

Challenges to Practice Agile Methods in Global Software Development – A Review of Literature

V. N. Vithana
Management and Science
University, Malaysia

D. Asirvatham
Taylor's University
Malaysia

M. G. M. Johar
Management and Science
University, Malaysia

ABSTRACT

Global Software Development (GSD) is the process of developing software by globally distributed teams and it has become increasingly popular due to the many benefits it offers. Reduced costs, access to a wide range of skilled people and reduction in application development time are the major attractions to GSD.

Agile methods are popular software development methodologies that help companies deliver high quality products rapidly. Though Agile methods were originally developed for co-located teams, it is now being used in GSD to achieve the benefits of Agile in GSD combination.

Agile methods heavily rely on frequent informal communication between all stakeholders of the project. GSD introduces time, cultural and geographical barriers to this requirement. This paper categorises the various challenges in to 9 broad categories and develops a hypothetical model on impact of the challenges on project success.

General Terms

Global Software Development, Agile Methods

Keywords

Challenges, Global Software Development, Issues, Offshore Software Development, Agile methods.

1. INTRODUCTION

Global software development (GSD) is to develop software by utilizing global resources and around the clock development practice [1] [2]. Today, it is in widespread use as it offers many benefits such as reduced cost, improved quality and reduced development time [44].

Software development methods constantly evolve due to new technologies and demands of the users. Organizations continuously adapt new strategies to stay competitive in today's dynamic business environment. Agile software development methods help organizations deliver high quality software rapidly with improved customer satisfaction [3]. Thus, Agile methods have become one of the most widely used methodologies to develop software today [4].

Agile methods were originated by a group of experienced professionals in 2001 based on their years of experience in the industry [5]. Agile methods are light weight and support requirement change even late in the software development life cycle. Agile methods were originally developed for small co-located teams where frequent informal communication exist. Today Agile methods are used in GSD which has introduced several challenges due to geographical distance, time differences, cultural differences and other reasons.

This paper attempts to summarise and categorise the various challenges discussed in the previous literature. The empirical and theoretical evidences stated in the literature were used to

develop a model of challenges to practice agile methods. Following research question would be addressed in this paper.

1. What are the challenges to practice Agile methods in global software development according to the peer reviewed articles in the literature?

Many previous literature which are case studies or industry reports do not categorise the challenges. This review study contributes to the research gap by synthesizing and categorizing challenges into broad themes and developing a conceptual model on impact of challenges on project success. Findings of the study would also be beneficial to the industry professionals to improve their practices to successfully complete projects.

The rest of this paper is organized as follows. Literature review section discusses the challenges to practice agile methods in GSD. Methodology section discusses the methodology followed to conduct the study; Results and Discussion section synthesizes the challenges and discusses the theoretical model developed based on the literature review. Finally conclusion sections concludes the paper.

2. LITERATURE REVIEW

Agile methods have become the standard of developing software today [3]. Under the umbrella Agile, there is a collection of methodologies such as Scrum, Kanban, XP and Lean Development to name a few. Agile philosophy emphasis the value of individuals and interactions, working software, customer collaboration and responding to change [5]. These methodologies are rooted in adaptive planning, early delivery and continuous improvement. Therefore frequent informal communication is a necessity to practice Agile methods efficiently. Heavy documentation is not practiced as requirements change is supported even late in the development cycle [4]. Team need to be self-organizing as they need to be autonomous in taking quick decisions. Thus practicing Agile in co-located teams is itself a challenge. In Global software development, (GSD) when teams are from different cultures, working in different time zones from remote locations, work becomes even more challenging. Previous literature states such identified challenges as discussed below.

