

Requirement Engineering in Agile

Tazeen Fatima ^a, Waqas Mahmood ^a

^a Department of Computer Science, Institute of Business Administration - IBA, Karachi, Pakistan

Received: 08 January 2019; Accepted: 20 June 2019; Published: 08 July 2019

Abstract

Agile focuses on delivering small increments of the project quickly. Unlike, in traditional practices, project development is initiated after gathering complete requirements. There are different techniques that are used for requirement engineering in agile. Few techniques includes, Extreme programming, Scrum and Feature-Driven Development. This research discusses the problems faced in requirement engineering in agile. Requirements variability and team size are few such problems faced in agile. A comparison of traditional and agile practices is presented in later sections through survey. The survey was filled by IT professionals in Pakistan and results were analyzed. The results indicate that requirements variability causes main problem in requirement engineering process. In addition, good customer relationships help to gather requirements effectively. Teams working in agile are quite satisfied by the customer as compared to teams implementing traditional practices. Moreover, the ways through which efficiency can be attained in requirement engineering is presented. There are different techniques such as, use cases and prototyping that can be used for requirement engineering for efficient results.

Index Terms: Requirement Engineering, Agile, Traditional, Scrum, Extreme Programming (XP)

© 2019 Published by MECS Publisher. Selection and/or peer review under responsibility of the Research Association of Modern Education and Computer Science

1. Introduction

Requirement engineering is a complete process of collecting and gathering requirements, providing its documentation and keeping track of the requirements. The activities that are involved in requirement reengineering are initiated by gathering requirements. Stakeholders and developers meet to discuss the requirements, features and functionalities of the project. The next activity analyzes the requirements and

* Corresponding author.

E-mail address: t.fatima.18162@khi.iba.edu.pk

negotiates with stakeholders to resolve any conflicts between them. Moreover, these requirements are documented, designed and presented as text or through graphical models. Lastly, it is important to manage the requirements, to check whether the requirements are being followed as they should be or not [1].

Agile development works with continuous user collaboration and interaction. Project features are delivered quickly and changes are accepted quite often. Hence, requirement engineering is different in agile as compared to traditional methods. Moreover, with the increase in usage of agile methods in organizations, it is important to find ways for gathering and managing requirements effectively. According to a research, 69% of the organization in software and IT industries follow agile practices [2]. Since agile methods follow an adaptive approach and deliver products in iterative manner. It is difficult to manage requirements for large scale teams because involving them continuously for multiple iterations is tough [2].

Requirement Engineering focuses on gathering requirements before initiating a project and relies on strong documentation [3]. While according to agile manifesto, agile projects focus on delivering working software rather than detailed documentation. Project development starts after getting minimal set of requirements [4]. Unlike in traditional methods such as, waterfall, once complete set of requirements are discussed then project development is initiated. In agile, gathering requirements is a never ending process. Changes in requirements and specifications can be adapted in later stages of development. Therefore, it is difficult to manage and gather requirements in agile [2, 3]. In order to manage requirement engineering in agile, various methods and techniques are used. These techniques will be discussed in the later sections.

Next section discusses the background of requirement engineering and its activities. After that literature review is presented which will focus on the ways through which requirement engineering can be done in agile with the reference of research done already. Furthermore, it will compare requirement engineering in traditional and agile developments. Later research methodology is explained and its responses are analyzed. The responses of current research are compared with the previous one. Lastly, conclusion of the research is summarized.

2. Background

There are few methods that are used for requirement engineering in agile as described in [2]. Firstly, Agile Modeling, it is a technique to develop design models that will serve as a part of documentation. Design models provide a clear picture about the requirements of the project. Secondly, Feature-Driven Development, in which team members makes and prioritizes a features list. A 30 minute weekly discussion analyses the status of each feature. Thirdly, Dynamic Systems Development Method, it analyses all features and presents a feasibility study about them, describing its need. It focuses on user involvement and periodic delivery. Furthermore, Extreme Programming is a famous agile technique based on small iterations and continuous user feedback. Lastly, in Scrum, work is divided in sprints and tasks are kept in backlog. Every day a 15 minute discussion is held to track the progress of each task and to discuss what tasks will be completed today. Reference [4] mentions Scrum as “empirical process control”. That is, at the end of each sprint, actual progress of the project can be seen. Additionally few methods are described in [3]. The Crystal Methodologies, among different methodologies appropriate methodologies are selected for any project. Moreover, Adaptive Software Development involves small, incremental and change tolerant cycles where customer also participates.

According to a research, almost 37% of the problems that occur in the project are related to requirement phases [2]. Hence, requirement engineering should be given good attention while developing a project. Agile and traditional development follows requirement engineering but the difference is when it should be done. Requirement engineering processes in agile development varies according to projects.

