palgrave
communications
HUMANITIES | SOCIAL SCIENCES | BUSINESS

ARTICLE

https://doi.org/10.1057/s41599-019-0384-9 OPEN

Uncovering the failure of Agile framework implementation using SSM-based action research

Kevin Suryaatmaja[1,2*], Dermawan Wibisono[1,3], Achmad Ghazali[1] & Rachma Fitriati[4]

ABSTRACT It is undeniable that the Agile software development (SD) framework has been commonly used in information technology projects. However, many of the early adopters failed to implement it properly. Some researchers had proposed a solution by using the adaptation framework as a support, but they cannot clearly explain its development. Furthermore, the correlation between adaptation framework and problem solving is dubious. This study reviews and discusses the literature and empirical studies on the failure of Agile SD implementation by revisiting the supporting framework that used to comprehend the issues in Agile SD implementation. A comparative approach is used to review the literature available in academic databases, such as ProQuest, EBSCO, IEEE, and Scopus, to ensure research originality and novelty. By filtering 87 papers that met the initial criteria, only 4 researchers focused on the area of knowledge management and used the learning process as a supporting framework in Agile SD implementation. However, they did not clearly explain the development of the framework for Agile SD. The author's initial research on a corporation reveals the failure of Agile SD implementation. Soft systems methodology (SSM)-based action research (AR) proposed to identify the real issues in Agile SD implementation. SSM-based AR provides a solution by clarifying the development of a framework using tacit and explicit knowledge to validate the development process. This study can help practitioners and academics understand the process of Agile SD implementation by early adopters who still need assistance with its application. This study is the first to explore Agile SD implementation in a corporation using the experience-based approach, contributing to knowledge improvement. It contributes to the scholarly literature by presenting concepts that combine human-centric aspects and knowledge management to improve the learning in the organization.

[1] School of Business and Management, Institut Teknologi Bandung, Jl. Ganesha No. 10, Bandung, West Java 40132, Indonesia. [2] School of Information Systems, Bina Nusantara University, Jl. K. H. Syahdan No. 9, West Jakarta, DKI Jakarta 11480, Indonesia. [3] Faculty of Economy and Business, Universitas Pertamina, Jl. Teuku Nyak Arief, Simprug, South Jakarta, DKI Jakarta 12220, Indonesia. [4] Faculty of Administrative Science, Universitas Indonesia, M Building 2nd Floor. Kampus UI, Depok, West Java 16424, Indonesia. *email: kevin.suryaatmaja@sbm-itb.ac.id

Introduction

Learning in an organization has been acknowledged to be a fundamental process for collecting, analyzing, storing, and disseminating information in order to maintain knowledge in an organization. It brings individual knowledge to the organization through a shared space that requires a mutual commitment between individuals that is sustained over time (Nonaka et al., 2000; Boateng, 2011; Dixon, 2017). The literature on learning, in general, is broad, ranging from learning at the individual level to the team and organizational levels. The link between each type of learning is enriching the knowledge stream.

Fiol and Lyles (1985), Kim (1993), and Hayes and Allinson (1998) suggest that individual learning is a crucial factor in organizational learning. This is consistent with Popper and Lipshitz's (2000) claim that an individual member is an intermediary for learning in an organization. Huysman and de Wit (2002) and Hecker (2012) further state that the constructive knowledge of the individual triggers organizational knowledge.

Studies by Berg (1993), Hurley and Cunningham (1993), Bennett and O'Brien (1994), Bain (1998), and Chan (2003) assess the relationship between the individual and team learning and find that a recurrent cycle of experience and knowledge among individuals triggers team learning through shared knowledge, a point that is consistent with Bierly and Hamalainen (1995) and Thompson and Zondlo (1995).

Brown and Duguid (1998) emphasize the relationship between individual learning and group learning in certain circumstances. Jones et al. (2006) also show that a group of individuals who discuss their experiences and contradictory judgments produces collaborative learning. Heikkila and Gerlak (2013) and Love et al. (2015) support this argument by finding a relationship between individual and collective learning. An equal collective knowledge is discovered through a procedure of persistent sense-making among individuals.

Some researchers also discuss the relationship between team learning and organizational learning. Chan (2003) discusses how organizational learning is a successor of the team's collective intelligence. Senge (1990), Bennett and O'Brien (1994), Argyris (1999), and Edmondson (1999) claim that organizational learning is a derivative of the collective intelligence gained from teams. This is consistent with Edmondson et al. (2007) statement that team learning is a construction of organizations. Another later study found a relationship between team learning and organizational learning (Tanyaovalaksna and Li, 2013).

A few papers have considered the role of knowledge management in software development (Walz et al., 1993; Robillard, 1999; Lindvall and Rus, 2002; Desouza, 2003; Tiwana, 2004; Levy and Hazzan, 2009; Dingsøyr et al., 2012; Von Krogh, 2012; Ozer and Vogel, 2015). Software development (SD) involves many forms of a particular knowledge to handle business issues (Tiwana, 2004). Issues defined as a complex and time-sensitive need to be solved by those with a high-quality of knowledge (Ozer and Vogel, 2015). Collective experience and the skills of software developers facilitate the development of high-quality software that addresses customer needs (Turk et al., 2005; Fernández-Sanz and Misra, 2011; Li et al., 2013). Chau et al. (2003) and Gervasi et al. (2013) discuss how software developers and customers frequently utilize both tacit knowledge and explicit knowledge in their informal communications. A casual interaction helps a development team comprehend the tacit knowledge held among them because systemic knowledge is difficult to understand (Ozer and Vogel, 2015). The development team will have a better understanding of customers' needs through discussions with the customers, which creates a mental model and better skills (Chau and Maurer, 2004; Dorairaj et al. 2012; Ghobadi and Mathiassen, 2014). A closed interaction among colleagues shapes team learning, which is fundamental for the evolution of an organization (Vidgen and Wang, 2009; Hummel et al. 2012, 2015).

Therefore, knowledge sharing in an organization is needed to share domain expertise between the customer and the development team in order to recognize the necessary software requirements, capture the tacit knowledge of the development team, incorporate organizational knowledge among the individuals involved, retain the knowledge gained, and improve organizational knowledge dissemination. Accordingly, Misra et al. (2010) emphasize the significance of tacit knowledge, skills, and knowledge level in Agile SD (Kalermo and Rissanen, 2002). Some researchers also mentioned the importance of tacit knowledge in the process of learning Agile SD (Boehm, 2002; O'Connor, 2004; King, 2011; Karlsen et al., 2011; Dingsøyr et al., 2012; Dingsøyr and Smite, 2013; Babb et al., 2014; Santos et al., 2015). We can presume that individuals and their interactions with others during the process of learning is the most crucial factor in software development.

The objectives of the Agile SD framework are to meet business needs (Anquetil et al., 2007; Gregory et al., 2014), increase customer satisfaction, decrease bug rates, shorten development cycles, and change and adapt to the business's requirements quickly (Boehm and Turner, 2003; Parrish et al., 2004; Miller and Larson, 2005; Joey Cho, 2010). These become the essential topic in the modern organization, as many organizations shift from the waterfall SD framework to the Agile SD framework (West et al., 2010). However, the implementation of Agile SD has often failed. Referring to the Table 1, it is clear that implementing Agile SD is challenging. Therefore, further assessment is needed to overcome this.

To overcome the issues mentioned in previous paragraph, some studies on frameworks supporting Agile SD implementation have

Table 1 Agile versus waterfall.

Chaos resolution (Agile vs Waterfall)

Size	Method	Successful (%)	Challenged (%)	Failed (%)
All Size	Agile	39	52	9
Projects	Waterfall	11	60	29
Large size	Agile	18	59	23
Projects	Waterfall	3	55	42
Medium size	Agile	27	62	11
Projects	Waterfall	7	68	25
Small size	Agile	58	38	4
Projects	Waterfall	44	45	11

Note: The resolution of software projects from FY2011–2015 within the new CHAOS database, segmented by the Agile process and waterfall method. The total number of software projects is over 10,000
Source: Chaos Report (2015)

been conducted, primarily in the knowledge management area (Sidky et al., 2007; Sureshchandra and Shrinivasavadhani, 2008; Mannaro, 2008; Qumer and Henderson-Sellers, 2008; Soundararajan and Arthur, 2009; McAvoy and Butler, 2009; Esfahani et al., 2010; Krasteva et al., 2010; Esfahani, 2012; Lee and Yong, 2013; Nikitina and Kajko-Mattsson, 2014; Rasnacis and Berzisa, 2016; Hoda and Noble, 2017; Pries-Heje and Baskerville, 2017). Furthermore, some researchers have focused on the factors that refer to the learning process, but only a few have included human factors as determinants (Qumer and Henderson-Sellers, 2008; McAvoy and Butler, 2009; Soundararajan and Arthur, 2009; Lee and Yong, 2013; Nikitina and Kajko-Mattsson, 2014; Hoda and Noble, 2017; Pries-Heje and Baskerville, 2017). Nevertheless, none of the research uses the interpretivist approach for a more appropriate framework to supplement the current studies on Agile SD framework implementation. Therefore, the objective of this study is to uncover the past research findings pertaining to the following questions. Is any current supporting framework able to discover the actual issues in Agile SD implementation? How does the supporting framework support the learning process exploration during Agile SD implementation?

To explore previous studies on the implementation of the Agile SD framework, the author searched the existing research on the framework supporting Agile SD implementation in the world research database, ranging from the initial investigation in the 1990s to the most recent studies conducted in 2018 (1359 studies were selected in total). The search was limited to the engineering and computer science disciplines as the general domain of Agile software development. Based on the search results, this paper reviews the research approach, research strategies, theoretical & conceptual analysis, units of analysis, framework models, describes the advancements, offers recommendations for further research, and summarizes the vital contributions to this domain of study.

To date, few studies have been conducted in the software development area on the effect of the functioning of the learning process. The functioning of the learning process is to enable the process of creating, maintaining, and transferring knowledge in an organization. Academics have attempted to understand what learning is and to understand how to develop learning in specific contexts and with specific contents. Despite the context and different kinds of content, there is not much we know about the process. The objective of this study is to focus on a specific aspect of a learning process in the area of software development rather than the learning process in general. Therefore, the main goal is to establish an awareness of the importance of the learning process in implementing Agile SD and to uncover any issues in Agile SD implementation.

Since this study focuses on the Software Development area, it is essential to use a research method (action research) relevant to information systems (Avison et al., 1999; Hartmann et al., 2009; Wieringa and Morali, 2012; Avison et al., 2017). Furthermore, this study uses action research, whose focus is twofold, to create knowledge and to take action, because of the need to solve social or organizational issues by the participant of the system him- or herself (Coughlan and Coghlan, 2010). To understand problems in the organization or process, it is important to consider the member of the system as the "owner" of the problem. Using action research also helps us to understand tacit knowledge, since action research is theoretically based on "something felt" (actual) from a rigorous academic perspective (Uchiyama, 2009; Hanafizadeh and Ghamkhari, 2019).

Soft systems methodology (SSM)-based action research is an experience-based methodological knowledge interested in accommodating people's interests in "actuality" (something felt). The word "accommodation" refers to living with different worldviews based on a shared "reality" (Uchiyama, 2009; Fan and Kuang, 2013). SSM-based action research fits the purpose of this research by organizing the exploration of learning from a complex system.

