



Implementation of Requirement Engineering in Extreme Programming and SCRUM

Apoorva Singh

Department of Information Technology
Babasaheb Bhimrao Ambedkar University
Lucknow-226025, India

Dhirendra Pandey

Department of Information Technology
Babasaheb Bhimrao Ambedkar University
Lucknow-226025, India

Abstract: Requirement Engineering is an essential process during software product development, it become more important when requirement is changing rapidly throughout the development process. In traditional development strategies requirements are kept initially and referenced by different stakeholders in later stage of development according to their use but this job is become toughest when requirements are volatile. In this article we studied about Agile Development Methodologies and strive to seek the relation with Requirement Engineering Techniques which are strongly rely on the concept of “No Change in Requirement” after Requirement Analysis phase and try to implement RE in Agile Software development process like XP and SCRUM.

Keywords: Extreme Programming, Test Oriented Development, Pair Programming, View point, Scrum Master, Product Backlog etc.

I. INTRODUCTION

Now days in software product development process, it is not easy to capture requirements at very initial stage of development because requirement has tendency to change overtime in response to the dynamically evolving needs of customer, value of product in market, stakeholders and working environment. The changes in requirements generate instability in development process and make development over budgeted. RE deals with such kind of problems well when we are developing product with traditional approach like waterfall model but it will make big chaos when we apply Requirement Engineering in Agile Development methodology. RE is so much relying on documentation for information sharing while Agile Methodology relying on customer and developer face-to-face collaboration. The purpose of this paper is to find out the some commonalities between RE and Agile approach and find out the way to embedded RE in Agile Development effectively.[1,3]

In following section we studied about some RE approaches which are useful in Agile Development. Section 2 discusses about the Implementation of RE in Agile methodology. Section 3 defines RE in Agile software development and section 4 is about general practices of RE having Agility.

II. IMPLEMENTATION OF REQUIREMENT ENGINEERING IN AGILE METHODOLOGY

- When we compare the tradition Methodologies (Iterative, Spiral, Waterfall, evolutionary etc.) with Agile Methodologies we will find following asymmetry in the process of accumulation of requirements.
- In traditional Methodologies requirements are collected, analyzed and refined at front end of process while in Agile Methods, Requirement Engineering is done simultaneously with other Software Engineering processes, requirements are refined and discovered after each system build.[10]
- Agile Manifesto focused on “Customer collaboration over contract negotiation” means it highly emphasize

on joint effort of customer and developers, all system developers are involved in RE activity each can and should have regular communication with the customer. [10]

- In traditional methodologies involvement of customer ends when Software Requirement Specification has been generated and approved, but in Agile Development changes are invited throughout the process.[10,11]

III. REQUIREMENT ENGINEERING IN AGILE SOFTWARE DEVELOPMENT

When we distinguishing traditional software processes, agile methodology has more focus on outcome rather than documentation. Agile methods are adaptive in nature and based on close collaboration between stakeholders i.e. business peoples and developers. In Agile software development, all planning are short termed, progress of development comes in small incremental steps.

Agile methods are basically based on proficiency of involved stakeholders and strong communication with customers, bonding of customer with development teams leads development process to produce high quality software.[2,5]

A. XP Extreme Programming

XP is most opted and widely used Agile Methodology, it focused on values of simplicity, inter-communication, effective feedbacks and courage to handle hurdles. The intent of XP is to start development with smaller development teams and relatively few detailed artifacts. XP implements iterative and vertical approach to its development cycles, when we compare XP with other traditional models like waterfall model, iterative model, Evolutionary model etc. have distinct requirement analysis, design, implementation and testing practices exactly same like waterfall model, XP treats these activities as being interrelated and more and less continuous in vertical manner. In XP, the elementary core four practices of software development “Planning”, “Coding”, “Designing”, “Testing” are break down into the

following 12 core practices to manage the changes and maintain the development process stream line.

1. Daily Standup Meeting

It is a formal communication between manager and developer team to know the status of current developing product, meeting summed up only in 15 minutes with 3 formal questions ‘what you did yesterday?’, ‘what are you going to do today?’ and ‘Is there any hurdles?’ After the meeting, manager create new plan to speed up the development process.



