

# Role of Agile Software Development in Today's Software Industry

Neha Agarwal<sup>1</sup>, Ashish Agarwal<sup>2</sup>, Mahnaaz Siddiqui<sup>3</sup>

<sup>1,2</sup>Astt. Professor, <sup>3</sup>M.tech Scholar, Department of CSE, RIMT Bareilly (India)

## ABSTRACT

*As the technology is growing day by day, customer's requirements also getting changed & getting bigger. If we talk about software industries, software development is now become a step by step task that requires a lot of attention and precision. So that a person or company get better returns on investment without any loss of time and cost to make software development better. In 2001 agile manifesto was published by 17 software professionals. From 2001 to 2017, Agile become the most promising software development framework. Agile is an umbrella team that consist of Scrum extreme programming, feature driven development etc. In this paper author is discussing about the current growth & usage of agile software framework.*

**Keywords:** Agile, scrum.

## INTRODUCTION

Software development process is one the high-tech development process which strongly influenced by human resources. It is mainly because of the people oriented nature of software products. Software teams most often consists of different people with different roles, background, attitude and perceptions which such a variety highly affects the development process. In fact, this is human roles which impress the flow of the software development. Agile represents a group of software engineering methodologies which promise to deliver increased productivity, quality and project success rate overall in software development projects. Such methodologies are SCRUM (Schwaber & Beedle, Agile Software Development with Scrum, 2001), XP (Beck & Andres, Extreme Programming Explained: Embrace Change, 2004), or the lesser-known Crystal (Cockburn, 2001). The outline of Agile Methodologies was laid down by the Agile Manifesto, published by a group of software practitioners (Becket. al, 2001). Agile software development has gained importance in the industry because of its approach on the issues of human agility and return on investment. Software development organizations have become more interested in agile methodologies, whose focus is client collaboration, individual value and adaptation to change. This interest has grown because these methodologies have shown productivity gains in several different software development project types. The definition of an agile methodology was created in February 2001 in a meeting of software process methodologists that resulted in what is now known as the Agile Manifesto. This manifesto is a simple and concise declaration that seeks to change the traditional lens that has-been used to see software development. Its intention is to value

- i) Of individuals and interactions over tools and processes.
- ii) Of working software over detailed documentation
- iii) Of client collaboration over contract negotiation and
- iv) Of change adaptation over plan following. The Agile manifesto is based on 12 principles.

1. Making customer satisfaction a priority though continuous and frequent deliveries.
2. Embrace requisite change, even in an advance project Phase.
3. Deliver software frequently, in the smallest possible time frame.
4. Create synergy between the business and development teams in order to allow them to work together daily.
5. Keep a motivated team providing the environment, the support and confidence needed.
6. Allow efficient information spread through face to face conversation.
7. Having a working system is the best progress Measurement.
8. Promote sustainable development through agile Processes.
9. Continuous attention to technical excellence and to a good project increase agility.
10. Be simple;
11. Allow teams to self organized using the best architectures, requisites and projects.
12. Make a reflection in regular intervals on how to become more efficient and adjust and optimize behavior.

### **Advantages of Agile:**

Agile methodology has been eventually developed from various lightweight software approaches and has also become the major reason for dislike to the classical Waterfall development process. Here are some of the Pros of Agile:

1. Change is accepted
2. End-goal can be unknown
3. Quick and high-quality delivery
4. Strong team interaction
5. Customers feedback taken
6. Continuous improvement

Cons of Agile  
The flexibility level in agile is usually very positive but still it comes with some negatives. It also becomes tough to finalize the product delivery date, maintaining the documentation and the final product delivered to the customer is somewhat deviates from the original requirements.

## **II.SCRUM**

Scrum was developed by Jeff Sutherland in 1993 and its goal is to be a development and management methodology that follows the principles of the agile methodology. The Scrum team is composed by:

- Team: it's the development project team, composed by up to ten developers in which each member has a specific skill. Nevertheless, members are not banned from performing task different from their expertise. Thus,

the team will become more integrated and team members will know better the software, minimizing the impact of another member's dismissal.

- Product owner. He is the one with the responsibility on the software functionality specification and to solve any doubts that might arise during development. He is the client's representative that must watch the project closely and help in the construction of a software that answers completely to the client's needs.
- Scrum master. He is the responsible to lead the team and to avoid any hurdles that might arise during the process. A hurdle is something that might impede a member from performing his work. For instance, requests to perform activities not related to the project, problems in the test server, difficulties with the technology and unplanned requisites might be examples of hurdles that might cause problems to the sprint.

