A Literature Review on Agile Model Methodology in software Development

Riya Shah*

[*]*Lecturer in Computer Engineering Department, S.B. Polytechnic, Savli, Gujarat, India.*

ABSTRACT

"Change is necessary, growth is optional" said by a John C. Maxwell. And this Sentence exists very truly in software engineering. Agile process is an iterative approach in which customer satisfaction is at highest priority as the customer has direct involvement in evaluating the Agile Software Development is the result of that changing environmental demand and struggle of researchers to overcome traditional model of software development. Agile development is modern approach which deals with fast delivery of quality software and full association customers, so the requirement of customer can be satisfied and achieve the goals. This review paper includes different approaches of agile.

Keywords: *Agile software development; Extreme Programming; Scrum; Feature Driven Development; limitation; agile manifesto*

I.INTRODUCTION

Agile is greater extent becoming the dictate developing method in the software industry. A lot of companies are opportunity close to agility in one way or another because of the need for fast delivery while at the same time dealing with fast changing requirements. Agile methodologies try to find an equilibrium point between no process and too much process, allowing it to survive in dynamic environments where requirements frequently change while Striving high quality software product [1] Agile encompasses different methodologies, including: Agile Software Development (ASD) [2], extreme Programming (XP)[3], Scrum, Feature Driven Development (FDD).

II.AGILE METHODS

The agile methods claim to place more emphasis on people, interaction, working software, customer collaboration, and change, rather than on processes, tools, contracts and plans [8].Many studies have been conducted on agile methods. In this review paper four agile methods are discussed as below:

- **Agile Software Development (ASD)**

Agile means "moving quickly "or "flexible", as the meaning it also work like that. In other words, agile development is a modern approach which deals with inspection, monitoring, and self-Organization, rapid delivery of quality software and approaches to customer needs with company goals. Agile software development is described as iterative and Incremental because all phases are revisited throughout the life cycle.

- **Agile Manifesto**

The manifesto declares: "We are uncovering better ways of developing software by doing it and helping others to do it [4], [8]." The Manifesto details four core values for high performance are as follow:
I. *Individuals and Interactions over Processes and Tools:* To achieve high performance no communication gap should exist which helps to team to perform better than industry average. So, agile methodology seeks to increase communication and collaboration through inspect-and-adapt cycle.
II *Working Software over Comprehensive Documentation:* The agile manifesto stress delivering small pieces of working software to the customer at set of intervals.

III. *Customer Collaboration over Contract Negotiation:* It is done by direct involvement of customer in the software development process which leads to success.
IV. *Responding to change over following a plan:* Agile methodology has criteria of built-in process to change their plans based on feedback from customer at regular intervals.

- **Principles of agile manifesto[5],[6]**

1. Our highest priority is to satisfy the customer through early and continuous Delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes tackle change for the Customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a reference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team isface-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity - the art of maximizing the amount of work not done is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

- **Extreme Programming (XP)**

Extreme programming is a lightweight development methodology. It is successful because it stressed on customer satisfaction rather than delivery time of product. Extreme Programming emphasizes team work and implements a simple, but adequate way to enable group work style development. The XP (extreme programming) team not only included developers, also managers and customers as equal part of team which work together and deliver high quality product. Five principles of XP are: Simplicity, communication, feedback, courage and quality work of team. But, Extreme Programming (XP) is not suitable for distributed teams. XP support collaborative code ownership i.e. no module is owned by a single person and that is the advantage of XP because it speed up the development process and also helps in detecting errors and faults at coding phase which improve the effectiveness of software. However, XP is not best suited for any project. There are some conditions that help to decide whether to apply the XP methodology for a software development project or not. Some projects have unclear or dynamic requirements, in such case XP will succeed, while other methodologies will fail. For projects with high risk that appear to be a new challenge for the developing company, XP practices can help to lower this risk, and increase the possibility of success.[9] Working of XP is shown in fig 1.

