Scrum-based Agile Maturity Assessment in Physical Product Development

Franziska Scharold[1,*], Julian Schrof[2], Kristin Paetzold-Byhain[1]

[1] Institute of Machine Elements and Machine Design, Technische Universität Dresden
[2] Institute for Technical Product Development, Universität der Bundeswehr München

** Corresponding author:*
Franziska Scharold
Institute of Machine Elements and Machine Design
George-Bähr-Straße 3c
01062 Dresden
☎ *+49 (0)893/82-40697*
✉ *franziska.scharold@mailbox.tu-dresden.de*

Abstract

Agile methods are increasingly being implemented in physical product development disciplines, such as automotive development. Once agile methods have been successfully introduced, teams strive to determine the status quo of their agile maturity. For this purpose, agile maturity models have been developed, but these models primarily refer to software development. This work aims at deriving a model that is suitable for teams in physical product development and is based on the agile framework Scrum. The presented Agile Maturity Model consists of eleven dimensions and five levels. In addition, the dimensions that have a high influence on the agile maturity of teams were identified during a study. These are: Agile Events and Iterative Approach, Backlog Management, Team and Agile Values and Principles.

Keywords

Agile Maturity Assessment, Physical Product Development, Scrum Maturity Model

 | DOI: 10.35199/dfx2023.04

1. Introduction

Today’s business environment is often characterized by a highly uncertain and volatile setting and the ability to swiftly respond to changes becomes crucial. Agile development is an approach to purposively react to these changes. This development approach accepts change and even perceives change as a chance for adapting the products to the customer’s needs. Agile development methods originally emerged in the field of software development and are based on fundamental values and principles that are outlined in the so-called "Manifesto for Agile Software Development" [1]. The agile approach has become an established standard in software development. Due to the promising benefits of agile software development, more and more companies are using agile methods for developing physical products [2]. The application of these methods is associated with fast, flexible and customer-centric development. For this reason, the status quo of agility is of increasing interest to companies in physical product development applying agile methods and the agility of teams from the software (SW) and hardware (HW) domains must be equally assessable.

Agile maturity models (AMMs) are suitable for the purpose of a status quo measurement in terms of agility [3].There are predominantly AMMs that relate specifically to software development. These models, however, are hardly applicable or only to a limited extent for the assessment of agile maturity in the development of mechatronic products. Therefore, existing AMMs from the software realm are insufficient. Within the context of agile development, Scrum [4] is the most frequently used framework [5]. In the literature, no AMM based on Scrum has yet been found that is equally applicable to both software and hardware development teams. Since agile approaches are particularly focused on the team level, this perspective is relevant. Although the model by Schmidt et al. [6] refers to agile physical product development at the team level, the model's contents are not especially based on Scrum. The **research objective** of this paper is to develop a Scrum based AMM that can be used to assess the agility of development teams (hardware and software). Derived from the research objective, the paper at hand is addressed to all readers who are interested in a status quo assessment of the agile maturity of a scrum-based development team in mechatronics.

The following **research questions** (RQs) are addressed in this study:

- **RQ 1:** What should a Scrum-based AMM look like that assesses the agility of hardware and software development teams?
- **RQ 2:** Which specific characteristics have a relevant influence on the agile maturity of development teams?

2. State of the Art

2.1. Agile Development Methods

In 2001, 17 programmers agreed upon fundamental values and principles that are essential to develop under ever-changing circumstances. This agreement was written down in the manifesto of agile software development. The manifesto emerged due to the ineffectiveness and inflexibility of traditional, plan-driven development approaches in dynamic and uncertain conditions and symbolizes the basis of agile methods in general. Iterative, incremental development, continuous customer integration and self-organized teams are central elements of the methodology. [1] The Scrum framework is considered to be the most widely used agile method in physical product development [2].

