Do Daily Scrums Have to Take Place Each Day?
A Case Study of Customized Scrum Principles at an E-Commerce Company

Daniel Pauly
University of Cologne
pauly@wiso.uni-koeln.de

Bjoern Michalik
University of Cologne
michalik@wiso.uni-koeln.de

Dirk Basten
University of Cologne
basten@wiso.uni-koeln.de

Abstract

Agile development approaches, such as Scrum, continue to gain importance in today’s world. Since previous research has predominantly treated development approaches as a black box, we answer the call for empirical research concerning adoption of agile methods. The study’s aim is to assess the adoption or adaptation of Scrum principles at an e-commerce company. The findings of our in-depth single case study reveal that not all Scrum principles are suitable in each context. By discussing the reasons for adopting or adapting the principles, we contribute to a deeper understanding of agile methods and help to open the black box of development approaches in information systems research. The in-depth insights gained at our case company provide practitioners a useful reference for adapting agile methods to their specific contexts.

1. Introduction

Popularity of agile development approaches has increased considerably in recent years [1–3]. As shown by a growing number of research studies [4], this is particularly the case for Scrum [5], which has been successfully applied to different types of companies and projects [1]. Due to this success, Scrum principles have been adopted in other domains, including product management [1, 6].

Although detailed textbook descriptions for Scrum exist, findings of former studies indicate that methods commonly in use are often not enacted as prescribed, but are rather a combination of practices implicit in different agile methods [7]. Often, this is done to adapt the methods to specific project, organization, and business settings [8]. Inevitably, this has led to the emergence, adoption, and use of various versions and implementations of a textbook Scrum method.

Moreover, development processes are often considered as a direct relation between input and output variables [9–11], thereby neglecting their inherent complexity. According to Siau et al. [10, p. 88], “The [development] process is either skimmed through in literature or mated as a black box”. Consequently, in-depth understanding and the theoretical foundations of agile development approaches are lacking [3, 12, 13].

For the purpose of closing this research gap, several areas for future research have been proposed. Among others, emphasis should be placed on empirical studies [14] related to the adaptation of agile methods [15]. Taking these calls as motivation for our study, and in line with the growing stream of Scrum research, we pose the following research question: How do organizations apply Scrum principles in product management?

We answer these questions by conducting a single, in-depth case study. Its findings contribute to a deeper understanding of concrete applications of Scrum principles in practice and help develop a better understanding of reasons for adopting or adapting the agile principles. We thus provide in-depth insights into daily Scrum work in product management for practitioners and help to advance Scrum’s theoretical underpinnings for researchers.

2. Product Management Scrum Principles

2.1. Artifacts

2.1.1. Product vision – An overarching guideline for product development [16, p. 24]. It describes the future product in a concise and simple manner to facilitate understanding [16, p. 27]. Thus, on one hand, the product vision serves as a starting point, while representing an overarching guideline during the entire development process on the other. It should be visible, for instance, in a document [16, p. 24]. Furthermore, a visioning sprint [16, p. 36] can be conducted if a more extensive effort for envisioning the product is required.

2.1.2. Product roadmap – A planning artifact that illustrates how the product will evolve within the subsequent versions. It is none of Scrum’s original principles, but rather a special extension for product management. It should include realistic launch dates of each version, a list of target customers (see 2.3.5) and their needs, as well as small selection of top features. A product roadmap should not be seen as a static document. Rather, it evolves as the understanding of the product increases [16, p. 41]. All stakeholders should be
involved in the creation and maintenance process to ensure their commitment.

2.1.3. Product backlog – A list of product changes for next releases. Such changes can be functional and non-functional requirements covering business as well as technical issues [16, p. 47]. It can also include vague ideas for future features [5, p. 33]. Thus, the product backlog is—in line with the Scrum’s preference for frequent customer feedback—a dynamic document with evolving requirements [5, pp. 32-33]. Furthermore, it is a prioritized list wherein higher priority items are described in more detail and implemented first [16, p. 41].