Thus, the first task is to define the scope and type of learning that increases the understanding of the critical theoretical concept, followed by an explanation of SSM to increase the understanding of this research methodology and to offer conclusions in Key concept. Section Existing Framework presents research on the current literature on each construct and reviews the findings. The methodology uses SSM-based AR to produce a conceptual analysis of Agile SD practices, which is explained in the section Methodology. Section Discussion discusses the research results using the proposed model. Finally, future research recommendations are made in the section Conclusion and future research.

Key concept

The relationship between individual and organizational learning has been discussed in many studies. One of the earliest articles in the literature was by Argyris and Schön (1978), which emphasized that for an organization to learn, it must consider individuals as "agents." Later research is still consistent with this and emphasized that the individual is the agent of organizational learning (Friedman, 2002). Nonaka and Takeuchi (2007) discussed how new knowledge always starts from the individual, and the process by which individual knowledge becomes available to others is a crucial activity for a knowledge-creating company. Ikehara (1999) claims that a learning organization integrates individual learning into an organization's learning process. This claim supports Kim (1993) and Matlay's (2000) finding that there is a significant relationship between individual learning and organizational learning. One of the later studies further discussed how individuals in the organization trigger the organization's learning process (Wang and Ahmed, 2003; Boateng, 2011; Dixon, 2017).

Although it is difficult to measure learning, some researchers have found ways to assess it. Hyland and Matlay (1997) mentioned that learning organization could be measured by accumulating individual learning and collective learning. This is reasonable since some knowledge is tangible and visible and, thus, it could be easily communicated, stored, and retrieved for future use. This knowledge is called "explicit knowledge" (Dalkir, 2011). Explicit knowledge constitutes content that has been captured in some tangible form, such as a word processor's documentation, in audio recording, or as images (Dalkir, 2011). However, Hyland and Matlay (1997) assert that tacit knowledge cannot be measured because it is in the head of other individuals. As Nonaka and Takeuchi (1995) noted, tacit knowledge is mainly in the heads of individuals, and it is not brought forth until it is needed.

Mahroeian and Forozia (2012) acknowledge that disseminating tacit knowledge is a challenge. In line with Dalkir (2011), they note that tacit knowledge is difficult to articulate because it tends to reside within the heads of those who have knowledge and it is very difficult to externalize to others; in terms of some content, it may be explicit for one person but tacit for another. van den Berg (2013) emphasizes that tacit knowledge is difficult and costly to use for other purposes because its complexity and abstractness is difficult for an organization to create, replicate, and share. The complexity of tacit knowledge is also noted in Choo (1996) and emphasized by van den Berg (2013), both of whom point out that tacit knowledge is uncodified knowledge that commonly refers to complex, unrefined, difficult to articulate, implicit, internalized, and abstract ideas. Moreover, this tacit knowledge is accumulated in the minds of employees through experience over time. As such, it tends to be more valuable when compared to explicit knowledge because tacit knowledge is difficult to articulate (Dalkir, 2011).

The following section explores the types of learning and bridges the gap between individual and organizational learning.

Types of learning

Individual learning. Fiol and Lyles (1985), Kim (1993), and Hayes and Allinson (1998) argue that individual learning is essential for organizational learning, as all organizations are composed of individuals who can learn independently. This caused psychologists, linguists, educators, and others to focus heavily on the topic of learning at the individual level. Popper and Lipshitz's (2000) claim that the individual is an intermediary of learning in the organization. Huysman and de Wit (2002) and Hecker (2012) strengthened this argument by stating that the constructive knowledge of the individual triggers organizational knowledge.

Berg (1993), Hurley and Cunningham (1993), Bain (1998), and Chan (2003) assessed the relationship between individual and team learning and found that there is a recurrent cycle of experience and knowledge among individuals that triggers team learning through shared knowledge, which is consistent with reports by Bierly and Hamalainen (1995) and Thompson and Zondlo (1995). Brown and Duguid (1998) emphasize that there is a relationship between individual learning and group learning in some circumstances. Jones et al. (2006) also describe how a group of individuals who are involved in a discussion of their experiences and contradictory judgments produces collaborative learning. In another study, Sadler-Smith (2006) claims that the processes of collective learning relate to a nonconscious way of learning from individual intuitions. Heikkila and Gerlak (2013) and Love et al. (2015) strengthen this argument by revealing a relationship between individual and collective learning. An equal collective knowledge is discovered through a procedure of persistent sense-making among individuals (Brix, 2017).

However, this theory is mainly subjective; thus, an empirical exploration is needed to prove that there is a relationship between individual and team learning. Garavan and McCarthy (2008) mention that team learning is not simply an aggregation of individual knowledge but also has attributes that are coupled with it.

Team learning. Knapp (2010) discussed how some articles in the literature has conceptualized team learning as a subset of a community of practice (Brown and Duguid, 1991), as a subset of a learning organization (Argyris, 2006), or as a useful outline of an organizational phenomenon where the team acts, retrieves, and reflects on feedback to adapt or improve in terms of a transformational process (Edmondson, 1999) or in terms of a dynamic learning process (Sessa and London, 2008). Jones et al. (2006) conceptualize collaborative learning as a situation in which a group of individuals engages in a discussion to reflect upon their experiences and judgments. As Senge (1990, p. 236) puts it, "team learning is the process of aligning and developing the capacity of a team to create the results its members truly desire."

Some researchers also have investigated the relationship between team learning and organizational learning. Chan (2003) discusses how organizational learning is a successor of the team's collective intelligence. This claim was originally made by Senge (1990), Bennett and O'Brien (1994), and Edmondson (1999), who argue that organizational learning is a derivative of the collective intelligence gained from teams. This argument was further strengthened by Argyris's (1999) claim that the learning organization is a part of team learning as well as Edmondson et al. (2007) claim that team learning is an organizational construction. Vidgen and Wang (2009) further reveal that a close interaction among team members triggers team learning, which is vital for an organization to evolve and co-evolve. A later study also found a relationship between team learning and organization learning (Tanyaovalaksna and Li, 2013).

Organizational learning. Organizational learning refers to the process of the acquisition and interpretation of information, direct learning from one's own experience, and learning from other organizations (Dermol, 2013) to which an organization must adapt (Shrivastava, 1983). Dalkir (2011) mentions that organizational learning is defined as the process of learning from what has worked and what did not work in the past and transferring this learning experience to the current situation and future knowledge workers. This includes making improvements over time to achieve innovation as well as making the organization more efficient, effective, and competitive (Senge, 1990; Garvin, 1993). Organizational learning may take place in a planned or informal and less defined learning environment (Nevis et al. 1995; Manuti et al. 2015). Nevis et al. (1995) describe how an organization could improve or maintain its performance based upon its experience, as well as how collective learning is needed for an organization to survive and grow. However, it is difficult to improve the organizational performance based on its experience because some learning is dysfunctional, such as when the organization focuses on its learning capacity and not the usefulness of the learning process. Therefore, collective learning is a dynamic and cumulative process that leads to knowledge production (Garavan and Carbery, 2012).

Bridging the gap between individual and organizational learning. Three factors that bridge the gap between individual learning and organizational learning must be considered, which are learning objectives, learning factors, and the learning process (Pauleen and Gorman, 2011). The learning objectives of individuals are different from the learning objectives of the organization. The learning objectives of individuals are social, personal, and job-related while the learning objectives of the organization, in general, are to improve effectiveness, responsiveness, efficiency, flexibility, and the ability to innovate (Pauleen and Gorman, 2011).

The learning factors of individuals are different from the learning factors of the organization as well. An individual's intrinsic learning factors can be summarized in terms of his or her personal characteristics, self-conception, orientation, readiness/immediate needs, and internal motivation while organizational learning factors can be grouped into organizational structure, culture, and technical infrastructure (Cross, 1981; Knowles et al. 1998; Levine, 2008). An organizational structure is essential for supporting effective knowledge organization and improving individual learning, collective learning, knowledge sharing, and innovation (King, 2008). Thus, in a highly centralized organization, the person who creates the knowledge will share it mainly through direct communication (Yao-Sheng, 2007).

The learning process of individuals is different from the learning process of the organization. Individual knowledge is found mainly in the form of tacit knowledge with subjective insights, intuitions, and hunches (Nonaka and Takeuchi, 1995; Kingston, 2012). Hence, it is highly personal and difficult to formalize, communicate, or share with others (Polanyi, 1966; Mahroeian and Forozia 2012; Kimble, 2013). Organizational knowledge is mainly found in the form of explicit knowledge, which is deeply ingrained in Western management tradition to be found only in the form of words and numbers. Thus, it is easily communicated and shared in the form of hard data, scientific formulae, codified procedures, or universal principles (Nonaka and Takeuchi, 1995). Jarvis (1987) describes how learning is a process of transforming experience into knowledge, skills, and attitudes. Forcheri et al. (2000) offer a more thorough explanation, suggesting that individual learning is the capacity to build knowledge through individual reflection and the re-elaboration of individual knowledge and experience through interactions with others and the environment. This is consistent with Crawford et al. (2006) assertion that a person in an organization learns by interacting with others and gains tacit knowledge as a consequence of this.

Unfortunately, since tacit knowledge is highly personal and difficult to formalize and communicate (Polanyi, 1966; Mahroeian and Forozia, 2012; Kimble, 2013), it is often challenging to externalize and archive it, since communication or collaboration among individuals is not supported by the repository (Crawford et al. 2006). This explains why an organization needs a framework that can connect everyone's knowledge in the organization. Such a framework should capture team learning as the team acts, retrieves, and reflects on feedback, and this framework should adapt or improve as time goes on (Edmondson, 1999) based on the team's discussion of its experiences and explains its contrasting judgments (Jones et al. 2006). The close interaction among members is vital for the evolution and co-evolution of an organization (Vidgen and Wang, 2009).

SSM-based action research. Two approaches have been proposed to answer these problems, the hard systems approach and the soft systems approach. The hard systems approach considers problems as a simple unit on which the participant relies as being shared and able to identify the goal, considering that people's shared values and beliefs and that systems are simple enough to be mathematically modeled. "System dynamics," "organizational cybernetics," and "complexity theory" approaches consider the problem contexts as a complex-unitary that needs to be addressed from a "structuralist" perspective that assumes that the participant has the capacity for goal-seeking and who remains viable in turbulent environments (Jackson, 2003).

The soft systems approach to solving either the simple or complex problems of plural participants who have the same goals could be used to evaluate their subjectivity, working with various worldviews during the methodological process and making explicit their various implications so that alternative perspectives can be explored, compared, and contrasted systemically. The aim is to generate a systemic learning process in which the participants in the problem situation learn to appreciate alternative worldviews and offer possibilities for making accommodations. This approach is fit for a researcher who likes to emphasize the crucial importance of values, beliefs, and philosophies that explore the culture and politics of organizations to see what change is feasible and gain commitment from participants to agree on specific courses of action (Jackson, 2003). The SSM is one of the methodologies used in soft system thinking.

Since this study focused on the software development area, it is essential to use a research method (action research) relevant to information systems (Avison et al., 1999, 2017). Furthermore, this study used action research to offer two approaches simultaneously, namely to create knowledge and to take action, as we need to find a solution to social or organizational issues solved by the participant of the system itself (Coughlan and Coghlan, 2010). To understand the problem in the organization or process, it is important to consider the human inside the organization as the owner of the problem (Mckay and Marshall, 2001; Goldkuhl, 2012). Using action research also helps us understand tacit knowledge, since action research is theoretically based on "something felt" (actual) from a rigorous academic point of view (Uchiyama, 2009). As such, it is distinguished from conventional action research with its vague boundary of positivism. Positivists look for explicit scientific knowledge while an action researcher wants to represent tacit or implicit knowledge (Uchiyama, 2009; Hanafizadeh and Ghamkhari, 2019).

