

# The Impact of Covid 19 on Agile Software Development: A Systematic Literature Review

Michael Neumann  
Hochschule Hannover  
[michael.neumann@hs-hannover.de](mailto:michael.neumann@hs-hannover.de)

Yevgen Bogdanov  
Hochschule Hannover  
[yevgen.bogdanov@hs-hannover.de](mailto:yevgen.bogdanov@hs-hannover.de)

## Abstract

*In 2020, the world changed due to the Covid 19 pandemic. Containment measures to reduce the spread of the virus were planned and implemented by many countries and companies. Worldwide, companies sent their employees to work from home. This change has led to significant challenges in teams that were co-located before the pandemic. Agile software development teams were affected by this switch, as agile methods focus on communication and collaboration. Research results have already been published on the challenges of switching to remote work and the effects on agile software development teams. This article presents a systematic literature review. We identified 12 relevant papers for our studies and analyzed them on detail. The results provide an overview how agile software development teams reacted to the switch to remote work, e.g., which agile practices they adapted. We also gained insights on the changes of the performance of agile software development teams and social effects on agile software development teams during the pandemic.*

## 1. Introduction

Today, agile methods are established approaches in software development and beyond [1]. Widely used agile methods such as Scrum [2] and XP [3] are focusing on value-based work and social aspects. For a consistent understanding of values and principles, the agile manifesto was published 20 years ago [4]. It is known that social aspects like collaboration and communication are of importance in agile software development [5, 6]. The iterative approach and recurring practices such as daily stand-up or planning meetings are characteristics of agile methods [7]. They were designed to allow companies and software development teams to react rapidly to new circumstances and adapt their approach to upcoming challenges. Usually, these new circumstances occur due to changing market conditions and changes concerning requirements.

The Covid 19 pandemic led to several changes in our working life since its outbreak in early 2020. Many countries and companies have taken measures to reduce the spread of the virus [8]. These measures included, in particular, contact restrictions in the first 18 months of the pandemic. For example, companies all over the world have sent their employees to remote work [9]. The switch to remote work has led to various new challenges, as it was common in many companies before the Covid 19 pandemic for the teams to work co-located in the office. Agile software development teams are affected by this change, as social aspects such as communication or collaboration are of great importance [10].

The relevance of social aspects in software development and in particular agile software development has been investigated in the past (e.g., [6, 11]). For instance, we know, that the quality of collaboration and communication with stakeholders, like users and customers is important concerning to the output of the development teams [12]. Another facet is the quality of team work in agile software development. Hennel and Rosenkranz investigated the effects of social debt and dealt in particular with the impact of psychology safety on team work in agile software development teams. They analyzed the effects of social focused agile practices and psychology safety on the performance of agile software development teams. The authors found a correlation between the relationship of psychology safety and positive effects of the use of social focused agile practices, which impacts the performance of the teams.

Furthermore, we know that a distributed type of work affects agile software development for example in terms of communication and collaboration due to cultural, language, or time zone differences (e.g., [13]). The effects of remote work on the social aspects and how teams react to such a change of work organization has not yet been extensively investigated in the area of agile software development before the Covid 19 outbreak in 2020. However, the first studies have already been

published, which are dealing with the effects of the pandemic-induced ad-hoc switch to remote work in the area of agile software development (e.g., [14, 15]) and beyond (e.g., [16, 17]). We want to investigate how agile software development teams deal with the upcoming challenges concerning their methods and practices in use. Furthermore we want to analyze what effects on social aspects are described in the literature. This is important to get an understanding of what upcoming challenges and barriers might be of relevance for agile software development as there is a correlation between social aspects and the productivity.

This paper presents the actual state of research and analyzes how agile software development teams have reacted to the challenges, which came up due to the switch to remote work. Also, we want to examine whether the productivity or performance of the agile software development teams changed during the switch to remote work and whether there are effects on social aspects of the teams. Thus, this paper aims to answer the following research questions:

- RQ1: How did agile software development teams adapt their approach due to the switch to remote work?
- RQ2: How did the performance or productivity of agile software development teams changes during the Covid 19 pandemic?
- RQ3: Are there any findings of social effects on agile software development teams?

The paper at hand is structured as follows: First, we describe the background of the study in Section 2 and give a brief overview of agile methods and the importance of social aspects. In the following, we explain the selected research approach for conducting this SLR in Section 3. The results of our study are presented in Section 4. We discuss our findings and provide practical implications in Section 5. Before the paper closes with a conclusion in Section 7, we discuss the limitations of our study in Section 6.

## 2. Background

### 2.1. Agile Software Development

The idea of iterative approaches for the implementation of software development projects goes back to the 1950s [18]. Today well-known agile approaches such as Scrum or XP were created in the USA in the 1990s [19]. The motivation for this lay in the increased dynamics of project environment factors. For instance, Cohen et al. [20] argue that customers

were often unable to define their needs at the beginning of a project, constantly changing requirements during the project period, and the increasing business and technology development in the IT industry. Thus, agile methods are often interpreted as reactions to plan-based approaches [21]. We know, that agile methods are iteratively structured and aim for fast response times during the project period [22]. Further, Abrahamsson et al. [23] characterize agile methods as incremental, cooperative and adaptable.