2.1.4. Backlog items – Part of product backlog, including the necessary requirements for product development. Different from traditional, sequential development approaches [17], requirements in Scrum are evolving; thus, backlog items are added, changed, or deleted as additional information becomes available. This additional information can emerge, for instance, from customer feedback [16, p. 51], grooming workshops of the Scrum team (see 2.3.1), and feedback from stakeholders in sprint review meetings (see 2.2.4) [16, p. 51]. If backlog items become overly extensive or have more than one dedicated goal, they must be broken down to fit a single sprint (see 2.2.2) [16, p. 62].

2.1.5. Definition of ‘done’ – List of tasks needed to be accomplished to retrieve a working increment at the end of each sprint. It thus serves Scrum’s objective to provide a potentially shippable product at the end of each sprint. It includes requirements, as well as efforts for testing and documentation, all of which ensure the increments’ quality [16, p. 49].

2.1.6. Sprint backlog – A list of all activities that must be completed for reaching a sprint’s goal. These activities arise from the product backlog and are sufficiently detailed to be approached in the respective sprint [5, p. 33]. The sprint backlog is created in the sprint planning meeting (see 2.2.1) and is updated daily [16, p. 101], since backlog items may not be sufficiently detailed at the beginning of the sprint [5, p. 49]. Finally, the sprint team is the sole authority for adding, modifying, or removing backlog items from the sprint backlog during the sprint [5].

2.2. Events

Events’ primary purpose is to enable transparency and work inspection [18, p. 29].

2.2.1. Sprint-planning meeting – Meeting in which the Scrum team plans the work for the following sprint [16, p. 98]. The resulting sprint goal is a commitment of the team for the subsequent sprint, which fosters its self-organization, and should be realistic with regard to the team’s capacities and capabilities to hold up motivation. The major document originating from this meeting is the sprint backlog [5, p. 47].

2.2.2. Sprint – Period of time the Scrum team works after having conducted the sprint-planning meeting. Its recommended duration should not exceed thirty days, but can be adapted as experience with Scrum increases [5, p. 47]. During a sprint, the team has full authority to do whatever is appropriate to reach the agreed sprint goal. Burndown charts are used to consider how the progress of the team relates to its intended results.

2.2.3. Daily Scrum – Daily 15-minute meeting of the Scrum team, allowing the members to inspect the progress of work and to remove impediments [5, p. 41]. It is established by the Scrum master (see 2.3.4) and takes place each day at the same place and time, thus avoiding unnecessary coordination. Each team member answers the following three questions: (1) What have you worked on since the last meeting? (2) What will you work on until the next meeting? (3) Have any impediments occurred?

2.2.4. Sprint review meeting – A four-hour meeting in which the development team presents the work of the preceding sprint to its stakeholders (management, customers, users, and product owner; see 2.3.2). Each group evaluates the results from a different perspective [5, p. 54]. Subsequently, stakeholders provide feedback and specifications for the following sprints [16, p. 101]. If the participants, for instance, realize that some items have not met the definition of ‘done’, these are returned to the product backlog and reprioritized [16, p. 102].

2.2.5. Sprint retrospective meeting – A meeting intended to allow the participants to evaluate effectiveness and efficiency of the team collaboration during the last sprint [16, p. 103]. Problems, their causes, and appropriate solutions to be implemented in the next sprint are identified. Thus, the meeting contributes to better collaboration and enhanced job satisfaction in the following sprints [16, p. 103].

2.2.6. Release – Incremental transition of potentially shippable product from the team into use by customers. In Scrum, early releases are key to success. Releases allow the team to collect early customer feedback and enable the product to evolve [16, pp. 79-80]. Compared to prototypes, releases are potentially shippable product increments that nevertheless require high quality standards [16, pp. 80-81]. Quality and time are the fixed variables in Scrum. Nonetheless, early products may provide fewer features than intended in the final version.

2.3. Roles

The Scrum team consists of a product owner, a Scrum master, and a development team. We also describe customers, since their feedback is instrumental.

2.3.1. Scrum team – Self-organizing and cross-functional teams. Their objective is to bring the product to life [16, p. 7]. Usually, they consist of five to nine people, as smaller teams lack interaction and synergy
effects, while larger teams make Scrum mechanisms inappropriate [5, pp. 36-37]. Additionally, the team composition should be stable to avoid losing productivity and self-organization capabilities [16, p. 8]. It should also possess all professional skills necessary to reach the sprint goals [5, p. 37]. Team members do not have a job description; rather, they should be able to participate in each task [5, p. 38]. Furthermore, the Scrum team is authorized to make any decision necessary to meet the sprint goals. This also includes decisions pertaining feature implementation [5, pp. 35-36]. The team members must trust each other and work in close collaboration [16, pp. 7-8]. For this reason, all Scrum team members should be collocated in a common team room, facilitating communication, and interaction and making work more enjoyable.