People who do not participate in a particular AR cycle are unable to share the actual learning as it takes place either from the epistemological difference between reality and actuality or from learning by doing something during the implementation of an action (Uchiyama, 2009; Robertson et al. 2017). As Checkland and Poulter note (2006, p. 17), "human situation is not only unique but changes through time and exhibits multiple conflicting worldviews. Action researcher enters a human situation, takes part in its activity, and uses that experience as the research object."

The SSM helps one to organize the exploration of learning in terms of a complex system. The observer perceives the real world as a messy and ill-defined problem and works to solve this by engaging in learning through exploration (Jackson, 2003). SSM-based action research takes off from experience-based knowledge methodologically and is interested in the possibility of accommodating people's interest in "actuality." The word "accommodation" refers to living with different individual worldviews based on a shared "reality" (Uchiyama, 2009). The model of "actuality" used by the SSM-based AR aims to trigger the autopoietic process of AR as experience-based knowledge embodied in tacit level (Uchiyama, 2009).

Existing framework

A systematic evaluation of the research has identified six criteria used in choosing and determining the potential studies for analysis (Meline, 2006). The first criterion is the study population. Empirical studies focus on a group or a team in the organization that were considered to restrict the sample to the relevant cases, such as a group in the area of technology. The second criterion is the nature of the intervention: the selected studies had to provide the framework supporting the implementation of the Agile SD as the intervention. The third criterion is the outcome variables: the selected studies had to address the performance by the learning process in the field of software development. The fourth criterion is the time period: the selected studies on Agile SD were conducted from the period of 1990 to 2018. The fifth criterion is the cultural and linguistic range: the selected studies are all in English to avoid problems in interpretation. The sixth criterion is the methodological quality: the selected studies had to be published in a top-tier journal article to ensure the articles' quality and to ensure the methodologies used in the study are academically acceptable.

Table 2 shows the end-to end systematic approach and study parameters used to identify the literature relevant to this study.

A detailed explanation of phases 1 through 5 is as follows. In phase 1, the author and co-authors discussed and chose an appropriate database that supports high-quality papers in the area of software development, which are mainly in ProQuest, IEEE, Science Direct, and Scopus. In phase 2, the paper searching process began with evaluating abstracts using the keywords "Agile Implementation," "Agile Adaptation," "Agile Adoption," "Agile Implementation Framework," "Agile Adaptation Framework," and "Agile Adoption Framework." Before downloading the papers, the author refined the search by filtering those that are relevant to the area of engineering and computer science. The results are as follows:

1. ProQuest resulted in 66 rows of records;
2. IEEE resulted in 309 rows of records;

Table 2 Overview of the adopted systematic review process.

Phase 1. Choosing the databases
ProQuest, IEEE, Science Direct, and Scopus were considered appropriate for selecting the qualified papers.
Phase 2. Search words used
•"Agile Implementation", "Agile Adaptation", "Agile Adoption", "Agile Implementation Framework", "Agile Adaptation Framework", "Agile Adoption Framework"
Phase 3. Articles refinement
•Limiting the filter to the Engineering and Computer Science area.
•Limiting the downloaded articles by choosing a relevant title based on search words.
Phase 4. Articles review
•To further limit the articles, papers were evaluated against the inclusion criteria.
•The papers that met the inclusion criteria were selected and stored in a summary word database.
Phase 5. Analysis
•The author carefully reviewed articles to understand the context and content of each paper.
•The articles were then summarized in an excel document that included the reference, research approach, research strategies, theoretical & conceptual analysis, unit of analysis, and framework model.

3. Science Direct resulted in 73 rows of records;
4. Scopus resulted in 911 rows of records.

From 1359 records matching the keyword, the author skimmed and selected 87 research papers related to the Agile SD implementation process using a knowledge management approach in a group or team. From the 87 research papers, the author reviewed, evaluated, and selected 22 research papers that discussed the frameworks to support Agile SD implementation. From the 22 research papers, the author reviewed, evaluated, and selectively picked 14 research papers that discussed those frameworks using the knowledge management or learning process approach. A summary of the 14 research papers is shown in Tables 3 and 4.

Three people were involved in the systematic review process, the author and two co- authors, and one of the co-authors supervised the author's dissertation. The author decided to eliminate each article based on criteria that had been decided together with all co-authors.

Tables 3 and 4 summarizes the selected 14 studies that focus on a framework supporting the Agile SD implementation using knowledge management and learning approaches. The selected papers were published between 2007 and 2018.

Based on the information in Tables 3 and 4, none of the studies could help the objective of this study, that is, to describe a framework that would explain the failure in Agile implementation using the learning process approach. Only a few researchers developed a framework using the learning process approach (Qumer and Henderson-Sellers, 2008; McAvoy and Butler, 2009; Hoda and Noble, 2017; Pries-Heje and Baskerville, 2017). However, they did not explain "how" they built the framework, although this is essential for understand- ing the empirical and theoretical baseline for building a framework. The framework used in these studies also does not explain "how" to discover and maintain the learning process to support the goal of this study, which was to address the exploration of the learning process to improve the future implementation processes of Agile SD in the organization. Moreover, previous studies investigated the framework only theoretically thus far and did not apply it practically in the study, which means the association between the conceptual model and actual problem solving is unclear.

McAvoy and Butler (2009) develop a model that is an ideal reference for this study because it examines how team learning occurs in addition to focusing on human centricity, knowledge management, and learning. The only drawback is that their research does not explain how their model supports the learning process in Agile SD implementation. Yu and Petter (2014) introduce the shared mental model theory to examine how a shared understanding in the team creates an internal knowledge-base and establishes a compatible expectation for the team. The framework reflects on how to expose the learning process by explaining the formation of shared mental models through the interdependencies related to the assignments performed inside the group. Unfortunately, they did not describe the formation of mental models in a practical situation, especially learning in Agile SD, even though building a learning framework requires understanding the process of learning empirically rather than just theoretically.

Methodology

In light of the information presented in the previous section, a framework that would capture tacit knowledge is needed, since it is challenging to articulate tacit knowledge given its complex and abstract nature. Accordingly, it is difficult for an organization to create, replicate, and share this knowledge (Choo, 1996; van den Berg, 2013). The framework should cover team learning as a consequence of the close interactions among team members (Vidgen and Wang, 2009). It should capture team learning to depict how the team acts, retrieves, and reflects on feedback as well as how it adapts or improves over time (Edmondson, 1999). Moreover, a framework should able to link employees' knowledge gained from interactions with others inside the organization (Crawford et al., 2006). The framework must support the exploration of the learning process to maintain or improve organizational performance based on a collective learning of individuals' tacit knowledge through team discussions of its experiences and explanations of its contrasting judgments (Jones et al., 2006).

Choosing the SSM. Several methodological challenges associated with the development of a framework that would capture tacit knowledge need to be acknowledged (Choo, 1996; van den Berg, 2013). Uchiyama (2009) mentions two kinds of methods to comprehend the real world: (1) learning from the real world by seeing it as an object and (2) learning from the real world by interacting with it as the first person and internalizing the experience-based knowledge acquired from this interaction. In positivism, the researcher observes the research subject and collects data to support his/her hypotheses. The aim is to prove whether the hypothesis is true or not based on observations; thus it is based on forms of explicit knowledge. With the SSM-based AR, the researcher defines an action based on the epistemological learning that develops through an interaction and experience embodied at the tacit level (Uchiyama, 2009).

Another challenge is that the proposed framework should support team learning as a consequence of close interactions

Table 3 Overview of the current literature of framework to support Agile SD implementation using knowledge management and learning process approaches.

Author	Sidky et al. (2007)	Sureshchandra and Shrinivasavadhani (2008)	Mannaro (2008)	Qumer and Henderson-Sellers (2008)	Soundararajan and Arthur (2009)	McAvoy and Butler (2009)	Krasteva et al. (2010)
Research Approach	Deductive	Deductive	Inductive	Deductive	Deductive	Deductive	Deductive
Research strategies	Interviews, surveys	Case study	Grounded and simulation	Case study, survey, interview	Not available	Case study	Case study, survey
Framework	The four stages 1. Identify discontinuing factors 2. Project level assessment 3. Organizational assessment 4. Reconciliation	(DATM) Distribution Agile Transition Model: Evaluation, Inception, Transition, and Steady	(DAM) Distributed Agile Methodology applying the PPT (People-Process-Technology) model to conceptualize the effect of organizational factors that determine the success or failure in software development	(AAIM) Agile adoption and improvement model: 1. Speed, Flexibility, Responsive 2. Communication 3. Executable artifacts 4. People 5. Learning 6. Lean Production	A soft-structured framework combines the principles of agile and conventionnal software development to address the issue of rapidly changing requirement	A learning experience to examine the traits of a programmer to determine the effect on the adoption of an Agile methodology	Systematic adoption of agile practices to the development method of organizations by analyzing the particular situation factors
Unit of analysis	Team	Team	Organization	Team	Organization	Team	Team
Theory	A structured approach to adopting agile practices	Smooth transition from a collocated to a distributed scenario	Adopting agile methodologies in distributed software development	A framework to support the evaluation, adoption, and improvement of agile methods in practice	A soft-structured approach to accommodating change on engineering larger-scale systems	Inhibiting the adoption of agile as a learning experience and traits of the individual programmer	Experience-based approach for adopting agile practices
Human centricity approach	No	No	No	Yes	No	Yes	No
Knowledge-based approach	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Learning approach	No	No	No	Yes	Yes	Yes	No

Table 4 Overview of the current literature of framework to support Agile SD implementation using knowledge management and learning process approaches.

Author	**Chiniforooshan Esfahani et al. (2010)**	**Chiniforooshan Esfahani (2012)**	**Lee and Yong (2013)**	**Nikitina and Kajko-Mattsson (2014)**	**Hoda and Noble (2017)**	**Rasnacis and Berzisa (2016)**	**Pries-Heje and Baskerville (2017)**
Research approach	Deductive	Deductive	Deductive	Deductive	Inductive	Deductive	Inductive
Research strategies	Case study	Case study, experiment report, ethnography, survey	Case study	Survey	Grounded research	Case study	Case study
Framework	(GOSM) Goal-oriented social modeling techniques for depicting social aspects of agile methods	(SAAF) Strategic agile pre-adoption analysis framework process	(AFSP) Agile framework small project incorporates the four evaluations tools from the agile literature	(SMA) Software Method Adoption model as guidelines for software organizations to adopt new software methods or set of processes in a structured and disciplined manner	Dimensions of transitions as teams, managers, and organization	Adaptation and implementation of the agile PM methodology	(FTRA) Fragments, Technological, Rules, and Articulation to adopt agile in a discourse of fragmentation and articulation
Unit of analysis	Team	Team	Team	Organization	Team	Team	Organization
Theory	Goal-oriented social modeling	Pre-adoption analysis using empirical knowledge and strategic modeling	Agile software development framework in a small project environment	Guiding the adoption of software development methods	How the development team transition to agile practices	The adaptation and implementation of Agile Project Management Methodology	The use of translation theory and the FTRA framework to explain how Agile adaptation emerge continuously in a process
Human centricity approach	Yes	Yes	No	No	Yes	Yes	Yes
Knowledge-based approach	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Learning approach	No	No	Yes	Yes	Yes	No	Yes

among team members (Vidgen and Wang, 2009) and link human knowledge gained from interactions with others within the organization (Crawford et al. 2006). SSM-based AR represents the "actuality" point of view that acknowledges the experience-based knowledge methodology, which considers accommodating people's interest in the "actuality" rather than focusing on the system or structuring the existence in the "reality." The word "accommodation" refers to acknowledging different individual worldviews based on the sharing of "actuality." Therefore, the "accommodation" process can only be designed or shaped in SSM (Uchiyama, 2009). As Checkland and Poulter (2006, p. 17) emphasize, the "human situation is not only unique but changes through time and exhibits multiple conflicting worldviews. Action researcher is to enter a human situation, take part in its activity, and use that experience as the research object."

