

Pia Čukić, Guido Pinkernell, Wiebke Werft & Anna Luther

Mathematikvorlesungen für Maschinenbau- studierende als Projektmanagement

Zusammenfassung

Besonders bei Studierenden an Hochschulen der Angewandten Wissenschaften sind die mathematischen Vorkenntnisse sehr heterogen. Mit dem traditionellen Konzept einer lehrendenzentrierten Wissensvermittlung ist ein Eingehen auf diese Bedingungen nur schwer möglich. Aus diesem Grund wird seit fünf Semestern an der Hochschule Mannheim in den Mathematikvorlesungen für Maschinenbaustudierende ein neues Vorlesungskonzept namens eduScrum erprobt, das auf der agilen Projektmanagementmethode Scrum basiert. Im Beitrag wird dargestellt, warum eduScrum als geeignete Vorlesungsmethode erscheint, woher eduScrum stammt und wie eduScrum an der Hochschule Mannheim umgesetzt wird. Weiterhin werden Rückmeldungen von Lehrenden und Lernenden über den Einsatz von eduScrum dargestellt.

Schlüsselwörter

eduScrum, Scrum, agile Hochschuldidaktik, selbstständige Studierende, nachhaltiges Lernen

Mathematics lectures for mechanical engineering students as Project Management

Abstract

Particularly among first-year-students at universities of applied sciences, the variation of mathematical knowledge is very high. The traditional concept with a lecturer standing in front of the class can hardly respond to these conditions. For this reason, a new lecture concept called eduScrum has been tested for four semesters at Mannheim University of Applied Sciences in the mathematics lectures for mechanical engineers. This concept is based on the agile project management method Scrum. It is shown why eduScrum appears as a suitable lecture method, where eduScrum was derived from and how eduScrum is implemented at Mannheim University of Applied Sciences. Furthermore, opinions of lecturers and learners about the use of eduScrum are listed at the end.

Keywords

eduScrum; Scrum; agile education; self-organized learners; sustainable learning

1. Einleitung

Studienanfängerinnen und Studienanfänger kommen mit unterschiedlichen Lernvoraussetzungen und Lernbiographien in das Studium. Severin und Teuber (2013) halten fest, dass lediglich die Hälfte der Studienanfängerinnen und Studienanfänger an Hochschulen über eine allgemeine Hochschulreife verfügt. Heublein, Hutzsch, Schreiber, Sommer und Besuch (2010) stellen fest, dass gerade Studierende an Hochschulen für angewandte Wissenschaften kaum über hinreichende mathematische und selbstregulatorische Fähigkeiten verfügen. Die Abbruchquoten sind entsprechend hoch: 42% für die ersten beiden Semester der Bachelorstudiengänge, und weiterhin hohe 31% für die folgenden dritten und vierten Semester. Mit mathematischen Vor- und Brückenkursen begegnen sowohl Hochschulen der Angewandten Wissenschaften als auch Universitäten der beschriebenen Heterogenität. Beispiele sind etwa ViaMINT (Landenfeld, Göbbels, Hintze & Priebe, 2014), VEMINT (Bausch et al., 2014). Auch im Verbundprojekt optes (Mechelke-Schwede, Wörler, Hübl, Küstermann & Weigand, 2018) werden den Studierenden elektronische Übungsaufgaben bereitgestellt. Zudem erfolgen Rückmeldungen über den Lernweg und den Leistungsstand der Studierenden durch (reale) und (elektronische) Mentorinnen und Mentoren.

Eine in unseren Augen vielversprechende Maßnahme, um auf die heterogenen Eingangsvoraussetzungen einzugehen, ist das Anbieten von Vorlesungen für Anfängerinnen und Anfängern, die in ihrem Ablauf an ein Format aus dem Projektmanagement angelehnt sind. Die Studierenden sind hierbei aufgefordert, ihre Lernaktivitäten nach einem vorgegebenen Lehrplan im Team selbst zu organisieren. Regelmäßige Überprüfungen kontrollieren das Erreichen der Lernziele. Unsere Erwartung ist, dass mit den Herausforderungen an die Selbstregulation auch die mathematischen Fähigkeiten steigen. Im Folgenden berichten wir über die Erfahrungen an der Hochschule Mannheim. Zunächst informieren wir mit Scrum über die zugrundeliegende Managementmethode und die hieraus abgeleitete Form – eduScrum – für Bildungsszenarien. Anschließend erläutern wir im Detail das für die Hochschule Mannheim konkretisierte Konzept. Erste Eindrücke schließen den Artikel ab.

2. Scrum

Allgemein legt ein Projektmanagement den organisatorischen Rahmen für Projekte jeglicher Art fest. Es gibt zum Beispiel ergebnisoffenen Entwicklungsprojekten Struktur und bietet den Handelnden Hilfestellung bei der Planung und der Umsetzung der Entwicklungsaufgaben (Broy & Kuhrmann, 2013). Methoden aus dem Projektmanagement bieten ein Rahmenwerk an, indem es Methoden und Artefakte von Entwicklungsprozessen zu standardisierten Projektablaufen zusammenfasst (Hilmer & Krieg, 2014). Grundsätzlich wird zwischen dem klassischen Projektmanagement und dem agilen Projektmanagement unterschieden. Das klassische Projektmanagement beruht weitgehend auf einem standardisierten Vorgehensmodell, das eine lineare und durch Meilensteine abgetrennte Abfolge von Projektphasen vorgibt. Gleich zu Anfang werden Ergebnisse, Ressourcen, Aufgaben etc. konkret und detailliert beschrieben. Änderungen dieses geplanten Vorgehens werden möglichst vermieden, da diese Kosten verursachen (Hilmer & Krieg, 2014). Das agile Projektmanagement hingegen zeichnet sich vor allem darin aus, dass ein spontanes Umdispo-

nieren des vorher erstellten Plans jederzeit möglich ist. Außerdem basiert es auf einer iterativen inkrementellen Vorgehensweise, wobei nach jeder Iteration ein Produkt-Inkrement fertiggestellt wird. Ein zentraler Aspekt des agilen Projektmanagements ist im Gegensatz zum klassischen Projektmanagement das selbstorganisierte Team (Hilmer & Krieg, 2014). Scrum zählt zu den agilen Projektmanagementmethoden. Scrum beschreibt innerhalb dieser Iteration Ereignisse, Artefakte und Rollen, die in einem sogenannten Scrum Guide (Schwaber & Sutherland, 2017) festgehalten sind. Das Kernstück in Scrum bilden die Iterationen. Zu Beginn einer Iteration wird beschlossen, was erarbeitet werden muss. In einer gewissen Zeitspanne namens Sprint arbeitet das Team selbstständig an den vorher definierten Aufgaben. Am Ende eines Sprints wird ein funktionsfähiges Teil des Projektes abgeliefert und es wird reflektiert, wie der Arbeitsablauf im nächsten Sprint verbessert werden kann (vgl. beispielsweise Broy & Kuhmann, 2013; Kniberg, 2015; Schwaber, 2004; Sutherland, 2014).

Agiles Projektmanagement ist in mittelständigen bis großen Unternehmen durchaus verbreitet. Laut einer Studie von Bitkom (2018) benutzen 79% der Unternehmen Scrum. Untersucht wurden dabei deutsche Unternehmen, die mehr als 500 Mitarbeitende haben und die aus Industrie, Handel und Dienstleistung stammen (genauer alle Wirtschaftszweige ohne Land- und Forstwirtschaft, Fischerei, Öffentliche Verwaltung, Verteidigung, Sozialversicherung, Erziehung und Unterricht). Dieser Studie zufolge ist Scrum mit einem Anteil von etwa 80% die mit Abstand am meisten verbreitete Methode. Hinter Scrum folgen Kanban, Extreme Programming (XP), Feature Driven Development (FDD) und Behavior Driven Development (BDD).