Scrum is also an incremental and iterative approach that emphasizes inspection, adaption and transparency. A cross-functional, small, empowered and self-organized team is a fundamental part of Scrum. The Scrum team consists of a Scrum Master, a Product Owner and developers. The Scrum Master is responsible for establishing and improving the team's practices according to the Scrum framework, while the Product Owner is responsible for

effective product backlog management. There is also a commitment from the developers to produce every aspect of a usable increment during each sprint. A sprint consists of time-boxed and predefined events. The initiating event of a sprint is a sprint planning meeting. Based on the prioritized product backlog items and a plan for delivery in the sprint backlog, the Scrum team defines a sprint goal at this event. The Scrum team monitors progress towards the sprint goal, identifies impediments during the daily Scrum event and adjusts work throughout their development phase. Before concluding a sprint, the sprint review event takes place, where the Scrum team presents the results of its work. The increment is discussed and inspected with stakeholders. The product backlog may be adjusted based on the feedback. The sprint retrospective marks the end of the sprint and allows the Scrum team to reflect on the last sprint and plan ways to improve the quality and effectiveness of their collaboration. The time-boxed and recurring nature of the events ensures consistency, regularity and predictability. [4]

2.2. Agile Maturity Models

Maturity can be understood as the transition from an initial state to another, more advanced state. The concept thus expresses a gradual development through intermediate states. [7] Maturity models show a desirable, typical and logical development path to maturity [8]. In addition, the models can be used to derive actions and improvements for specific issues. Moreover, a maturity model can be utilized to determine an actual level of maturity and then to plan for and achieve a target level of maturity. [9] Maturity models typically include four characteristics: dimensions, levels, maturity principle and assessment tool. A maturity model initially consists of a few maturity levels and several structuring dimensions. Dimensions are specific areas of capability that organize and group the subject area in question. The levels or degrees of maturity consist of an explicit description of the characteristics. The maturity principle of such a model may be continuous or graduated. Qualitative descriptions or quantitative methods may be used as assessment tools. [10] [11]

The Capability Maturity Model Integration (CMMI) is also used to define different levels of maturity for describing the maturity of established processes [12]. Similarly, several AMMs have been developed to provide an objective assessment of agility in enterprises. AMMs typically describe the requirements for achieving a higher level of maturity and act as a roadmap for improvement. [3] A variety of AMMs from academia and consultancy exist - around 40 have been published in the academic field [13]. Systematic literature reviews and case studies have been conducted by authors such as Leppänen [14], Schmitt et al. [3] and Ozcan-Top and Demirörs [15]. However, none of these AMMs are suitable for assessing the agile maturity of development teams in mechatronics. This is either because they focus exclusively on agile software development, or they are assessed at an overall company level. There are AMMs for assessing the agility of development teams. With the exception of the model cited by Schmidt and Paetzold [6], none of them take into account the requirements of physical product development. Furthermore, they focus on agile development in general rather than on Scrum.

3. Research method

To reach the goal of this study, a Scrum-based AMM at team level must be designed first. For this purpose the design process according to Lahrmann et al. [16] was chosen. The design process is divided into five general stages: identifying the problem, scoping, model design and evaluation [16]. The fifth phase, evolution, is not considered in the paper at hand. **Problem identification** has already been covered in the introduction. The **objective** of the model is to assess the agile maturity of teams through self-assessment. Everyone involved, regardless of their hierarchical position, should find the AMM and its contents easy to understand and comprehend. The model should also be as objective as possible, despite the self-assessment

of the teams. In addition, agility should be assessed across a comprehensive spectrum. This allows an actual state to be captured and possible improvement measures to be derived based on an individually defined target state. For the **model design**, the top-down approach was chosen, which is widely used according to de Bruin et al. [17]. The **model** was **evaluated** by pilot users in the investigated automotive company, who provided initial feedback on the content. Based on the feedback, changes were made to the AMM to increase acceptance.

