Volume 5, Issue 2
April 2016

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

# Comparison of Traditional RE with Agile RE

[1]**Nosheen Khan**
Internal Islamic university, Islamabad, Pakistan
Department of Computer Science,
E-mail: [1]nosheen.msse357@iiu.edu.pk

**2Ahthasham Sajid**
Department of Computing,
SZABIST, Islamabad, Pakistan
E-mail: [2]gullje2008@hotmail.com

[3]**Natasha Khan**
Department of Computer Science
COMSATS Institute of Information Technology,
Islamabad, Pakistan
E-mail: [3]Kevaa.92@gmail.com

## ABSTRACT

The use of agile practices is growing rapidly in the field of requirement engineering. To compare Agile RE with traditional requirement engineering, a systematic literature review is performed. The review highlights positive and negative aspects of requirement engineering in agile. The focus was on the two leading methodologies of agile i.e. SCRUM and XP comparing with iterative model from traditional models. The findings of the study would be helpful for the readers to understand the benefits and challenges of application of RE through agile methods.

**Keywords:** Requirement engineering, agile software development, agile methodologies, SCRUM, XP, Iterative.

## 1. INTRODUCTION

Requirement engineering is a process in which requirements of a system are identified, analyzed, documented and validated. It consists of multiple phases i.e. requirement elicitation, analyzing requirements, requirement specification, requirements validation and management. [2], [3].

Requirement elicitation is the process of discovering the systems requirements from users/customers. Requirement analysis is the process that determines the consistency, completeness, feasibility and level of conflicts among different requirements. In requirements documentation phase we document the requirements and it acts as a source of communication between different stack holders. In requirement validation phase we check the current set of requirement is the perfect description of the system to be developed or further changes are still needed. The purpose of requirement management phase is to record, document, distribute and manage information. [3]

Agile is relatively a new approach for developing software. It is an incremental and iterative process for developing software by self-organizing teams, which produces high quality software within time and budget. It deals with the changing needs of stakeholders [8].

SCRUM is one of the most popular agile methodologies. It is a process that consists of short phases called “sprints” that usually lasts from one to four weeks. The process is performed by small teams, where tasks are placed in product backlog. The items in product backlog are prioritized at the start of each sprint. The product backlog items with highest priority are refined and effort for each task is estimated. The customer is involved in sprint meetings but they are not permitted to influence the team. During each sprint, a short meeting named as daily SCRUM meeting is conducted to discuss the daily work plan [13].

Extreme Programming (XP) is a methodology of agile software development. It focuses on improving software quality and responding to frequently changing requirements of customer. The main elements of XP are programming in pairs, extensive code reviews, unit testing of all code, clarity in code, the frequent communication with the customers and among programmers. [12] The traditional requirement engineering relies on documentation while in agile approach focus is on communication between stockholders and the development team to achieve the desired goals.

The process of software development has been changed in recent years. Software is now involved in every field, and there success is also dependent on software quality. So it becomes important to handle requirement changes more flexibly for that reason agile and incremental developments are adopted. [7]

In iterative development software is developed in iterations. A component is implemented, tested and then reviewed by the customer. With passage of time it is refined. With each iteration software is improved. [4]

This paper compares the traditional requirement engineering approaches to requirement engineering methods used in agile software development. The objective of this study is to identify the positive and negative aspects of RE using agile with the help of literature.

## 2. LITERATURE REVIEW

The positive and negative impacts of agile practices in requirement engineering have been identified. In agile, requirement engineering is carried out informally. Requirements are not documented but gathered in the form of user stories.