Scrum wird auch außerhalb der Industrie zum Management von Forschungsvorhaben verwendet (Hicks & Foster, 2010; Marchesi, Mannaro, Uras & Locci, 2007; Ribeiro Lima, de Castro Freire & Costa, 2012). Außerdem wird es mit der Variante eduScrum in vielfältigen Bildungsbereichen angewendet. In López-Alcarria, Olivares-Vicente und Poza-Vilches (2019) findet sich eine wissenschaftliche systematische Übersicht über den Gebrauch von agilen Methoden in Bildungsszenarien. Insbesondere wird dabei auf die Methoden Scrum und Kanban eingegangen. Ebenfalls finden sich Erfahrungsberichte über den Einsatz von eduScrum an einer niederländischen Schule in (De Jager, 2015). Selbach Borges, Schmitt und Nakle (2014) berichten über den Einsatz bei der Anfertigung von Projektarbeiten. In Wijnands und Stolze (2019) findet sich eine Beschreibung der Hauptkomponenten und Prozesse von eduScrum für den Schuleinsatz. Der sogenannte eduScrum Guide (Delhij, van Solingen & Wijnands, 2015) enthält ebenfalls eine Anleitung für Lehrende an Schulen.

3. eduScrum an der Hochschule Mannheim

eduScrum zeigt sich als vielversprechender Rahmen zur methodischen Organisation von Lernprozessen in schulischen und universitären Bildungsszenarien, in denen eine hohe Heterogenität bzgl. mathematischer und selbstregulatorischer Fähigkeiten vorherrscht. Ein Teil der Lernenden an der Hochschule Mannheim können ein Abitur vorweisen, der andere Teil hat eine Ausbildung abgeschlossen und hat aufgrund einer Meisterqualifikation die Zulassung zum Studium erhalten. Die Lerneingangsvoraussetzungen sind also stark heterogen. Die Anzahl der Lernenden pro Kurs beträgt in Mathematik 1 ungefähr 40 Personen.

Im Folgenden erläutern wir unser Konzept eduScrum@HSMA für den Einsatz in den Anfängervorlesungen Mathematik im Studiengang Maschinenbau an der Hochschule Mannheim. Zu Beginn unseres Lehrprojekts stand lediglich Delhij et al. (2015) als Orientierungsgrundlage zur Verfügung. Weil dort im Wesentlichen schulische Bildungseinrichtungen im Fokus stehen, waren deutliche Modifikationen für den Einsatz an Hochschulen notwendig.

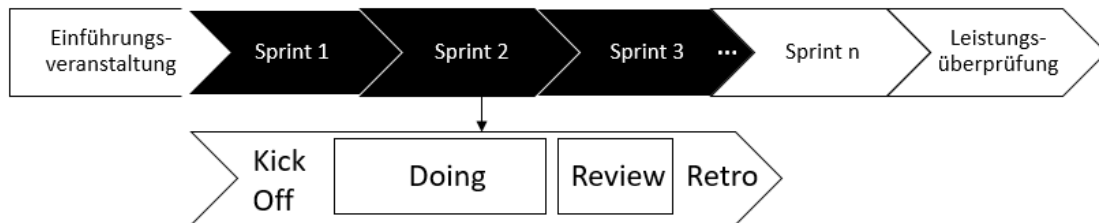


Abbildung 1: Schematische Darstellung eines Semesters

Wie im Vorbild Scrum ist der Lernprozess in sogenannte Sprints unterteilt (Abb. 1): Nach einer Einführung folgen eine Reihe von Lerneinheiten (Sprints), in denen jeweils ein eigener Inhalt thematisiert ist. Während die Länge der Sprints variabel ist, folgen alle demselben Aufbau: Kick Off, Doing, Review und Retrospektive. Zum Abschluss (Semesterende) der Vorlesung folgt eine Leistungsüberprüfung in Form einer Klausur. Im Nachfolgenden gehen wir auf die Bedeutung dieser einzelnen Phasen ein, beginnen aber zunächst mit den Rollen, die die Lehrenden und Studierenden aus der Perspektive von Scrum einnehmen.

3.1 Rollen und Rollenaufgaben

In eduScrum@HSMA gibt es zwei Hauptakteure: Die/Der Lehrende und Teams. Diese Rollen werden im Folgenden weiter erklärt. Das Rollenverständnis beruht auf Delhij et al. (2015), stellt aber gleichzeitig eine Anpassung der Rollen für den Hochschuleinsatz dar.

3.1.1. Die/Der Lehrende

Die grundsätzliche Annahme, dass sich die Lernenden die Inhalte der Vorlesung selbst erarbeiten können und müssen, definiert die Rolle der Lehrenden als Beratende und Unterstützende im Lernprozess. Entsprechend reduzieren sich ihre Aufgaben auf:

1. **Bereitstellung des Lernmaterials.** Die/Der Lehrende gibt den Lerninhalt vor, wie er im Curriculum festgelegt ist. Der Lerninhalt wurde in eduScrum@HSMA in Lernzielen operationalisiert und im sogenannten Backlog (Arbeitsvorrat) festgehalten, das im Lernmaterial, namens Booklet, zu finden ist. Das Lernmaterial (s. Kap. 3.3) wird am Anfang eines jeden Sprints zur Verfügung gestellt.
2. **Organisation der Sprint-Phasen.** Die/Der Lehrende ist ein Coach für den (Lern-)Prozess in eduScrum@HSMA und steht den Lernenden bei Fragen rund um diesen Prozess zur Verfügung. Dies ist vor allem in der Anfangsphase für die Lernenden wichtig (vgl. Rolle eduScrum Master und Product Owner in Delhij et al., 2015). Der Prozess muss den Lernenden zu Anfang erläutert werden. Die erste Stunde im Semes-

ter dient also der Anleitung und Team-Bildung. Mit Hilfe einer Präsentation erfahren die Lernenden, was eduScrum ist, was es ausmacht und warum es angewendet wird. Zusätzlich wird eine schriftliche und videobasierte Anleitung zu eduScrum@HSMA zur Verfügung gestellt (s. Hochschule Mannheim, 2019). Die/Der Lehrende legt den thematischen Ablauf fest und achtet darauf, dass dieser eingehalten wird. Von den vier Sprint-Phasen verantwortet die/der Lehrende ausschließlich das Kick Off. Dieses wird z. B. in Form einer Präsentation durchgeführt (s. Kap. 3.2.1). Zusätzlich werden das Arbeitsmaterial, die Booklets und Kurzlösungen online sowie in gedruckter Form zur Verfügung gestellt. Im Doing (s. Kap. 3.2.2) ist die/der Lehrende zum einen in beratender und zum anderen in einer wissensvermittelnden Funktion tätig. Außerdem fordert die/der Lehrende für eduScrum@HSMA den Einsatz typischer Techniken wie beispielsweise das Burn Down Chart zur Fortschrittskontrolle von den Lernenden ein (s. Kap. 3.3.3). Durch Interaktionen und gemeinsame Kurzerklärungen an der Tafel stellt die/der Lehrende während der Doing-Phase sicher, dass die Lernenden auch anspruchsvolle Inhalte angehen. Ebenso wird von der/dem Lehrenden auf die formal korrekte Schreibweise von Rechenwegen (insbesondere für Mathematik wichtig) geachtet und den Lernenden vermittelt. Das Review, eine Wissensüberprüfung (s. Kap. 3.2.3), wird von der/dem Lehrenden vorbereitet und angeleitet. Das Review überprüft zum einen den theoretischen Wissensstand durch einen kurzen Ankreuz-Test und in einem zweiten Teil können die Studierenden ihre Fähigkeit, klausurrelevante Aufgaben zu lösen, auf die Probe stellen. Die Retrospektive (s. Kap. 3.2.4), in der die Teams ihre Zusammenarbeit reflektieren, wird lediglich kurz von der/dem Lehrenden moderiert. Dabei werden Reflexionsanstöße in Form von Fragen in die Teams gegeben. Ziel ist es, dass die Retrospektive nach einiger Zeit von den Teams selbstständig durchgeführt werden kann. Die Anleitung der Retrospektive nimmt also für die/der Lehrende über die Zeit hinweg ab.