There are few activities for agile requirement engineering which includes Feasibility Study, Requirement Elicitation, Analysis, Documentation, Validation and Management [2, 3]. Firstly, Feasibility Study which

collects important project features. Also, it analyses if organization can handle the project and is it valuable enough to spend resources. Next, Requirement Elicitation is a process where stakeholders and agile team gather to discuss what the system should provide. There are few techniques that are followed which includes, interviewing the customer, brainstorming ideas for the project, use case analysis demonstrating user and system interaction in a particular scenario and ethnography. It is an observational practice for gathering information about how people work. In addition, [3] mentions Focus groups as another requirement gathering technique in which 4-9 users belonging to different backgrounds discuss what features should be in the project.

Furthermore, Requirement Analysis checks whether there are any ambiguous or conflicting requirements and resolves them. Some strategies include, Joint Application Department, Modeling and Prioritization. Joint Application Department is a workshop held while project is developed; it involves all stakeholders to discuss how to resolve conflicts. Modeling decreases the gap between the design and analysis phase. It is a pictorial representation of the project status. Prioritization analyses critical requirements so that they can be developed at highest priority. In Agile, requirements can be considered as prioritized stack, newly identified and previous requirements are implemented according to their priorities.

In addition, Requirement Documentation provides minimal documentation for the project. It records features on storyboards and use cases by assigning it to two members of the team for continuous documentation. However, minimum documentation can be a problem for long term projects. User story is an effective method to document requirements. As mentioned in [4], user story is briefly defines the requirement and estimates the effort required. A user story should be, Specific, Measurable, Achievable, Relevant and Time-boxed. Moreover, Requirement Validation, it includes, Acceptance Testing to validate if the system is according to user requirements. It consists of Requirement Reviews, Unit Testing and Evolutionary Prototyping. Evolutionary Prototyping starts from a simple feature and adds features according to priority with continuous user involvement. Lastly, requirement management controls and maintains interaction among stakeholders and project team members [2].

3. Literature Review

Agile methods have been successful as compared to the traditional requirement engineering. Reference [8] compares both approaches and provides an analysis on their success rate. In agile, changes in requirements can be done at any stage of the project. Whereas, in traditional RE requirements are gathered in the beginning and changes creates a gap between documentation and development. A survey conducted on opinions of the experts who have experienced both approaches, results that agile is more successful in terms of flexibility, developing good customer relationships and it's responsiveness to change [8].

There are various activities and methods used for Requirement Engineering in Agile projects. Reference [5] discusses different techniques that are used for RE and their critical analysis. It provides a literature review on different techniques proposed for RE in agile projects. RE can be effective in agile if automated tools and techniques are used. It is important to select RE technique that is suitable and already experienced in the similar scenario. User interaction, observation and ethnography are better approaches for agile projects where quick software delivery is essential [5]. According to survey presented in [5], combing different techniques can be productive. Techniques that involve prototyping increases project resources and are also limited for small to medium sized projects.

An effective technique discussed in [6] is crowd-based requirement gathering. It involves individuals from different backgrounds and age groups to gather requirements for a project. Priority of tasks and conflicts are resolved through voting or discussions with crowd. A survey presented in [6] concludes that crowd-based requirements increase creativity and diversity.

Project teams play an important part in requirement engineering. There are researches performed in order to identify and solve problems associated with large and distributed teams. Similarly, a research based on requirement engineering methodologies that can be used for large scale teams is presented in [7]. It is difficult

to manage agile projects with teams above 25 members. As there is less documentations and quick decisions are made, communication with larger teams can be a problem. In addition, ref. [7] discusses problems arises if teams are distributed and collated. Hence, it is better to restrict teams in agile to 15-20 members for effective results. Furthermore, [15] discusses challenges faced by distributed teams and how can they be solved. Problems arise due to variety in team members and cultural differences among them. The solutions identified are as follows; increase customer representative interaction, keeping all the team members on one page, every communication should be documented for future and tools and techniques should be used to keep the track of the project.

Moreover, research on the benefits and issues of agile practices in large scale RE is presented in [9]. One of the important issues is over scoping; that is project scope varies and lot of features are added continuously. These newly added features need to re-prioritize; hence, project does not get completed in the estimated time. Furthermore, customer is not always involved in discussions, and communication gap increases which lead to poor documentation. Therefore, finalize and document requirements at the time of implementation solve these problems [9].

Reference [10] proposes SCRUM methodology for agile project. Project is distributed in sprints, which consists of 2-4 weeks. Large projects can be divided into sub projects which can be handled by scrum teams. In addition, JAD technique and identifying viewpoints can improve efficiency in the project.

