

Lessons Learned on Communication Channels and Practices in Agile Software Development

Muhammad Ovais Ahmad¹, Valentina Lenarduzzi², Markku Oivo¹ and Davide Taibi²

¹ M3S Research Unit, University of Oulu, Oulu, Finland
{ovais.ahmad; markku.oivo}@oulu.fi

² Tampere University of Technology, Tampere, Finland
{valentina.lenarduzzi; davide.taibi}@tut.fi

Abstract—Communication plays an important role in Agile Software Development (ASD). In each ASD practice (e.g., stand-up or retrospective meetings), different communication practices and channels are adopted by different companies. Several works have analyzed the impact of communication channels and practices. However, there are no secondary studies summarizing their impact on ASD. This study presents a Systematic Mapping Study (SMS) that aggregates, summarizes, and discusses the results of 25 relevant primary studies concerning the impact of communication channels and practices in ASD. We followed the well-known systematic mapping methodology in software engineering and analyzed empirical studies published before the end of June 2018. The results of our study have yielded several strategies that can be adopted by practitioners. Communication practices are context dependent. In the case of a distributed team, blended usage of rich-media communication tools, such as shared mind-map tools, videoconferencing, and promoting the exchange of team members between teams, is beneficial. In conclusion, communication can be expensive if teams do not apply the right strategies. Future research direction is to understand how to maximize product quality while reducing communication cost and how to identify the most beneficial communication strategy for the different stages of ASD.

Keywords—communication practices, communication process, agile software development, scrum, extreme programming

I. INTRODUCTION

Constant communication and sharing information about a project's development among the whole team is essential. Agile methods preaching the empowered and self-organizing development teams where focus is on constant communication and information sharing [18]. One of the most important factors for project success is continuous and active communication with the customer and the team members [16], effective communication among developers, operations, support, customers, management, and business areas [19]. During the development process, communication plays an important role in terms of coordination among the different teams involved, in order to manage dependencies between the actors in the process [6] [7] [8]. Communication can be considered as a mediating factor that influences both coordination and control activities during the development process [9].

In software development, it is of crucial importance to have effective communication starting from the project beginning, from the definition of the Minimum Viable Product [5]. Poor team communication is often leads to failure for engineering projects. Such poor communication become more complex in

distributed software development. When the teams are unable to find good communication strategies or channels, it affects the quality of the product. However, communication is not always beneficial and can even decrease productivity [17].

Communication in Agile teams can be formal or informal. Formal communication includes specification documents and review meetings [10], while informal communication takes place via conversations among the teams within a company [10] and is usually based on telephone or video calls, audio or video conferences, email, and face-to-face meetings [11]. As suggested by Henttonen and Kirsimarja [11], face-to-face meetings can increase trust among the team members and, consequently, the quality of the development process and the final product.

ASD requires constant communication and information sharing within the team and between the team and the customers [20], [21]. Communication in ASD can be further classified as internal and external. In internal communication, developers and project leaders are the main actors involved in the process, while the development team and the stakeholders are the ones mainly involved in external communication [11]. Regarding the type of communication used, we can distinguish between active and passive. Active communication is mainly based on physical and synchronous approaches, such as face-to-face meetings, while passive communication is based on asynchronous approaches, such as email [20].

Various studies highlighted the problem of identifying the most effective communication channel [16][18][19]. Moreover, besides selecting the appropriate channel, practitioners still face the issue of selecting the most appropriate communication processes and practices [19][22].

In recent years, research has focused on the communication aspects in the Agile development process [12],[13],[14],[15], investigating key success factors [17],[18],[19] and the most effective communication channels that lead to positive effects on the process [22]. To the best of our knowledge, no secondary studies comparing the advantages and disadvantages of communication channels in the context of ASD exist to date.

Therefore, the aim of this study is to provide a systematic mapping review on the benefits and issues of the different communication practices and channels adopted in ASD. We also identify best practices and lessons learned in order to help teams working with Agile development processes in order to communicate more effectively.

Other reviews investigated communication practices in ASD from different point of views. Rizvi et al. [27] investigated distributed agile software engineering for years 2007–2012. Their main goal was to investigate reasons for adopting agile methods to GSD as well as risks. Alzoubi et al. [28] investigated the communication challenges in distributed teams that adopt agile, classifying challenges in six categories, but their focus is different to ours because agile practices are not taken into account. Hossain et al. [29] conducted an SLR on Scrum and GSD, but in contrast to our SLR their focus is limited to Scrum. Vallon et al. [30] performed a SLR on the application of agile practices in GSD. Finally, Hoda et al. [31] performed a tertiary study providing an overview of the SLRs on ASD research topics.

This paper is structured as follows. Section 2 presents the systematic mapping review protocol. In Section 3 we report the obtained results. In Section 4, we describe the threats to validity, and in Section 5, we draw conclusions and present an outlook on future work.

II. RESEARCH METHODOLOGY

The section outlines the adopted systematic mapping study (SMS) process, which follows the established guidelines and procedures proposed by Petersen et al. [2]. The motivation to conduct a SMS is to focus on the “*classification and thematic analysis of literature on a software engineering topic*” [23] [24]. SMS guidelines involves the following tasks: (1) defining the research questions; (2) outlining the search strategy; (3) extracting and analyzing the data.

A. Goal and Research Questions (RQs)

The goal of this study is to investigate the role of communication in ASD, focusing on channels and practices used during the communication process. We aim at identifying best practices and lessons learned in order to help ASD teams to communicate more effectively. To achieve goal of this study we addressed the following Research Questions (RQs):

(RQ1) What is currently known about communication in ASD?