3. **Begleitung und Unterstützung des Lernprozesses der Studierenden.** Eine wichtige Aufgabe der/dem Lehrenden ist die Begleitung des Lernprozesses. Während sich die Lernenden den Inhalt selbst erarbeiten, steht die/der Lehrende jederzeit für Fragen zur Verfügung und geht dabei auf die Teams, aber auch auf einzelne Lernende ein. Charakteristisch für eduScrum@HSMA ist es, dass nicht die/der Lehrende fragt, wo die Lernenden gerade Schwierigkeiten haben, sondern dass die Lernenden Unterstützung einfordern. Im Zuge dieser Unterstützung werden Erklärungen in Form des Prinzips der minimalen Hilfe nach Trebing (2013) und Aebli (1983) in den Teams gegeben.
4. **Festlegung des Zeitrahmens eines Sprints.** Die/Der Lehrende gibt einen zeitlichen Rahmen vor, in dem ein Lerninhalt erarbeitet werden muss. Möglich sind dabei nach Delhij et al. (2015) zwei bis maximal fünf Wochen. Im Zuge der agilen Kultur ist dieser Zeitraum allerdings nicht unumstößlich. Es hat sich gezeigt, dass Teams

frühzeitig einen Sprint beenden und das nächste Thema einfordern. Durch Absprachen können Sprints für Teams vorgezogen werden. Lediglich das Verlängern eines Sprints ist im Gegensatz zu Scrum (Schwaber & Sutherland, 2017) nicht möglich, da das Semester eine feste Zeitspanne vorgibt, an dessen Ende eine Leistungsüberprüfung steht.

3.1.2. Die Teams

Eine Grundvoraussetzung von eduScrum@HSMA ist es, dass die Lernenden in Teams den Inhalt erarbeiten. Die Lernenden werden dazu nach Delhij et al. (2015) in Teams von maximal fünf Personen eingeteilt. Die Team-Bildung zu Beginn des Semesters wird von der/dem Lehrenden angeleitet. Es ist von großer Bedeutung, dass die Teams möglichst heterogen bezüglich für die in der Team-Arbeit wichtigen Softskills zusammengesetzt sind, um möglichst leistungsfähige Teams zu bilden, in denen alle von allen lernen können (Delhij et al., 2015; Schwaber & Sutherland, 2017). Die Lernenden werden gebeten sich selbst bezüglich verschiedener Fähigkeiten/Softskills wie zum Beispiel „dokumentieren“, „planen“ oder „Überblick behalten“ einzuschätzen. Auf Basis dieser Selbsteinschätzung werden die Teams möglichst divers gebildet. Einteilungen der Teams, in denen explizit nach mathematischen Fähigkeiten oder sogar Noten gefragt wird, sollten laut unseren Erfahrungen vermieden werden, um zu zeigen, dass nicht nur die mathematische Leistung in Teams hilfreich ist. Nach dem zweiten oder dritten Sprint besteht die Möglichkeit, die Teams zu wechseln oder neu zusammen zu setzen. Dies hat sich in eduScrum@HSMA als notwendig erwiesen, da sich die Lernenden vor allem in den ersten Wochen dazu entscheiden können, den Kurs oder gar das ganze Studium ganz abzubrechen. Es wurde aber auch beobachtet, dass Studienanfängerinnen und -anfänger durch das Knüpfen erster sozialer Kontakte von der Teambildung profitieren und ein Teil der Teams über die Semester hinweg bestehen bleiben.

Die Aufgaben eines jeden Teams sind die Folgenden:

1. **Erarbeitung der Lerninhalte.** Die Teams müssen wie in Delhij et al. (2015) selbst entscheiden, wie der Inhalt erarbeitet werden soll. Während die/der Lehrende Auskunft über das „Was wird erarbeitet und warum“ gibt, wird den Teams das „Wie wird etwas erarbeitet“ eigenverantwortlich überlassen. Das zur Verfügung gestellte Lernmaterial (Booklet) und die Angabe weiterer Quellen (Bücher, E-Books, YouTube Channel, Lernplattformen) sind dabei nur einige Möglichkeiten, wie sich die Lernenden im Rahmen von eduScrum@HSMA die Lerninhalte erarbeiten. Neben dem „Wie“ müssen die Teams ebenfalls sicherstellen, dass der Inhalt vollständig behandelt wird. Dazu stehen den Teams Hilfsmittel wie der Lernplan und das Burn Down Chart zur Verfügung (Kap. 3.3.3). Die Teams können während der Bearbeitung immer auf die Hilfe der/des Lehrenden zurückgreifen, falls sie an einer Stelle nicht weiterarbeiten können (Kap. 3.1.1).
2. **Einhaltung der Zeit.** Der von der/dem Lehrenden angegebene Zeitrahmen muss von den Teams eingehalten werden. Die Teams bestimmen innerhalb des Sprints das Lerntempo selbst. Ist ein Lernziel bereits aus dem schulischen Kontext bekannt

oder für die Teams subjektiv leicht zu bearbeiten, kann das nächste Lernziel bearbeitet werden. Umgekehrt kann innerhalb der vorgegebenen Zeit individuell intensiv an einem Lernziel gearbeitet werden. Das Zeitmanagement liegt folglich in den Händen der Teams.

3.2 Sprint

Ein Semester wird wie in Delhij et al. (2015) in mehrere kleine Zeiteinheiten, sogenannte Sprints, aufgeteilt. Es hat sich herausgestellt, dass die Anzahl der Themen im Curriculum oder anhand der Kapitel des Skripts dabei die Anzahl der Sprints bestimmen kann. So kann ein Semester beispielsweise in sechs oder auch nur vier Sprints aufgeteilt werden, je nachdem wie viele Themen im Curriculum vorgeschrieben sind. Ein Sprint folgt auf den nächsten (vgl. Abb. 1). Ein Sprint dauert zwei bis maximal fünf Wochen. Dies hängt vom Umfang und der Komplexität des Inhalts ab. Ein Sprint von nur einer Woche ist in eduScrum@HSMA nicht zu empfehlen, da so die einzelnen Sprint-Phasen zu kurz geraten. Ein Sprint besteht wiederum aus vier Phasen: Kick Off, Doing, Review, Retrospektive (vgl. Abb. 1 unten). Diese vier Phasen werden für jeden Sprint neu durchlaufen. Was diese vier Phasen kennzeichnet, wird im Weiteren näher erklärt. Die Sprint-Phasen in eduScrum@HSMA heben sich von denen aus Delhij et al. (2015) ab. So finden beispielsweise keine Stand-Up Meetings statt.

3.2.1 Kick Off

Das Kick Off wird in eduScrum@HSMA von der/dem Lehrenden abgehalten (vgl. Kap. 3.1.1) und umfasst ca. eine halbe Stunde am Anfang eines jeden Sprints. Die/Der Lehrende gibt den zeitlichen Rahmen für den Sprint sowie wichtige Daten wie beispielsweise Beginn und Ende der Doing-Phase sowie das Datum des Reviews für den Sprint bekannt. Außerdem umreißt die/der Lehrende den Inhalt, der erarbeitet werden soll. Die/Der Lehrende gibt einen Überblick, in welchen Bereichen des Maschinenbaus die Themen zum Einsatz kommen, um die Motivation zu erhöhen, warum etwas gelernt werden soll (Alpers, 2000). Dies ist vor allem für Mathematikveranstaltungen in den Ingenieurwissenschaften wichtig, da viele Lernende erfahrungsgemäß nicht verstehen, warum Mathematik im Maschinenbau gebraucht wird. Ebenso wird ein grundlegender Einstieg in das Thema von der/dem Lehrenden präsentiert. So erhalten die Lernenden die minimalen Grundkenntnisse für den Sprint vermittelt, sodass sie für die Literaturarbeit vorbereitet sind. Am Ende des Kick Offs teilen die/der Lehrende das Booklet aus.