Mind mapping is an effective technique to gather requirements competently [11]. It identifies stakeholders and discusses their requirements for the project. These requirements will help in developing an initial product backlog. After which requirements can be prioritized and set for initial phase. When the initial phase is completed, more requirements and changes can be adopted [11].

Another approach for requirement engineering is goal-oriented requirement engineering, GORE, is important for software development process [12]. The success of the project can be determined by the fulfillment of goals of stakeholders. Hence, the first step of GORE is to identify the stakeholders. Next, determine user stories which will include features stakeholders' want in the system. Furthermore, high level, middle level and low level goals are gathered. The use of user stories showed productive results and teams were clear about stakeholders' expectations from the product.

Usability engineering, as described in [16] involves user in important stages of RE, that is elicitation, implementation and validation. Two agile approaches, XP and Scrum are compared with common usability engineering activities. Product Backlog in Scrum and user stories in XP is written by customer. For validating requirements customers should be shown early prototypes and mockups to take their feedback. Agile focuses in sharing the knowledge with all the members of the team rather than documenting them. Hence, knowing the complete picture of the project and using user stories for defining requirements ensures good project quality.

Ref. [17] provides an analysis on different traditional and agile methodologies such as waterfall, spiral, XP and Scrum. The analysis was based on five analysis techniques that is, CHAPL; Contextual, Historical, Analogy, Phenomenological and Linguistic. Contextual analysis defines how many resources are required for different practices in a methodology. Historical analysis identifies the changes that are made over the years in the methodologies. Analysis by analogy compares the practices of one methodology with another to explore and understand them better. A phenomenological analysis considers the experiences of software engineers who have worked on these methodologies. Lastly, linguistic analysis clarifies any physical or theoretical use of terms. Moreover, this research concluded that CHAPL analysis of any methodology helps to determine its efficiency according to a particular scenario.

Yaser Ghanam [18] provided an iterative model combining agile software development and software product line engineering. The first step is to identify team and their responsibilities. Next, develop tests according to user requirements, gathered by the customer and creating acceptance tests from them. In addition, acceptance tests will be re factored and categorized according to variability. Moreover, newly developed modules of code are added so that they can be reused in future. Finally this new model is incorporated in

system for development. Hence, this iterative product line model will help to develop requirements through reusing existing artifacts and ensuring flexibility for adding new developed artifacts.

4. Research Methodology

A research survey is presented in [13, 14]. It comprises on the different approaches of taken by traditional and agile companies for requirement gathering. The survey includes questions related to problems faced during requirement engineering and how much customer satisfaction is achieved. A similar research will be presented in this paper; the research will be done through questionnaires from people belonging to organizations following traditional or agile practices. The questions will categorize as; first, questions related to the organization's basic information. Next, what problems are arise while gathering requirements and to what extent they are solved in agile. Furthermore, how much customer satisfaction is achieved? Lastly, what, when and how requirements are gathered and how much requirements are vary. The questions included in the questionnaire comprise of 17 questions, 1-9 questions are taken from [13] and 10-17 questions are taken from [14].

The responses of the questionnaire will be analyzed to compare the success rate of the projects in traditional and agile environments. What changes can be adopted to solve the problem that exists.

5. Results of the Questionnaire

The survey was conducted among professionals working in IT industry which was made on Google forms. Total 50 responses were collected and their results are analyzed. The first part of the questionnaire consists of some basic questions regarding the organization.

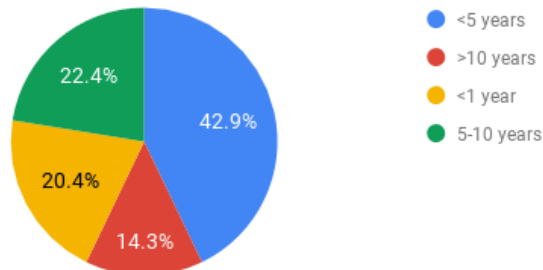


Fig 1. Since how long your organization is established?

The responses collected from individuals with 42.9% belonging to organizations established for less than 5 years, 20.4% were established for less than a year, 22.4% established among 5-10 years and 14.3% established for more than 10 years as shown in fig 1. The experience of the individuals working in the organizations was as follows; 46.9% employees had less than 5 years of experience, 32.7% had less than 1 year experience and 18.7% had 5-10 years' experience.

The number of employees in an organization affects the techniques used for requirement engineering [7]. The survey also analyses the number of employees in an organization to determine its consequences. Fig 2 represents, 10-50 employees were part of 42.9% of the organizations, 50-100 and less than 10 employees were in 20.4% of the organizations and greater than 100 employees exist in 16.3% of organizations.

