Agile Processes and Methodologies: A Conceptual Study

Sheetal Sharma
Amity School of Engineering & Technology
Amity University Noida
Sheetal.meenu@gmail.com

Darothi Sarkar
Amity School of Engineering & Technology
Amity University Noida
darothisarkar@rediffmail.com

Divya Gupta
Amity School of Engineering & Technology
Amity University Noida
Fromdivya81@gmail.com

Abstract— This paper deals with the comparative study of agile processes. The paper will serve as guide to other software development process models. Agile processes have important applications in the areas of software project management, software schedule management, etc. In particular the aim of agile processes is to satisfy the customer, faster development times with lower defects rate. This paper compares the agile processes with other software development life cycle models. Agile processes are not always advantageous, they have some drawbacks as well; the advantages and disadvantages of agile processes are also discussed in this paper.

Keywords: Agile Development, Software Development Life Cycle (SDLC).

I. INTRODUCTION

In software development life cycle, there are two main considerations, one is to emphasize on process and the other is the quality of the software and process itself. Agile software processes is an iterative and incremental based development, where requirements are changeable according to customer needs. It helps in adaptive planning, iterative development and time boxing. It is a theoretical framework that promotes foreseen interactions throughout the development cycle. There are several SDLC models like spiral, waterfall, RAD which has their own advantages. SDLC is a framework that describes the activities performed at each stage of a software development life cycle[1]. The software development activities such as planning, analysis, design, coding, testing and maintenance which need to be performed according to the demand of the customer. It depends on the various applications to choose the specific model. In this paper, however, we will study the agile processes and its methodologies. Agile process is itself a software development process[2]. Agile process is an iterative approach in which customer satisfaction is at highest priority as the customer has direct involvement in evaluating the software[3].

The agile process follows the software development life cycle which includes requirements gathering, analysis, design , coding , testing and delivers partially implemented software and waits for the customer feedback. In the whole process , customer satisfaction is at highest priority with faster development time. The following Figure. 1 , depicts the software development life cycle of Agile Process.
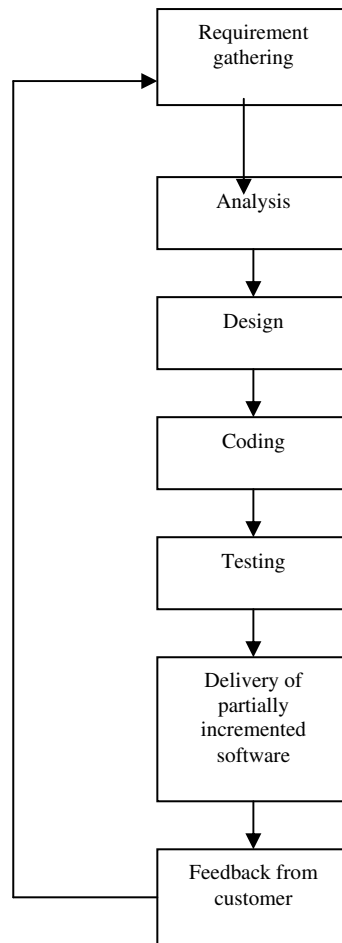
Figure 1. Phases of Agile Process.

This paper is organized as follows: Characteristics of Agile processes are presented in section II. Section III discusses the agile methodologies. Section IV discusses the advantages of Agile Process, Section V discusses the disadvantages of Agile Process, Section VI discusses the comparison of Agile with other SDLC models. Finally, conclusions are discussed in the section VII.

II. CHARACTERISTICS OF AGILE PROJECTS

Agile process requires less planning and it divides the tasks into small increments. Agile process is meant for short term projects with an effort of team work that follows the software development life cycle. Software development life cycle includes the following phases 1.Requirements gathering, 2.Analysis, 3.Design, 4.Coding , 5.Testing, 6.Maintenance. The involvement of software team management with customers reduces the risks associated with the software. This agile process is an iterative process in which changes can be made according to the customer satisfaction. In agile process new features can be added easily by using multiple iterations.

1. Iterative

The main objective of agile software processes is satisfaction of customers, so it focuses on single requirement with multiple iterations.

2. Modularity

Agile process decomposes the complete system into manageable pieces called modules. Modularity plays a major role in software development processes.

3. Time Boxing

As agile process is iterative in nature, it requires the time limits on each module with respective cycle.

4. Parsimony

In agile processes parsimony is required to mitigate risks and achieve the goals by minimal number of modules.

