Impact and Challenges of Requirement Engineering in Agile Methodologies: A Systematic Review

Sehrish Alam  
Department of Software Engineering  
Bahria University Islamabad, Pakistan

Shahid Nazir Bhatti  
Department of Software Engineering  
Bahria University Islamabad, Pakistan

S. Asim Ali Shah  
Department of Electrical Engineering  
Bahria University Islamabad, Pakistan

Dr. Amr Mohsen Jadi  
Department of CCSE,  
University of Hail, KSA

Abstract—Requirement Engineering is one of important stage in development life cycle. All requirements required for development of product is collected in this phase. A high standard product can be developed by agile methodology in less budget and time. Importance of agile practices have been enhanced since it offers assist cooperation too software engineering. Being basic phase of software engineering, requirement engineering has different processes. The elements of direct correspondence is one of sly way which not at all likely to other conventional and traditional approaches. Although a lot of research has been done on agile practices and role of requirement in agile methodologies but still there is need of studies on change manage management requirement prioritization, prototyping and nonfunctional requirement in agile methodologies. Aim of this review paper is to present the limitations in presentation of requirement engineering phases in agile practices and what are the issues and challenges that agile person faces in implementation of agile practices. Many research studies from different sources have been reviewed on basis of inclusion and exclusion criteria. Most RE activities has been discussed in review. Evidence helps to prove that how RE process was performed in scrums. Mostly research has been conducted on general agile methodologies, few authors specified RE practices in other methodologies of agile. Finding of this research is the work of researchers that will be beneficial for those who are interested in finding interesting area of research in this field because many techniques of agile (extreme programming, crystal methodology, lean) requires further study and practical results as clarified by studies.

Keywords—Requirement Engineering; Traditional approaches; Agile methodologies; Challenges in RE; Requirement prioritization; Nonfunctional Requirements; Dynamic system development method; Scrum; Extreme programming

I. INTRODUCTION

A. Requirement Engineering

Before broadly descriptive, let us have a debate with respect to requirement engineering. It is basically a procedure including documentation, legitimate association, and collection of all necessities of a framework. In this practice, the product need is distinguished; without this procedure, making extraordinary progress in providing efficient software might be incomprehensible. Despite the fact that process of requirement engineering requires a lot of time, it is the best way to get best out of likelihood for the efficient product to meet the conditions and requirements [1]. In this manner, requirement engineering performs a key part in development of software. Requirements Engineering (RE) is a main area in software production and engineering. A high quality RE procedure often dominates an effective project. Traditional Software Engineering (SE) usually includes the RE process which consist of requirements elicitation, analysis, documentation, management and validation [2]. According to Kotonoya et al. most issues associated to the RE are resultant from the inadequacy or irregularity of specifications and requirements and conflicts between product stakeholders [3]. Effective RE requires a better consideration of the domain area, the circumstances in which the framework will run, and requirements of the product's stakeholders including developers, designers, end users and customers [4].

B. Agile

Definitions of Agile development software grown in the 90s as a response against conventional methodologies that were regarded as substantial, rigid and not supportive for essential tasks of developer. In 2001, research of the Agile Manifesto [5] was a landmark for development of Agile Software. Practices were proposed by 17 different groups for improvement in software development by using practices and methodologies of Agile. Purpose of Agile development methods was to provide product within budget, on schedule, satisfaction of customer and high quality. Agile methodology includes many techniques and methods. Frequently used methods are Scrum, Extreme Programming, Adaptive Software Development, Dynamic Systems Development Method and crystal family.

1) Extreme Programming (XP): Extreme Programming (XP) depends on values of correspondence, simplicity, courage, and criticism [6]. Purpose of XP is to associate old and tested methods in manner that they strengthen each other.
Whole team works together by using practices and feedback makes it possible to know status of work. XP discusses general practices and processes of software product development and what could possibly be done during procedure, it does not unequivocally discuss requirement techniques in detail. A few extreme programming techniques (or methods utilized as a part of these practices) can be contrasted and compared with minor adapted requirements procedures. XP utilizes story cards for requirements gathering. A user story describes business value feature to clients. Use cases are depiction of interactions of users and system and don't required need to give business value. Before writing of story cards, clients need to consider what they anticipate that the system will do (brainstorming). When we think about specific functionality, more ideas and stories are emerged. Customer prioritizes stories on the basis of implementation efforts and time for next iterations. Comparison of extreme can be done with evolutionary prototyping and requirements. The distinction amongst XP and prototyping is that XP requires mature substantiated code while prototypes can be “threw together”. Clients can survey and test the usefulness and outline of the delivered software and examine issues that need to be changed or included in next iteration.