Communication

Software development is a collaborative effort of a team which involves intensive knowledge sharing. Effective communication is very important in software development especially using agile methods [6]. Communication is the most listed challenge in Agile in GSD combination [7]. Geographical distance and different time zones hinder synchronous communication. GSD heavily relies on technology for communication [8] [9] [10]. Absence of required communication media or the improper use of media creates communication gaps within the team [11] [12] [13]. Communication with geographically distributed team

members often take place via asynchronous communication methods. Effective communication might be difficult to achieve with asynchronous methods. Email, chat, teleconferencing, video conferencing are the commonly used tools for communication [7]. In some instances video conferencing was used only at critical stages of the project as it was the most expensive.

Cultural differences, language barriers and attitude are other factors that hinder effective communication among project stakeholders [7] [14] [15] [16].

Though there are well developed tools used for communication, it is still a challenge to achieve 100% effective communication as in face to face communication [16]. Inefficient communication among project stakeholders is considered as the main reason for project failure.

Cultural Differences

GSD comprises of virtual teams comprised of different cultural backgrounds. Cultural differences hinder mutual understanding among team members [15] [13]. Some personal prefer direct and open communication as a result of their culture. Some cultures are opposed to that [17]. Being autonomous and taking rapid decisions is important in practicing Agile. Some cultures lack autonomy. Cultural differences affect the team bond and lead to communication gaps [18]. This adversely affects project success [13].

Team Spirit

The success of a software project depends on the project human resources and their relationships [19] [13] [14]. This is very significant in agile methods as agile philosophy emphasis the value of individuals and interactions more than following procedures [5]. Team need to be self-organizing to adopt to changing requirements [3]. Thus a good understanding and team bond is crucial for project performance [14].

Geographically distributed teams may not share the same team spirit. Distance and cultural differences create a ‘They’ versus ‘Us’ feeling among the onsite and the offshore team [14] [20].

Technical Competency

Agile methods require technically competent personal with an Agile mind set. Members with a mind set for plan driven approach will not be suitable in this environment [3]. In a self-organizing team members should be skilled and matured enough to adapt and work efficiently. Team members need to be autonomous rather than waiting to be given orders by the project manager. Hence selecting the right team members with skill and attitude is considered as a challenge [18] [19].

According to the literature different companies used different strategies in selecting the personnel for projects. Many successful scenarios emphasized the importance of having a good team lead who can coordinate the various activities. According to the literature some companies chose to have well experienced, creative people who have worked with each other before to be in the team. Some companies chose to have self-organizing teams who could manage their work responsibilities with minimum supervision [13][21].

Collaborative Development

Agile methods deliver high quality working software at the end of an iteration. Thus, there are some important practices such as continuous code integration, test driven development (TDD) and test automation that should be adopted to ensure project success [23].

Software is developed by several members collaboratively. Therefore building the code to a common repository at the end of the day is important and necessary to ensure that your code is in compliance with other modules. This practice will help the whole team to see the progress of the project and obtain shared vision to progress the work. It is recommended for developers to build the code and wait for the successful message from the system before leaving office [8]. Building this discipline in developers is considered as a challenge.

Access to required systems and availability of required bandwidth is essential to build code regularly [24].

As development is conducted in several iterations, testing start early and in parallel to the development activities.

According to previous research it is difficult to complete the testing process within an iteration [24] [25]. Some companies find it difficult to complete regression testing within the sprint. Being unable to implement test automation, not having clearly defined acceptance test criteria, lack of developer involvement in testing are some of the major causes for this delay.

There are no distinct roles such as Quality Assurance (QA), project manager (PM) or Developer in agile methods. Maintaining software quality would be the whole team’s responsibility. It is important that entire team contribute in definition and execution of acceptance tests. Building this mind set in the team is considered a challenge [24] [19] [26].

In order to complete testing within a sprint developer contribution to testing plays an important role. Testing should start in parallel to development. Developers should practice Test Driven Development (TDD) and should conduct unit tests to test the functionality. This will save time for testers to focus on acceptance tests and regression tests [24, 26, 27, 28]. Developers need to build and deploy their code early in order for testers to start, which otherwise delay testing process [19, 24, 29]. Requirements should be clearly defined to define the acceptance criteria of tests. This could be challenging specially in offshore software development, in the absence of continuous customer engagement [12] [19] [17]. At the same time, testing personnel should be technically competent to help the developers with integration testing and API testing [24] [30]. Implementing test automation is essential to complete testing on time [21] [19] [26].