For answering the second RQ, eight development teams of an automotive manufacturer were identified. The teams differ from each other as they work in different areas, but they have all worked on the same project. The experience in using agile methods ranged from six months to eight years among the teams. Two of the teams develop hardware, while the remaining six teams are involved in software development. The sizes of the teams varied between seven and nine team members. The contents of the AMM served as a guideline for the pre-structured interviews. As the sample size consisted of eight teams, eight pre-structured interviews were conducted by the corresponding author as a moderator. Qualitative data was collected exclusively as part of retrospectives and the team interviews took 1.5 to 2h each. During the interviews, the teams discussed the eleven dimensions of the AMM in chronological order, starting with the Roles cluster, followed by Culture and Process. For each dimension, the characteristics of the five levels were also discussed sequentially and without exception in the order given and assessed by the teams based on the three evaluation levels.

The interview's qualitative results were quantified using a scoring method resulting in clear numerical values for the agile maturity of the teams. Depending on the degree to which the characteristics of the levels are fulfilled, each field is either associated with the scoring levels 0 (not yet existing), 0.5 (in progress) or 1 (completely fulfilled). To assess a single dimension numerically, the five scores per field are summed. To determine a team's agile maturity, the arithmetic mean is calculated across all eleven dimensions. Therefore, a team's agile maturity score ranges from 0 to 5. A high score is associated with a highly agile team. A correlation calculation is then used to investigate whether certain characteristics of the development teams have a relevant influence on their agile maturity. Therefore, the intensity of the relationship between each dimension and the agile team maturity score is examined. Since this study deals with interval-scaled data, the correlation is measured according to Pearson's method. The correlation coefficient r can take on values between -1 and +1 and is calculated using the formula below. [18]

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

The sample size is represented by n and $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ as well as and $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ are the mean values of the variables x_i and y_i respectively [18]. Table 1 shows the classification of the correlation coefficients (absolute value) and their corresponding interpretation based on Cohen [19]. Given the small sample size, significance was not tested.

Table 1: Correlation coefficient and interpretation [19]

Absolute value correlation coefficient r	**Interpretation**
$\|r\| \leq 0{,}5$	Weak correlation
$0{,}5 < \|r\| \leq 0{,}8$	Moderate correlation
$0{,}8 > \|r\|$	Strong correlation

4. Findings

4.1. AMM set-up

According to the chosen top-down approach, the first step is to identify the **agile levels** or maturity levels. These are based on the levels of the AMM by Patel and Ramachandran [20]. The maturity levels of this model are derived from CMMI and its evolutionary idea is familiar to many people working in the software environment. The agile levels and their corresponding descriptions are listed in Table 2. The degree of agility is low at the Initial level but increases over time and is highest at the Sustained level.

Table 2: Agile levels and corresponding level names

Agile levels	Level 1	Level 2	Level 3	Level 4	Level 5
Level names	Initial	Explored	Defined	Improved	Sustained

Level 1 is the first step towards Scrum-based agility and is the basic requirement for reaching further levels. *Explored* builds on the first step and already shows the first changes compared to the starting point. Substantial changes compared to Initial can be seen in *Level 3*. The fourth level of the AMM is called *Improved*. Significant changes from Level 1 can be seen here. The fifth step, and therefore the fifth level of the model, is called *Sustained* since the Manifesto for Agile Software Development emphasizes the importance of continuous improvement. The content of each level must also be assessable based on different characteristics. For this purpose, three evaluation categories are defined (see section 3). '*not yet existing*' means that the team does not implement or apply the characteristics of the level in question. If the level in question is partially implemented or applied by the team, it is '*in progress*'. The level in question is '*completely fulfilled*' if all persons in the team apply or implement the characteristic of that level.

