International Journal on Advanced Science Engineering Information Technology

Vol.7 (2017) No. 2
ISSN: 2088-5334

Key Factors for Selecting an Agile Method: A Systematic Literature Review

Mashal Alqudah#1, Rozilawati Razali#2

#Research Center for Software Technology and Management, Faculty of Information Science and Technology,
Universiti Kebangsaan Malaysia, 43600, Bangi, Selangor, Malaysia
E-mail: 1mashal_alqudah@yahoo.com, 2rozilawati@ukm.edu.my

***Abstract*— Agile methods have become popular in recent years because the success rate of project development using Agile methods is better than structured design methods. Nevertheless, less than 50 percent of projects implemented using Agile methods are considered successful, and selecting the wrong Agile method is one of the reasons for project failure. Selecting the most appropriate Agile method is a challenging task because there are so many to choose from. In addition, potential adopters believe that migrating to an Agile method involves taking a drastic risk. Therefore, to assist project managers and other decision makers, this study aims to identify the key factors that should be considered when selecting an appropriate Agile method. A systematic literature review was performed to elicit these factors in an unbiased manner, and then content analysis was used to analyze the resultant data. It was found that the nature of the project, development team skills, project constraints, customer involvement and organizational culture are the key factors that should guide decision makers in the selection of an appropriate Agile method based on the value these factors have for different organizations and/or different projects.**

***Keywords*— Agile method selection; factors of selecting Agile methods; SLR**

I. INTRODUCTION

Many companies want to implement good-quality systems, and they want to do it quickly, at a reasonable cost and within a specific scope [1], [2]. Therefore, many companies have started to shift to so-called Agile methods to develop such systems [3], and it has been found that the extent of the organizational team's skills [1], [4], [5], culture of the organization [2], [4], nature of project [5], and project constraints [5] must be given in-depth consideration [6] when selecting an Agile method. However, selecting the most appropriate Agile method for a project based on organizational capabilities is a difficult task [7], [8] because there are many Agile methods to choose from, such as Extreme Programming (XP) [9], [10], SCRUM [11], Test-Driven Development (TDD) [12], Feature-Driven Development (FDD) [13], Lean Development (LD) [14], Adaptive Software Development (ASD) [15], Agile Modelling (AM) [16], Crystal Family Group (Clear, Red, Orange, Yellow) [17], Internet Speed Development (ISD) [18], [19], Disciplined Agile Delivery (DAD) [6], Agile Software Process Development [20], DevOps [21], Kanban [22] and the Dynamic Systems Development Method (DSDM) [23]. Moreover, many firms around the world are still unaware of the importance of employing a suitable Agile method, while others may be aware of some of these methods but do not have the courage or the confidence to migrate to these methods [4], [24]. As a result, despite the increasing use of Agile methods in software development projects, some firms believe that migrating to Agile methods involves a drastic risk [25]. Further, there has been little research conducted to comprehend the difficulties associated with the selection of Agile methods [25].

Therefore, the main aim of this study is to identify the key factors that need to be considered when selecting Agile methods. This will assist decision makers in selecting the most suitable Agile methods for different project conditions in order to ensure a successful outcome. To this end, a systematic literature review (SLR) was undertaken to find and evaluate the relevant research studies in order to identify the factors that may influence the selection of Agile methods [26], [27].

The rest of this paper is organized as follows: Section II provides the background to Agile methods and some of the techniques that have been adopted for their selection. In addition, it describes the research methodology that was employed in this study to identify the crucial factors for selecting Agile methods. Next, Section III presents the results obtained from the SLR of the relevant works, and it discusses the factors that were identified. Finally, Section IV provides a conclusion and defines the scope of further work that will be undertaken on this topic.

II. Material and method

This section provides a brief background to the emergence of Agile methods as well as the main techniques that have been used to select Agile methods. In the early days of system development during the 1950s and 1960s, Code-and-Fix and Stagewise methods were used [28] and then, in 1970, Structured Design Methods (SDMs) were introduced which enhanced these methods [29]. In the following decade, Rapid Application Development (RAD) emerged to address the weaknesses of SDMs [30], [31]. Then, in the late 1990s to early 2000s, Agile methods emerged to facilitate the early development of high-quality software products [23]. The chronological evolution of these systems development life cycle (SDLC) methods is illustrated in Fig. 1.

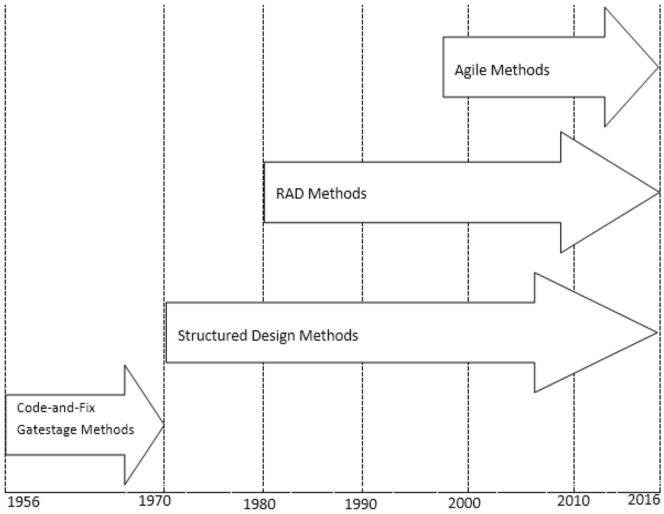

Fig. 1 Chronology of SDLC methods

Then, in 2001, a group of 17 software developers (members of the Agile Alliance) collaborated to enhance and explore the new Agile development methods. They proposed Agile methods based on some specific values and principles under the so-called Agile Manifesto [32]. The four values are defined as follows: “Individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan” [32]. Also, to help developers and project managers understand Agile methods and apply their values, the Alliance refined the values captured in their manifesto into 12 principles, the following are the examples of the principles [32]:

1) *Stakeholders (customers and users) and developers must work together daily throughout the project.*

2) *Build projects around motivated individuals.*

3) *Face-to-face conversation is the most effective and efficient method of conveying information to and within a delivery and development team.*

4) *Working software (solutions) which quantify the business value is the best way to measure progress.*

5) *Sustainable delivery and development by maintaining a constant pace.*

Over the years, a range of practices has been developed for use with different Agile methods to cover and achieve the above principles [19]. For instance, active stakeholder participation, the planning game, and the sprint planning meeting are practices used in different Agile methods (AM, XP, and SCRUM, respectively) that all focus on the principle of having face-to-face conversations (principle 3 above). On the other hand, daily meeting (SCRUM), whole team (AM), and pair programming (XP) focus more on working together daily throughout the project (principle 1 above) [33], while, 40-hour weeks is an XP practice that focuses on maintaining a constant pace until the end of project development (principle 5 above).

Currently, there are numerous Agile methods obtainable. Therefore, it is difficult to select the most appropriate one in order to allow the correct practices to be adopted, especially for developing large, critical or complex projects [34]. Moreover, organizations cannot follow the same method to implement all of their projects [16], [17], [35]-[37] because of the different requirements of each project. In addition, some Agile methods are not covering all the stages of the development, they are considered as partial methods that require significant tailoring to make them work in practice. This includes the Crystal family of methods [17], AM [16] and, to a lesser extent, XP [10] and FDD [38]. On the other hand, some methods such as DSDM [23] and ISD [18] are more complete and require less tailoring. Obviously, a way needs to be found to ensure that the most appropriate Agile method is selected [39]-[41] for a specific project. Therefore, researchers and practitioners in the Agile methods domain have started to investigate various techniques to select the most suitable Agile method for a particular project. Table 1 below shows the different selection techniques that have been proposed so far.

