

# Agile Based Software Development Model : Benefits & Challenges

Tajinder Kumar

*Assistant Professor, IT Department  
JMIT Radaur, Haryana*

Vipul Gupta

*Assistant Professor, IT Department  
JMIT Radaur, Haryana*

**Abstract - Agile software development teams need highly collaborative knowledge to achieve the software development activities. Agile teams are self organizing and cross functional teams which advocate sharing of intensive knowledge through the face to face connection, active communication and close collaboration between team persons to work together as a single team. In distributed agile development is defined a group of activities which have ability to work together as a single distributed team. These group of members work as collaborate team on a software project separated by long distance, different time zone and culture. Distributed agile team is used for work as a distributed team over the world but due to separation of long geographical distance, different time zone and culture affect leads to weak collaboration which negatively affects the software project. But using agile approach in distributed development is useful for various organisations to increase performance and quality of the software project. Agile is important to achieve benefit of markets, customer satisfaction and reduce the cost of software which is distributed all over the world. The aim of the paper is to define the agile process and methodology such as Scrum, Extreme programming in distributed software development as a strategy to provide faster and efficient result to the customer and defined onshore and offshore interaction for distributed agile development. The goal of this approach is to understand the benefits and challenges of distributed agile development.**

## I. INTRODUCTION

In recent years, software engineering has been adopted to the decisions appearing in the distributed software development process. In the background, two processes could be emerged and they are discussing of this paper i.e. Distributed Software Development and Agile Practices. Both processes have been combined recently for developed the software efficiently . In the last few decades, a great interest is being made to replace local market to distributed market. This process needs more practice and collaboration. This process involves various challenges like project failure have to be dealt with. Many institutions start the experiment with the distributed software development to solve these challenges. Distributed software development is not a new method. Large organization built the software across multiple sites with the multiple teams. Teams are working from different geographical locations, different time zones which is the part of distributed software development. Responsibilities of these teams are built a cohesive product. Distributed development is used in reducing cost and access relevant information. The main focus of distributed team is to develop cohesive product at the lower cost, cross site modularization with the interaction of multiple teams. Product is developed in multiple sites and distributed environment with the product engineers, managers, designers, developers and testers which help to build the product and delivered to the customer. Some previous years distributed software development process is adopted agile practices for distributed agile development. Agile is iterative and incremental development process where collocation supports face to face conversation, close communication between cross functional teams. Agile methodologies work very well in IT organisation and developed highly collaborative software at lower rates. The objectives of this paper define the benefits achieved by adopting agile practice with distributed environment and provide business value early to customer. In distributed agile development, members of distributed team have ability to work as a single team. The paper is structured as follow: Section II defines the distributed agile development. Section III defines the challenges which meet in the distributed agile development. Section IV defines the agile methodology which can help to improve the challenges. Section V discusses the approaches for distributed agile development and define efficient offshore and onshore interaction.

Section VI defines the Benefits which achieved using the agile methods and approaches. Conclusion is in Section VII.

## II. DISTRIBUTED SOFTWARE DEVELOPMENT

Software has become a main part of the industries. The success of industries depends on the software which is developed over the other competitors. Many industries have started software development remotely to reduced cost and access the resources for software development. Distributed development is a traditional technique where number of organization built large software systems across different locations with different teams to achieve the higher productivity, profits and good quality. This type of development is the responsibility of integration teams where developers, testers and designers work in several teams on different location in the form of Onshore and Offshore. Onshore which is located at the same site and Offshore is located out of the country and both works on the same project. Both Onshore and Offshore work independently at different locations to achieve higher productivity according to the customer requirements. The development can be classified into:

1. Distributed evolvment which involves collaboration between different teams which located at different places and having different time zones. It involves 24 hours development and high experienced labour. It will incorporate the effort where large amount of work has to complete with Onshore and Offshore teams with the stakeholders.
2. Distributed development includes the individual team members i.e. (Developers, Testers, Designers, Project managers). These team members are located differently and working as a team over a network. These teams include the experienced members who are working at their home and office.
3. Minimise the risk in case of natural disasters and other problem

There are different ways to distribute the software. Distribution can be done on the basic of geographical locations. It can also distribute by the control structure that may be done by two things i.e. Outsourcing and Insourcing. Outsourcing means organization purchases the software from outside. Insourcing means organization provide software services itself internally.

The geographical location is defined by onshore teams which are located in the same country and in perform operation in the same country and offshore teams which are located abroad. Offshore distribution is also called global software development. Distributed software development is spread over the world but there are several issues on the road of success to the different geographical locations, different time zones and different culture. Overcome these challenges, we use Agile practice in distributed development. Distributed Agile Development is a practice in which agile methods are performed by teams for software development that are distributed over the different location.