2. Incessant Small Release

Agile software development relies on quick and frequent release of working product, the XP team works in a strict priority orders, features of product is to be developed according to customer’s priority. XP teams work on small sprints maximum 2 weeks, after each sprint XP team releases working deliverable. [4,6]

3. Designate New Task

After compilation of each sprint, new task from the product backlog is assign to XP teams, XP team has ability to swap with other team with in the iteration to relive other team, facing issue on particular sprint.

4. Customer’s Availability

Involvement of customer or customer’s representative at development site creates healthy environment for understanding the actual need of customer and his imagination about future product. XP provides platform to their customers and development team to work under same roof to refine the requirements during development process.

5. Pair Programming

Pair programming is a unique technique of coding where two developers work on same code together to achieve same goal, this technique increases understanding between coders. In Pair Programing one coder is writing the code, at same time other one, sitting next to him, observe the standards of coding, spotting errors, finds way to simplify codes and help him to write program.

6. Coding Ownership

XP strictly follows the coding rules and standards to decrease the dependency on one coder, In XP any pair of programmers can improve any code at any time.

7. Discovering Easiest Solution

XP accentuate to discover simplest solution for given problem, XP team design capable and simplest solution according for requirement produced by the owner. Design proposed by XP team always open to implement change on existing design.

8. Plan for Future Change

Market research is an essential task before development of any product; consumer will not accept the product which is outdated and having high price as compared to similar product in market. XP team design product which can survive in future as well as present.

9. Continuous Refactoring

XP always focuses on business values, refactoring is the best way to improve the designing of the system after each release. Refactoring of code improves the lousy design and increases the cohesion and minimizes the coupling to increase the effectiveness of the product. Refactoring process also ensures that nothing is broken or damaged during enabling changes in design.[7]

10. New Test Case Generation

In XP, feedback after each release is provided by the customer to improve the software development, these feedbacks are necessary to write new test case. In XP new test cases are also written after each sprint or whenever bug is found.

11. Unit Test For All Code

Everyday each code is tested before storing into the repository, and releasing from repository to maintain the credibility of written code.

12. Design Test Framework

In XP, automated test frameworks are used to provide bug free and quality code for shorten development. It saves manual testing time and speed up the development time.

B. SCRUM

Scrum is a part of Agile Movement which deals with big projects. Scrum provides a framework for software development process which includes particular set of practices which are performed under the guidance of ‘Scrum Master’ (Facilitator) and Team (Development Team) in two week time box called Sprint.

In Scrum, Product Owner provides his wish list called Product Backlog. Product Backlog is a prioritized list of all features, functions, enhancements, and bugs which is created by Product Owner in form of user stories. When we consider about RE this product backlog is compared with an incomplete and changing requirements document containing information needed for development. According to priorities of requirement, Scrum Master moves highest priority requirement to Sprint Backlog from Product Backlog. One’s Sprint backlog is created, development process starts. The planning of sprint is depending upon the requirement of Product Owner; the Sprint Backlog will not change during the Sprint. [2,4, 7,9]

After compilation of each sprint potentially shippable product will delivered to the product owner. Scrum framework has three core scrum roles-

- Product Owner
- Scrum Master
- Development Team

1. Product Owner

Product owner is a whole sole person of any Project, in general terms Customer is called Product owner. It provides a wish list of requirement in form of user stories called Product Backlog to Scrum Master.

2. Scrum Master

Scrum Master is a facilitator who facilitates Scrum. SM works as bridge between Product Owner and Development Team. It perform following task for implementing Scrum-

- a) *Creates Sprint Backlog*
- b) *Make plan for Sprint*
- c) *Remove hurdles of Development Team*
- d) *Facilitate the Daily Scrum Meeting*

3. Development Team

The scrum development team is short in size gene minimum 5-9 members, scrum development team having self-organizing and cross-functional skills. The team has any type of member like Analyst, Designer, Developer, Tester, Requirement Engineer etc.

IV. RE PRACTICES HAVING AGILITY

- Head on communication
- Test driven development
- View point
- Constant Planning
- Extreme Prioritization

A. Head on communication

Interaction with customer is major objective in agile software development, when we take concern about RE techniques Interview is a finest method to figure out the requirements and feasibility of future developed product. Interview provides direct and ‘unfiltered’ access to the needed information. It makes direct communication with customers which avoid misunderstanding and help to create pool of rich collection of knowledge. Direct interaction with customer also helps establishing trust relationship between customer and developer.