SSM provides accommodation function to reflect on feedback and adapt to different opinions, different experiences, and different genetic dispositions. This accommodation is essential for finding another version of a situation that would be acceptable to all and would be less problematic (Checkland and Poulter, 2006).

Lastly, the framework must support the exploration of the learning process to maintain or improve organizational performance based on the collective learning of each individual's tacit knowledge, following the team's discussion of its experiences and contrasting judgments (Jones et al. 2006). SSM facilitates a structured discussion of possible changes that could improve the problematic situation. In step 5 of SSM, the conceptual model should be used to debate the differences between the idealized models and reality. The model should represent and enable an organized and structured discussion pertaining to questions such as, "Who does what? How? When?" and consider how stakeholders perceive human activity (Checkland and Poulter, 2006). Uchiyama (2009) mentioned that during step 5, following the comparison of the conceptual model and reality model, the first learning occurs when a researcher bridges the gap between the conceptual model and the reality model. Uchiyama (2009) also mentioned that during step 7 of SSM, a researcher engages in the second learning process as the action plan is executed, and he/she might experience a gap between the plan and reality. Therefore, SSM-based AR is the right methodology to understand how to support the learning process during Agile implementation.

The research methodology using SSM is summarized in Table 5.

SSM-based AR in action. To give more context and to understand how SSM can support the exploration of the learning process to solve the problems of Agile SD implementation, the author provides a real case to validate the proposed methodology. One corporation has been striving to achieve the company's vision to be the leading financial institution that always uses the latest technology. Thus, it has initiated many strategies in products or technology enhancement to cater to its business needs. To deal with the situation, their IT department had launched a new software development life cycle (SDLC) framework, "Agile." After more than a year of implementing Agile SD, misunderstandings emerge regarding the application of Agile SD. Differences in understanding the Agile SD concept itself are observed. To prove the theoretical framework proposed in the section Existing Framework, the author tried to use SSM described in the following sub-sections based on the summary in Table 5.

Entering the situation considered problematical (unstructured problems). This first step is to gather information about the structure and processes through observation (e.g., informal observation or formal sampling procedures), secondary data (e.g., minutes of committee meetings), and informal interviews. Using Agile SD after informal observation has been validated by informal interviews. The head of business thinks that the IT team is unproductive compared to other lines of business (LoB). He/she thinks knowledge and communication gaps happen between the IT team and feels that the IT team does not know the business process, which leads to a silo effect. The head of department thinks that some roles should not be filled by the IT team. The product owner (PO)/scrum master (SM) thinks that minimal documentation leads to unclear business requirements. He/she feels that it is difficult to understand the business context/process because an effort is needed to recall the business requirements.

Expressing the problem situation (structured problems). Three types of analyses can be used to describe the current problematic situation in the real world. The rich pictures will draw attention to people or groups who could be seen as stake-holders in any human situation, and the first type of analysis, research intervention, always considers possible "problem owners" selected by the "problem solver" as the main source of ideas for "relevant systems" that might usefully be modeled. Based on the problem statement, the following elements are described:

1. Client: promoter, co-promoter, and the university (doctoral science in management ITB program).
2. Practitioners: researcher, promoter, and co-promoter
3. Owners of the issues addressed: head of business, head of department, and PO/SM.

Figure 1 depicts the structured problems mentioned.

Formulate the root definitions of the relevant systems of purposeful activity. A root definition is a definition of the purpose of the system of human activity based on the CATWOE framework. Different from those conducted in step 1 and step 2 of SSM-based AR, which is a cultural stream of analysis, an analysis conducted in step 3 and step 4 of SSM-based AR is a logic-based stream of thinking (Peter and Scholes, 1990). The "PQR" formula means do P by Q to achieve R. This formula helps us understand the what, how, and why, of a transformational process that is captured in Q that declares the how.

It is important to define a transformation process (T) and worldview (W) that mean something, require people (A) to do the activities that make up the transformation (T) that effect people who are beneficiaries or victims of the process (C), environment (E) will provide various constraints as given when doing T, and someone could stop or change this T as the owner of the process (O). The (T) transformation process should be able to measure to sharpen up the thinking of modeling the purposeful activity model, by the following three criteria:

1. Efficacy, whether the transformation is working as intended outcome;
2. Efficiency, whether the transformation is achieved with minimum used resources;
3. Effectiveness, whether the transformation is supporting the achievement of higher-level or longer-term objective.

Table 6 shows the Root definition 1 is a system owned and operated by the SM, PO, and software developer to improve the working process (P) through the productive and optimal task completion (Q) to ensure comparative output that gives more values to the business (R).

Table 7 shows the Root definition 2 is system owned and operated by PO to align PO (P) through a specific forum (Q) to ensure that the organization of knowledge is disseminated properly and achieves organization performance (R).

Table 8 shows the Root definition 3 is system owned and operated by the head of department, PO, and software developer

Table 5 Summary of SSM relates to data collection, analysis, and output.

No	SSM steps	Description	Data collection	Data processing	Output
1	Enter situation considered problematical (*unstructured problems*)	Researcher collects information related to the problematic situation via primary and secondary data	Observation, interview, and literature study	Real-world situation problem analysis	Finding out the problem situation
2	Express the problem situation (*structured problems*)	Researcher structures the problematic situation systematically	Interview and discussion	Intervention analysis, social analysis, and political analysis	Structured problem via Rich picture
3	Formulate root definitions of relevant systems of purposeful activity	The researcher defines root definitions from various problems describing the human activity system. Root definitions address "what," "how," "why" of the human activity system.	Interview and discussion	Using PQR and CATWOE analysis	The generic purposeful activity model
4	Build conceptual models of the systems named in the root definitions	Researcher develops a conceptual model based on root definitions, PQR analysis, CATWOE analysis, and 3E (*Efficacy*, *Efficiency*, and *Effectiveness*).	Interview and discussion	Define three (3) groups of activities and connect the activities to indicate the dependency upon another. Add monitoring and controlling activities	Purposeful activity model
5	Comparing models with real-world situations	Researcher compares the conceptual model and real-world model using a matrix table as a comparison tool to generate a discussion about changes for improvement.	Interview and discussion	Discussing and debating the problem for accommodation purposes	Accommodation of different worldviews
6	Define possible changes which are both desirable and feasible	Researcher analyses and interprets a problematic situation based on step 5 comparison matric.	Interview and discussion	Summarizing the analysis from step 1 to 5 to define the action to improve	List of possible action for desirable and feasible changes
7	Take action to improve the problem situation	Researcher takes action to improve	Observation	Conscious critical reflection	Improvement and residual of the problematic situation after improvement

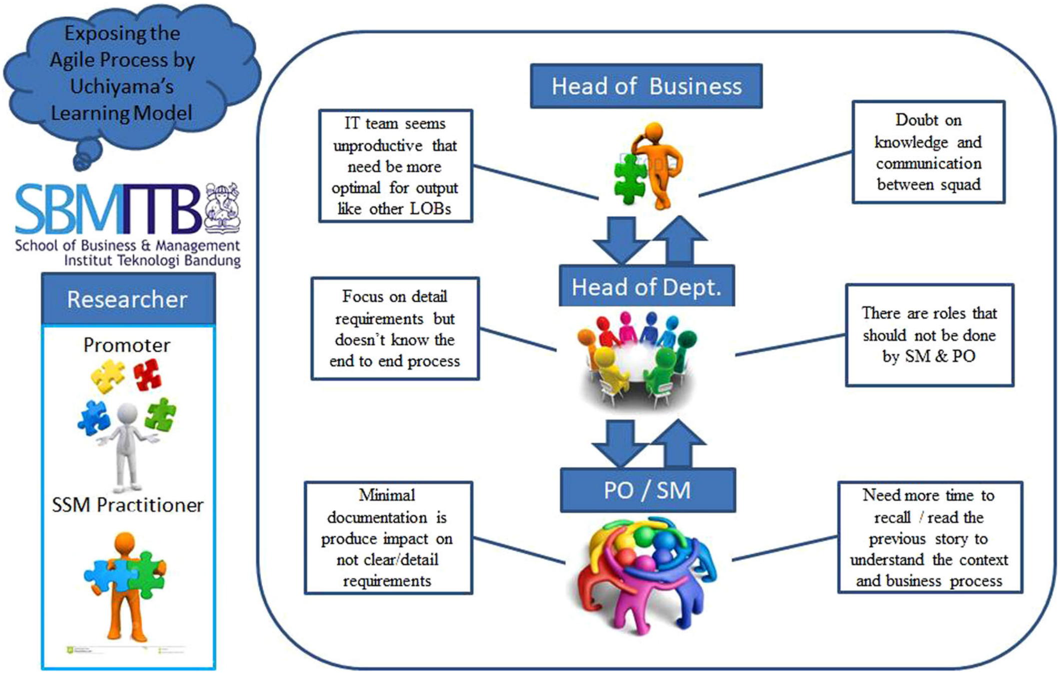
Exposing the Agile Process by Uchiyama's Learning Model
SBMITB
School of Business & Management Institut Teknologi Bandung
Researcher
Promoter
SSM Practitioner
Head of Business
IT team seems unproductive that need be more optimal for output like other LOBs
Doubt on knowledge and communication between squad
Head of Dept.
Focus on detail requirements but doesn't know the end to end process
There are roles that should not be done by SM & PO
PO / SM
Minimal documentation is produce impact on not clear/detail requirements
Need more time to recall / read the previous story to understand the context and business process

Fig. 1 Rich picture. The rich pictures will draw attention to people or groups who could be seen as stake-holders in any human situation.

Table 6 CATWOE—Root Definition 1.

CATWOE	Values
Customer	Head of Business, Head of Department
Actors	Product Owner, Scrum Master, Software Developer
Transformation	Creating a productive and optimal working process for the team
Weltanschhaung	Productive and optimal are seen by management as a comparative output that giving more values to the business
Owner	Head of Business, Head of Department
Environment	Working process is not executed based on Agile framework/values
Efficacy	Optimal working process is accepted by the team to perform and by business to agree
Efficiency	Using minimum resources (financial and time)
Effectiveness	Optimal working process as an output that giving more values to business

Table 7 CATWOE—Root Definition 2.

CATWOE	Values
Customer	Head of Business, Head of Department
Actors	Product Owner
Transformation	Product owner to create a specific forum to discuss as alignment between them
Weltanschhaung	Knowledge transfer and coordination is needed to ensure organization knowledge is disseminating properly for organization performance and values
Owner	Head of Business, Head of Department
Environment	Product Owner is not disciplined to attend forum
Efficacy	A specific forum supports the alignment to ensure organization knowledge dissemination
Efficiency	Using minimum resources (financial and time)
Effectiveness	Knowledge dissemination supports the achievement of organization performance and values

to understand the business requirement (P) through frequent interaction and collaborations (Q) to understand the end-to-end process (R).

