

Agile Requirement Engineer : Roles and Responsibilities

Vishvadeep Tripathi¹, Arvind Kumar Goyal²

School of computer science and IT, Devi Ahilya University
Indore, Madhya Pradesh 452009, Indore
¹vishvadeep@gmail.com, ²goyalkcg@yahoo.com

Abstract

Software development in its own is a very complex process and if requirement is not stable and keep changing from the requirement gathering to development phase then it becomes very difficult to implement. Software projects which are getting frequent requirement change and requirements are very dynamic in nature uses agile software development methodology. In this type of software development software requirement management and software configuration management happens to be the most important part. Agile software development needs considerable modification in the software requirement phase, software design phase and coding phase as agile methodology works on iterative manner in this methodology one part of software is analyzed, designed, developed and delivered during each iteration. Traditional requirement engineering and agile requirement engineering are two very different approaches and have their own methods to plan and control requirements. In this paper we will distinguish agile and traditional requirement engineering approaches and we will identify roles and responsibilities of requirement engineering.

Keywords: Agile methods, Requirement engineering, Requirement engineer, Roles and responsibilities

1. Introduction

Agile software development approaches emerged as a light-weight alternative to phase driven traditional software development and became more popular during the last few years. Agile delivery methods are mainly based on early delivery of the product, high customer involvement, self-organized and small team, just enough documentation, continuous delivery with short iteration cycles and small releases. Traditional software development approach are based on upfront planning, writing complete requirement upfront, analyzing the requirement then start design and development. Some time it takes months to produce workable software in front of customer and when customer see working software

then customer come with changes as developed software does not have the all the functionality which customer was expecting in traditional approach. Agile projects work on the principal of continuous and face to face communication with customer to get the requirement for each iteration cycle. All agile methods worked on the common principal which include high customer satisfaction, adapting to changing requirements, frequently and early delivering of working software, and close collaboration with customer and developers. In agile methodology requirements evolves throughout the development.

Most difficult part of the building software is deciding precisely what to build. It is very important to have detailed, clear, unambiguous and precise technical and functional requirement to develop successful software. Researches show that Inability to produce correct, complete and unambiguous requirements is major cause of software failure. Requirements are nothing but details of system and specify that what the system must do, how should it behave, what all properties it must have, qualities it must possess, and the constraints that the software system must satisfy.

Requirement Engineering is one of the branches of software engineering. It has different methods, process and techniques for the requirements elicitation, requirement analysis, specification and verification of the requirement. Business needs which are also called requirements are developed by the requirement engineers and developers needs to understand these requirements to develop software. Requirement engineer needs to be fully aware of different features and limitations of the system being developed.

Requirement engineering is a one of the critical phase in software development life cycle. Success or failure of a software system is dependent upon the requirement engineering. Requirement engineering is the process of requirement gathering, understanding, formulating, documenting, maintaining and managing of the requirements. Systematic and disciplined approach is needed for managing the software requirement. Requirement engineer should work with the following Objective.

- 1) Engineer needs to focus on understanding customers and all the stake holders' desire and their requirement. They should create it and manage it and it will reduce the risk of failure of the software and it should full fill customer's demand
- 2) Requirement engineer should give emphasis to know the relevant requirement, remove the conflict and create consensus among the stake holders if any for any requirement. Create unambiguous documentation with given standards and manage requirements systematically.

Quality of the any software project is decided by the requirement engineering process. For managing the requirement, team needs to follow many practices and process and these process play important role in success of the project. Now a day's many projects works in the agile environments in order provide rapid delivery of high quality software. Team which is involve in adopting agile process and practices should have open mind and flexible enough to work and adjust themselves with changing requirement and dynamic environment. In agile methods project see frequent requirement change, which is a different from the traditional requirement in order to have long detailed signed off documentation before starting the project.