2.3.2. Product owner – Represents the perspective of the customer in the Scrum team. The product owner is officially accountable for the product and has to ensure that the right product is built in the right way [16, p. 2]. The product owner is responsible for involving all stakeholders into the development process. His or her main tasks are creating the product vision, managing the product backlog, planning the release, preparing the product launch, attending to the different sprint meetings, and collaborating with the team [5, p. 34, 16, p. 4]. Thus, he or she must possess the necessary knowledge and experience, as well as be a visionary, a leader, a team player, a communicator, and a negotiator [16, p. 4]. Additionally, it is crucial that the product owner has the necessary management commitment and authority for decision-making [16, pp. 5-6].

2.3.3. Chief product owner – Guides the product owners in the process of coordinating and synchronizing the work of different teams. The chief product owner facilitates the collaborative decision-making and has the power to make final decisions if, for instance, no consensus is reached among product owners [16, p. 13].

2.3.4 Scrum master – Responsible for ensuring that Scrum is applied in the way that yields the highest return. Compared to the product owner, the Scrum master is not responsible for the ‘what’, but rather for the ‘how’ [16, p. 13]. That means that the Scrum master needs to establish values, practices, and rules of Scrum within the team [5, p. 31]. The Scrum master has to establish all relevant Scrum meetings and foster self-organization by facilitating the work of the team. Furthermore, he or she is responsible for resolving impediments arising from the Daily Scrum, protecting the team from outer and inner disturbances, and ensuring that the team stays healthy and motivated by intervening when, for instance, pace of work is too high [16, p. 9].

2.3.5. Customers – Recipients of the product being developed. As Scrum thrives on continuous inspection and adaptation, it is mandatory to integrate customers as early as possible in the development process. Thus, the team receives early feedback and a detailed understanding of what customers need and can therefore develop the product in a way that provides the highest customer benefit.

The Scrum principles serve as framework for exploring the Scrum application in our case company.

3. Research Approach

We conducted a descriptive single case study, that is “an empirical inquiry that investigates a contemporary phenomenon in depth and within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident” [19, p. 18]. It is therefore especially well-suited if research problems under investigation are complex, dynamic, and cannot be studied outside the context in which they occur [20]. Furthermore, case studies are the preferred method of investigation when the aim is to answer ‘how’ and ‘why’ questions [19, 21]. Concerning the underlying philosophical paradigm, we position our work ontologically and epistemologically as positivist. The positivist approach stems from natural sciences [22, 23]. Therein, it is believed that an objective reality exists independently from the observer and that the world can be comprehended by identifying cause-effect relations not bounded in context and time [23].

3.1. Research site

The case company (henceforth referred to as TrustMeCorp) is a medium-sized e-commerce company founded in 1999. It is one of the leading providers of certification services for online shops in Europe. Aiming to build consumer trust in the e-commerce industry, it certifies online shops by a extensive number of criteria and secures purchases with its money-back guarantee. In 2013, sales constituted 30 million Euro while, in 2014, its workforce surpassed 200 employees for the first time in the company’s history. Since its establishment, it has expanded its business to ten European countries and presently covers 13-15% of the German online stores with its seal of approval. With offices in three major European cities, TrustMeCorp has certified more than 17,000 online shops thus far. Operating in the dynamic e-commerce market, it is continually forced to quickly respond to upcoming market changes. This holds especially true for the company’s Product & Technology department, where new and existing products are developed. In order to ensure this high responsiveness to market changes, the company has introduced Scrum as an agile development framework. Although parts of Scrum have already been used in earlier projects in 2012, an extensive utilization was first introduced in mid-2013. At that time, the product management...
department was severely extended by introduction of new employees. In addition, it merged with the information technology department.

Four product teams have been set up (henceforth referred to as Team A, B, C, and D). Each of the four teams consists of seven to nine members, mainly located in the Product & Technology department. However, so called ‘field experts’ are assigned from other departments, so that the teams are inter-divisional and, through their personal competencies, cross-functional as well. While the team members were less experienced in Scrum at its introduction, the Scrum processes in the company have gradually improved over time.