Following the top-down approach, the next step is to derive the **dimensions** and their characteristics. The conceptual design of the AMM led to a model of eleven dimensions, which are derived from the Scrum Guide's theory, definition, values as well as events and associated artifacts and the Scrum team. [4]. Furthermore, the AMM at team level was divided into the clusters of Roles, Culture and Process. According to the Scrum Guide, the relevant Roles are the *Product Owner*, the *Agile Master* and the *Team* itself. In addition, the Role of the *Disciplinary Leader* was included, as this role is still found in traditional organizations. The Scrum Guide also states that *Agile Values and Principles*, as well as *Feedback, Improvement and Adaptation* are important in terms of Culture. *Accountability and Self-organization* are also relevant in this context. When looking at Processes within the team, the dimensions of *Product*, *Backlog Management*, *Customer Integration*, *Agile Events and Iterative Approach* need to be considered. [4]

The entire AMM at team level with its levels, dimensions and their characteristics can be found in Table 3 below.

Table 3: Proposed AMM at team level

Cluster	Dimension	**Initial**	**Explored**	**Defined**	**Improved**	**Sustained**
Roles	Team	The team consists of 4-9 members, PO and Agile Master. Everyone is in the team at least 50% of the time.	The focus is on the work of the sprint. The team collaborates cross-functionally. There is a daily information exchange.	Everyone is in the team at least 80% of the time, there are no sub-teams. The team is routinised and they support each other.	The team is a functioning unit that has a common goal and works together collaboratively.	The team develops in a self-organized way.
	Product Owner (PO)	The PO knows his product, his role and responsibilities as PO.	The PO fulfils his responsibilities in the team (e.g. prioritization, vision), and is responsible for maximizing the value of the product.	The PO fulfils his responsibilities externally (e.g. customer integration, stakeholders).	The PO is recognizable as the PO and is accepted as such and his decisions are respected.	The PO continues to grow in his role and acts as an ambassador and mentor to others in agile development.
	Agile Master	The Agile Master knows his team, his role and responsibilities as an Agile Master.	He creates the basic conditions for agile development. There is a regular exchange with the team and PO.	The Agile Master creates an effective working environment (e.g. addressing impediments) for the team and is accepted in his role.	The Agile Master is also active beyond his team (e.g. in communities).	The Agile Master continues to develop himself and the agile approach, also across teams in the organization.
	Disciplinary Leader (DL)	The DL is supporting agile development and knows the reasons and advantages of this approach.	The DL knows that agile teams are managed differently: they give the team freedom for self-organization and personal responsibility.	The DL follows an agile approach, removing barriers and ensuring a functioning team with all the necessary resources.	The DL communicates its commitment to an agile approach and defends it in the face of internal and external opposition.	The DL is an ambassador of the agile principles and the agile approach and actively drives the agile transformation (e.g. mindset change).
Culture	Agile Values and Principles	The agile values and principles are known to the team.	There is an open communication about the agile values and principles. They also regularly reflect on their own actions.	There is a strong understanding within the team of how to live the values and principles. Adherence to the values is actively encouraged.	Day-to-day business is guided by the values and principles and there is regular feedback.	There is more focus on being agile than on doing agile.
	Feedback, Improvement and Adaption	There is regular reflection within the team and challenges are identified	Agile events are used to derive feedback, improvements and adjustments. Mistakes are seen as learning opportunities and are addressed openly.	Measures from the reflections are prioritized, sustainably implemented and impediments eliminated.	All team members are actively involved in the improvement of work practices, tools and processes.	Through open feedback, a good error culture and regular reflection, the working environment gets better every day.
	Accountability and Self-organization	The team takes responsibility.	The team is self-organized and delivers the agreed results consistently and reliably.	The team solves challenges independently and ensures high quality.	The team takes end-to-end responsibility for their product.	The team has been empowered by the organization to be able to manage its own work.
Process	Product	There is one or more products with a clear and unambiguous description.	The product(s) are clearly differentiated and there is a clear vision. The team knows its product(s).	The product(s) and its/their sub-products and interfaces are clearly described.	The product(s) and its sub-product(s) can be developed with minimal dependencies.	A continuous (partial) product integration takes place, within which the product (cut) is optimized.
	Agile Events and Iterative Approach	There is an agreed, steady cadence. The agile events are agreed and their goals are known.	All roles act as agreed in the events. The agile events in the iterative process create tangible transparency.	The team works on part of its tasks in the cadence and the goals of the agile events are continuously achieved.	The cadence is protected: The team works entirely in cadence. Rituals are routinised without any noticeable extra effort.	The entire schedule of events and meetings is being improved and simplified on an ongoing basis.
	Backlog Management	Everyone in the team knows where and what the backlog is.	Product requirements are derived and prioritized by the PO as backlog tickets. For each iteration, the prioritized backlog tickets are delivered to the team.	Tickets are regularly updated by all team members. Tickets comply with the definition of ready and with the definition of done.	The backlog contains all relevant information about the product. Issues have a clear relationship to the product vision, the iteration goal is clearly defined.	All issues are prioritized by the PO via the backlog. Backlog Management is continuously adapted to the needs of the team.
	Customer Integration	For each product, it is clear who the customer is.	The PO is in contact with the customer on an ongoing basis.	The customer was involved in the development of the product vision.	The customer provides regular feedback on the status of the product(s).	The customer is willing to invest time to take an active role in product development.