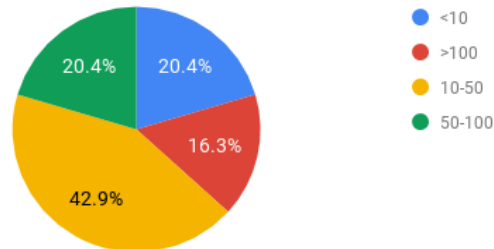


Fig 2. What is the total number of employees in the organization?

The second part of the survey consists of questions regarding the techniques used for requirement engineering, what problems are identified while gathering requirements and relationships with the customer. Among these responses 44.9% of the organizations follow plan based (Traditional) methodologies and 55.1% follow agile methodology as shown in fig 3.

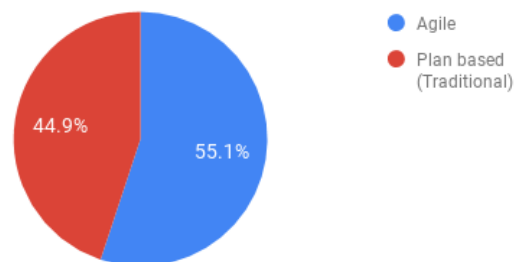


Fig 3. What methodology is followed for requirement engineering?

Most of the organizations faced difficulty to deliver the software with all required functions on time. Fig 4a shows among 22 organizations 10 of the organizations find it difficult to deliver all functions within deadline following a plan based methodology. In addition, 6 of the 22 companies find it difficult to provide excessive documentation of the code. And 7 companies had difficulty in managing relationships with customer.

Fig 4b indicates the problems that occur in agile software development. 14 of the 28 companies had difficulty in delivering software on time with all required functions. Moreover, 5 companies find difficult to manage relationships with the customer. Other problems include high competition and difficulty in managing relationships within the development group.

Requirement Engineering in Agile

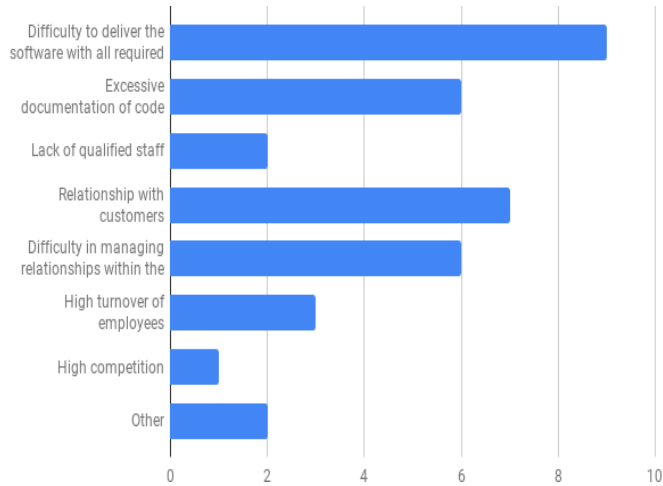


Fig 4a. What are the problems that occur in software development? (Traditional)

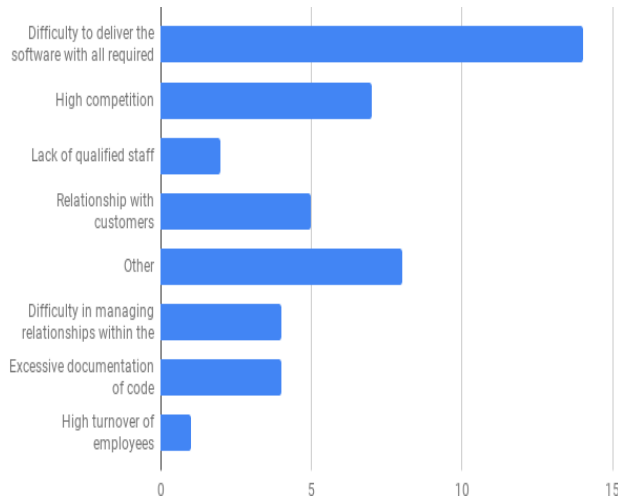


Fig 4b. What are the problems that occur in software development? (Agile)

The techniques used in agile like Scrum and XP lead to solve problems of difficulty in delivering software with all requirements and relationships with customers. Fig 5 represents the number of problems migrated after using agile in software development.