Previous works have demonstrated that there is a need for systematic identification of the crucial factors for selecting an appropriate Agile method. However, even though several studies have investigated the selection of Agile methods, the researchers are not aware of any SLR that has been published to identify the factors involved in the selection of Agile methods. Therefore, the main objective of this study is to undertake an SLR to precisely identify the crucial factors for selecting appropriate Agile methods. The main aim of undertaking an SLR is to identify the available research studies relevant to a particular research question [26, 27]. As will be explained below, a total of 1073 studies were found to be related to the search string for the SLR. However, only 53 were reviewed because only these were found to contribute to answering the research question. The SLR consisted of three stages: planning, performance, and reporting. The first two stages are explained in this section, while the reporting stage is covered in Section III.

TABLE I
TECHNIQUES FOR SELECTING AN AGILE METHOD

Selection technique	Description of technique (based on factors considered)
Alistair Crystal Family [17]	In this work, it is argued that the size and criticality of the project are the main factors that could affect the selection of an Agile method (see also Section III).
The Radar Chart [42]	In this work, the researchers propose a radar chart technique to confirm the suitability of Agile methods and the following factors were considered: high team skill personality, nature of the project in terms of dramatic changes, less criticality and small team size (small project). Else, traditional approaches are more suitable. The researchers state that a flexible organizational culture is also needed when selecting Agile methods [42]. However, the radar chart is provided to confirm whether Agile methods are suitable or not and it does not determine which Agile methods should be used [43].
Informal Selection [40]	In this work, the researchers propose an informal tactic for selecting the most suitable Agile or non-Agile methods [40]. The researchers state that in order for project managers to select a suitable method, they need to have a deep understanding of both types of methods and anticipate the problems that might occur in each. Thus, the researcher suggests empirical validation and investigation of the selection factors which might affect a specific category of SDLC methods [40].
The Slider [42]	In this work, the researchers propose a technique called the Slider to consider four selection criteria when any organization decides to migrate to Agile methods [42]. The researchers suggest that in cases of project uncertainty or instability, high customer responsiveness, the existence of an innovative culture, and less criticality, Agile methods would be better than traditional methods. Else, traditional methods would be more suitable, or there may be a need to adopt some traditional practices [42]. On the other hand, a related work states that in the case of mixed results a combination of Agile methods and traditional methods is required [43]. However, this method of selection does not show which Agile methods are more suitable [43].
Selection of Practices in Agile Methods [41]	In this work, the researcher focuses on evaluating which practices to adopt by taking into account the project constraints and the nature of the project. When project managers tested the provided selection method, they suggested that additional criteria could be incorporated for accurate decision making in selecting Agile method practices. However, the additional criteria were not discussed [41].
Selection of Practices in Agile methods [39]	The researchers proposed a formal method of prioritizing the practices in Agile methods, whereby two approaches are derived from the Analytical Hierarchy Process (AHP) decision making technique, namely Rule-Description-Practice (RDP) techniques and cost-value analysis [39]. Both are combined to elicit and prioritize the best Agile method practices [39]. However, the researchers state that this approach is limited to only two criteria and it could be enhanced by some more selection factors [39]. It has been suggested that issues such as quality, scope and time should also be taken into consideration [5].
General Agile Adoption [44]	In this work, the researchers suggest that the following five major factors should be considered when migrating to Agile methods: Supportive organization culture especially in the case of negotiation Trusted team including customers High team skills even among developers Developers' opinion must be highly considered when selecting a method Suitable organizational environment that facilitates communication between development team members and the customer. However, these factors are provided as isolated factors and the researchers did not focus on the value of each factor nor on the selection of an Agile method based on the provided selection factors.
Methodology Selection Framework [45]	In this work, it was found that project criticality and size highly affect the selection of Agile methods.

A. Planning Stage

In the planning stage, the researchers started by establishing the research question. Then the search strategy was developed based on the research question. The method for assessing the quality of the studies, definitions for data extraction, and methods for synthesizing the findings were all discussed in the planning stage.

1) Research Question: The main goal of the study was to identify the crucial factors for selecting Agile methods. Thus, the research question was formulated as follows: "What are the factors that influence the selection of the available Agile methods?" This question helps in summarizing and categorizing the crucial selection factors, as well as in identifying gaps for further investigation. Another question that was investigated based on the result of the SLR was "What are the techniques that have been used to select Agile methods?" This question was included in order to investigate the previous techniques that have been used in selecting Agile methods.

2) Search Strategy: In the planning stage it was decided that Science Direct, IEEE Xplore, Emerald FullText, ACM, Springer and Google Scholar would be the libraries to search for primary studies. The researchers created the search string shown in Table 2 to search these libraries. The search string covers the concepts that represent the selection of Agile methods and consists of three parts. The first part was designed to retrieve any study that is related to Agile methods, the second part to retrieve any study that is related to the selection of Agile methods, and the third part to represent the factors. Boolean 'OR' was used to join the alternative terms that have been used in the literature and to join the synonyms, while Boolean 'AND' was used to join all three parts including their alternative terms and synonyms.

TABLE II
SEARCH STRING APPLIED TO SELECTED LIBRARIES

Concept	Alternative terms used in the literature and synonyms	
Agile method	(Agile method* OR Agile Process*)	AND
Selection	(Select* OR Choose* OR Opt*)	AND
Factors	(Factor* OR Criteria* OR Aspect)	

Note: The asterisk symbol '*' was used to include any word variation of each search term. For instance, Method* included different words used in the literature such as Methods OR Methodology OR methodologies.

The period reviewed covered the years 2001 to 2015. It should be noted that 2001 is the starting date of the Agile Manifesto and the year in which the term 'Agile method' was publically coined [46].

3) Selection of Primary Studies: The retrieved studies were saved at Mendeley [47], [48], and they were evaluated based on the following inclusion and exclusion criteria:

Inclusion criteria (studies that met one of the following criteria were included):

- Studies on any of the selection factors for Agile methods;
- Studies reporting the crucial success factors that might influence the selection of Agile methods.

Exclusion criteria (studies that met one of the following criteria were excluded):

- Studies that do not concentrate on Agile methods;
- Studies that partially discuss Agile methods and do not include any information about the selection factors for Agile methods;
- Studies on the general reasons behind the adoption of Agile methods in comparison with non-Agile methods;
- Studies on approaches for evaluating the agility of Agile methods;
- Studies on the implementation of Agile methods;
- Studies on factors related to adopting tools when tailoring Agile methods;
- Studies on the selection of security activities for Agile methods or the selection of software;
- Studies focusing on feedback models to control Agile methods in general rather than factors;
- Studies on critical attributes that affect the running of a project using Agile methods.

The selected studies were then thoroughly analyzed and the references cited in them were glanced over quickly to find other relevant studies that might provide an answer to the research question.

4) Quality Assessment: A questionnaire was designed to assess the quality of the relevant selected studies and to make sure that they contributed to the SLR in terms of answering the research question. The questionnaire was given to two reviewers, who have done research in the same field. The questionnaire required the evaluators to respond to five dichotomous-scale statements (Yes or No). The first three questions were subjective, and the next two questions were objective and were based on those used in previous studies as follows:

Subjective questions:

- The study presents at least one factor that influences the selection of Agile methods [49].
- The study discusses the value of at least one factor to adopt specific Agile methods [50].
- The study represents well-known experts' opinions [26], [27].

Objective questions:

- The study has been published in a reputable journal or conference proceedings [26], [27].
- The study has been cited [51]. (Note: Following this question, the researcher considers Google Scholar citation, 2015 studies were not included in this question [51].)

5) Data Extraction Strategy: Full details of the selected and reviewed studies were recorded [26, 27]. In this step, the researchers sought to identify more specific information about how each study contributed to answering the research question. The included studies were recorded using the following data categories: Study ID, Citation, Objective, Research Methodology, Selection Factors, Finding and Conclusion, and Number of Citations.

6) Synthesis of the Findings: A quantitative synthesis method was applied to count the frequencies of the different factors in the primary studies, it allows researchers to retrieve adequate data and understand the subject better [52]. In addition, a qualitative synthesis was applied to code and summarize the factors appropriately. Then, content analysis, which is a scientific method that provides a summary and analysis of textual messages, was employed to investigate the factors in-depth. Using content analysis, words in a text are systematically and directly compressed and the content categories thus generated are therefore fewer in number based on the explicit rules of the coding [53]. During coding, each text segment is labelled and the text segment can range from a few words to a complete paragraph. This coding enables the rearrangement and integration of words, sentences or paragraphs that are interrelated so that a meaningful portrayal of the data can be generated [53]. For this study, the outcomes are categories of factors affecting the selection of Agile methods.