4.2. Application of the AMM at team level

Applying the AMM to a team was originally planned to take 1.5 hours to complete. However, this time frame was not always adhered to. As the proposed procedure is a self-assessment of the team, the discussion of the content was highly relevant. Depending on the size of the team and the participation of individual members, the implementation took up to 2 hours. Not every team member always took part in the discussion, therefore not every opinion was included in the assessment. The characteristics of the dimensions were to be commented on and rated by each team member; this was explained to each team during the introduction of the method. On the other hand, some team members were dominant with their respective contributions, causing individual opinions to dominate the overall assessment. In addition, it was found that some team members occasionally tried to give a particularly high rating when the criteria for doing so were clearly not met. In such cases, the moderator pointed out the deficiency and asked for a new evaluation. The teams found the proposed method very helpful for reflecting about their collaboration. Furthermore, the teams perceived the procedure as a kind of roadmap for agile team development. In particular, the dimensions in the cluster of Roles were often discussed the longest. Some teams struggled to assess the Product and Customer Integration dimension. The survey showed that the limitations of physicality were a particular obstacle. In addition, it was often unclear to the teams who the customer of their product was since none of the teams had direct contact with the end customer. For some characteristics of the dimensions, the teams wished for a more precise description with examples. The interviews also revealed that team members often held back in their assessment of the Product Owner and the Disciplinary Lead if that role was present during the interview. The interviews also revealed that, due to the content of the guided process, in-depth team issues were brought up for discussion. The interviewees reported that the targeted questions and the external moderator had contributed to this and eventually had a positive influence on the reflection of the cooperation.

Table 4 presents the results of the data collection. The eight development teams were coded as T1 to T8. Table 4 shows the results of the eleven dimensions based on the calculated scores (see Section 3).

Table 4: Results from all development teams: Score of dimensions and agile maturity score

Cluster	Dimensions	T1	T2	T3	T4	T5	T6	T7	T8
Roles	Team	0.5	3	1.5	3.5	2.5	4.5	4.5	5
	Product Owner	0	4.5	3	4.5	3.5	4	2.5	5
	Agile Master	0	3	1.5	3.5	1	4	0	5
	Disciplinary Lead	3.5	1	5	2.5	3.5	4.5	4	1
Culture	Agile Values and Principles	1	0.5	3	2	5	4	4	5
	Feedback, Improvement and Adaption	1	3.5	1	1.5	3.5	3	4.5	4.5
	Accountability and Self-organization	3	2.5	2.5	4	5	4	5	5
Process	Product	2	3.5	4	3	4	3	4.5	4
	Agile Events and Iterative Approach	0.5	3	4	3	4.5	4.5	5	5
	Backlog Management	0.5	3	3	2.5	4	3.5	4	3.5
	Customer Integration	3	3.5	3.5	4	3.5	2	5	4.5
	Agile Maturity Score	1.36	2.82	2.91	3.09	3.64	3.73	3.91	4.32