### A. Comparison of TRE and ARE Practices

In study [1], one of major advantages of agile is that the requirements are reviewed with customer before each release as a result reduces the chances of changes in the next release, hence reduce the rework. Another advantage is that multiple tasks like requirement gathering, implementation and documentation all are performed by a single person so it maintains a consistency. Further the results showed that development is monitored with requirements which makes the updation of the product easy when the requirements are changed. Another benefit is that the gap between engineer and business roles are reduced by user stories. It captures the users expectations in such a way that the captured requirements are of good quality is. The results of scope planning revealed that there is a risk of over scoping in using the agile approach. A side effect revealed by some interviewee was that system is completed in later stages of the project and hence system issues are controlled lately. Further the results revealed that there exists a communication gap in agile because sometimes the customer representative is unavailable.

In [4] Agile RE is compared with traditional RE, where literature review was done after conducting a case study to check which methodology is better. Agile RE and traditional RE are compared by selecting two frameworks one from agile requirement engineering i.e. agile collaborative and innovative framework based on XP and SCRUM and other from traditional i.e. a linear iterative traditional requirement engineering model . There are three phases in iterative traditional RE framework; elicitation, specification and validation. In this framework customer only participate in elicitation and validation phases. SCRUM and XP methodologies are used in agile framework.

The SCRUM - SCRUM teams work together to divide user stories. And then note down their decisions same as traditional process. Paper [10] summarizes the agile RE practices and then a comparison is made with traditional RE practices. Furthermore an evaluation of agile requirement engineering applicability in modest-sized projects is discussed.

For the summarization of agile practices, twenty developers were interviewed from eleven software houses. As a result following agile practices came into account. Iterative RE, rapid change of requirement priorities, face to face meetings, plans for managing requirement changes, inspective meetings, prototyping, testing drives development and reception testing. A comparison took place among these practices and the traditional Requirement engineering practices. Results are as follows.

a. Requirement Capture

In requirement capture, there is interactive communication among developers and client but more often in ARE. In TRE the client should know all the requirements initially otherwise he feels hesitant. On the other side in ARE, due to face to face communication there is trust among both parties.

b. Requirement modeling

In ARE the prioritization of requirements takes place before each cycle whereas in TRE prioritization takes place only once. In TRE models of requirements are documented but not in ARE. Requirement prioritization is done on the basis of business values, risks and cost in TRE. But in ARE, prioritization is done only on the basis of business values.

c. Requirement Specification

In ARE, specification of requirements is not done carefully and very small level documentation is performed. On the other side in TRE, a document is formed having precise and complete requirements. But this is detailed documentation which results in difficulty in controlling scope and increase in cost.

d. Requirement Verification

Advance testing and review meetings are done for the verification of requirements in ARE. To check whether development is going on target or not, prototyping can also be used.

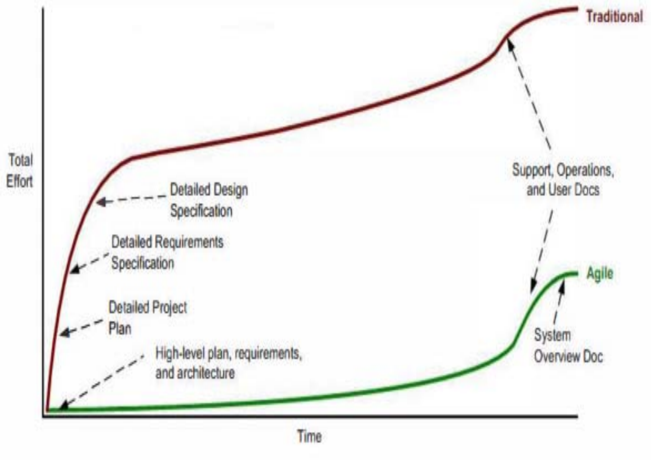
Some features of medium sized projects are discussed in next section and to check whether ARE practices map the medium sized projects, mapping to ARE practices is performed. Following are the features of IS projects,

- As these projects are initiated by modest-sized companies, they have limited budget.
- For the purpose of clear requirements, these companied initiate new projects as there is deficiency of existing systems.
- These systems are supposed to be developed fast or quickly because mostly they are proposed to solve urgent problems.
- As these systems are made on urgent basis, requirements come in small chunks and are not clear at first. Requirements may get changed at times.
- There is a strong bonding and well coordination among the staff of modest-sized companies as they are less in number.