3.2. Research design

We conducted a holistic single case study, as this approach is appropriate if the case is unique [24]. Our research focuses on such an example, as the application of Scrum in the particular research situation of our chosen company can only be investigated in this context. Moreover, it focuses on the product teams (unit of analysis) of one of the leading providers of trust solutions for e-commerce in Europe. Throughout our research, we had unlimited access to all resources and relevant information, as one of the paper co-authors worked part-time as one of the teams’ Scrum master. As an employee in the company, he could contribute invaluable insights into the functioning of his own, as well as other teams.

3.3. Data collection

Our data sources included participant observations, semi-structured expert interviews, and documents. Participant observation comprised collaboration of one of the co-authors in the case company, who had worked as Scrum master since the introduction of Scrum in 2013. Thus, he was able to take part in most of the formal and informal meetings, workshops, and training sessions. He took comprehensive field notes throughout these events, as recommended by Yin [19] as well as Miles and Huberman [25]. Moreover, he conducted numerous informal discussions with all project members. Although participant observation as a data collection tool is inherently biased, as participants develop an emotional relation to people, processes, and contexts [20], we took care to objectively consider and evaluate actions and processes. As a result, we are confident that the participant observation did not conflict with our underlying positivist stance (e.g., [26, 27]).

Finally, we had unrestricted access to all the Scrum teams’ documents. This enabled us to investigate all documents (e.g., presentations and minutes) in order to identify information pertinent to our research question.

3.4. Data analysis

As recommended [19, 21, 25], during our study, data analysis and data collection overlapped, for instance, due to the researcher on-site taking field notes. Raw data (i.e., interview transcripts, field notes, and documents) was coded using the software NVivo. Codes were derived from data [29]. Researcher triangulation was broadly applied, as two researchers independently coded each data item and consolidated their results. This ensured all results to be developed independently from the subjective view the researcher on-site may have generated due to his strong involvement.

4. Results

In this section, we describe how the four Scrum teams at our case company adapted the Scrum principles (cf. Table 2).

4.1. Artifacts

4.1.1. Product vision. In contrast to recommendations in the extant literature (cf. section 2), none of the four teams studied formulates a product vision in a written form. However, a common understanding exists concerning the products’ goals and strategies. A frequently stated reason for not using a product vision in its proposed form is team members’ implicit
Table 2. Summary of the Scrum application

<table>
<thead>
<tr>
<th>Scrum principle</th>
<th>Team</th>
<th>Team</th>
<th>Team</th>
<th>Team</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
</tr>
<tr>
<td>Product vision</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>Product roadmap</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Product backlog</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Definition of “done”</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Sprint Backlog</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Sprint planning meeting</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Sprint review meeting</td>
<td>O</td>
<td>X</td>
<td>--</td>
<td>X</td>
</tr>
<tr>
<td>Sprint retrospective meeting</td>
<td>O</td>
<td>X</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

X: fully applied, O: partially applied, --: not applied

4.1.2. Product roadmap. A product roadmap is used in all four teams and covers a period of three months. In three of the four teams, the product owner initiates its creation and guides the team by predetermining the key topics. All team members are subsequently involved in the creation process by delivering valuable input from their perspectives. In the fourth team, product owner and Scrum master prepare a draft of the roadmap and involve the team in later stages only. In all teams, the product roadmap is created on a regular basis.

A product roadmap is seen as a medium-term planning tool, which offers clear goals (AC3, B1, C1, D2) and thus prepares the team for work in the next three months (A1, B3, D2). Moreover, it serves as an internal marketing instrument (D1, D2), helping illustrate what the teams are working on (A1, C1, B1). Thus, the product roadmap is a helpful tool for creating transparency, both within and outside the teams. It signals reliability (AC3) and helps justify the team’s goals (AC3, BC2, D3): “The roadmap is like a wall that protects the team” (D1, Team D product owner).

Despite its application in all teams, disadvantages of the product roadmap are apparent. The interviewees frequently mentioned problems related to estimating the development effort. If requirements are unavailable or insufficiently clear, estimates are either unrealistic (D1, D2) but perceived as realistic within the company (D1), or not provided at all, with their absence hard to explain (B1). Additionally, product roadmaps require time for creation (C1) and defer the teams’ work progress (B1).

