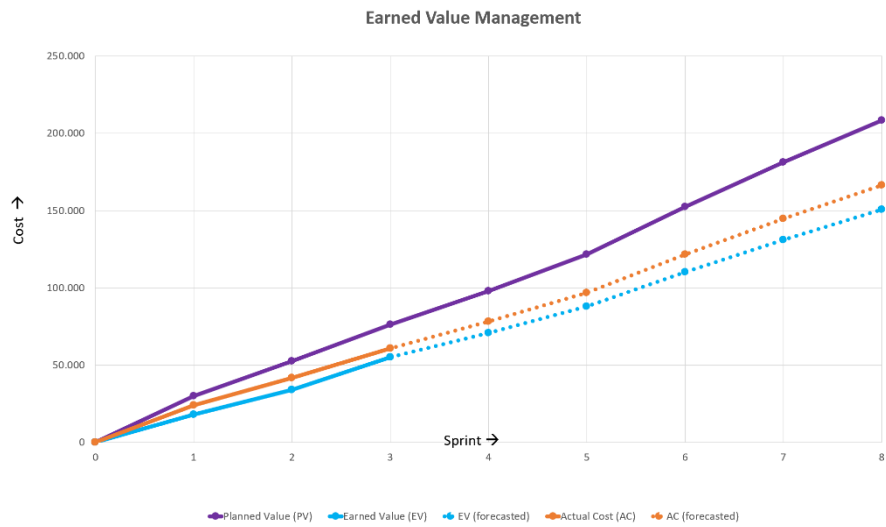


Abbildung 4-6: EVM-Diagramm (Sprint 3)

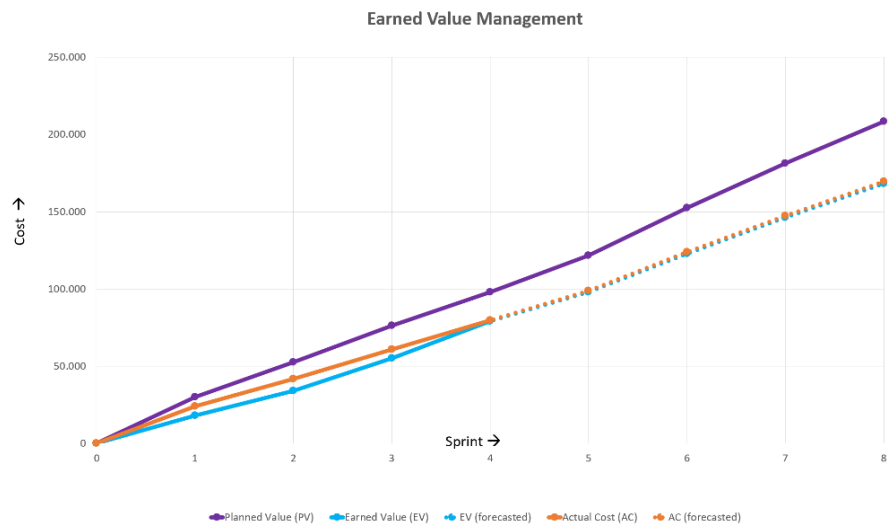


Abbildung 4-7: EVM-Diagramm (Sprint 4)

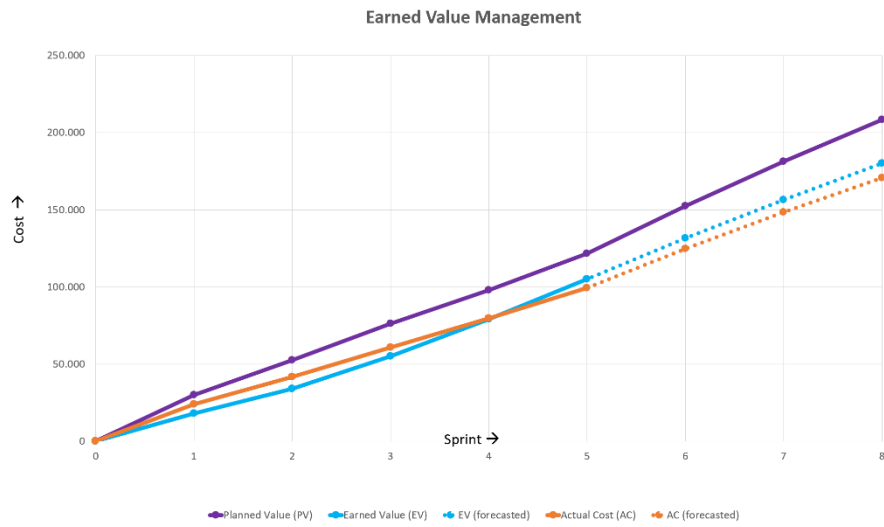


Abbildung 4-8: EVM-Diagramm (Sprint 5)

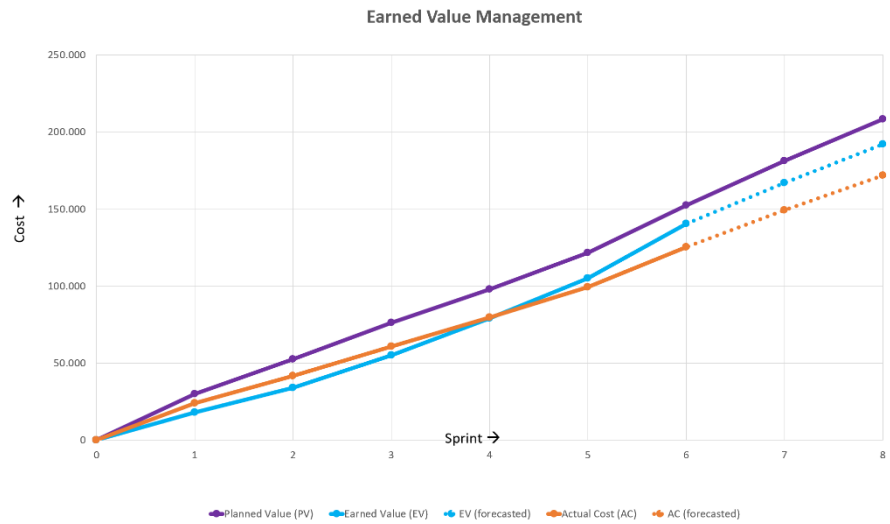


Abbildung 4-9: EVM-Diagramm (Sprint 6)

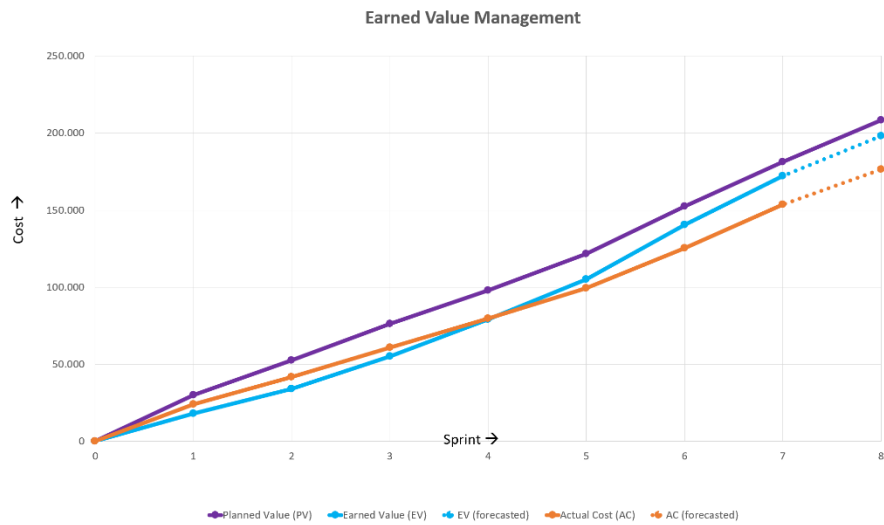


Abbildung 4-10: EVM-Diagramm (Sprint 7)