The components of agile methods such as artifacts, roles or practices, and the rules for specific application in practice are usually described in guidelines (e.g., the Scrum Guide [2]). These guidelines specify how particular practices should be implemented to be able to achieve the goals described above, such as a quick reaction to changing requirements. Due to the high degree of social aspects due to regularly recurring practices, values and principles are assigned great importance in applying agile methods. The agile manifesto defines four pairs of values and 12 principles that serve as the basis for working together in agile software development teams [4]. Specific values are also often described in the guidelines for agile methods (e.g., [2, 3]).

Another aspect of agile methods is the integrated aspect of continuous optimization of the approach in use. This Kaizen<sup>1</sup> approach offers agile software development teams the opportunity to inspect and adapt their own work regularly. This results in a wide variety of agile methods in practice [24].

### 2.2. Distributed vs. Remote Work

Different types of work organization are described in the literature. The main difference is made between onsite and offsite work. Concerning the topic of this paper, we focus on offsite work organization and first discuss the different types, in particular distributed and remote work.

Global distributed software development is understood to mean a team at different locations across national borders or time zones [25]. The (globally) distributed work has been intensively examined in recent years, especially in the area of global software engineering/development (e.g., [26, 27, 28]). In particular, cultural differences [29] or language barriers [30] were identified as factors for the quality of communication and collaboration.

In contrast to distributed work, remote work (also named as telework) refers to different types of work

---

<sup>1</sup>Wang et al. defines Kaizen as "...continuous improvement to establish a smoother flow".

outside the office. A distinction is usually made between working from home [31] and working from anywhere [32]. In this paper we use the term of remote work, as it summarizes several variants of work outside the office as mentioned above [33].

The distributed work across national borders presents agile software development teams with significant challenges concerning agile practices, such as daily stand-up meetings and coordination in general, due to language or time zone differences [13]. Even if these challenges are weakened in remote work, for example, because the team members speak the same language, we emphasize several challenges, e.g., concerning the team works quality, collaboration and communication. These challenges are significant if the teams were not used to this type of work organization before the Covid 19 pandemic. Furthermore, we point to different effects, which can primarily occur in remote work. For instance, work interruptions by family members or challenges concerning the separation of work and family time and space.

### 3. Research Methodology

To ascertain the current state of research in order to be able to answer our research questions described in Section 1, we decided to perform a systematic literature review (SLR). We use the guidelines according to Kitchenham and Charters [34] to prepare, conduct and report our SLR.

Kitchenham and Charters recommend creating a SLR protocol that should promote the traceability of the selected research method. The following subsections describe the information based on the research protocol and are used to explain our approach.

#### 3.1. Search strategy

In order to perform a systematic search in digital databases, it is necessary to define a search string. This search string is intended to limit the result set of literature but not to exclude potentially relevant studies. Therefore, we created topic related keyword categories based on our research questions:

<Agile Software Development> AND <Covid 19 pandemic>

Specific search terms are assigned to the categories. If there are several search terms in a category, we have linked them with Boolean operators. We have refined the individual search terms of these two categories in iterative searches and used a combined search string from these two categories for our search. Thus, the following search string was used:

*(("agile software development" OR "agile method\*") AND (("covid 19" OR "sars cov 2") AND ("pandemic" )))*

For the literature search, we used Google Scholar. We argue this choice with the overlap of the results with other search engines and digital libraries in other areas with an academic focus (e.g., ACM or ScienceDirect) [35]. Another aspect was the short time period since the Covid 19 pandemic impacted agile software development teams. Therefore, we assumed that we could identify more potentially relevant literature via Google Scholar. We performed iterative test runs at Scopus and ACM. These tests confirmed our decision, as these searches showed lower numbers of results.

We used the advanced search functionalities and the full text search at Google Scholar and defined the date range filter as "since 2020", because the Covid 19 pandemic arised in the first quarter of 2020. We performed the final search run on the 30th of August 2021. The result set contains 859 potentially relevant studies.

#### 3.2. Study selection

According to Kitchenham and Charters [34] it is important to define inclusion and exclusion criteria for selecting the most relevant studies in a systematic manner. Following this recommendation we defined four inclusion and five exclusion criteria, which are described in the following Table:

These criteria aim to identify the relevant studies to answer the research questions of this SLR. As shown in Table 1, we defined structural (e.g., IC1, IC2 or EC1, EC2, EC3) and content-related criteria (e.g., IC4 and EC4).

Before we started verifying the content, we checked the primary studies based on the structural criteria IC3, EC1, EC2 and EC3. In total, we excluded 427 studies based on the structural exclusion and inclusion criteria (see Table 2). The majority of the studies were excluded due to exclusion criteria 3 (no peer review). Based on the content-related criteria, we performed a four stage study selection process.

**First step:** Checking title and keyword of the studies.

**Second step:** Verifying the abstract based on the content-related criteria IC4 and EC4. Studies, which can not be ruled out based on these criteria are defined as borderlines and checked in the next step.

**Third step:** Reading the introduction and conclusion and checking the studies based on the content-related criteria. As in the second step, borderline cases are moved for a final verification in the fourth step.