B. Performance Stage

In this stage, the output of the review protocol was retrieved, the result of which is shown in Table 3.

A total of 1073 studies were retrieved and 62 were selected based on the inclusion and exclusion criteria. In this stage, where there were studies by the same author(s) that had similar content, only the most up-to-date or latest one was selected for further analysis.

TABLE III
RESULTS RETRIEVED FROM LIBRARIES

Source	No. of retrieved studies	No. of included studies
Science Direct	593	17
IEEE Xplore	178	17
Emerald Insight	19	1
ACM	32	6
Springer	53	5
Google Scholar	198	16
Total	1073	62

III. RESULTS AND DISCUSSION

The third and final stage of the SLR was the reporting stage, which involved the analysis of the results retrieved from the libraries based on the search strategy. That is, the identification of the factors that influence the selection of Agile methods. The results of the analysis are presented in Table 4. The table is followed by a detailed explanation of each identified factor and its respective sub-factors. For more details on each study cited in Table 4 readers can refer to the original works.

TABLE IV
TECHNIQUES FOR SELECTING AN AGILE METHOD

No.	Selection factors and sub-factors	Sources	Frequency
	Development Team skills*		
1.	Development Teams' Communication Skills	[7, 54-63]	10
2.	Development Teams' Competency	[1, 2, 6, 56, 64-71]	12
3.	Development Teams' Domain Knowledge	[72-74]	4
	Customer involvement		
4.	Customer Collaboration	[7, 35, 54, 75-79]	8
5.	Customer Commitment	[10, 80-82]	4
6.	Customer's Domain Knowledge	[54, 67, 83]	3
	Organizational		
7.	Organizational Culture	[6, 56, 71, 75, 84-91]	12
	Nature of project		
8.	Size	[7, 16, 55, 64, 66, 80, 84, 92-100]	16
9.	Criticality	[17, 83, 84, 92, 94, 96, 97, 101, 102]	9
10.	Decomposability	[84, 103, 104]	3
	Project constraints		
11.	Cost	[25, 75, 92, 105]	4
12.	Time	[7, 9, 54, 64, 75, 105]	6
13.	Scope	[64, 65, 83]	3
14.	Quality	[7, 83, 93, 101]	4

Note: *Development Team are the programmers, analysts, designers, testers, scrum master and Agile manager

A. *Development Team Skills*

It has been stated that "a bunch of novices will deliver a useless product, the choice is yours" [106]. Therefore, development teams' skills are a crucial factor in any system implementation, especially that of a critical system [17]. The development team must be able to increase their skills when migrating to Agile methods [17]. In any project, Agile methods focus on the talents of the team. However, different organizations have different teams' skills and may follow different Agile methods [6]. For instance, XP requires highly skilled programmers and effective interaction and communication among them [7], [10], but other Agile methods may require a different set of skills. Also, the skills of teams may vary [6] but a team could still adopt any of the FDD practices based on their experience [13]. However, there is no concrete guidance on how to adopt these practices [107]. The main sub-factors of development team skills are communication skills, technical skills and experience (competency), and domain knowledge. Each of these sub-factors is briefly described in the following subsections.

1) Development Teams' Communication Skills: The personal characteristics of team members such as collaborative attitude are crucial [108] to the success of a project. Empirical results demonstrate that having extensive communication skills is the main factor that influences the success of Agile methods [56], [109]. Studies have shown that team communication is vital in implementing an Agile method [35], [68] and in improving the method's effectiveness [57], [69]. In a project implementation, effective team communication is one of the most important factors for the successful adoption of Agile methods [6], [68]. Thus team communication between developers is important when selecting appropriate practices to follow in an Agile method [57], [59].

For instance, in a multi-team environment, especially when implementing large projects, the adoption of XP is difficult because its practices focus on the needs of individual teams [7], [59]. Communication within the team is also crucial when implementing projects using Agile methods [7], [55], [60], [61]. Further, when considering the best practices for team communication, it is important to take into account the physical setting (e.g., direct or indirect communication) [62]. Therefore, taking this factor into consideration can assist in the selection of an appropriate Agile method for a given project [54], [63] and ensure effective implementation of the method [57], [69].

2) Development Teams' Competency (Technical Skills and Experience): The skills and abilities required in a self-managed team following an Agile method are different to those required in traditional methods [31]. They even differ between Agile methods themselves [94]. Thus the factor of competency is an important one. Competency in this context refers to whether one has real-world experience in the technology domain, has built similar systems in the past, and has good personal and interpersonal skills [108]. Task allocation has been found to be heavily dependent on team experience, which is directly related to the competency of teams [70], [110]. In addition, individual skills are important in task assignment, especially following a new innovation such as Agile methods [70], [71]. For instance, Agile methods emphasize producing and delivering working software quickly, which relies on the team members having the necessary skill [1], [40], [85]. Hence, companies have started to adopt and implement practices to empower and increase the skills of their employees [56].

It has been found that Agile methods require each team to contain at least 25 percent of technically competent people when implementing any project [108]. Previous experience in developing similar software is also of value [68]. Team members' competencies affect the development pace and the delivery of software according to customers' requirements. Thus, the team's technical skills is a crucial factor to consider in the selection of an Agile method as team members may not be familiar with all Agile methods [6], [64], [66], [68], [70], [97], [111].

Developers' skills and experience highly affect the adoption of Agile methods [67], [83]. Thus, inexperienced developers may end up adopting inappropriate Agile methods [17], [31], [67]. Additionally, pair programming has been a common XP practice but has been found to be inefficient [61], [112], [113], especially when the programmers have incompatible and inappropriate sets of experience and skills for the tasks at hand [114]. In fact, it has been argued that it might be better if a problem were solved by only one programmer [115]. Thus, technical skills and experience are crucial factors in Agile method selection.

3) Development Teams' Domain Knowledge: Eighteen studies show that team skill and domain knowledge are common selection criteria in Agile team formation [72]. For instance, excellent domain knowledge has been found to contribute to the success of XP use [73]. Team's domain knowledge is considered as an important factor in the selection of Agile methods [72], [74]. Also, inappropriate selection of the practices in an Agile method leads to poor implementation [72], [116].

B. Customer Involvement

The Agile Manifesto [32] highly values customer involvement. In particular, face-to-face communication is vital. However, involving valuable customers can be a challenging process [117]. The importance and the degree of customer involvement have been determined by Agile researchers based on investigating how developers who started to follow Agile methods involved their customers [117]-[119]. However, there are contradictory findings regarding customer involvement. For instance, it was found that full customer involvement is not necessary; rather, a 21 percent level of involvement is sufficient [118]. Meanwhile, other researchers state that customers do not need to be on-site at all and they can be replaced by a product management team [117], [119]. However, others indicate that many software developers constantly value on-site customers and prefer them to be available on-site [10], [120]-[123]. However, another case study indicates that sometimes on-site customers could not be consistently available and were not actively involved [120]. However, it has been pointed out that the customer is the only person who knows the business domain well and can decide on the resources, scope, and schedule, and clarify vague complex projects [124], [125].

An empirical study indicates that customer involvement is one of the most controversial topics in relation to Agile methods, and found that the customer must be involved, collaborative, knowledgeable and committed but not necessarily present all the time [118]. Therefore, to ensure timely delivery of project iterations [9], [10], [125], when project managers contemplate the adoption of practices to find a suitable Agile method they take the degree of customer involvement into consideration [126].

Based on the results of the SLR, the collaboration, knowledge, and the commitment of the customer were identified as the main sub-factors of customer involvement that need to be considered when selecting an appropriate Agile method. These sub-factors are discussed in the following subsections.