Scrum is based in practices represented by

(i) Daily meetings, (ii) sprint planning meetings, (iii) sprint review meeting, (iv) backlog sorting and (v) release presentation.

Daily meeting are performed with the team members standing in front of the kanban, which is a set of cards (post-it) that indicate the status of a specific task, such as, To Do, Doing or Done .

Meetings last approximately 15 minutes, and in them we discuss questions from team members, what everyone intends to do and what were the hurdles found during that day so that the Scrum Master becomes aware of them and may eliminate them.

Some of the key characteristics of SCRUM include:

- Self-organized and focused team
- No huge requirement documents, rather have very precise and to the point stories.
- Cross functional team works together as a single unit.
- Close communication with the user representative to understand the features.
- Has definite time line of max 1 month.
- Instead of doing the entire "thing" at a time, Scrum does a little of everything at a given interval
- Resources capability and availability are considered before committing any thing.

### **III.IMPORTANT SCRUM TERMINOLOGIES:**

#### **3.1 Scrum Team**

Scrum team is a team comprising of 7 with + or – two members. These members are a mixture of competencies and comprise of developers, testers, data base people, support people etc. along with the product owner and a scrum master.

#### **3.2Sprint**

Sprint is a predefined interval or the time frame in which the work has to be completed and make it ready for review or ready for production deployment. This time box usually lies between 2 weeks to 1 month.

#### **3.3Product Owner**

The product owner is the key stakeholder or the lead user of the application to be developed.

#### **3.4Scrum Master**

Scrum Master is the facilitator of the scrum team. He/she make sure that the scrum team is productive and progressive.

### 3.5 Business Analyst (BA)

A BA plays a very important role in SCRUM. This person is responsible for getting the requirement finalized and drafted in the requirement docs (based on which the user stories are created).

### 3.6 User Story

User stories are nothing but the requirements or feature which has to be implemented.

## IV. ADVANTAGES AND DISADVANTAGES OF SCRUM:

Scrum is a highly injunctive framework with peculiar roles and ceremonies. While it can be a plenty to learn, these rules have a lot of favorable factors. The benefits of Scrum include:

- More crystalline transparency and more project visibility and insight
- Increased in accountability among the team members
- Flexibility to accommodate changes
- Reduction of expenses

While Scrum has some advantages it also has a few disadvantages: Here are the disadvantages of Scrum:

- Risk of scope creep
- Team requires intense experience and absolute commitment
- an ineffective Scrum Master can ruined.

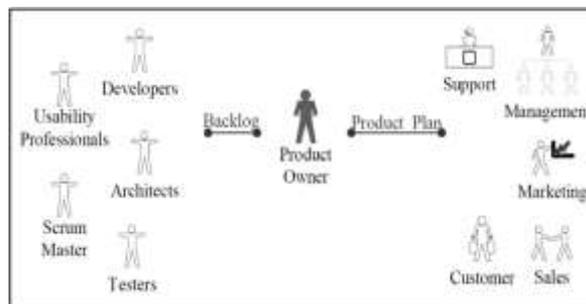


Fig.1

## V. CONCLUSIONS

The agile methodology is meant to increase the flexibility, agility in the projects dealing in software development. The basic idea of Agile approach is to break down the large tasks into smaller ones which helps them to get completed within the given time span. Agile makes use of Scrum and XP. Methodology to improvise the same. The application of these two depends on organization to organization.

**REFERENCES**

- [1]. Mona Najafi Sarpiri ,Taghi Javdani Gandomani “How Agile managers affect the process of software development? “IJCSNS International Journal of Computer Science and Network Security, VOL.17 No.5, May 2017.
- [2]. T. J. Gandomani, H. Zulzalil, A. A. Abdul Ghani, A. B. M. Sultan, and K. Y. Sharif, "How human aspects impress Agile software development transition and adoption," International Journal of Software Engineering and its Applications, vol. 8, no. 1, pp. 129-148, 2014.
- [3]. T. J. Gandomani and M. Z. Nafchi, "Agile transition and adoption human-related challenges and issues: A Grounded Theory approach," Computers in Human Behavior, vol. 62, pp. 257-266, 2016.
- [4]. <http://www.martinfowler.com/bliki/FlaccidScrum.html>
- [5]. <http://www.sei.cmu.edu/cmml/>
- [6]. <http://www.nunit.org/index.php>
- [7]. <http://junit.org/>
- [8]. Ayewah, W. Pugh, J. D. Morgenthaler, J. Penix, and Y. Zhou, "Evaluating Static Analysis Defect Warnings On Production Software," in 7th ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering, San Diego, CA, USA,2007, pp. 1-8.