B. Companies Shifting To Agile Methodology

Some reasons found in this literature review shows why most of the companies are switching from traditional requirement engineering to agile are:

- The traditional methodology focuses on planning while agile methodology is flexible and focuses on changes and user needs.
- Due to the flexibility of agile approach customer satisfaction level increases.
- In traditional approach a huge documentation is done and the final product is done at the end. Sometimes the documentation is totally irrelevant to the product. But in agile approach the documentation is created at the end so that is more relevant to the final product.
- Delivered product satisfies the customer needs in agile approach because customer participation is very important.



]

Fig. 1. comparison of traditional and agile development.

Furthermore study [5] shows how Agile approach for RE is different from traditional approach for RE, and explains the advantages and disadvantages using these practices. A case study was performed in sixteen companies that uses agile practices.
Following results were concluded from the case study.

a. Face-to-face meetings

In agile approach for RE the ideas are communicated between customer and developer in face to face meetings that carries out throughout software development process. One of the major advantage of this practice is that customer can guide the developers for the changes they need in the requirements. Unnecessary documentations are ignored saving the time.

A major drawback of this practice is that it heavily depends upon the communication and collaboration between customer representative and developers. It involves many factors like availability of customer, customer group's agreement and trust and reliance between developers and customers. Many companies revealed that customer availability is one of the major tasks in agile methodology. For that reason mostly the project manager act out as representative of customer.

Also the agreement between different customers is hard and time taking process because they all have different interests. Another issue is of customer's familiarity with traditional RE approach, they might not feel comfortable with agile approach of RE.

b. Iterative process of RE

In the case study 14 companies faced the problem of emerging requirements in implementation phase that were not defined earlier. Few requirements of high level were predefined. The reasons behind this problem might be continuous requirement changes, less domain and technology knowledge, and the customer who is unable to explain the requirements but once they see they can clearly explain what they actually need.

The major advantages of iterative process of RE are that the requirements gathered through this process are more understandable and are of good quality because of customer involvement at every stage. As a reason it forms a good and comfortable connection between customer.

Major issues arose by this practice are; time and budget estimations are difficult, less documentation that cause problems when system complexity increases or workforce turnover. Another issue is that NFR's might get ignored as their handling is important at the earlier stages. Only usability factor is assured because customers are involved.

c. Requirement Prioritization Process

In agile approach requirements are prioritized at the start of each cycle and those prioritized requirements are developed first these prioritizations are done depending on customers decisions about business. Whereas requirement prioritization is done only once in traditional approach of RE depending on many factors.

The positive point of agile is that customer's priorities are clearly understood as a result customers' expectations are fulfilled as well as there is a high probability of requirements reprioritization.

A major drawback is that prioritization of requirements are only dependent on business values that might ignore other important factors.

d. Requirement Change Management Planning

In agile methodology requirement changes are managed properly and easily because planning is done before each iteration. As compared to traditional approach of RE less cost is spent on changes because early validation is done that reduces the change requests.

e. Prototyping

Errors are reduced as requirement prioritization list is shown to the customer representative. After every cycle software increment is shown to the customer representative that is just like prototyping. It is helpful for those for projects that needs marketing. But a major drawback is that sometimes the customer considers the prototype as final product and enhances their demands.

f. Development Based On Testing

In this practice test cases are written before implementing the actual code. The advantage of this approach is that changes can be easily incorporated and quick response can be taken in the case of wrong assumptions. But a major issue that arises is mostly implementers do not have experience of writing test cases before the actual code. Another problem is that requirements should be thoroughly known.

g. User acceptance testing and review meetings

In agile approach for requirement engineering, requirement validation is done at the end of each iteration, for that review meetings are held between customer representative and implementers. The main objective of such meetings is to take response from the customer about developed functionalities and to review them. This helps to make sure that project is according to its scope and requirements, satisfying customer's expectations, and to control the issues in the early stages.