1) Customer Collaboration: Three of the 12 principles in the Agile Manifesto mentioned above are strongly related to the collaboration factor [32]. One of the principles is prioritizing customer satisfaction through early and continuous delivery of valuable software [32], which requires that the customer demonstrates high motivation, a sense of responsibility towards the project element, active participation, and on-site availability to the software development team [1], [2]. Customer collaboration is, thus important in the selection of an Agile method [7].

On-site customer collaborative practice usually involves interaction with one person as a representative, who is normally an employee within the client organization. However, for large projects that require the involvement of more than one representative, collaboration among several customers may be necessary [77]. In such cases, the practices in Agile methods that highly support customer collaboration for each project should be followed accordingly [75]. For example, in XP, an effective relationship between the customer and developers is necessary for project success [7], [35], [77], [78], [127]. Further, different practices for different levels of customer involvement are required, especially as the adequacy of customers who work on different Agile projects may vary [79]. Moreover, project managers need to be aware that a lack of customer collaboration can lead to a breakdown in the collaborative atmosphere [128].

Close customer collaboration allows end-users to closely align the system to their actual needs and feel a greater sense of ownership [90]. However, correct practices of Agile methods in a different type of involvement (Less or High involvement) would highly impact project implementation [54]. For instance, in a comparison of different methods such as SCRUM, XP, FDD and Crystal, which each have a different approach to customer collaboration, it was demonstrated that the selection of a particular method based on this factor leads to the success of project implementation [54].

2) Customer Commitment: Commitment throughout all the stages of the development process can put customers under stress [81]. Thus, on-site customer commitment is crucial when selecting the appropriate Agile methods [80], [82]. For instance, an uncommitted on-site customer negatively affects the efficiency of the XP method [10].

3) Customer's Domain Knowledge: In Agile method practices, incapable and unknowledgeable customers will put project implementation at risk [54]. For instance, feedback from an unknowledgeable on-site customer (XP practice) will not assist in providing correct functionality

[67]. Hence, customer knowledge of the domain is one of the main software process determinants [54], [83].

C. *Organizational Culture (Type of Organization)*

When any organizations adopts new Agile methods, if the organization has a flexible organizational culture (which is collaborative) it could easily adopt any Agile method (if we consider this factor in isolation), whereas an organization that is inflexible (rigid or hierarchical) and has a command-and-control culture will face difficulties [6, 84]. Thus, the type of organizational culture is one of the major factors to consider when deciding which Agile method to adopt [86, 88].

There are four main types of organizational culture, namely hierarchical, random, collaborative and synchronous, which could be used to analyze the match between an Agile method and an organizational culture [87]. This categorization has been applied to determine the relationship between the organizational culture and XP, and it was found that XP practices can be adopted successfully by all four organizational cultures equally effectively [88]. In addition, four different types of organizational cultural dimensions have been identified, namely clan, democratic, hierarchical and disciplined [129]. Agile methods are generally appropriate for the democratic culture [130], while some XP practices fit well with the hierarchical organization [88]. On the other hand, the Schneider Culture Model divides organizational culture into four types: collaboration, cultivation, control and competence [131]. It has been found that collaboration, cultivation and competence cultures are suitable for Agile methods, while the control culture is not [132].

In an organization with a hierarchical or collaborative culture, different features and practices can be found in different Agile methods to suit the given project [89]. Thus, organizational culture is considered an important factor affecting the successful adoption of Agile methods [56], [71], [86], [133]. Ignoring the culture of the organization when selecting an Agile method can lead to problems because changing an organizational culture is very difficult [134]. In fact, a multi-case study of nine projects demonstrates that specific organizational cultures are correlated with the effective use of Agile methods [86]. Therefore, it is necessary to take organizational culture into consideration when selecting an Agile method [75], [90], [134], especially because organizational culture cannot be changed to adapt to the selected Agile method [84], [134]. Organizations are not the same [85], [135]. Thus, certain Agile methods may be successful in one organization but may cause system failure in others [85]. Therefore, organizational culture type is an important consideration in Agile method selection [84], [91].

D. *Nature of Project*

Software development problems can be complicated and thus can require complex and different solutions that involve the adoption of different techniques and strategies [136]. Additionally, projects come in different sizes and have different levels of criticality, so there is no single Agile method that can solve all organizational problems [40]. Therefore, the nature of the project, in terms of its size and criticality as well as its decomposability, has to be considered when selecting the appropriate Agile method. These sub-factors are described in the following subsections.

1) Size: Team and project size can affect the outcome of different Agile methods in different ways [40], [101], [137]. Thus, when composing an Agile team, the size of the project and the team must be considered [55]. In Agile method selection, the project nature is taken into account [97], and particularly the project size [17], [41], [64], [66], [80], [95], [97], [102].

Most Agile methods can be used for small projects. However, careful selection of the Agile method is necessary when the size of the team and the project are large [40], [96], [98]. Small-sized teams could easily perform a smoke or unit test to test the functionality of their implemented system and conduct integration testing accordingly. However, for a large and complex project, other types of testing such as hardware-in-the-loop and a comprehensive test environment are required at the integration testing phase to detect most of the defects [98]. The Crystal Orange method is suitable for large projects consisting of around 40 developers [94]. Meanwhile, FDD is suitable for projects with a very large team [96]. However, methods such as Crystal Clear [17] and XP [99] are not fitting for large projects due to implementation difficulty [17], [93], [99]. In addition, the XP method requires no more than 10 developers in the project team [10], [92]. Thus it is clear that project size highly affects the selection of Agile methods [7].

2) Criticality: An understanding of the criticality of the project can help in determining differences in the results of applying different Agile methods [94], [101]. The project's criticality appears to highly influence the cost of the project [5] and can affect human lives [34]. It also therefore affects the selection of an Agile method [41], [83], [94], [96], [97]. For instance, the Crystal Clear method is not suitable when a defect in the implementation of a specific project could cause monetary loss [17]. In this case, the Crystal Orange method could be suitable [96], while FDD could be more appropriate for critical project development [92], [107]. Additionally, the activity of project inspection can be practised in some Agile methods (e.g., DSDM, ISD) and is valuable in critical and complex projects [107].

3) Decomposability: A decomposable project also requires appropriate Agile method selection [136]. An aeroplane, radiation beam system or other critical systems could be implemented using Agile methods but would necessitate in-depth consideration of the nature of the problem [136]. For instance, if the project cannot be decomposed into different sub-projects, scaling and adopting the most appropriate Agile method practices that suit the project becomes difficult [136]. Also, SCRUM and DSDM are not suitable for non-decomposable projects such as video games [103], [104].

E. *Project Constraints*

Delivery date (or schedule), cost, quality, scope, and value are crucial considerations in any project implementation [1] and these factors determine the implementation success of any project [5]. When a customer insists on a specific delivery date, as well as specifies the quality, cost, scope, and value of the system, the developers need to follow the most appropriate method to satisfy the customer [5] and this could result in a change to their initially selected method. With flexible constraints, the team could adopt any familiar method that they prefer. However, with rigid constraints, the team must find the most appropriate method to suit those constraints [136]. Hence, project constraints should be considered when selecting the most appropriate Agile method. The project constraint sub-factors cost, time, scope, quality are described in the following subsections.

1) Cost: Cost is one of the key factors under project constraints [138]. When prioritizing certain requirements, the implementation cost should be considered [105], [139]. Thus, checking the cost-effectiveness and acceptableness of the adopted practices (in Agile methods) is very important [25]. For instance, FDD and ASD do not support keeping the process cost-effective, whereas the Crystal group of methods is more cost-effective compared to other Agile methods [75]. On the other hand, the XP and Kanban methods focus on cost saving [92].

2) Time: Time is also a crucial project constraint. When prioritizing the requirement, the implementation time should be considered [105]. Some Agile methods, such as the XP method, focus on implementing the projects in a fast manner [9] while other Agile methods, such as Crystal Orange [94], are not intended for this purpose [140]. Therefore, checking the suitability of the adopted Agile method for an implementation that has a short timescale is very important, especially when the implementation schedule is specified [64], [75] because it critically influences the project cost [5]. Additionally, the time of system delivery to market must also be considered when selecting Agile methods [54]. In relation to this, the SCRUM and Crystal group methods supersede other Agile methods in terms of project delivery speed [75].