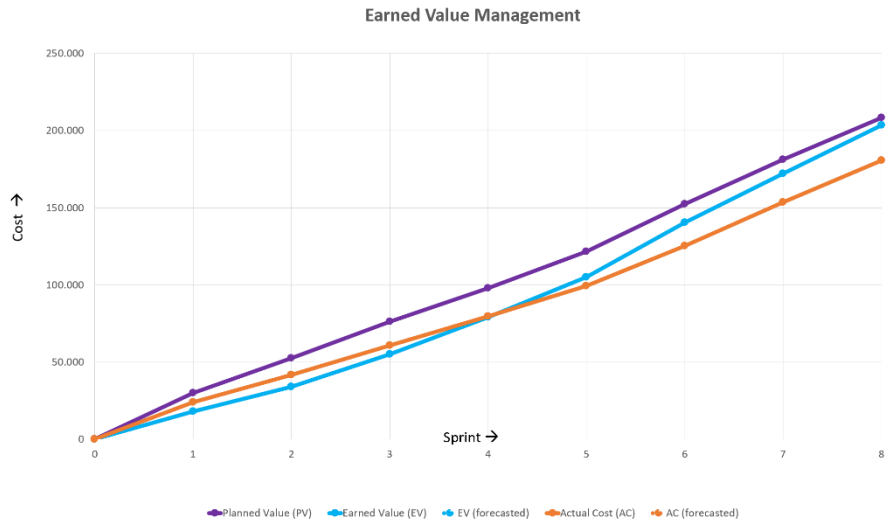


Abbildung 4-11: EVM-Diagramm (Sprint 8)

4.3 Evaluierung des Prototyps

Dieser Abschnitt beschäftigt sich mit der Evaluierung der zuvor beschriebenen Forschungsergebnisse dieser Arbeit. Die Evaluierung erfolgt mittels Interviews mit Expertinnen und Experten und einer qualitativen Inhaltsanalyse des transkribierten Textes. Dies soll Aufschluss darüber geben, inwieweit EVM in agilen Softwareentwicklungsprojekten eingesetzt werden kann.

Nach einer einleitenden theoretischen Beschreibung der Interviewart wird der Aufbau der Befragung erläutert. Im Aufbau wird die Untergliederung der Befragung und der grobe Leitfaden für die Interviews beschrieben. Bei der qualitativen Auswertung wird zunächst die Methode der qualitativen Inhaltsanalyse und die induktive Kategorienbildung nach Mayring (2015) beschrieben, bevor die Ergebnisse der Befragungen aufgezeigt werden.

4.3.1 Interviews mit Expertinnen und Experten

Meuser und Nagel (Meuser & Nagel, 2009) beschreiben Interviews mit Fachleuten als eine seit langem bewährte und beliebte Methode zur Erhebung von Daten. Im Vergleich zur teilnehmenden Beobachtung oder quantitativen Umfragen sind sie kompakter und zeiteffizienter. Expertinnen und Experten können stellvertretend für einen größeren Kreis von Akteuren befragt werden und gelten als Personen, welche aus Sicht der Forscherin oder des Forschers über Wissen verfügen, das im untersuchten Handlungsumfeld nicht jeder oder jedem zugänglich ist. Diese Erhebungsmethode bietet sich vor allem in Situationen an, in welchen der Zugang zu speziellem Wissen schwer oder eingeschränkt möglich ist und der Wissensvorsprung gegenüber Laien genutzt werden soll.

Nach Gläser und Laudel (2010) können Expertinnen und Experten als Beteiligte in sozialen Kontexten gesehen werden, die besonderes Wissen zur Verfügung stellen. Die wesentlichen Begriffe werden dabei wie folgt unterschieden:

„Experte' beschreibt die spezifische Rolle des Interviewpartners als Quelle von Spezialwissen über die zu erforschenden sozialen Sachverhalte. Experteninterviews sind eine Methode, dieses Wissen zu erschließen.“ (Gläser & Laudel, 2010, S. 12)

Interviews können der Literatur zufolge auf viele verschiedene Arten durchgeführt werden. Eine der am weitesten verbreiteten Unterteilungen von Interviews ist jene von Marquardt (2007) anhand der Strukturierung. Marquardt unterscheidet dabei zwischen den folgenden drei Arten:

1. Das wenig strukturierte Interview

Das wenig strukturierte Interview bietet den größten Gestaltungsspielraum, da für das Interview kein vorbereiteter Leitfaden oder Fragenkatalog vorgesehen ist. Es ermöglicht der

Interviewerin oder dem Interviewer das Gespräch je nach Bedarf zu lenken und die Formulierung der Fragen entsprechend dem Verlauf individuell anzupassen.

2. Das stark strukturierte Interview

Zur Durchführung eines stark strukturierten Interviews ist die Erstellung eines Fragebogens im Vorfeld erforderlich. Im Gegensatz zum wenig strukturierten Interview wird hinsichtlich der Auswahl und Formulierung der Fragen wesentlich weniger Flexibilität geboten. Der vorbereitete Fragebogen gibt sowohl die Anzahl der zu stellenden Fragen als auch die Formulierung und Reihenfolge vor.

3. Das teilstrukturierte Interview

Das teilstrukturierte Interview kann als Kombination der wenig strukturierten und der stark strukturierten Interviewart gesehen werden. Es ist zwar ein Leitfaden mit vorbereiteten Fragen vorgesehen, jedoch kann die Auswahl der Fragen und deren Reihenfolge je nach Verlauf des Interviews angepasst werden. Umgekehrt können in bestimmten Situationen auch vertiefende Fragen gestellt werden, welche nicht Teil des vorbereiteten Fragebogens sind.

4.3.2 Aufbau der Interviews

Zur Durchführung der Interviews werden fünf Expertinnen und Experten aus unterschiedlichen Softwareentwicklungsunternehmen konsultiert. Es wird Wert darauf gelegt, dass die interviewten Personen unterschiedliche Rollen in ihren Unternehmen ausüben. Damit soll sichergestellt werden, dass das erhobene Feedback aus den Interviews nicht auf eine spezifische Rolle zurückzuführen ist.

Die Interviews werden an unterschiedlichen Orten durchgeführt. Drei Interviews finden vor Ort jeweils in Räumlichkeiten der Unternehmen statt, in welchen die Expertinnen und Experten tätig sind. Die beiden anderen Interviews werden auf Wunsch der Expertinnen und Experten remote über eine Videokonferenz durchgeführt. Unabhängig vom Ort der Durchführung wurde für alle Interviews ein zeitlicher Rahmen von 20 bis 30 Minuten veranschlagt. Mit den Expertinnen und Experten wurde vereinbart, dass die Befragung anonym stattfindet.

Zu Beginn der Interviews werden die Interviewpartnerinnen und Interviewpartner jeweils um eine mündliche Bestätigung gebeten, dass das Interview vollständig aufgezeichnet werden darf. Danach wird nochmals auf die voraussichtliche Dauer und den geplanten Ablauf des Termins hingewiesen. Anschließend wird das Ziel der vorliegenden Arbeit erläutert und mit der Befragung begonnen. Für die Art des Interviews wurde das teilstrukturierte Interview gewählt. Für die Durchführung wird also ein Leitfaden vorbereitet, welcher den groben Ablauf und die geplanten Fragestellungen beinhaltet. Aufgrund der Natur dieser Interviewart können Fragen flexibel vertieft sowie bei Bedarf vorgezogen oder gänzlich übersprungen werden. Das Interview gliedert sich grundsätzlich in drei Teile.

Der erste Teil des Interviews beinhaltet die Einleitung und Fragestellungen in Bezug auf die Rolle und Erfahrungen der Expertinnen und Experten. Darüber hinaus werden Fragen zur Ausrichtung des Unternehmens bezüglich der Agilität sowie zur unternehmensinternen Projektsteuerung und die dabei eingesetzten Methoden und Kennzahlen gestellt. Die Fragestellungen zur Agilität und Projektsteuerung werden aus der theoretischen Auseinandersetzung mit agilen Vorgehensweisen und Methoden im Projektmanagement abgeleitet, welche in den Kapiteln 2 und 3 behandelt werden. Zusammenfassend werden im ersten Teil der Befragung folgende Fragen gestellt:

- Welche Rolle üben Sie im Rahmen Ihrer derzeitigen Beschäftigung aus?
- Welches Vorgehensmodell und welche Frameworks werden in Ihrem Unternehmen zur Umsetzung von Projekten verwendet?
- Wie viel Erfahrung in Jahren konnten Sie in der Abwicklung von Projekten mit agilen Methoden wie Scrum bisher sammeln?
- Wie würden Sie den Grad der Agilität des Unternehmens, in dem Sie tätig sind, beurteilen?

Vordefinierte Antwortmöglichkeiten:

- Sehr agil
 - Eher agil
 - Eher traditionell
 - Sehr traditionell
- Werden in Ihrem Unternehmen Methoden, Werkzeuge oder Kennzahlen zur Überwachung und Steuerung von Projekten eingesetzt?
 - Können aus Ihrer Sicht traditionelle Ansätze zur Überwachung und Steuerung von Projekten in agilen Frameworks etabliert werden?