(RQ2) Which communication channels have been studied in ASD?

(RQ3) What are the best communication practices commonly adopted in ASD?

A. Search Strategy

The search strategy adopted in this SMS is depicted in Figure 1. We first identified the bibliographic sources, then we defined the inclusion and exclusion criteria and selected the search keywords. Based on these, we carried out the selection process and finally extracted the data from the selected papers.

Bibliographic Sources Identification. We selected the list of relevant bibliographic sources suggested by Kitchenham [4]: ACM Digital Library, IEEE-Xplore, Scopus ScienceDirect, Citeseer Library, Inspec, Springer. The selected databases are pertinent to this study as they are adopted by most of the literature reviews.

Search Keywords Definition. We defined search keywords based on the PICO structure [4]. We extracted the keywords

from Population and Intervention terms. As suggested by Kitchenham [4], the Outcome and Comparison terms cannot always be considered in software engineering if the research focuses on general investigation.

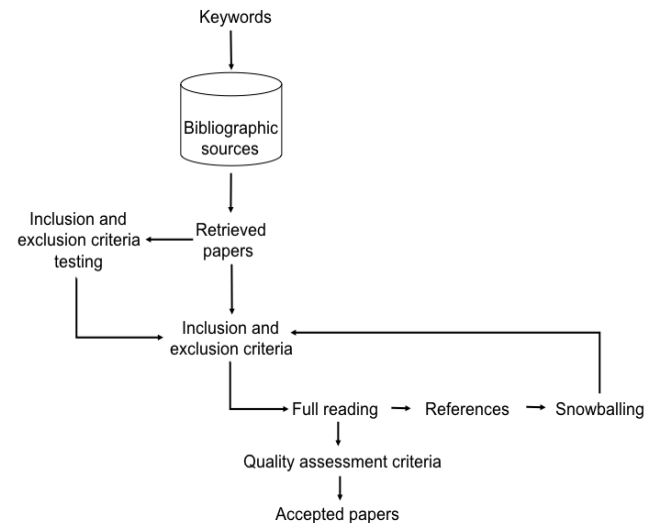


Fig. 1. Search Strategy process

Table 1. Search keywords

Population	Intervention
P: Agile development	I: Communication
P terms: Agile, Scrum, Extreme Programming	I terms: Communication

Based on the identified search keywords, we derived the following query: (agile OR scrum OR "extreme programming") AND communication)

Inclusion Criteria Definition. Papers were eligible for inclusion in the SMS if they presented data related to communication in ASD. The inclusion criteria used were:

- Papers reporting communication practices in Agile;
- Papers reporting lessons learned, advantages, or disadvantages regarding the usage of communication channels;
- Papers reporting the influence of communication on the process;
- Study written in English and published before 07/2018;
- If published in more than one journal/conference, the most recent version of the study.

Exclusion criteria. Papers not fulfilling any of the inclusion criteria were left out, according to the following criteria:

- Papers not written in English;
- Duplicate articles;
- Not peer-reviewed scientific papers (books or book chapters, presentations, prefaces, gray literature, etc.);
- Simulation studies (e.g., mathematical modeling);
- Papers adopting the term "Agile" for purposes other than ASD (e.g., Agile manufacturing)
- Short papers, workshop papers, and work plans (papers not reporting results).

Search and Selection Process. The application of the search keywords in the selected bibliographic sources returned 2042 unique papers. Next, we applied the inclusion and exclusion criteria to the retrieved papers, regarding both title and abstract. As suggested by Kitchenham [4], we tested the applicability of the inclusion and exclusion criteria as follows: 90 papers randomly selected out of the 2059 were used as a sample; The three authors applied the inclusion and exclusion criteria to 60 papers each (each paper was reviewed by two authors); There was disagreement on 9 of the 90 selected papers (10%). For these 9 papers, the third author provided his/her opinion.

We applied the inclusion and exclusion criteria, for both title and abstract, to the remaining 1969 papers, and we included in this step 600 papers. Then, after the full reading process we selected only 82 papers.

In order to retrieve the most relevant papers, we integrated the procedure, taking into account also forward and backward systematic snowballing [3] on the 74 remaining papers. Regarding backward snowballing, we considered all the references in the papers retrieved, while for forward snowballing, we evaluated all the papers referencing the retrieved ones. We added another two papers and thus ended up with a new set of 84 papers as primary studies.

Assessing the Suitability of the Papers. In this last step, we checked whether the quality of the selected papers was sufficient to provide the required information needed to support the goal of our study. We considered all 76 papers obtained from the search and selection process. Each paper had to provide us with the following information:

- **Communication processes.** The paper had to report the communication process adopted during various development process phases and the research method adopted.
- **Communication channels.** The paper had to report the channels used for communication among the team, the stakeholders, and each partner involved in the process.
- **Best Practices or lessons learned** on the usage of communication channels in ASD.

The final dataset was reduced to 25 papers for the review, as reported in Table 2. In Appendix A, we present the list of the selected Primary Studies (PS).

Table 2. Search keywords

Step	# Papers
Retrieval from bibliographic sources	2059
Inclusion and exclusion criteria (title)	-1266
Inclusion and exclusion criteria (abstract)	-193
Full reading	-518
Snowballing process	+2
Assessing the papers' suitability	-59
Primary Studies (PS)	25

Data Extraction and analysis. Once the primary studies are selected, two of the authors analyzed and extracted the data, and the third one verified the correctness of the extraction. This practice helps to avoid researcher bias which is validity threat. The primary studies were analyzed based on study properties:

- **Context Data:** Study type of the paper; paper's goal(s); purpose of the communication; type of teams (distributed teams, collocated, distributed different times, different countries); communication stakeholders.
- **Process Data:** Communication definition; communication frequency; communication channels; communication tools; reported communication challenges for each communication channel.
- **Outcome:** Future direction from each paper; strength of evidence for each communication channel; Best Practices.