5. Incremental

As the agile process is iterative in nature, it requires the system to be developed in increments, each increment is independent of others, and at last all increments are integrated into complete system.

6. Adaptive

Due to the iterative nature of agile process new risks may occurs. The adaptive characteristic of agile process allows adapting the processes to attack the new risks and allows changes in the real time requirements.

7. Convergent

All the risks associated with each increment are convergent in agile process by using iterative and incremental approach.

9. Collaborative

As agile process is modular in nature, it needs a good communication among software development team. Different modules need to be integrated at the end of the software development process.

10. People Oriented

In the agile processes customer satisfaction is the first priority over the technology and process. A good software development team increases the performance and productivity of the software.

III. METHODOLOGIES

There are several methodologies through which we can implement Agile Projects. Here we have discussed three methodologies which are most widely used in Industry. The agile methods are focused on different aspects of the software development life cycle. Some focus on the practices (extreme programming, pair programming), while others focus on managing the software projects (the scrum approach).

A. Extreme Programming (XP)

XP is the most successful method of developing agile software because of its focus on customer satisfaction. XP requires maximum customer interaction to develop the software. It divides the entire software development life cycle into several number of short development cycles. It welcomes and incorporates changes or requirements from the customers at any phase of the development life cycle.
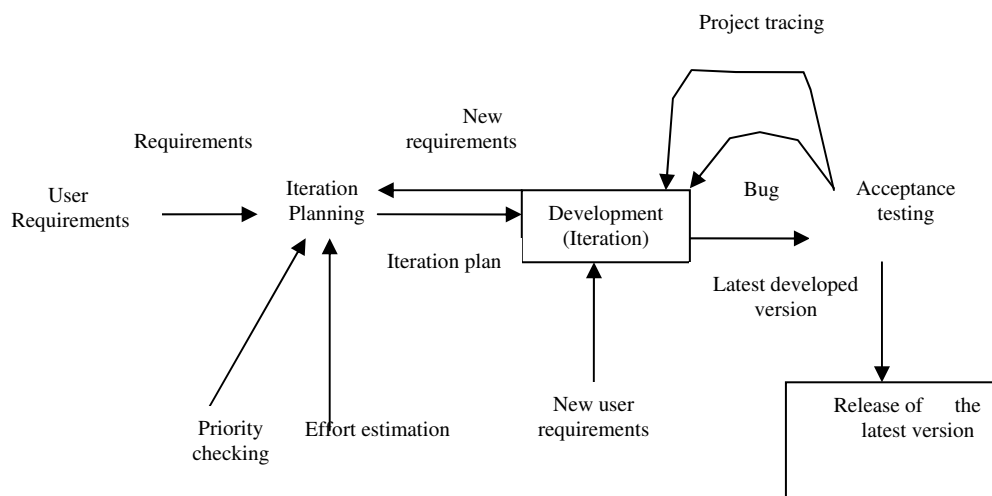

Figure 2 : Method of Developing Agile Processes using Extreme Programming

The above diagram shows the complete method of developing agile process using XP method. Extreme programming starts with collecting user requirements. Depending upon these requirements the whole development process is divided into several small no of cycles. So the next phase is iteration planning i.e. deciding the no of cycles, prioritizing the requirements and estimating the amount of effort required to implement each cycle. Now each iteration is developed using pair programming. During the development phase new user requirements may come and the iteration plan should be adjusted according to that. Next step is to test the latest developed version for bugs, if detected; the bugs will be removed in the next iteration. After every acceptance testing project tracing should be done in which feedback is taken from the project that how much job has already been done.

XP has introduced many new things for developers like pair programming, extensive code review, code refactoring and open workspace [4].

B. Scrum

Scrum is another popular method of agile development through which productivity becomes very high. It is basically based on incremental software development process. In scrum method the entire development cycle is divided into a series of iteration where each iteration is called as a sprint. Maximum duration of a sprint is 30 days.
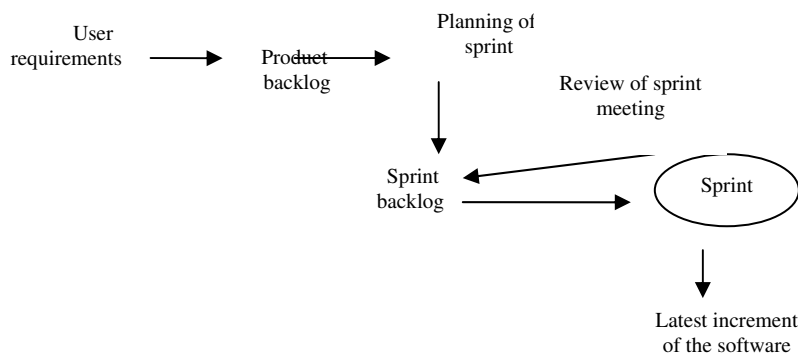