3) Scope: The scope of a project is one of the key software process determinants [64], [83]. The scope is sometimes changed especially when the nature of business requires multiple and frequent changes. However, in SCRUM, no new functions can be added once the sprint towards project delivery is underway [109]. If changes to the scope are required, Kanban works better because it focuses on minimizing and limiting the work in process [65].

4) Quality: quality of adopting a particular practice must be considered especially when the implementation of high-quality projects is sought [25]. Therefore, practitioners should be encouraged and guided when deciding on the best practices to adopt because this could improve the quality of the processes used [83], [93].

In an investigation of the quality of implemented projects that compared the pre and post process adoption data to determine the severity of errors and defects of passed and failed test cases, it was concluded that SCM obviously improved software quality [141]. The sprint review meeting is the main practice for improving quality in the SCRUM method [96]. However, pair programming, refactoring and simple design, which are XP practices, also focus on delivering high-quality products [7].

Principle four and principle five of ISO 9001 on quality management identify the best processes (methods) that can effectively and efficiently accelerate activities. Additionally, principle five also aims at aligning complementary processes (methods) to improve efficiency [142]. Thus, selecting the appropriate Agile methods would increase the quality of any project implementation.

To conclude the result, the conceptual framework in Fig. 2 below illustrates the factors discussed above that have been found to affect the selection of an Agile method. The methods whose efficacy is affected by these factors are XP, SCRUM, TDD, FDD, LD, ASD, AM, Crystal Family, ISD, DAD, DSDM, DevOps, and Kanban (centre box in Fig. 2). Each method comprises different practices and can be used for different projects in different situations. For instance, when the project size is large (more than 10 developers), then the XP method or Crystal Clear would not be an appropriate choice, whereas other methods such as DAD and Crystal Red would be more appropriate. By considering the degree to which each of the identified factors is important to the project, project managers should be able to select the most appropriate Agile method for their needs.

As discussed in Section III above, the five factors that affect any software development project are: Nature of Project (size, criticality and decomposability), Development Team Skills (communication, domain knowledge and competency), Project Constraints (Cost, time, scope and quality), Customer Involvement (collaboration, commitment and domain knowledge) and Organizational Culture (hierarchical, collaborative, cultivation and competence).

IV. Conclusion

It is difficult for project managers and other decision makers to select the most suitable Agile method for a specific project. A range of selection factors needs to be considered when selecting an Agile method. To aid project managers in this important decision, a systematic literature review was undertaken to identify the crucial selection factors for the most well-known Agile methods. The review identified the nature of the project, development team skills, project constraints, organizational culture and customer involvement as the key factors that can assist project managers in selecting the best Agile method for their purposes, and thereby help them to decrease project costs and the likelihood of project failure. Further work will be undertaken by the researchers to validate the identified selection factors empirically, investigate how they can be used in selecting Agile methods and finally how the selected method can be adapted.

Acknowledgments

Special thanks to the reviewers for assessing and checking the selected studies. We also acknowledge the Faculty of Information Science and Technology at Universiti Kebangsaan Malaysia (UKM) for supporting this research under Fundamental Research Grant Scheme project (FRGS/2/2013/ICT01/UKM/02/2).

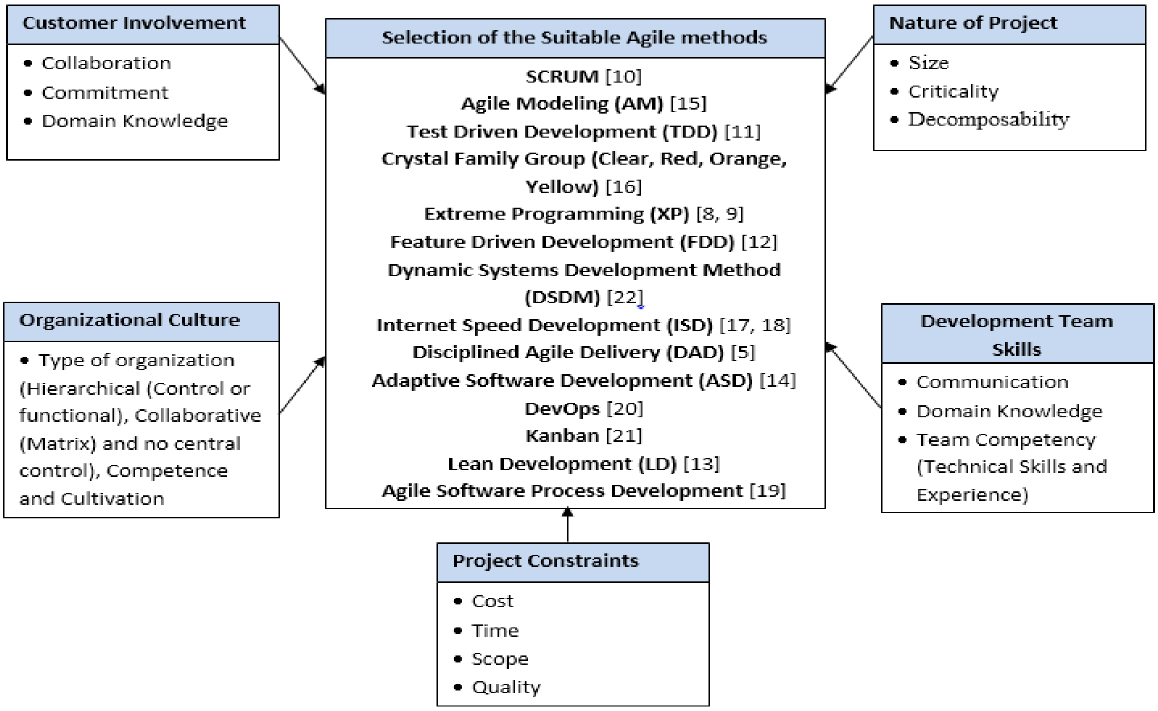
Customer Involvement
Collaboration
Commitment
Domain Knowledge
Selection of the Suitable Agile methods
SCRUM [10]
Agile Modeling (AM) [15]
Test Driven Development (TDD) [11]
Crystal Family Group (Clear, Red, Orange, Yellow) [16]
Extreme Programming (XP) [8, 9]
Feature Driven Development (FDD) [12]
Dynamic Systems Development Method (DSDM) [22]
Internet Speed Development (ISD) [17, 18]
Disciplined Agile Delivery (DAD) [5]
Adaptive Software Development (ASD) [14]
DevOps [20]
Kanban [21]
Lean Development (LD) [13]
Agile Software Process Development [19]
Nature of Project
Size
Criticality
Decomposability
Organizational Culture
Type of organization (Hierarchical (Control or functional), Collaborative (Matrix) and no central control), Competence and Cultivation
Development Team Skills
Communication
Domain Knowledge
Team Competency (Technical Skills and Experience)
Project Constraints
Cost
Time
Scope
Quality

Fig. 2 Conceptual selection framework based on SLR

REFERENCES

[1] T. Chow and D. B. Cao, "A survey study of critical success factors in agile software projects", Journal of Systems and Software, vol. 81, (2008), pp. 961-971.

[2] S. C. Misra, V. Kumar and U. Kumar, "Identifying some important success factors in adopting agile software development practices", Journal of Systems and Software, vol. 82, (2009), pp. 1869-1890.

[3] A. A. Hamid and Z. Mansor, "Client's Readiness Assessment Success Factors for Outsourcing Software Projects", International Journal on Advanced Science, Engineering and Information Technology, vol. 6, no. 6, (2016), pp. 820-827.

[4] A. L. Asnawi, A. M. Gravell and G. B. Wills, "Emergence of agile methods: perceptions from software practitioners in Malaysia", AGILE India (AGILE INDIA), (2012), pp. 30-39.

[5] D. Stankovic, V. Nikolic, M. Djordjevic and D. B. Cao, "A survey study of critical success factors in agile software projects in former Yugoslavia IT companies", Journal of Systems and Software, vol. 86, (2013), pp. 1663-1678.