Category	Criteria
Inclusion	<p><b>IC1:</b> Studies published between January 2020 and June 2021</p> <p><b>IC2:</b> Studies published in the field of agile software development</p> <p><b>IC3:</b> Studies written in English</p> <p><b>IC4:</b> Studies dealing with the impact of the switch to remote work during the Covid 19 pandemic on agile software development</p>
Exclusion	<p><b>EC1:</b> Gray literature (e.g., technical or experience reports)</p> <p><b>EC2:</b> Contributions with less than three pages</p> <p><b>EC3:</b> Study is not peer-reviewed (such as theses or books)</p> <p><b>EC4:</b> Studies dealing with practical software development and related topics (such as education) without a connection on agile software development and the Covid 19 pandemic</p>

**Table 1. Study selection criteria**

Criteria	No. of excl. studies
IC3 (non english)	55
EC1 (gray literature)	92
EC2 (less 3 pages)	0
EC3 (no peer review)	280

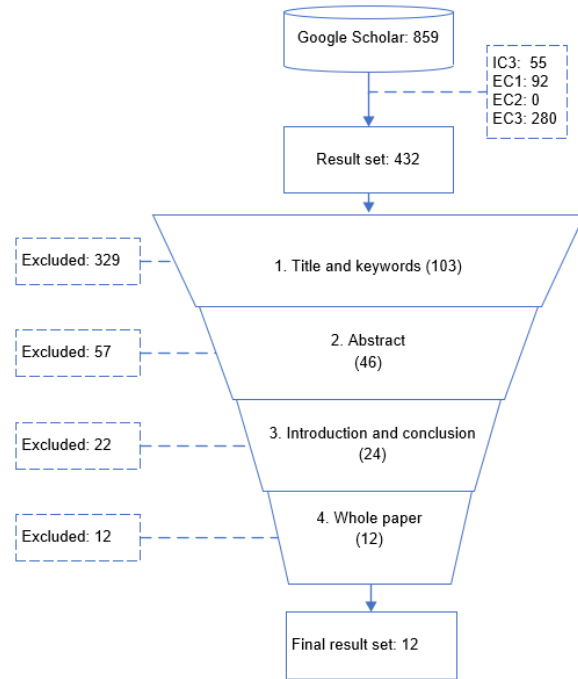
**Table 2. Overview of excluded papers based on structural criteria**

**Fourth step:** Reading and verifying the whole content of the study in terms of relevance.

The result set for the content-related study selection based on IC4 and EC5 contains 432 studies. While checking title and keyword(s) of the studies, we excluded 329 studies due to the non-content relation to our SLR. In the next step we excluded further 57 studies due to verifying the abstract and 22 studies based on the verification of the introduction and conclusion. We finally excluded 12 studies in the final verification step, reading the whole content. Thus, the final result set for the data extraction of our SLR consists 12 studies (see Figure 1).

### 3.3. Data extraction

After the identification of the relevant primary studies, we executed the data extraction with two tools: Citavi and Microsoft Excel. In a first step, we added the primary studies to our Citavi project using the import



**Figure 1. Results of the study selection process**

functionality. The Citavi importer extracts the structural data (e.g., author(s), title, year) automatically from the source and adds them to the Citavi project. We also added the abstracts of the primary studies manually to the entries in Citavi.

In the second step, we exported the Citavi project to a Microsoft Excel file and added further columns. The added columns were used to extract more detailed data like the research approach used in the respective primary study, the findings related to our research questions and the validation procedure used by the authors.

## 4. Results

### 4.1. Overview of the results

Before we discuss the results of the SLR and answer the research questions in the following subsections, we give a structural overview of the studies.

Nine primary studies were published in 2021, three in 2020. In addition, various research methods were used for the novel phenomenon under study. We present the relation of the included studies and the used research methods in Table 3. The authors of five studies used a quantitative research approach. In four studies, single, multiple, or longitude case studies with a qualitative approach were used. Da Camara et al. have followed an action research approach, and Poth et al. used design science research. Nolan et al. conducted a multivocal

Research approach	Reference(s)
Action research	[36]
Design science research	[37]
Multivocal literature review	[38]
Qualitative (multiple) case study	[39, 40, 41, 42]
Quantitative survey	[43, 44, 45, 46, 47]

**Table 3. Overview of the used research designs**

literature review.

Seven primary studies were published in conference proceedings ([38, 39, 40, 42, 43, 44], the remaining six were published in journals [36, 37, 41, 45, 46, 47].

#### 4.2. Adaption of agile practices due to the switch to work from home

Based on the upcoming discussion in this subsection we will answer our first research question: *How did agile software development teams adapt their approach due to the switch to remote work?*

The results from eight primary studies show, that the agile software development teams rapidly adapted the agile practices and artifacts in use. The adaptation of the agile methods used by the agile software development teams is due in particular to the virtualization of work and the importance of knowledge sharing and communication [38, 37]. Agile software development Teams implement the virtualization of the work with the help of tools. Three studies show that the majority of agile software development teams use Microsoft Teams for virtual communication and collaboration [40, 44, 47]. Also, Slack, Zoom, and Google Hangouts are in use. Marek et al. [44] also report in their study that the majority of agile software development teams use Jira for their work organization.