Figure 3: Method of Developing Agile Processes using Scrum

The method starts with collecting user requirements but it is not expected that all the requirements should come out from the user at the beginning. User can change their mind at any time during development; they can add new features, remove or update some existing features. Next phase is to prioritize the requirements and the list is known as product backlog [5]. A proper planning for sprint should be done i.e. how many sprints are needed to develop the software, duration of the sprint, and what are the requirements from the product backlog should be implemented in each sprint. This particular list is known as sprint backlog. During each sprint one sprint meeting is held every day to take the feedback how much work has been done. After each sprint review is taken to determine whether all the requirements for that particular sprint have already been implemented or not and to decide the requirements should be implemented at the next sprint. After each sprint we get a working increment of the software

C. Feature Driven Development (FDD)

FDD is one of the agile development methods. The key advantage of this method is to design the domain of the software to be produced before development.
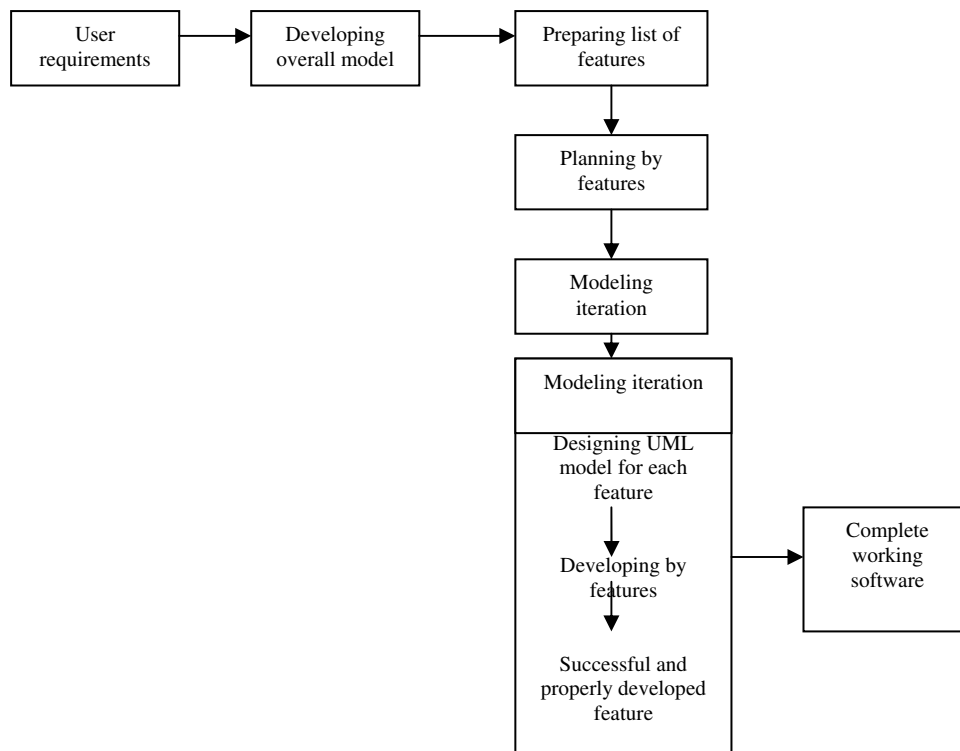
Figure 4: Method of Developing Agile Processes using FDD Graph

The method starts with collecting the requirements from the users and building up the overall model of the project. The model gives the clear idea about the scope of the software. Next step is to make a list of features which are the client-valued functions [5]. For example 'authenticate the password', 'calculate the salary for each employee', 'calculate the income tax for each employee'. Now several groups of features are made based on their domains i.e. related features are combined into a single group. Next step is to make a plan for developing the features. Each group of features is assigned to a development team which is headed by one chief programmer. Last step is modeling iteration in which first UML modeling is done for each feature and then developing that particular feature. This step continues unless all the features get implemented successfully.

IV. ADVANTAGES

1)   Adaptive to the changing environment: In agile software development method, software is developed over several iterations. Each iteration is characterized by analysis, design, implementation and testing. After each iteration the mini project is delivered to the customer for their use and feedback. Any changes that upgrade the software are welcome from the customer at any stage of development and that changes are implemented.