3.2.2 Doing

Die Doing-Phase ist die längste Phase in einem Sprint und folgt direkt nach dem Kick Off. Die Hauptakteure im Doing sind die Teams (s. Kap. 3.1.2). Das Doing findet zu den Vorlesungszeiten statt, kann von den Teams aber auch individuell auf andere Zeiten gelegt werden. Die/Der Lehrende steht während des Doings in den Vorlesungszeiten jederzeit für Fragen zur Verfügung (s. Kap. 3.1.1).

Im Doing arbeiten die Lernenden in ihren Teams zusammen und recherchieren mit ihren Smartphones und Laptops die Inhalte, schauen Videos, lesen das Backlog (s. Kap. 3.3.1) mit den Lernzielen im Booklet, diskutieren in den Teams, erklären sich Inhalte in den Teams,

fragen die/den Lehrende(n) oder rechnen zum großen Teil an den im Booklet bereitgestellten Aufgaben. Mit Hilfe eines sogenannten Burn Down Chart (s. Kap. 3.3.3) können die Lernenden zum einen selbst den eigenen Lernstand überprüfen, zum anderen wird auch für die/den Lehrende(n) der Lernstand dadurch transparent. Das Doing lebt von der Aktivität eines Einzelnen, aber auch von der Interaktion in den Teams sowie von der Interaktion von den Teams und der/dem Lehrenden.

Die letzte Stunde des Doings ist in eduScrum@HSMA hervorzuheben, weil sie sich vom selbstständigen Arbeiten der Teams löst. In der letzten Stunde des Doings bereiten sich die Lernenden auf das anschließende Review vor. Dies geschieht in der Regel durch eine Präsentation mit Diskussion, vorbereitet von der/dem Lehrenden. Auf diese Weise werden wichtige Algorithmen, Definitionen und/oder Sätze herausgegriffen, die zum einen für das Review relevant sind, weil sie dort im ersten Teil abgefragt werden (s. Kap. 3.2.3), und zum anderen um das formale Verständnis des Inhalts zu fördern. Durch die Arbeit mit unterschiedlichen Quellen und Literaturwerken kann die Notation Unterschiede aufweisen. Daher benötigen sie eine spezielle Unterstützung beim mathematischen Formalismus. Aus den Booklets kann dieser bereits herausgelesen werden, in der Vorbereitungsstunde zum Review wird er erneut aufgegriffen.

3.2.3 Review

Das Review folgt auf das Doing und bildet wie in Delhij et al. (2015) den inhaltlichen Abschluss des Sprints. Das Review dient den Lernenden als Spiegel ihres Kenntnisstandes. Auf diese Weise erhalten sie bereits während des Semesters eine Rückmeldung, ob das Gelernte dem zu Lernenden entspricht. Das Review wird nicht im Team absolviert und ist für jeden Lernenden verbindlich, selbst wenn die Teams während der Vorlesungszeiten des jeweiligen Sprints nicht anwesend waren. Im ersten Semester stellt die erfolgreiche Teilnahme am Review sogar die Prüfungszulassung zur Klausur dar. Das Review dauert in eduScrum@HSMA zusammen mit der anschließenden Retrospektive ca. zwei Semesterwochenstunden, also eineinhalb Stunden. Das Review in eduScrum@HSMA besteht in Mathematik 1 aus zwei Teilen: Erstens einer Wahr-Falsch-Abfrage von mathematischen Ausdrücken, die in dem jeweiligen Sprint relevant waren und zweitens einer Bearbeitung von für die Klausur charakteristischen Aufgaben. In Mathematik 2 und Mathematik 3 besteht das Review lediglich aus einer Bearbeitung von charakteristischen Aufgaben. Die verschiedenen Phasen des Reviews in Mathematik 1 werden nun weiter erläutert.

1. **Wahr-Falsch-Abfrage.** Der erste Teil vom Review hat das Ziel die formale mathematische Sprache zu verstehen und Sätze und Definitionen lesen zu können. Die mathematische Sprache besteht aus Symbolen und ihren Verknüpfungen, aus denen ein Inhalt gebildet wird. Bei der Wahr-Falsch-Abfrage geht es um das genaue Lesen eines mathematischen Ausdrucks, wie er zum Beispiel in einer Definition oder in einem Algorithmus auftaucht. Die mathematischen Ausdrücke sollten den Lernenden bekannt sein, da sie zum einen in der letzten Stunde vor dem Review besprochen wurden und zum anderen, da sie diese bereits bei der Erarbeitung des Inhalts kennen gelernt haben. Die mathematischen Ausdrücke wurden teilweise leicht ver-

ändert, indem ein Symbol weggelassen, hinzugefügt oder durch ein anderes ausgetauscht wurde. Dies kann den Sinn des Ausdrucks maßgeblich verändern. Die Lernenden sollen folglich erkennen, ob der Ausdruck immer noch gültig ist oder nicht. Die Lernenden erhalten dazu jeweils elf mathematische Ausdrücke vorgelegt und müssen entscheiden, ob diese richtig oder falsch sind. Um das Review zu bestehen, müssen in Summe fünf Punkte erworben werden. Dabei gehen falsche Aussagen mit Minuspunkten in die Bewertung ein.

2. **Bearbeitung von charakteristischen Aufgaben.** Direkt im Anschluss an die Wahr-Falsch-Abfrage folgt die Bearbeitung von charakteristischen Aufgaben. Zur Selbstkontrolle für die Lernenden werden zusätzlich zum ersten Teil des Reviews zentrale Inhalte in Form von charakteristischen Klausuraufgaben abgefragt. An dieser Stelle können die Lernenden überprüfen, ob das Gelernte praktisch eingesetzt werden kann. Hierfür erhalten die Lernenden von der/dem Lehrenden zwei bis drei klausurrelevante Aufgaben für das bearbeitete Thema. Die Aufgaben sind so ausgewählt, dass sie nach erfolgreichem Durcharbeiten des Lernmaterials ohne große Probleme bewältigt werden können. Aufgaben mit hoher Komplexität sind nach den gemachten Erfahrungen in eduScrum@HSMA für ein Review ungeeignet. Alle Lernenden müssen die Aufgaben alleine bearbeiten und die Lösung niederschreiben. Für die Aufgabenbearbeitung in eduScrum@HSMA haben die Lernenden ungefähr 30 Minuten Zeit. Nach den 30 Minuten werden die Lösungen verteilt und in den Teams für ca. 15 Minuten besprochen. Bleiben noch Fragen offen, kann die/der Lehrende jederzeit gefragt werden. Die/Der Lehrende hat zu diesem Zeitpunkt die Möglichkeit, die Ergebnisse auf den Gruppentischen einzusehen und bei Bedarf weitere Erklärungen in den Gruppen zu geben.

3.2.4 Retrospektive

Die Retrospektive, kurz Retro, bildet den Abschluss eines Sprints. Während in den vorherigen Phasen die inhaltliche Bearbeitung des Themas im Vordergrund steht, sollen die Lernenden in der Retro die Zusammenarbeit in ihrem Team reflektieren. Die Retro in eduScrum@HSMA dauert ungefähr 15 Minuten und wird von der/dem Lehrenden angeleitet. Ziel ist es, dass die Teams die Retro mit fortschreitendem Semester möglichst eigenständig durchführen. Die Anleitung passiert in Form von Reflexionsanlässen zu individuellem Verhalten der Einzelnen und Interaktionen im Team. Die Reflexionen sollen in Verbesserungsvorschlägen münden, die ein Team im nächsten Sprint umsetzt. Bei den ersten Retros wird gefordert, dass der Verbesserungsvorschlag im Plenum bekannt gegeben oder der/dem Lehrenden schriftlich mitgeteilt wird. So wird die Bedeutung dieser Phase unterstrichen, da die Erfahrung gezeigt hat, dass die meisten Lernenden den Mehrwert einer Retrospektive erst mit der Zeit erkennen.