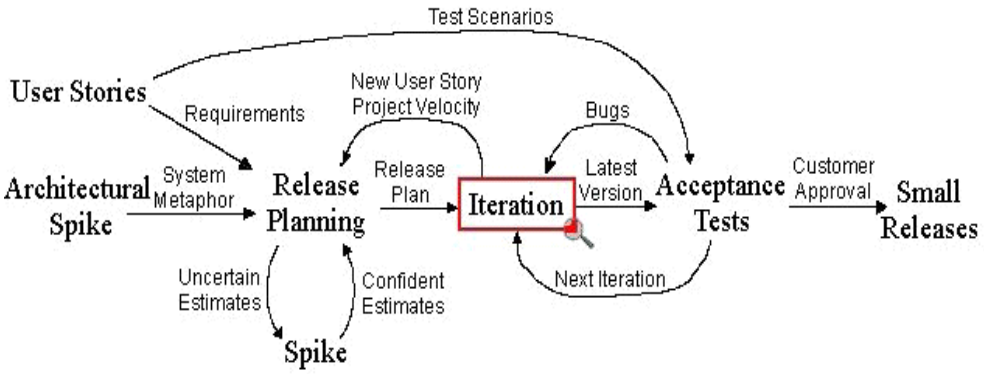

Fig 1. Working of XP

- **Scrum**

Scrum is another lightweight method for the development of software. Scrum is unique because it introduced the idea of practical experience rather than theories that is known as “empirical process control”. In Scrum, projects are divided into compact work sections, known as sprints, which are typically one to three weeks in duration. At the end of each sprint, stakeholders and team members meet to estimate the progress and make plan for its next steps. Scrum has three roles product owner who is responsible for communicating the vision of the product to development team and also represents the customer’s interest through requirements and prioritization; scrum master who acts as a facilitator for the product owners and team; team member they are responsible for completing work and team consist of seven cross- functional members. Scrum shares the basic concepts and practices with the other agile methodologies, but it comprises project management as part of its practices. These practices guide the development team to find out the tasks at each development iteration. For the team of SCRUM, three main roles are defined. The first role is the product owner, who mainly would be the voice of business. The second role is the SCRUM team which comprises developers, testers, and other roles. This team would make initial contact with customer and identify the need for a new product. SCRUM master, the third role, is responsible for keeping the team focused on the specific goals, and help the team members to solve problems when they appear [10][11]. Phases of scrum methodology are illustrated in fig 2.

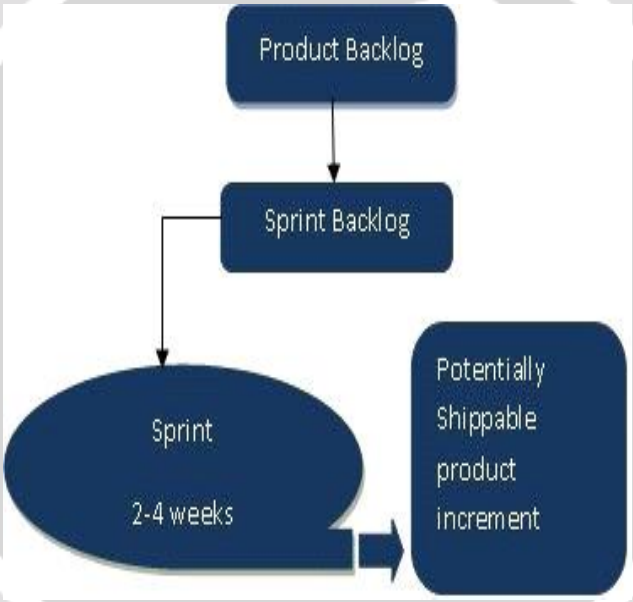

Fig 2 Scrum methodology

- **Feature Driven Development**

Feature-Driven Development (FDD) is a client-centric, architecture-centric, and pragmatic software process. The term "client" in FDD is used to represent what Agile Modeling (AM) refers to as project or eXtreme Programming (XP) calls customers. FDD was first introduced to the world in 1999 via the book Java Modeling in Color with UML, a combination of the software process followed by Jeff DeLuca's company and Peter Coad's concept of features.FDD is an iterative and incremental method based on dividing the software into many different features (models), and then builds each model separately. Feature Driven Development (FDD) is a model- driven software development process tailored to the delivery of frequent, tangible and working results. There are six primary roles on an FDD project: Project Manager, Chief Architect, Development Manager, Chief Programmer, Class Owner, and Domain Expert. It is an iterative process intended for use by large teams working on a project as illustrated in following fig3:

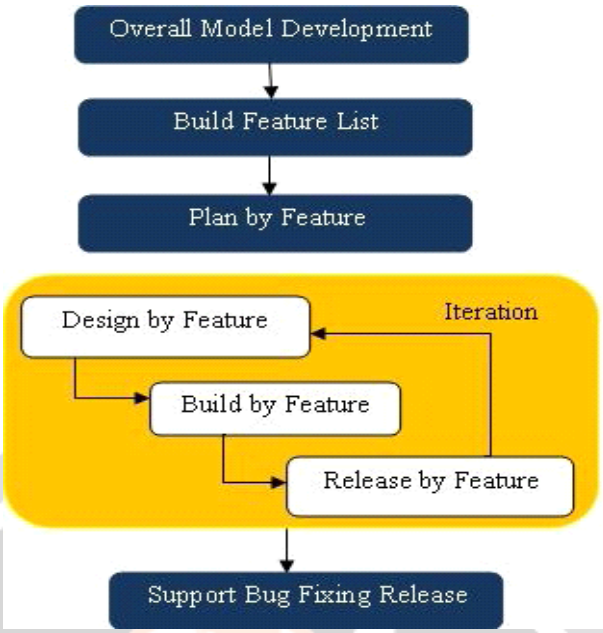

Fig. 3 FDD Methodology

III. LIMITATIONS OF AGILE METHODOLOGY

There are some limitations to implement agile methodology:

- Agile methodology is not perfect solution for green- field engineering.
- Due to less documentation agile is not suitable for maintenance purpose.
- According to agile manifesto, involvement of customer is very high, which makes success of project
- Dependent upon user cooperation and communication.
- Agile focus on custom or specific problem solution not on general solutions.
- Agile methodology is rapidly used by companies and solves specific problems after that much effort
- Software is not reusable.

IV. CONCLUSION

Agile methodologies are gaining popularity in industry. In the past few years research on agile software development suggests that agile methods are effective and suitable for many situations and environments. This paper reviews some of the latest work on agile software development in this review paper we describe our analysis of agile methodology and their different methods or techniques which implement agile.

V. REFERENCES:

[1] El-Haik, B.S. and Shaout, A. (2010) Software Design for Six Sigma: A Roadmap for Excellence. Wiley, Hoboken.

[2]High smith, J. (1997) Messy, Exciting, and Anxiety-Ridden: Adaptive Software Development. AmericanProgrammer, 10, 23-29.

[3] Anderson, A., Beattie, R. and Beck, K. (1998) Chrysler Goes to Extremes. DisrutedComputers, 24-28.

[4] Ville Ylimannela, "A Model for Risk Management in Agile Software Development", 2011

[5] Beck, Kent; et al. (2001), "Principles behind the Agile Manifesto", Agile Alliance. Archived from the Original on 14 June 2010. Retrieved 6 June 2010.

[6] Kaushal Pathak, Anju Saha, "Review of Agile software Development Methodology", Volume No. 3,2nd Feburary, 2013

[7] Tobin J. Lehman, Akhilesh Sharma, "Software Development as a Service: Agile Experiences", 2011Annual SRII Global Conference

[8] M.A. Awad, "A Comparision between Agile and Traditional Software Development Methodologies".

[9] Extreme Programming. What is Extreme Programming? [Online] Retrieved 18th March 2009.Available at: www.extremeprogramming.org

[10]M. Cristal, D. Wildt and R. Prikladnicki, Usage of SCRUM Practices within a Global Company. Global Software Engineering, 2008. ICGSE 2008.

[11]M. Singh, U-SCRUM: An Agile Methodology for Promoting Usability. In Ag. AGILE '08. Conference, Toronto, 2008, 555-560.

BIOGRAPHY

	Riya Shah working as a lecturer in S.B. Polytechnic, Savli from July 2015. Qualification:B.E(I.T)