Neumann et al. [40] show that some agile software development teams worked with physical artifacts such as Kanban boards before the pandemic. Due to the switch to remote work the teams now use tools like workflow management applications and virtual whiteboards to virtualize their artifacts. Poth et al. [37] introduce their Self-Service Kit (SSK) to activate teams to optimize their knowledge sharing skills and activities. The authors deal also with the effects of the switch to remote work and emphasize how the SSK helps agile software development teams to support their expertise in digitization.

The adaptation of agile practices due to the virtualization mainly affects the methodical implementation of the respective practice. In remote

work, for example, the estimates in planning meetings are carried out with the help of chats in collaboration tools or integrated plug-ins in the respective workflow management applications [41, 44]. Although methods such as planning poker are still in use, Neumann et al. [40] point out that the specific implementation has lost its playful character and is now more objective.

Even retrospectives are now mainly carried out using tools. Da Camara et al. [36] and Neumann et al. [40] describe that the teams use both virtual whiteboards and implement new methods like storytelling.

Smite et al. [42] describe in their study the effects of remote work on the agile practice of pair programming. The authors note that the use of pair programming has decreased since switching to remote work. They justify this, among other things, with the more significant effort and faster fatigue of those involved. For some agile software development teams, these findings are also described by Neumann et al. [40].

The study results show that the agile software development teams rapidly adapted their approach to the new situation. Although the adaptations primarily relate to the virtualization of the practices as well as their methodical implementation and agile artifacts, it should be noted that the recurring character of the adaptations is described in the studies by Neumann et al. [40] and Marek et al. [44]. The adjustments are not unique at the methodological level when implementing and using agile practices. Rather, the teams are constantly optimizing their approach and using, for example, new functions in the applications such as Microsoft Teams or virtual whiteboards [40].

#### 4.3. Changes on productivity and performance during Covid 19

The discussion in this subsection leads to the answer of our second research question: *How did the performance or productivity of agile software development teams changes during the Covid 19 pandemic?*

Concerning the answer to RQ2, we need to differentiate between performance (i.e., perceived efficiency) and actual productivity in the form of output (e.g., delivered features, bug fixes, or product increments).

Three studies show that the performance of agile software development teams has not permanently decreased due to the switch to remote work [40, 41, 44]. According to Neumann et al. [40] and O Connor et al. [41] a short-term drop in performance for some agile software development teams occurred immediately after switching to remote work, which

lasted only a few weeks. The authors of the studies [40, 41, 44] describe several reasons for the consistently stable or increased performance of the agile software development teams. In addition to the correlation between performance and well-being (see the following subsection), the increased transparency of the agile approach used is mentioned in particular [40]. The virtualization of various agile artifacts and practices (see subsection above) provides optimized transparency. A high level of transparency about the artifacts and communication as well as collaboration, e.g., in agile practices such as planning or retrospective meetings, is important for agile software development teams to continuously optimize their own approach. This optimized transparency is the basis to analyze their approach in a meaningful and targeted manner and to take appropriate measures for optimization (see inspect and adapt in Scrum [2]). Another relevant aspect for stable performance is the improved involvement of stakeholders and product owners (see the previous subsection). As described above, this enables faster and more targeted coordination.

In addition to the performance, other authors found a (slightly) decreased productivity of agile software development teams [38, 43, 45, 47]. Butt et al. [43] attribute this to the increased stress caused by working in the home office. You justify this, for example, with the conflict-ridden combination of work and private life, especially in stressful and pressure-prone situations in project business. The authors also point to the more inferior quality of communication in the team. This also applies to the cooperation with the customer if they have low availability or the Internet connection is of poor quality. Ralph et al. [45] correlate productivity with team members' well-being and, according to their study, productivity decreased slightly. They note that the well-being is also influenced by aspects outside the working life of the team members and that this must be taken into account in particular under pandemic conditions. The conditions in the home office of the individual persons must also be taken into account [38, 45].

#### **4.4. Effects on the social facets of agile software development teams by switching to remote work**

The following discussion is the basis for answering our third research question: *Are there any findings of social effects on agile software development teams?*

The switch to remote work has resulted in several changes in the daily collaboration of agile software development teams. We found five papers, in which the

authors describe a changed character of communication through the distributed form of remote work and the virtual type of communication and collaboration [38, 39, 40, 44, 47]. For instance, the communication between the team members while conducting agile practices is more objective [40]. Dreesen et al. point out several lacks regarding aspects of communication, like depth and latency [39]. These effects also arise in teams whose employers have introduced a camera always on rule during meetings. Objective communication, in turn, has an impact on the meeting culture [38] and in a decreased informal communication [39]. Neumann et al. [40] describe the meeting culture as more effective and straightforward. Communication also affects bilateral teamwork, for instance, concerning dedicated agile practices such as pair programming [42]. Another challenge described by Neumann et al. [40] and Nolan et al. [38] is the on-boarding process of new team members due to the switch to remote work. This challenge concerns both the socialization of the new employees to the agile software development team, as well as the familiarization with the methods and practices used. Also, Neumann et al. [40] describe a positive aspect of virtual communication, affecting interface roles such as Product Owners. Asynchronous communication results in an optimized integration of these roles and improved and faster coordination, e.g., regarding questions regarding requirements.