B. View Point

View point is a technique of organizing the requirement of system on the basis of end user and manager perspective. View point deals with different perspectives and provides the framework for discovering conflicts in the requirement proposed by different stakeholders. In agile development requirements are gathered in different perspective to consolidate the actual need of customer. [2]

C. Extreme Prioritization

Scheduling and reshuffling of requirements according to need is a common practice in all agile methodologies. In Agile development methodology Prioritization of requirement is done after each release, highest priority features are implemented first to deliver business values. After each release, involvement of customer increases, which helps developers to reset the priority list according to their point of view. [2, 3]

D. Constant Planning

Agile methodology is based on future forecast road map for future product is planned before development is going to start. In Agile methodologies mainly two types of plans are executed, one for future point of view and one for current situation. Agile planning starts with big picture, defines the vision and outline the strategies (resource availability, time

for development, cost for development, performance, user experience, security and many more) then after identifies hurdles which comes currently or in future. After that big picture is divided in to the small pictures and short planning is start for each release (panning according to priority of product backlog).[4]

After smart release, planning for make change in current developed product started with feedbacks comes from customers and other stakeholders. The process is still continues till best result will not come.

E. Test Oriented Development

Test Oriented Development instructs developer to write acceptance test before writing functional code. In TOD developer performs three major activities- Testing, coding and Design.

In TOD developer writes test case (unit test) before coding every functionality of developing product, obviously test became fail because it was written before coding than developer refactor the design till we did not get high quality design.

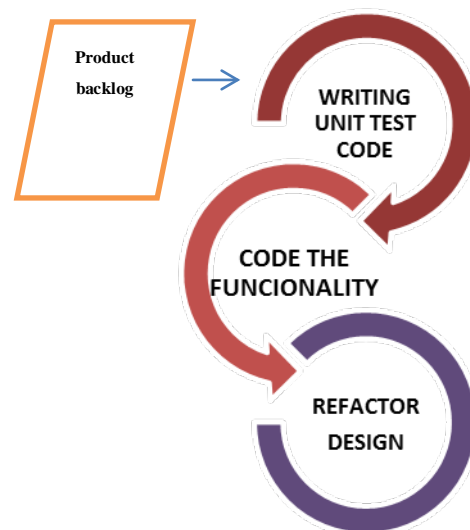


Figure 1. Test Oriented Development.

V. CONCLUSION

This paper has presented various approaches to apply Requirement Engineering in to the Extreme Programming (Agile Methodology) and different RE practices for XP. Since these methods are advantageous for XP, subject is still evolving and many techniques are under investigation. XP is a valuable approach for small projects; it can also be used for big projects but their limitations should be defined.

VI. REFERENCES

- [1] Frauke Paetsch, Dr. Armin Eberlein, and Dr. Frank Maurer, “Requirements Engineering and Agile Software Development”, IEEE (WETICE’03),2003.
- [2] Manoj Kumar, Manish Shukla, and Sonali Agrawal, “A Hybrid Approach of Requirement Engineering in Agile Software Development”, International Conference on Machine Intelligence Research and Advancement, (pp.513-519), 2013.
- [3] Georgia M. Kapitsaki and Marios Christou, “Where Is Scrum in the Current Agile World?,” IEEE (ENASE), 2014.
- [4] Phillip A. Laplante, “Requirement Engineering For Software and System,” CRC Press, (pp 139-153),2009.

- [5] Anitha P.C., Deepti Savio and V. S. Mani, "Managing Requirements Volatility while 'Scrumming' within the V-Model," IEEE EmpiRE, 2013.
- [6] Mohd Ali Babar, Emam Hossain and Hye-young Paik, "Using Scrum in Global Software Development: A Systematic Literature Review," IEEE ICGE, (pp 175-184),2009.
- [7] Martin Flower, "Refactoring Improving the Design of Existing Code",Adisson Wesley, 2012.
- [8] Linda A. Macaulay," *Requirements Engineering*", Springer,Verlag, 1996.
- [9] Ken Schwaber and Mike Beedle, "*Agile Software Development with Scrum*", Prentice Hall, 2001.
- [10] G. Kotonya, I. Sommerville, *Requirements Engineering*, John Wiley & Sons, New York, 1998.
- [11] D Pandey, U Suman, AK Ramani, "Impact of Requirement Engineering Practices in Software Development Processes for Designing Quality Software Products", National Conference on NCAFIS, DAVV, Indore, 2008.