3.3 Das Lernmaterial in eduScrum@HSMA

Zu jedem Sprint erhalten die Teams von der/dem Lehrenden ein begleitendes Lernmaterial (Booklet), das auf eduScrum@HSMA zugeschnitten ist. Pro Team wird eine ausgedruckte

Version ausgeteilt. Digital steht das Booklet für alle Lernenden zur Verfügung. In einem Booklet wird zunächst der zu lernende Inhalt speziell für Maschinenbaustudierende motiviert. Das Kernstück eines Booklets sind die im sogenannten Backlog niedergeschriebenen Lernziele mit wie in einem Skript dazugehörigen mathematische Definitionen, Sätzen und Beispielen. Ferner werden ausführliche Erklärungen des mathematischen Inhalts im Booklet gegeben, die allerdings auf Inhalte beschränkt sind, die den Lernenden erfahrungsgemäß Schwierigkeiten bereiten. Zusätzlich finden sich im Booklet eine Aufgabensammlung zu jedem Thema, ein Burn Down Chart mit Aufwandsschätzung und ein Quiz. Beispiele für ein vollständiges Booklet finden sich auf der Onlinepräsenz von eduScrum (siehe Hochschule Mannheim, 2019). In den nächsten Kapiteln werden besonders das Backlog, die Aufgaben und das Burn Down Chart samt Lernplan vorgestellt. Zunächst soll ein Hinweis zu den im Booklet angegebenen Definitionen und Sätzen vorangestellt werden.

Es sind nur solche Definitionen und Sätze angegeben, die in der Literatur bezüglich der Notation oder des Verständnisses nicht eindeutig beschrieben sind. So könnte ein Vektor als Tensor ersten Grades definiert werden, was im ersten Semester einen ungeeigneten Zugang für Maschinenbaustudierende darstellt. Um ein gemeinsames Verständnis zu schaffen, wird auf eine einheitliche Notation Wert gelegt. Hingegen werden einfach zu recherchierende Sätze wie beispielsweise die skalare Multiplikation der Vektoralgebra nicht explizit im Booklet genannt.

3.3.1 Backlog

In jedem Sprint wird ein Themengebiet bearbeitet. So wird beispielsweise in Sprint 1 der Mathematik 1 Veranstaltung der Inhalt Vektoralgebra behandelt. Dieses Thema lässt sich in Unterthemen aufgliedern (zum Beispiel in Rechnen mit Vektoren und Anwendung der Vektoralgebra). Jedes Unterthema erhält in eduScrum@HSMA sein eigenes Backlog. Auf diese Weise wird die Übersicht für die Lernenden gewährleistet und die Lernmenge in kleinere Einheiten aufgeteilt. Jedes einzelne Backlog besteht aus einer Auflistung an Lernzielen. Diese Lernziele sind anhand der genormten Operatoren in Deutschland/Baden-Württemberg formuliert (vgl. Bildungspläne, 2016). Zu Anfang bestand das Backlog aus der reinen Formulierung der Lernziele. In der Weiterentwicklung des eduScrum@HSMA Konzeptes wurde das Backlog mit Definitionen, Beispielen sowie Beispielrechnungen angereichert. Diese werden minimal gehalten, um den Lernprozess nicht zu stark einzuschränken, aber diesen auch nicht komplett offen zu lassen. Durch die Minimalität sollen die Lernenden zum eigenständigen Arbeiten in der Literatur und Entdecken angeregt werden. In der Mathematik kann ein Konstrukt auf verschiedene Weisen definiert werden und im Zuge dessen auch auf verschiedene Arten formuliert werden. Um eine einheitliche Notation für den Kurs zu schaffen, werden Minimaldefinitionen und -beispiele im Backlog gegeben. Diese Minimaldefinitionen und -beispiele werden farblich von den Lernzielen abgesetzt.

Im Weiteren werden drei Beispiele für das Backlog gezeigt.

1. Beispiel einer reinen Lernzielformulierung im Backlog.

Der in Abbildung 2 dargestellte Eintrag im Backlog enthält das zentrale Stichwort (Betrag eines Vektors) in fettgedruckter Schrift sowie die korrekte allgemeine mathematische Notation für den Betrag eines Vektors. Es wird nicht weiter spezifiziert, ob es sich um einen

zweidimensionalen oder dreidimensionalen Vektor handelt. Der Operator “ausrechnen“ präzisiert das Lernziel als prozedurale Fertigkeit. Der Betrag eines Vektors bleibt bewusst undefiniert, um die Lernenden zu eigenen Recherchen zu aktivieren. Das funktioniert auch deshalb, weil die Definition für den Betrag eines Vektors in der Literatur sehr einheitlich behandelt wird. Auf ein Beispiel wird ebenso verzichtet, um die Teams anzuhalten, selbst passende Beispiele zu finden, diese nachzuvollziehen und auch selbst durchzurechnen.

Sie können den **Betrags eines Vektors**, $|\vec{a}|$, ausrechnen

Abbildung 2: Beispiel einer reinen Lernzielformulierung im Backlog

2. Beispiel einer Lernzielformulierung mit anschließender Definition im Backlog.

Der in Abbildung 3 nachfolgende Eintrag im Backlog behandelt die Rechenregeln zum Vektorprodukt. Das Vektorprodukt haben sie bereits vorher erarbeitet bzw. kennengelernt. Da bei einer Internetsuche zu diesen Stichwörtern eine Fülle an Rechenregeln möglich ist, werden hier die konkreten Rechenregeln angegeben bzw. es wird definiert, welche für die Vorlesung benötigt werden. Definitionen bzw. Rechenregeln erkennen die Teams immer an den blauen Umrahmungen.

Sie kennen die folgenden **Rechenregeln** in Bezug auf das **Vektorprodukt**.

$$\text{Antikommutativität } \vec{a} \times \vec{b} = -(\vec{b} \times \vec{a})$$

$$\text{Multiplikation mit einem Skalar } \lambda(\vec{a} \times \vec{b}) = (\lambda\vec{a}) \times \vec{b} = \vec{a} \times (\lambda\vec{b}), \lambda \in \mathbb{R}$$

$$\text{Distributivgesetz } \vec{a} \times (\vec{b} + \vec{c}) = \vec{a} \times \vec{b} + \vec{a} \times \vec{c}$$

$$\text{Keine Assoziativität } (\vec{a} \times \vec{b}) \times \vec{c} \neq \vec{a} \times (\vec{b} \times \vec{c})$$

Abbildung 3: Beispiel einer Lernzielformulierung im Backlog mit anschließender Definition

3. Beispiel einer Lernzielformulierung mit zugehörigem Beispiel.

Der Eintrag im Backlog zu den Grundrechenarten der Vektoralgebra enthält neben der Beschreibung in Textform weiter eine allgemeine Definition mit einem Beispiel (s. Abb. 4).

Sie beherrschen die **Grundrechenarten** (Addition $\vec{a} + \vec{b}$, Subtraktion $\vec{a} - \vec{b}$, skalare Multiplikation $\lambda \cdot \vec{a}$) der Vektoralgebra und deren geometrische Bedeutung (als Beispiel folgt die Addition).

$$\vec{a} + \vec{b} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} = \begin{pmatrix} a_1 + b_1 \\ a_2 + b_2 \\ a_3 + b_3 \end{pmatrix}$$

Abbildung 4: Beispiel einer Lernzielformulierung mit zugehörigem Beispiel

Auf ein Rechenbeispiel wird an dieser Stelle verzichtet, da es sich lediglich um die komponentenweise Addition der Vektoreinträge handelt. Das Beispiel für die Visualisierung der Addition ist aber unabdingbar. In den Aufgaben sind die Lernenden angehalten, die Subtraktion und die skalare Multiplikation selbst zu visualisieren, um das Lernziel zu erreichen. Dabei können sie sich am gegebenen Beispiel orientieren.