Im zweiten Teil der Befragung werden Fragen zur Bekanntheit von EVM und zur Erfahrung der Expertin oder des Experten gestellt. Anschließend wird das im Zuge dieser Arbeit entwickelte Artefakt demonstriert. Nach einer einführenden Definition und Erläuterung des Zwecks von EVM wird auf die Anforderungen des Prototyps eingegangen. Im Zuge der Demonstration werden zunächst die erforderlichen Basisdaten erläutert. Anschließend wird die Kalkulation der Kennzahlen und die Darstellung des EVM-Diagrammes veranschaulicht. Die Demonstration soll einerseits zur Erklärung des Instruments dienen, und andererseits die Erfüllung und Darstellung der zuvor erläuterten Anforderungen bezwecken. Die Fragen zur Kenntnis und Erfahrung der Expertin oder des Experten lauten:

- Sind Ihnen das Projektmanagement-Instrument EVM oder darin vorkommende Kennzahlen bekannt?

- Konnten Sie bereits Erfahrung mit EVM sammeln oder es zur Abwicklung von Projekten einsetzen?

Im dritten Teil des Interviews werden spezifische Fragen zum praktischen Einsatz des Prototyps gestellt. Durch die gezielten Fragen wird einerseits versucht mögliche Einsatzgebiete im Unternehmen der Expertinnen und der Experten zu identifizieren. Ein möglicher Einsatz des Prototyps im Hinblick auf die vier wesentlichen Anwendungskategorien gibt Aufschluss darüber, ob die praktische Anwendung von EVM einen positiven Effekt auf die Steuerung von agilen Projekten hat. Darüber hinaus werden die befragten Personen darum gebeten, etwaige Limitationen des Prototyps in Bezug auf den praktischen Einsatz im eigenen Unternehmen und Möglichkeiten zur Erweiterung zu nennen. Die möglichen Erweiterungen des Prototyps können dabei sowohl funktionaler als auch nicht-funktionaler Natur sein. Die Fragen des dritten Befragungsteils lauten wie folgt:

Basisdaten

- Wie wichtig ist Ihnen der Status des geplanten Fortschritts zum Ende eines Sprints während der Projektlaufzeit?
- Wie wichtig ist Ihnen der Status des tatsächlichen Fortschritts zum Ende eines Sprints während der Projektlaufzeit?
- Wie wichtig ist Ihnen der Status zu den aktuellen Kosten zum Ende eines Sprints während der Projektlaufzeit?
- Setzen Sie zur Umsetzung bereits Werkzeuge ein, um diese Anforderungen zu erfüllen?
- Könnten Sie sich vorstellen den Prototyp für diesen Zweck einzusetzen?

Abweichungen

- Wie wichtig ist Ihnen die Abweichung zwischen dem tatsächlichen und dem geplanten Fortschritt zum Ende eines Sprints während der Projektlaufzeit?
- Wie wichtig ist Ihnen die Abweichung zwischen den tatsächlichen Kosten und dem Wert des aktuellen Fortschritts zum Ende eines Sprints während der Projektlaufzeit?
- Setzen Sie zur Umsetzung bereits Werkzeuge ein, um diese Anforderungen zu erfüllen?
- Könnten Sie sich vorstellen den Prototyp für diesen Zweck einzusetzen?

Leistungsindikatoren

- Wie wichtig ist Ihnen die Ermittlung eines fortschrittsbezogenen Index zur Darstellung des Verhältnisses zwischen geplantem und tatsächlichem Fortschritt?

- Wie wichtig ist Ihnen die Ermittlung eines kostenbezogenen Index zur Darstellung des Verhältnisses zwischen dem Wert des aktuellen Fortschritts und den tatsächlichen Kosten?
- Wie wichtig ist Ihnen die Ermittlung eines kostenbezogenen Index zur Darstellung des erforderlichen Verhältnisses zwischen dem Wert des aktuellen Fortschritts und den tatsächlichen Kosten?
- Setzen Sie zur Umsetzung bereits Werkzeuge ein, um diese Anforderungen zu erfüllen?
- Könnten Sie sich vorstellen den Prototyp für diesen Zweck einzusetzen?

Forecasts

- Wie wichtig sind Ihnen Forecasts zur erwarteten Abweichung der Umsetzung des geplanten Aufwandes?
- Wie wichtig sind Ihnen Forecasts zum wirtschaftlichen Erfolg des Projektes?
- Setzen Sie zur Umsetzung bereits Werkzeuge ein, um diese Anforderungen zu erfüllen?
- Könnten Sie sich vorstellen den Prototyp für diesen Zweck einzusetzen?

Darstellung, Limitationen und Erweiterung

- Wie wichtig ist Ihnen die visuelle Darstellung der Basisdaten, Abweichungen und Forecasts des EVM?
- Welche Limitationen sehen Sie für den vorliegenden Prototyp bei der Steuerung von agilen Projekten?
- Welche Erweiterungsmöglichkeiten wären aus Ihrer Sicht für den vorliegenden Prototyp zur Steuerung von agilen Projekten nützlich?

Das Feedback der Expertinnen und Experten zur Wichtigkeit und Einsetzbarkeit der EVM Funktionalitäten soll in weiterer Folge zur Bestätigung oder Verwerfung der Hypothese aus Abschnitt 1.5 herangezogen werden. Unter Berücksichtigung der genannten Limitationen und Erweiterungsmöglichkeiten soll auch die Beantwortung der Forschungsfrage dieser Arbeit erfolgen.

4.3.3 Qualitative Auswertung

Zur Auswertung der durchgeführten Interviews mit den Expertinnen und Experten wurde eine qualitative Inhaltsanalyse nach Mayring (2015) durchgeführt. Zu diesem Zweck wurden die Befragungen mit den Expertinnen und Experten auf Basis der jeweiligen Aufzeichnung

transkribiert. Die Tonaufnahmen der einzelnen Interviews wurden mithilfe einer Transkriptionssoftware von Amberscript Global B.V. (2022) in Text konvertiert. Im Anschluss wurden Fehler der Transkription korrigiert sowie Satzpausen, Wortwiederholungen und Füllwörter entfernt. Darüber hinaus wurden Passagen mit Dialekt korrigiert oder ins Schriftdeutsche übersetzt, ohne die Bedeutung des Inhalts zu verändern. Englische Fachbegriffe, welche auch im deutschsprachigen Raum gebräuchlich sind und dieselbe Bedeutung haben, wurden nicht übersetzt oder geändert.

Nach Mayring (2015) soll bei der qualitativen Inhaltsanalyse die Richtung der Analyse bestimmt werden. Die Richtung spiegelt dabei das Ziel der Auswertung wider. Im vorliegenden Fall ist das Ziel der Analyse festzustellen, ob und inwieweit das Projektsteuerungsinstrument EVM in agilen Umfeldern eingesetzt werden kann. Nach der Definition des Ziels ist die Entscheidung hinsichtlich der Form der Inhaltsanalyse zu treffen. Mayring (2015) beschreibt drei unterschiedliche Formen der Inhaltsanalyse. Während die Explikation grundsätzlich dazu verwendet wird, um unklare Passagen aus dem vorliegenden Material mit zusätzlichen Informationen anzureichern, wird mit der Methode der Strukturierung der Inhalt des Materials in vordefinierte Kategorien unterteilt. Die dritte Form der Inhaltsanalyse, die Zusammenfassung, wird verwendet, um den Inhalt des zu analysierenden Materials zunächst auf das Wesentliche zu reduzieren und die Erkenntnisse anschließend in Kategorien einzuteilen. Im Gegensatz zur Strukturierung werden die Kategorien bei der Zusammenfassung nicht vorab definiert, sondern erst im Zuge der Analyse erstellt. Entscheidend bei der zusammenfassenden Inhaltsanalyse ist daher die Kategoriendefinition und die Abstraktionsebene, auf welche das Interviewmaterial zusammengefasst werden soll.

In der vorliegenden Arbeit wird die induktive Kategorienbildung nach Mayring (2015) angewandt. Die Kategorien werden auf Basis des vorliegenden Materials gebildet und leiten sich aus den Meinungen der Expertinnen und Experten ab. Das Abstraktionsniveau wird so festgelegt, dass konkrete Anwendungsgebiete und Limitierungen abgeleitet werden können. Neben der Kategoriendefinition und des Abstraktionsniveaus sind bei der qualitativen Inhaltsanalyse nach Mayring (2015) auch die Auswertungs-, Kontext- und Kodiereinheit zu bestimmen. Diese Analyseeinheiten werden für die vorliegende Arbeit wie folgt bestimmt:

- Die Auswertungseinheit legt fest, welches Material zur Verfügung steht und analysiert werden soll. Sie entspricht jeweils den transkribierten Interviews mit den Expertinnen und Experten in Textform.
- Die Kontexteinheit gibt an, welche Informationen für die Kodierung verwendet werden sollen. In dieser Arbeit entspricht eine Kontexteinheit jeweils einem Abschnitt in einem Interview wie unter 4.3.2 beschrieben.
- Die Kodiereinheit bestimmt den geringstmöglichen Bestandteil einer Kontexteinheit, welche ausgewertet und einer Kategorie zugeordnet werden kann. Im vorliegenden Fall entspricht dies ganzen Sätzen aus den Antworten der Expertinnen und Experten.

