

Issues in Adopting Agile Methodologies in Global and Local Software Development: A Systematic Literature Review Protocol with Preliminary Results

Fawad Ghafoor
Department of Computer
Sciences and Software
Engineering,
University of Engineering and
Technology, KPK, Pakistan

Ibrar Ali Shah, PhD
Department of Computer
Software Engineering,
University of Engineering and
Technology, KPK, Pakistan

Nasir Rashid
Department of Computer
Science and IT,
University of Malakand, KPK,
Pakistan

ABSTRACT

Software Development companies are adopting agile methodologies in globally software development (GSD) as well as local software development (LSD). Beside the extensive benefits that agile methods offer to GSD and LSD, it also possesses a number of critical issues to both environments during the whole course of software development life cycle (SDLC). This paper presents our contribution in constructing a systematic literature review (SLR) protocol for exploring the critical issues in GSD and LSD environment. The desired outcome of the SLR protocol will be a group of critical risk factors/issues and their concerned practices that will be helpful for vendors to avoid these issues while adopting agile principles in GSD and LSD. This paper also presents the preliminary results of our SLR protocol.

Keywords

Agile Global Software Development (AGSD), Global Software Engineering (GSE), Agile Local Software Development (AGSD), Systematic Literature Review (SLR).

1. INTRODUCTION

Agile Software Development (ASD) means gathering of software development methodologies supported on iterative development, where requirements and solutions develop through cooperation between self-managing cross-working groups [1]. The point is to make faster and lighter development processes, fit them more receptive to change. Agile Methodologies have been steered as the programming methodologies of decision for the rapid, unstable universe of Internet and web programming development [2]. Agile methodologies favor programming development over documentation. It trusts in giving numerous versions of the software product in short iterations, and after that redesigns the product as per the customer's response. This methodology helps in beating the issues like incessant changes, quick development and client fulfillment [1].

The term Global Software Development (GSD) suggests groups of programmers from different parts of the globe making financially usable software. In last decade, looking for lower costs and access to talented assets, numerous associations started to research with remotely found programming facilities with outsourcing [3]. Dispersion of software development is turning out to be increasingly regular to save the generation cost and decrease an ideal opportunity to advertise. In GSD, partners from various national and hierarchical societies are included in creating software and the

many advantages incorporate access to a vast work pool, cost advantage and 24 hours development [1].

Software engineers and project managers are attempting to evaluate the suitability of agile processes for software development [4]. Prosperous agile acceptance prompts to creating higher quality software, improve developers moral at a lower cost than the conventional waterfall model approach. Using agile practices for distributed development is obtaining strength in different associations to raise the quality and execution of the project. However, agile adoption always has extraordinary challenges and so, crucial hierarchical changes are important for fruitful result.

Following research questions are answered in this paper.

Research Question 1: What are the issues as described in the literature in adopting agile methods in Global Software Development (GSD)?

Research Question 2: What are the issues as described in the literature in adopting agile methods in Local Software Development (LSD)?

Research Question 3: Is there any difference between the issues in global software development and the local software development?

2. BACKGROUND

According to [5], the authors highlighted challenges with Agile (scrum) and gives an insight to the user whether the agile is the silver bullet? Is agile better than traditional waterfall model? Will agile work effectively with distributed teams? However, the paper come out with a fair comparison of the models like waterfall and Agile Scrum and makes the users aware of the characteristics of each, in order to enable them to match the same with their experience.

Kamaruddin et al [6], the authors discussed the issues that have occurred related to the communication in Agile Global Software Development (AGSD). Literature survey has been carried out to gather information from the previous works as well as from forum feedback by practitioners globally. The chaos issues are then illustrated using conceptual model, tabulated based on authors and then discussed. Literature survey has identified 13 chaos issues related to communication and whereby different culture and lack of frequent face-to-face contact are the main issues that have been discussed in the literature as well by the practitioners.

Helena et al [7] have, the authors presented some discoveries from a case study in which they investigate the specific issues related with handling GSD.

Florin et al [8] have, investigated the issues of incorporating the two methodologies ASD and GSD in view of their clashing attributes with the aim to characterize a procedure for organizations that desire to follow AGSD. The technique proposed is established on gathering the different methods proposed by the literature in three classes: preliminary measures, communications technology tools, fitting agile methods. To be effective with AGSD, it is important to combine the different arrangements in three classifications: preliminary, technological and agile practice fitting. The organization which aims to follow agile in global projects needs to begin with usage of preliminary measures, trailed by technological ones, and simply after that applying the fitting measures.

Layman et al [9], have conducted an industrial case study of a dispersed group in USA (project management) and the Czech Republic (Development Team) that utilized extreme programming. The objective was to see how this globally dispersed group made an effective project in another domain utilizing an approach that is pendent on informal, face to face communication. They gathered quantitative and subjective information and utilized grounded hypothesis to recognize four key elements for communication in universally dispersed XP groups working inside another challenging domain. On the basis of systematic literatures, the authors propose four parameters for internationally dispersed XP projects working in recent issue area. These include the significant role of the development manager, who commit as communication medium between two groups and played the promoter of both gatherings regularly. They take note of the significance of short, asynchronous communication circles that can serve as an adequate alternate for synchronous communication. A distinctive customer potency is important for managing requirements related issues, and customer's high process clearness is necessary for guidance of developers while they are working on new and strange problems. While some of these guesses allude to practices that could be gainful to any product development team, the outcomes show these practices are fundamental for dispersed teams utilizing a communication rich strategy.