Traditional Requirement Engineering follows the document driven approach and gives emphasis on software development that is developed by gathering all product requirements from the business users at very beginning of the project. In the traditional methodology requirement elicitation which is the process of discovering, reviewing, documenting, and understanding the customer's needs is completed at very early stage of the project. Developers have to deliver all the functionality of the system in one go and user see the result after long span of time at the time of acceptance testing. This process creates a very difficult situation and consists of more cost if clients want some changes in requirements which have been already frozen and development has been completed for that. Traditional Requirement Engineering process is a complicated process as in current dynamic business scenario client may not be aware of all the requirements at the beginning of the project. Now a day's Real life development in the software industry demands very flexible, adoptive and speedy delivery process. Efficient Requirement Engineering process required for delivering successful software product.

In agile methodology requirement engineering handled by understanding the requirement by continuous

communication with customer, customers do the prioritization of the requirement and customer is responsible for managing the requirement.

In agile methodology requirements are given by the customer in the form of user stories which consist of user related requirement and acceptance criteria. Product owner do the prioritization of all such requirements. Product owner explain the requirement in the beginning of iteration to the developer's and involve in the development of software and product demonstration. Product owner can have one or more business analyst who can help him to documents formal requirements. Requirements engineering suggests the use of systematic and repeatable techniques to ensure completeness, consistency, and relevance of the project requirements.

Agile Requirement Engineering is comparatively new and emerging area that makes the requirement engineering process flexible and faster. It does not give emphasis on getting at the requirement in the beginning of the project rather than it offers an iterative, incremental approach that guides the agile teams to focus on faster delivery of products in short span of time maximum of one month. It focuses on constant face to face communication between customers and agile team developers. As a result of this developers are delivering the software system within time and which has high customer satisfaction and full fill the customer's expectations which results in increase of the business value.

In recent few years Agile Requirement Engineering has proven track record and success of projects depend up on following reason. Firstly, it creates transparency between customer and developer and that is visible to customer, customers are involved in day to day activity and see the demonstration in completion of every sprint. Customer gives feedback, suggestions which are incorporated in coming sprints. Secondly, it gives emphasis on face-to-face communication rather than documentation which makes requirements more clear.

The remaining of this paper is organized as follows. Section II describes basic concept of scrum. Literature review is described in section III. Section IV presents the finding of this paper in the form of identifying issues and challenges of the requirement engineering and guideline to follow to resolve them. Finally section IV concludes the papers.

2. BASIC CONCEPT

Scrum is most commonly used frame work which is used in agile development. It is used to manage system

development by applying flexibility adoptability and productivity. Scrum gives emphasis on team collaboration for producing high quality work in changing environment. Scrum uses product backlog, sprint back log, user stories, sprint and daily scrum meeting.

Product Backlog: product back log play very important role in the scrum. All requirement related with the product are listed in the product back log. The product backlog contains set of functionalities which are called user stories, enhancement and bugs according to its importance and business value.

Sprint Backlog: product owner decides the priority of the requirement and high priority stories from product backlog are moved to sprint backlog for development in scrum process. Subset of product back log is called Sprint backlog.

Sprint: It is a 30 days development iteration sprint back log item is taken by the development team and start doing the development. No Changes are accepted to the sprint backlog during the sprint execution.

Scrum Master: Scrum master work like a facilitator to the team and responsible to remove the entire obstacle which team faces during the sprint. Scrum master is responsible for conducting daily scrum meeting.

Product Owner: product owner is owner of the product. Product owner is responsible for creating product backlog and assign priority of the requirement. Sprint review and product demo needs presence of the product owner to track the progress, clarify requirement if needed and for accepting of the product and providing the feedback during the demo.

Scrum Team: Scrum team consist multi skilled people like developer, tester, business analyst, and quality assurance team members are included in team for better performance of team work.

User Story: user stories are the business requirement or product which is written in the index card.

Scrum Meetings: There are three type of sprint meeting.