Table 9 shows the Root definition 4 is system owned and operated by SM and PO to improve the roles and responsibilities (P) through defining clear roles and responsibilities

Q to specify who is doing what and what is to be done to achieve higher individual performance (R).

Table 10 shows the Root definition 5 is system owned and operated by PO to improve the quality of software requirement (P) through learning to create clear requirements with good quality (Q) to produce working software instead of minimal documentation (R).

Table 11 shows the Root definition 6 is System owned and operated by the dead of department, PO, SM, and software developer to create a process/container that is easy to store and to retrieve information (P) through an efficient and effective method

Table 8 CATWOE—Root Definition 3.

CATWOE	Values
Customer	Head of Department, Product Owner, Software Developer
Actors	Product Owner
Transformation	Understand the business requirements by more interaction and collaboration
Weltanschhaung	In Agile software development a short iteration is expected. Therefore to understand business requirements, the more interactions and collaboration between team are needed to understand end-to end process
Owner	Head of Department, Product Owner
Environment	Working process is not executed based on Agile framework/values
Efficacy	Product Owner able to understand business requirements
Efficiency	Using minimum resources (financial and time)
Effectiveness	Product owner able to understand end-to end process while review business requirements

Table 9 CATWOE—Root Definition 4.

CATWOE	Values
Customer	Head of Department, Product Owner, Scrum Master
Actors	Scrum Master, Product Owner
Transformation	Define the clear roles and responsibilities for PO and SM
Weltanschhaung	Clear roles and responsibilities are needed to ensure employees knows who is doing what and what needs to be done, to achieve higher individual performance
Owner	Head of Department
Environment	Job Description that not aligned with the SM & PO roles The head of department has additional task for PO that not include in role
Efficacy	Clear roles and responsibilities which can be used and executed by PO and SM
Efficiency	Using minimum resources (financial and time)
Effectiveness	Clear roles and responsibilities to achieve higher individual performance

Table 10 CATWOE—Root Definition 5.

CATWOE	Values
Customer	Head of Department, Product Owner, Software Developer
Actors	Product Owner
Transformation	Understand how to create clear requirements with good quality
Weltanschhaung	In Agile software development, working software is expected instead of comprehensive documentation. The importance of a clear requirement to produce working software is seen as a best way than minimal documentation approach
Owner	Head of Business, Head of Department
Environment	Using another documentation tools that not supported. Working process is not executed based on Agile framework/value
Efficacy	Product Owner able to create clear requirements
Efficiency	Using minimum resources (financial and time)
Effectiveness	Clear requirements ensure working software, bug/defect is reducing

Table 11 CATWOE—Root Definition 6.

CATWOE	Values
Customer	Head of Department, Product Owner, Scrum Master, Software Developer
Actors	Scrum Master, Product Owner, Software Developer
Transformation	Create a process/container that enable team to store and retrieve information easily
Weltanschhaung	In Agile software development, the most efficient and effective method of conveying information to and within a development team is face-to-face conversation. Therefore, need to create an efficient and effective method of storing and retrieving information from face-to face communication
Owner	Head of Department, Product Owner, Head of IT
Environment	Working process is not execute based on Agile framework/values
Efficacy	Process/container that enable team to store and retrieve information are created
Efficiency	Using minimum resources (financial and time)
Effectiveness	Efficient and effective method of storing and retrieving information for face-to face communication

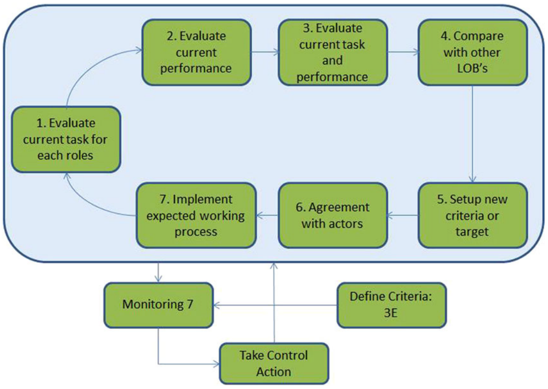
2. Evaluate current performance
3. Evaluate current task and performance
4. Compare with other LOB's
1. Evaluate current task for each roles
7. Implement expected working process
6. Agreement with actors
5. Setup new criteria or target
Monitoring 7
Define Criteria: 3E
Take Control Action

Fig. 2 Conceptual Model 1—Optimal working process in Squad. The modeling of relevant purposeful activity systems based on the statement describing the activity system in the root definition.

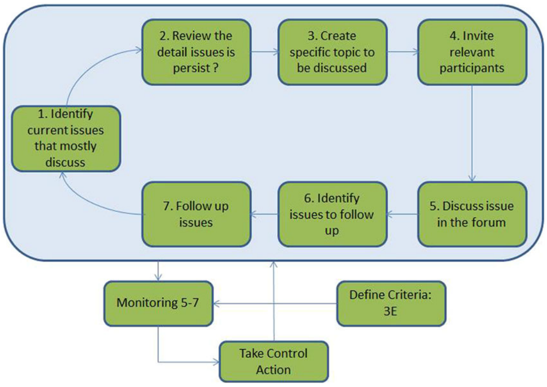
2. Review the detail issues is persist ?
3. Create specific topic to be discussed
4. Invite relevant participants
1. Identify current issues that mostly discuss
7. Follow up issues
6. Identify issues to follow up
5. Discuss issue in the forum
Monitoring 5-7
Define Criteria: 3E
Take Control Action

Fig. 3 Conceptual Model 2—Improve alignment between PO. The modeling of relevant purposeful activity systems based on the statement describing the activity system in the root definition.

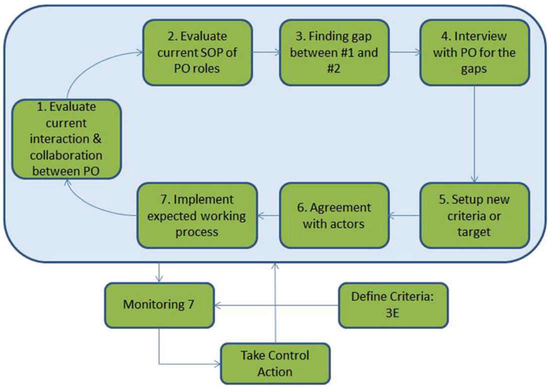
2. Evaluate current SOP of PO roles
3. Finding gap between #1 and #2
4. Interview with PO for the gaps
1. Evaluate current interaction & collaboration between PO
7. Implement expected working process
6. Agreement with actors
5. Setup new criteria or target
Monitoring 7
Define Criteria: 3E
Take Control Action

Fig. 4 Conceptual Model 3—Improvement of interaction and collaboration. The modeling of relevant purposeful activity systems based on the statement describing the activity system in the root definition.

that supports face-to-face communication (Q) to achieve an efficient and effective recall process (R).

Build conceptual models of the systems named in the root definitions. Step 4 of SSM is the modeling of relevant purposeful activity systems based on the statement describing the activity system in the root definition (RD) in step 3. The model is built according to declared single worldview by looking at complex reality, which exists only as a device the job of which is to ensure the learning process is organized, not just random and recoverable. The

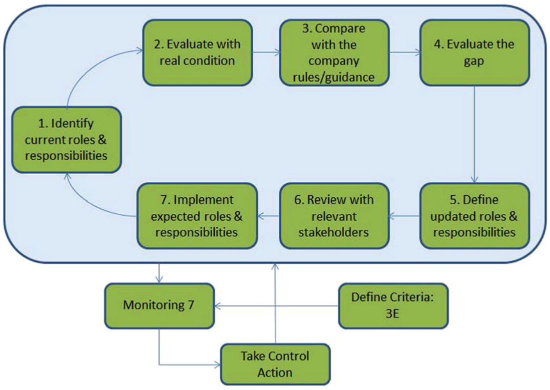
2. Evaluate with real condition
3. Compare with the company rules/guidance
4. Evaluate the gap
1. Identify current roles & responsibilities
7. Implement expected roles & responsibilities
6. Review with relevant stakeholders
5. Define updated roles & responsibilities
Monitoring 7
Define Criteria: 3E
Take Control Action

Fig. 5 Conceptual Model 4—Setup PO roles and responsibilities. The modeling of relevant purposeful activity systems based on the statement describing the activity system in the root definition.

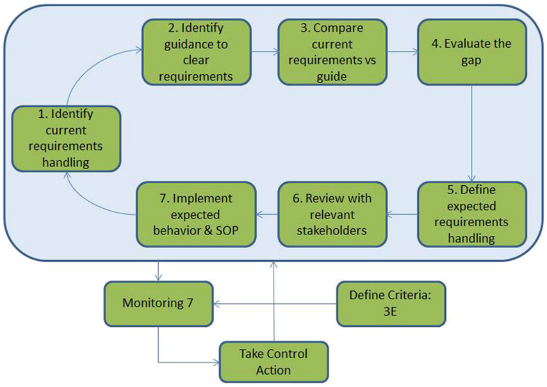
2. Identify guidance to clear requirements
3. Compare current requirements vs guide
4. Evaluate the gap
1. Identify current requirements handling
7. Implement expected behavior & SOP
6. Review with relevant stakeholders
5. Define expected requirements handling
Monitoring 7
Define Criteria: 3E
Take Control Action

Fig. 6 Conceptual Model 5—Improve quality of requirements. The modeling of relevant purposeful activity systems based on the statement describing the activity system in the root definition.

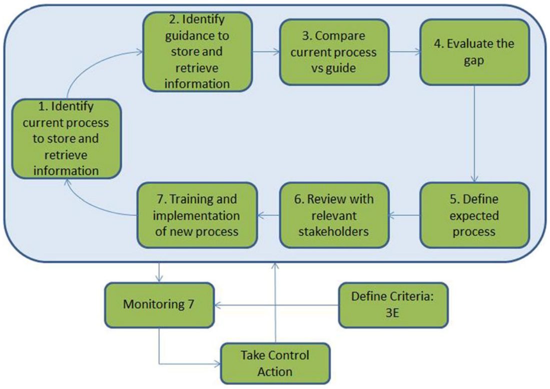
2. Identify guidance to store and retrieve information
3. Compare current process vs guide
4. Evaluate the gap
1. Identify current process to store and retrieve information
7. Training and implementation of new process
6. Review with relevant stakeholders
5. Define expected process
Monitoring 7
Define Criteria: 3E
Take Control Action

Fig. 7 Conceptual Model 6—Enable process to store and retrieve information easily. The modeling of relevant purposeful activity systems based on the statement describing the activity system in the root definition.

following figures are the conceptual models of the systems based on the root definitions stated in the previous section.

Figure 2 describes the optimal working process in the squad. The expected transformation is creating a productive and optimal working process for the team.

Figure 3 describes how to improve the alignment between PO. The expected transformation is to create a specific forum to discuss as alignment between the POs.

