iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2 **Issue:** X **Month of publication:** October 2014

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology(IJRASET)

Detailed Study on Agile Software Development Methodologies

Mayank Sharma[1], DhruvBatra[2], Abhishek Sharma[3]

Computer Science and Engineering, KIIT College of Engineering
Sohna road, Gurgaon, Haryana

Abstract - In this paper, we are going to through light on the concept of agile software development starting from the basic definition of agile and what does this type of software development offer us. This series is then followed by various methodologies involved in agile software development. There are a number of dexterous practices being adopted to develop software projects today. Methods like Feature driven Development (FDD), Adaptive software development (ASD) SCRUM, Extreme programming (XP), etcare being used at an increasing rate to develop software using a transformation approach rather than a predictive one. Thus the purpose of this paper is to provide an in-depth understanding of the major advantages of agile development approach to software development industry, as well as provide a study on threats related to this approach.

Keywords – methodologies, xp, scrum, agile, threats.

I. INTRODUCTION

Software development is an organized boom to deliver products in faster, better as well as in cheaper ways. There are a number of studies and suggestions being processed in improving the development process. Many remedies have been suggested, from the standardization and assessment of the software development process to a swarm of techniques, and practices. Lately, many of the experienced practitioners have suggested various improvements in this area and many of themhave even marked their methods as agile software development. But before talking about the methods that can be brought in use for agile software development, one needs to be thoroughly clear about the core meaning of this concept. The literal meaning of the term agile is "move quickly and easily". Agilityin terms of software development is the power of software to choose and react diligently and fittingly to various changes inits surroundings and to the demands imposed by these surroundings. Agile development methods, often abbreviated as ASD, represent an apparently new approach for planning and managing software development projects.ASD differs from traditional approaches in a sense as it puts less emphasis on up-front plans and morefocus on components for change in management during the project.

II. BACKGROUND OR RELATED WORK

Agile Software Development is currently a powerful emerging discipline in the field of Software
Engineering. It is presently advocated by a large number of software professionals.An agile process is the one that readily envelopes and supports high degree of flexibility. So, it is not just about the size of the process or the delivery speed; it is also about flexibility. This term was agreed during a big gathering when seventeen developers of the "lightweight" approaches to software development came together in a workshop in early 2001.

A. Agile Manifesto

In early 2001, seventeen of the developers gathered and discussed the lightweight approaches to software development. There then the Manifesto for Agile Software Development was published which defined the approach now known as agile software development. It is also said that the agile manifesto states the following as the main stress of the agile development:

1) Individuals and interactions over processes and tools.
2) Working software over comprehensive documentation.
3) Customer collaboration over contract negotiation.
4) Responding to change over following a plan.

The above four values are further elaborated by some more principles like:

1) Highest priority of ours is to satisfy the customer through easy and quick delivery of valuable software.

2) Will welcome changing requirements, even late in development. Agile processes tackle change for the customer's competitive advantage.

3) Deliver working software more often, from a couple of weeks to a couple of months, with a preference to the shorter time span.

The Agile Manifesto accentuates the common principles and beliefs related to these methods.
Among the first and probably the best known agile methods are Scrum and XP.Scrum method targets at providing an agile approach for managing software projects while increasing the probability of successful development of software. While on the other hand, XP emphasizes more on the project level activities of implementing a software. However, both approaches, satisfies and follows the primary principles of agile software development.