Another integral part of the requirement engineering is the relationships with the customer. Most agile and traditional companies focus on collaborative relationships with the customers. Fig 6 represents around 40% of the traditional companies have collaborative relationship with client and 27% have fixed contracts. In comparison, 53% of agile companies prefer client on site and 21% follow collaborative relationships with client. Client being on site can look into the progress of the project and make changes then and there. Hence, it makes team very satisfied with the client, fig 7 shows average satisfaction of clients achieved according to the relationship of the client.

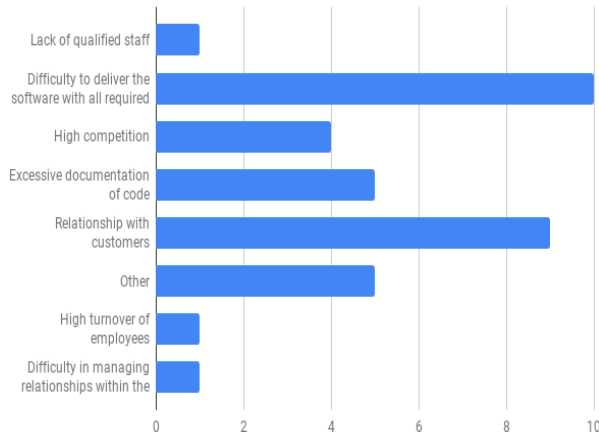


Fig 5. Software development problems mitigated by adopting agile

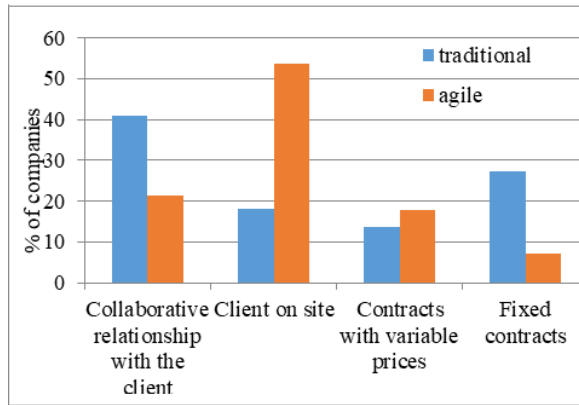


Fig 6. What is the type of relationship with customer?

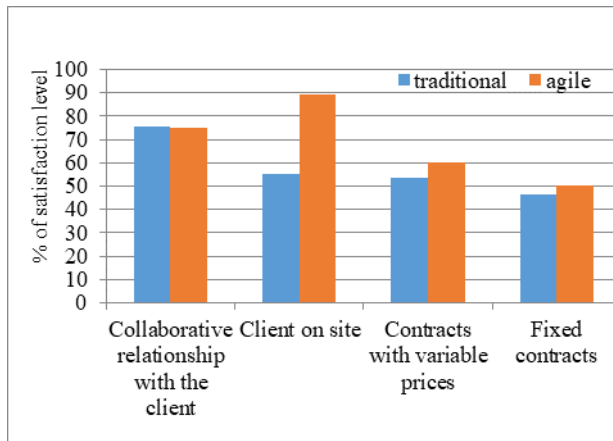


Fig 7. How much you are satisfied with the customer?

According to the survey conducted, that is 55% of the organizations face major customer problems due to variable requirements. 24.5% of organizations customer problem exists due quick delivery of the product as shown in fig 8. Similarly, another research [9], discusses that changing requirements leads to change scope and causes project failures.

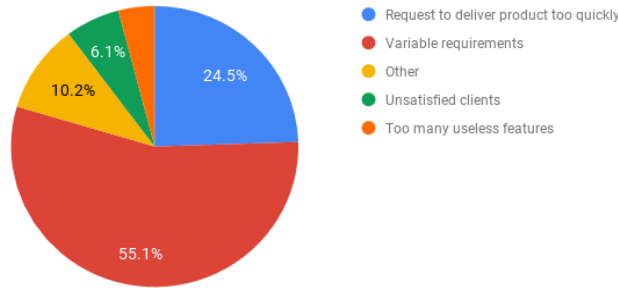


Fig 8. What constitutes the most important customer problems?

Fig 9 shows the frequency of variability in requirements in a company. Requirements often changes in 46% of agile companies and in 54% of traditional companies. Fig 10 represents the main reason for the requirement variability, improvement and deepening of knowledge results in 42% of the variability in requirements. Moreover, discovery of error in a project constitutes to 22% of the variability.