Sprint planning meeting: High priority user stories are moved from product back log to sprint back log and then estimation are done after that developer choose user stories for the development.

Daily-status meeting: Each day short 15 minute meeting is conducted by the scrum master and ask three questions. What did you do yesterday, what will you do today and Are there any impediments in your way?

Sprint review meeting: this meeting is conducted to demonstrate the new functionality to the customer and receive feedback of the product owner and business person.

3. Literature Review

Literature in the field of agile requirement engineering contains a large body of research work. However, we have focused this literature survey on Roles and Responsibilities of requirement engineer while moving from traditional methodology to agile methodology. We have suggested guideline need to be followed to gather the entire requirement in agile methodology.

Vlaanderen et al presented their study on agile requirements refinery and specified that complex requirement should be treated specially structured detail should be provided for them. Complex requirements need to be converted into themes, concepts and high level and low level requirements. High level and low level requirements needs to put in the product backlog and it should be handled by the appropriate person at right time [1].

Daneva et al presented their study on large scale outsourced system software and proposed requirement prioritization process. Other than the business values knowledge of the software architecture and understanding of the requirement dependencies play very important role on requirement prioritization. Vendor domain knowledge is one of the considerable factors to reprioritization the requirement. Before addition to the new requirements in the product back log priority needs to be decided to all the element of the product back log [2].

Batool et al did the comparison between traditional and agile requirement engineering. They have used a case study and suggested that agile requirement engineering perform well than the traditional requirement engineering. They found that concise documentation is present in the agile methodology and traditional RE constructs huge documentation and concentrate on it for system development. Changes which received later stage are difficult to incorporate in the traditional methodology [3].

Sillitti et al conducted survey in document driven and agile companies and provided commonalities and differences during managing uncertainty in requirement. Companies are aware of problems of inability of the customer to write down all the list of the requirement. Companies need to use automated tools to support requirement management. The high interaction between customer and team creates satisfactory relationship in agile methods comparison with traditional [4].

Damian and Zowghi presented their study and investigated requirement engineering challenges in multi-site organization introduced by stakeholders. Based on their study they have suggested model which represents how communication between team located remotely and knowledge management, cultural difference and time

differences impact on requirements eliciting, negotiation and specification. Lack of awareness of common understanding of the requirement together with cultural differences reduces the trust between the teams and impacts a lot on requirement engineering process [5].

Paetsch et al presented their study and specified possible ways from which agile software development can be benefited from requirements engineering methods. According to them elicitation, analysis, and validation all are present in the agile methodology and repeated in each iteration which make very difficult to distinguish it from the phases. Documentation is part of agile but it should be responsibility of the development team to ensure enough documentation is available for future Maintenance [6].

Lucia and Qusef done surveys on real process and activity which are used in agile requirement engineering and specified that experience project manager, customer collaboration with the team and requirement engineer and self-organized experienced team is needed for successful agile projects. They have mention that customer involvement, agile project contract, frequent release, requirement eliciting language like NLP and traceability matrix are important factors and needs to be taken care while doing the agile software development [7]

Gupta et al presented their study on requirement prioritization and suggested multilayered dynamic approach which can be used for re-prioritization of software requirement in dynamic changing environment. Rather than using static prioritization method for dynamic requirement dynamic method will help agilest on the re-prioritization [8].

Wei Ho1 et al proposed an evolutionary model for finding out performance requirements specification for the software system and corresponding validation testing. They have proposed principals which need to be used while gathering performance requirement and those principals can be integrated with agile methods and help requirement engineer to get performance requirement and write the test cases to validate the performance requirement [9]

Eberlein and Leite conducted their study and find out points which need to be taken care while doing requirement management for the agile project where quality is major concern. They have suggested that for requirement gathering customer interaction, Verification & Validation, focus on non-functional requirement are major factors and they need to consider for delivery successful and high quality agile project[10].