A. Agile Software Development Methods:

There are a number of methods that can be adopted for developing agile software.These agile methods are focused on a number of aspects of the software development life cycle. Some of these focus on the practices like, XP, Pragmatic Programming, Agile Modeling etc., while others focus on managing the software projects such as Scrum. Crystal methodology, Dynamic software development methodology (DSDM), Feature driven development (FDD), Extreme Programming (XP), Scrum, Lean software development (LSD) are some of the main methodologies for ASD. We are going to discuss in detail about some of the highlights of these methodologies one by one in the upcoming sections:

B. Extreme Programming (XP):

Extreme programming (XP) was one of the first agile processes that were proposed. XP consists of a set of individual practices which are when put together result in a successful software p practice. After that, the focus of XP is on the business aspect of any project which results in increasing productivity.
XP attempts to reduce the cost of changes in requirements by having more than one development cycles which are short rather than a long one. In this domain, changes are an inescapable and desirable aspect of software-development projects. These changes and should be planned for already instead of attempting to define a fixed set of requirements. Extreme programming even intends to introduce a number of basic values, principles and practices on top of the agile programming framework. One needs to consider the aspect that planning is needed for calculation of estimate of how much cost is needed to create any project. This planning trick is very efficient because it enhances product visibility all the time. There are two types of planning in XP methodology, Release planning and Iteration Planning.

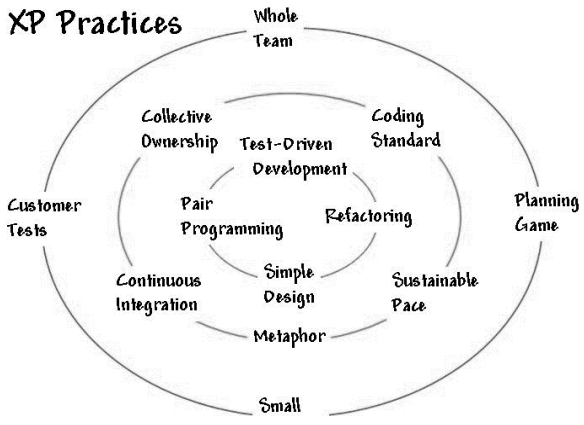


Fig. 1 A pictorial representation of practices followed by Extreme Programming (XP).

C. Scrum

Scrum is another light weight method used for the agile development of software. Its working principle lies in the fact that small teams working cross functionally produce successful results. Scrum is more revenue oriented with attention on improving revenue and quality of the software. An essence of Scrum is that during a project a customer can change his mind about what he is willing to do and that unpredicted challenge cannot be easily addressed in a planned manner.
Following is given which enumerates and **envelopes all possible types of functionalities and principles of Scrum methodology in agile software development process:**

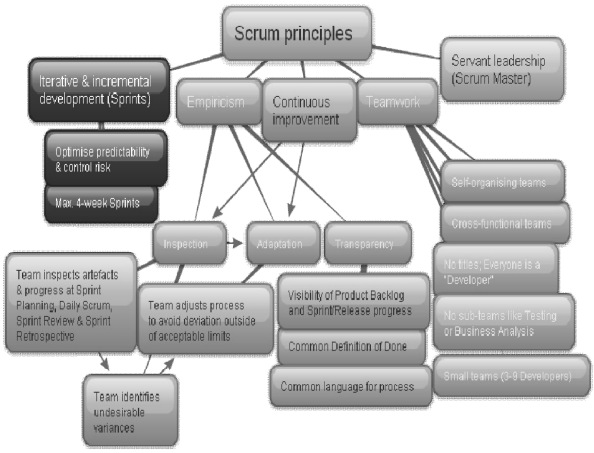


Fig. 2: Representation of functionalities and principles of Scrum methodology

The basic three things necessary in scrum are the product owner, scrum master and then these team. The product owner specifies the details like various features of software, its release date and priorities. The scrum master makes sure that the team is functioning productively and cooperating across all roles and functionality. In addition to all of this, Scrum has a set of events associated with it. They include the sprint planning meeting, Daily Scrum Meeting plus Sprint review meeting. If we talk about a sprint in particular, then it is the basic unit of development in Scrum. The sprint is a "time boxed" effort, i.e. it is restricted to a specific time limitation. The duration is fixed in advance for each sprint and is normally from one week to one month. Each sprint is started by a planning meeting where the tasks for the sprint are identified and an estimated commitment for the sprint goal is made. This is then ended by a sprint review meeting, where the progress of the completion of that goal is reviewed and lessons for the next sprint are brought into nitice Scrum emphasizes working product at the end of the Sprint that is really completed. In the case of software, this means a system that is integrated, fully tested, end-user documented, and potentially able and testified to be shipped.