All primary studies have been analyzed separately by each author and then a combined peer-review has been conducted. In cases of disagreement, the third author was requested for his input. Finally, the author four ensure consistency in the analysis and consolidation of the results.

III. RESULTS

This section presents the results from the analysis of the 17 primary studies, which is based on the research goal previously mentioned in Section II. The results represent summary of results regarding the communication channels adopted in the primary papers, and then classify the lessons learned. The results identified the usage of four synchronous and two asynchronous communication channels as shown in Table 3.

The PS highlighted communication channels that are the main medium of contacting and exchanging information in ASD teams. In synchronous communication channels, face-to-face communication is the most frequently adopted channel, including both formal and informal communication. Phone calls are used mainly by project managers, while other roles commonly rely on video-conferencing, chat, or email [P1] [P2] [P16]. In case of asynchronous communication, ASD teams often use email, even in the case of co-located teams, while documentation is rarely adopted.

The PS reported that in Global Software Development (GSD), temporal distance challenges play an important role. Coordination between teams increases in complexity with increasing difference between time zones. Face-to-face communication is seldom employed in GSD, except in rare cases such as kickoff meetings. However, unexpectedly, even in GSD, pair programming (via videoconference) can be easily applied and turns out to be highly beneficial [P2].

For instance, Holmström et al. [P2], exemplified that developers with up to eight hours-time differences can work efficiently in pairs, supported by video-conferencing, but they need to shift their working hours so that they can have at least six overlapping hours per day. There is not silver bullet for communication channel, each channel has its own purpose. However, the PS recommended to experiment and blend various tools based on the team requirements and projects purpose to solve information sharing issues in ASD.

3.1 General Communication Channels Benefits in ASD

The PS reported that Agile teams use a variety of channels for communication to ensure open and multidirectional interaction.

Table 3. Classification and description of the communication channels adopted in the studies

Channel	Description	Primary studies (PS)
Synchronous		
Face to face	Can be either formal or informal. Includes communication during Agile ceremonies, pair programming, or any other face-to-face communication related to the project under development.	P1, P2, P3, P6, P8, P9, P10, P12, P13, P16, P17, P18, P19, P21, P22, P23, P24, P25
Video-conference	Used as medium for high-level discussions between non-co-located teams or between teams and customers.	P1, P7, P10, P11, P23, P24
Telephone/Audio conference	Includes phone calls and any other audio conference tools adopted, such as Skype conferences (only audio) or phone calls.	P1, P2, P4, P5, P7, P10, P11, P14, P16, P19, P19, P21
Online chat	Instant Messaging applications such as Skype or Facebook chat. Mainly used as a means for exchanging technical information between co-located teams (e.g., passing information such as short configuration files) or for quick and informal communication between distributed teams.	P2, P7, P8, P5, P15, P16 P22
Asynchronous		
Documentation	Documentation is another form of communication that occurs between developers who need to report a set of written information. In the two PS, Wiki was the only tool reported and was used to communicate with other teams during project implementation or to keep track of technical choices.	P1, P10, P19, P20, P21, P24, P25
Email	Email is used for different purposes in ASD. Can be used both formally and informally.	P1, P2, P4, P5, P7, P8, P10, P16

Table 4 exhibits benefits, presented in the PS. It is important to identify tools for communication in early phases of a project, based on context and needs. Customers play a critical role in terms of identifying the communication tools to be used in a project. Such early identification is beneficial for Agile teams to achieve optimal performance and strengthen the relationship with the customer.

In all PS it is discussed that face-to-face communication yields a lot of positive outcomes compared to the use of other communication channels. For example, in case of requirement gathering is preferred way of communication [P5]. The face to face communication helps to reduce the capability of conveying ambiguous information [P1]. However, when face-to-face communication is not possible, online communication tools can be used efficiently. Video conferences, supported by rich media such as mind-mapping tools or desktop sharing [P15], improve the quality of the communication, while voice calls (Skype or telephone) are not as effective and should only be used for unofficial meetings. Chat is deemed to be more effective and useful for daily, informal information exchange or asking question from an expert about software functionality [P5].

Email was recognized as more formal way of communication and is more effective in case of getting approval on documents or requirements from the customer (where an email message constitutes a sort of contract) [P5]. On the other hand, the PS highlighted it as concern regarding effectiveness in the use of email for person-to-person communication or formal approval of documents [P5]. However, blended usage of different tools for different purposes can solve most

information-sharing issues [P5]. Further, continuous Integration tools useful and helps to facilitate and to communicate the project status from development to final delivery.

Communication effectiveness decreases paired with the level of interaction provided by the communication channel.

3.2 Communication Benefits and Challenges in Agile Practices ASD includes several practices prescribed by the different Agile approaches, such as Scrum, Extreme Programming (XP), and others. Some practices are shared by different approaches while others are not. In Table 5, we report the list of Agile practices along with their communication related benefits from PS.

Many software companies practicing pair programming. One reason is that pair programming helps to improve individual commitment and efficient way to implement code review [P2]. However, it is very challenging to use them on a daily basis as they are time and resource consuming [P6].

All PS highlighted that scrum meeting (e.g. daily standup, retrospectives) yields various benefits such as help to keep track of the project status, increase communication, enhance collaboration, reduce temporal distances and culture barriers. It is very important to note that the meeting which are conducted in front of project board are appreciated. One reason is that visibility and open discussions in the team/organization helped the spread the information and solve issues quickly. For example, in daily standup meeting, the visual board with different color cards is useful for keeping track of different types of stories [P12].