2) Scrum: Scrum is a used for management of development process by applying concepts on adaptability, efficiency and flexibility from industrial process control hypothesis. Scrum concentrates on team work and product quality in versatile environment. [7, 8]. Sprints, product backlog and daily scrums are core techniques of Scrum. Product backlog plays distinct role in scrum with respect to requirement engineering. Necessary requirements for product, prioritized functions, features and bugs are part of Product backlog. The software product backlog accumulation can be contrasted and compared by inadequate, versatile and implementation document. For each sprint, tasks having high priority are added to sprint backlog without any alterations, however requirements can be reprioritize for next iteration of sprints. Ultimately a meeting is held on sprint review that exhibits the innovative requirements to the client and requests input. All information assembled from the meeting on sprint review and product backlog is utilized in next meeting of sprint. The sprint audit meeting can be contrasted and compared to review of requirements and evolutionary prototype. Fig of frame work is given below[9].

3) Crystal: Crystal Methodology are a group of various systems from which the proper techniques can be selected for each software product/project. Followers of this family can be deviate to fit in different circumstances. Indication of "heaviness" is practiced by different color indexing of members of crystal family. Frequently used colors are yellow, clear, red, orange, blue, magenta and violet. [10]. Up to now three Crystal procedures (blue, violet and magenta) have been utilized. Some Orange policy standards and Crystal Clear can be analyzed and compared with RE methods given below.[9][30].

4) Dynamic System Development Method (DSDM): DSDM explicitly highlights the use of JAD sessions and emphasizes prototyping [11]. Dynamic Systems Development Method (DSDM) gives a structure for rapid development of application.[11]. Business study and feasibility study are two initial phases of DSDM. Only base requirements are gathered in these two phases and other requirements are gathered in
development phase. There is no restriction in use of any technique for development however, testing is practiced in duration of complete life cycle phase. Developers and users both are involved in incremental testing (functional, technical).

II. LITERATURE REVIEW PROCESS

Systematic literature review of related papers has been made as directed in B. Kitchenham et al [12] in 2008 shown in fig. A review has been planned to analyze the studies conducted in domain of requirement engineering in agile mythologies. Review has been made on basis of requirement techniques, agile practices, published year and methodology used so that research trend can be find out in this area. Keywords were identified for searching criteria of research and journal paper in this field .inclusion and exclusion criteria of papers selected, research questions are identified for analysis of research.

A. Review Planning:

A literature review was planned out in order to analyze the study in different phases of software requirements with the help of agile methodologies and challenges.

B. Keywords Identification/Selection:

Research keywords were identified related to agile practices such as agility, agile, Scrum, XP ,crystal programming, usability, requirement management in agile, requirement quality, change management in agile and requirement engineering. These keywords were searched in combination of each other and single terms.

<table>
<thead>
<tr>
<th>Requirement Engineering</th>
<th>Requirement Elicitation, Analysis, Requirement Management, Requirement Validation, Usability, Change management, Traditional agile approaches, traditional model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agile Approaches</td>
<td>Agile software development, Extreme programming, crystal programming, lean, crystal programming, traditional model, Agile in large organizations, advantages and disadvantages of agile, challenges and Dynamic Systems Development Method</td>
</tr>
</tbody>
</table>

Fig. 3. Review process
After identification of keywords, search engines were selected for searching review studies. IEEE Explorer, Science Direct, Google Scholar, ACM, and Taylor & Francis are search engines used for review of papers and journals.

C. Inclusion and Exclusion:

After selection of keywords and search databases, selection of primary review is required. A lot of material has been published in this topic, so we need to refine the selected study by addition and removal of material so that the purpose of the study can be fulfilled. The reviews were chosen in the wake of taking after the incorporation and rejection criteria given below.

1) Inclusion: Inclusion criteria includes Peer reviewed study, relevant to search keywords, empirical study using agile methodologies, studies including requirement engineering practices and agile in software engineering as depicted in fig. 3.