4. Identified Roles and Responsibilities

We did extensive literature survey and interacted with many agile practitioners to identify roles and responsibilities of requirement engineer. Based on our extensive literature survey we have identified the common roles which played by the requirement engineer. We have provided the guideline that should be followed by the requirement engineer to deliver correct, unambiguous and proper requirement to the agile developer. Following are the common challenges faced by the requirement engineers.

Communication and knowledge sharing: In the distributed agile software development lack of communication between the onshore and off-shore team can be one of the reason for misunderstanding the software requirement which can create a lot of problem while developing the software. In agile methodology team distribution plays important role. We suggest that while applying agile methodology team collaboration is very necessary. Team should physically co-locate and each and every team member should be available throughout the life cycle of the project development. While doing off-shoring of the project which is a necessarily for the software companies to control their cost of the project, communication is the main issue. In agile distributed environment teams can be geographically distributed, can have large time zone differences, cultured difference and language differences. Some time there might be no face to face communication possible. One of the common problem in communication is team of one location is waiting for the response from the team which is sitting in another location. Requirement engineer plays important role to get the proper requirement and make it available to the developers. Software development success is depending upon the knowledge sharing. If we are working on big complex and multiyear program which is going to deliver big and complex software product which involve so many vendors. Knowledge management plays a key role in the success of this type of projects. Since in agile methodology very less focus is given for the documentation and focus is given to face to face communication, knowledge is not documented and most of it is present in the mind of the developers. Requirement Engineers supposed to create knowledge management documents which will be useful during the transition of the software from development to maintenance and it can be used by the new team member.

Customer role for requirements engineering activities: One of the major challenge for requirement engineering is roles and responsibility of customer is not defined properly. Roles and responsibility must be defined before

starting the agile development. It becomes more critical if development is going to be done in the distributed development environments. Roles should have clear instruction about customer authorities and decision making body. Customer role is very important in agile requirement engineering because customer needs to present to give developers' questions answer, should be able to take project functionality and scope decision, Must be easily accessible, should provide customer acceptance criteria and should have very vast interest in the project. We suggest that in the distributed agile methodology there should be one key member of one team physically located with team can provide two way essential communications. This team member should have strong hold on the project, good communication skill, and strong grasping power and should be able to effectively communicate across the team. We recommend that at the time of team formation project managers has to create requirement engineer role within agile team whose purpose will be work with different development team and with managers on daily basis. If project is distributed in different geography and they speak different language then involved requirement engineer should have good knowledge of both the language. In distributed agile development when face to face communication is not possible team should take the help of the tools and use of an email/ messengers to increase communication and prompt response should be encouraged. Following needs to be make sure by the customers which help the requirement engineer's.

Availability: Customer should make sure his availability to the team members and all the questions needs to be answered by him or her.

Complete knowledge: Customer should be domain expert and should be able to give input/output data and all the information regarding domain as well.

Decision power: Customer should have power to take decision and priority of the requirement.

Domain and product knowledge: In requirement engineering domain and product knowledge is biggest challenge. If product is going to be developed using onshore and offshore model then mostly business person presents in the onshore. Agile principles believes that business person should be available with the agile team so that any time if team can walks out and clear their question doubt with them but in distributed environment it is not possible and agile teams has to find out new ways to handle these type of situation. Requirement engineer plays very important role in this and they work like a bridge between the onshore and offshore and clear all the doubts of the developers.

In agile development every team member has to play much greater role than the traditional methodology. In

agile methodology developers needs to have good understanding of the domain and it is expected from them to provide intelligent solution of the problem. In agile experienced, self-organized, self-motivated team members are required who should have good business knowledge as well. Mainly In the countries to which the projects are offshore have a very high volatile job and software professionals switch jobs between companies regularly because of the primary reason is to get high salaries. It is very difficult to find developers who have good knowledge of the domain and product. Requirement engineer plays very important role and always present with team to solve team's domain related problem. They create documentation which has knowledge about domain and requirement and helps developer when developer moves out from the company. Creation of precise and clear requirement is needed which help developer's and role of requirement engineer become more important.