3.3.2 Aufgabensammlung

Zu jedem Backlog stehen den Lernenden Aufgaben zur thematischen Auseinandersetzung zur Verfügung. Die Aufgabenzusammenstellung in eduScrum@HSMA berücksichtigt die verschiedenen Lernvoraussetzungen der Lernenden. Dazu sind die Aufgaben mit Sternchen versehen. Lernende, die kein oder wenig Vorwissen zu einem bestimmten Thema haben, steigen bei den Ein-Sternchen-Aufgaben ein. Diese Aufgaben schulen zum einen das Identifizieren und zum anderen verlangen sie nur einfache Berechnungen. Die Zwei-Sternchen-Aufgaben sind für Lernende, die bereits mit dem Thema vertraut sind und direkt mit Standard-Berechnungsaufgaben beginnen können. Abschließend gibt es auch die Drei-Sternchen-Aufgaben mit relativ hohem Komplexitätsgrad. Das können Anwendungsaufgaben sein oder Aufgaben, bei denen mehrere Denkschritte hintereinander beachtet werden müssen. Die Lernenden kennen die Bedeutung der Sternchen-Notation. Aufgrund dessen, dass die Aufgaben auf die verschiedenen Lernenden zugeschnitten sind, ist die Anzahl der Aufgaben hoch. Bei einigen Inhalten, die auf Kenntnissen vorheriger Semester beruhen oder sich auch gegebenenfalls auf schulische Vorkenntnisse beziehen, gibt es in eduScrum@HSMA sogenannte Warm Up-Aufgaben. Diese wiederholen als bekannt vorausgesetzte Inhalte, die für das aktuelle Thema benötigt werden. Durch die Warm Up-Aufgaben wird gezeigt, dass die Inhalte in der Mathematik miteinander verknüpft sind und aufeinander aufbauen. Die/Der Lehrende kommunizieren, dass nicht alle Lernenden alle Aufgaben bearbeiten müssen. Die Lösungen zu den Aufgaben stehen den Lernenden von Anfang an in Form der reinen Ergebnispräsentation zur Verfügung. Ab der Mitte eines Sprints oder nach Anfrage von den Lernenden werden zusätzlich die ausführlichen Lösungen mit Rechenwegen freigegeben.

Die Aufgabensammlung enthält gerade in der ersten Mathematikveranstaltung viele Aufgaben, die das Faktenwissen und das prozedurale Wissen fokussieren. Konzeptuelles Wissen findet sich vor allem ab dem zweiten Semester verstärkt in den Aufgabensammlungen wieder. eduScrum ist gerade für komplexe Projekte bzw. Aufgabenstellungen gedacht (Wijnands & Stolze, 2019).

Exemplarisch sollen hier Aufgaben zu dem Backlog-Eintrag „Sie können den Betrag eines Vektors ausrechnen gezeigt werden“ (s. Kap. 3.3.1). Abbildung 5 zeigt eine Ein-Sternchen-Aufgabe. Um zu zeigen, dass sich die Lernenden wirklich um die Definition des Betrages bemühen sollen, wird dies in der Aufgabe explizit eingefordert. Wenn Lernende dies schon beim Durcharbeiten mit dem Backlog erledigt haben, kann diese Aufgabe übersprungen werden.

Aufgabe 8 (Definition: Betrag eines Vektors *)

- (a) Wie lautet die Formel zur Berechnung eines Betrags? Schreiben Sie die Formel jeweils für allgemeine zweidimensionale und dreidimensionale Vektoren auf.
- (b) Wie würde die Definition für einen n -dimensionalen allgemeinen Vektor lauten?

Abbildung 5: Beispiel für eine *-Aufgabe zu dem Backlog-Eintrag „Vektorbetrag berechnen können“

Weiter gibt es in der Aufgabensammlung charakteristische Berechnungsaufgaben, wie in Abbildung 6 zu sehen ist. Um darauf hinzuweisen, dass der Betrag nicht nur für dreidimensionale Vektoren definiert ist, werden sowohl dreidimensionale als auch zweidimensionale Vektoren in die Aufgabe eingebaut.

Aufgabe 9 (Betrag berechnen **)

Berechnen Sie den Betrag der folgenden Vektoren:

$$(a) \vec{u} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \quad (b) \vec{v} = \begin{pmatrix} -0,5 \\ 2 \end{pmatrix}$$

$$(c) \vec{w} = \begin{pmatrix} -1 \\ 2 \\ \frac{6}{3} \end{pmatrix} \quad (d) \vec{n} = \begin{pmatrix} 0 \\ 3 \end{pmatrix}$$

Abbildung 6: Beispiel für eine **-Aufgabe zum Backlog-Eintrag „Vektorbetrag berechnen können“

Außerdem stehen den Lernenden Aufgaben zur Verfügung, die ein Thema aus einer anderen Perspektive betrachten. In Abbildung 7 ist beispielsweise eine klassische Umkehraufgabe zu sehen.

Aufgabe 10 (Betrag eines Vektors **)

Welche z -Koordinate hat ein Vektor mit x -Koordinate 2, y -Koordinate -3 und Länge 7?

Abbildung 7: Beispiel für eine **-Aufgabe zum Backlog-Eintrag „Vektorbetrag berechnen können“

3.3.3 Burn Down Chart und Lernplan

Das Burn Down Chart ist eine Technik, die aus der Projektmanagementmethode Scrum übernommen wurde. Das Burn Down Chart ist im Lernmaterial enthalten und kann von den Lernenden dazu genutzt werden, den aktuellen Bearbeitungsstand für sich selbst, die

Teams und der/dem Lehrenden sichtbar zu machen. Das Burn Down Chart dient der Visualisierung bereits geleisteter Arbeit und noch verbleibender Arbeit. In einem Burn Down Chart wird auf der Abszisse die Zeit des Sprints in Tagen und auf der Ordinate der Bearbeitungsaufwand des Inhalts abgetragen (vgl. Abb. 5). Weiter ist eine Ideallinie eingezeichnet, die einen linear abfallenden Bearbeitungsstand darstellen soll. Diese Linie ist optimal und kann von den Lernenden als Referenz herangezogen werden. Liegen die Lernenden oberhalb der Ideallinie, bedeutet es, dass die Lernenden zu langsam arbeiten und dass sie mit dieser Geschwindigkeit, vorausgesetzt diese bleibt konstant, die Inhalte nicht in der vorgegebenen Zeit vollständig erarbeiten können. Um die Ideallinie zu zeichnen, werden zwei charakteristische Punkte herausgegriffen und miteinander verbunden: Der Startpunkt, an dem der Sprint beginnt ($t=1$) und der Aufwand A noch vollständig abzuarbeiten ist ($A=\max$), und der Endpunkt, an dem der Sprint vorbei ist ($t=\max$) und der Aufwand Null ($A=0$) ist.

Die Lernenden tragen selbst die bereits geleistete Arbeit in dem Burn Down Chart ein. Dadurch werden der Bearbeitungsstand und damit der Fortschritt der Lernenden transparent. Es ist nicht erkennbar, was konkret geleistet wurde, es wird jedoch sichtbar, dass etwas geleistet wurde.

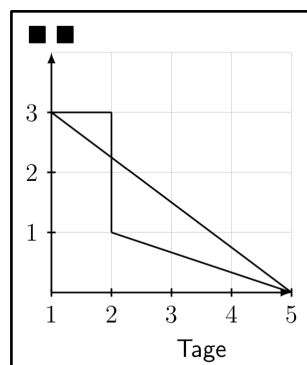


Abbildung 5: Schematische Darstellung für ein Burn Down Chart mit einer linearen Ideallinie und einem fiktiven Bearbeitungsverlauf. Die Quadrate stellen ein Symbol für den relativen Aufwand dar.