Basierend auf den zuvor bestimmten Analyseeinheiten sowie der Kategoriendefinition und des Abstraktionsniveaus konnte die induktive Kategorienbildung im Rahmen der qualitativen Inhaltsanalyse nach Mayring (2015) durchgeführt werden. Wie zuvor beschrieben lautete das Ziel der Analyse festzustellen, ob und inwiefern EVM als Projektsteuerungsinstrument in agilen Umfeldern eingesetzt werden kann. Dies beinhaltete im Wesentlichen sowohl die Fragestellung nach möglichen Anwendungsgebieten im Projektmanagement als auch die damit einhergehenden Restriktionen und mögliche Erweiterungsmöglichkeiten des Instruments. Aus diesem Grund wurden die aus der qualitativen Inhaltsanalyse abgeleiteten Kategorien in drei Rubriken unterteilt. Voraussetzung zur Auflistung der Kategorien ist die Nennung der Kategorie von mindestens drei der fünf befragten Expertinnen und Experten. Die Tabelle 4-7 zeigt die abgeleiteten Anwendungsgebiete des Projektmanagement-Werkzeuges EVM in agilen Umfeldern.

Aus den Interviews ging hervor, dass der Einsatz des EVM Werkzeuges vor allem in Projekten mit Fixpreis-Charakter hilfreich sein kann. Dies lässt sich einerseits dadurch begründen, dass das zur Verfügung stehende Ressourcenkontingent bereits im Vorfeld feststeht. Andererseits wird die Planung des Projektumfangs von Fixpreis-Projekten in der Regel vorab durchgeführt, wodurch sich Planwerte über längere zeitliche Horizonte ermitteln lassen. Darüber hinaus können sich die Expertinnen und Experten vorstellen, EVM zum Monitoring des Projektfortschritts einzusetzen. Das frühzeitige Erkennen von Abweichungen kann dabei unterstützen rechtzeitige Maßnahmen zu treffen, um das Projekt wieder auf den geplanten Kurs zu bringen. Ebenfalls wurde die Möglichkeit zum Erstellen von Vorhersagen bezüglich der erwarteten Projektkosten und Liefertermine als wertvolle Eigenschaft identifiziert. Zu guter Letzt sind effizienzbezogene Kennzahlen wie der CPI oder SPI nützlich, um ein einfaches Verhältnis der Plan- und Istwerte darzustellen und zu kommunizieren.

#	Kategorie	Beschreibung
K01	Fixpreis-Projekte	Das Werkzeug ist besonders einsatzfähig bei Projekten mit Fixpreis-Charakter.
K02	Monitoring	Das Werkzeug ermöglicht ein Monitoring des Fortschritts und der Kosten eines Projektes.
K03	Abweichungen	Mit dem Werkzeug können Abweichungen von Soll- und Ist-Werten im Projekt dargestellt werden, um frühzeitige Maßnahmen einzuleiten.
K04	Vorhersagen	Mit dem Werkzeug können Vorhersagen im Hinblick auf Termine und Kosten zum Projektende gemacht werden.
K05	Effizienz	Mit dem Werkzeug können Kennzahlen zur Bewertung der Team-Effizienz ermittelt werden.

Tabelle 4-7: Anwendungsgebiete von EVM in agilen Umfeldern

Neben den möglichen Anwendungsgebieten wurden die Expertinnen und Experten ebenfalls zu den Limitationen von EVM befragt. In der Tabelle 4-8 werden jene Restriktionen aufgelistet, welche bei den Befragungen von mindestens drei Personen genannt wurden.

Die wohl bedeutsamste Restriktion, welche von allen fünf Expertinnen und Experten genannt wurde, ist jene, dass EVM in produktentwickelnden Softwareunternehmen nur bedingt eingesetzt werden kann. Der Grund dafür ist der Umstand, dass sich der Fokus bei Unternehmen, welche ein neues Produkt oder eine neue Version eines Produkts auf den Markt bringen möchten, nicht darauf richtet, einen strikten Zieltermin oder Kostenrahmen einzuhalten. Sowohl der Fortschritt als auch die Kosten spielen bei der geplanten Veröffentlichung eines Produkts eine untergeordnete Rolle. Vielmehr liegt das Ziel der Produktentwicklung darin, einen möglichst hohen Wert für die Kundinnen und Kunden bei maximaler Qualität zu erzeugen. Darüber hinaus wurde der mangelnde Planungshorizont in agilen Umfeldern als Restriktion von EVM genannt. Zwar können Schätzungen und Planwerte in EVM während eines Projektes angepasst werden, jedoch sind Forecasts zum Projekterfolg bei Änderungen des Umfangs und Aufwandes während der Projektlaufzeit nicht nützlich. Als dritte Restriktion wurde die mangelnde Tauglichkeit von EVM in größeren und komplexeren Umfeldern genannt. Während EVM in kleinen Projekten von Nutzen sein kann, fällt ein Einsatz mit zunehmender Anzahl an involvierten Entwicklungsteams aufgrund der steigenden Komplexität schwerer.

#	Kategorie	Beschreibung
K06	Produktentwicklung	Bei der Entwicklung von Softwareprodukten hat die Lieferung hoher Qualität eine höhere Priorität als die Einhaltung der Plankosten und Zieltermine.
K07	Planungshorizont	Die typischerweise kurzfristige operative Planung in agilen Projekten verringert die Vorhersagekraft der erwarteten Kosten und Termine zum Projektabschluss.
K08	Komplexität	Mit zunehmender Komplexität eines Projektes oder Programms wird es schwieriger und Umständlicher das Werkzeug zu benutzen.

Tabelle 4-8: Restriktionen von EVM in agilen Umfeldern

Neben den Anwendungsgebieten und Restriktionen konnten zudem auch Erweiterungsmöglichkeiten identifiziert werden. Die Erweiterungsmöglichkeiten beziehen sich auf den Prototyp, welcher zur Veranschaulichung des Instruments genutzt wurde. Die Erweiterungsmöglichkeiten zur Erhöhung der Tauglichkeit des EVM Instruments werden in der Tabelle 4-9 aufgelistet.

Eine nützliche Erweiterung würden die Expertinnen und Experten in der Darstellung der Business Value sehen. Der PV und der EV stellen jeweils den Wert des geplanten beziehungsweise gelieferten Fortschritts auf Basis der Selbstkosten eines Story Points dar. Interessant wäre jedoch zu sehen, wie hoch der tatsächliche Geschäftswert für die Kundinnen und Kunden wäre. Durch

dessen Ermittlung und Gegenüberstellung zu den tatsächlichen Kosten, könnte man in weiterer Folge den ROI eines Produkts, einer Version oder eines Inkrements berechnen. Als weitere gewünschte Funktionalität wurde die Möglichkeit zur Schätzung des Aufwandes in Stundeneinheiten genannt. Dies würde die Nutzung von EVM oder des Prototyps auch für Teams möglich machen, welche ihre Schätzungen nicht mit Story Points erstellen. Die Velocity eines Teams, würde sich damit nicht mehr aus den durchschnittlich erledigten Story Points je Iteration errechnen, sondern aus dem Durchschnitt der geschätzten Stunden jener Aufgaben, die in einer Iteration erledigt werden konnten. Auch die Skalierbarkeit auf mehrere Teams von EVM wurde als Erweiterungsmöglichkeit genannt. Obwohl die Komplexität von EVM mit zunehmender Projektgröße und Anzahl involvierter Teams steigt, ist der Einsatz in kleinen Projekten mit mehreren Teams denkbar und nützlich. Abschließend wurde die Integration von bestehenden Softwarelösungen als Erweiterungsmöglichkeit genannt. Damit könnten die dem EVM zugrunde liegenden Daten basierend auf Zeitprotokollen aus Produktivsystemen entnommen werden. Dies erspart manuelle Arbeit zur Datenerfassung und ermöglicht die Kalkulation von Projektkennzahlen in Echtzeit.