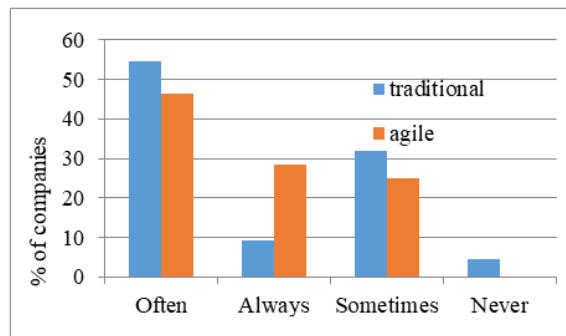


Fig 9. Frequency of variability in requirements

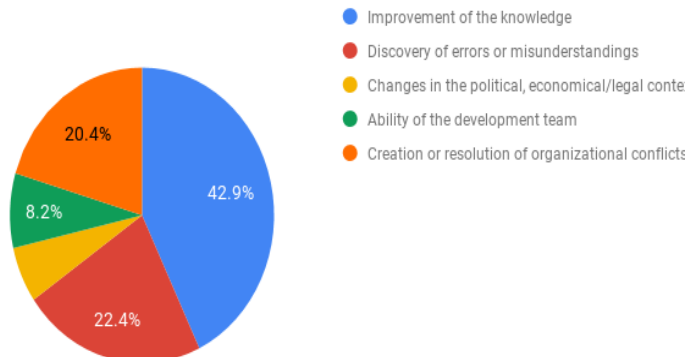


Fig 10. Main reason for requirements variability

The next part of the survey analysis the complete requirement engineering process. When, how and who gather requirements, also what is the goal of gathering requirements. Fig 11 shows the percentage of companies gathering requirements initially or continuously. 62% of agile companies gather requirements continuously, whereas, 72% of traditional companies gather requirements initially.

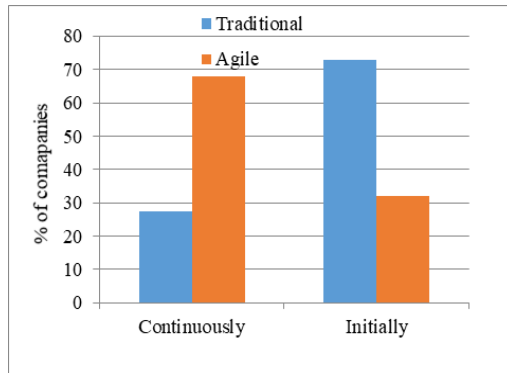


Fig 11. When requirements are gathered variability?

How requirements are gathered and what techniques are used in the organizations is shown by fig 12. 40% of the traditional companies use interviews to gather requirements and 36% use focus groups and brainstorming techniques. While, 57% of the agile based companies follow use cases and prototyping for gathering requirements.

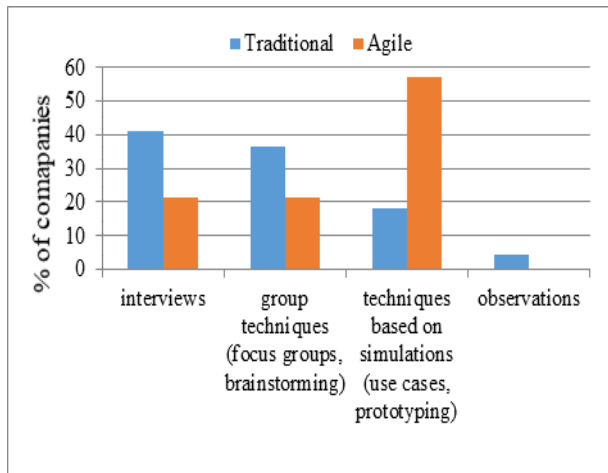


Fig 12. What techniques are used for gathering requirements?

6. Comparison of results

The questionnaire used in the research is adopted from previous research as mentioned in section 4. In this section, the results of this research are compared with the previous research.

Table 1. Comparison of results with previous research

| | Previous research | | Current research | |
|--|-------------------|-------------|------------------|-------------|
| | Agile | Traditional | Agile | Traditional |
| Problems faced in software development | | | | |
| Difficulty to deliver the software with all required functions on time | 70% | 80% | 50% | 41% |
| Relationship with customers | 10% | 50% | 20% | 32% |
| Excessive documentation of code | 0 | 0 | 14% | 27% |
| Difficulty in managing relationships within the development group | 10% | 20% | 14% | 27% |
| Software development problems mitigated by adopting agile | | | | |
| Difficulty to deliver the software with all required functions on time | 40% | | 36% | |
| Relationship with customers | 60% | | 32% | |
| Excessive documentation of code | 20% | | 20% | |
| Relationship with customer | | | | |
| Collaborative relationship with the client | 50% | 60% | 21% | 40% |
| Client on site | 60% | 40% | 53% | 18% |
| Fixed contracts | 20% | 40% | 18% | 13% |
| Contracts with variable prices | 30% | 20% | 7% | 27% |
| Customer satisfaction | | | | |
| Not satisfied | 10% | 30% | 10% | 31% |
| Satisfied | 60% | 50% | 14% | 36% |
| Very satisfied | 30% | 20% | 75% | 31% |
| Frequency of variability in requirements | | | | |
| Always | 61% | - | 28.5% | 9% |
| Often | 11% | 61% | 46% | 54% |
| When gather requirements | | | | |
| Initially | 12% | 88% | 27% | 73% |
| Continuously | 88% | 12% | 68% | 32% |
| Techniques for gathering requirements | | | | |
| interviews | 78% | 100% | 22% | 41% |
| group techniques (focus groups, brainstorming) | 78% | 22% | 21% | 36% |
| techniques based on simulations (use cases, prototyping) | 50% | 50% | 57% | 18% |
| observations | 22% | 20% | 0 | 5% |