 | DOI: 10.35199/dfx2023.04

In addition, the team's agile maturity score was calculated. The results show that the agile maturity of the teams is evenly distributed between 2.82 and 4.32. Only T1 has a relatively low score of 1.36. Looking at the clusters, the highest scores were achieved by the Process cluster, followed by Culture. The cluster Roles had the lowest scores. This is mainly due to the fact that the Role of the Agile Master was often insufficiently trained or even non-existent in the teams studied.

5. Discussion

The interviews showed that there was broad acceptance of the model among the interviewees. The concept was understandable to the participants. In addition, the interviews revealed that the chosen formulations are applicable to both software and hardware development teams. The results of the data collection were consistent. It can be concluded that the validity of the AMM at team level is given for the automotive OEM teams investigated. Users benefit from the maturity model by knowing the status quo of their agile maturity and by being able to derive potentially hidden opportunities for improvement. Nevertheless, the application of the model is not about achieving the highest agile maturity score. Rather, it is about finding the appropriate level of agility in relation to Scrum for each team individually. The maturity model guides teams on the path to agile transformation, as agility is not a binary state [21]. The AMM helps organizations to identify teams that perform outstandingly and can act as role models. In this example, team T8 would be particularly relevant.

The study also investigated whether there were specific characteristics that positively influenced the agile maturity of teams. For this purpose, the agile maturity of all teams was compared with the team-specific scores on each dimension (see Table 4). A correlation calculation was conducted to examine the relationship between these variables. The results of this calculation are shown in Table 5.

Table 5: Correlation calculation on each dimension

Dimension	**Correlation coefficient r**	**Interpretation**
Agile Events and Iterative Approach	0.962	Strong positive
Backlog Management	0.913	
Team	0.876	
Agile Values and Principles	0.815	
Feedback, Improvement and Adaption	0.773	Moderate positive
Product	0.759	
Accountability and Self-organization	0.734	
Product Owner	0.719	
Agile Master	0.479	Weak positive
Customer Integration	0.381	
Disciplinary Lead	-0.127	Weak negative

The dimensions *Agile Events and Iterative Approach*, *Backlog Management*, *Team* and *Agile Values and Principles* show a strong positive correlation with the agile maturity score. In this context, dimensions such as *Product* show a moderate positive correlation, while the correlation for *Agile Master*, for example, is weakly positive. On the other hand, there is a weak negative correlation for the *Disciplinary Lead*. The reasons for this are manifold. First, the interviews revealed that some managers felt a loss of power as a result of the accountability

and self-organization of the teams. Second, a holistic assessment was not possible because the disciplinary managers were not always present during the interviews. Although *Agile Values and Principles* are considered as the foundation of agile methods, this research showed that this dimension does not have the strongest influence on the agile maturity of a team (r=0.815). Instead, the strongest influencing factor is *Agile Events and Iterative Approach* (r=0.962). The teams reported that the Scrum events provided structure and stability to the working environment and that the iterative approach reinforced this. The *Backlog Management* (r=0.913) is an important tool for this purpose, but also the development of the *Team* itself (r=0.876). In this context, the weak correlation for the *Agile Master* (r=0.479) was surprising, as he is considered to be the establisher of the Scrum process. A weak correlation is also found for *Customer Integration* (r=0.381). However, as mentioned in section 4.2, this is due to a lack of understanding of the definition of the customer. *Feedback, Improvement and Adaptation* (r=0.773), *Product* (r=0.759), *Accountability and Self-organization* (r=0.734) and *Product Owner* (r=0.719) are in the moderately positive range.