Table 4. Findings regarding communication benefits and issues.

Practices	Findings and PS
Open Communication	<ul style="list-style-type: none"> - Open communication must be encouraged and assured in order to get the benefits of Scrum [P11]. - Direct peer-to-peer communication between developers must be enabled to achieve successful results [P4]. - Frequent communication may be a symptom of a good and trusting relationship [P9]. - Team members communicate more with those they are aware of or with those they know can help [P8]. - Collocated teams over-communicate overheard problems [P12]. - A multicultural environment stimulates and increases productivity and creativity [P5].
Face-to-Face	<ul style="list-style-type: none"> - Preferred communication for collecting requirements [P5]. - Informal face-to-face communication should be encouraged to increase knowledge transfer [P14].
Videoconference	<ul style="list-style-type: none"> - Distributed teams should be equipped with video-conferencing instead of only using audio-conferencing or telephone [P7][P14][P18][P19][P21]. - Should be accessible to the entire team for regular meetings [P11].
Telephone, Chat	<ul style="list-style-type: none"> - Useful for unofficial meetings [GSD P11]
Email	<ul style="list-style-type: none"> - When face to face is not feasible, use email to increase the chance of response and encourage prompt, useful, and conclusive responses [GSD P4]. - Asynchronous communication due to temporal distance impacts coordination mechanisms [GSD P7]. - Effective to approve customer requirements when email message constitutes a sort of contract [P5].
Tool-supported Communication	<ul style="list-style-type: none"> - Use globally available project management tools to record and monitor project status on daily basis [P3] - Blended usage of different tools for different purposes can solve most information-sharing issues [P5]. - Mind-mapping tools increase communication effectiveness and help mediate issues between distributed teams [P15]. - Continuous Integration tools are <ul style="list-style-type: none"> - Useful to communicate the current status of the project to testers, thus making system testing activities easier [P6]; - Facilitate testing. Help quality engineers get information on the status of the end product.

There is misconception that documentation is not important in ASD. The PS pointed out the importance of documentation, it helps both current and future team to work more efficiently and understand the logic easily. Code documentation is an important channel when there is needs to modify the code, helps traceability and test validation [P17]. In general, Agile teams rely on ad hoc communication and dynamic patterns of knowledge sharing [P8].

3.2.1 Team-related Communication Practices

Table 7 summarizes the findings of the PS on team-related communication practices. Leaders should be aware of a variety of culturally sensitive behavior and values. At the same time team members should respect the leader’s views and should not be underestimated. A mutual trust is essential and it is duty of management to build such trust between developers and their first level manager [P5][P8].

It is also very important to exchange members in teams which are working on same or similar projects. The PS highlighted that a visiting engineer or outside expert is highly beneficial to support inexperienced teams during the first iteration [P10]. It is important for both co-located and distributed teams. Frequent exchange visits of team members are beneficial at the beginning of the project or in critical phases to get in touch with other members and learn how to work together [P11].

The primary studies stress that open communication should be encouraged among software development team members. This is helpful in various ways. For example, it improves team interaction and fosters good understanding between project team and management; in multicultural environments, it stimulates and increases productivity and creativity.

To increase interaction between teams, it is good to exchange team members in distributed project or interdependent teams. This helps them to interact more closely and fosters interpersonal relationships within teams. The use of emergent members helps to spread/share knowledge. Further, pair programming helps to increases mutual understanding and collaboration within and between teams [P2] as well as reduces social and cultural distances [P2]. However, it is difficult and problematic practice for daily use [P6].

Customer communication and close collaboration is crucial for development team and project success. During requirement elicitation customer absences bring challenges for challenge and it more difficult to perform remotely [P5]. The situation become more complex in distributed teams, where customer requirements are presented by other teams. It is also argued that upfront fixed requirements should be less ambiguous than deliberately vague agile requirements.

Table 5. Communication Benefits and Challenges in Agile practices.

Practice	Findings
Pair Programming	+ Increases time overlap and reduces temporal distance [P2] + Efficient way to implement code reviews [P6] + Increases mutual understanding and collaboration within and between teams [P2] + Reduces social and cultural distances [P2] - Difficult and problematic practice for daily use [P6]
Scrum/Sprint Planning Meetings	+ Set the scene for iterations involving negotiations with the customer [P12] + Reduce geographical distances[P2][P23] + Help to keep track of the project status [P6] + Increase awareness of the next iteration in the whole project team [P6][P23] + Provide close interaction among distributed project stakeholders [P7] + Help to minimize misunderstandings and misinterpretations regarding project standards [P2] [P19] [P21] [P23] + Increase mutual understanding and collaboration within and between teams [P2] + Help to reduce the confusion about what to developed from both the developers' and customer perspectives[P6]
Reflection Retrospective Reviews	+ Provide an efficient way to deploy and improve Agile practices [P6] + Good to use for assessing teamwork in completed sprint [P7] + Help to understand project standards among distributed project stakeholders [P7] + Increase project visibility and transparency [P7] + Help project managers with more efficient project supervision [P7]
Daily Standup Meetings	+ Increase awareness of project status among developers, product leaders, and customers [P6] + Help to quickly respond to changes in the project [P14] + Reduce coordination breakdown caused by temporal and geographical distance [P7] + Reduce cultural issues (e.g., perception of authority/hierarchy, frames of reference) [P7] + Convey strategy to the stakeholders [P7] + Increase knowledge sharing, thereby fostering a collaborative approach to problem solving [P12]
Story/Task Board	+Provides project status information to all stakeholders [P6] [P12]
Story Cards	+ Different color cards are useful for keeping track of different kinds of stories [P12]
Test-driven Development	+ Helps to maintain a shared standard view [P7] + Improves understanding of the functionalities required from the customer perspective [P7]
Refactoring	+ Improves communication, simplifying it and adding flexibility [P7]
Documentation	- Code documentation is an important communication channel when the customer needs to modify the code [P17] [P19] [P21] [P23] [P24] [P25]. - Documenting decisions is important in order to communicate them to future team members [P17]. - Test documentation helps to communicate information about traceability and test validation [P17].
Note: + (plus sign) indicates benefits; – (minus sign) indicates a challenge.	