Technology

Usage of correct tools and infrastructure has a huge impact on the project success [13]. Unavailability of the correct tools and infrastructure or improper handling of the tools will impede the project success [8] [32]. Branches of large companies will have all the facilities where as small and medium organizations might not be able to afford the cost.

Continuous Customer Engagement

Agile manifesto emphasis the importance of customer collaboration over contract negotiation. One of the principles of Agile manifesto is to satisfy the customer through early and continuous delivery of valuable software (Agile manifesto, 2001). Thus, customer is not only required to be present onsite with the development team, he should also be actively engaged throughout the life cycle of the project. He should actively provide input to define requirements and improve the quality of the software product. Absence of continuous customer engagement will lead to incomplete requirements which would result in poor quality software. Hence, absence of continuous customer engagement is identified as a

challenge to achieve project success. In previous research studies, this is also identified as customer commitment [31].

In offshore software development, when the development team is offshore and customer is onshore, it might be impossible to achieve continuous customer engagement throughout the project. Even when the customer attends the meeting, most of the time, it will be through video conferencing. The communicating via video conferencing does not give the same impact as a face to face, co-located meeting.

Coordination and Control

Agile teams are said to be self-managed. Team members should understand each other well [7]. Team should possess the necessary skills to manage a situation on their own [23] without much guidance from the external entities. Team members should be given lot of autonomy from senior management to take decisions on important project matters. Obtaining support from senior management and organizational culture is considered as a challenge [37]

The distributed team can lead to problems in progress estimation. The project status is less visible and controllable. More teams have to be coordinated, often from a distance. Therefore offshoring projects are riskier and project management and controlling is more difficult.

Knowledge Management

Software development is knowledge intensive work, where employee knowledge is critical for project success. Documentation plays an important role in distributed agile more than in the co-located teams. Agile philosophy emphasis on 'just enough documentation'. But with the absence of proper documentation certain knowledge might not be conveyed efficient to all stakeholders [32]. Various tools such as wikis, code repositories are used for knowledge management. It is important that proper strategies and tools are used for efficient knowledge management in order to effectively utilize tacit knowledge of experienced programmers. [15]

Training and Mentoring

Agile methods work best with skilled and experienced professionals [8] [36]. Organizations should provide the necessary training to make sure that team members have the right knowledge and skill. Senior members should be willing to mentor the junior members. It is equally important that junior members have the attitude to learn from others and continuously improve their skills and knowledge. When teams comprises of lot of junior members with less experience in Agile methods, it would adversely affect the project success [34].

Agile Project Success

From a project management perspective, a successful project should meet the original planned cost, schedule, quality, and scope by the efficient use of resources [37]. Customer satisfaction is an important factor in Agile projects [38]. Agile methods emphasis continuous improvement for improved customer satisfaction. Recently, improving the processes of the organization is also considered as a factor to measure project success.

3. METHODOLOGY

This research aims to determine the challenges to practice Agile methods in GSD through a thorough literature review. The scope of this study included all Agile methods in all types of distributed development. We only searched for papers that are written in English and available online. We obtained

papers from electronic databases such as IEEEExplore (www.ieeexplore.ieee.org/Xplore/), ACM Digital library (www.portal.acm.org/dl.cfm), Google Scholar (<http://scholar.google.com>), Elsevier Science Direct (www.sciencedirect.com/) and AIS eLibrary (www.aisel.aisnet.org/).

Search strings were formulated by combining challenges and different Agile practices and different types of team distributions. It can be summarized as: (X1 OR X2 ... OR Xn) AND (Y1 OR Y2 ... OR Yn) AND (Z1 OR Z2 ... OR Zn), where X denotes synonyms of challenges and Y

covers Agile methods and Z includes different alternatives of GSD as presented below.

X: {Challenges, Issues, Barriers}

Y: {Agile, Scrum, Extreme Programming, XP}

Z: {offshore software development, global software engineering, global software development, distributed software development, GSE, GSD}.

The titles, abstracts, and keywords of the articles were searched for key words. It should be noted that only peer-reviewed publications were taken into consideration and gray literature has not been explored. Studies were excluded if their focus was not agile software development in Global software development or if they did not present empirical data.

4. RESULTS AND DISCUSSION

After synthesizing the results of literature review, we identified 9 challenges based on the frequency listed in Table 1.

Challenges in Communication

Communication is the most listed challenge in the literature [7]. Various conditions and factors such as listed below, contribute to communication gaps.

Time difference – Differences in working times result in asynchronous communication methods such as email. This would introduce delays in information exchange. At the same time, important information might not be communicated immediately due to large time gaps.

Language barriers – When English is not the native language, it would introduce gaps in communicated information. Team members might be reluctant to communicate openly as they lack confidence. Information could be wrongly interpreted.

Cultural barriers – Some cultures do not promote open and direct communication. This would adversely affect proper information exchange.

Technology – Network bandwidth, required tools and systems should be in place for proper information exchange. Absence of technology would introduce information gaps between team members.

Geographical distance – Virtual team members might not have physically met and might lack healthy relationships among them. As a result, there could be less trust and open communication among team members.

Challenges in Technical Competency

Technical competency is considered an important factor in Agile projects [6]. Project work is carried out with minimum guidance as teams are 'Self organizing'. Team members are provided with lot of autonomy. Thus Agile projects advocate

experienced, skilled people. Thus selecting a team with right skill is considered a challenge.

Challenges in Collaborative Development

Agile methods use various techniques such as continuous integration, test driven development, test automation which plays an important role in project success. Team members should be equipped with proper skills and attitude to follow these techniques. Organization should also facilitate the team with the proper technology and infrastructure.

Challenges in Customer Engagement

Agile methods are iterative and incremental methods. In each iteration, customer involvement plays a major role to proceed with the next iteration. Thus customer should be well know about Agile requirements, well committed to project and should effectively contribute to the project with ideas. Lack of these factors would result in project failure.

Challenges in Team Coherence

Software is developed as a team effort. Team should be responsible for the failure or success of project. Thus it is important that team members trust each other and have strong and healthy relationships to ensure the project success.

Challenges in Knowledge Management

Software development as a knowledge intensive process, need to store and manage the knowledge efficiently. Proper

templates and systems should be in place to ensure information consistency. Though Agile methods promote ‘less documentation’ team members should have the mind set to document and update the template information as required.

Challenges in Training and Mentoring

Members should be continuously trained to keep updated on Agile methods and various techniques and tools such as test automation. Organizational support plays a vital role in providing the required training. Senior members should mentor and support the junior members to keep their work morale.

Challenges in Coordination and Control

Agile team are autonomous. They need to take decision quickly. Organizations should support autonomy and decision making by team members. It is a known fact that employee turnover is high especially in software companies. Organizations should take necessary steps to keep the skilled experienced members with the organization. Lack of these would lead to project failure.

Challenges in Technology

Organizations should provide with required network bandwidth, tools and systems for smooth conduct of project work. Lack of tools and systems would hinder effective communication and delays in project work.

Table 1 – Challenges as listed in the research papers

Challenge	References	Number of Times
Communication	[6,7,8,9,10, 11, 12, 13, 14, 15, 26, 29, 33, 42]	14
Team coherence	[11, 16, 17, 19, 29]	5
Technical Competency	[7, 11, 22, 24, 29]	5
Continuous Integration	[7, 21, 22, 23, 28]	5
Testing in Distributed Agile	[8, 20, 22, 23, 28]	5
Technology	[11, 21, 22, 23, 28]	5
Customer engagement	[29, 39, 40, 42]	4
Coordination and Control	[11, 21, 24, 35, 42]	5
Training and Mentoring	[8, 11, 22, 24, 42]	5
Knowledge Management	[8, 9, 11, 29, 30, 43]	6

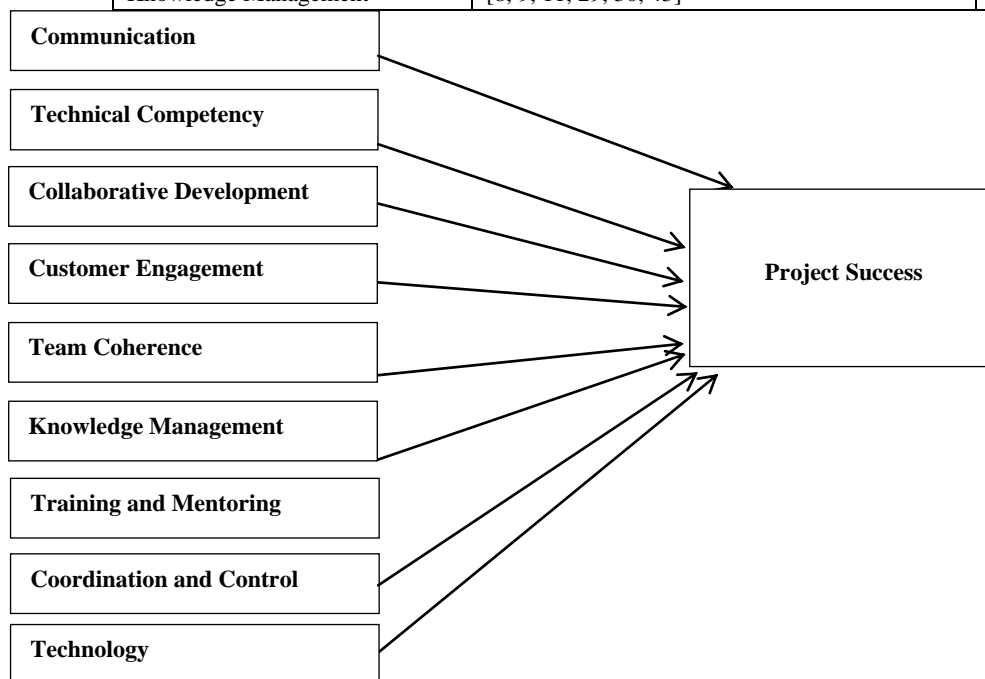


Figure 1 - Conceptual Framework

5. CONCLUSION

Global Software Development (GSD) is the process of developing software by globally distributed teams and it has become increasingly popular due to the many benefits it offers. Agile methods are popular software development methodologies that help companies deliver high quality products rapidly. When Agile methods are used in GSD several challenges are introduced due to geographical, time and cultural differences among team members. This study categorized the various challenges in to 9 broad themes such as Communication, Team Coherence, Technology, Customer Engagement, Collaborative Development, Training and Mentoring, knowledge Management, Coordination and Control and Technical Competency.

A conceptual framework was developed to measure the impact of challenges on project success. In the future studies, this framework could be statistically tested to evaluate the significance of challenges on project success

6. REFERENCES

- [1] S. Jalali and C. Wohlin, "Global Software Engineering and Agile Practices: A Systematic Review," *Journal of Software: Evolution and Process*, 2011.
- [2] R. Sriram and S. K. Mathew, "Global software development using agile methodologies: A review of literature," in *Management of Innovation and Technology (ICMIT)*, Sanur Bali, Indonesia, 2012.
- [3] M. Cohn, *Succeeding with agile: software development using Scrum.*, Upper Saddle River, NJ: Addison-Wesley, 2010.
- [4] W. Cunningham, 2001. [Online]. Available: <http://www.agilemanifesto.org/>.
- [5] A. Cockburn, *Agile Software Development*, Boston, USA: Addison-Wesley Longman Publishing Co, 2002.
- [6] Y. I. Alzoubi and A. Q. Gill, "Agile Global Software Development Communication Challenges: A Systematic Review," in *Pacific Asia Conference on Information Systems*, 2014.
- [7] M. Fowler, "Using an Agile Software Process with Offshore Development," 2006. [Online]. Available: <http://www.martinfowler.com/articles/agileOffshore.html>.
- [8] M. Paasivaara, S. Durasiewicz and C. Lasse, "Distributed Agile Development: Using Scrum in a Large Project," 2008.
- [9] M. Paasivaara, S. Durasiewicz and C. Lassenius, "Using Scrum in Distributed Agile Development: A Multiple Case Study," in *Proceedings of the 2009 Fourth IEEE International Conference on Global Software Engineering*, 2009.
- [10] B. Ramesh, L. Cao and K. Mohan, "Can Distributed Software Development be Agile?," *Communications of the ACM*, vol. 49, no. 10, October 2006.
- [11] I. Bose, "Lessons Learned from Distributed Agile Software Projects: A Case-Based Analysis," *Communications of the Association for Information Systems*, vol. 23, no. 1, pp. 619-632, 2008.
- [12] A. Marambe and C. Jayasundara, "The Challenges of Offshore Agile Software Development in Sri Lanka and Effects on the Project Outcome," *International Scientific Journal of Management Information System*, vol. 9, no. 3, pp. 10-20, 2014.
- [13] H. Kaur, H. M. Haddad and J. (Selena) He, "Distributed Agile Development: A Survey of Challenges and Solutions," in *International Conference on Software Engineering Research and Practice*, 2015.
- [14] S. Dorairaj, J. Noble and P. Malik, "Effective Communication in Distributed Agile Software Development Teams," *Springer-Verlag Berlin Heidelberg*, 2011.
- [15] E. Bjarnason, K. Wnuk and B. Regn, "Requirements Are Slipping Through the Gaps -A Case Study on Causes & Effects of Communication Gaps in Large-Scale Software Development," *IEEE 19th International Requirements Engineering Conference*, pp. 37-46, 2011.
- [16] H. D. Yaggahavita, "Challenges in Applying Scrum Methodology on Culturally Distributed Teams," *Sheffield Hallam University (SHU)*, UK, 2011.
- [17] S. Jalali, C. Gencel and D. Smite, "Trust Dynamics in Global Software Engineering," in *Empirical Software Engineering and Measurement (ESEM)*, Bolzano, 2010.
- [18] K. Schwalbe, *Information Technology Project Management*, United States: Cengage Learning, 2011.
- [19] V. Lalsing, S. Kishnah and Sameerchand , "People Factors in Agile software development and Project Management," *International Journal of Software Engineering & Applications (IJSEA)*, vol. 3, no. 1, pp. 117-137, 2012.
- [20] R. Bavani, "Distributed Agile, Agile Testing and Technical Debt," *IEEE Software* , vol. 29, no. 6, pp. 28-33, 2012.
- [21] E. Collins, G. Macedo, N. Maia and A. Dias-Neto, "An Industrial Experience on the Application of Distributed Testing in an Agile Software Development Environment," in *IEEE Seventh International Conference on Global Software Engineering*, 2012.
- [22] M. Lindvall, V. Basili, B. Boehm, P. Costa, K. Dangle, F. Shull, R. Tesoriero, L. Williams and M. Zekowitz, "Empirical Findings in Agile Methods," in *Proceedings of Extreme Programming and Agile Methods - XP/Agile Universe 2002*, 2002.
- [23] M. Sumrell, "From Waterfall to Agile – How does a QA Team Transition?," in *IEEE*, 2007.
- [24] R. Phalnikar , V. Deshpande and S. Joshi , "Applying Agile Principles for Distributed Software Development," in *International Conference on Advanced Computer Control*, New York, 2009.
- [25] S. V. Shrivastava and U. Rathod, "Risks in Distributed Agile Development: A Review," *Science Direct*, vol. 133, pp. 417-424, 2014.
- [26] C. Sepulveda, "Agile development and remote teams: learning to love the phone," in *Proceedings of the Conference on Agile Development*, Washington DC, 2003.
- [27] C. Gurram and S. G. Bandi, "Teamwork in Distributed Agile Software Development, Masters Thesis," *Blekinge Institute of Technology, Karlskrona*, September 2012.

- [28] D. Talby, A. Keren, O. Hazzan and Y. Dubinsky, "Agile Software Testing in a Large-Scale Project," *Agile Software Development: The Business of Innovation*, by Jim Highsmith and Alistair Cockburn; IEEE Computer: September 2001 (Vol. 31, No. 9), pp. 120-12, vol. 23, no. 4, pp. 30-37, 2006.
- [29] S. C. Misra, V. Kumar and U. Kumar, "Identifying some important success factors in adopting agile software development practices," *Journal of Systems and Software*, vol. 82, no. 11, pp. 1869-1890, 2009.
- [30] T. Dingsøy and D. Smite, "Managing Knowledge in Global Software Development Projects," *IEEE Computer Society*, IT Pro, pp. 22-29, 2014.
- [31] M. M. S. Arefin and D. Korzun, "Improvement of "the new agile process for distributed projects," Master Thesis," Chalmers University of Technology, Göteborg, 2010.
- [32] K. Conboy, S. Coyle, X. Wang and M. Pikkarainen, "People over Process: Key Challenges in Agile Development," *IEEE Software*, vol. 28, no. 4, pp. 48-57, 2011.
- [33] T. Jaanu, M. Paasivaara and C. Lasseni, "Effects of Four Distances on Communication Processes in Global Software Projects," in *ESEM '12: Proceedings of the ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, Lund, Sweden, 2012.
- [34] D. A. Vogel and J. E. Connolly, "Best Practices for Dealing with Offshore Software Development," *Hand book of Business Strategy*, 2005.
- [35] M. Špundak, "Mixed Agile/Traditional Project Management Methodology–Reality or Illusion?," *Procedia-Social and Behavioral Sciences*, vol. 119, p. 939–948, 2014.
- [36] S. Sharma, D. Sarkar and D. Gupta, "Agile Processes and Methodologies: A Conceptual Study.," *International Journal on Computer Science and Engineering*, 2012.
- [37] U. Sekaran and R. Bougie, *Research Methods for Business-A Skill Building Approach*, New Delhi: John Wiley & Sons, Ltd, 2010.
- [38] K. Schwaber and J. Sutherland, "The Scrum Guide," 2013. [Online]. Available: <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf#zoom=100>.
- [39] B. Ramesh, R. Baskerville and L. Cao, "Agile requirements engineering practices and challenges: An empirical study," *Information Systems Journal*, vol. 20, no. 5, pp. 449-480, 2010.
- [40] I. Inayat, S. S. Salim, S. Marczak, M. Daneva and S. Shamshirband, "A systematic Literature review on agile requirements engineering practices and challenges," *Computers in Human Behavior*, 2014.
- [41] J. D. Herbsleb and A. Mockus, "An Empirical Study of Speed and Communication in Globally Distributed Software Development," *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING*, vol. 29, no. 6, pp. 481-494, June 2003.
- [42] R. Bavani, "Critical Success Factors in Distributed Agile for Outsourced Product Development," in *International Conference on Software Engineering*, Chennai, India, 2009.
- [43] V. N. Vithana, S. G. S. Fernando and M. Kapurubandara, "Success Factors for Agile software Development- A Case Study from Sri Lanka," *International Journal of Computer Applications*, vol. 113, no. 17, March 2015.
- [44] P. F. B. H. H. a. Ó. C. E. Ågerfalk, "Benefits of Global Software Development: The Known and Unknown," in *International Conference on Software Process*, Berlin Heidelberg, 2008.