6. Conclusion, limitations and future work

The aim of this paper was to design a Scrum-based maturity model at team level. The model designed consists of eleven dimensions that are emphasized in the Scrum Guide. These dimensions can be evaluated using three different evaluation criteria in five levels. Besides designing the model, the AMM was also utilized to assess eight teams from the development department of an automotive OEM. The consistent results obtained in the broad application of the model demonstrate the validity of the proposed AMM concerning the teams studied. The data collection shows that the agile maturity of the surveyed teams can be quantified within a spectrum from 1.36 to 4.32. The maximum score of 5 was not reached. In addition, it was investigated whether there are team-specific characteristics which have a relevant influence on the agile maturity of development teams. The strength of the correlation between the agile maturity score and each of the dimensions of the model was examined for this purpose. A strong positive correlation with a correlation coefficient of $r \geq 0.8$ is shown by the dimensions of Agile Events and Iterative Approach, Backlog Management, Team, and Agile Values and Principles. Consequently, these are considered to be key elements of the agile maturity of development teams.

In terms of limitations, it should be noted that the AMM was only applied to eight teams from the same company. Despite the generally valid formulations and the proven validity of the model, the teams are working in the software environment in six out of eight cases. From a methodological point of view, it should be noted that the qualitative data collection is a self-assessment by the respondents. Consequently, the assessments made are not always objectively verifiable. Future research should therefore focus on improving the objectivity of the AMM. This can be overcome through an improved standardized data collection procedure. In addition, the application of the model to other areas of agile physical product development should be a focus for further development of the model.

References

[1] 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. ; Mellor, S. ; Schwaber, K. ; Sutherland, J. ; Thomas, D.: *Manifesto for Agile Software Development*. URL agilemanifesto.org – Überprüfungsdatum 2023-05-15

[2] Michalides, Marvin ; Nicklas, Simon J. ; Weiss, Stefan ; Paetzold-Byhain, Kristin: *Agile Entwicklung physischer Produkte*. 2022

[3] Schmitt, Anna ; Theobald, Sven ; Diebold, Philipp: Comparison of Agile Maturity Models, Bd. 11915. In: Franch, Xavier; Männistö, Tomi; Martínez-Fernández, Silverio (Hrsg.): *Product-Focused Software Process Improvement*. Cham : Springer International Publishing, 2019 (Lecture Notes in Computer Science), S. 661–671

[4] Schwaber, Ken ; Sutherland, Jeff: *The Scrum Guide* (2020). URL https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf – Überprüfungsdatum 2023-05-21

[5] Komus, Ayelt: *Status Quo (Scaled) Agile 2019/2020* : *4. Internationale Studie zu Nutzen und Erfolgsfaktoren (skalierter) agiler Ansätze*. URL https://www.process-and-project.net/studien/studienunterseiten/status-quo-scaled-agile-2020/ – Überprüfungsdatum 2023-05-15

[6] Schmidt, Tobias Sebastian ; Paetzold, Kristin: Maturity assessment of teams developing physical products in an agile manner. In: *2017 International Conference on Engineering*, S. 351–360

[7] Antoniades, Pericles: SOA, Maturity Models, SOA MM and Relevant Work. In: Antoniades, Pericles (Hrsg.): *SOA Maturity Model*. Cham : Springer International Publishing, 2014 (SpringerBriefs in Information Systems), S. 9–17

[8] Becker, Jörg ; Knackstedt, Ralf ; Pöppelbuß, Jens: *Developing Maturity Models for IT Management*. In: *Business & Information Systems Engineering* 1 (2009), Nr. 3, S. 213–222

[9] Ahlemann, Frederik ; Schröder, Christine ; Teuteberg, Frank: *Kompetenz- und Reifegradmodelle für das Projektmanagement* : *Grundlagen, Vergleich und Einsatz*. Osnabrück : ISPRI, 2005 (ISPRI-Arbeitsbericht 2005,01)