Korkala et al [P1] exemplify that due to some reason project manager and customer group did not help developers and architects to analyze the requirements as well as deliberately hiding information. The companies should carefully plan their practices and recommended to follow people- vs. process-oriented control strategy [P1]. Further, some tools and practices can improve collaboration with the customer even in the case of geographical distance. In any case, the customer's role must be defined upfront and the customer should be enabled to make conclusive decisions regarding the project's functionality and scope [P3].

3.2.2 Organizational Responsibilities

The primary studies reported that in order to utilize the optimal capacity and skills of Agile teams, it is essential to

provide proper method, process, and tool training (see Table 6).

Management should provide support along with a combination of internal and external coaching [P10] [P13]. Management should also provide access to everything that is necessary for the team's work, so that dependencies can be avoided easily. In the case of distributed development teams, the manager needs to understand the languages in which the various stakeholders communicate and needs to be sensitive to culture differences. The management need to make developers aware that they should be careful about other cultures. For instance, nobody felt the need to point out any cultural factor that would be disturbing or (even more surprising) stimulating [P5].

The introduction of new practices should be clearly communicated to the whole team paying attention to adapt

the process to the team needs, instead of forcing the team to adopt to the process as prescribed. The management needs to avoid practices which are process- vs. people-oriented control. “*Weak customer relationship and organizational politics that restrict information sharing may cause any communication medium to become inefficient*”. The management should focus on creating an efficient customer relationship and environment that enables effective communication [P1]. Afterwards, management can focus on communication channels and tools.

Another interesting point highlighted by primary studies is an open office environment. In the case of co-located development work, it helps to decrease documentation and reduces the number of email communications. In an open office environment, the team members can easily keep up to date with the whole project view and notice obstacles in their colleagues’ work [P6] [P17].

Table 6. Organizational responsibilities

Practices	Findings
Management role	<ul style="list-style-type: none"> -Training on ASD is needed if the team has never used it before. -Reading documentation is not enough [GSD - P10] [P13]. -Necessary resources (corporation intranet, documentation ...) must be available to all team members [GSD [P5]. -The project manager should speak all the languages of the developers involved in the project [GSD P3]. -The project manager should collaborate on a daily basis with all the distributed teams [GSD P3]. -Project managers are key players in distributing information to others and being aware of others [GSD P8]. -The role of project managers is essential to the development of project requirements [GSD P8]. -Focus and practice people- vs. process-oriented control [P1].
Open office spaces	<ul style="list-style-type: none"> -Help to decrease the need for documentation [P6] -Enable everybody to have knowledge of the project status and common goals [P6] -Co-located office spaces close to the customer improve customer relationships since they foster communication [P17].

Table 7. Team-related communication practices and benefits.

Practice	Findings
Leadership & Trust	<ul style="list-style-type: none"> -Team leaders need to trust other team members [P5]. -The role of the team leader cannot be underestimated [P5].
Effective management	<ul style="list-style-type: none"> -Effective management helps to balance power [P5]. -Lack of necessary resources (corporation intranet, documentation) can lead to frustration and lower the motivation of the team [P5]. -Effective management makes projects successful [P5].
Emergent team members	<ul style="list-style-type: none"> -On-demand involvement of emergent members in a team helps to smooth out difficulties rather than getting stuck on a certain point that may lead to delay and failures [P8]. -Gathering information from outside members (i.e., support team, management team, executives...) is more necessary at the start of the project [P8].
Exchange of team members	<ul style="list-style-type: none"> -A visiting engineer or outside expert is highly beneficial to support inexperienced teams during the first iteration [P10]. -Frequent exchange visits of team members are beneficial at the beginning of the project or in critical phases to get in touch with other members and learn how to work together [P11]. -Exchange visits of team members and visiting schedules must be properly planned without the focus on saving traveling budget [P11]. -Agile teams need to consist of experts. Novices should be introduced to stakeholders gradually [P13].
Customer communication	<ul style="list-style-type: none"> -The customer must be readily accessible for communication with the development team [P3]. -Daily communication with customer reduces effort overrun [P9]. -Information hiding and lack of efficient customer relationship may lead to inefficient communication [P1] [P10][P19][P20][P21]. -Face-to-face communication between customer and development team is beneficial during project inception to discuss project goals [P14].

IV. DISCUSSION

Communication plays a crucial role in software, services, and systems development. The literature suggests synchronous and asynchronous communication channels in both co-located and distributed ASD. Many communication

challenges can be avoided with the right strategies. When face-to-face, synchronous communication is infeasible, the use of email increases the chance of getting a response and encourages prompt, useful, and conclusive responses. The blended approach of using various kinds of technology-mediated communication helps to avoid communication gaps among various development sites and/or teams. Software development teams need to promote, and be encouraged to maintain, healthy cooperation. Daily stand up meetings with the aid of various communication tools ensures a synchronous communication environment, as result build mutual understanding among distributed project stakeholders [P7].