D. Feature-Driven Development:

The Feature- Driven Development (FDD) method emphasizes on the features of the software of the system. These arethe main driver of the wholedevelopment process. It is significantly different from the rest of the agile processes because they put more emphasis on planning and upfront design. The first step of the FDD process is to create a detailed model of the system, which encloses all the requirements of stakeholders. Once the domain model is built, a list of the features of the system is printed by the team members.

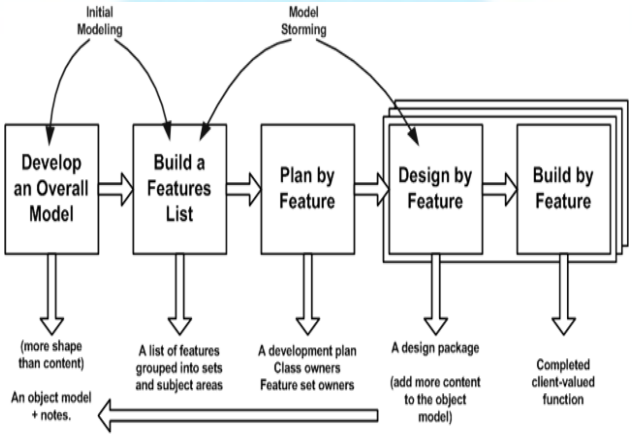


Fig 3: Showing the flow chart of principles involved in FDD Methodology.

A detailed list of features was identified using the same knowledge that was gathered during the initial modeling. This was done by functionally decomposing the domain into various subject areas. Subject areas contain things like business activities, the steps within each business activity etc.Once the feature list is completed, the next step is the production of the development plan. Class ownership has been done by ordering and assigning features as classes to chief developers.

III. THREATS OR DIFFICULTIES INTRODUCED DURING IMPLEMENTATION OF AGILE METHODS

One needs to consider various possible threats and difficulties related to agile software development. Some of these broadly categorized threats are internal validity, external validity and construct validity. Internal validity is the cause of effect relationship. There are developers who fear that the agile process could bring their deficiencies in front of others. So, there is always a pressure on that developer. To combat this challenge, developers need an environment where they feel safe to expose their weaknesses. Many of theagile practices include pair programming, meetings, and increasing social interaction, communication, and your presentation skills. Out there, there are people who are technically rich but have inherently weak communication and presentation skills, while all managers saw the benefits of constant face-to-face communication. Another threat to be taken into consideration is external threat. This arises when the researcher forms inaccurate conclusion from the data obtained and apply it to some other experiment. The study of various companies and the data obtained from there might bring valuable information. But we believe it is not sufficient to make assumptions and recommend, generally speaking, that a particular project should be done using an agile method.