Figure 4 describes the improvements in interaction and collaboration. The expected transformation is to understand the

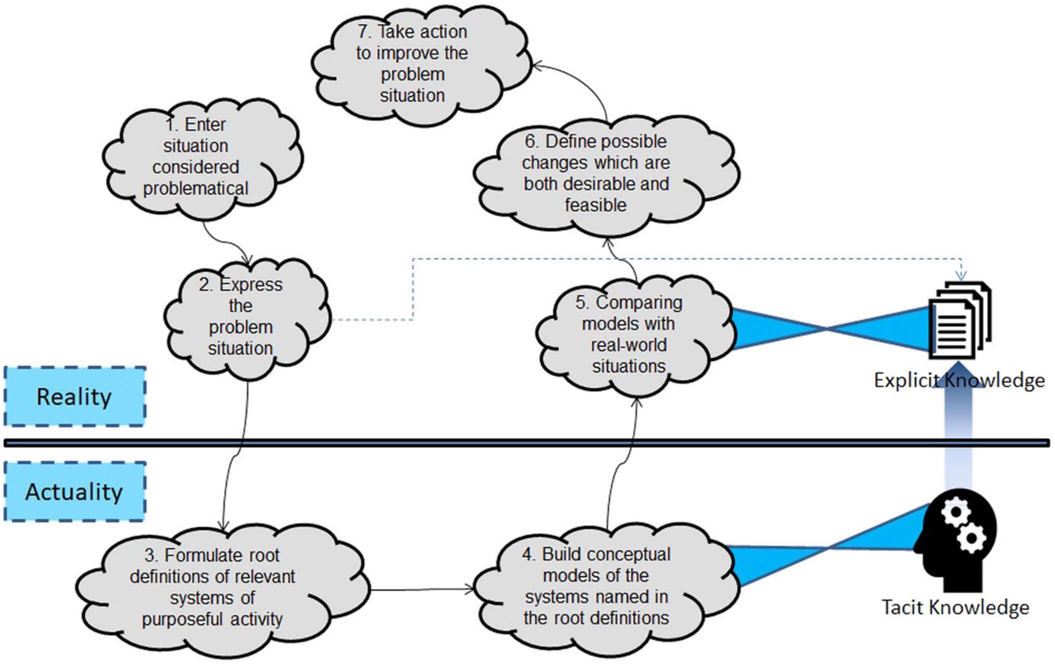
7. Take action to improve the problem situation
1. Enter situation considered problematical
6. Define possible changes which are both desirable and feasible
2. Express the problem situation
5. Comparing models with real-world situations
Explicit Knowledge
Reality
Actuality
3. Formulate root definitions of relevant systems of purposeful activity
4. Build conceptual models of the systems named in the root definitions
Tacit Knowledge

Fig. 8 First learning during step 5 of SSM. The first learning occurs when a researcher bridges the gap between the conceptual model and the reality model.

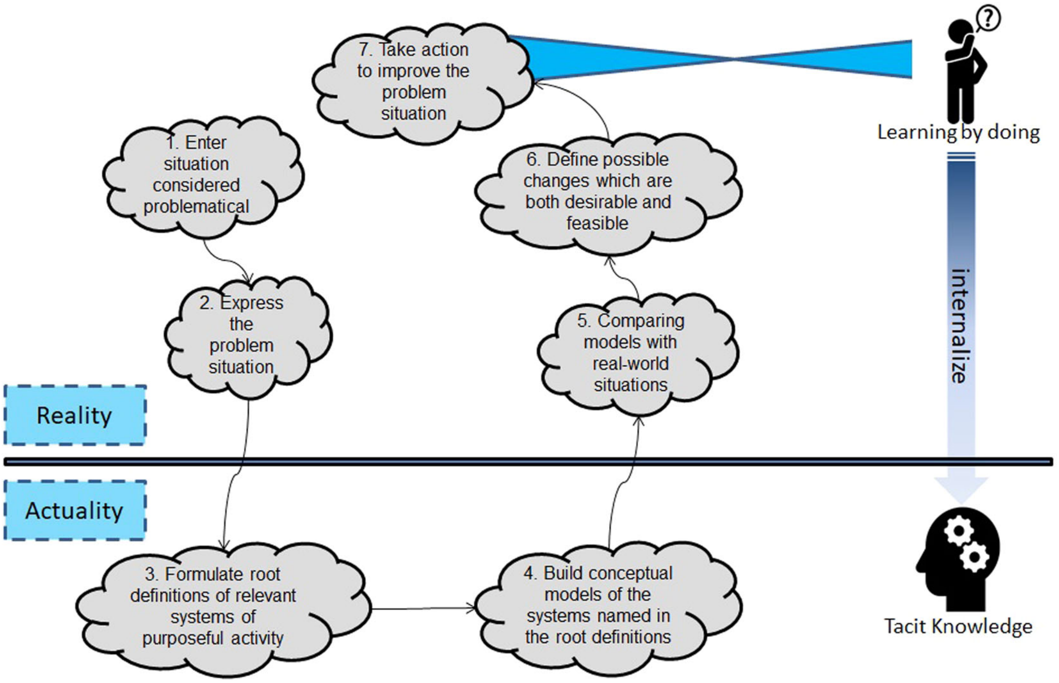
7. Take action to improve the problem situation
Learning by doing
1. Enter situation considered problematical
6. Define possible changes which are both desirable and feasible
2. Express the problem situation
5. Comparing models with real-world situations
internalize
Reality
Actuality
3. Formulate root definitions of relevant systems of purposeful activity
4. Build conceptual models of the systems named in the root definitions
Tacit Knowledge

Fig. 9 Second learning during step 7 of SSM. This second learning mode involves the internalization of tacit knowledge obtained through learning by doing.

business requirements by more interaction and collaboration between teams.

Figure 5 describes how to set up roles and responsibilities. The expected transformation is to define the clear roles and responsibilities of PO and SM.

Figure 6 describes how to improve the quality of requirements. The expected transformation is to understand how to create clear requirements with good quality.

Figure 7 describes how to enable the process of storing and retrieving information easily. The expected transformation is to create a process/container that enables the team to store and retrieve information easily.

Comparing models with a real-world situation. Step 5 is to generate a structured discussion about the situation of possible changes that could improve the problem situation. The conceptual model described in step 4 explains the differences between the idealized models and reality. The model should represent and enable organized and structured discussions, such as “Who does it? How? When?” about the human activity as perceived by

Table 12 Exposing the issues of Agile SD implementation by Uchiyama's learning model.

Root definition	Description	Step 5	Learning process
1	System owned and operated by Scrum Master, Product Owner, and software developer to improve the working process (P) through the productive and optimal task completion (Q) to ensure comparative output that gives more values to the business (R)	Rich picture (Step 2): IT team seems unproductive and needs to be more optimal for output like other LOBs. Conceptual model (Step 4): Creating a productive and optimal working process for the team	The team has learned that comparison with other LOB's (Line of Business) cannot be done since different LOBs have different objectives, strategies, and systems
2	System owned and operated by Product Owner to align Product Owner (P) through a specific forum (Q) to ensure that the organization of knowledge is disseminated properly and achieves organization performance (R)	Rich picture (Step 2): The Head of Business doubts knowledge transfer and coordination between PO. Conceptual model (Step 4): Product owner creates a specific forum to discuss agreement within the team	The team has learned that the coordination between PO is exist but there was no specific forum to discipline them to share information
3	System owned and operated by the Head of Department, Product Owner, and Software Developer to understand the business requirement (P) through frequent interaction and collaboration (Q) to understand the end-to-end process (R)	Rich picture (Step 2): The Head of Dept. feels that PO focuses on detail requirements but does not know the end-to end process Conceptual model (Step 4): Product owner understands the business requirements through more interaction and collaboration	The team has learned that PO feels the business has less interaction and collaboration with IT. This creates knowledge gap of end-to end process
4	System owned and operated by Scrum Master and Product Owner to improve the roles and responsibilities (P) through clarifying roles and responsibilities (Q) to specify who is doing what and what needs to be done to achieve higher individual performance (R)	Rich picture (Step 2): SM and PO should not play certain roles Conceptual model (Step 4): Define the clear roles and responsibilities for PO and SM	The team has learned that there were no clear rules/guidelines for PO to follow
5	System owned and operated by Product Owner to improve the quality of requirement (P) through understanding how to create clear requirements with good quality (Q) to produce working software instead of minimal documentation (R)	Rich picture (Step 2): Minimal documentation leads to unclear requirements Conceptual model (Step 4): Understand how to create clear requirements with good quality	The team has learned that there was a misunderstanding that using Agile SD can lead to minimal documentation
6	System owned and operated by Head of Department, Product Owner, Scrum Master, and Software Developer to create a process/container that allows for easy storage and retrieval of information (P) through an efficient method that supports face-to face communication (Q) to achieve efficient and effective recall process (R)	Rich picture (Step 2): Need more time to recall/read the previous story to understand the context and business process Conceptual model (Step 4): Create a process/container that enables the team to store and retrieve information easily	The team has learned that using current documentation tools is not enough to support face-to-face communication

Table 13 Recommendation for possible action.

Root definition	Desirable	Feasible	Possible action
1: Optimal working process in Squad	Y	Y	To define, agree, and execute the expected working process
2: Improve alignment between PO	Y	Y	Create PO Forum to specifically discuss current issues to be solved
3: Improvement of interaction and collaboration	Y	Y	To strengthen the interaction and collaboration in Agile scrum
4: Setup PO roles & responsibilities	Y	Y	To define, agree, execute, and frequent socialization on PO roles
5: Improve quality of requirements	Y	Y	To define, agree, execute, and frequent socialization on requirements handling
6: Improve storing & retrieving information through face-to face communication	Y	N	The use of the existing tool should be enforced at every meeting since using a new tool to capture face-to-face communication is too costly

stakeholders. Step 5 is explained in the section Discussion as a process of discussion.

Discussion

As mentioned in the section Key concept, the objective of this study was to uncover the issues of Agile SD implementation and to explore the framework that supports the learning process during Agile implementation. Uchiyama (2009) mentions that two stages of learning are obtained by using SSM-based AR. The first, which occurs during step 5 of SSM, is based on the comparison of the conceptual model and the reality model, which clarifies the gap between the models.

The first learning mode involves exposing and converting tacit knowledge into explicit knowledge, as shown in Fig. 8.

The second occurs during step 7 of SSM, following the execution of an action plan, which might reveal a gap between the action plan and the reality model. This second learning mode involves the internalization of tacit knowledge obtained through learning by doing as shown in Fig. 9. The aim of this study is to discuss only the first learning mode since the research is currently still in progress. Thus, the study on second learning will be forthcoming in the future.

SSM-based AR aims to discover the real issues in Agile SD implementation, as explained in step 2, by expressing the problematic situation systematically. The framework is developed through epistemological learning using tacit knowledge in step 4 and using explicit knowledge in step 5. In step 5, the learning process is explored to improve the implementation process of Agile SD. The learning process occurs in step 5 upon comparing between step 2 (structured problem via rich picture) and step 4 (conceptual model). Table 12 depicts the issues in implementing Agile SD by using Uchiyama's learning model.

Conclusion and future research

This study contributes to the research and fills a gap in the empirical discussion of the learning process in Agile SD. The learning process is an essential factor in achieving better software development, as software development often relies on tacit knowledge (Ozer and Vogel, 2015). Unfortunately, tacit knowledge is mainly in the heads of individuals and exposed only when it needs to be reused (Nonaka and Takeuchi, 1995). It is mostly found in the form of informal communications (Chau et al., 2003; Gervasi et al., 2013), is difficult to articulate since it tends to reside within the heads of knowers, and is very difficult to externalize (Dalkir, 2011) and understand (Ozer and Vogel, 2015). As such, it is difficult for an organization to create, replicate, and share it due to its complex and abstract nature (Choo, 1996; van den Berg, 2013).