According to four studies, the more objective virtual communication is also at the expense of the social exchange between the agile software development team members [36, 38, 40, 41, 44]. While breaks were spent together in the co-located work or small talk was held before or after meetings, the social exchange almost came to a standstill, especially at the beginning of the switch to remote work [38]. Several agile software development teams try to counter these negative effects with socializing events. Here, Neumann et al. describe a high level of creativity [40]. The agile software development teams set up, for example, virtual team rooms to spend team breakfasts, team time-outs, or coffee breaks together. Game events or virtual walks are also organized. Similar to the adaptation of agile practices described in the subsection above, the socializing events are also regularly adapted, or new practices are tried out.

Another effect identified by two studies is related to the well-being of agile software development team members. Butt et al. [43] point out a decreased well-being of agile software development team members and argue this with mental health stress due to the Covid 19 pandemic. In contrast, Russo et al. describe [46] an increased well-being. They argue this

with the increased autonomy of the team members. The authors also point out the correlation with productivity (see subsection above).

Furthermore, Schmidtner et al. [47] describe a clear vision of the agile software development teams concerning the future form of the work. The majority of respondents stated that they expected an increase in remote work. Along with this, the importance of tools for virtual collaboration and communication will increase.

## 5. Discussion and practical implications

Based on the findings of our literature review, we have identified four main aspects that we will discuss below with the aim to provide practical implications. We believe that the presented findings contribute to the agile community and researchers as well as practitioners can derive valuable contribution.

**First:** Agile software development teams are able to react quickly to new situations and challenges by adapting and optimizing their approach to the new circumstances [38, 40, 44]. This ability is not related to specific contexts, like companies or project goals. The research results show various effects on agile artifacts and practices due to the virtualization. The agile software development teams use several tools for virtualization. In addition to communication applications (MS Teams is the most one used), the teams also use collaboration tools such as virtual whiteboards (like Miro). Particularly in the case of agile practices, the methodical implementation, e.g., the estimation of requirements, changed. Agile practices such as the retrospective or the review are used in combination with methods such as storytelling to ensure the quality of these practices and the willingness to collaborate in the teams. Also, we found in the literature effects on intensely collaborative practices such as pair programming [42]. These practices are no longer carried out to the same extent as in the co-located working method before the pandemic. Due to the firmly anchored optimization of requirements in agile methods, we assume that further adaptations to agile practices will be carried out in the future, especially at the methodological level. We also expect that new agile practices will be established, which promote the virtual collaboration of the distributed remote work of agile software development teams.

**Second:** We need more focus on social aspects both, on the team level concerning communication, collaboration and the individual level of the team members, e.g. concerning their well being and stress level. We argue this especially with the correlation

to performance and productivity presented in the literature (e.g., [46, 48]). Several influencing factors are described in the literature we analyzed. The virtualization of collaboration described above leads to more objective communication (e.g., [48]). As a result, effects on the discussion culture in the agile software development teams can be observed. These are described as more effective and goal oriented [40]. The meetings and other agile practices such as pair programming are perceived as more strenuous. This change in communication is at the expense of social exchange in the teams, which came to an almost complete standstill, especially at the beginning of the pandemic and remote work. In comparison to co-located work, there are hardly any shared lunches or coffee breaks. The teams react to this by introducing and adapting socializing events. Examples of this are virtual walks, team breakfasts, or even game evenings in the studies. In addition to collaboration in teams, these effects also affect the on boarding of new employees. This makes it particularly difficult for new team members to familiarize themselves with the agile software development team and to socialize. We point to the high relevance of social aspects as impact factors on the success of agile software development teams. For both, practitioners as well as researchers the facets of social aspects should become more importance. We emphasize this relevance detached from specific areas, such as software engineering or the context (e.g., specific companies, business fields or projects).

**Third:** We assumed, that the productivity and performance should be decreased in agile software development teams due to their need of high quality team work and social aspects, like communication and collaboration (as described above). As we analyzed the findings from the literature we found different results in the studies concerning the effects on the performance and productivity of agile software development teams. However, after a short drop in the agile software development teams examined in the studies, performance is at a stable level and comparable or even better before the pandemic and the work in remote work [40, 44]. Neumann et al. emphasize the increased transparency of the agile approach, e.g., artifacts and communication/collaboration in agile practices [40]. This allows the agile software development teams to analyze their work and carry out targeted optimization measures. The increased transparency is due, in particular, to the virtualization of the collaboration in the agile software development teams described above. We point out the vital relation to the importance of transparency in empirical based or oriented agile methods (such as Scrum or XP).

This aspect should have an essential meaning for practitioners in the future, detached from the pandemic and its effects.