The “teamness” and one-team attitude is a good strategy. It brings together team members across different locations and encourages cooperation between the team and the customer. Scrum planning meetings help increase “teamness” and reduce geographical distance. To get effective communication among team members, non-verbal communication is very important. In this regard, communication in co-located and distributed teams is not the same. Body language and hand gestures are difficult to observe in distributed teams. However, we can address such challenges to some extent by using technology-mediated communication. Minimizing the physical distances and using heavily technology-mediated communication can help software development teams build trust and work efficiently. The use of video-conferencing, the exchange of developers between different sites, and the deployment of emergent members, which helps in knowledge sharing, play a significant role in enabling effective communication in ASD teams. Furthermore, the PS discussed that Extreme Programming is useful for the more technical and coding aspects of GSD projects, whereas Scrum practices are good for GSD planning and tracking [P2].

Management needs to understand that spending some of the budget on exchanging team members among various sites fosters understanding of different cultures and facilitates communication. The challenges related to communication do not always stem from the use of the various communication media themselves, but may also be due to other reasons, such as fixed requirements, process-oriented control, lack of efficient customer relationship, proxy customer with no conclusive decision power. This results in inefficient communication and reduced efficiency of the communication media. These challenges can be avoided through various strategies, such as: defining the role of the customer up front and providing conclusive decision power regarding the project’s functionality and scope; customer being readily accessible; customer having a vested interest in the project. Furthermore, the use of globally available project management tools is recommended in order to record and

VI. CONCLUSION

In this work, we investigated communication channels and practices adopted in Agile software development using a Systematic Mapping Study. The 25 primary studies provide

monitor the project status on a daily basis. To avoid low motivation of co-located and distributed software development teams, they should be granted access to the necessary resources (e.g., corporation intranet, email, product documentation, etc.). Additionally, it is recommended providing the necessary methodology training (e.g., Scrum, Kanban) and allow teams to experiment or pilot the new method in their work. Such training should be followed up with internal coaching to reap maximum benefits.

V. THREATS TO VALIDITY

In this section, we report the threats to validity, applying the structure suggested by Yin [1]. We identified and how we mitigated them based on SMS guidelines [2-5]. Moreover, the guideline proposed by Petersen et al. [2] suggests an objective checklist for assessing the quality of a study. The checklist considers information about activities conducted in the review, the need of the review, the search strategy adopted and its evaluation, extraction and classification process. We achieved an excellent score of 72% compared to the average (33% - 48%) of similar studies [2]. This value is the ratio of the number of actions taken in a review compared with the total number of actions required by the checklist.

Internal Validity. We defined the protocol based on the guideline proposed by [4] in a rigorous manner. As this protocol is the one most frequently used by researchers in the software engineering domain, we are sure that we have avoided any possible bias regarding the design of the methodology.

External Validity. Regarding the representation of the state of the art on communication in Agile development processes, we avoided this issue in our search and selection strategy by using a combination of automatic search in the bibliographic sources and backward-forward snowballing on the references of the selected studies. We did not consider papers that were not peer-reviewed in order to obtain high quality in our results.

Construct Validity is about bringing the right measures for the concept being investigated [2]. In order to reduce this threat, a data collection process was designed as suggested by Kitchenham and Charters [4]. We iteratively refined the inclusion and exclusion criteria by selecting a set of initial papers on which we tested their performance with regard to our goal. We also guaranteed inter-researcher agreement during the search and selection process.

Reliability. The results obtained from the selected papers allowed us to answer the defined research questions in the best possible way. This means that the data extraction process was well designed. Performing our SMS according to the guidelines [3] and providing raw data, will allow other researchers to easily replicate this study.

a detailed background regarding communication in ASD. It is identified that synchronous communication (i.e. face to face and phone calls) is dominantly used in ASD compare to asynchronous communication channels. It is evident that, even in GSD, pair programming with help of

videoconference easily applied and turns out to be highly beneficial. Various Agile practices such as Scrum/Sprint planning Meetings, reflection, retrospective reviews and daily standup meetings are beneficial in both ASD and GSF. However, the team managers and leaders should know that variety of culturally sensitive behavior and values. Further, along with the identified communication practices and channels, we found the following strategies for promoting effective team interaction in development teams:

- *Leadership & Trust: proactive role of management;*
- *Promoting the exchange of team members between sites;*
- *Active role of customer with conclusive power;*
- *Establishment of open office spaces in the case of co-located teams;*
- *Necessity of good information sharing tools selection at the beginning of project especially in case of GSD;*
- *Blended use of technology-mediated communication channels;*
- *Management should avoid in assigning work beyond capacity because teams easy burn and excused;*
- *Practice people- vs. process-oriented control*

The primary studies highlighted that practicing these strategies promotes team interaction between members from different sites/locations/units. These strategies are not the only ones for promoting communication; other strategies might exist but did not emerge from our analysis. Future works will include a set of industrial case studies and surveys to validate the results presented here.