As reasons for the involvement of team members in developing a product roadmap, the interviewees stated that team members are experts in their area of action and deliver a valuable input (B1, BC2, D1, D3); thus, direct involvement motivates the team (B1, BC2, C1, D2), and leads to common commitment (C1, D2, D3).

In contrast, the complexity and time required for the creation process increases with the number of people involved (B3, D1, D2). Additionally, teams can also be demotivated if ideas proposed by certain members are not reflected in the final product (A1, AC3, D2).

Several reasons exist for the adaptation of the roadmap. The product owner might consider given topics as inappropriate due to changes in the environment. Additionally, adaptations can be necessary if features are more complex than expected (AC3, B1, B3, D1), new dependencies to other teams or departments occur (AC3), new strategic projects arise (B1, C1, D1, D3), or other topics are reprioritized (B1, BC2, D1, D2). Moreover, short-dated requirements, such as marketing or exhibition efforts, may emerge, necessitating quick implementations (AC3). Finally, certain team-related reasons (e.g., absences due to sickness, shared resources, or vacations) may require roadmaps’ adaptation (B1, D1, D3).

Disadvantages can occur if changes to the product roadmap are granted. First, if plans are not finalized, uncertainty is inevitable (AC3, BC2, D2). Additionally, if a team has already worked on topics that are adapted or replaced, resources are wasted (AC3, B3), team motivation decreases (D2, B3), and stress (B3) emerges.

4.1.3. Product backlog. All four teams use a product backlog, covering topics ranging from high to low priority, as well as bug reports. However, every team handles the product backlog in a unique way.

Team A splits its backlog into two parts. One part is stored in an issue-tracking tool with a Scrum extension called Jira, while the other is kept in an Excel file. The Jira part is visible to all stakeholders having access to this platform. The product backlog is not prioritized. Team B stores the entire product backlog in Jira. It is visible to all stakeholders having access to Jira and is prioritized. Team C stores the entire product backlog in Jira. As above, it is visible to all stakeholders having access to Jira, but is not prioritized. However, not all backlog items are transparent, since some are just notes and thus incomplete. Similarly, Team D stores the entire product backlog in Jira. It is visible to all stakeholders having access to Jira, but not prioritized.
One reason for using a product backlog is the availability of a central repository for issues that have to be addressed in the future (AC3, B3, D1, D2). Moreover, it offers a good basis for planning further product versions (A1) and thus supports the sprint planning (B3, D3). In addition, it supports the planning of smaller work packages (AC3, B1).

However, some teams explain that their product backlog is too extensive to be completely processed in the given time (B3, D1). Additionally, product backlogs are not synchronized with the product roadmap. As a result, the road map does not reflect the current status of all recognized issues. Consequently, work time is sometimes wasted for backlog maintenance (A1). Finally, in many cases, product backlogs are used as notepads. In such cases, only the backlog authors can fully understand what is meant by the entries, which leads to reduced transparency (C1, D3).

Advocates argue for the backlog items’ prioritization as a means of increasing transparency for employees within and outside the team (BC2, D1). This also makes the sprint planning more efficient, as the team picks the highest prioritized backlog items, rather than having to trawl through the entire backlog (BC2, AC3).

Conversely, time constraints are the main reason for not prioritizing backlogs (A1, D1, D3). Additionally, backlogs are quite extensive, leading to effort-intensive prioritization processes (B1, D1, D3). Prioritization sometimes raises controversies, as it mainly depends on the evaluation of the product owner (AC3, B3, D1, D2).

The fact that all stakeholders (not just the team members) have access to the backlogs increases transparency inside and outside the team. This transparency offers an overview of the upcoming topics for the team (D1, D2) and empowers external employees, as they can monitor the status of their feedback (AC3, B1, D3), which makes communication and stakeholder management more efficient (A1, D1). It also supports acceptance and valuation of the teams’ work in other departments within the company (BC2).

A problem, on the other hand, is keeping the product backlog in ‘good shape’ (BC2). This is necessary for ensuring that stakeholders understand the items’ meaning and increases transparency (B1). Additionally, fears exist concerning issues discussed outside the team (A1), which results in a challenge to handle stakeholder feedback in an appropriate way (C1).

4.1.4. Definition of ‘done’. In contrast to the recommendations made in the extant literature (cf. section 2), none of the four teams applies a common definition of ‘done’. Each team has its own, individual ‘done’ criteria on a backlog item level.

Although interviewees admit the importance of clearly knowing what has to be done to complete a backlog item (BC2, D3), many also agree that it is not necessary to have a common definition of ‘done’ that covers needs of all teams (B1, D1, D2). In their view, acceptance criteria on a backlog item level are sufficient (BC2, C1). Moreover, the knowledge of team members helps to make the acceptance criteria concrete in cases when these are not properly prescribed (AC3, BC2). This, in turn, has the advantage that the acceptance of an issue can be managed on an individual level (C1). Finally, developing a common definition of ‘done’ simply requires too much effort (AC3, BC2, D2).

4.1.5. Sprint backlog. A sprint backlog is used in all four teams and changes are made cautiously. Team A only makes changes to the sprint backlog if additional requirements arise due to dependencies on other teams. Team B only adapts the backlog if estimates made at the beginning of the sprint are inaccurate. However, it explicitly never enhances the sprint backlog. In case of Team C, the backlog is only adapted if requirements were not clear at the beginning of a sprint, or the complexity of specific topics was underestimated. Finally, Team D changes the backlog if ad-hoc requirements occur, bugs are reported, or in order to make inaccurate planning more specific.

The advantages and thus the reason for using a sprint backlog are clear to the interviewees. Elaborating, estimating, and assigning the backlog items to the sprint backlog encompasses assignments of tasks to team members and external resources, gaining their commitment, and scheduling due dates (A1, C1, D1, D3). The prioritization yields knowledge about what has to be done and in which order (D3). Additionally, it is possible to control the progress during the sprint, since all items are updated/tracked in the sprint backlog (C1, D1), leading to transparency (A1, AC3). Finally, using a sprint backlog enables using electronic tools (like Jira) to manage the backlog and enhance efficiency (AC3).

Conversely, there are almost no disadvantages, with the exception of over-planning (i.e., the planning of too many backlog items for a sprint) that can lead to overheads (D3) decreasing team members’ motivation. Among the recognized advantages of adopting the sprint backlog, most notable are higher flexibility (D1, D2, D3) and greater stakeholder satisfaction, if additional tasks can be completed in a short time (D3).

However, some disadvantages of adopting a sprint backlog are also noted by the interviewees. For example, enhancing the backlog can lead to time pressure for team. It can also reduce output quality as, for instance, insufficient testing capacities are available (AC3, D3).

It can also cause a reduction in the overall development speed, as introducing new topics requires time and resources (D3). Finally, discomfort inside the team increases, resulting in decreased motivation (AC3).
4.2. Events

4.2.1. Sprint planning meeting. This principle is applied by all teams at the beginning of each sprint, that is, every two weeks. The meeting usually takes about one to two hours and, while all team members are involved, other stakeholders are not.

The main reasons for conducting sprint-planning meetings are related to planning (B1, AC3, D3), in particular, determining responsibilities (D1, C1) and estimating effort (C1, D1, AC3) by creating the sprint backlog. Additionally, the meeting fosters common commitment (B1) to the tasks for the next sprint. The team aligns itself (D3) to the goals that must be achieved. Accordingly, and due to the fact that team members take tasks on a voluntary basis, a kind of team spirit emerges, which leads to increased team motivation (B1). Other advantages mentioned are planning certainty (D1), goal-orientation (A1), and ensuring that each team member is working to capacity (BC2).

However, such meetings sometimes take longer than necessary (B1) since, in most teams, effort estimation for all tasks is conducted solely during this meeting.

According to the interviewees, all team members participate in each sprint-planning meeting, primarily because this increases motivation (D2, D3, BC2, AC3). Each team member has the feeling that he or she can decide upon the direction the team will be working in the next sprint. Sprint planning meetings also foster transparency, since each team member is aware of the upcoming tasks (AC3, A1, BC2). Thus, communication efforts are reduced (D2). Furthermore, the meetings help new team members to be quickly integrated into the team (D1). Finally, everyone has specific qualities that he or she can contribute to the team in order to reach the best possible planning outcome (B3, D2).

These advantages notwithstanding, several pitfalls were mentioned to occur in daily work. These mainly concern the time efforts (B3, Ac3, D2, C3), which are considered comparatively high, especially for team members who do not work full-time in the team (A1, AC3, D3). As it is not necessary for each team member to be familiar with every task, the meeting is not universally important to everyone (BC2, D1).

4.2.2. Daily Scrum. The daily meeting is arranged by one of the four teams only. However, the other three teams also conduct regular meetings, but only two or three times a week. In each case, the meeting takes place at the same time and place and lasts about 15 minutes.

Due to the limited time, topics requiring more extensive discussion are postponed to other meetings.

In general, our interviewees consider this regular meeting positive. For example, many smaller issues can be discussed in a quick manner (D2, AC3, A1) and uncertainties can be disclosed (C1, BC2). Thus, problems do not have a strong impact on the team’s work (AC3). Daily Scrums help the team to check the status of backlog items on a daily basis (D1) and thus keeps the team focused (B1, C1). Moreover, they serve as a reminder (D1) and foster the transparency of the work progress (A1) as well as communication and interaction (BC2) within the team. Daily Scrums are a source of new ideas (B1) and lead to better team spirit (B1, C1).

Among the disadvantages the interviewees noted, time constraints lying outside the teams’ influence are the main reason for not conducting these meetings on a daily basis. Three of the four teams do not have the time for conducting the daily meetings (BC2, D2, AC3, D1), especially in cases when some team members are assigned to more than one team at the same time (BC2). Thus, the reasons for not conducting Daily Scrums mainly stem from the organization. However, work progress might not be sufficient to require additional updates (D2, B3, B1).

Daily Scrums take place at the same time and place in all four teams to avoid confusions caused by organizational matters (BC2, A1) and to minimize coordination efforts (D2, C1, AC3).

A disadvantage, on the other hand, is that team members from other departments require higher effort for coming to the meeting rooms in the basement (BC2).

If topics cannot be clarified in Daily Scrums, only the members that can contribute to the topics discussed are invited to additional meetings. This prevents wasting resources (A1, AC3, D2, D1, D3) and preserves motivation for the Daily Scrum (D2, BC1). No disadvantages related to Daily Scrums were mentioned.

4.2.3. Sprint review meeting. Such meetings are conducted on regular basis by two teams (in one team, this activity is combined with the sprint planning meeting). Another team has conducted it twice since the implementation of Scrum. The remaining team has not performed such a meeting yet but is currently thinking of its implementation. Customer participation in such meeting is generally rare. Arguments for conducting a sprint review meeting are that it creates awareness for conducted work (AC3, B1, D1, D2) and results in increased motivation of the team members (D1, D2). Furthermore, feedback is collected (B3) and areas for product improvements are identified (D1, D2, D3). Although not strictly a meeting’s core purpose, status and performance of the sprint work can be assessed (D3). As a result, problems (C1) and lessons learned (A1, C1) during the development process are exposed, which is particularly important if a team does not conduct a sprint retrospective meeting (see 4.2.4).

Despite its advantages, the majority of the interviewees cite the time dedicated to the sprint review meeting as an issue (A1, AC3, D1, D2), which stems from external time pressures. This applies to both the time required to
hold the meeting itself and the time spent preparing the presentation of the results (AC3). Despite potential benefits, such as reduced communication effort (A1, AC3, D2) and direct customer feedback (A1, AC3, D2), customers are typically not invited to the review meetings. Our interviewees see increased time effort due to a greater number of participants and too many discussions on a content level (A1, AC3, B2) as factors that threaten the success of such meetings.

4.2.4. Sprint retrospective meeting. One team conducts sprint retrospective meetings on a regular basis. Another team has conducted it twice since the implementation of Scrum. The remaining two teams have not performed such a meeting yet. As a positive experience and thus as an argument for conducting a sprint retrospective meeting, members of Team A and B mention that its main advantages are the ability to discuss and clarify problems within the collaboration process. Respective solutions can be directly implemented in the next sprint (AC3, BC2). Moreover, members of Team B reported no fear of contact (B2). Discussion takes place in an organized way (D2) and improvements are proposed and agreed upon by the team itself (B1, B2, D2). However, the main reason for omitting this meeting is extensive effort (A1, C1, D2, D3) related to the time pressure dictated from outside the team. Instead, informal talks take place, although experiences have shown that these are not as efficient as structured meetings (D1). Team A and B also consider combining sprint retrospective meetings with the sprint review meetings to save time (A1, B3).

4.3. Roles

4.3.1. Scrum team. Since Scrum cannot be applied without a Scrum team, all four teams have a core Scrum team, which develops the product features (cf. section 2.3.1). As already described in section 2.3.1, the four Scrum teams have on average eight members (Team A: eight, Team B: eight, Team C: seven, Team D: nine).

In comparison to common project teams, Scrum teams in product management are advantageous, since they focus on one dedicated product and thus possess higher expertise and knowledge with regard to this product (A1, AC3, B1, BC2, C1, D1, D2, D3). The regular and constant cooperation in a stable team offers opportunities for enhancing product knowledge and ensure a steep learning curve, which increases effectiveness and efficiency (A1, BC2, D2, D3). Teams can work on products on an unlimited time horizon (B1, BC2, D1, D2). The small teams also facilitate quicker development of features (AC3, B1, C1, D3) and can improve individual team dynamics (BC2). Additionally, less overhead is required for planning and internal marketing compared to normal project organization (C1).

However, products must allow to be developed in encapsulated iterations. In one of TrustMeCorp’s teams, the implementation of such iterations was challenging, since a more holistic approach was necessary in some cases (D1). Finally, when introducing scrum teams, the company has to be aware of the need for a set-up phase, which is necessary to become familiar with the product and Scrum itself (BC2).

Concerning advantages or disadvantages of spatial proximity of the team, the majority of the interviewees argued for sharing office space with other team members, instead of having only functional representatives in close proximity (A1, AC3, B1, C1, D1, D2, D3). One of the advantages of such an arrangement is faster interaction with other team members (A1, AC3, B1, D1, D2). Some interviewees state that this is especially important for developers, as they need to be able to exchange technical information (C1, BC2). Moreover, it fosters the social contact with other team members (D1) and thus has an important team-building function.

Disadvantages, or rather challenges, pertain to the need to create a seating plan that merges team members but also offers connections to other teams and functions (C1). This may not always be possible in the office space available.

4.3.2. Product owner. Each Scrum team in our case company has a product owner. The four product owners perform their role in a manner similar to that described in the pertinent literature. However, there are minor adaptations. The product owner of Team B is also involved in implementation and testing. Therefore, the Scrum master supports the product owner when it comes to the maintenance of the product backlog. The product owner of Team D is practically in charge of two products, but does not maintain the product backlog and the backlog items. Additionally, the product owner of Team C acts in line with the published recommendations, but fulfills two roles at the same time, since he is also the chief product owner (cf. section 2.3.3). Except for these adaptations, all interviewed product owners closely collaborate with the Scrum masters of their teams and try to integrate the team members into this process.

The reason for the close collaboration is the better outcome of the work process (B1). The chief product owner adds: “Both have to be a well-functioning team who get along with each other” (C1, Team C chief product owner). Our interviewees did not mention any disadvantages concerning this collaboration.

While a close collaboration of the product owner with other team members is seen as important for the team, it can also lead to too many distractions (D1).

When it comes to decision-making, advantages of involving the entire team include the opportunity to ask
The team can rely on the statements of both roles (D2).
company emphasize that sufficient progress needs to be made for a useful discussion. While our results thus corroborate the criticality of regular meetings in which the entire product team takes part in order to discuss a project’s current state, the team context also shows that such meetings do not have to take place on a daily basis.

6. Conclusion

With our study of how Scrum principles are applied in practice, we answer the call for empirical research concerning adoption of agile methods. While showing that customizing the core principles is perceived as beneficial by practitioners, we argue that Scrum principles are contingent on the context they are applied to. By discussing the reasons for adopting or adapting the principles, we contribute to a deeper understanding of agile methods in practice and help to open the black box of development approaches in information systems research. While our study can be seen as a starting point for quantitative evaluations of contingency theory in the context of agile development, the in-depth insights of Scrum principles at TrustMeCorp provide practitioners a useful reference for adapting agile methods to the context of their companies.

7. References