[10] Lahrmann, Gerrit ; Marx, Frederik: Systematization of Maturity Model Extensions, Bd. 6105. In: Hutchison, David; Kanade, Takeo; Kittler, Josef; Kleinberg, Jon M.; Mattern, Friedemann; Mitchell, John C.; Naor, Moni; Nierstrasz, Oscar; Pandu Rangan, C.; Steffen, Bernhard; Sudan, Madhu; Terzopoulos, Demetri; Tygar, Doug; Vardi, Moshe Y.; Weikum, Gerhard; Winter, Robert; Zhao, J. Leon; Aier, Stephan (Hrsg.): *Global Perspectives on Design Science Research*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2010 (Lecture Notes in Computer Science), S. 522–525

[11] Marx, Frederik ; Wortmann, Felix ; Mayer, Jörg H.: *A Maturity Model for Management Control Systems*. In: *Business & Information Systems Engineering* 4 (2012), Nr. 4, S. 193–207

[12] CMMI Product Team: *CMMI for Development, Version 1.3*. Software Engineering Institute (Hrsg.) : Carnegie Mellon University

[13] Schweigert, Tomas ; Vohwinkel, Detlef ; Korsaa, Morten ; Nevalainen, Risto ; Biro, Miklos: Agile Maturity Model: A Synopsis as a First Step to Synthesis, Bd. 364. In: McCaffery, Fergal; O'Connor, Rory V.; Messnarz, Richard (Hrsg.): *Systems, Software and Services Process Improvement*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2013 (Communications in Computer and Information Science), S. 214–227

[14] Leppänen, Mauri: A Comparative Analysis of Agile Maturity Models. In: Pooley, Rob; Coady, Jennifer; Schneider, Christoph; Linger, Henry; Barry, Chris; Lang, Michael (Hrsg.): *Information Systems Development*. New York, NY : Springer New York, 2013, S. 329–343

[15] Ozcan-Top, Ozden ; Demirörs, Onur: Assessment of Agile Maturity Models: A Multiple Case Study, Bd. 349. In: Woronowicz, Tanja; Rout, Terry; O'Connor, Rory V.; Dorling, Alec (Hrsg.): *Software Process Improvement and Capability Determination*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2013 (Communications in Computer and Information Science), S. 130–141

[16] Lahrmann, Gerrit ; Marx, Frederik ; Mettler, Tobias ; Winter, Robert ; Wortmann, Felix: Inductive Design of Maturity Models: Applying the Rasch Algorithm for Design Science Research, Bd. 6629. In: Jain, Hemant; Sinha, Atish P.; Vitharana, Padmal (Hrsg.): *Service-Oriented Perspectives in Design Science Research*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2011 (Lecture Notes in Computer Science), S. 176–191

[17] de Bruin, Tonia ; Michael Rosemann ; Ronald Freeze ; Uday Kulkarni: Understanding the main phases of developing a maturity assessment model. In: *ACIS 2005 Proceedings - 16th Australasian Conference on Information Systems*, 2005

[18] Schober, Patrick ; Boer, Christa ; Schwarte, Lothar A.: *Correlation Coefficients: Appropriate Use and Interpretation*. In: *Anesthesia and analgesia* 126 (2018), Nr. 5, S. 1763–1768

[19] Cohen, Jacob: *Statistical Power Analysis for the Behavioral Sciences* : Routledge, 2013

[20] Patel, C. ; Ramachandran, M.: Agile maturity model (AMM): a software process improvement framework for agile software development practices. In: *Int. J. of Software Engineering 2(1)*, S. 3–28

[21] Schweigert, Tomas ; Nevalainen, Risto ; Vohwinkel, Detlef ; Korsaa, Morten ; Biro, Miklos: Agile Maturity Model: Oxymoron or the Next Level of Understanding, Bd. 290. In: Mas, Antonia; Mesquida, Antoni; Rout, Terry; O'Connor, Rory V.; Dorling, Alec (Hrsg.): *Software Process Improvement and Capability Determination*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2012 (Communications in Computer and Information Science), S. 289–294