Non-functional requirements (NFR): One of the biggest challenges of agile methodologies requirement engineering is non-functional requirement. Agile methods do not have explicit practice for eliciting nonfunctional requirements. In agile methodology requirements are directly given by the end users and it comes in the form of user stories which has requirement in the form of product feature and does not contain information about NFR. For delivering successful project nonfunctional requirement has very important role. It is common observation that so many projects have failed dues to the lack of consideration of non-functional requirements. NFR needs to be taken care from the beginning of the project development, If NFRs are considered during software design level then it can cause several problems. In agile methods non-functional aspects are considered normally at the implementation level and it can create bigger problem than considering it design level. Clients and sometime end users of the product who are giving requirement in agile methods do not give information about transaction time, security, ease of use, levels of confidentiality, availability, reliability etc. Ideally these NFR's should have been taken care at the same time when User Stories are scoped and given by the customers. It contains product performance requirement, space requirement, and reliability requirement. We suggest that Techno functional requirement engineer should be involved from the beginning of the project and nonfunctional requirement should consider from the beginning. Performance testing should be taken in consideration and automation performance testing should be executed before delivering the project for demo to the client.

Requirement priorities: Software which used agile methods is dynamic in nature and due to this requirement

reprioritization is an important activity. Reprioritization gives emphasis on re-allotment of the priority for both implemented as well as non-implemented requirement. Implemented requirement needs prioritization because it needs to be consider for regression and stress testing and non-implemented requirement needs to be reprioritization to consider the delayed because of priority change and added new requirements relationship between client and developer is one of the important aspects of agile methods. Mostly product owner who is responsible for managing the product back log is responsible for doing this activity. Requirement Reprioritization is the decision making process. Product owner needs to consider Complexity of the requirement, Business value of the requirement, dependencies of the requirement on the different modules, delivery dependency and overall budget and cost implication of the new added requirement before giving new priority. Requirement engineer needs to provide sufficient information to the product owner regarding new added requirement and provide suggestion and make product owner's job easy.

Test case development and test support /verification and validation: In large scale agile development project s, requirement engineer of agile methodology has to provide extra support to develop integration and acceptance test cases for incremented agile development. Concept of inception can be used and well established check list can be created and requirement should be always verified by using this check list. Including check list and making sure that it is followed for verification of the requirement can improve the quality of the project. We can use tools which can generate the test cases using user stories and can use those test cases to test the application it can help the validating the software.

Challenges in conducting reviews and accepting the product: at the time or generating the user stories, user has to write the acceptance criteria and these acceptance criteria needs to be converted into the unit test cases by the developers and must be reviewed by the peers. System has to pass all the acceptance criteria unit test cases before the pressing the demo of the working software. It has been found very difficult for customer to provide clear requirement and specify what they need in one go. So many times business environment is very dynamic and they are not sure about the future requirement and completeness of the requirement as well. After seeing and feeling the running system in each and every iteration customer provides requirement and feedback of the system. Main challenge has been faced by the organizations to get the client stake holders for review and take their feedback in periodic manner and if team does not get client review in early build then it is very for the team to deliver

product as per the client need. Client has to make sure their availability for the every feedback meeting, participate well and give their productive feedback without making delay. Requirement engineer plays key role to clarify questions which comes from client and work collaboratively with team to provide successful demo to the customer.

Managing Change /Requirement traceability: Agile methodology works on the principal of respond to change over following a plan. Managing change is a fundamental practice of software engineering. Requirement management is very important aspect of the agile software development and center of requirement management is traceability matrix. For developing large and complex software system requirement eliciting and elaboration process play important role. While doing deployment using agile methodology sometime this gets less important. During requirement elicitation process collaboration and communication with business owner lead to increase understanding of the problem domain which is very critical part of successful product development.