Vor der Anfertigung eines Burn Down Charts muss der Aufwand A eingeschätzt werden. Im Gegensatz zu Delhij et al. (2015) wird der Aufwand von der/dem Lehrenden kapitelweise in Form eines Lernplans (Abb. 6) eingeschätzt. Dort gibt die Anzahl der Quadrate den erwarteten Aufwand wieder. Bewährt hat sich in eduScrum@HSMA eine Anzahl von einem Quadrat bis fünf Quadraten. Wurde ein Inhalt beispielsweise mit einem Quadrat deklariert, so sollten die Lernenden nicht drei Wochen mit diesem Inhalt verbringen, wenn der Sprint nur vier Wochen lang ist; andersherum hat ein Kapitel fünf Quadrate erhalten, kann der Inhalt in der Regel nicht innerhalb von einer halben Stunde erarbeitet werden. Die Angaben sind relativ zu verstehen und sind für die Lernenden untereinander nicht gleichzusetzen. Die Anzahl der Quadrate kann unter Umständen auch für einen Lernenden nicht immer die gleiche Bearbeitungszeit bedeuten, wenn bei einem Thema schon Vorkenntnisse vorhanden sind und bei einem anderen Thema keine Vorkenntnisse vorliegen.

Thema	Aufwand	Done
Backlog1	■	
Backlog2	■ ■	
Backlog3	■ ■ ■	

Abbildung 6: Schematische Darstellung eines Lernplans mit fiktiver Aufwandsschätzung. Der Aufwand wird in Form von Quadraten von eins bis fünf angegeben.

4. Erste Erfahrungen

eduScrum@HSMA ist derzeit Gegenstand einer wissenschaftlichen Evaluation. Darin werden insbesondere die Effekte des Konzepts auf Lernleistung, Motivation, Einstellung zur Mathematik, Lernstrategien und Akzeptanz untersucht. An dieser Stelle soll eine differenzierte Darstellung erster mündlich vermittelter Erfahrungen von der Lehrenden und den Lernenden genügen. Für weitere Informationen zum Studiendesign siehe Raab, Werft, Pinnkernell und Luther (2018).

Die Lehrenden genießen vor allem die Lernaktivität und die einhergehende Interaktion mit den Lernenden. Die Lernenden kommen auf die Lehrenden zu und fordern Erklärungen ein. Dies hat unserer Einschätzung nach zur Folge, dass den Erklärungen der Lehrenden aufmerksam Beachtung geschenkt wird. Im Gegensatz zu einer klassischen Vorlesung, in der einzelne Lernende deshalb einer Erklärung nicht folgen, weil sie die Erklärung gerade nicht brauchen, im Thema eingearbeitet sind, oder sich die Frage noch nicht selbst gestellt haben. Weil die Lernenden selbstständig an den Themen arbeiten, können eventuelle Ausfälle durch Krankheiten oder Feiertage der Vorlesung gut aufgefangen werden. Es muss keine Vorlesung ausfallen. Durch die agile Arbeitsweise kann es für die Lehrenden vorkommen, dass kurzfristig Änderungen z.B. im Zeitplan eingebaut werden müssen. Ist ein Sprint von allen Teams schneller bearbeitet worden, als die Lehrenden geplant haben, muss darauf zeitnah reagiert werden. Ebenfalls kann es vorkommen, dass sich Teams in unterschiedlichen Sprints befinden. Somit müssen die Lehrenden in ihrem Wissen flexibel sein, um alle Teams in den verschiedenen Themen unterstützen zu können.

Lernende berichten über einen höheren Zeitaufwand während des Semesters im Vergleich zu anderen Vorlesungen. Gleichzeitig sei durch das nachhaltige Lernen gegen Ende des Semesters der Vorbereitungsaufwand für die Klausur geringer. Das Üben bzw. Rechnen während der Vorlesungszeiten wird von den Lernenden sehr geschätzt. Es gibt zwar ein begleitendes Tutorium, dies wird aber selten genutzt. Übungsaufgaben abzugeben ist nicht verpflichtend bzw. nicht vorgesehen. Durch die Einführung von eduScrum, erhalten die Lernenden mehr Routine beim Rechnen von Aufgaben während des Semesters. Die Hürde, Fragen zu stellen sinkt, da zum einen die Lernenden im geschützten Kreis ihrer Teams Fragen stellen können und sich nicht exponieren müssen und zum anderen, da die Lehrenden durch die Nähe zu den Lernenden direkt ansprechbar werden. Einige Studierende berichten ebenfalls, dass sie sich bei der Erledigung von Arbeitsaufträgen dem Team gegenüber stärker verpflichtet fühlen als in anderen Vorlesungen den Arbeitsaufträgen der Lehrenden gegenüber. Wiederum andere Studierende bevorzugen es alleine zu arbeiten

und scheuen die Team-Arbeit. Dadurch, dass sich Lernende gegenseitig Erklärungen geben, fühlen sie sich kompetent. Allerdings kann es vorkommen, dass falsche Erklärungen in den Teams verbreitet werden. Dies muss von den Lehrenden aufgefangen werden, indem sie immer wieder mit den Teams in Diskurs treten. Die Lernenden benötigen nach eigenen Angaben zwei oder drei Sprints bis sie den Zyklus von eduScrum@HSMA verinnerlicht haben. Die anfängliche Suche nach geeigneter Literatur oder weiterführenden Materialien fällt den Lernenden allerdings weiterhin schwer. Sie wissen oft nicht, dass Bücher in der Bibliothek ausgeliehen oder als e-Book heruntergeladen werden können. Die Menge an weiterführendem Material überfordert die Lernenden oft. Sie wissen nicht, welche Quellen geeignet sind und welche nicht. Diesem Feedback wurde durch das Einarbeiten geeigneter Quellenangaben, direkt im Backlog, begegnet.

5. Zusammenfassung

Aufgrund der heterogenen Lerneingangsvoraussetzungen im Fach Mathematik an Hochschulen der Angewandten Wissenschaften wurde an der Hochschule Mannheim in den Grundlagenvorlesungen eine aus dem Projektmanagement stammende Methode namens eduScrum in der mathematischen Grundvorlesung von Maschinenbaustudierenden eingeführt.

In eduScrum@HSMA erarbeiten sich Lern-Teams, die von der/dem Lehrenden vorgegebenen Lernziele selbstständig und eigenverantwortlich. Ein Semester wird dazu in unterschiedlich lange Sprints je nach mathematischem Thema eingeteilt. Ein Sprint ist ein iterativer, sich wiederholender, Prozess von vier feststehenden Phasen, Kick Off, Doing, Review und Retrospektive, wobei die ersten drei Phasen die inhaltliche Erarbeitung darstellen und die letzte Phase auf die Zusammenarbeit im Team fokussiert. Im Kick Off motiviert die/der Lehrende den mathematischen Inhalt und geben eine Einführung in das zu erlernende Thema. Zusätzlich wird das Lernmaterial (Booklet) ausgeteilt, das u.a. die Lernziele kapitelweise im Backlog zusammenfasst und Aufgaben zu den jeweiligen Lernzielen enthält. Die Lernziele sind dabei so visualisiert, dass die Lernenden direkt Definitionen und Algorithmen sowie Beispiele erkennen und unterscheiden können. Ferner sind die Lernziele so formuliert, dass die Lernenden einen Anhaltspunkt für ihre Recherchen haben. Im Doing erarbeiten sich die Teams den mathematischen Inhalt und dürfen dabei selbst entscheiden, wie der Inhalt erarbeitet wird. Das Doing in eduScrum@HSMA zeichnet sich durch eine starke Interaktion sowohl innerhalb der Personen im Team als auch zwischen den Teams und der/dem Lehrenden aus. Als Unterstützung und zur Übersicht stehen den Teams dabei der Lernplan inklusive Aufwandsschätzung und das Burn Down Chart zur Verfügung. Durch die regelmäßigen Reviews wird der Lerninhalt kontinuierlich in Form von Klausuraufgaben überprüft und es wird den Lernenden frühzeitig der eigene Wissensstand gespiegelt. In der Retrospektive erarbeiten die Teams einen Verbesserungsvorschlag für die gemeinsame Zusammenarbeit im nächsten Sprint.