In three papers, we found a deterioration in productivity in their studies [43, 45, 47]. For instance, Butt et al. and Ralph et al. argue the decreased productivity with effects concerning the well-being of the team members and the prevailing correlation to productivity in software development teams [43, 45]. Furthermore, Butt et al. point out that the stress has increased due to the closer integration of work and private life [43]. Likewise, the cooperation with customers due to lower availability for coordination and the infrastructure (especially the internet connection quality) is given as a reason by Butt et al. [43]. However, Schmidtner et al. also describe that productivity has only decreased slightly compared to the pandemic [47].

The productivity and performance of agile teams depend on many factors and the individual situation of the teams (e.g., effects due to the maturity of the agile method in use). We, therefore, recommend analyzing the specific situation in the respective agile software development teams. Based on the results of these analyzes, specific optimization measures can then, if necessary, be carried out, for example, concerning the integration of stakeholders.

**Fourth:** We expect, that the discussion on what exactly the new normal of work organization should look like will be getting more and more attention and relevance in the future. We argue our expectation with the findings in the literature concerning the question how and where the team members want to work in the future. Further we took part on several discussions with practitioners and also noticed discussions in the international media landscape related to the question if the employees need to come back to work in the office or not. The remote work comes with several positive perceived effects. For instance, the agile software development team members perceive the higher degree of autonomy as positive. Schmidtner et al. describe that employees can no longer imagine switching to pure co-located work [47]. Instead, the agile software development teams want a hybrid operation in which they work co-located in the office a few days a week (e.g., for the implementation of planning, review, or retrospective meetings) and work remotely for the remaining days. We attach importance to this aspect, as we assume that the future behavior of companies will have an impact on the well-being of employees. In addition, stabilization leads to numerous strategic and organizational questions, for example, concerning the physical conditions in the offices. We assume that collaboration work spaces such as open work spaces will

be more relevant in the future. In contrast to this, at least in agile software development, the question arises as to whether individual workstations must be provided for each employee.

## 6. Limitations

A major challenge in systematic literature reviews is ensuring the completeness of the result set. We decided to use Google Scholar to identify the largest possible result set of potentially relevant primary studies. Different publishers are taken into account here, and the search results show a high level of agreement with those of digital libraries such as ScienceDirect (see Section 3). We took the Google Scholar guidelines into account when creating the search terms and performing the search. We also conducted iterative search runs and continuously optimized the search terms.

The selection of the literature followed a systematic procedure based on defined inclusion and exclusion criteria. These criteria covered both structural and content-related aspects. Therefore the limitation comes up that one author mainly carried out the study by himself. A systematic support of a second author was conducted at the revision stage of the submission procedure. To reduce the risk of bias, other researchers from a research group partially checked the results of the particular steps of the SLR process.

## 7. Conclusion and Future Work

The Covid 19 pandemic had many influences on the working world since its outbreak in 2020. Due to the permanent switch to remote work, we observed several effects on social aspects of work in agile software development. When using agile methods, communication and collaboration in the team are of great importance for success.

We conducted a systematic literature search to ascertain the current state of research and map the findings on the consequences of permanent switching to remote work in agile software development teams. We integrated 12 studies in our result set and analyzed them on detail related to our three research questions. The review provides several implications for both, the research community and practitioners.

The current state of research raises several new questions. As shown above, the findings of various studies are contrary to one another. This affects the productivity and performance of agile software development teams, which relates to the great diversity of the current situation in different countries and companies. Correlations between performance and



well-being as well as productivity and social facets are presented in the literature. The social aspect, in particular, makes further research necessary. We assume many unknown facets and influencing factors on agile software development due to the changes how agile software development teams operate, e.g. offsite (remote or distributed) and onsite or a mix of both.

We will use the results of this SLR as a basis for an international quantitative study. Besides the identified literature, we aim to use our findings for discussions with the research community and practitioners. We will incorporate the feedback from these discussions into our planned survey. The survey will focus on agile software development team members' expectations for the future of their work. The study aims to be how agile software development teams imagine their work organization (co-located, remote or mixed) in the future and what sustainable effects they expect on their agile approach.