Table 1 depicts the results of both researches according to the survey questions. Majority of the individuals face problem to deliver software on time in previous and current research. Previous research results indicate majority of individuals are satisfied by customers, whereas, current research indicates a very satisfied relationship. The techniques used for requirement gathering includes interviews and group techniques (focus groups, brainstorming) according to the previous research. In contrast results of current research show techniques based on simulations (use cases and prototyping) are used in agile.

7. Conclusion and future work

This research discusses the process of requirement engineering in agile. It strengthens the relationship of the customer with the team by continuous interaction and feedback. Requirements are gathered initially in companies following traditional practices, whereas, a continuously in agile. A search survey is conducted to compare the requirement engineering in traditional and agile companies. Results of the responses have been analyzed and discussed above. Overall, organizations face problems in delivering complete features of the

project on time. Moreover, according to this research, this problem has reduced in agile practices as shown in figure 5. In addition, maintaining good relationships with client is a key to attain project success. 52% of agile companies prefers client on site and continuous interaction resulting in 90% customer satisfaction. Lastly, techniques used for gathering requirements are prototyping and use cases. In a research [5], prototyping have proved to increase the efficiency of requirements engineering among different approaches. However, considering different problems faced in agile and traditional practices, maintaining good customer relationships and prototyping has proved effective.

For future a comparison on specific agile methodologies like Scrum, XP, feature driven development and etc. can be done. The comparison would identify which technique works best in a specific scenario like large and small scale teams or projects. Moreover, this research was based on 50 responses, in future research could be conducted on much larger scale.

References

- [1] "Requirements Engineering - Wikipedia." 2018 Accessed October 18, 2018. https://en.wikipedia.org/wiki/Requirements_engineering#Activities.
- [2] Lucia, Andrea De, and Abdallah Qusef. 2010. "Requirements Engineering in Agile Software Development." *Joursinal of Emerging Technologies in Web Intelligence* 2 (3). <https://doi.org/10.4304/jetwi.2.3.212-220>.
- [3] HYPERLINK "<https://www.zotero.org/google-docs/?xlhVzH>" Paetsch, F., A. Eberlein, and F. Maurer. 2003. "Requirements Engineering and Agile Software Development." In *WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003*, 308–13. Linz, Austria: IEEE Comput. Soc. <https://doi.org/10.1109/ENABL.2003.1231428>.
- [4] R, Kavitha C., N. Parur, and Sunitha Mary Thomas. 2011 Requirement Gathering for Small Projects Using Agile Methods. 122-128, NCCSE. IJCA.
- [5] Rehman Tousif ur, Muhammad Naeem Ahmed Khan, and Naveed Riaz. 2013. "Analysis of Requirement Engineering Processes, Tools/Techniques and Methodologies." *International Journal of Information Technology and Computer Science* 5 (3): 40–48. HYPERLINK "<https://doi.org/10.5815/ijites.2013.03.05>" <https://doi.org/10.5815/ijites.2013.03.05>.
- [6] Ghanyani Umesauda, Maham Murad, and Waqas Mahmood. 2018. "Crowd-Based Requirement Engineering", In *International Journal of Education and Management Engineering (IJEME)*, 8. : 10. HYPERLINK "<https://doi.org/10.5815/ijeme.2018.03.05>" <https://doi.org/10.5815/ijeme.2018.03.05>.
- [7] Zia, Ahmed, Waleed Arshad, and Waqas Mahmood. 2018. "Preference in Using Agile Development with Larger Team Size." In *International Journal of Advanced Computer Science and Applications*, 9:7. HYPERLINK "<https://doi.org/10.14569/IJACSA.2018.090716>" <https://doi.org/10.14569/IJACSA.2018.090716>.
- [8] Batool, A, Yasir Motla, Bushra Hamid, Sohail Asghar, Muneeb Riaz, Mehwish Mukhtar, and Mehmood Ahmed. 2013. "Comparative Study of Traditional Requirement Engineering and Agile Requirement Engineering." In *Advanced Communication Technology (ICACT), 2013 15th International Conference*, 9.
- [9] Bjarnason, Elizabeth, Krzysztof Wnuk, and Björn Regnell. 2011. "A Case Study on Benefits and Side-Effects of Agile Practices in Large-Scale Requirements Engineering." In *Proceedings of the 1st Workshop on Agile Requirements Engineering - AREW '11*, 1–5. Lancaster, United Kingdom: ACM Press. HYPERLINK "<https://doi.org/10.1145/2068783.2068786>" <https://doi.org/10.1145/2068783.2068786>.
- [10] Kumar, Manoj, Manish Shukla, and Sonali Agarwal. 2013. "A Hybrid Approach of Requirement Engineering in Agile Software Development." In *2013 International Conference on Machine Intelligence*