REFERENCES

- [1] Yin R.K. , “Case Study Research: Design and Methods”, 4th edition, Sage, 2009.
- [2] Petersen, K., Vakkalanka, S., Kuzniarz, L., “Guidelines for conducting systematic mapping studies in software engineering: An update”. *Information and Software Technology*. vol. 64, pp. 1-18. 2015.
- [3] Wohlin, C., “Guidelines for snowballing in systematic literature studies and a replication in software engineering”. 18th International Conference on Evaluation and Assessment in Software Engineering (EASE '14), Article 38. 2014.
- [4] Kitchenham, B., Charters, S., “Guidelines for Performing Systematic Literature Reviews in Software Engineering”, Version 2.3. 2007.
- [5] D. Taibi and Lenarduzzi, V. , “MVP explained: A Systematic Mapping on the Definition of Minimum Viable Product”, in *Proceedings of the 42th Euromicro Conference Series on Software Engineering and Advanced Applications*, 2016
- [6] Espinosa J.A. and Carmel E., “The impact of time separation on coordination in global software teams: a conceptual foundation”. *Software Process: Improvement and Practice*. vol. 8(4), pp. 249–266. 2003.
- [7] Harbring, C. “The effect of communication in incentive systems—an experimental study”. *Manage. Decis. Econ.*, vol. 27. pp. 333–353. 2006.
- [8] Malone T. W. and Crowston K., “The interdisciplinary study of coordination”. *ACM Comp. Surv.* v. 26(1), p. 87-119. 1994.
- [9] Carmel E. and Agarwal R., “Tactical Approaches for Alleviating Distance in Global Software Development”. *IEEE Softw.* vol. 18(2), pp. 22-29. 2001.
- [10] Herbsleb, J.D. and Mockus A., “An empirical study of speed and communication in globally distributed software development”. *IEEE trans. on Soft. Eng.* v. 9(6), p. 1-14. 2003.
- [11] Henttonen K. and Kirsimarja B., “Managing distance in a global virtual team: the evolution of trust through technology-mediated relational communication”. *Strat. Change*. vol.14. pp. 107–119. 2005.
- [12] Korkala M., Abrahamsson P., and Kyllonen P., A case study on the impact of customer communication on defects in agile software development. *AGILE 2006*, pp. 76-88, 2006.
- [13] Melnik, G., and Maurer, F., *Direct Verbal Communication as a Catalyst of Agile Knowledge Sharing*. *AGILE 2004*, 2004.
- [14] Sarker, S., and Sarker, S., Exploring Agility in Distributed Information Systems Development Teams: An Interpretive Study in an Offshoring Context. *Information Systems Research*, Vol.20(3), pp.440-461, 2009.
- [15] Wang, X., Conboy, K., and Pikkarainen, M., Assimilation of agile practices in use. *Information Systems Journal*. Vol 22(6), pp. 435-455, 2012.
- [16] Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., and Still, J., The impact of agile practices on communication in software development. *Empirical Software Engineering*. Vol. 13(3), pp. 303-337, 2008.
- [17] Koskela, J., and Abrahamsson, P., On-Site Customer in an XP Project: Empirical Results from a Case Study. *Torgeir Dingsoyr (Ed.) Software Process Improvement*, Springer, Berlin Heidelberg, pp.1-11, 2004.
- [18] Mishra, D., and Mishra, A., Effective communication, collaboration, and coordination in eXtreme Programming: Human-centric perspective in a small organization. *Human Factors and Ergonomics in Manufacturing & Service Industries*. Vol 19(5), pp.438-456, 2009.
- [19] Mishra, D., Mishra, A., and Ostrovska, S., Impact of physical ambiance on communication, collaboration and coordination in agile software development: An empirical evaluation. *Information and Software Technology*. Vol 54(10), pp.1067-1078, 2012.
- [20] Bhalerao, S., Puntambekar, D. and Ingle, M., Generalized agile software development life cycle. *International Journal of Computer Science and Engineering*. Vol I (3), 2009.
- [21] Turner, R. and Boehm, B., *Balancing Agility and Discipline: A Guide for the Perplexed*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA. 2003
- [22] Taibi, D., Lenarduzzi, V., Ahmad, M.O., and Liukkunen, K., “Comparing Communication Effort within the Scrum, Scrum with Kanban, XP, and Banana Development Processes”. *EASE*. pp. 258-263. 2017.
- [23] Ahmad, M.O, Denis, D, Kieran, C, and Markku, O. "Kanban in software engineering: A systematic mapping study." *Journal of Systems and Software* 137 (2018): 96-113.
- [24] B.A. Kitchenham, D. Budgen, O.P. Brereton. “Using mapping studies as the basis for further research—a participant-observer case study” *Inf. Softw. Technol.*, 53 (6) (2011), pp. 638-651.
- [25] D. Taibi, Lenarduzzi, V. , Janes, A., Liukkunen, K. , and Ahmad, M. Ovais, “Comparing Requirements Decomposition Within the Scrum, Scrum with Kanban, XP, and Banana Development Processes” *XP 2017*