## References

- [1] VersionOne and Collabnet, "14th annual state of agile survey report," 2020.
- [2] K. Schwaber and J. Sutherland, "The scrum guide," 2020.
- [3] K. Beck, *Extreme programming explained: Embrace change*. Boston: Addison-Wesley, 5. print ed., 2000.
- [4] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Greening, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas, "Agile manifesto," 2001.
- [5] S. McCarthy, P. O'Raghallaigh, C. Fitzgerald, and F. Adam, "Building bridges, burning bridges: The use of boundary objects in agile distributed isd teams," in *Proceedings of the 53rd Hawaii International Conference on System Sciences*, 2020.
- [6] E. Whitworth and R. Biddle, "The social nature of agile teams," in *Agile 2007* (J. Eckstein, ed.), (Los Alamitos, Calif.), pp. 26–36, IEEE Computer Soc, 2007.
- [7] L. Williams, "Agile software development methodologies and practices," in *Advances in computers* (M. v. Zelkowitz, ed.), vol. 80 of *Advances in Computers*, pp. 1–44, London: Academic Press, 2010.
- [8] A. Ghose, B. Li, M. M Y, C. Sun, N. Z. Foutz, and J. Anton, "Trading privacy for social good: Did america unite during covid-19?," in *Proceedings of the 41st International Conference on Information Systems (Virtual ICIS 2020)*, (Hyderabad, India), 2020.
- [9] M. Mandviwalla, L. Dignan, D. Desai, C. Kearns, L. Descano, and R. Sankaran, "The industry perspective on covid-19," in *Proceedings of the 41st International Conference on Information Systems (Virtual ICIS 2020)*, (Hyderabad, India), 2020.
- [10] T. Mens, M. Cataldo, and D. Damian, "The social developer: The future of software development [guest editors' introduction]," *IEEE Software*, vol. 36, no. 1, pp. 11–14, 2019.
- [11] S. Matook and K. Kautz, "Mindfulness and agile software development," in *Proceedings of the 19th Australasian Conference on Information Systems*, 2008.
- [12] L. M. Maruping and S. Matook, "The multiplex nature of the customer representative role in agile information systems development," *MIS Quarterly*, vol. 44, no. 3, pp. 1411–1437, 2020.
- [13] S. Jalali and C. Wohlin, "Global software engineering and agile practices: a systematic review," *Journal of Software: Evolution and Process*, vol. 24, no. 6, pp. 643–659, 2012.
- [14] L. Griffin, "Implementing lean principles in scrum to adapt to remote work in a covid-19 impacted software team," in *Proceedings of the 5th International Conference on Lean and Agile Software Development*, vol. 408 of *Lecture Notes in Business Information Processing*, pp. 177–184, Springer, 2021.
- [15] D. Mancl and S. D. Fraser, "Covid-19's influence on the future of agile," in *Agile Processes in Software Engineering and Extreme Programming – Workshops* (M. Paasivaara and P. Kruchten, eds.), vol. 396 of *Lecture Notes in Business Information Processing*, pp. 309–316, Cham: Springer International Publishing, 2020.
- [16] M. Janssen and H. van der Voort, "Agile and adaptive governance in crisis response: Lessons from the covid-19 pandemic," *International journal of information management*, vol. 55, p. 102180, 2020.
- [17] R. Doyle and K. Conboy, "The role of is in the covid-19 pandemic: A liquid-modern perspective," *International Journal of Information Management*, vol. 55, no. 2, 2020.
- [18] C. Larman and V. R. Basili, "Iterative and incremental developments. a brief history," *Computer*, vol. 36, no. 6, pp. 47–56, 2003.
- [19] J. A. Highsmith, *Agile software development ecosystems*. The Agile software development series, Boston, Mass.: Addison-Wesley, 2002.
- [20] D. Cohen, M. Lindvall, and P. Costa, "An introduction to agile methods," in *Advances in Computers* (M. Zelkowitz, ed.), vol. 62 of *Advances in Computers*, pp. 1–66, s.l.: Elsevier textbooks, 2004.
- [21] T. Dingsøy, T. Dybå, and N. B. Moe, eds., *Agile software development: Current research and future directions*. Berlin: Springer, 2010.
- [22] H. K. Flora and S. v. Chande, "A systematic study on agile software development methodologies and practices," *International Journal of Computer Science and Information Technologies*, no. 5, pp. 3627–3637, 2014.
- [23] P. Abrahamsson, O. Salo, J. Ronkainen, and J. Warsta, "Agile software development methods: Review and analysis," no. 478, pp. 7–94, 2002.
- [24] M. Kuhrmann, E. Hanser, C. R. Prause, P. Diebold, J. Münch, P. Tell, V. Garousi, M. Felderer, K. Trektore, F. McCaffery, and O. Linssen, "Hybrid software and system development in practice: waterfall, scrum, and beyond," in *Proceedings of the 2017 International Conference on Software and System Process* (R. Bendraou, D. Raffo, H. LiGuo, and F. M. Maggi, eds.), (New York, NY), pp. 30–39, ACM, 2017.
- [25] M.-A. Vanzin, M. B. Ribeiro, R. Prikladnicki, I. Ceccato, and D. Antunes, "Global software processes definition in a distributed environment," in *29th Annual IEEE/NASA Software Engineering Workshop, 2005*, (Los Alamitos, Calif. [u.a.]), pp. 57–65, IEEE Computer Society, 2005.