- and Research Advancement, 515–19. Katra, India: IEEE. HYPERLINK "<https://doi.org/10.1109/ICMIRA.2013.108>" <https://doi.org/10.1109/ICMIRA.2013.108>.
- [11] Mahmud, Imran, and Vito Veneziano. 2011. "Mind-Mapping: An Effective Technique to Facilitate Requirements Engineering in Agile Software Development." In 14th International Conference on Computer and Information Technology (ICCIT 2011), 157–62. Dhaka, Bangladesh: IEEE. HYPERLINK "<https://doi.org/10.1109/ICCITTechn.2011.6164775>" <https://doi.org/10.1109/ICCITTechn.2011.6164775>.
- [12] Lin, Jun, Han Yu, Zhiqi Shen, and Chunyan Miao. 2014. "Using Goal Net to Model User Stories in Agile Software Development." In 15th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 1–6. Las Vegas, NV, USA: IEEE. HYPERLINK "<https://doi.org/10.1109/SNPD.2014.6888731>" <https://doi.org/10.1109/SNPD.2014.6888731>.
- [13] Ceschi, M., A. Sillitti, G. Succi, and S. De Panfilis. 2005. "Project Management in Plan-Based and Agile Companies." IEEE Software 22 (3): 21–27. HYPERLINK "<https://doi.org/10.1109/MS.2005.75>" <https://doi.org/10.1109/MS.2005.75>.
- [14] Sillitti, A., M. Ceschi, B. Russo, and G. Succi. 2005. "Managing Uncertainty in Requirements: A Survey in Documentation-Driven and Agile Companies." In 11th IEEE International Software Metrics Symposium (METRICS'05), 17–17. Como, Italy: IEEE. HYPERLINK "<https://doi.org/10.1109/METRICS.2005.29>" <https://doi.org/10.1109/METRICS.2005.29>.
- [15] Lin, Jun, Han Yu, Zhiqi Shen, and Chunyan Miao. 2014. "Using Goal Net to Model User Stories in Agile Software Development." In 15th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 1–6. Las Vegas, NV, USA: IEEE. <https://doi.org/10.1109/SNPD.2014.6888731>.
- [16] DÜchting, Markus, Dirk Zimmermann, and Karsten Nebe. 2007. "Incorporating User Centered Requirement Engineering into Agile Software Development." In Human-Computer Interaction. Interaction Design and Usability, edited by Julie A. Jacko, 4550:58–67. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-73105-4_7.
- [17] Jiang, Li, and Armin Eberlein. 2008. "Towards a Framework for Understanding the Relationships between Classical Software Engineering and Agile Methodologies." In Proceedings of the 2008 International Workshop on Scrutinizing Agile Practices or Shoot-out at the Agile Corral - APOS '08, 9–14. Leipzig, Germany: ACM Press. <https://doi.org/10.1145/1370143.1370146>.
- [18] Ghanam Yaser, and Frank Maurer. 2008. An Iterative Model for Agile Product Line Engineering. 12th Int. Software Product Line Conference - SPLC 2008, 9,377-384.

Authors' Profiles



Tazeen Fatima: Tazeen is currently studying MS in Computer Science from International Business Administration, IBA. She has completed her Bachelor's in Computer Science from SZABIST in 2017.



Waqas Mahmood: Waqas is a professor at IBA, he has completed MS in Economics and Finance from IoBM in 2012 an MS in Software Project Management from FAST in 2010.

How to cite this paper: Tazeen Fatima, Waqas Mahmood," Requirement Engineering in Agile", International Journal of Education and Management Engineering(IJEME), Vol.9, No.4, pp. 20-33, 2019.DOI: 10.5815/ijeme.2019.04.03