Referring to the section Discussion of the Uchiyama learning model, implementing Agile SD is problematic because of the need to convert the tacit knowledge into explicit knowledge. Which becomes the first question of this study, the needs of supporting framework to discover the actual issues in Agile SD implementation. SSM-based AR aims to discover the real issues in Agile SD implementation, namely solving the issues that occur when converting tacit knowledge into explicit knowledge. SSM-based AR answers the second aim of this study, which is to support the learning process during Agile SD implementation. Using SSM-based AR, the researcher defines an action based on the epistemological learning that develops through an interaction and experience embodied at the tacit level, supports team learning as a consequence of close interactions among team members, links human knowledge, gains from interactions with others within the organization, and supports the exploration of the learning process to maintain or improve organizational performance based on the collective learning of each individual's tacit knowledge. Therefore, SSM-based AR can be used as a framework that captures tacit knowledge, covers team learning, links knowledge that resides within each human inside the organization, and supports the exploration of the learning process to maintain or improve organizational performance.

Future research can be focused to determine the change by accommodating the interests of the actors involved, which is step 6 of SSM-based AR. Table 13 shows possible actions that are acceptable and culturally possible based on the root definition describes in the section Methodology.

This study, however, has not yet identified the second type of learning that occurs during step 7 (taking action), which is concerned with exposing and converting explicit knowledge into tacit knowledge. The authors suggest conducting an extensive research study to perform each step from step 1 to step 7.

Data availability

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to restrictions of their containing information that could compromise the privacy of research participants.

Received: 6 July 2019; Accepted: 17 December 2019;
Published online: 14 January 2020

References

Anquetil N et al. (2007) Software maintenance seen as a knowledge management issue. Inf Softw Technol 49(5):515–529. https://doi.org/10.1016/j.infsof.2006.07.007

Argyris C (2006) Teaching smart people how to learn. In: Gallos JV (Ed.), Organization development: a Jossey-Bass reader. Jossey-Bass, San Francisco, CA, USA

Argyris C, Schön DA (1978) Organizational learning: a theory of action perspective. Revista española de investigaciones sociológicas, 77–78, 1997, Reis, p 345–350

Argyris, C (1999) On organizational learning. Blackwell Business, Oxford; Malden, Mass

Avison D, Kock N, Malaurent J (2017) Action Research in Information Systems. Taylor & Francis

Avison D et al. (1999) To make academic research relevant, researchers should try out their theories with practitioners in real situations and real organizations, Retrieved January, 16, p 2008

Babb J, Hoda R, Nørbjerg J (2014) Embedding reflection and learning into agile software development, IEEE software. IEEE 31(4):51–57

Bain A (1998) Social defenses against organizational learning, Human relations. Sage Publications Sage CA, Thousand Oaks, CA, 51(3):413–429

Bennett JK, OBrien MJ (1994) The Building Blocks of the Learning Organization, Training. ERIC, 31(6), p 41

Berg D, The D, Vol C (1993) Expanding perceptions, possibilities and profits. The Journal for Quality and Participation. American Society for Quality 16(7), 6

Bierly PE, Hamalainen T (1995) Organizational learning and strategy. Scand J Manag 11(3):209–224

Boateng R (2011) Do organizations learn when employees learn: the link between individual and organizational learning, Development and Learning in Organizations: An International Journal. Emerald Group Publishing Limited, 25(6):6–9

Boehm B (2002) Get ready for agile methods, with care, Computer. IEEE (1):64–69

Boehm B, Turner R (2003) Observations on balancing discipline and agility In: Proceedings of the Agile development conference, ADC 2003, Salt Lake City, UT, USA, pp 32–39

Brix J (2017) Exploring knowledge creation processes as a source of organizational learning: A longitudinal case study of a public innovation project, Scandinavian Journal of Management. Elsevier 33(2):113–127

Brown JS, Duguid P (1991) Organizational learning and communities-of-practice: toward a unified view of working, learning, and innovation. Organ Sci 2 (1):40–57. https://doi.org/10.1287/orsc.2.1.40

Brown JS, Duguid P (1998) Organizing Knowledge. Calif Manag Rev 40(3):90–111. https://doi.org/10.2307/41165945

Chan C C A (2003) Examining the relationships between individual, team and organizational learning in an Australian hospital. Blackwell Publishing Ltd. Learning in Health and Social Care, New Jersey, US, pp 223–235

Chaos Report (2015) The Standish Group, 2015 Chaos Report. Retrieved from: https://www.infoq.com/articles/standish-chaos-2015

Chau T, Maurer F (2004) Knowledge Sharing in Agile Software Teams. Citeseer

Chau T, Maurer F, Melnik G (2003) Knowledge sharing: Agile methods vs. Tayloristic methods. In Proceedings of the workshop on enabling technologies: infrastructure for collaborative enterprises, WETICE, Linz, Austria, Austria, pp 302–307

Checkland P, Poulter J (2006) Learning for action: a short definitive account of soft systems methodology and its use for practitioner, teachers, and students, vol. 26. Wiley, Chichester

Chiniforooshan Esfahani H (2012) Transitioning to Agile: a framework for pre-adoption analysis using empirical knowledge and strategic modeling. Graduate Department of Computer Science University of Toronto. http://search.proquest.com/docview/1353360860?accountid=27231

Chiniforooshan Esfahani H, Cabot J, Yu E (2010) Adopting agile methods: can goal-oriented social modeling help? In: Yakhno T, Neuhold EJ (eds) Advances in Information Systems. ADVIS 2006 (Lecture notes in computer science), vol 4243. Springer, Berlin, Heidelberg, Nice, France, pp 223–234

Choo CW (1996) The knowing organization: how organizations use information to construct meaning, create. Int J Inf Manag 16(5):329–340

Coughlan P, Coghlan D (2010) Action research, in Researching operations management. Routledge, pp 250–278

Crawford B, Castro C, Monfroy E (2006) Knowledge management in different software development approaches. In: International conference on advances in information systems. Izmir, Turkey, pp 304–313

Cross KP (1981) Adults as learners. Increasing participation and facilitating learning. ERIC

Dalkir K (2011) Knowledge management in theory and practice. The MIT Press, Cambridge, Massachusetts, US

Dermol V (2013) Relationship between learning, knowledge creation and organizational performance. Scientific Annals of Economics and Business. 60 (1):67–82. https://doi.org/10.2478/aicue-2013-0008

Desouza KC (2003) Barriers to effective use of knowledge management systems in software engineering, Commun. ACM, New York, NY, USA, 46(1):99–101. https://doi.org/10.1145/602421.602458

Dingsøyr T, Smite D (2013) Managing knowledge in global software development projects, IT Professional. IEEE 16(1):22–29

Dingsøyr T et al. (2012) A decade of agile methodologies: Towards explaining agile software development. Elsevier

Dorairaj S, Noble J, Malik P (2012) Knowledge management in distributed agile software development, in 2012 Agile Conference, pp 64–73

Dixon NM (2017) The organizational learning cycle: How we can learn collectively. Routledge

Edmondson A (1999) Psychological safety and learning behavior in work teams. Adm Sci Q 44(2):350–383. https://doi.org/10.2307/2666999

Edmondson A, Dillon J, Roloff K (2007) Three Perspectives on Team Learning: Outcome Improvement, Task Mastery, and Group Process, Academy of Management Annals. https://doi.org/10.5465/078559811

Fan D-P, Kuang Y-Y (2013) A Study on Management Methodology for the Complexity of Social System—the Combination of MCDM and SSM, International Journal of Operations Research, 10(2):49–55

Fernández-Sanz L, Misra S (2011) Influence of human factors in software quality and productivity, in International Conference on Computational Science and Its Applications, pp 257–269

Fiol CM, Lyles MA (1985) Organizational learning Academy of Management Review, 10(4):803–813

Forcheri P, Molfino MT, Quarati A (2000) ICT driven individual learning: new opportunities and perspectives. Educ Technol Society JSTOR 3(1):51–61

Friedman VJ (2002) The Individual as Agent of Organizational Learning. California management review, 44(2)

Gartner WB (1985) A conceptual framework for describing the phenomenon of new venture creation, Academy of Management Review I, pp 696–706

Garavan TN, Carbery R (2012) Collective learning, Encyclopedia of the Sciences of Learning. Springer, pp 646–649

Garavan TN, McCarthy A (2008) Collective learning processes and human resource development, Advances in developing human resources. Sage Publications Sage CA, Los Angeles, CA, 10(4):451–471

Garvin DA (1993) Summary: Building a learning organization. Harv Bus Rev 71 (4):78–91. https://doi.org/10.1016/S0267-3649(00)88914-1

Gervasi V et al. (2013) Unpacking Tacit Knowledge for Requirements Engineering. Managing Requirements Knowledge, pp 23–47. https://doi.org/10.1007/978-3-642-34419-0_2

Ghobadi S, Mathiassen L (2014) Perceived barriers to effective knowledge sharing in agile software teams, Information Systems Journal. https://doi.org/10.1111/isj.12053

Goldkuhl G (2012) From action research to practice research. Australasian Journal of Information Systems 17(2)

Gregory P et al. (2014) The Agile Research Network. In: ECRM 2014, 13th European Conference on Research Methodology for Business and Management Studies. Bridging the gap between research and practice: the Agile Research Network. London, UK

Hanafizadeh P, Ghamkhari F (2019) Elicitation of Tacit Knowledge Using Soft Systems Methodology, Systemic Practice and Action Research. Springer, 32 (5):521–555

Hartmann T, Fischer M, Haymaker J (2009) Implementing information systems with project teams using ethnographic-action research, Advanced Engineering Informatics. Elsevier 23(1):57–67

Hayes J, Allinson CW (1998) Cognitive style and the theory and practice of individual and collective learning in organizations, Human relations. Sage Publications Sage CA, Thousand Oaks, CA, 51(7):847–871

Hecker A (2012) Knowledge Beyond the Individual? Making Sense of a Notion of Collective Knowledge in Organization Theory, Organization Studies 33 (3):423–445. https://doi.org/10.1177/0170840611433995

Heikkila T, Gerlak AK (2013) Building a conceptual approach to collective learning: Lessons for public policy scholars, Policy Studies Journal. Wiley Online Library 41(3):484–512

Hoda R, Noble J (2017) Becoming agile: a grounded theory of agile transitions in practice. In: Proceedings—2017 IEEE/ACM 39th international conference on software engineering, ICSE 2017, Buenos Aires, Argentina, pp 141–151

Hummel M, Rosenkranz C, Holten PR (2012) The Role of Communication in Agile Systems Development An Analysis of the State of the Art. https://doi.org/10.1007/s12599-013-0282-4

Hummel M, Rosenkranz C, Holten R (2015) The Role of Social Agile Practices for Direct and Indirect Communication in Information Systems Development Teams, CAIS, 36, p 15

Hurley B, Cunningham I (1993) Imbibing a new way of learning, Personnel management-London-. Personnel publications, 42

Huysman M, de Wit D (2002) Knowledge sharing in practice. Kluwer Academic Publishers, Norwell, MA, USA

Hyland T, Matlay H (1997) Small businesses, training needs and VET provision. J Educ Work 10(2):129–139. https://doi.org/10.1080/1363908970100202

Ikehara HT (1999) Implications of gestalt theory and practice for the learning organization. Learn Organ 6(2):63–69. https://doi.org/10.1108/09696479910262587

Jackson MC (2003) Systems Thinking: Creative Holism for Managers

Jarvis P (1987) Meaningful and meaningless experience: towards an analysis of learning from life. Adult Educ Q 37(3): 164–172.