## III. CHALLENGES

The reason to used distributed development is to utilize lower cost, high productivity and quality. But the communication between the teams is very low due to the different locations and time zones and also difficult to collaborate face to face. These issues are also creating miscommunications between the teams. Due to the separation of distance motivation of the team has been reduced and no relationship between the teams. There are few occurrences of challenges which have to overcome:

### 1) *Reduced Communication :*

Communication between Onshore and Offshore team is reduced due to the different time zones because the working hour of both countries are different and few hours in a day when both present at the same time. This factor reduced the communication bandwidth.

### 2) *Reduced sensing of project progress:*

In distributed system is difficult for project managers to know correct sense of the project progress and correct status of the project because project teams are located in different site and collaboration made worse between the teams.

- 3) *Configuration Reduced:*  
Implementation is one of the most part of the software development when all team members work together but it difficult in distributed team to bringing it together because offshore teams have been troubled by few problem when the integration time come.
- 4) *Decreased project estimation:*  
The project estimation is done by the project manager and customer but due to the separation of different countries, teams have no any interaction with the project manager and customer. So it is difficult to estimate and get the progress of the software project.
- 5) *Distant collaboration & communication:*  
In distributed environment, it is difficult to find out the way of communicating efficiently between the Onshore & Offshore teams. Some emails are used to maintaining the daily communication but we need the detailed discussion for the software development. Emails message gave slow estimate communication result and create miss communication between teams and difficult to know the customer requirements. The project grew in size and difficult to maintain the emails spread sheets.
- 6) *Increased documentation:*  
Documentation increased between Onshore & Offshore teams due to the close communication. Both teams had to commit on the written collaboration means documentation. Due to distributed nature import information like conference calls, meeting details, customer requirements should be send as documented form create heavily documentation.
- 7) *Wasteful remote collaboration:*  
Remote collaboration is used for organize the meeting with the Onshore & Offshore teams. The problem arise when some of team members become offline during the remote collaboration calls meeting, by technical issues, bad quality combined with the remote call like video conference and voice chat due to bad quality remote listeners lose track.
- 8) *Team coherence and bounding:*  
In distributed environment teams are distributed over different locations and need some extra efforts for team cohesion and maintain trust between the team members. When a new member comes in offshore team it creates problems in team collaboration and chemistry due to the lack of cohesion and close face to face communication. This factor also affects the trust between the teams.
- 9) *Methodology differences:*  
Both the teams have differences in their methodologies because there is no visibility of the way of working due to the lack of close collaboration.
- 10) *Unavailability of customer proxy:*  
Offshore teams that located at different location of the country had no collaboration with the customers located at Onshore and it makes difficult to the availability of customers proxy for offshore team. It also create problem of unavailability of customer proxy with the project manager and developer of the offshore teams.
- 11) *Technical problems:*  
Problems in technologies and infrastructure between the Offshore & Onshore teams-  
Example: The Onshore & Offshore team abort to maintain coding standards and focused on the previous ones. Problems create due to the different operating systems and applications used by the onshore & offshore teams.

## IV. AGILE PRACTICE USED BY DISTRIBUTED DEVELOPMENT

Agile is iterative and incremental method with a group of methodologies. The goal of the model is to develop the software efficiently and effectively. These methodologies contains light weight document that are reused at any stage and deliver the software according to the customers satisfaction.

In agile systems, feedback comes immediately from the customer's side which improve the quality of software. It affects the big changes in modern business. The business environment needs the development teams which continue to change the IT business and give response early in favour of the customer's satisfaction. So the agile practice fulfils these conditions.

Agile manifesto has been implemented on 17 February 2001 and contain some core points i.e.

- 1) Individuals and intercommunications over methods and techniques.
- 2) Developed software over intensive documentation.
- 3) Provide close communication between plans for development.
- 4) Provide customer satisfaction by providing good quality software.
- 5) Develop the software rapidly and estimate the progress.
- 6) Close and intercommunication occurs daily between the Onshore & Offshore team.
- 7) Changes done in any time according to the requirement.
- 8) Use advance techniques like video conference and voice chat for face to face communication.
- 9) Software developed by the experienced and motivational members.
- 10) Trusts maintain between the Offshore & Onshore teams.
- 11) Self-describing oriented teams.
- 12) Daily meetings to maintain circumstances.

Agile is a process in which agile methods and key values are developed by a team who are distributed in the same place and out of place. Agile methodologies work with the distributed environment for development of efficient software using agile practice. Agile approach is dependent on cross functional and self-describing team which have good communication and collaboration between the team members and customers. Customers resented in one place builds the trust between the teams. The waterfall model also used by the distributed teams but in waterfall model each stage work separately and commitments are made earlier and it is difficult to react with late changes in the requirement. In agile method, Product completely developed and tested at the same time means, we have to perform coding and testing together in single iteration. Agile perform one or more iteration for product development in every week. Many methods are used but mostly scrum and extreme methods are used in programming. In scrum, software is implemented using sprints with planning and iterative process. Each sprint starts with developers, testers and sprint manager in a week when product developed, product is sending to customers for reviews in the end. In scrum, one or more sprints are performs for a small period of time with detailed plans and backlog. In backlogs, all sprint information, daily meeting information for close communication between team members with scrum master are stored and product manager is used for developing the software.

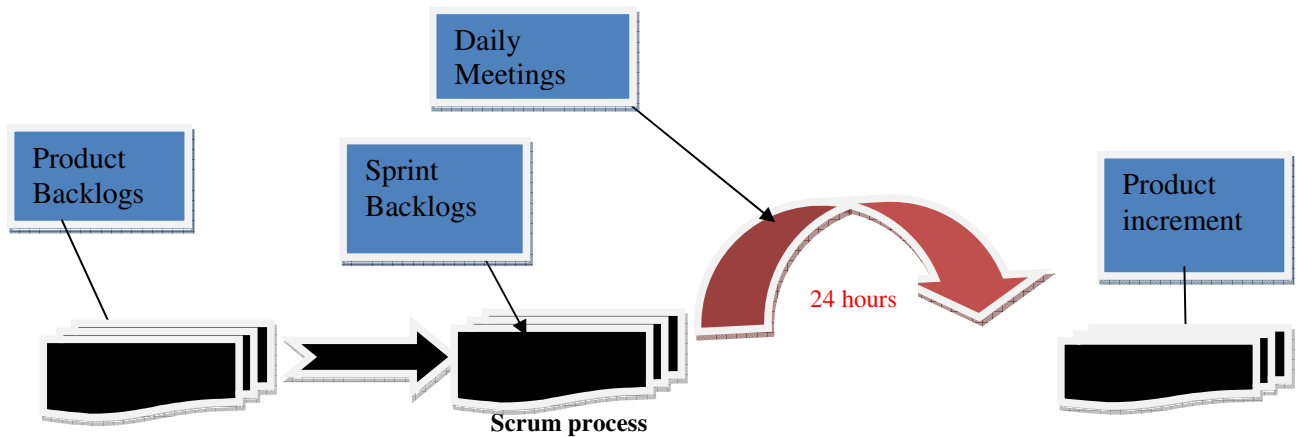
In extreme programming, we concentrate on the development of the processes. Best process like pair programming is used in XP. Pair programming, where the pair of developer implements the software. Agile methods have produced more productive methods than the traditional methods.

In 2008, agile development serve accomplished by the version one, say that 57% organisations teams are distributed while 41% organisations said that they were using the agile practice approach to combine with the distributed development. We describe how scrum were applied in the distributed environment:

*Scrum meetings held every day:*

Scrum meetings organize between the onshore and offshore teams using the video conferencing and voice chat. In scrum, sharing work also has done. The meetings organize within two and three hours daily when all members are presented. Meetings run consecutively and take place in the same room. The first meeting start at early morning and another one start after few hours and so on. This consecutive order makes it easy to all members to attend the all meetings. Example, few additions meeting of the members in the team which are located London and project owner also join the meeting which is located India and meeting is still running. After the some discussion it's easy to know the each other. In the beginning, it was difficult to communicate

each other and tell enough related to their tasks because most of team members are located at different country and meeting held only a few minutes. But using scrum, daily scrum meetings are organized and communication is improve. If the more discussion are needed then plan of few additional meetings.



#### *Distributed sprint:*

The sprint is used for planning meeting which are divides into few phases: Distributed scrum meeting, local meeting if one team located in India and local meeting if another team is located in London.

First sprint is for three or four hours. Both team members attend this meeting using the teleconferencing. Teams use sharing application. In sprint meetings, web cameras are not used. The scrum framework is used which describe the how the scrum process is applied and take list of reviews from the customer. In every spring meeting, start discussion about the previous sprint meeting and new concept plan according to the new requirements. The sprint meetings are provided the training to the team members. After the product owner starts talk about the product in the backlog, it must be necessary to maintain the backlogs where all information related to the sprint meeting, future plan and customer requirement are stored. Both teams divide the backlog items to perform the task separately with the estimation of the product by product owner. The onshore team start the work on the product because they are located at same site and offshore team continues the work next morning and meet with the onshore team. In next morning, start discussion about the work which is done by onshore team and start work.

#### *Weekly scrum of scrum:*

Scrum of scrum meeting is organized in a week. Project manager of the team participates in this meeting and all scrum masters also involve in this meeting. The meeting takes for 1 hour and during the meeting talk about the project progress and scrum question. The question is not for only single person but for whole team. Each project member of team tells the team progress and what has done since last meeting, what do before the next meeting and what type of implementation will achieve by the team. Moreover, they have asked few additional questions are to ensure successful implementation.

#### *4 week sprints for synchronization:*

The framework team have organized 4 week sprints. This means all sprint meeting start and finish at the same time. All sprints have the same structure and after the end of sprint they organize the environment for the demonstrations. Changes of modules between the teams and exchange members from one to another also have done.

#### *Maintenance sprints for 2 week:*

The maintenance sprint is used for overcome the drawbacks of 4 week sprint cycle. The reasons why use 2 week maintenance sprint are hot fixes release every 2 weeks. The maintenance team has the all detail of the previous sprint and they pick the set of challenges from the backlog and overcome these challenges because sometimes customers need fixes at that time. Maintenance team have fast track plans for handling the challenges and team have a buffer of the more capacity for overcome these challenges. Buffer is needed only for some level when large issues are requested.

#### *Sprint Demonstration:*

Demonstration is organized using the technologies in the sprint meetings i.e. (Shared application and conferencing tool). In this demonstration, product owner and scrum manager participate. The team prepared presentation for the demonstration. In presentation, all challenge overcomes and everybody satisfy.

*Retrospective Discussion:*

Retrospective discussion starts after the demonstration. Project owner and scrum manager participate in the retrospective discussion. Retrospective meeting is arranged in the form of distributed meeting. The time limit is maximum 1 hour and during the retrospective meeting team discussed few questions: What has been done during the sprint? What has not been done during the sprint? What type of progress could we do?

*Perform automated testing at night:*

The teams use the centralized version control system where all code is stored. The centralized version control system can be accessed by all team members. The whole product is developed and tested during the night. If the development is not successful then team will fix the faults and redeveloped after the faults have been resolved.

*Managing separate backlogs:*

The teams use a tool for managing the backlogs called Jira. Every team have directly access to this tool. Backlogs are updated by the product owners and teams have an own backlog where product owner add the new issues. All customers directly connected to Jira where they report of fault. The organization check the issues if the issues are related to the product then they assign to product development team for resolve it. These issues are based on priority. The product owner stores the issues in the backlogs with the status. If the issues resolve by the team then status is verified and issue is resolve otherwise unverified. All teams have been fulfilled with this tool.

*Frequent Visits:*

The team members travel quickly between the sites because several members are located out of the site. These visits between two or four week which makes it better communication between team members. In the previous method, there have not plan for the travelling schedule but in scrum process, people travel when needed. The half of team member has travelled to the other site and maintains close collaboration.

## V. BENEFIT ACHIEVE USING THE AGILE PRACTICE

*Communication:*

Good communication occurs between all teams using the agile practice.

*Security:*

Using the routers, confidential and secure data is transferred between the team members. The secure and private network will be helpful.

*Remote Communication:*

Agile method provides the close communication in distributed development. In agile we used video conferencing, shared application, web based collaboration tool and messenger for face to face communication.

*Repositories maintain:*

Repositories code is a requirement of all software development systems.

*Advance technologies used:*

Used screen shared application, web cameras and video conferencing .Also used the combination of Skype and web cameras to achieve daily meeting.

*Improve cultural differences:*

Frequent visit is used for improve the difference of culture differences because it helpful to know the each team members which is located at the different location.

*Customer proxy and Caching:*

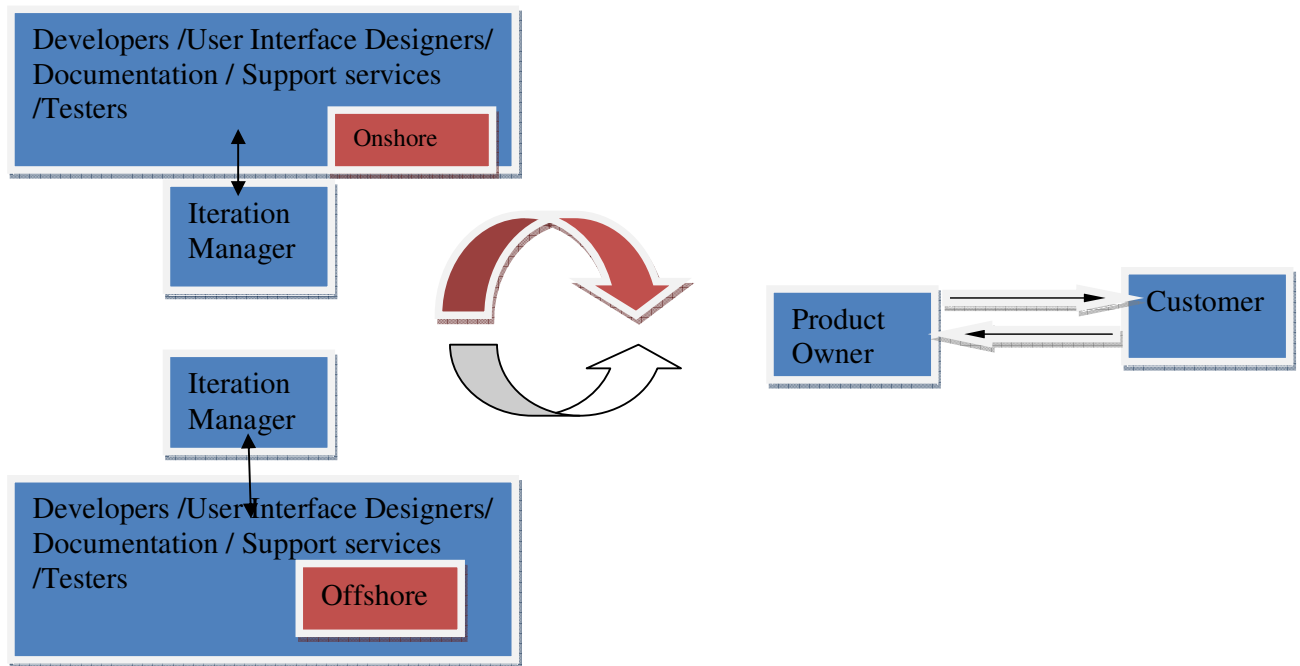
The system has ability of managing multiples servers on the distributed environment and the customer interaction is done by the system remotely.

*Advance technique:*

Used some advance technique to improve the efficiency of the product.

VI. PROPOSED STRUCTURE FOR ONSHORE AND OFFSHORE DEVELOPMENT

In proposed structure , the customer send the requirements list to the project manager which have directly contact with the both offshore and onshore team. Project manager send the requirement list to the both onshore and offshore teams where developer, user interface designers, documentation, support engineer and testers are presented for software development. Both teams separately work according to the customer requirement and developed the product according to the customer needs where iteration performs for the high productivity of the software. At the last, they send the product to iteration manager and iteration manager send the product to product owner which is distribute the product to customer.



VII. CONCLUSION

With the help of distributed agile development is possible to get the better software product with the high productivity, low cost and better progress. There are many benefits of agile process with the distributed software development using the agile methodology like scrum.

Agile combine with the distributed development to overcome the different location problem, different time zones problem and improve cultural differences. The agile practice is used for maintain the close collaboration between onshore and offshore teams.

In this report we discuss the distributed software development using agile practice with the intension of high productivity, quality and low cost. The future work is to find the more agile methods which also used in the distributed development to improve the software quality.

REFERENCES

- [1] SEI, “ CMMI for Systems Engineering, Software Engineering,Integrated Product and Process Development, and Supplier Sourcing(CMMI-SE/SW/PPD/SS, V1.1) Staged Representation,” Technical Report CMU/SEI-2002-TR-012 ESC-TR-2002-012, Software Engineering Institute, 2002.
- [2] Agile Manifesto, Manifesto for Agile Software Development, <http://agilemanifesto.org>.

- [3] Manish A, Kaushal C, "Software Effort Quality and Cycle Time: A Study of CMM level 5 Projects ," IEEE Transaction on Software Engineering, vol. 33, no.3, pp. 145-156, 2007.
- [4] Scott W. Ambler, "Requirement Specification in Agile Projects" <http://www.ambysoft.com/scottAmbler.html>.
- [5] IEEE Computer Society, " IEEE standard for Software Reviews," IEEE Std. 1028-1997.
- [6] Cockburn A., "Williams L, The Cost and benefits of Pair Programming," XP (2000).
- [7] Torgeir Dingsoyr, "Postmortem Reviews: Purpose and Approaches in Software Engineering, Elsevier B.V.(2008).
- [8] Israel Gat, "Coherence of Vision, Cutter Consortium," <http://blog.cutter.com/author/israelgat>.