- [26] B. Rizvi, E. Bagheri, D. Gasevic, D. "A systematic review of distributed Agile software engineering". *Journal of Software: Evolution and Process*, 27(10), 723–762. 2015.
- [27] E. Hossain, M.A. Babar, H.Y. Paik "Using scrum in global software development: a systematic literature review". *ICGSE 2009*. pp. 175-184.
- [28] Y.I. Alzoubi, A.Q. Gill, A. Al-Ani. "Empirical studies of geographically distributed agile development communication challenges: a systematic review" *Inf. Management*, 53 (1), pp. 22-37. 2016
- [29] R. Vallon, B.J.d.S. Estácio, R. Prikładnicki, T. Grechenig. "Systematic literature review on agile practices in global software development" *Information and Software Technology*, Vol(96), pp. 161-180 2018
- [30] R. Hoda, N. Salleh, J. Grundy, H. Mien Tee.. Systematic literature reviews in agile software development. *Inf. Softw. Technol.* 85, C (May 2017), 60-70. 2017.
- [31] Ahmad, M. O., Markkula, J., & Oivo, M. (2013, September). Kanban in software development: A systematic literature review. In *Software Engineering and Advanced Applications (SEAA), 2013 39th EUROMICRO Conference on*. pp. 9-16.
- P10. Korkala, Mikko, Minna Pikkarainen, and Kieran Conboy. "Distributed Agile Development: A Case Study of Customer Communication Challenges". *Agile Processes in Software Engineering and Extreme Programming: 10th International Conference, XP*, pp.161-167. 2009.
- P11. Paasivaara, M., Durasiewicz, S., and Lassenius, C. "Using Scrum in Distributed Agile Development: A Multiple Case Study". In *Proceedings of the 2009 Fourth IEEE International Conference on Global Software Engineering*. 2009.
- P12. Sharp, H., and Robinson, H. "Three 'C's of Agile Practice: Collaboration, Co-ordination and Communication". *Agile Software Development: Current Research and Future Directions*. pp. 61-85. 2010.
- P13. Gulliksen, V. Stray, V., Moe, N.B., and Aurum, A. "Investigating Daily Team Meetings in Agile Software Projects". *EUROMICRO 2012*.
- P14. Dorairaj, S., and Noble, J., and Malik, P. "Effective Communication in Distributed Agile Software Development Teams". *Agile Processes in Software Engineering and Extreme Programming XP 2011*, pp. 102-116. 2011.
- P15. Persson, J.S., Mathiassen, L., and Aaen, I. "Agile distributed software development: enacting control through media and context". *Info. Sys. Jour.* Vol. 22(6). pp. 411-433. 2012.
- P16. Inayat, I.N., Muhammad A., and Zubaria I. "Facilitating an Off-Site Customer in Product-Based Agile Software Development: An Industrial Case Study". *Emerging Trends and Applications in Information Communication Technologies, IMTIC 2012*, pp. 210-221. 2012.
- P17. Hummel, M., Rosenkranz, C., and Holten, R. "The Role of Social Agile Practices for Direct and Indirect Communication in Information Systems Development Teams". *Communications of the Association for Information Systems: Vol. 36, Article 15*. 2015.
- P18. Lenarduzzi V., Lunesu I., Matta M., Taibi D. "Functional Size Measures and Effort Estimation in Agile Development: A Replicated Study". *XP 2015. LNBIP*, vol 212. 2015
- P19. Taibi D., Lenarduzzi V., Pahl C. "Processes, Motivations, and Issues for Migrating to Microservices Architectures: An Empirical Investigation," in *IEEE Cloud Computing*, vol. 4, no. 5, pp. 22-32, September/October 2017.
- P20. Munari S., Valle S., Vardanega T. (2018) Microservice-Based Agile Architectures: An Opportunity for Specialized Niche Technologies. In: Casimiro A., Ferreira P. (eds) *Reliable Software Technologies – Ada-Europe 2018*. Ada-Europe 2018.
- P21. D. Taibi and Lenarduzzi, V. , "On the Definition of Microservice Bad Smells", *IEEE Software* , v. 35, no. 3, 2018.
- P22. Zykov, S. "Agile Patterns and Practices." pp. 107-134. Springer, Cham, 2018.
- P23. D. Taibi, Diebold, P., and Lampasona, C. , "Moonlighting Scrum: An Agile Method for Distributed Teams with Part-Time Developers Working during Non-Overlapping Hours", in *ICSEA - International Conference on Software Engineering and Advances, Venice (ITALY)*, 2013.
- P24. D. Taibi, Lenarduzzi, V., Diebold, P., and Lunesu, I. , "Operationalizing the Experience Factory for Effort Estimation in Agile Processes", in *21th Evaluation and Assessment in Software Engineering (EASE)*, 2017.

APPENDIX A: SELECTED PRIMARY STUDIES (PS)

- P1. Korkala, M., Abrahamsson, P., and Kyllonen, P. "A Case Study on the Impact of Customer Communication on Defects in Agile Software Development". In *Proceedings of the conference on AGILE 2006 (AGILE '06)*. 2006.
- P2. Holmström, H., Fitzgerald, B., Ågerfalk, P.J., and Conchúir O.E. "Agile Practices Reduce Distance in Global Software Development". *Information Systems Management*. vol. 23(3), pp. 7-18. 2006.
- P3. Layman, L., Williams, L., Damian, D., Bures, H. "Essential communication practices for Extreme Programming in a global software development team". In *Information and Software Technology, Volume 48, Issue 9, Pages 781-794*. 2006.
- P4. Korkala, M., and Abrahamson, P. "Communication in Distributed Agile Development: A Case Study". *EUROMICRO*. 2007
- P5. Cichocki, P., and Maccari, A. "Empirical Analysis of a Distributed Software Development Project". In *Balancing Agility and Formalism in Software Engineering*. Vol. 5082, pp.169-181. 2008.
- P6. Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., and Still, J. "The impact of agile practices on communication in software development". *Empirical Softw. Engg.* Vol. 13,(3), pp. 303-337. 2008.
- P7. Emam, H., Babar, M.A., and Verner, J. "How Can Agile Practices Minimize Global Software Development Co-ordination Risks?" *Software Process Improvement: 16th European Conference, EuroSPI 2009*, pp 81-92. 2009
- P8. Inayat, I., Marczak, S., Salim, S.S., and Damian, D. "Patterns of Collaboration Driven by Requirements in Agile Software Development Teams". *23rd International Working Conference, REFSQ 2017*. pp.131-147. 2007.
- P9. Molokken-Ostfold, K., and Furulund, K.M. "The Relationship between Customer Collaboration and Software Project Overruns". *AGILE 2007*. pp. 72-83. 2007.