Sometime companies adopt agile methodology due to restrictions in traditional methods. Why companies opt to switch from traditional software development to agile software development is discussed in [6]. The main purpose of the paper is to what decisions should be taken if the organizational size increases. The basic suggestion provided by the paper two or more agile approaches should be combined.

C. Issues with Agile Adoption

Below are highlighted the situations related to the process of RE where agile uses, how to adopt agile in those circumstances and the challenges that may arise by introducing agile.

a. Requirements as user stories

In agile, user stories is a well-known method for the purpose of describing user requirements. In this approach, user prioritizes the requirements according to business values. User stories are of great importance in the systems in which user is highly interactive e.g. mobile phones. But the problem is that mobile phones have a lot of customers so how to get all these customers involved in the development process and acquire stories from all of them.

After the collection of user stories another problem is to make a connection between the actual implementation and the user stories. These requirements can be converted into system requirements by agile requirement engineering. In paper [6], it is discussed that user stories can be easily converted in to system requirements because there is a little gap between them.

b. User Stories and allocation of work

Two types of teams exist in agile development, which include Component and feature based teams. A team that is most concerned with specific component is a component based team. These teams are divided with their architecture for the purpose of making their components discretely. The teams that do the scaling of agile development are the feature teams. [9]

Component teams are adopted when there is the case that a company has not well trained teams for handling components. Component teams require specialization in functionality so they do not work well.

Complex and large user stories need to be sub divided and then assign to different component teams. For the purpose of assigning different parts of user stories, a method of SCRUM - SCRUM teams is discussed in the paper. These teams include a representative from every component.

D. Architectural issues in Agile RE

This paper [11], discusses the agile requirement engineering and focus on different issues especially the architectural issues in Agile RE. To overcome these challenges it presents a methodology.

- The requirement engineering activities are not clearly defined in Agile RE.
- As requirements come in chunks so the interface between the requirement is missing. This may cause some re-work in next iteration because of some conditions imposed by newly arrived requirements.
- In Agile RE there is not a defined approach for eliciting non-functional requirements only usability is assured because customer is fully involved in development. As the NFRs are not clear so it is not possible to make architecture because usually NFRs are the base for software architecture.
- The user requirements are prioritized by customer before next iteration and the customer do not have architectural knowledge he prioritize only on the base of business value and do not consider what will be its impact on the architecture.

## 3. LIMITATIONS AND OPEN RESEARCH ISSUES

There are following limitations of this study,

- This study is based on case studies that cannot be empirically validated.
- Comparison of only XP and SCRUM methodologies is made.
- Due to less practical knowledge of author in this area conclusion are drawn just on literature review.
- Due to limited time, all research papers cannot be included.

The practices of agile requirement engineering need to be practically applied in different companies for the purpose of getting feedback and for finding out to what degree we can apply the agile methodology.

There is a lot more space in field of tailoring agile practices for different type of companies and products and to handle architectural related issues. A new model can be presented by combining agile methodologies for scaling up companies and projects.

## 4. CONCLUSION

In this paper a systematic literature review is conducted to find the differences between traditional requirement engineering practices and agile requirement engineering practices. Two methodologies of agile SCRUM and XP are selected for comparison with traditional i.e. iterative model. The research papers are selected on the basis that they cover all aspects of RE e.g. RE in large scale organizations, modest sized organizations, scaling up the RE practices and the architectural or NFRs issues. The literature shows differences of traditional RE and Agile RE and it also show what are the benefits and challenges of adapting agile RE.

The main differences of traditional RE and Agile RE are as follows,

- In agile RE the customer is fully involved while in traditional RE customer is involved only in some initial phases. So the benefit of agile RE is that it removes the uncertainty in final product and the customer satisfaction is ensured while the disadvantage is that it requires customer to be available when needed.
- In traditional RE precise documentation is conducted. While in Agile RE limited documentation is conducted. The benefit of documentation is that it helps in case the project is scaling up and it reduces the dependency on different personal of the project. While the disadvantage is that when the requirements are changed during project the documentation

becomes useless as it does not match final product. And also its time and cost consuming work.