2) Exclusion: After searching a lot of papers and review, we excluded irrelevant material on basis of exclusion conditions and criteria. Especially studies who don’t directly focus on Agile practices in requirement engineering, tutorials, editors, and weblinks are excluded. Figure of selected study is given below.

3) Research Questions:
- What are requirement Engineering practices used in Agile?
- What are issues, limitations and challenges in agile requirement engineering?

Complete selection process of literature review is given below.

III. LITERATURE REVIEW

A. RE practices in Agile:

Bose, Kurhekar and Ghoshal (2008) [13] focused modifications in requirements engineering to be implemented in agile practices. They proposed that stakeholders should be involved for different point of views for requirements gathering and interviews should be conducted. There should be verification and validation of early description of functional and non-functional requirements involved. Moreover, they specify that requirements traceability should be ensured by adoption of requirement management, which is critical when the requirements are probably going to be changed. We can get continuous feedback from customers by agile requirement; however, limits of agile practices are not well defined.

Silliti and Succi in 2005 [14] depict the approaches of agile to deal with requirements management and elicitation. Purpose of these practices is to delete out dated requirements and add new ones by continuous interaction with customers. Interaction between the Agile developers and the clients must be immediate and with no go-betweens and complete development team should be involved for requirements elicitation and gathering. In beginning of each iteration requirements can be changed and prioritized so conclusion is...
that requirements are effectively managed in small teams by agile practices and customer involvement reduces wastage of time in requirements during production.

According to Eberlein and Leite (2002) [15] the requirements are ineffectively handled in four agile practices- requirements management, validation and verification, nonfunctional requirements and customer interaction but can be ignored in agile domain. Requirement management is basis for managing changes in customer requirements so agile methods must ensure it. For improving quality of agile process requirements validation is applied with requirements verification. According to authors techniques are needed to identify nonfunctional requirements. [16]. However, more focused has been given to customer involvement and interaction but different techniques of interviews and context free questions should also be included.

Paetsch, Eberlein and Maurer (2003) [17] look at the distinctions and comparisons between Agile approaches and traditional RE approaches and attempt to discover what advantages of RE techniques can applied to Agile approaches. According to them there is least involvement of customers in traditional RE approach (only involved in early stages) whereas agile involves stakeholders throughout in agile process. They conceived that agile approaches are less documented and traditional approaches are well documented in detail. Traditional RE approach uses diverse levels of abstraction models while Agile uses throw away model. Agile methods use index cards for requirement management and change management has great importance in traditional approach. In their opinion, is that the RE stages are repeated and merged in most iterations and not clear in agile. Recommendations of authors are need of documentation in agile environment for future maintenance and implication of distinct phases of RE. [33]

B. Requirement Prioritization in agile:

The progress of innovation and business needs have activated client prerequisites to evolve almost each day. It is in this manner difficult to satisfy the necessities on the double. Agile practices have been acquainted with relieve such issues by actualizing client necessities incrementally and iteratively. Author has discussed about the variables that must be thought about while performing requirement prioritization in agile practices. This is essential as erroneous requirement prioritization may build the cost of advancement and prompt project and framework disappointments. In this review, the contributing components concerning viable requirement prioritization in writing were grouped through substance examination as a theoretical system. The studies demonstrate

Fig. 6. Frame work of requirement prioritization [18]
that there are three angles required in requirement prioritization prepare in lithe improvement [18].

Choosing a gathering of unmistakable necessities to be executed in iterations is considered as fundamental in agile practices. This can be accomplished by execution of requirement prioritization. Conceptual framework of requirement prioritization is depicted in fig [18].

Some of challenges in agile requirement prioritization are depicted in fig below.

---

C. Uses of Agile Requirement Engineering

Tomayko (2002) [19] perceives 12 extreme Programming groups and discovers that agile prototype answer queries about requirements content. Customers accordingly will recognize what else they need and dispose of the obligation of giving correct requirements. Another advantage of agile approach is that requirements (unstable state) are pre tested for detection of faults in extreme programming however, changes in architecture causes incorrect estimation and extra cost in requirements elicitation in agile environment.