Sowohl auf Seiten der Lehrenden als auch auf Seiten der Lernenden gibt es Vor- und Nachteile, die im Zuge von eduScrum@HSMA berichtet werden. Als größter Vorteil ist die Interaktion zwischen Teams und Lehrenden zu nennen und als größter Nachteil gelten die anfänglichen Schwierigkeiten mit der Recherche von zusätzlichem Material.

Literatur

- Aebli, H. (1983). *Zwölf Grundformen des Lehrens: eine allgemeine Didaktik auf psychologischer Grundlage*. Klett-Cotta.
- Alpers, B. (2000). Verknüpfung von Mathematik und ingenieurwissenschaftlichen Anwendungsfächern mit Hypertext und Computeralgebra. In *Tagungsband Computeralgebra-Symposium* (Hrsg.), Konstanz.
- Bausch, I., Biehler, R., Bruder, R., Fischer, P. R., Hochmuth, R., Koepf, W. & Wassong, T. (2014). VEMINT–Interaktives Lernmaterial für mathematische Vor- und Brückenkurse. In I. Bausch et al. (Hrsg.), *Mathematische Vor- und Brückenkurse. Konzepte und Studien zur Hochschuldidaktik und Lehrerbildung Mathematik* (S. 261-276). Wiesbaden: Springer Spektrum.
- Bildungspläne Baden-Württemberg (2016). *Operatoren*. <http://www.bildungsplaene-bw.de/Lde/LS/BP2016BW/ALLG/GYM/M/OP>
- Bitkom Research GmbH (2018). Etengo-Freelancer-Index. Ein Index zur Bedeutung von Freelancern für deutsche Unternehmen von Bitkom Research im Auftrag von Etengo. 5. Folgemessung. https://www.etengo.de/fileadmin/user_upload/Change_Enabler/EFX/EFX_6.PDF
- Borges, K. S., Schmitt, M. A. R. & Nakle, S. M. (2014). Eduscrum projetos de aprendizagem colaborativa baseados em scrum. *Revista Novas Tecnologias na Educação*, 12(1). doi: 10.22456/1679-1916.49839
- Broy, M. & Kuhmann, M. (2013). *Projektorganisation und Management im Software Engineering*. Heidelberg: Springer.
- Delhij, A., van Solingen, R. & Wijnands, W. (2015). *Der eduScrum Guide*. Verfügbar unter http://eduscrum.nl/en/file/CKFiles/Der_eduScrum_Guide_DE_1.2.pdf [18.05.2020]
- De Jager, T. W. (2015). *Using eduScrum to introduce project-like features in Dutch secondary Computer Science Education*. (Masterthesis). Universität Utrecht.
- Heublein, U., Hutzsch, C., Schreiber, J., Sommer, D., Besuch, G. (2010). *Ursachen des Studienabbruchs in Bachelor- und in herkömmlichen Studiengängen*. (Projektbericht). HIS - Hochschul-Informationssystem.
- Hicks, M. W. & Foster, J. S. (2010). SCORE: Agile Research Group Management. *Commun. ACM*, 53(10), 30-31. doi: 10.1145/1831407.1831421
- Hilmer, S. & Krieg, A., (2014). Standardisierung vs. Kultur: Klassisches und agiles Projektmanagement im Vergleich. In M. Engstler, E. Hanser, M. Mikusz & G. Herzwurm (Hrsg.), *Projektmanagement und Vorgehensmodelle 2014 - Soziale Aspekte und Standardisierung*. Bonn: Gesellschaft für Informatik e.V. (S. 47-57).
- Hochschule Mannheim (2019). *Mathematikvorlesungen für Maschinenbaustudierende mit eduScrum*. <https://www.eduscrum.hs-mannheim.de/materialien.html> [18.05.2020]
- Kniberg, H. (2015). *Scrum and XP from the Trenches* (2. Aufl.). Lulu.com.
- Landenfeld, K., Göbbels, M., Hintze, A. & Priebe, J. (2014). viaMINT–Aufbau einer Online-Lernumgebung für videobasierte interaktive MINT-Vorkurse. *Zeitschrift für Hochschulentwicklung*, 9(5), 201-217. doi: 10.3217/zfhe-9-05/12
- Lima, I. R., de Castro Freire, T. & Costa, H. A. X. (2012). Adapting and using scrum in a software research and development laboratory. *Revista de Sistemas de Informação da FSMA*, 9, 16-23.

- López-Alcarria, A., Olivares-Vicente, A. & Poza-Vilches, F. (2019). A Systematic review of the use of agile methodologies in education to foster sustainability competencies. *Sustainability*, 11(10), 2019. doi: 10.3390/su11102915
- Marchesi, M., Mannaro, K., Uras, S. & Locci, M. (2007). Distributed Scrum in research project management. In: G. Concas, E. Damiani, M. Scotto & G. Succi (Eds.), *Agile Processes in Software Engineering and Extreme Programming. XP 2007. Lecture Notes in Computer Science* (S. 240-244). Heidelberg: Springer.
- Mechelke-Schwede, E., Wörler, J. F., Hübl, R., Küstermann, R. & Weigand, H.-G. (2018). Das Verbundprojekt optes – Optimierung der Selbststudiumsphase: Konzepte, Inhalte und Ideen. In Fachgruppe Didaktik der Mathematik der Universität Paderborn (Hrsg.), *Beiträge zum Mathematikunterricht 2018*. Münster: WTM-Verlag. doi: 10.17877/DE290R-19526
- Raab, P., Werft, W., Pinkernell, G. & Luther, A. (2018). eduScrum-ein methodischer Rahmen in einer Mathematik-Vorlesung in der Grundausbildung von Ingenieuren. In Fachgruppe Didaktik der Mathematik der Universität Paderborn (Hrsg.), *Beiträge zum Mathematikunterricht 2018*. Münster: WTM-Verlag. doi: 10.17877/DE290R-19591
- Schwaber, K. (2004). *Agile project management with Scrum* (1. Aufl.). Microsoft press.
- Sutherland, J. (2014). *Scrum: The Art of Doing Twice the Work in Half the Time*. New York: Crown Business.
- Trebing, T. (2014). Tutorien: Das Prinzip der minimalen Hilfe in der universitären Rechenübung. In O. Zitzelsberger (Hrsg.), *Neue Wege in der tutoriellen Lehre in der Studieneingangsphase* (S. 101-115.). Münster: WTM-Verlag.
- Wijnands, W. & Stolze, A. (2019). Transforming Education with eduScrum. In D. Parsons & K. MacCallum (Eds.), *Agile and Lean Concepts for Teaching and Learning* (pp. 95-114). Singapore: Springer.

Autor/-innen

Pia Josephine Čukić, geb. Raab, M.Sc.. Hochschule Mannheim, Institut für mathematisch-naturwissenschaftliche Grundlagen, Mannheim, Deutschland; Email: pcukic.eduScrum@posteo.de

Prof. Dr. Guido Pinkernell. Pädagogische Hochschule Heidelberg, Institut für Mathematik, Heidelberg, Deutschland; Email: pinkernell@ph-heidelberg.de

Prof. Dr. Wiebke Werft. Hochschule Mannheim, Fakultät Maschinenbau, Deutschland; Email: w.werft@hs-mannheim.de

Prof. Dr. Anna Luther. Hochschule Mannheim, Fakultät Maschinenbau, Mannheim, Deutschland; Email: a.luther@hs-mannheim.de



Zitiervorschlag: Čukić, P.; Pinkernell, G.; Werft, W. & Luther, A. (2020). *Mathematikvorlesungen für Maschinenbaustudierende als Projektmanagement*. Die hochschullehre, Jahrgang 6/2000, online unter: www.hochschullehre.org