Requirements have to be as independent as possible in order to clearly identify what to implement and make the order of their implementation irrelevant. Requirement traceability becomes important while doing maintenance of the project. Traditional way of requirement traceability will need manual tracking and maintenance from artifact to artifact, across different stages of the project. While doing development with agile methodology normally we use user stories. User stories are short description of the feature of the system. Stories need to be discussed in detail and then task is created with has more detail information. Then after that it should have high level design document, low level design document and then code needs to be written .It is very important to have the paths that link these user stories to design document and then their source as well as to code. For understanding the relationship between the problem domain and solution domain traceability matrix need to be specified. Link between the changing requirement design document and code is only successful when there is clear cut understanding of how changing customer's needs are mapped with requirement. Requirement engineers needs to provide proper requirement, make them independent, create traceability and manage the requirement.

5. Conclusion

Requirement engineering is very important aspects of agile methodology we have identified roles and

responsibilities of the requirement engineer. Requirement engineer plays very important role on Communication and knowledge sharing, Customer role for requirements engineering activities, Domain and product knowledge, on-functional requirements, Requirement priorities, Test case development and test support /verification and validation, Challenges in conducting reviews and accepting the product and Managing Change /Requirement traceability. Requirement engineer's need to work on their domain knowledge, communication, and negotiation skills, as it helps them to adopt agile methodology successfully.

- [10] A. Eberlein and J. C. S. P. Leite, "Agile requirements definition: A view from requirements engineering," In Proceedings of the International Workshop on Time-Constrained Requirements Engineering (TCRE'02), 2002 pp 4-8.

References

- [1] K. Vlaanderen, S. Jansen, S.Brinkkemper, and E.Jaspers, "The agile requirements refinery: Applying SCRUM principles to software product management," *Information and software technology* 53, Vol. 1, 2011 PP. 58-70.
- [2] M.Daneva, E.V.D Veen, C. Amrit, S.Ghaisas, K.Sikkel, R. Kumar, N. Ajmeri, U. Ramteerthkar, and R. Wieringa. "Agile requirements prioritization in large-scale outsourced system projects: An empirical study," *Journal of Systems and Software* 86, Vol. 5, 2013 PP 1333-1353.
- [3] A. Batool, Y.H.Motla, B. Hamid, S.Asghar, M. Riaz, M.Mukhtar and M. Ahmed, "Comparative study of traditional requirement engineering and agile requirement engineering," In *Advanced Communication Technology (ICACT)*, 15th International Conference on, 2013 PP 1006-1014.
- [4] A.Sillitti, M. Ceschi, B.Russo, and G.Succi, "Managing uncertainty in requirements: a survey in documentation-driven and agile companies," In *Software Metrics. 11th IEEE International Symposium*, 2005 pp 10-20.
- [5] D.E.Damian and D. Zowghi, "RE challenges in multi-site software development organizations," *Requirements engineering* 8, Vol. 3, 2003 PP. 149-160.
- [6] F. Paetsch, A.Eberlein, and F. Maurer, "Requirements engineering and agile software development," In *2012 IEEE 21st International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, IEEE 2003 PP. 308-308.
- [7] A. D. Lucia and A. Qusef. "Requirements Engineering in Agile Software Development." *Journal of Emerging Technologies in Web Intelligence* 2, 2010 Vol 3 .
- [8] V.Gupta, D. S. Chauhan, K. Dutta, and C. Gupta, "Requirement Reprioritization: A Multilayered Dynamic Approach," *International Journal of Software Engineering & Its Applications* 7, 2013 Vol. 5.
- [9] Ho, Chih-Wei, Michael J. Johnson, Laurie Williams, and E. Michael Maximilien. "On agile performance requirements specification and testing." In *Agile Conference*, 2006, pp. 6-pp. IEEE, 2006.