2)   Ensures customer satisfaction: This methodology requires active customer involvement throughout the developmen[6]t. The deliverables developed after each iteration is given to the user for use and improvement is done based on the customer feedback only. So at the end what we get as the final product is of high quality and it ensures the customer satisfaction as the entire software is developed based on the requirements taken from customer.

3)   Least documentation: The documentation in agile methodology is short and to the point though it depends on the agile team. Generally they don't make documentation on internal design of the software. The main things which should be on the documentation are product features list, duration for each iteration and date. This brief documentation saves time of development and deliver the project in least possible time.

4)   Reduces risks of development: As the incremented mini software is delivered to the customers after every short development cycle and feedbacks are taken from the customers, it warns developers about the upcoming problems which may occur at the later stages of development. It also helps to discover errors quickly and they are fixed immediately.

V. DISADVANTAGES

1)   Customer interaction is the key factor of developing successful software: Agile methodology is based on customer involvement because the entire project is developed according to the requirements given by the customers. So if the customer representative is not clear about the product features, the development process will go out of the track.

2)   Lack of documentation: Though the least documentation saves development time as an advantage of agile method, on the other hand it is a big disadvantage for developer. Here the internal design is getting changed again and again depending on user requirements after every iteration, so it is not possible to maintain the detail documentation of design and implementation because of project deadline[7]. So because of less available information, it is very difficult for the new developers who join the development team at the later stage to understand the actual method followed to develop the software.

3)   Time consuming and wastage of resources because of constant change of requirements: If the customers are not satisfied by the partial software developed by certain iteration and they change their requirements then that incremented part is of no use. So it is the total wastage of time, effort and resources required to develop that increment.

4)   More helpful for management than developer: The agile methodology helps management to take decisions about the software development, set goals for developers and fix the deadline for them. But it is very difficult for the baseline developers to cope up with the ever changing environment and every time changing the design, code based on just in time requirements.

VI. COMPARISON OF AGILE PROCESS WITH OTHER SDLC MODELS

TABLE I.   PRCOESS MODELS

Features	Different Process Models		
	Agile Process	*Spiral Model*	*RAD Model*
Definition	Agile process is the ability to both create and respond to changing requirements of software.	Spiral model is the software development model which focuses on managing risks.	RAD model is "high speed adaptation of linear sequential model, in which component based construction is used.
Adaptability	✓	✓	X
Testing Phase	Unit, Integration , System testing	Unit, Integration and System testing	Unit
Quality Factors	✓	✓	X
Risk Analysis	X	✓	X
Off-the- Tools	X	X	✓
Failure normally due to	Code	Code	Architecture and design
Knowledge Required	Product and domain	Product and domain	Domain
Entry & exit Criteria	X	X	✓
Mock up	✓	✓	X
Extendability	✓	✓	X
Project management involvement	✓	X	✓
Higher Reliability	✓	✓	X
Time Boxing	✓	X	✓

Features	Different Process Models		
	Agile Process	*Spiral Model*	*RAD Model*
Status of Development Team	Less experience required	Less experience required	More experience required
Use of reusable components	X	X	✓
Flexibility	✓	✓	X
Customer Involvement	✓	✓	X

VII. CONCLUSION

In this paper we have discussed the software development life cycle models, the characteristics of agile process, and spiral model, methodologies of agile process, advantages and disadvantages. In the comparative study of agile software development with other software development models we conclude that agile project is much better than other software development process in terms of productivity, performance, faster time cycles, risk analysis. Agile processes are implemented in important applications such as web based, testing tools, etc.

REFERENCES

[1] Tobin J Lehman, Akhilesh Sharma , "Software Development as a service: Agile Experiences", in annual SRII Global Conference (2011).

[2] A. Ahmed, S. Ahmad, Dr. N Ehsan, E. Mirza, S.Z. Sarwar,"Agile Software Development: Impact on Productivity and Qulaity", in the Proceedings of IEEE ICMIT.(2010).

[3] B.Boehm and R.Turner, "Balancing Agility and Discipline: A Guide for the Perplexed, Addison, Wesley, 2003.

[4] Jeffery A. Livermore, "Factors that Impact Implementing an Agile Software Development Methodology" in the Proceedings of IEEE (2007)

[5] http://en.wikipedia.org/wiki/Scrum_(development)

[6] B.Boehm, " Anchoring the Software Process," IEEE Software, July 1996.

[7] B. Boehm and D.Port,"Balancing Discipline and Flexibility with the Spiral Model and MBASE". Crosstalk, Dec. 2001.