#	Kategorie	Beschreibung
K09	Business Value	Durch die Ermittlung und Gegenüberstellung des Business Value mit den tatsächlichen Kosten könnte man den ROI berechnen.
K10	Basis für Schätzung	Die Möglichkeit zur Schätzung der Aufwände in Stunden ermöglicht Teams, welche nicht mit Story Points arbeiten, das Werkzeug zu nutzen.
K11	Teams	Die Kalkulation und Darstellung der EVM Kennzahlen soll für mehrere Teams, welche an einem Projekt arbeiten, möglich sein.
K12	Integration	Die Kalkulation und Darstellung der EVM Kennzahlen soll auf Daten aus der produktiven Projektmanagement-Software basieren.

Tabelle 4-9: Erweiterungsmöglichkeiten von EVM in agilen Umfeldern

5 ABSCHLUSS DER ARBEIT

Dieses Kapitel stellt den Abschluss dieser Arbeit dar. Es dient dazu, einen Überblick über die gewonnenen Erkenntnisse zu geben, wodurch einerseits die Beantwortung der Forschungsfrage und andererseits die Beurteilung der ebenfalls im Abschnitt 1.5 beschriebenen Hypothese ermöglicht werden soll. Darüber hinaus werden im Zuge dieses Kapitels die Ergebnisse dieser Arbeit diskutiert und ein Ausblick für mögliche weitere Forschungen beschrieben.

5.1 Zusammenfassung

Im Rahmen dieser Arbeit wurde der Einsatz des Projektsteuerungsinstruments EVM in Softwareentwicklungsprojekten mit agilen Vorgehensweisen untersucht. Dafür wurde eine umfangreiche Literaturrecherche in den Bereichen der agilen Softwareentwicklung, der Steuerung von Softwareprojekten und zum Thema EVM durchgeführt.

Zu Beginn dieser Arbeit wurden die Grundlagen der agilen Softwareentwicklung sowie deren Werte und Prinzipien erörtert. Die theoretische Auseinandersetzung mit diesem Thema ergab, dass agile Ansätze keine strikten Regeln zur Durchführung von Softwareprojekten vorgeben, sondern vielmehr eine Empfehlung zur Priorisierung von projektspezifischen Grundwerten. Da die Rahmenbedingungen eines Projektes jedoch stark von dessen Komplexität abhängen, kann die Priorität der Basiswerte von Projekt zu Projekt variieren. Im Anschluss dazu wurden unterschiedliche Vorgehensmodelle im Projektmanagement beleuchtet. Im Zuge dessen wurden einerseits traditionelle, wasserfallbasierte Modelle beschrieben und andererseits agile Vorgehensmodelle und Frameworks wie Scrum erläutert. Darüber hinaus wurde auch auf hybride Formen, mit Ansätzen aus beiden Welten, eingegangen. Hybride Modelle kommen in der Praxis sehr häufig vor und finden vor allem dann Anwendung, wenn die Vorteile agiler Methoden genutzt werden sollen, während ein gewisser Grad an Disziplin in der Abwicklung von Projekten oder ein regelmäßiges Monitoring des Umsetzungsstatus notwendig ist. Als Vertreter der agilen Vorgehensmodelle wurde danach das Framework Scrum näher beschrieben. Dabei wurden neben einer grundsätzlichen Einführung auch die wesentlichen Rollen, Events und Artefakte erklärt.

Im nächsten Kapitel dieser Arbeit stand die Steuerung von Projekten im Fokus. Dabei wurden wesentliche Komponenten wie die Planung und Schätzung von Aufgaben im Projekt beschrieben. Zur Schätzung einzelner Aktivitäten oder ganzer Projektzeitpläne kommen in agilen Projekten neben stundenbasierten Schätzungen häufig auch Schätzungen basierend auf Story Points zum Einsatz. Wenn die Entwicklungskosten bekannt sind, kann auf Basis der Aufwandsschätzung eines Projektes das geplante Projektbudget berechnet werden. Oftmals werden im Zuge von Projekten auch Forecasts auf Basis des bisherigen Projektstatus erstellt, damit, abhängig vom Verlauf, entsprechende Maßnahmen getroffen werden können. Im Anschluss daran wird ein Überblick über gängige Berichtsmethoden, Metriken und Kennzahlen gegeben, welche in der agilen Softwareentwicklung zum Einsatz kommen. Im letzten Abschnitt dieses Kapitels erfolgte

die Beschreibung von EVM, einem Instrument zur Überwachung des Projektstatus sowie zur Erstellung von fortschritts- und kostenspezifischen Forecasts. Neben den generellen Anwendungsgebieten wurden dabei auch die wesentlichen Begriffe und Kennzahlen beschrieben. Zu guter Letzt wurde auch der Einsatz des EVMs als Ergänzung zu gängigen Berichtsmethoden in agilen Projekten beleuchtet.

Im empirischen Teil dieser Arbeit wurde zunächst der Design Science Ansatz beschrieben. Dieser designwissenschaftliche Ansatz wurde gewählt, da in dieser Arbeit die Erstellung eines Prototyps zur Veranschaulichung des EVMs im Fokus steht. Im Anschluss zur Erläuterung des Ansatzes wurde die Umsetzung des Prototyps beschrieben. Eingangs wurden dafür die Anforderungen bestimmt. Diese konnten auf Basis der theoretischen Auseinandersetzung mit dem EVM abgeleitet werden. Da die Anforderungen auf den zuvor beschriebenen EVM Kennzahlen basieren, wurde im Zuge der Umsetzung zur einfacheren Veranschaulichung ein Fallbeispiel mit Beispieldaten dargestellt. Dabei wurden auch die nötigen Formeln zur Kalkulation der Kennzahlen erläutert. Anhand des Fallbeispiels konnten auch Ergebnisse zu unterschiedlichen Zeitpunkten im Beispielprojekt veranschaulicht werden. Im Fokus stehen dabei die Kennzahlen zur Ermittlung der fortschritts- und kostenbezogenen Basisdaten, Abweichungen, Leistungsindikatoren und Forecasts im Projekt. Im Anschluss zur Umsetzung des Prototyps wurden die Forschungsergebnisse dieser Arbeit in Form einer qualitativen Auswertung evaluiert. Dazu wurden teilstrukturierte Interviews mit Expertinnen und Experten anhand eines groben Leitfadens durchgeführt. Aus der qualitativen Inhaltsanalyse der Befragungen konnten Anwendungsgebiete für das EVM in agilen Softwareprojekten abgeleitet werden. Darüber hinaus konnten auch Restriktionen des Instruments und Erweiterungsmöglichkeiten für einen Einsatz in der Praxis identifiziert werden.

5.2 Ergebnisse und Beantwortung der Forschungsfrage

Durch die qualitative Auswertung der Interviews mit den Expertinnen und Experten konnten Anwendungsgebiete, Einschränkungen und Erweiterungsmöglichkeiten für den Einsatz von EVM in agilen Umfeldern abgeleitet werden.

Die Expertinnen und Experten nannten als mögliche Einsatzgebiete allen voran Projekte, welche mit einem vorab definierten Budget oder Umfang geplant werden. In sogenannten Fixpreis-Projekten stehen die notwendigen Ressourcen fest, wodurch sich Plan- und Zielwerte vor und während der Projektabwicklung ermitteln lassen. Ebenfalls wurde das Instrument EVM als einsatztauglich in der Überwachung und Verfolgung von agilen Projekten bewertet, da neben üblichen agilen Berichten wie Burn-Charts damit auch die Kosten betrachtet werden. Aus dem Monitoring der fortschritts- und kostenspezifischen Soll- und Ist-Werte lassen sich Abweichungen für den betrachteten Projektzeitraum ermitteln, was als weitere Einsatzmöglichkeit identifiziert wurde. Auf Basis des bisherigen Projekterfolgs und der Abweichungen können in weiterer Folge auch Vorhersagen bezüglich der Kosten und Liefertermine bestimmt werden. Zur Berechnung werden zuvor errechnete Effizienzkennzahlen herangezogen, welche für die interne

Kommunikation des bisherigen und erwarteten Projekterfolgs herangezogen werden können und einen einfach verständlichen Überblick geben.

Neben den Anwendungsgebieten wurden von den Expertinnen und Experten auch Limitationen von EVM in agilen Projekten genannt. Als solche Restriktion wurde der Einsatz in Unternehmen genannt, welche an der Entwicklung von Softwareprodukten arbeiten. Dies liegt vor allem daran, dass bei der Produktentwicklung die Lieferung von möglichst hoher Qualität im Vordergrund liegt und die Einhaltung von Plankosten oder Zeitplänen zur Lieferung von Ergebnissen nachrangig ist. Darüber hinaus wurde erwähnt, dass der Einsatz von EVM nur bedingt sinnvoll ist, wenn es während der Umsetzung häufig zu Änderungen an Umfang und Aufwand kommt. Durch den kurzfristigen und sich laufend ändernden Planungshorizont sind die erstellten Abweichungen und Vorhersagen nicht aussagekräftig. Zu guter Letzt sehen die Expertinnen und Experten auch Limitationen bei zunehmender Komplexität des Projektumfelds. Während EVM auf Teamebene ein nützliches Instrument zur Projektsteuerung sein kann, gestaltet sich der Einsatz umso schwieriger, je mehr Teams zur Umsetzung eines Projektes involviert werden.

Basierend auf den Fragen zu möglichen Erweiterungen wurde die Kalkulation und Darstellung der Business Value genannt. Durch die Ermittlung des geplanten und gelieferten Business Value könnte man für einen betrachteten Zeitraum den ROI eines Projektes basierend auf dem tatsächlichen Wert für Kundinnen und Kunden errechnen. Darüber hinaus wurden auch weitere Möglichkeiten zur Aufwandsschätzung genannt. Dies würde den Einsatz von EVM auch für Teams ermöglichen, welche ihre Schätzungen nicht auf Story Points, sondern beispielsweise auf Stundenbasis erstellen. Auch wenn die Tauglichkeit von EVM laut den Meinungen der Expertinnen und Experten mit zunehmender Anzahl an beteiligten Teams sinkt, sollte es möglich sein, dass EVM zumindest auf eine geringe Anzahl von Teams skaliert werden kann. Als abschließende Erweiterungsmöglichkeit wurde die Integration von bestehenden Tools genannt. Projektspezifische Daten wie Aufgaben oder Schätzungen sowie Zeitprotokolle könnten aus Projektmanagement- oder Zeiterfassungstools abgefragt werden, damit eine automatisierte Kalkulation der EVM Kennzahlen ermöglicht wird.

In der Problemstellung des einleitenden Kapitels dieser Arbeit wurde die Forschungsfrage wie folgt definiert:

Inwieweit kann das Earned Value Management als Werkzeug zur Steuerung von agilen Projekten in Softwareunternehmen eingesetzt werden?

Diese Frage kann durch die zuvor genannten und beschriebenen Anwendungsgebiete, Restriktionen und Erweiterungsmöglichkeiten beantwortet werden. Auch wenn der Einsatz von EVM zur Steuerung von agilen Softwareentwicklungsprojekten mit Restriktionen verbunden ist, konnten mehrere Einsatzbereiche identifiziert werden, in welchen das Instrument EVM von Nutzen ist. Die Ergebnisse dieser Arbeit haben auch gezeigt, dass der Einsatz von EVM einen positiven Einfluss auf die Steuerung von agilen Projekten in Softwareunternehmen hat. Damit kann die aus Abschnitt 1.1 beschriebene und dieser Arbeit zugrunde liegende Hypothese akzeptiert und die komplementäre Nullhypothese verworfen werden.

5.3 Diskussion und Ausblick

Die Recherchen zum Thema Agilität haben gezeigt, dass agile Praktiken in der Softwareentwicklungsbranche weit verbreitet sind und häufig zum Einsatz kommen. Auch branchenübergreifend lässt sich ein Trend von traditionellen zu agilen Vorgehensweisen erkennen, wobei häufig Praktiken aus beiden Paradigmen kombiniert werden. Ein bekanntes Framework für die agile Abwicklung von Projekten mit vordefinierten Rollen, Events und Artefakten ist Scrum. Agilität setzt auch Flexibilität voraus, daher lassen sich Frameworks wie Scrum nach Bedarf anpassen oder erweitern. Die Anwendung von traditionellen Methoden zur Steuerung von Projekten als Ergänzung agiler Modelle ist daher kein Widerspruch und in der Praxis durchaus üblich. Ein bekanntes Projektsteuerungsinstrument mit Wurzeln im traditionellen Projektmanagement ist das EVM. Es ermöglicht die Kalkulation von projektspezifischen Kennzahlen und Forecasts basierend auf Soll- und Ist-Werten für einen betrachteten Zeitraum. Die Befragungen von Expertinnen und Experten hat gezeigt, dass ein Einsatz dieses Steuerungsinstruments in agilen Projekten bestimmte Voraussetzungen mit sich bringt. So ist beispielsweise die Bestimmung der teamspezifischen Kosten und der Wert einer Umsetzungseinheit erforderlich. Der Wert der geplanten Umsetzung kann jedoch nur ermittelt werden, wenn eine gründliche Planung samt Schätzung des umzusetzenden Aufwandes im Vorfeld des Projektes möglich ist. In agilen Projekten können sich Anforderungen basierend auf Kundenfeedback jedoch sowohl inhaltlich als auch vom Umfang her kurzfristig ändern. Dies trifft vor allem auf Teams zu, welche sich auf die inkrementelle Entwicklung von Softwareprodukten fokussieren. Im Gegensatz dazu können agile Entwicklungsteams, welche mit einem im Vorfeld des Projektes festgelegten Umfangs arbeiten, sehr wohl Gebrauch vom Instrument EVM machen. Mögliche Einsatzgebiete erstrecken sich dabei vom Monitoring von Projekten über die Ermittlung von Planabweichungen und Kennzahlen bis hin zur Erstellung von projektspezifischen Forecasts. Darüber hinaus nimmt die Komplexität bei steigender Anzahl an involvierten Entwicklungsteams zu. Zusammenfassend lässt sich also sagen, dass EVM vor allem bei Softwareentwicklungsprojekten mit vordefiniertem Umfang und einer geringen Anzahl involvierter Entwicklungsteams eingesetzt werden kann, um die Projektsteuerung effektiv zu unterstützen.

Der Einsatz von EVM sollte speziell in Unternehmen untersucht werden, in welchen der Umfang und das geplante Budget von Projekten im Vorfeld definiert sind. Durch diese statischen Rahmenbedingungen können valide Schätzungen auf Ebene der einzelnen Entwicklungsteams erstellt werden. Die kalkulierten Plankosten für die Umsetzung sowie die Abweichungen zum betrachteten Zeitpunkt ermöglichen wiederum Vorhersagen zum erwarteten Projekterfolg und zu künftigen Lieferterminen. Darüber hinaus könnte der Prototyp dem Design Science Ansatz folgend erweitert werden, um mehrere Entwicklungsteams zu unterstützen. Durch eine schrittweise Annäherung könnte man ermitteln, ab welcher Anzahl involvierter Teams der Einsatz von EVM zur Steuerung von Projekten nicht mehr praktikabel ist. Zusätzlich könnten Schätzungen für den Aufwand der Umsetzungen wahlweise in Story Points oder auf Stundenbasis erfolgen und die Basisdaten aus bestehenden Produktivsystemen integriert werden. Zu guter Letzt könnte man auch Methoden untersuchen, um den Geschäftswert einzelner Softwareinkremente zu bestimmen. Durch die Gegenüberstellung des Geschäftswertes

zu den tatsächlichen Kosten könnte man in weiterer Folge den ROI eines Softwareprodukts, einer Version oder eines Inkrements berechnen. Mit dieser Funktionalität könnte der gezielte Einsatz von EVM in produktentwickelnden Softwareunternehmen evaluiert werden.

ABKÜRZUNGSVERZEICHNIS

AC	Actual Cost
BAC	Budget at Completion
CPI	Cost Performance Index
CV	Cost Variance
EAC	Estimate at Completion
ETC	Estimate to Completion
EV	Earned Value
EVA	Earned Value Analyse
EVM	Earned Value Management
EVMS	Earned Value Management System
IRR	Internal Rate of Return
NPV	Net Present Value
PV	Planned Value
ROI	Return on Investment
SPI	Schedule Performance Index
SV	Schedule Variance
TCPI	To Complete Performance Index
VAC	Variance at Completion

ABBILDUNGSVERZEICHNIS

Abbildung 1-1: Aufbau der Arbeit.....	3
Abbildung 2-1: Wasserfallmodell nach Royce (1970).....	8
Abbildung 2-2: Das Agile Enterprise Framework nach Highsmith (2010).....	10
Abbildung 2-3: Schrittweise Steigerung der Agilität nach Cobb (2011).....	11
Abbildung 2-4: Das Scrum Framework (Scrum Alliance, 2022).....	14
Abbildung 2-5: Scrum-Board nach Cobb (2015).....	17
Abbildung 3-1: Planung auf Release- und Iterationsbasis (Project Management Institute, 2021).....	23
Abbildung 3-2: Geringe Genauigkeit und hohe Präzision (Project Management Institute, 2021).....	25
Abbildung 3-3: Anwendung von Metriken in einer Feedbackschleife (Davis, 2015).....	28
Abbildung 3-4: Vorhersagen in agilen Projekten (Dimitrov, 2020).....	30
Abbildung 3-5: Burndown-Diagramm (Project Management Institute & Agile Alliance, 2017).....	34
Abbildung 3-6: Burnup-Diagramm (Rothman, 2017).....	34
Abbildung 3-7: Lead und Cycle Time (Project Management Institute & Agile Alliance, 2017).....	36
Abbildung 3-8: Hierarchie der EVM Terminologie (Wanner, 2019).....	40
Abbildung 3-9: Zusammenhang der EVM Kennzahlen nach Wanner (2019).....	44
Abbildung 4-1: Zusammenhänge in der Design Science (Johannesson & Perjons, 2014).....	48
Abbildung 4-2: EVM Kalkulation nach Abschluss von Sprint 3.....	55
Abbildung 4-3: Kennzahlen im EVM-Diagramm.....	57
Abbildung 4-4: EVM-Diagramm (Sprint 1).....	58
Abbildung 4-5: EVM-Diagramm (Sprint 2).....	58
Abbildung 4-6: EVM-Diagramm (Sprint 3).....	58
Abbildung 4-7: EVM-Diagramm (Sprint 4).....	58
Abbildung 4-8: EVM-Diagramm (Sprint 5).....	59
Abbildung 4-9: EVM-Diagramm (Sprint 6).....	59
Abbildung 4-10: EVM-Diagramm (Sprint 7).....	59
Abbildung 4-11: EVM-Diagramm (Sprint 8).....	59

TABELLENVERZEICHNIS

Tabelle 2-1: Erfolgsraten von IT-Projekten (The Standish Group International, 2020)	10
Tabelle 3-1: Bedeutung der Leistungskennzahlen im EVM (Anbari, 2003)	43
Tabelle 4-1: Sieben Leitlinien des Design Science Ansatzes (Hevner et al., 2004).....	49
Tabelle 4-2: Anforderungen an den Prototyp	51
Tabelle 4-3: Beispiel-Teamdaten	51
Tabelle 4-4: Ermittlung der Kosten	52
Tabelle 4-5: Beispiel-Sprintdaten.....	53
Tabelle 4-6: Kalkulation der EVM Kennzahlen	54
Tabelle 4-7: Anwendungsgebiete von EVM in agilen Umfeldern	66
Tabelle 4-8: Restriktionen von EVM in agilen Umfeldern	67
Tabelle 4-9: Erweiterungsmöglichkeiten von EVM in agilen Umfeldern	68

LITERATURVERZEICHNIS

Agile Alliance. (2022a). *Agile 101*. Agile Alliance. Zugriff am 4. Juni 2022, verfügbar unter <https://www.agilealliance.org/agile101/>

Agile Alliance. (2022b). *Scrum*. Agile Alliance. Zugriff am 4. Juni 2022, verfügbar unter <https://www.agilealliance.org/glossary/scrum>

Amberscript Global B.V. (2022). *Amberscript*. Amberscript Global B.V. Zugriff am 4. Juni 2022, verfügbar unter <https://www.amberscript.com/de/>

Anbari, F. T. (2003). Earned Value Project Management Method and Extensions. *Project Management Journal*, 34(4), 12–23. <https://doi.org/10.1177/875697280303400403>

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J. & Thomas, D. (2001a). *Manifesto for Agile Software Development*. Zugriff am 4. Juni 2022, verfügbar unter <http://agilemanifesto.org/>

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J. & Thomas, D. (2001b). *Principles behind the Agile Manifesto*. Zugriff am 4. Juni 2022, verfügbar unter <http://agilemanifesto.org/principles.html>

Benner-Wickner, M., Kneuper, R. & Schlömer, I. (2020). *Leitfaden für die Nutzung von Design Science Research in Abschlussarbeiten (2. Aufl.)*. IUBH Discussion Papers IT & Engineering: Vo. 2 No. 2. IUBH Internationale Hochschule.

Boehm, B. W. & Turner, R. (2004). *Balancing agility and discipline: A guide for the perplexed*. Addison-Wesley.

Carroll, J. (2017). *Earned Value Management in easy steps: Keep tabs on the real status of all projects, including agile projects*.

Cline, A. (2015). *Agile Development in the Real World*. Apress. <https://doi.org/10.1007/978-1-4842-1679-8>

Cobb, C. G. (2011). *Making sense of agile project management: Balancing control and agility*. Wiley. <https://onlinelibrary.wiley.com/doi/book/10.1002/9781118085950>
<https://doi.org/10.1002/9781118085950>

- Cobb, C. G. (2015). *The project manager's guide to mastering agile: Principles and practices for an adaptive approach* (Online-Ausg.). Wiley.
<http://site.ebrary.com/lib/alltitles/Doc?id=11004211>
- Coelho, E. & Basu, A. (2012). Effort Estimation in Agile Software Development using Story Points. *International Journal of Applied Information Systems*, 3(7), 7–10.
<https://doi.org/10.5120/ijais12-450574>
- Cohn, M. (2005). *Agile estimating and planning*. Robert C. Martin series. Prentice Hall PTR.
- Cole, R. & Scotcher, E. (2015). *Brilliant Agile project management: A practical guide to using Agile, Scrum and Kanban*. Brillian. Pearson.
<https://ebookcentral.proquest.com/lib/gbv/detail.action?docID=5174883>
- Davis, C. W. H. (2015). *Agile metrics in action: How to measure and improve team performance*. Manning Publications. <https://learning.oreilly.com/library/view/-/9781617292484/?ar>
- Dimitrov, D. (2020). *Software project estimation: Intelligent forecasting, project control, and client relationship management*. Apress.
- Fairley, R. E. (2009). *Managing and leading software projects*. Wiley; IEEE Computer Society.
<https://onlinelibrary.wiley.com/doi/book/10.1002/9780470405697>
<https://doi.org/10.1002/9780470405697>
- Freedman, R. (2016). *The agile consultant: Guiding clients to enterprise agility*. Apress; distributed to the book trade worldwide by Springer Science+Business Media New York.
- Gläser, J. & Laudel, G. (2010). *Experteninterviews und qualitative Inhaltsanalyse als Instrumente rekonstruierender Untersuchungen* (4. Aufl.). Lehrbuch. VS Verlag.
<http://www.lehmanns.de/midvox/bib/9783531172385>
- Hartmann, D. & Dymond, R. (2006). Appropriate Agile Measurement: Using Metrics and Diagnostics to Deliver Business Value. In *AGILE 2006 (AGILE'06)* (S. 126–134). IEEE.
<https://doi.org/10.1109/AGILE.2006.17>
- Hazzan, O. & Dubinsky, Y. (2014). *Agile Anywhere: Essays on Agile Projects and Beyond* (2014. Aufl.). *SpringerBriefs in Computer Science*. Springer International Publishing.
- Hevner, March, Park & Ram (2004). Design Science in Information Systems Research. *MIS Quarterly*, 28(1), 75. <https://doi.org/10.2307/25148625>
- Highsmith, J. (2010). *Agile project management: Creating innovative products* (2. Aufl.). *The agile software development series*. Addison Wesley.
- Hill, P. R. (Hrsg.). (2011). *Practical software project estimation: A toolkit for estimating software development effort & duration*. McGraw-Hill.
<https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=4658112>
- Johannesson, P. & Perjons, E. (2014). *An introduction to design science* (1. Aufl.). Springer.

