Agile Software Development in Global Software Engineering

Pawanpreet Kaur
Computer Science Department
Chandigarh University, Gharuan, India

Sumit Sharma
Computer Science Department
Chandigarh University, Gharuan, India

ABSTRACT

Global software development is the emerging trend in today's industry. The software development industry is investigating the use of agile development methodologies in distributed environment due to its benefits of better communication and coordination, improved productivity and quality. However research has shown that implementing agile practices in global software development is beneficial and challenging too. Therefore it is necessary to carefully understand the benefits and challenges of combining agile methods in distributed projects. This paper presents the benefits and challenges of applying agile methods in global software development in order to understand it properly.

Keywords

Agile Software Development, Global Software Engineering (GSE), Agile in GSE, Benefits, Challenges.

1. INTRODUCTION

1.1 Agile Software Development

Over the last decade, introduction of agile software development has brought considerable changes to software engineering field. Nowadays many organizations and software developers are using agile methods to deliver the efficient and high-quality product in the shortest time. The various types of agile methodologies being used by these organizations are extreme programming (XP), scrum, lean software development, feature-driven development (FDD), DSDM and crystal methodologies. The characteristics of Agile software development processes can be summarized as iterative and incremental development, people-oriented development, customer collaboration, light and fast development cycles, adaptability throughout software development life cycle, time-bound, parsimony and fast delivery[2]. In year 2001, several practitioners met to discuss if there was anything in common among the various agile methods and then they created agile manifesto, in which the items which were considered valuable by Agile Software Development were revealed and great attention has been provided by research community to Agile Software Development. Agile Software Development methods focuses on agile manifesto given by four values:

- Individuals and interactions over Processes and tools
- Working software over Comprehensive documentation
- Customer collaboration over Contract negotiation
- Responding to change over Following a plan[11]

The first value refers that role of humans and team interactions should be considered more valuable than defined processes and development tools. The second value implies that main objective of software team should be to deliver the working software at frequent intervals rather than large amount of documentation. One can observe from third value importance should be given to the relationship between developers and customers over strict contracts. The fourth value refers the members in the team should be prepared to make changes whenever required during the software development life cycle rather than following a plan. Agile methods helps to improve communication and reduce coordination problems in software products. These agile methods are opposite in nature with respect to traditional approaches. When following the traditional approach software is developed in a sequential manner. Requirements are gathered from clients, there is little interaction between clients and members of the development team. For designing and coding different roles are assigned to developers and finally software is tested and integrated. In doing all these ,a large amount of documentation is produced that contains information about developmental processes. Agile Software Development approach aims at overcoming these shortcomings of the traditional approaches.

1.2 Global Software Engineering

The increasingly complex and competitive market situation has resulted in Global Software Engineering (GSE). Global Software Engineering (GSE) is becoming popular in today's industry due to the globalization of many organizations. Global Software Engineering(GSE) can be defined as "Software development with teams situated at different geographic locations, from different national and organizational cultures, and different time zones". Such kind of development is known as Distributed Software Development(DSD), Global Software Development (GSD), Global Software Engineering and Dispersed Software Development. Software organizations are shifting towards global software engineering to get GSD benefits such as reduced software development costs by delegating work to countries with low labour cost, transfer of knowledge and resources, access to talented workforce, better-quality software, rapid innovation and increased productivity [9]. Despite of these benefits GSE presents a number of challenges like: communication, coordination-due to cultural differences and control-due to lack of direct supervision of the team members. All these challenges arises from three kinds of distances: temporal, geographical and cultural. The combination of all these distances makes GSE complex [16].

1.3 Agile Methods in GSE

Increased globalization has resulted in greater competition among various software development companies. There is a recent trend of applying agile methods in Global Software Engineering to mitigate various challenges of GSD like: communication, coordination and control. These methods are best for the projects facing high level of uncertainty. Research has shown that software companies are interested in applying Agile Software Development to develop the quality product and to have the combined benefits of Agile methods and

Global Software development [10]. The combination of Agile development methods and GSD is known as Distributed Agile Software Development. This combination has shown the ability to meet the various success factors of the software industry such as quality, cost and time. The problems related to complexity of software development and more focus on coordination, communication and collaboration are main reasons for growing interest in applicability of agile methods in global software development. Global software development uses asynchronous communication and computer mediated activities but applicability of these coordination mechanism is generally insufficient. This motivates the applicability of agile methods based on mutual adjustment and teamwork for coordination in distributed projects. These differences between agile and Global software development suggest that applicability of agile methods in distributed development is unrealistic and hard to implement. Large number of methods like Scrum, Extreme programming(XP) and pair Programming etc. have been tried out in Distributed development to experiment and check the viability of these agile methods in GSD. There are some empirical and industrial experience reports which state the successful implementation of agile methods in global software development.

Emam Hossain[4] conducted a systematic literature review of the primary studies that report the results of using Scrum in GSD projects and findings are expected to help researchers and practitioners to understand the challenges involved in using Scrum for GSD projects and the strategies available to deal with them. Benefits of combining these agile and GSD, increased collaboration and communication, access to talented work forces, transfer of knowledge and resource, reduced time-to-market pressures, higher-quality software, rapid innovations, and increased productivity[4].

According to Farooq(2010), effective distributed extreme programming practices are continuous integration, and on-site customer development. While, distributed Scrum practices are daily Scrum, weekly Scrum of Scrums, sprints, sprint planning meetings, sprint demo, retrospective meeting, and backlogs .

Lucas Layman[6] conducted an industrial case study of a distributed team that used Extreme Programming to understand how this globally-distributed team created a successful project in a new problem domain using a methodology that is dependent on informal, face-to-face communication. They identify four key factors for communication in globally distributed XP teams that can be helpful for methodologies dependent on informal communication in global software development projects[6].

Bose, I. (2008) performed an interesting study and show how Scrum practices were adopted to benefit from distributed software development. Multiple case studies were conducted and the collected lessons learned from twelve case studies detailing successful implementation of distributed agile software projects were summarized. The cases were analyzed from the perspective of the agile manifesto to determine how closely they follow its values and principles and to what extent they are taking the advantages of the agile methodology[5].

Previous work shows that research in the area is exploratory in nature and mostly reports the cases in which some challenges were faced and some strategies were applied. A standard approach that cover all agile methods in GSE does not exist.

2. BENEFITS OF AGILE IN GSE

Agile software development can be beneficial with respect to Global software engineering. Agile practices in GSE gives a no. of benefits [10][11] like better coordination and communication, product visibility and control, increased trust, customer involvement, clarification of requirements, improved productivity, improved knowledge sharing, quick development are explained below:

2.1 Better Communication

Many studies shows that use of agile practices like scrum and XP results in the better communication rich environment which is required in the globally distributes software projects. Communication, specially informal, face-to-face communication plays a important role in the success of a GSD projects. Frequent communication between the sites and team members improve teamwork and provide the opportunity for teams at each site to share issues and their solutions. Globally distributed teams often use emails, instant messaging (IM), telephone communication, unstructured wikis to communicate about the status of the project[6][10].

2.2 Reduce Coordination Issues

Globally distributed team generally faced coordination issues due to project distribution. The task assigned to geographically distributed team members requires strong coordination between them, because team members are dependent on each other. According to [5][6] the coordination between distributed teams was increased by making use of daily meetings, informal face-to-face communication, iteration planning meetings etc. These meetings helped in minimizing the coordination problems which are usually caused by temporal and geographical distances. As a result the coordination problems between distributed team members decreased which results in better and quick software development[10][11].

2.3 Increased Trust

Trust is one of the key factor in determining the success or failure of globally distributed agile projects. Trust can be generated and maintained in the distributed agile projects by understanding cultural differences and by increasing communication between team members[12].Team members communicate with each other on a daily basis via IM and videoconferencing, email etc. They experienced the trust between the developers located at different sites because daily communication between the sites really helped the team to gain confidence and build trust[10].

2.4 Project Visibility and Control

Frequent communication and meetings provide an effective way for distributed team members to review the current status of the project and evolution of the system's features. In Global Software Development high project visibility can help in achieving project control. According to [10] daily meetings are helpful to monitor the distributed development situation ,as team members participate in the discussion in which they share the current status of work. The daily meetings in distributed development considers the involvement of participants from all sites via audio or video conferencing, IM etc provide the project visibility and helps in spreading knowledge about current project status.

2.5 Clarification of Requirements

In agile software development, the communication is very important for requirements management activities. Discussion meetings can help in removing confusion about the work and

requirements which are in the mind of team members. The team members communicate with team leaders, project management team about the confusions which are in their mind. The main purpose of these meetings was to identify the issues and difficulties faced by the team members and to find the strategies for these issues. The members at geographic locations got the opportunity to clarify their work and requirements during these meetings[10][6].

2.6 Improved Productivity
As mentioned in [10] the use of agile practices like pair programming in distributed environment increases the team productivity. In pair programming pairs are located at different sites and communicate with each other via communication tools like teleconferencing and web conferencing. Pairing helped in sharing ideas and views which remove misunderstandings.

2.7 Improved Knowledge Sharing
Implementation of agile practices in distributed development results in improved knowledge sharing. According to [10][6] in distributed agile development junior developers are paired with senior and experienced developers. In this way there is an opportunity for junior developers to learn from senior developers and results in sharing of knowledge between them.

2.8 Quick Development
With the help of daily meetings problems are identified and resolved earlier. There is more visibility of the current project situation in distributed agile development. According to [3][10] by following agile practices problems are identified quickly and processes are improved which results in better and quick software development.

3. CHALLENGES OF AGILE IN GSE
Agile methods are well suited when customers and developers are collocated and there is frequent interaction between the two groups. Incorporating agile methods in distributed software development requires considerable effort. Despite of the benefits identified above some of the papers also described certain challenges and this section describes the various challenges involved in adoption of agile practices in distributed projects are as follows:-

3.1 Communication
One of the main challenge in the agile global software development is the communication. Communication is considered as one of the important element is GSE specially in distributed agile environment where it allows sharing of knowledge between team members, understanding of customer requirements[13]. Developers need to communicate frequently but teams located at different locations, different time zones makes this difficult[5]. Various causes of the communication challenges in Distributed Agile software development are: Different working hours/ Time Zones, Lack of communication tools, Language barriers, Lack of frequent face-to-face contact, Lack of teamwork, Low quality of communication bandwidth, High communication cost, Miscommunication of requirements, Different project background, Different culture[13][14].
These issues in communication are represented in model from as shown below:
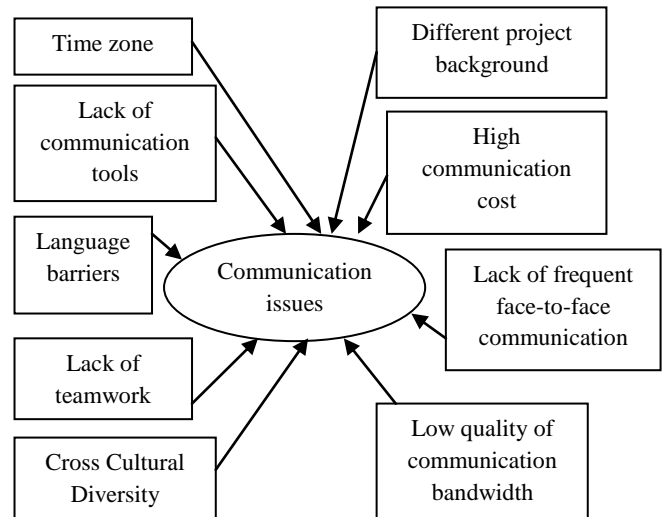
Fig 1: Model for communication issues

3.1.1 Different Time Zones
Differences in working hours is one of the important issues that needs to be acknowledged. When there in difference in time zones it increases the communication gap i.e. the communication between the globally distributed team become difficult[6][13][14].Time zone difficulties are common in distributed agile projects because the client companies are usually located in developed countries like USA, UK etc. and the development teams are located in developing countries like India[14].

"The time difference between USA and India is 12.5 hours therefore communication among distributed team members can be difficult."-[13]

3.1.2 Lack of Communication Tools
When development involves distance in space and time, communication is the main medium of interaction between distributed teams[14]. Distributed agile projects requires suitable communication tools for communication between onshore and offshore development teams as well as with the customers. Distributed agile projects may fail if distributed teams do not have proper communication tools like videoconferencing or web-based conferencing facilities. Therefore distributed teams should have access to various communication tools and use them according to their need for effective communication[13][6].

3.1.3 Different Project Background
Different project background is one the important issue in communication. Different developers from different countries have different ways of working according to their culture[6] .When there is a cross-platform collaboration then this may lead to problematic issues. If agile methods are new to development teams, then communication is very difficult for distributed members and it takes time for them to learn which information is necessary and should be transferred to other members[14][6].

3.1.4 Language Barriers
Distributed Agile development involves collaboration among different team members from different countries with different languages and it is often difficult to communicate when communication is not in their native language[14]. Language barriers are common in distributed agile projects because a particular language(often English) is used for

communication between distributed teams. Agile team members whose native language is not English often face language barriers when communication is in English and meeting takes longer time than usual because it is difficult for them to express their views[6][13][14]. According to [15] to remove the language barriers the computer-mediated communication, machine translations can be used which increases the availability of intercultural collaboration beyond the language barrier.

3.1.5 High communication cost
The cost of communication tools/facilities used for communication purposes is quite high. Companies must be able to spent money in order to provide better communication facilities for distributed teams[14].

3.1.6 Lack of Teamwork
In distributed development all the team members should communicate and collaborate with all other team members not only with selected team members in order to produce a better quality product[13]. All the team members should understand the importance of team spirit and contribute to the entire team 's effort. Lack of communication may lead to lower team spirit and productivity. Overall communication is affected when some team members do not contribute to the team's effort[13][14].

3.1.7 Lack of frequent face-to-face communication
According to agile manifesto agile methods considers the role of people communication more valuable than defined processes and tools. In distributed setting the distance between customers and developers may be more which may lead to difficulty in frequent face-to-face communication among them[6][14].

3.1.8 Cross Cultural Diversity
Culture is one of the important issue in communication. An example of cultural issues is when some offshore team members pass only assured information to onshore team and don't pass critical issues[14].Culture issues may impact collaboration and communication if not handled properly.

3.2 Personnel Selection
Selection of appropriate personnel for the team is very necessary. In agile software development requirements keep changing so it become necessary to have cross-functional teams consisting of developers, testers, analysts and project managers. Programmers who used plan driven approach and not able to handle changing requirements will not be a good choice for agile team. Therefore pair programming that is one of the agile practice is used for development and it is essential to select a coding partner with similar mindset.

3.3 Trust
Trust is one the main problem in distributed environment and extremely difficult to achieve because trust needs face-to-face and informal communication and technology cannot replace trust [6]. Some of the challenges of distributed agile teams in establishing trust are:

3.3.1 Feeling of Insecurity
Junior team members or members who are new to the project often feel insecure. They don't open up with other team members and don't put their ideas in team discussion. Due to feeling of insecurity building trust in a team is difficult[10].

3.3.2 Sense of belonging to a team
As team members in distributed agile development are located at different locations so it is difficult to have continuous communication between teams which results in poor team bonding. If team members don't feel belongingness to a team then building trust among team members is very difficult. Socio-cultural differences of distributed team members can also be a reason for building trust within the teams [10].

Table 1. Overview of benefits and challenges of Agile in GSE

Benefits	Challenges
Better Communication	Communication issues:
Better coordination	Time Zone
Increased Trust	Lack of Communication tools
Project visibility and control	Language barriers
Clarification of requirements	Lack of teamwork
Improved Productivity	Different culture
Improved Knowledge sharing	Different project Background
Quick Development	High communication cost
	Lack of frequent face-to-face communication
	Cross Cultural Diversity
	Personnel Selection
	Trust

4. PROPOSED SOLUTION TO LANGUAGE BARRIERS
Language barrier is one of the most critical issue in intercultural collaboration. In this section we have proposed a solution for addressing language barriers in distributed agile development. In distributed development team members are located at different locations and they communicate with each other via communication tools. To increase the intercultural collaboration and usability of language services computer mediated communication or machine translations can be used and Google Translate API can be used for this purpose. Google Translate API allows you to dynamically translate simple text from one language to another by using English as an intermediate language. Google Translate API is an interface that return information about places and which converts the written documents generated by sender into receiver understandable form according to his/ her native language and response to sender is further translate according to sender's native language. Google Translate API provides the language translation services to create a language environment necessary for Intercultural collaboration.

5. CONCLUSION

In conclusion, this paper demonstrates the applicability of agile methods in Global software development. We examined the current research literature to determine the benefits and challenges of agile practices in distributed development. After reviewing the current research on the use of agile practices we can say that there is the possibility of applying agile methods in GSE. Agile software development can be beneficial with respect to Global software engineering and gives a no. of benefits like better coordination and communication, product visibility, trust, customer involvement, early problem identification, clarification of requirements etc. Communication, Language barrier, different culture, lack of trust, personnel selection are the main challenges usually faced in global software engineering and regardless of these challenges the software companies are successfully applying agile methods in their distributed development. The results of this study suggests that implementation of agile methods in distributed environment by means of appropriate tool support can helps in mitigating these challenges. Various software industries are getting benefits in distributed agile development with the usage of computer mediated tools like videoconferencing, emails, Skype, wikis etc. In future work empirical investigation can be done in order to identified more challenges and strategies in Agile global software development. In this paper we had not focused on the mitigation strategies to deal with the challenges that are being faced by distributed teams. Therefore for future work an framework should be developed that contains the information regarding various challenges in a particular Global software development context and mitigation strategies for these challenges.

6. REFERENCES

[1] Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, 85(6), 1213-1221.

[2] Miller, G. G. (2001, July). The Characteristics of Agile Software Processes. In *TOOLS (39)* (pp. 385-387).

[3] Jalali, S., & Wohlin, C. (2010, August). Agile practices in global software engineering-A systematic map. In *Global Software Engineering (ICGSE), 2010 5th IEEE International Conference on* (pp. 45-54). IEEE.

[4] Hossain, E., Babar, M. A., & Paik, H. Y. (2009, July). Using scrum in global software development: a systematic literature review. In *Global Software Engineering, 2009. ICGSE 2009. Fourth IEEE International Conference on* (pp. 175-184). Ieee.

[5] Bose, I. (2008). Lessons Learned from Distributed Agile Software Projects: A Case-Based Analysis. *Communications of the Association for Information Systems*, 23.

[6] Layman, L., Williams, L., Damian, D., & Bures, H. (2006). Essential communication practices for Extreme Programming in a global software development team. *Information and software technology*, 48(9), 781-794.

[7] Yap, M. (2005, July). Follow the sun: distributed extreme programming development. In *Agile Conference, 2005. Proceedings* (pp. 218-224). IEEE.

[8] Taylor, P. S., Greer, D., Sage, P., Coleman, G., McDaid, K., & Keenan, F. (2006, May). Do agile GSD experience reports help the practitioner?. In *Proceedings of the 2006 international workshop on Global software development for the practitioner* (pp. 87-93). ACM.

[9] Gurram, C., & Bandi, S. G. (2012). Teamwork in Distributed Agile Software Development.

[10] Farooq, U., & Farooq, M. U. (2010). *Exploring the Benefits and Challenges of Applying Agile Methods in Offshore Development* (Doctoral dissertation, Master Thesis, Software Engineering, Blekinge Institute of Technology, Karlskrona, Sweden).

[11] Dullemond, K., van Gameren, B., & van Solingen, R. (2009, July). How technological support can enable advantages of agile software development in a GSE setting. In *Global Software Engineering, 2009. ICGSE 2009. Fourth IEEE International Conference on* (pp. 143-152). IEEE.

[12] Dorairaj, S., Noble, J., & Malik, P. (2010). Understanding the importance of trust in distributed Agile projects: A practical perspective. In *Agile Processes in Software Engineering and Extreme Programming* (pp. 172-177). Springer Berlin Heidelberg.

[13] Dorairaj, S., Noble, J., & Malik, P. (2011). Effective communication in distributed Agile software development teams. In *Agile Processes in Software Engineering and Extreme Programming* (pp. 102-116). Springer Berlin Heidelberg.

[14] Kamaruddin, N. K., Arshad, N. H., & Mohamed, A. (2012, April). Chaos issues on communication in Agile Global Software Development. In *Business Engineering and Industrial Applications Colloquium (BEIAC), 2012 IEEE* (pp. 394-398). IEEE.

[15] Nomura, S., Ishida, T., Yasuoka, M., Yamashita, N., & Funakoshi, K. (2003, June). Open source software development with your mother language: Intercultural collaboration experiment 2002. In *International Conference on Human-Computer Interaction (HCI-03)* (Vol. 4, pp. 1163-1167).

[16] Holmström, H., Fitzgerald, B., Ågerfalk, P. J., & Conchúir, E. Ó. (2006). Agile practices reduce distance in global software development. *Information Systems Management*, 23(3), 7-18.