16 software development establishments were examined by Cao and Ramesh (2008) [22] for advantages and disadvantages of 7 agile approaches. Iterative requirement engineering, face-to-face correspondence over specification in written form, utilization of review meetings and acceptance tests, prototyping, test bases development, constant planning for change requirement management and prioritization of requirements [25]. They dissect both of the advantages and difficulties of these practices and recommend that they are neither beneficial nor useless to establishments and can be used on basis of project specifications and conditions.

A goal sketching method was introduced by Harrison and bones in 2007 [24]. First they create unclear goal graph for basic understanding of functionality then development of stages similar to scrum started. Graph consists of many stages termed as stage graph and refines all stages of development. Stage graph is updated during stages and system with completed graph is updated at the end of last stage by using iterative developments. Testing of these practices has been done in industrial organizations.

Model of Agile Requirements Abstraction, a model of 4 abstraction level agile was presented by Birgisson, Svensson, Hedin and Regnell (2008) [26]. Feature level, product level, component level and function level are 4 levels from general to specific. Whenever Requirement engineering phases use ARAM model, these phases use three steps: Elicitation step, placement step and workup step. Collection of real requirements is included in elicitation, placement include identification of abstraction level of requirements and old and new requirements are combined by identity numbers in workup step. ID numbers are assigned with respect to algorithm in prioritization of requirements for scrum backlog. Documentation can be part of these steps if traditional methods are practice in this model. These agile practices are just assumed within the groups so the clients just observe the delivered outcomes as developments and don't need to be required in the RE procedure. Evidence shows that how RE process was executed in scrums. Most of research has been conducted on general agile methodologies, few authors specified RE practices in Scrum. Moreover the greater part of the accessible scholars discussed elicitation stage of requirement engineering, other phases requirement management, prioritization, validation and management are rarely cited.

D. RE Practices in Agile:

Following are requirement engineering practices used in agile methodology reflected in review study.
### TABLE II. REQUIREMENT ENGINEERING PRACTICE IN AGILE

<table>
<thead>
<tr>
<th>Requirement Engineering practices in Agile</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer requirements require user stories. It develops understanding b/w stakeholders. These stories has changed trend from documentation to discussion</td>
<td>Daneva, M., van der Veen, E, Amrit, (2013) Ghaisas, S., Sikkel, K., Kumar Bjarnason, E., Wnuk, K., &amp; Regnell (2010)</td>
</tr>
<tr>
<td>Continuous planning</td>
<td>Jun, L., Quizhen, W., &amp; Lin, G. (2010).</td>
</tr>
<tr>
<td>There is always continuous planning to adjust new upcoming requirements and change management.</td>
<td>Yu, Y., &amp; Sharp, H. (2011)</td>
</tr>
<tr>
<td>To write tests before functional codes, enhances feedback by test cases. Another approach is ATDD.</td>
<td>Abdullah, N. N. B., Honiden, S., Sharp, H., Nuseibeh, B., &amp; Notkin, D. (2011)</td>
</tr>
<tr>
<td>Acceptance tests and meetings are developed requirements and product backlogs .Meetings are held to review these backlogs</td>
<td>Abdullah, N. N. B., Honiden, S., Sharp, H., Nuseibeh, B., &amp; Notkin, D. (2011)</td>
</tr>
</tbody>
</table>
F. RE Issues/challenges in Agile Methodology:

Following are requirement engineering challenges that can be described in Agile.

<table>
<thead>
<tr>
<th>Features</th>
<th>Details</th>
<th>Challenge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct Communication</td>
<td>Communication between clients and teams members, minimum documentation in Agile.[20][21]</td>
<td>How to track changes in requirements?</td>
</tr>
<tr>
<td>Large interaction of clients</td>
<td>Client interaction is required for large amount of feedback</td>
<td>Large work load</td>
</tr>
<tr>
<td>Change requirements</td>
<td>Handle changes</td>
<td>Work again and again</td>
</tr>
<tr>
<td>Negligence of NFR</td>
<td>User stories just store functional requirements</td>
<td>Security and usability</td>
</tr>
<tr>
<td>Estimation of Budget, time</td>
<td>Cant estimate due to changes requirements</td>
<td>Overrun and high cost project</td>
</tr>
<tr>
<td>Innovation in requirement engineering</td>
<td>Creativity in requirement engineering</td>
<td>Creativity issue in agility and in release</td>
</tr>
<tr>
<td>Missing Requirements</td>
<td>How to discover missing requirements?</td>
<td>A large no of iterations for missing requirements</td>
</tr>
<tr>
<td>Conflicting requirements</td>
<td>Conflict and ambiguity in requirements</td>
<td>A large no of iterations</td>
</tr>
</tbody>
</table>