Joey Cho J (2010) An exploratory study on issues and challenges of agile software development with scrum. All Graduate Theses and Dissertations

Jones C et al. (2006) Collaborative learning with group interactive technology: a case study with postgraduate students. Manag Learn 37(3):377–396. https://doi.org/10.1177/1350507606067173

Kalermo J, Rissanen J (2002) Agile software development in theory and practice, University of Jyväskylä. http://1goal-educationforall.webs.com/-NewFolder/COMPUTERSOFTWAREENGINEERING.pdf

Kimble C (2013) Knowledge management, codification and tacit knowledge, Information Research 18(2):255–256

Kim DH (1993) The link between individual and organizational learning. Sloan Manage Rev 35(1):37–50

Kingston JKC (2012) Tacit knowledge: Capture, sharing, and unwritten assumptions, Journal of Knowledge Management Practice 13(3):533–545

King WR (2011) Knowledge transfer, in Encyclopedia of Knowledge Management, Second Edition. IGI Global, pp 967–976

Knapp R (2010) Collective (team) learning process models: a conceptual review. SAGE. Human Resource Development Review 9(3):285–299. https://doi.org/10.1177/1534484310371449

Knowles M S et al. (1998) The adult learner: the definitive classic in adult education and human resource development. Elsevier. https://books.google.com.sg/books?id=J6qGsHBj7nQC

Krasteva I, Ilieva S, Dimov A (2010) Experience-based approach for adoption of agile practices in software development projects. In: Pernici B (ed) Advanced information systems engineering. CAiSE 2010. Lecture notes in computer science, vol 6051. Springer, Berlin, Heidelberg, pp 266–280

OConnor SR, R. V (2004) Team Tacit Knowledge as a Predictor of Performance in Software Development Teams, Proceedings of the SixteenthInternational Conference on Software Engineering & Knowledge Engineering

Lee S, Yong HS (2013) Agile software development framework in a small project environment. J Inf Process Syst 9(1):69–88. https://doi.org/10.3745/JIPS.2013.9.1.069

Levine S J (2008) The challenge of helping adults learn: characteristics of adult learners & implications for teaching technical information. https://www.learnerassociates.net/workshop/adltlrn1.pdf

Levy M, Hazzan O (2009) Knowledge management in practice: The case of agile software development, in 2009 ICSE Workshop on Cooperative and Human Aspects on Software Engineering, pp 60–65

Li K et al. (2013) Analysis of the key factors for software quality in crowdsourcing development: An empirical study on topcoder. com, in 2013 IEEE 37th Annual Computer Software and Applications Conference, pp 812–817

Lindvall M, Rus I (2002) Knowledge management in software engineering, IEEE Software 19(3):26–38. http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Knowledge+management+in+software+engineering#0

Love PED et al. (2015) From individual to collective learning: A conceptual learning framework for enacting rework prevention, Journal of construction engineering and management. American Society of Civil Engineers 141 (11):5015009

Mckay J, Marshall P (2001) The dual imperatives of action research, Information Technology & People 14:46–59. https://doi.org/10.1108/09593840110384771

Mahroeian H, Forozia A (2012) Challenges in managing tacit knowledge: A study on difficulties in diffusion of tacit knowledge in organizations, International Journal of Business and Social Science. Centre for Promoting Ideas, USA, 3 (19)

Manuti A et al. (2015) Formal and informal learning in the workplace: a research review, International journal of training and development. Wiley Online Library, 19(1):1–17

Mannaro K (2008) Adopting Agile methodologies in distributed software development. Universita' degli Studi di Cagliari. http://www.diee.unica.it/DRIEI/tesi/19_mannaro.pdf

Matlay H (2000) Organisational learning in small learning organisations: an empirical overview. Educ Train 42(4/5):202–211. https://doi.org/10.1108/00400910010373642

Meline T (2006) Selecting studies for systematic review: Inclusion and exclusion criteria, Contemporary issues in communication science and disorders. ASHA 33(21–27)

McAvoy J, Butler T (2009) A failure to learn by software developers: inhibiting the adoption of an agile software development methodology. J Inf Technol Case Appl Res 11(1):23–46. https://doi.org/10.1080/15228053.2009.10856152

Miller KW, Larson DK (2005) Agile software development: human values and culture. IEEE Technol Soc Mag 24(4):36–42. https://doi.org/10.1109/MTAS.2005.1563500

Misra SC, Kumar V, Kumar U (2010) Identifying some critical changes required in adopting agile practices in traditional software development projects. Int J Qual Reliab Manag 27(4):451–474

Nevis EC, Ghoreishi S, Gould JM (1995) Understanding organizations as learning systems introduction core themes a model of organizations as learning systems general directions for enhancing learning references. Long Range Plan 28(3):119

Nikitina N, Kajko-Mattsson M (2014) Guiding the adoption of software development methods. ICSSP 2014: proceedings of the 2014 international conference on software and system process, (April) May 2014, p 109–118

Nonaka I, Takeuchi H (1995) The knowledge creating company: how Japanese companies create the dynamics of innovation. Oxford University Press, Inc.

Nonaka I, Takeuchi H (2007) The knowledge-creating company. Harv Bus Rev 85 (7/8):162

Nonaka I, Toyama R, Konno N (2000) SECI, Ba and leadership: a unified model of dynamic knowledge creation. Long Range Plan 33(1):5–34. https://doi.org/10.1016/S0024-6301(99)00115-6

Ozer M, Vogel D (2015) Contextualized relationship between knowledge sharing and performance in software development. J Manag Inf Syst 32(2):134–161. https://doi.org/10.1080/07421222.2015.1063287

Parrish A et al. (2004) A field study of developer pairs: productivity impacts and implications. IEEE Softw 21:76–79. https://doi.org/10.1109/MS.2004.1331306

Pauleen DJ, Gorman GE (2011) Personal knowledge management: individual, organizational and social perspectives. Ashgate Publishing Ltd, Farnham, http://cds.cern.ch/record/1412773

Peter C, Scholes J (1990) Soft systems methodology in action, Chicester, John Wiley & Sons Ltd

Polanyi M (1966) The tacit dimension. University of Chicago Press

Popper M, Lipshitz R (2000) Organizational learning mechanism, culture, and feasibility. Management Learning 31(2):181–196

Pries-Heje J, Baskerville R (2017) The translation and adaptation of agile methods: a discourse of fragmentation and articulation. Inf Technol People 30 (2):396–423. https://doi.org/10.1108/ITP-08-2013-0151

Qumer A, Henderson-Sellers B (2008) A framework to support the evaluation, adoption and improvement of agile methods in practice. J Syst Softw 81 (11):1899–1919. https://doi.org/10.1016/j.jss.2007.12.806

King WR (2008) An integrated architecture for an effective knowledge organization. J Knowl Manag 12:29–41. https://doi.org/10.1108/13673270810859497

Rasnacis A, Berzisa S (2016) Method for adaptation and implementation of Agile project management methodology. Proc Comput Sci. 104 (December): 43–50. https://doi.org/10.1016/j.procs.2017.01.055

Robertson F et al. (2017) Participatory action and duel lens research, Qualitative Research Journal, 17, p. 0. https://doi.org/10.1108/QRJ-12-2016-0075

Robillard PN (1999) The role of knowledge in software development, Commun. ACM, New York, NY, USA 42(1):87–92. https://doi.org/10.1145/291469.291476

Sadler-Smith E (2006) Learning and development for managers: Perspectives from Research and Practice. Wiley-Blackwell. http://epubs.surrey.ac.uk/816055/

Santos V, Goldman A, De Souza CRB (2015) Fostering effective inter-team knowledge sharing in agile software development, Empirical Software Engineering. Springer 20(4):1006–1051

Senge P (1990) Thefifth discipline: the art and science of the learning organization. Currency Doubleday, NewYork, NY

Sessa VI, London M (2008) Work group learning: understanding, improving &assessing how groups learn in organizations. L. Erlbaum Associates, NewYork, NY, http://lib.ugent.be/catalog/rug01:001338234

Shrivastava P (1983) A typology of organizational learning systems. J Manage Stud. https://doi.org/10.1111/j.1467-6486.1983.tb00195.x

Sidky A, Arthur J, Bohner S (2007) A disciplined approach to adopting agile practices: the Agile adoption framework. Innov Syst Softw Eng 3(3):203–216. https://doi.org/10.1007/s11334-007-0026-z

Soundararajan S, Arthur JD (2009) A soft-structured agile framework for larger scale systems development. Proceedings of the international symposium and workshop on engineering of computer based systems, San Francisco, CA, USA, p 187–195

Sureshchandra K, Shrinivasavadhani J (2008) Adopting agile in distributed development, Proceedings—2008 3rd IEEE international conference global software engineering, ICGSE 2008, Bangalore, India, p 217–221

Tanyaovalaksna S, Li X (2013) Is there a relationship between individual learning, team learning, and organizational learning? Alberta Journal of Educational Research 59(1):1–16

Terje Karlsen J, Hagman L, Pedersen T (2011) Intra-project transfer of knowledge in information systems development firms, Journal of Systems and Information Technology. Emerald Group Publishing Limited, 13 (1):66–80

Thompson CJ, Zondlo JA (1995) Building a case for team learning. Healthc Forum J 38(5):36–37, 39–40, 42–43

Tiwana A (2004) An empirical study of the effect of knowledge integration on software development performance. Inf Softw Technol 46(13):899–906. https://doi.org/10.1016/j.infsof.2004.03.006

Turk D, France R, Rumpe B (2005) Assumptions underlying agile software-development processes. J Database Manag 16(4):62–87. https://doi.org/10.4018/jdm.2005100104

Uchiyama K (2009) A concise theoretical grounding of action research: based on checkland's soft systems methodology and Kimura's phenomenological psychiatry. Institute of Business of Daito Bunka University, Japan

van den Berg HA (2013) Three shapes of organizational knowledge. J Knowl Manag 17(2):159–174. https://doi.org/10.1108/13673271311315141

Vidgen R, Wang X (2009) Coevolving systems and the organization of Agile software development. Inf Syst Res 355–376. https://doi.org/10.1287/isre.1090.0237

Von Krogh G (2012) How does social software change knowledge management? Toward a strategic research agenda, The Journal of Strategic Information Systems. Elsevier 21(2):154–164

Walz DB, Elam JJ, Curtis B (1993) Inside a Software Design Team: Knowledge Acquisition, Sharing, and Integration, Commun. ACM, New York, NY, USA 36(10):63–77. https://doi.org/10.1145/163430.163447

Wang C L, Ahmed P K (2003) Structure and structural dimensions for knowledge-based organizations. Meas Bus Excellence 7(1), 51–62. https://doi.org/10.1108/13683040310466726

West D et al. (2010) Agile development: mainstream adoption has changed agility. Forrester research. http://www.forrester.com/Agile+Development+Mainstream+Adoption+Has+Changed+Agility/fulltext/-/E-RES56100

Wieringa R, Morali A (2012) Technical action research as a validation method in information systems design science, in International Conference on Design Science Research in Information Systems, pp 220–238

Yao-Sheng L (2007) The effects of knowledge management strategy and organization structure on innovation. Int J Manag Int J Manag 24(1):53

Yu X, Petter S (2014) Understanding agile software development practices using shared mental models theory. Inf Softw Technol 56(8):911–921

Acknowledgements

The research work for this study was initiated in a Banking JV corporation by the approval of the Head of Business. The Research Question, Literature Review, and Research Methodology were formulated together with my supervisor, Prof. Dr. Dermawan Wibisono, Achmad Ghazali, Ph.D., Dr. Rachma Fitriati to support the extensive guidance during the process.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to K.S.