[6] S. W. Ambler and M. Lines, "Disciplined agile delivery: A practitioner's guide to agile software delivery in the enterprise: IBM Press", (2012).

[7] Y. A. Harb, C. Noteboom and S. Sarnikar, "Evaluating Project Characteristics for Selecting the Best-fit Agile Software Development Methodology: A Teaching Case", Journal of the Midwest Association for Information Systems (JMWAIS), vol. 1, (2015), pp. 4.

[8] J. A. Livermore, "Factors that significantly impact the implementation of an agile software development methodology", Journal of Software, vol. 3, (2008), pp. 31-36.

[9] K. Beck, "Embracing change with extreme programming," Computer, vol. 32, (1999), pp. 70-77.

[10] K. Beck, "Extreme programming explained: embrace change", Addison-Wesley Professional, (2000).

[11] M. Cohn, "Succeeding with Agile: Software Developing Using Scrum", Upper Saddle River, Addison-Wesley, (2009).

[12] K. Beck, "Test-driven development: by example" Addison-Wesley Professional, (2003).

[13] J. M. Felsing and S. R. Palmer, "A Practical Guide to Feature-Driven Development", IEEE Software, vol. 7, (2002), pp. 67-72.

[14] M. Poppendieck and T. Poppendieck, "Lean software development: an agile toolkit", Addison-Wesley Professional, (2003).

[15] J. A. Highsmith, "Agile software development ecosystems", Addison-Wesley Professional, vol. 13, (2002).

[16] S. W. Ambler, "Agile modeling", John Wiley & Sons New York, (2002).

[17] A. Cockburn, "Crystal clear: a human-powered methodology for small teams", Pearson Education, (2004).

[18] M. Cusumano and D. B. Yoffie, "Software development on Internet time", Computer, vol. 32, no. 10, (1999), pp. 60-69.

[19] R. Baskerville, L. Levine, J. Pries-Heje, B. Ramesh and S. Slaughter, "How Internet software companies negotiate quality", Computer, (2001), pp. 51-57.

[20] M. Aoyama, "Agile software process and its experience", Proceedings of the 20th international conference on Software engineering, IEEE, (1998), pp. 3-12.

[21] M. Httermann, "DevOps for developers", Apress, (2012).

[22] D. J. Anderson, "Kanban: successful evolutionary change for your technology business", Blue Hole Press, (2010).

[23] J. Stapleton, "dynamic systems development method (DSDM): the method in practice", Publishers Addison-Wesley Longman, Cambridge University Press, (1997).

[24] O. N. A. Al-Allaf, "The Adoption of Agile Processes in Large Web Development Enterprises: A Survey in Jordan", IJWA, vol. 2, (2009), pp. 206-216.

[25] G. Mikulenas and K. Kapocius, "An Approach for Prioritizing Agile Practices for Adaptation", Information Systems Development, Springer, (2011), pp. 485-498.

[26] B. Kitchenham, "Procedures for performing systematic reviews", Keele, UK, Keele University, vol. 33, (2004), pp. 1-26.

[27] P.Brereton, B. A. Kitchenham, D. Budgen, M. Turner and M. Khalil, Lessons from applying the systematic literature review process within the software engineering domain. Journal of systems and software, vol. 80, no. 4, (2007), pp. 571-583.

[28] B. W. Boehm, "A spiral model of software development and enhancement", Computer, vol. 21, (1988), pp. 61-72.

[29] W. W. Royce, "Managing the development of large software systems", proceedings of IEEE WESCON, (1970), pp. 328-388.

[30] A. M. Davis and D. A. Leffingwell, "Using requirements management to speed delivery of higher quality applications", Rational Software Corporation, vol. 20, (1996), pp. 2004.

[31] A. Dennis, B. H. Wixom and R. M. Roth, "Systems analysis and design", John wiley & sons, (2012).

[32] A. Alliance, "Agile manifesto", online at http://www.agilemanifesto.org. 6, no. 1, (2001).

[33] M. Alqudah and A. A. Abdallah, "Implementing computer-aided language learning tool using hybrid agile method: A case study". Informatics and Creative Multimedia (ICICM), International Conference on. IEEE, (2013), pp. 174 - 180.

[34] A. Cockburn, "Selecting a project's methodology", IEEE Software, (2000), pp. 64-71.

[35] B. Boehm and R. Turner, "Balancing agility and discipline: A guide for the perplexed", Addison-Wesley Professional, (2003).

[36] R. L. Glass and I. Vessey, "Contemporary application-domain taxonomies", IEEE Software, (1995), pp. 63-76.

[37] C. Potts, "Software-engineering research revisited", Software, IEEE, vol. 10, (1993), pp. 19-28.

[38] P. Coad, E. Lefebvre and J. D. Luca, "Java Modeling In Color With UML: Enterprise Components and Process", César Ricardo Alducin-Ruiz, Jorge Octavio Ocharán-Hernández, Lizbeth A. Hernández-González, (1999).

[39] G. Mikulenas, R. Butleris and NemuraitLina, "An approach for the metamodel of the framework for a partial agile method adaptation", Information Technology And Control, vol. 40, (2011), pp. 71-82.

[40] P. Kettunen and M. Laanti, "How to steer an embedded software project: tactics for selecting the software process model", Information and software technology, vol. 47, (2005), pp. 587-608.

[41] E. Mnkandla and B. Dwolatzky, "Agile methodologies selection toolbox", Software Engineering Advances, International Conference on ICSEA, (2007), pp. 72-72.

[42] M. Griffiths, "Agile suitability filters", Retrieved January, (2013).

[43] D. Duka, "Agile experiences in software development", MIPRO, Proceedings of the 35th International Convention, (2012), pp. 692-697.

[44] L. Lehtola and M. Kauppinen, "Empirical evaluation of two requirements prioritization methods in product development projects", Software Process Improvement, Springer, (2004), pp. 161-170.

[45] J. B. Barlow, J. Giboney, M. J. Keith, D. Wilson, R. Schuetzler, P. B. Lowry and A. Vance, "Overview and guidance on agile development in large organizations", Communications of the Association for Information Systems, vol. 29, (2011), pp. 25-44.

[46] K. Beck, M. Beedle, A. Van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries and J. Kern, "The agile manifesto", (2001).

[47] K. Salija, R. Hidayat and A. A. Patak, "Mendeley Impact on Scientific Writing: Thematic Analysis", International Journal on Advanced Science, Engineering and Information Technology, vol. 6, no. 5, (2016), pp.657-662.

[48] A. A. Patak, H. A. Naim and R. Hidayat, "Taking Mendeley as Multimedia-based Application in Academic Writing", International Journal on Advanced Science, Engineering and Information Technology, vol. 6, no. 4, (2016), pp.557-560.

[49] P. D. Leedy and J. E. Ormrod, "Practical research Planning and design", Pearson Education Inc, vol. 9, (2010), pp. 10.

[50] T. Dyba and T. Dingsøyr, "Empirical studies of agile software development: A systematic review", Information and software technology, vol. 50, (2008), pp. 833-859.

[51] A. Fernandez, E. Insfran and S. Abrahão, "Usability evaluation methods for the web: A systematic mapping study", Information and software technology, vol. 53, (2011), pp. 789-817.

[52] M. Zammani and R. Razali, "An Empirical Study of Information Security Management Success Factors", International Journal on Advanced Science, Engineering and Information Technology, vol. 6, no. 6, (2016), pp. 904-913.

[53] K. Krippendorff, "Content analysis: An introduction to its methodology", Sage, (2012).

[54] M. Qasaimeh, H. Mehrfard and A. Hamou-Lhadj, "Comparing Agile Software Processes Based on the Software Development Project Requirements", Computational Intelligence for Modelling Control & Automation, International Conference on, (2008), pp. 49-54.

[55] C. d. O. Melo, D. S. Cruzes, F. Kon and R. Conradi, "Interpretative case studies on agile team productivity and management", Information and software technology, vol. 55, (2013), pp. 412-427.

[56] F. Loforte Ribeiro and M. Timóteo Fernandes, "Exploring agile methods in construction small and medium enterprises: a case study", Journal of Enterprise Information Management, vol. 23, (2010), pp. 161-180.

[57] C. Young and H. Terashima, "How did we adapt Agile processes to our distributed development?", Agile, AGILE'08. Conference, (2008), pp. 304-309.

[58] T. Kahkonen, "Agile methods for large organizations-building communities of practice," Agile Development Conference, (2004), pp. 2-10.

[59] R. Phalnikar, V. S. Deshpande and S. D. Joshi, "Applying agile principles for distributed software development", Advanced Computer Control, ICACC'09 International Conference, (2009), pp. 535-539.

[60] D. Karlstrom and Runeson, Per, "Combining agile methods with stage-gate project management", IEEE Software, (2005), pp. 43-49.

[61] K. Petersen and C. Wohlin, "A comparison of issues and advantages in agile and incremental development between state of the art and an industrial case", Journal of Systems and Software, vol. 82, (2009), pp. 1479-1490.

[62] D. Mishra, A. Mishra and S. Ostrovska, "Impact of physical ambiance on communication, collaboration and coordination in agile software development: An empirical evaluation", Information and software technology, vol. 54, (2012), pp. 1067-1078.

[63] D. Wahyudin, M. Heindl, B. Eckhard, A. Schatten and S. Biffl, "In-time role-specific notification as formal means to balance agile practices in global software development settings", Balancing Agility and Formalism in Software Engineering, Springer, (2008), pp. 208-222.

[64] Townsend, "Choosing the best Agile methodology for your development needs", Computer, (2011).

[65] M. O. Ahmad, J. Markkula and M. Oivo, "Kanban in software development: A systematic literature review", Software Engineering and Advanced Applications (SEAA), 39th EUROMICRO Conference on, (2013), pp. 9-16.

[66] H. K. Flora, S. V. Chande and X. Wang, "Adopting an Agile Approach for the Development of Mobile Applications", International Journal of Computer Applications, vol. 94, no. 17, (2014), pp. 43-50.

[67] D. Kirk and E. Tempero, "A lightweight framework for describing software practices", Journal of Systems and Software, vol. 85, (2012), pp. 582-595.

[68] J. McManus, "Team agility", The Computer Bulletin, vol. 45, (2003), pp. 26-27.

[69] G. Melnik and F. Maurer, "Direct verbal communication as a catalyst of agile knowledge sharing", Agile Development Conference, (2004), pp. 21-31.

[70] J. Lin, "Context-aware task allocation for distributed agile team", Automated Software Engineering (ASE), IEEE/ACM 28th International Conference on, (2013), pp. 758-761.

[71] R. W. van Dijk, "Determining the Suitability of Agile Methods for a Software Project", 15th Twente Student Conference on IT, (2011) June 20.

[72] F. Q. B. da Silva, A. C. s. C. França, M. Suassuna, L. M. R. de Sousa Mariz, I. Rossiley, R. C. G. de Miranda, T. B. Gouveia, C. V. Monteiro, E. Lucena, E.S. Cardozo and E. Espindola, "Team building criteria in software projects: A mix-method replicated study", Information and software technology, vol. 55, no. 7, (2013), pp. 1316-1340.

[73] M. R. Hilkka, T. Tuure and R. Matti, "Is extreme programming just old wine in new bottles: A comparison of two cases", Journal of Database Management, vol. 16, (2005), pp. 41-61.

[74] T. Field, "When Bad Things Happen to Good Projects: The warning signs of failure are detectable long before a project falls apart", CIO-FRAMINGHAM MA, vol. 11, (1997), pp. 54-56.

[75] A. Qumer and B. Henderson-Sellers, "An evaluation of the degree of agility in six agile methods and its applicability for method engineering", Information and software technology, vol. 50, (2008), pp. 280-295.

[76] K. Korhonen, "Evaluating the impact of an agile transformation: a longitudinal case study in a distributed context", Software Quality Journal, vol. 21, (2013), pp. 599-624.

[77] K. Conboy and L. Morgan, "Beyond the customer: Opening the agile systems development process", Information and software technology, vol. 53, (2011), pp. 535-542.

[78] A. Leon and A. S. Koch, “Agile software development evaluating the methods for your organization”, Artech House, Inc., (2004).

[79] R. Hoda, J. Noble and S. Marshall, "Self-organizing roles on agile software development teams", Software Engineering, IEEE Transactions on, vol. 39, (2012), pp. 422-444.

[80] A. Elssamadisy and G. Schalliol, "Recognizing and responding to bad smells in extreme programming", Proceedings of the 24th International conference on Software Engineering, (2002), pp. 617-622.

[81] A. Martin, R. Biddle and J. Noble, "The XP customer role in practice: Three studies", Agile Development Conference, (2004), pp. 42-54.

[82] R. Hoda, J. Noble and S. Marshall, "The impact of inadequate customer collaboration on self-organizing Agile teams", Information and software technology, vol. 53, (2010), pp. 521-534.

[83] A. Geras, M. Smith and J. Miller, "Configuring hybrid agile-traditional software processes", Extreme Programming and Agile Processes in Software Engineering, Springer, (2006), pp. 104-113.

[84] S. W. Ambler, "Scaling agile: an executive guide", Agility Scale Whitepaper, (2010).

[85] S. Nerur, R. Mahapatra and G. Mangalaraj, "Challenges of migrating to agile methodologies", Communications of the ACM, vol. 48, (2005), pp. 72-78.

[86] D. E. Strode, S. L. Huff and A. Tretiakov, "The impact of organizational culture on agile method use", in System Sciences, 42nd Hawaii International Conference on HICSS'09, (2009), pp. 1-9.

[87] A. Cockburn and J. Highsmith, "Agile software development: The people factor", Computer, (2001), pp. 131-133.

[88] H. Robinson and H. Sharp, "Organisational culture and XP: three case studies", null, (2005), pp. 49-58.

[89] J. Iivari and N. Iivari, "The relationship between organizational culture and the deployment of agile methods", Information and software technology, vol. 53, (2011), pp. 509-520.

[90] P. McBreen and K. Foreword By-Beck, “Questioning extreme programming”, Addison-Wesley Longman Publishing Co., (2002).

[91] F. K. Y. Chan and J. Y. L. Thong, "Acceptance of agile methodologies: A critical review and conceptual framework", Decision Support Systems, vol. 46, (2009), pp. 803-814.

[92] A. S. Campanelli and F. S. Parreiras, "Agile methods tailoring–A systematic literature review", Journal of Systems and Software, vol. 110, (2015), pp. 85-100.

[93] H. Altarawneh and A. E. Shiekh, "A theoretical agile process framework for web applications development in small software firms", Software Engineering Research, Management and Applications, Sixth International Conference on SERA'08, (2008), pp. 125-132.

[94] A. Cockburn, “Agile software development: the cooperative game”, Publisher Pearson Education, (2006).

[95] R. F. Paige, R. Charalambous, X. Ge and P. J. Brooke, "Towards agile engineering of high-integrity systems", Computer Safety, Reliability, and Security, Springer, (2008), pp. 30-43.

[96] F. Kanwal, K. Junaid and M. A. Fahiem, "A hybrid software architecture evaluation method for fdd-an agile process model", Computational Intelligence and Software Engineering (CiSE), International Conference on, (2010), pp. 1-5.

[97] K. Conboy and B. Fitzgerald, "Method and developer characteristics for effective agile method tailoring: A study of XP expert opinion", ACM Transactions on Software Engineering and Methodology (TOSEM), vol. 20, (2010), pp. 2.

[98] A. Kasoju, K. Petersen and M. V. Mäntylä, "Analyzing an automotive testing process with evidence-based software engineering", Information and software technology, vol. 55, (2013), pp. 1237-1259.

[99] B. Boehm, "A view of 20th and 21st century software engineering", Proceedings of the 28th international conference on Software engineering, (2006), pp. 12-29.

[100] D. J. Reifer, F. Maurer and H. Erdogmus, "Scaling agile methods," IEEE software, vol. 20, (2003), pp. 12-14.

[101] L. Layman, L. Williams, D. Damian and H. Bures, "Essential communication practices for Extreme Programming in a global software development team", Information and software technology, vol. 48, (2006), pp. 781-794.

[102] T. Ozcan, S. Kocak and P. Brune, "Agile Software Development with Open Source Software in a Hospital Environment-Case Study of an eCRF-System for Orthopaedical Studies", Web Engineering, Springer, (2013), pp. 439-451.

[103] J. Mateos-Garcia and J. Sapsed, "Adopting ‘Agile’and ‘Scrum’practices as ‘Organisational becoming’: Cases from the UK video games industry", Euromot: the Third European Conference on Management of Technology, (2008).

[104] D. Verma and V. Gupta, "agile methodologies for different industries", IJREAT International Journal of Research in Enginee ring & Advanced Technology, vol. 2, no. 1, (2014), pp. 1-9.

[105] R. H. Al-Ta'ani and R. Razali, "Prioritizing Requirements in Agile Development: A Conceptual Framework", Procedia Technology, vol. 11, (2013), pp. 733-739.

[106] K. Schwaber, G. Laganza and D. D’Silva, "The truth about agile processes: frank answers to frequently asked questions", Forrester Report, (2007).

[107] P. Abrahamsson, N. Oza and M. T. Siponen, "Agile Software Development Methods: A Comparative Review1", Agile Software Development, Springer, (2010), pp. 31-59.

[108] M. Lindvall, V. Basili, B. Boehm, P. Costa, K. Dangle, F. Shull, R. Tesoriero, L. Williams and M. Zelkowitz, "Empirical findings in agile methods", Conference on Extreme Programming and Agile Methods, XP/Agile Universe, Springer Berlin Heidelberg, (2002) Aug 4, pp. 197-207.

[109] H. F. Cervone, "Understanding agile project management methods using Scrum", OCLC Systems & Services, International digital library perspectives, vol. 27, (2011), pp. 18-22.

[110] C. L. n. de la Barra, B. Crawford, R. Soto, S. Misra and E. Monfroy, "Agile software development: It is about knowledge management and creativity", Computational Science and Its Applications, ICCSA, Springer, (2013), pp. 98-113.

[111] K. Kuusinen, T. Mikkonen and S. Pakarinen, "Agile user experience development in a large software organization: Good expertise but limited impact", Human-Centered Software Engineering, Springer, (2012), pp. 94-111.

[112] S. Ilieva, P. Ivanov and E. Stefanova, "Analyses of an agile methodology implementation", 30th Proceedings Euromicro Conference, (2004), pp. 326-333.

[113] A. Mackenzie and S. Monk, "From cards to code: How extreme programming re-embodies programming as a collective practice", Computer Supported Cooperative Work (CSCW), vol. 13, (2004), pp. 91-117.

[114] G. Melnik and F. Maurer, "Perceptions of agile practices: A student survey", Extreme Programming and Agile Methods, XP/Agile Universe Springer, (2002), pp. 241-250.

[115] M. M. Muller, "Do programmer pairs make different mistakes than solo programmers?", Journal of Systems and Software, vol. 80, (2007), pp. 1460-1471.

[116] E. Egorova, M. Torchiano and M. Morisio, "Actual vs. perceived effect of software engineering practices in the Italian industry", Journal of Systems and Software, vol. 83, (2010), pp. 1907-1916.

[117] S. Mohammadi, B. Nikkhahan and S. Sohrabi, "An Analytical Survey of" On-Site Customer" Practice in Extreme Programming", Computer Science and its Applications, International Symposium on CSA'08, (2008), pp. 1-6.

[118] J. Koskela and P. Abrahamsson, "On-site customer in an XP project: empirical results from a case study", Software Process Improvement, Springer, (2004), pp. 1-11.

[119] P. S. Grisham and D. E. Perry, "Customer relationships and extreme programming", ACM SIGSOFT Software Engineering Notes, (2005), pp. 1-6.

[120] K. Conboy, "Agility from first principles: reconstructing the concept of agility in information systems development", Information Systems Research, vol. 20, (2009), pp. 329-354.

[121] L. A. Griffin, "A customer experience: Implementing XP", XP Universe, Raleigh, NC, USA, (2001).

[122] C. Farell, R. Narang, S. Kapitan and H. Webber, "Towards an effective onsite customer practice", Third International Conference on eXtreme Programming and Agile Process in Software Engineering, Intelliware Development Inc., (2002), pp. 1-5.

[123] M. Korkala, P. Abrahamsson and P. Kyllonen, "A case study on the impact of customer communication on defects in agile software development", Agile Conference, (2006), pp. 11-88.

[124] G. Chin, “Agile project management: how to succeed in the face of changing project requirements”, AMACOM Div American Mgmt Assn, (2004).

[125] S. Bartsch, "Practitioners' Perspectives on Security in Agile Development", Sixth International Conference on Availability, Reliability and Security (ARES), (2011), pp. 479-484.

[126] M. Alqudah and A. Abdulsalam, "Basic English language tools for beginners: Using animations and audio", International Journal of Scientific Engineering Research, vol. 4, no. 4, (2013), pp. 228-233.

[127] A. S. Koch, "Agile Software Development: Evaluating the Methods for Your Organization”, Artech House, Inc. London, (2005).

[128] R. Coffin, "A tale of two projects [agile projects] ", Agile Conference, (2006), pp. 7-164.

[129] G. H. Hofstede and G. Hofstede, “Culture's consequences: Comparing values, behaviors, institutions and organizations across nations” Sage, (2001).

[130] K. V. Siakas and E. Siakas, "The agile professional culture: A source of agile quality", Software Process, Improvement and Practice, vol. 12, (2007), pp. 597-610.

[131] W. E. Schneider, “The reengineering alternative: A plan for making your current culture work”, Richard D Irwin, (1994).

[132] M. Sahota, "An agile adoption and transformation survival guide: working with organizational culture", InfoQ, http://www.infoq. com/minibooks/agile-adoption-transformation, (2012).

[133] C. Sepulveda, "Agile development and remote teams: learning to love the phone", Proceedings of the Agile Development Conference (ADC), (2003), pp. 140-145.

[134] C. Tolfo and R. S. Wazlawick, "The influence of organizational culture on the adoption of extreme programming", Journal of Systems and Software, vol. 81, (2008), pp. 1955-1967.

[135] H. Berger, "Agile development in a bureaucratic arena - A case study experience", International Journal of Information Management, vol. 27, (2007), pp. 386-396.

[136] S. Ambler, “Scaling Agile: The Software Development Context Framework”, Available: http://www.disciplinedagiledelivery.com/sdcf/, (2013) May 10.

[137] M. Alqudah and R. Razali, "A Review of Scaling Agile Methods in Large Software Development", International Journal on Advanced Science, Engineering and Information Technology, vol. 6, no. 6, (2016), pp. 828-837.

[138] A. A. Hamid and Z. Mansor, "Client’s Readiness Assessment Success Factors for Outsourcing Software Projects", International Journal on Advanced Science, Engineering and Information Technology, vol. 6, no. 6, (2016), pp. 220-227.

[139] R. H. Al-Ta’ani and R.Razali, "A Framework for Requirements Prioritisation Process in an Agile Software Development Environment: Empirical Study" International Journal on Advanced Science, Engineering and Information Technology, vol. 6, no. 6, (2016), pp.846-856.

[140] J. Highsmith, “Adaptive software development: a collaborative approach to managing complex systems”, Addison-Wesley, (2013).

[141] R. Tufail and A. Malik, "A case study analyzing the impact of software process adoption on software quality", 10th International Conference on Frontiers of Information Technology (FIT), (2012) Dec 17, pp. 254-256.

[142] S. Biazzo and G. Bernardi, "Process management practices and quality systems standards: risks and opportunities of the new ISO 9001 certification", Business Process Management Journal, vol. 9, (2003), pp. 149-169.