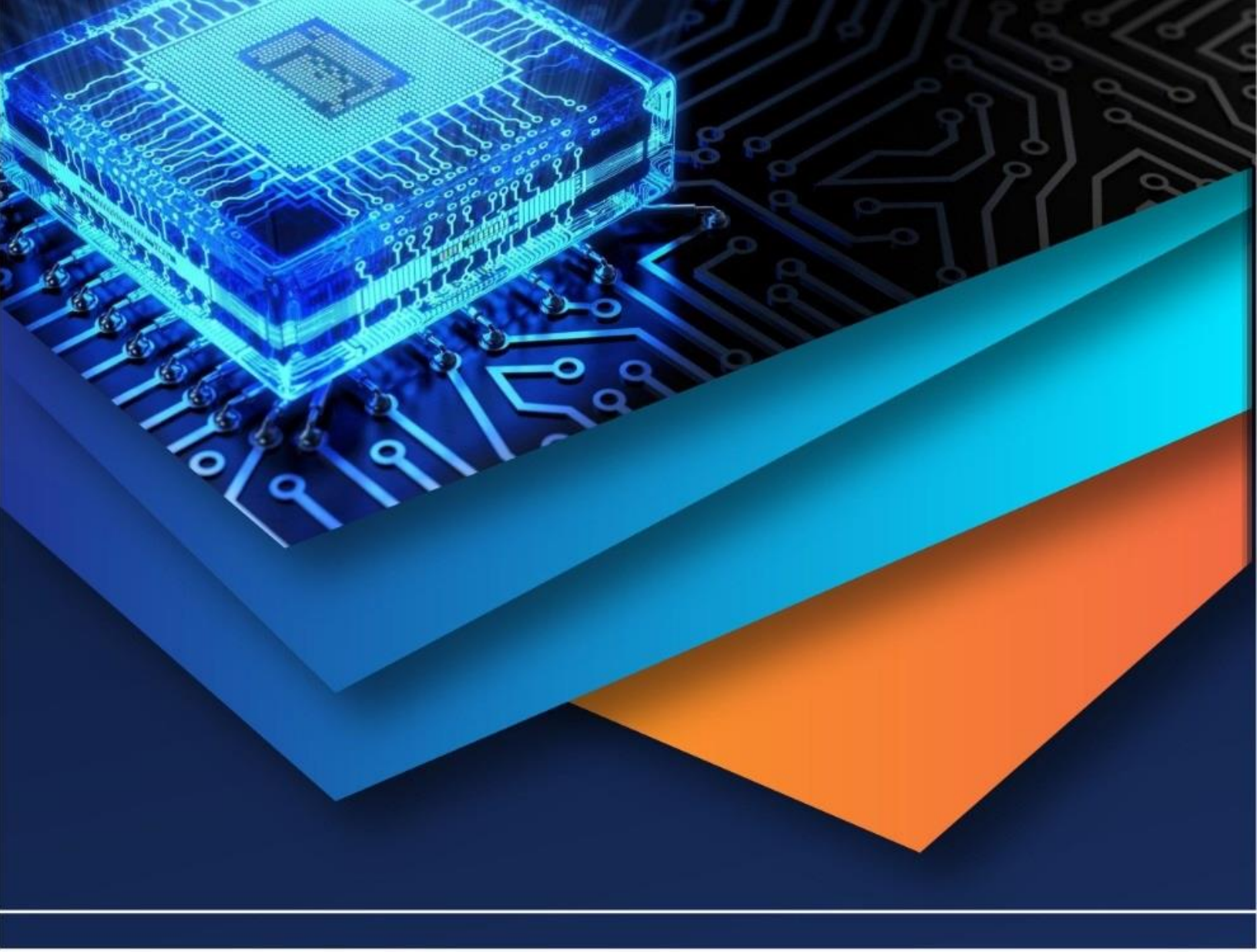
IV. CONCLUSION

This paper described a new and very convincing technique for software development, i.e. agile software development where agile approaches need to be very flexible. There is an urgent increasing need to have a better understanding of agile methods to be used in software development industry. So, in this article we discussed the background of agile software development which was the Agile Manifesto. This section was followed by various methodologies for ASD such as, XP, Scrum and FDD. Later on, light was thrown on the potential threats which endanger the agile software development methodologies. The basic purpose of this paper is to provide an in-depth understanding of the benefits of agile development approach into the software development industry.

REFERENCES

[1] http://www.ijarcsse.com/docs/papers/Volume_3/2_February2013/V3I2-0251.pdf
[2] http://www.sersc.org/journals/IJSEIA/vol5_no2_2011/4.pdf
[3] http://arxiv.org/ftp/arxiv/papers/1307/1307.3356.pdf
[4] http://en.wikipedia.org/wiki/Extreme_programming
[5] http://en.wikipedia.org/wiki/Featuredriven_development

.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY