



Approaches for development of Software Projects: Agile methodology

Bonthu Kotaiah

Assistant Professor, Dept. of CS & IT,
Maulana Azad National Urdu University, Hyderabad, India

Md Asif Khalil

Student (M.Tech.), Dept. of CS & IT,
Maulana Azad National Urdu University, Hyderabad, India

Abstract: In the new era of software development, project developers and teams are facing many problems while adopting traditional software development methodologies. In traditional methodologies there is a scope for the wastage of different resources like people, cost and others. Hence there is a need to develop a methodology to minimize the usage of minimum resources for the maximization of chances of developing reliable software. Hence, to overcome this problem software developers adapt a new methodology that is “Agile methodology”. Agile methodology is nothing but a software development methodology that gives major priority to the customer at every stage of the development process. There is a chance to match expected results and actual results at every phase of Software Development Life Cycle (SDLC) for the betterment of end software product. Also, scope will be given at every phase for the adjustment of minor/major modifications to improve the level of customer satisfaction phase by phase. Our paper mainly focuses to make software development requirements as dynamic to fulfill the highest level of customer satisfaction. In our paper, we are going to implement the concept of iterative and incremental processes to develop the project in the IT industry with the help of time-boxing technique. A hybrid Software Development Life Cycle (SDLC) model, which is a combination of Iterative and Incremental Software Development Process models was designed and developed in our study. Agile methodology mainly focuses on the inclusion of the customers/end users in the software development process itself. This methodology also releases the project after completion of each iteration for the betterment in the next iteration and so on.

Keywords: Agile Software Development, Dynamic requirements, Limited Time-frame, Iterative approach, SCRUM methodology, SDLC, Incremental Approach, Customer Satisfaction

INTRODUCTION

Agile is a software development methodology based on time-box mechanism as well as iterative process. Time-box means a software developer assigned with specific time - frame at that period of time you have to develop and release the software with their proper functionality. The time-frame may be approximate 1 to 4 weeks to complete each iteration of the module. It may be even longer in synchronous with higher customer satisfaction [17]. This time frame will be continued until the software does not become completed. Agile methodology has developed for design the software without redesigning [19], rebuilding and refactoring [4]. Therefore, 90% of the IT industries are implementing Agile methodology to develop the software [10, 17].

This methodology helps us to know what the customer has actually requirement or desire [20], because the discussions that have taken place with customer and developer are developing exactly same project or not. Related to the module of the software that is required to the customer that must be available in the product which is under developing phase or has been developed. Means that the customer observes whatever they planned for the same

product is delivered or not, if it is not same as earlier discussion than developer will apply one or more iteration to complete their requirement or functionality properly.

The aim of the agile methodology to involve the clients with developer and remove the communication gap between clients and developers [6]. The customer also see and observe the developing phase of software at the end of each iteration, due to this customer always involved with software development cycle itself more and more, that is introduced as agile methodology.

An Agile software development process is defined as the combination of methods, best practices, various activities and complete transformation that are being used to build and assure the software and their related product. The main issue of the software developer is that how they organize the software development to fast deliver, better performance and cheaper cost that are discussing in software development industry for decades [2]. The concept of Agile methodology comes in mid-nineties with various principles like SCRUM [6], eXtreme Programming (XP) [12], Crystal [16] and Feature Driven Development (FDD) etc. [10].

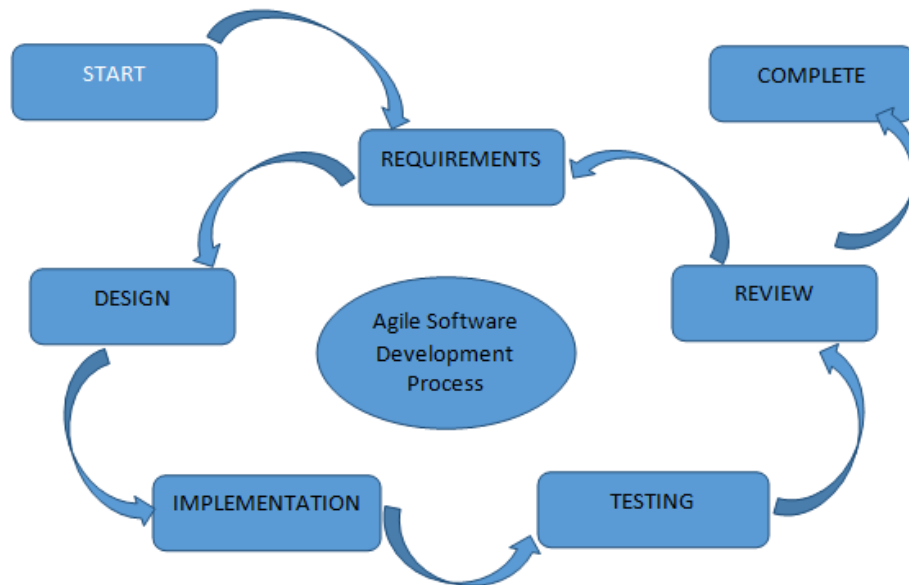


Fig: Agile Software Development Process [13]

BACKGROUND

Agile software development concept has come in 2001 through agile teams in a conference in University of Utah [3, 17]. Then after so many software companies and project development team understand and accepted agile methodology and implemented after that team turned into a modern era [1]. An agile process is only one that are always ready to support the degree of changes and flexibility [2, 5]. In traditional methodology if customer don't know about requirements then they take help form developers but in agile methodology if customer and developer both of them don't have full idea about requirements then still project starts [7].

Traditional software development methodologies are not enough sufficient to change the requirement quickly [16]. Like as in waterfall model, generally clients were involved in very early phase after that clients has no role to interact with developer by this reason client sit down at their home, and wait for completion of the project. Due to this client has no idea about which module is under development and what is going to be delivered to the customer. Because at earlier phase developer has freeze the client requirements within a particular time period so that customer was not able to add/modify any features flexibility.

Advantages of agile method for software development:-

- To make requirement as dynamic to change requirement according to customer satisfactions [9].
- Client gets opportunity to see what is getting developed during the development phase /process itself.
- Maintain the high communication between customer and project developer.
- No need to wait the customer to see the project till the end of product development.

- Organization ensures to get the confidence to the customer that what we have developed it is equivalent to our expected.
- We can easily add and modify new requirement and deliver to the client, even though the development has been already started.
- In Agile methodology customers feels 100% satisfaction.

Agile developments and principles:-

1. The main priority is to provide customer certifications by continuous delivery of software.
2. Always ready to change the requirement, even sometimes software developed late. We change the requirements according to customer compatibility.
3. Develop the software which is being developed within a couple of weeks to reduce the time scale.
4. Client people and project developer team must work simultaneously at project development time [10].
5. Develop the project as individuals, provide the environment, fully support, requirements and believe them to complete their work on time [2].
6. Develop the project by individuals, and they should have better environment to support what client has requirement.
7. The main objective is to maintain face to face conversation between customer and project development teams[9,11].
8. Maintain as simplicity as customer convenient for minimize the work load.
9. Agile process maintain the sustainable of project development.
10. Focus continuously for excellence design.
11. Every some period of interval, the team reflects to get more effective then adjust their behavior according to the situation [1].

Agile Introduction?

Agile is a software development methodology based on several iterative [9] and incremental methods. This method first splits the project into small builds. And this builds works with iterations. Each of the iteration continues up to 1 to 4 weeks and each iteration includes cross functional team to develop the product simultaneously [17]. All of the iterations have following phases like requirements, designing, implementations, testing, and reviews. Therefore at the end of each iteration project shown to the customer to confirm their requirement satisfactions. This is an umbrella having well defined methods, with reference to English dictionary, “agile” work has two meanings:-

- (Mentally quick) should be able to think properly and clearly.
- (physically quick) should be able to flexible your body quickly and frequently) [2].

Traditional Methodology versus Agile Methodology

In traditional software development approach like waterfall we cannot change the requirement once decision is completed. If we want to change the decision and requirement then we have to start the project from initial stage. Whereas Agile process provides the background scalability to use the iteration. Where we change the decision and implementation according to client convenience [5].

Agile provides flexibility to check the error at the time of development phase wherever clients and developer wants. Agile has less error and bugs free in compare of waterfall model, because waterfall checks the bugs at the time of completion of module development.

Since, Agile provides the degree of flexibility to change the project requirement for maximum customer satisfaction. Whereas waterfall model is striated to change any modification once module has been developed completely [1].

Table 1: How Agile Development is different from Traditional Development [2], [3], [11] [16], [18].

Traditional Methodology	Agile Methodology
Maintain the linear orderly, step by step, rule driven.	Implements iterative & incremental process.
Follows synchronous and sequential process.	Follows asynchronous concurrent process.
People changes in each phase.	Same people works until completion of project.
Maximum documentation is require.	Less documentation is require.
Developers wait to get ready the architecture.	All the developers work all together on same iteration.
Slow to response to the client.	Give quick response to client feedback.
Difficult to change requirement at later stage.	Easy to change requirement at any stage.
Spent more time to construct the design.	Less time to construct the design.
Prohibited to access the architecture.	Allows to everyone to see and understand the architecture.

View of the environment is stable, predictable.	View of the environment is turbulent, difficult to predict.
Type of learning is Single-loop/adaptive.	Type of learning is Double-loop/generative.
No communication required between client and user at development time.	More communication and iteration required at development time.
No any daily basis meeting.	Arrange daily meeting and discuss about development process.
Project will handover after completion of all the development phase.	At any stage of development client can see and analyze the project.
No client involvements at development time.	Clients always involve and discuss.

DIFFERENT TYPES OF COMMON AGILE DEVELOPMENT METHODOLOGIES

Agile software development methodology provides a way to develop the project in short time, best quality and having less cost [16].

Extreme programming (XP):- Extreme programming is a methodology that is used to develop the software in an unstable environment. We should use Extreme Programming when the size of team is generally small that is 2 to 10. Generally iteration round should be about 2 weeks. XP methodology is not better for distributed teams [4].

Extreme programming development can be done with following five values. First maintain the strong communication between customer and developer. In second design the project with as simplicity as customer understand. In third round take the feedback [18]with customer after continues changing of design. Fourth, developer encourage the customer to detect the problem, and finally in fifth give full respect to the customer for arising new requirement [3, 16].

Extreme Programming is very famous because it focuses the customer satisfactions. It also allows the flexibility in the modeling process. The main objective of (XP) is to reduce the cost of changing when we change the requirement with traditional software development methodology, like waterfall methodology requirements of the software always “frozen” at the beginning of the project development [1]. That’s why, if client wants to change the requirement at later stage of the project then cost can be very high [16].XP is first agile process.

XP contains a group of individual best practices and we group together for the successful practices [2].

- **Whole team:-**In extreme, team performs very important role. That may contain software developers to develop the software, software tester who takes care for quality assurance [16], and it also include analysts to cooperate in designing phase and also generate the feedback.
- **Planning:** - Planning is most important to calculate the estimation of effort and cost project development. These

planning are very mandatory because projects are visible in everywhere in all the times. There are following two types of planning in XP Methodology.

- **Release planning:** - The customers describes all the features to the developer what they expected by him in the project and also expect when it will be delivered [20]. The developer goes through all the features and detects all the problem constraints. After reviewing all the features developer decide about the initial release for next development of software.
- **Iteration planning:** - A project must work with iteration which should be a few weeks. After completion of each iteration project shows to customer to confirm their expected feature so that developer can ready for next iteration to add new features.
- **Small releases:** - The project team releases the running software after completing of each iteration [3]. The customer review and evaluate the project and give the feedback to the developer [18]. If customer is not satisfied with their project then he/she tells the entire problem and developer adds the entire feature for next release.
- **Pair programming:** - In pair programming two programmers write the code in a single machine. This helps the programmer to review the code of another programmer. This will help for better designing, coding and test the programmer. With the help of pair programming a developer can switch their work and this will also help to all other programmer to understand the system.
- **Collective code ownership:** - In XP code is developed by whole team, no any module is developed by single person. In XP every programmer do their work based on given time. So all the people can see the entire code that improves the quality and minimize the defects. If code is owned by single then the probability of error may increase.
- **Sustainable pace:** - The projects team develops the software at constant speed. Due to sustainability of time the code of each module developed on time for better delivery having better quality.
- **Customer Tests:** - In extreme programming customer checks the project with the help of acceptance test. This will support to validate the product features [2].

Feature driven development (FDD): - Feature driven development concentrate only on feature of the software system. These are the main backbone of the whole project development process.

- In initial stage of the FDD process construct the detailed model of the system that contains all requirement assumptions of the stockholders.
- When the domain model is built once. Then the project team members print a list of system features. Every feature should be developed in some period of time like hours or days but not more than two weeks.

- The project development will be done in parallel for all the features [10].
- The FDD process imposes strict guideline to detect the defection in the system. It also implements coding standards [19] and encourage to add newly designed features to the system in daily or weekly basis.
- For parallel development of all the features it is compulsory to maintain a configuration management system to integrate the entire feature which has been changed for the system [1].

Crystal: - Crystal methodology is only the adaptable and lightweight approaches for software development. Crystal methodology is developed by Alistair Cockburn. This methodology is mainly used for small project when project team has maximum 10 developers [16]. Actually crystal methodology includes a group of agile methodology. Those are like as crystal clear, crystal yellow, crystal orange, crystal red, crystal diamond, and crystal sapphire and so on. It inherits their unique advantages with the help of several factors like as project priorities, team member size and system criticalness.

In compare of other agile methodology, crystal provides early development, frequent delivery of working project, high involvements of users, easily adaptation, remove the distraction. It includes many key features like teamwork, flexibility, communication and simplicity to improve the frequency of process [1].

SCRUM:-Scrum is a framework that is used to develop small as well as complex projects [16].This is also lightweight methodology for the software development. This concept came in 1995 by Ken Swaber [3]. Scrum methodology helps to developer to find out some tasks after each iteration. Since due to light weight it can easily manage the changing of requirements. And also release the software in each iteration or small releases [19] cycle that is called sprint [3]. Scrum generates more revenue and best quality of the software [1]. In Scrum iteration size goes to 1 to 6 weeks. Here project may include multiple teams so it supports distributed teams [4].

In this approach software development is divided into small chunks that are called as **sprints**. Where each of the **sprints** delivers a working piece of software. Here development team does not go for next phase until it will not successfully completed current phase [3].Here project owner prepares a backlogs [6] that contains the pending works [3].Generally this backlog says the features to be developed or any improvements in the existing functionality. After that we prioritized the items with most important on the top and less important at the bottom which can be changed after sprint on iteration. This **backlogs** act as the input to **sprint** planning meeting.

Sprint planning: -In Sprint planning the development teams decides what we have to develop in next sprint. This also helps the software developer to calculate how much time is required to complete the project [20]. The Scrum teams select the work for the coming sprint and send it from product backlog to sprint backlog.

Product backlogs: -A product backlog is set of work in prioritized based for the development teams. It contains all the features of each functionality.

The main purpose of this meeting to decide the work to be completed with proper planning. Then after planning work would be started, here every day development team come together that is called scrum, which discusses what every employee have completed today and what they have to do before next day meeting this will help us track the process and increase the speed up process of development.

Scrum uses three important roles to complete the tasks:-

- **Product owner:-** In scrum, the product owner maintain the communication between client and development team. He/she gives full effort to increase the interests having requirement and prioritization. Because the product owner has all the authority and responsibility to manage each module of the project.
- **Scrum master:-**The scrum master provides all the facilities to the project team and product owner which are related to the software development. Actually Scrum master does manage the project development team but act as a facilitator. Scrum master reduces the impediments [6].
- **Team member:-** All the team members have responsibility to complete their work at the time. Actually team members are functionally combination of seven different types of cross functional members. For every software development is having combination of software engineers, architects, UI designers, analysts, programmers, QA experts and testers. In each sprint, team members are responsible to determine the project and gives full effort for better performance [1].

On final scrum day, it arranges two additional meetings:-

Sprint review /demo: - Here the aim is to improve the product quality. We show the product to client and the project owner whatever team has completed, and take feedback from the client and project owner [6].

Sprint retrospective: - It tracks all the possible blocking in the sprint and tries to solve out these problem which is occurring [6]

CONCLUSION

The various software development methodologies have been involved since 1970s. Due to those existing methodologies an Agile methodology has introduced for the convenience of software developers and project teams. Agile methodology uses iterative processes to provide the flexibility of customer requirements that makes the project as lightweight. Agile methodology helps the customer to change the requirement as dynamically. This paper described a new type of software development methodology that is 'Agile Software Development'. So, Agile provides various benefits in compare of previously implementation of waterfall methodology. That's why people turning to agile development with the help of various agile methodologies such as XP,

Crystal, FDD and mainly most popular framework that is SCRUM methodology. Actually agile methodologies are developed for increase the flexibility, reduce the project costs, fast development, very less error rates, good qualities, high customer satisfaction, less documentations and maintain the strong communication between customer and project developer to understand the requirements. Agile development determines the requirements and percentage of project completion time with the help of various iteration.

REFERENCES

- [1] Choudhary, B., & Rakesh, S. K. (2016, February). An approach using agile method for software development. In Innovation and Challenges in Cyber Security (ICICCS-INBUSH), 2016 International Conference on (pp. 155-158). IEEE.
- [2] Pathak, K., & Saha, A. (2013). Review of agile software development methodologies. International Journal, 3(2).
- [3] Hneif, M., & Ow, S. H. (2009). Review of agile methodologies in software development. International Journal of Research and Reviews in Applied Sciences, 1(1), p1-8.
- [4] Nagwani, N. K., & Singh, P. (2009). An Agile Methodology Based Model for Change-Oriented Software Engineering. International Journal of Recent Trends in Engineering, 1(1), 128-132.
- [5] Rao, K. N., Naidu, G. K., & Chakka, P. (2011). A study of the Agile software development methods, applicability and implications in industry. International Journal of Software Engineering and its applications, 5(2), 35-45.
- [6] Jha, M. M., Vilardell, R. M. F., & Narayan, J. (2016, August). Scaling Agile Scrum Software Development: Providing Agility and Quality to Platform Development by Reducing Time to Market. In Global Software Engineering (ICGSE), 2016 IEEE 11th International Conference on (pp. 84-88). IEEE.
- [7] Ionel, N. (2009). Agile software development methodologies: An overview of the current state of research. Annals of Faculty of Economics, 4(1), 381-385.
- [8] Alawairdhi, M. (2016, May). Agile development as a change management approach in software projects: Applied case study. In Information Management (ICIM), 2016 2nd International Conference on (pp. 100-104). IEEE.
- [9] Elghariani, K., & Kama, N. (2016, August). Review on Agile requirements engineering challenges. In Computer and Information Sciences (ICCOINS), 2016 3rd International Conference on (pp. 507-512). IEEE.
- [10] Ambler, S., & Lines, M. (2014). Scaling Agile Software Development: Disciplined Agility at Scale. Disciplined Agile Consortium White Paper Series.
- [11] Greer, D., & Hamon, Y. (2011). Agile software development. Software: Practice and Experience, 41(9), 943-944.
- [12] Dingsøy, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development.
- [13] Mateen, A., Tabassum, M., & Rehan, A. (2017). Combining Agile with Traditional V Model for Enhancement of Maturity in Software Development. arXiv preprint arXiv:1702.00126.
- [14] Sanghvi, A., Sehgal, P., Kapur, A., & Singh, S. (2017). INFLUENCE OF THE NUMBER OF PEOPLE IN AGILE SOFTWARE DEVELOPMENT.
- [15] Cuellar, R., & Augustine, S. (2010). Agile Software Development: Teams.
- [16] Ahmed, A., Ahmad, S., Ehsan, N., Mirza, E., & Sarwar, S. Z. (2010, June). Agile software development: Impact on productivity and quality. In Management of Innovation and Technology (ICMIT), 2010 IEEE International Conference on (pp. 287-291). IEEE.

- [17] Vijayasathy, L. E. O. R., & Turk, D. (2008). Agile software development: A survey of early adopters. *Journal of Information Technology Management*, 19(2), 1-8.
- [18] Dyba, T., & Dingsoyr, T. (2009). What do we know about agile software development?. *IEEE software*, 26(5), 6-9.
- [19] Adelyar, S. H., & Norta, A. (2016, September). Towards a Secure Agile Software Development Process. In *Quality of Information and Communications Technology (QUATIC)*, 2016 10th International Conference on the (pp. 101-106). IEEE.
- [20] Edeki, C. (2015). Agile Software Development Methodology. *European Journal of Mathematics and Computer Science Vol*, 2(1).
- [21] Ajmal, S., & Ali, S. AGILE-waterfall hybrid model for software development processes.
- [22] Chóliz, J., Vilas, J., & Moreira, J. (2015, August). Independent Security Testing on Agile Software Development: A Case Study in a Software Company. In *Availability, Reliability and Security (ARES)*, 2015 10th International Conference on (pp. 522-531). IEEE.
- [23] Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Kern, J. (2001). Manifesto for agile software development.
- [24] Agrawal, A., Singh, S., & Maurya, L. S. (2015, March). A study on the growth of Agile methods in India till 2014. In *Computer Engineering and Applications (ICACEA)*, 2015 International Conference on Advances in (pp. 370-374). IEEE.
- [25] Cottmeyer, M. (2008, August). The good and bad of Agile offshore development. In *Agile, 2008. AGILE'08. Conference* (pp. 362-367). IEEE.
- [26] Cho, J. (2008). Issues and Challenges of agile software development with SCRUM. *Issues in Information Systems*, 9(2), 188-195.
- [27] Stoica, M., Mircea, M., & Ghilic-Micu, B. (2013). Software Development: Agile vs. Traditional. *Informatica Economica*, 17(4), 64.
- [28] Pohl, C., & Hof, H. J. (2015). Secure Scrum: Development of Secure Software with Scrum. arXiv preprint arXiv:1507.02992.
- [29] Mishra, A., & Dubey, D. A Cynical View on Agile Software Development from the Perspective of a new Small-Scale Software Industry.
- [30] Balaji, S., & Murugaiyan, M. S. (2012). Waterfall vs. V-Model vs. Agile: A comparative study on SDLC. *International Journal of Information Technology and Business Management*, 2(1), 26-30.

AUTHOR'S PROFILE



Dr. Bonthu Kotaiah obtained his Bachelor's degree in Computer Applications from ANU in 2001 and M.C.A from ANU in 2008. During the period from September, 2001 to 2011. He worked as an engineer(L-Cube Innovative Solutions), a Corporate Trainer (SyncSoft & Datapro(Vijayawada), COSS(Hyd.)), a Computer Programmer(Acharya Nagarjuna University).

His research interests include software Engineering, Neural networks, Fuzzy Logic. He completed his Ph.D. in Information Technology(IT) from Babasaheb Bhimrao Ambedkar University (A Central University) in the Dept. of I.T. Presently, he is working as Assistant Professor in the Dept. of CS &IT, MANUU, Hyderabad.



Md. Asif Khalil completed his B.C.A. from MU, Bodh Gaya in 2011 and M.C.A. from MANUU, Hyderabad in 2015. He is a researcher in the Dept. of CS&IT, MANUU, Hyderabad and his research interests include Software Engineering and Programming Languages. Presently he is pursuing in M.Tech. (CS) from MANUU, Hyderabad.