- [26] A. Altaf, U. Fatima, W. H. Butt, M. W. Anwar, and M. Hamdani, "A systematic literature review on factors impacting agile adaptation in global software development," in *ICCCM 2019*, ICPS, (New York, New York), pp. 158–163, The Association for Computing Machinery, 2019.
- [27] T. Dingsoyr and D. Smite, "Managing knowledge in global software development projects," *IT Professional*, vol. 16, no. 1, pp. 22–29, 2014.
- [28] A. Lamersdorf and J. Münch, "Studying the impact of global software development characteristics on project goals: A causal model," *Open Software Engineering*, vol. 4, no. 2, pp. 2–13, 2010.
- [29] H. Holmstrom, E. Conchuir, P. Agerfalk, and B. Fitzgerald, "Global software development challenges: A case study on temporal, geographical and socio-cultural distance," in *Proceedings, 2006 IEEE International Conference on Global Software Engineering*, (Los Alamitos, Calif.), pp. 3–11, IEEE Computer Society, 2006.
- [30] C. Ebert and P. de Neve, "Surviving global software development," *IEEE Software*, vol. 18, no. 2, pp. 62–69, 2001.
- [31] C. Sullivan, "What's in a name? definitions and conceptualisations of teleworking and homeworking," *New Technology, Work and Employment*, vol. 18, no. 3, pp. 158–165, 2003.
- [32] C. F. Prithwiraj Choudhury and B. Larson, "Forthcoming work-from-anywhere: The productivity effects of geographic flexibility," *Strategic Management Journal*, vol. 42, no. 4, pp. 655–683, 2020.
- [33] M. Das, J. Tang, K. E. Ringland, and A. M. Piper, "Towards accessible remote work: Understanding work-from-home practices of neurodivergent professionals," *Proc. ACM Hum.-Comput. Interact.*, vol. 5, Apr. 2021.
- [34] B. Kitchenham and Charters S., "Guidelines for performing systematic literature reviews in software engineering," 2007.
- [35] A. Yasin, R. Fatima, L. Wen, W. Afzal, M. Azhar, and R. Torkar, "On using grey literature and google scholar in systematic literature reviews in software engineering," *IEEE Access*, vol. 8, pp. 36226–36243, 2020.
- [36] R. Da Camara, M. Marinho, S. Sampaio, and S. Cadete, "How do agile software startups deal with uncertainties by covid-19 pandemic?," *International Journal of Software Engineering & Applications*, vol. 11, no. 4, pp. 15–34, 2020.
- [37] A. Poth, M. Kottke, and A. Riel, "The implementation of a digital service approach to fostering team autonomy, distant collaboration, and knowledge scaling in large enterprises," *Human Systems Management*, vol. 39, no. 4, pp. 573–588, 2020.
- [38] A. Nolan, R. White, M. Soomro, B. C. Dopamu, M. Yilmaz, D. Solan, and P. Clarke, "To work from home (wfh) or not to work from home? lessons learned by software engineers during the covid-19 pandemic," in *Systems, Software and Services Process Improvement* (M. Yilmaz, P. Clarke, R. Messnarz, and M. Reiner, eds.), (Cham), pp. 14–33, Springer International Publishing, 2021.
- [39] T. Dreesen, P. Hennel, C. Rosenkranz, and T. Kude, "The second vice is lying, the first is running into debt. antecedents and mitigating practices of social debt: an exploratory study in distributed software development teams," in *Proceedings of the 54th Hawaii International Conference on System Sciences*, 2021.
- [40] M. Neumann, Y. Bogdanov, M. Lier, and L. Baumann, "The sars-cov-2 pandemic and agile methodologies in software development: A multiple case study in germany," in *Proceedings of the 5th International Conference on Lean and Agile Software Development*, vol. 408 of *Lecture Notes in Business Information Processing*, pp. 40–58, Springer, 2021.
- [41] M. O Connor, K. Conboy, and D. Dennehy, "Covid-19 affected remote workers: a temporal analysis of information system development during the pandemic," *Journal of Decision Systems*, 2021.
- [42] D. Smite, M. Mikalsen, N. Moe, S. V., and K. E., "From collaboration to solitude and back: Remote pair programming during covid-19," in *International Conference on Agile Processes in Software Engineering and Extreme Programming. (XP)*, vol. 419 of *Lecture Notes in Business Information Processing*, pp. 3–18, Springer, 2021.
- [43] S. A. Butt, S. Misra, M. W. Anjum, and S. A. Hassan, "Agile project development issues during covid-19," in *Proceedings of the 5th International Conference on Lean and Agile Software Development*, vol. 408 of *Lecture Notes in Business Information Processing*, pp. 59–70, Springer, 2021.
- [44] K. Marek, E. Winska, and W. Dabrowski, "The state of agile software development teams during the covid-19 pandemic," in *Proceedings of the 5th International Conference on Lean and Agile Software Development*, vol. 408 of *Lecture Notes in Business Information Processing*, pp. 24–39, Springer, 2021.
- [45] P. Ralph, S. Baltés, G. Adisaputri, R. Torkar, V. Kovalenko, M. Kalinowski, N. Novielli, S. Yoo, X. Devroey, X. Tan, M. Zhou, B. Turhan, R. Hoda, H. Hata, G. Robles, A. Milani Fard, and R. Alkadhhi, "Pandemic programming: How covid-19 affects software developers and how their organizations can help," *Empirical Software Engineering*, pp. 1–35, 2020.
- [46] D. Russo, P. Hanel, S. Altnickel, and N. van Berkel, "Predictors of well-being and productivity among software professionals during the covid-19 pandemic: A longitudinal study," *Empirical Software Engineering*, vol. 26, no. 62, 2021.
- [47] M. Schmidtner, C. Doering, and H. Timinger, "Agile working during covid-19 pandemic," *IEEE Engineering Management Review*, 2021.
- [48] P. Hennel and C. Rosenkranz, "Investigating the "socio" in socio- technical development: The case for psychological safety in agile information systems development," *Project Management Journal*, vol. 52, no. 1, pp. 11–30, 2021.