IV. ASSESSMENTS AND FINDINGS

Hasnain in 2010 [27] given a efficient review to describe customers, users, agile techniques and specialized issues recognized in published literature review from 2002 to 2008. According to them review demonstrates that requirement engineering methods are reviewed in generic agile approaches and these practices are not reviewed in specific agile methodologies like extreme programming, test driven development, crystal programing. Finding of Ramesh [28] in her survey prescribed that exploratory and experimental results are more required agile practices particularly in extreme Programming (XP) [16], crystal programing and Scrum [17].

Silva and Martin [31] provided systematic review about user centered design and agile practices integration. Usability issue, related to design of agile methodology was discussed in literature review [31]. Finding of review identified that agile team can resolve usability issue by adopting specialist of user-centered design. These practices have been identified in agile techniques like big design up front, little design up front and testing of user.

Barlow and Lowry [32] valued the impact of the use of agile practice in large level projects. The survey gave the design framework of a system that conveys methodologies to large projects for implementing agile practices. The survey outcomes help the designers and developers to implement development practices in large organizations. Agile performance in three constraints time, scope and cost is depicted in Fig.7 [9]. Scope of Agile is most beneficial in IT organizations as shown in fig.

![Fig. 8. Agile in triple constraints [9]](image-url)

Usability, performance and security of system described by nonfunctional requirements are less focused in Agile practices NFRs that decide the ease of use, security and execution of the framework are anticipated to find solution of issue in agile practices. [31,32]. Most of authors concentrated on specific nonfunctional requirement e.g. security [33][34] and propose structures for supporting elicitation of security requirements while others referred modeling framework as component of requirement engineering in order to integrate nonfunctional activities into agile practices.

V. CONCLUSION AND FUTURE WORK

In the previous decade, agile practices has picked up a huge worldwide recognition related to software development because of its center idea of esteeming people and connections, working program, client cooperation, and reacting to change. We firmly trust that agile consummately fits the changing way of RE, along these lines bringing lithe into the universe of RE will ensure a more noteworthy and a speedier achievement. Theses research papers represent a systematic review of agile methodologies, its limitations and challenges in requirement engineering. Systematic literature review of related papers has been made as discussed in B. Kitchenham et al [12] in 2008. About more than 60 research papers were searched, exclusion criteria was applied to shortlist relevant research papers. Paetsch, Eberlein and Maurer [17] look at the distinctions and comparisons between Agile approaches and traditional RE approaches and attempt to discover what advantages of RE techniques can applied to Agile approaches RE stages are repeated and merged in most iterations and not clear in agile. Recommendations of authors are need of documentation and prioritization in agile environment for future maintenance and implication of distinct phases of RE Finding of this research is the work of researchers that will be beneficial for those who are interested in finding interesting area of research in this field because many techniques of agile (extreme programming, crystal methodology, lean) requires further study and practical results.
as clarified by studies. When processes are not well defined and clarified, conflicts in requirements causes problem. There is need of proper framework for prioritizing requirements in agile environment. Nowadays, Organizations are more focusing on providing requirements focus on value of customers and organization viewpoint because of time limitations, concentrate on cost saving and satisfaction of consumer By prioritizing requirements, business value based on customer needs can be established. There is need of process with defined criteria and process that can help in well prioritized backlog.

REFERENCES
[38] P. Jain, L. Ahuja and A. Sharma, "Current state of the research in agile quality development," 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, 2016, pp. 1177-1179
doi: 10.1109/ICWR.2016.7498453

[41] Shahid N. Bhatti, Maria Usman, Amr A. Jadi, 2015, Validation to the
Requirement Elicitation Framework via Metrics. ACM SIGSOFT
Software Engineering Notes 40(5): 17, USA.

[42] N. Kukreja, S.S. Payyavula, B. Boehm, S. Padmanabhuni, Selecting an
Appropriate Framework for Value-Based Requirements Prioritization: A