- Jones, C. (2008). *Applied Software Measurement: Global analysis of productivity and quality* (3. ed.). New York.
- Kerzner, H. (2017a). *Project management: A systems approach to planning, scheduling, and controlling* (Twelfth edition). Wiley. <http://lib.myilibrary.com/detail.asp?id=995407>
- Kerzner, H. (2017b). *Project Management Metrics, KPIs, and Dashboards: A Guide to Measuring and Monitoring Project Performance* (3. Auflage). John Wiley & Sons.
- Kim, E., Wells, W. G. & Duffey, M. R. (2003). A model for effective implementation of Earned Value Management methodology. *International Journal of Project Management*, 21(5), 375–382. [https://doi.org/10.1016/S0263-7863\(02\)00049-2](https://doi.org/10.1016/S0263-7863(02)00049-2)
- Lester, A. (2017). *Project Management, Planning and Control: Managing Engineering, Construction and Manufacturing Projects to PMI, APM and BSI Standards* (7th ed.). Elsevier Science. <http://www.sciencedirect.com/science/book/9780081020203>
- Lewis, J. P. (2011). *Project planning, scheduling & control: The ultimate hands-on guide to bringing projects in on time and on budget* (5th ed (Online-Ausg.)). McGraw-Hill. <http://lib.myilibrary.com/detail.asp?id=289283>
- Lysenko, D. & Fediai, Y. (2021). Series: Engineering science and architecture. *Municipal economy of cities*, 3(163), 159–164. <https://doi.org/10.33042/2522-1809-2021-3-163-159-164>
- Marquardt, V. (2007). *Datenerhebungstechniken im Vergleich: Befragung, Beobachtung, Inhaltsanalyse*. GRIN.
- Marsanu, R. (2010). *Project Management Metrics: complementing "agile" methods with EVM for improved project performance*. Article in *Oeconomics of Knowledge*, Volume 2, Issue 2, 2Q 2010. https://www.researchgate.net/publication/46486100_Project_Management_Metrics
- Martinelli, R. J. & Milošević, D. (2016). *Project management toolbox: Tools and Techniques for the Practicing Project Manager* (Second edition). Wiley. <https://onlinelibrary.wiley.com/doi/book/10.1002/9781119174820>
<https://doi.org/10.1002/9781119174820>
- Mayring, P. (2015). *Qualitative Inhaltsanalyse: Grundlagen und Techniken* (12., überarb. Aufl.). Beltz. http://ebooks.ciando.com/book/index.cfm/bok_id/1875625
- McKenna, D. (2016). *The art of scrum: How scrum masters bind dev teams and unleash agility*. Apress; CA Press. <http://www.springer.com/>
- Meuser, M. & Nagel, U. (2009). The Expert Interview and Changes in Knowledge Production. In A. Bogner, B. Littig, W. Menz & B. Kittel (Hrsg.), *Research methods series. Interviewing experts* (S. 17–42). Palgrave Macmillan. https://doi.org/10.1057/9780230244276_2
- Microsoft. (2022). *Microsoft Excel*. Zugriff am 7. Juni 2022, verfügbar unter <https://www.microsoft.com/de-at/microsoft-365/excel>

- Moran, A. (2016). *Valuing agile: The financial management of agile projects*. TSO (The Stationery Office).
<https://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=2956367>
- Moreira, M. E. (2017). *The agile enterprise: Building and running agile organizations*. Apress.
- Müller, P. C. (2021). *Scrum und Kanban - doppelter Erfolg durch Kombination: Scrum und Kanban erfolgreich kombinieren : bessere Prozessbeherrschung im Sprint : eine Vorbereitung auf die Professional Scrum Kanban (PSK-1)-Zertifizierung (Aktualisiert für Scrum Guide V. 2020)*.
- Nee, N. Y. (2010). *Metrics for agile projects: finding the right tools for the job*. Paper presented at PMI® Global Congress 2010. Project Management Institute.
<https://www.pmi.org/learning/library/agile-metrics-progress-tracking-status-6564>
- Nikravan, B. & Forman, J. B. (2010). *Beyond backlogs and burndowns: complementing "agile" methods with EVM for improved project performance*. Paper presented at PMI® Global Congress 2010. Project Management Institute.
<https://www.pmi.org/learning/library/earned-value-management-understand-agile-6567>
- Österle, H., Becker, J., Frank, U., Hess, T., Karagiannis, D., Krcmar, H., Loos, P., Mertens, P., Oberweis, A. & Sinz, E. J. (2010). Memorandum zur gestaltungsorientierten Wirtschaftsinformatik. In *Gestaltungsorientierte Wirtschaftsinformatik: Ein Plädoyer für Rigor und Relevanz* (S. 1–6). Infowerk.
- Pham, A. T. & Pham, D. K. (2013). *Business-driven IT-wide agile (Scrum) and Kanban (Lean) implementation: An action guide for business and IT leaders*. CRC Press.
- Popović, F., Atanasijević, T. & Atanasijević, S. (2019). EARNED VALUE MANAGEMENT IN AGILE PROJECTS. In *LIMEN - International Scientific-Business Conference - Leadership, Innovation, Management and Economics: Integrated Politics of Research, 5th LIMEN Selected Papers (part of LIMEN conference collection)* (S. 17–24). Association of Economists and Managers of the Balkans, Belgrade, Serbia.
<https://doi.org/10.31410/LIMEN.S.P.2019.17>
- Pries, K. H. & Quigley, J. M. (2011). *Scrum project management*. Taylor & Francis eBooks. CRC Press Taylor & Francis Group.
<https://www.taylorfrancis.com/books/9780429103926>
<https://doi.org/10.1201/9781439825174>
- Project Management Institute. (2017). *A guide to the project management body of knowledge (PMBOK guide)* (Sixth edition). Project Management Institute.
<https://learning.oreilly.com/library/view/-/9781628253900/?ar>
- Project Management Institute. (2019). *The Standard for Earned Value Management*. Project Management Institute Inc.

- Project Management Institute. (2021). *The standard for project management*. Project Management Institute Inc.
- Project Management Institute & Agile Alliance. (2017). *Agile practice guide*. Project Management Institute Inc.
- Rothman, J. (2017). *Create your successful agile project: Collaborate, measure, estimate, deliver*. The Pragmatic Bookshelf.
- Royce, W. W. (1970). *Managing the Development of Large Software*. Technical Papers of Western Electronic Show and Convention.
- Rozenes, S., Vitner, G. & Spraggett, S. (2006). Project Control: Literature Review. *Project Management Journal*, 37(4), 5–14. <https://doi.org/10.1177/875697280603700402>
- Schwaber, K. & Sutherland, J. (2020). *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game*. Zugriff am 4. Juni 2022, verfügbar unter <https://scrumguides.org/scrum-guide.html>
- Scrum Alliance. (2022). *The Scrum Team Roles and Accountabilities*. Scrum Alliance. Zugriff am 4. Juni 2022, verfügbar unter <https://resources.scrumalliance.org/Article/scrum-team>
- The Standish Group International. (2020). *The CHAOS Report*.
- Stark, E. (2014). *Agile project management quickstart guide: A simplified beginners guide to agile project management*. ClydeBank Media LLC.
<https://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=1638983>
- Stellman, A. & Greene, J. (2021). *Learning Agile* (1st edition). Upfront Books; Safari.
<https://learning.oreilly.com/library/view/-/1449331920/?ar>
- Sulaiman, T., Barton, B. & Blackburn, T. (2006). AgileEVM - Earned Value Management in Scrum Projects. In *AGILE 2006 (AGILE'06)* (S. 7–16). IEEE.
<https://doi.org/10.1109/AGILE.2006.15>
- Timinger, H. (2021). *Modernes Projektmanagement in der Praxis: Mit System zum richtigen Vorgehensmodell* (1. Auflage). Wiley-VCH GmbH. <http://www.wiley-vch.de/publish/dt/books/ISBN978-3-527-53053-3/>
- Trendowicz, A. (2013). *Software Cost Estimation, Benchmarking, and Risk Assessment: The Software Decision-Makers' Guide to Predictable Software Development*. SpringerLink Bücher. Springer. <https://doi.org/10.1007/978-3-642-30764-5>
- Vanhoucke, M. (2009). *Measuring Time: Improving Project Performance Using Earned Value Management* (1. Aufl.). *International Series in Operations Research and Management Science Ser: v.136*. Springer.
<https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=993216>
- Vanzant Stern, T. (2017). *Lean and agile project management: How to make any project better, faster, and more cost effective*. CRC Press Taylor & Francis Group.

- <https://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=1463533>
- vom Brocke, J., Hevner, A. & Maedche, A. (2020). Introduction to Design Science Research. In J. vom Brocke, A. Hevner & A. Maedche (Hrsg.), *Progress in IS. Design Science Research. Cases* (S. 1–13). Springer International Publishing.
https://doi.org/10.1007/978-3-030-46781-4_1
- Wanner, R. (2019). *Earned value management: Die wichtigsten Methoden und Werkzeuge für ein wirkungsvolles Projektcontrolling. Startbereit zum Projekterfolg*. Proconis Publishing.
- Wieringa, R. (2014). *Design science methodology for information systems and software engineering*. Springer.
- Wysocki, R. K. (2006). *Effective software project management*. Wiley Pub; Safari Books Online.
<https://learning.oreilly.com/library/view/-/9780764596360/?ar>
- Zirkler, B., Nobach, K., Hofmann, J. & Behrens, S. (2019). Das Projektcontrolling. In B. Zirkler (Hrsg.), *Projektcontrolling: Leitfaden Für Die Betriebliche Praxis* (S. 23–38). Gabler.
https://doi.org/10.1007/978-3-658-23714-1_3
- Zizlavsky, O. & Fisher, E. (2021). *Innovation Scorecard: A Method to Measure Innovation in Agile Projects and Business Environments* (1. Aufl.). *Springer eBook Collection*. Springer International Publishing; Imprint Springer. <https://doi.org/10.1007/978-3-030-82688-8>