- In agile RE requirements are modeled repeatedly while in traditional RE requirements are prioritized only once. So the changing values can easily be handled by changing requirements priorities in agile RE.

- In Agile RE the development is done in iterations so after each iteration the delivered component can be treated as prototype so it brings the benefits of prototyping but the disadvantage is that the customer treats it as final product and his/her expectations increases.

- In agile RE the non-functional requirements are not handled properly except usability because of customer involvement. It makes it difficult to build architecture of the system.

- In traditional RE requirements are prioritized based on many factors like time, cost etc but in agile RE requirements are prioritized based only on business value.

The literature review shows that in case of product lines where architecture plays basic role and huge customer is involved it is difficult to adopt agile RE. In modest sized companies it is best practice to adopt agile RE. When the company is scaling up it can adopt agile by combining its more than one methodologies. And to handle the architectural issues the agile methodologies can be tailored to add more phases related to architecture development.

## REFERNCES

[1]. E. Bjornson, K. Wunk, B. Regnell, “A Case Study on Benefits and Side-Effects of Agile practices in Large-Scale Requirement Engineering,” In Proceedings of the 1st Workshop on Agile Requirements Engineering, p. 3. ACM, 2011.

[2]. Lisi Romano, Breno, and Alan Delgado Da Silva. "Project Management Using the Scrum Agile Method: A Case Study within a Small Enterprise."Information Technology-New Generations (ITNG), 2015 12th International Conference on. IEEE, 2015.

[3]. F. Paetsch, A. Eberlein, and F. Maurer, “Requirement Engineering and Agile Software Development” In Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003. WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on, pp. 308-313. IEEE, 2003.

[4]. A. Batool, Y. Hafeez, B. Hamid, S. Asghar, M. Riaz, M. Mukhtar and M. Ahmed, “Comparative Study of Traditional Requirement Engineering and Agile Requirement Engineering”. 15th international conference on, pp 1006-1014, IEEE, 27-30 January 2013.

[5]. L. Cao, B. Ramesh “Agile Requirement Engineering Practices: An Empirical Study.” Software, IEEE 25, no. 1 (2008): 60-67.

[6]. J. Savolainen, J. Kuusela, A. Vilavaara, “Transition to Agile Development- Rediscovery of Important Requirement Engineering Practices.” In Requirements Engineering Conference (RE), 2010 18th IEEE International, pp. 289-294. IEEE, 2010.

[7]. Batool, Asma, et al. "Comparative study of traditional requirement engineering and agile requirement engineering." *Advanced Communication Technology (ICACT), 2013 15th International Conference on*. IEEE, 2013.

[8] Kapitsaki, Georgia M., and Marios Christou. "Where is Scrum in the current Agile world?." *Evaluation of Novel Approaches to Software Engineering (ENASE), 2014 International Conference on*. IEEE, 2014.

[9] C. Lrman, B. Vodde, “Feature Teams” http://featureteamprimer.org/ last accessed April 29, 2013

[10] L. Jun, W. Qiuzhen “Application of Agile Requirement Engineering in Modest-sized Information System Development”, In Software Engineering (WCSE), 2010 Second World Congress on, vol. 2, pp. 207-210, IEEE 2010.

[11] W. Helmy, A. Kamel, O. Hegazy, “Requirements Engineering Methodology in Agile Environment”, in IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 5, No 3, 2012.

[12] Münch, Jürgen, et al. *Software Process Definition and Management*. Springer Science & Business Media, 2012.

*[13]. Sutherland, Jeff, and Ken Schwaber. "The SCRUM guide." The Definitive Guide to SCRUM: The Rules of the Game (2011).*