Hossain et al [10] have, systematically reviewed the literature about scrum in GSD. They determine numerous issues of using scrum in GSD. Through Systematic Literature Review (SLR) they evaluated different findings and make the conclusion:

- There is a developing interest and literature needs more observational review to comprehend the utilization of scrum practices in globally dispersed projects,
 - The utilization of scrum practices might be limited by different GSD factors,
 - Geographically dispersed scrum's teams usually face various issues as project dispersion consequence on communication, coordination and collaboration,
 - All inclusive appropriated scrum groups for the most part face various difficulties as venture dispersion effect on correspondence, coordination and joint effort forms,
- Scum practices should be extended or altered to support globally dispersed software development teams.

Korkala et al [11] have, performed two distributed agile software development case studies and did comparison of findings with existing suggestions about communication in distributed agile teams. The findings and existing literature generalize that exhibited suggestions are beneficial considering in dispersed agile development, however with some alert. The observationally based findings demonstrate that the role of distinct customer is the key suggestion. The absence of an distinct customer ready to meet duties, and in addition unstable and inefficient communication, can bring serious issues even in little scale distributed agile software development projects. The objective of paper was to analyze these suggestions, and rise the knowledge about their utilization based on current literature and experimental findings from two distinctive little scale distributed agile development intentions with various levels of dispersion.

Jalali et al [12] have, displayed the outcomes of systematically reviewing existing literatures on the utilization of agile practices and incline software development in global software engineering (GSE). The main role is to highlight under which conditions they have been applied proficiently. Some basic terms corresponding with agile practices (e.g. scrum, XP) were considered in theorizing the search strings, alongside with various substitute for GSE, for example, off-shoring, outsourcing, and virtual teams.

The existing literature on the utilization of various agile practices in GSE was outlined in the literature review. Furthermore, the effective empirical reviews were investigated to research under which conditions they have been proficiently practices in software organizations.

Akbar et al [13] have, proposed agile methods for distributed development environment "The DAD Model" targeted on the problems that arise when client is offshore. Agile focuses on frequent synchronous communication where this is not possible in GSD. The proposed model is here to diminish the communication gap amongst customers and developers.

3. RESEARCH METHODOLOGY

Systematic Literature Review (SLR) is a kind of literature review that gathers and precariously evaluates various research studies or papers We have adopted Systematic Literature Review (SLR) [14, 15, 16] a research methodology because SLR is specified and efficient methods of distinguishing, evaluating, and analyzing published papers in order to explore a particular research question. Systematic Literature Review (SLR) is different from ordinary literature review in being properly planned and efficiently executed. In finding, assessing, and outlining all accessible proof on a particular research question, a systematic literature review (SLR) may give a more noteworthy level of validity in its findings than may be conceivable in any of the reviews studied in the systematic literature review (SLR). A systematic literature review (SLR) protocol is composed to depict the plan for the review.

The main steps in our methods are:

- Construct search query, then determine the search strategy, then perform the search for required data.
- Perform the study selection process.
- Apply study quality assessment.

- Extract data and then synthesize that data.

4. SEARCH STRATEGY

In this section we show the strategy used for searching of primary studies. The search strategies used to recognize and gather all the literature satisfies with the inclusion and exclusion criteria.

4.1. Resources To Be Searched

We have searched the following search venues for identifying the required papers.

- Google Scholar: <http://scholar.google.com/>
- Springerlink: <http://www.springer.com/in/>
- ACM: <https://www.acm.org/>
- IEEEExplore: <http://ieeexplore.ieee.org/>
- Science Direct: <http://www.sciencedirect.com/>

4.2. Search Strings

The following search strings have been run on each of the above search engines.

1. ((((((Agile extreme Programming) OR Agile Scrum) OR Agile Software Development) OR Agile development) OR Agile process) OR agile engineering practices)
2. ((Issues and challenges of agile in Global Software Development) OR Issues and challenges of agile in Local Software Development)
3. Comparison of agile LSD and GSD issues
4. (((Face to face communication issues in Global Software Development) OR Pair programming issues in Global Software Development) OR Dynamic requirements issues in Global Software Development) OR Less documentation issues in Agile Global Software Development)
5. (((Face to face communication issues in Agile Local Software Development) OR Pair programming issues in Agile Local Software Development) OR Dynamic requirements issues in Agile Local Software Development) OR Less documentation issues in Agile Local Software Development)
6. ((SCRUM in Agile Global Software Development) OR SCRUM in Agile Local Software Development)
7. ((Issues and Challenges of SCRUM in Agile Global Software Development) OR Issues and Challenges of SCRUM in Agile Local Software Development)

5. SELECTION CRITERIA

This section describes the inclusion and exclusion criteria that are used to confirm that only relevant papers are selected.

5.1. Inclusion Criteria

The following inclusion criteria are used to determine relevant and most suitable papers for data extraction.

1. Papers written in English language will be considered only.
2. Papers contain issues of agile methodologies in GSD will be considered only.

3. Papers contain issues of agile methodologies in LSD will be considered only.
4. Papers contain challenges of agile methodologies in GSD will be considered only.
5. Papers contain challenges of agile methodologies in LSD will be considered only.
6. Papers contain comparison of ALSD and AGSD will be considered only.

5.2. Exclusion Criteria

The following exclusion criteria are used to determine which papers will be excluded:

- a) Papers those are not relevant to research questions.
- b) Papers that do not describe Agile methodologies are excluded.
- c) Papers that do not describe Agile in context of Local Software Development are excluded.
- d) Papers that do not describe Agile in context of Global Software Development are excluded.

5.3. Selection Process

The selection process is divided into two phases:

1. Primary sources are selected based on the title and abstract of papers. Non relevant papers are excluded in this phase. Based on this phase, about 115 papers are selected.
2. Selected sources are then reviewed carefully according to inclusion and exclusion criteria to extract more relevant papers in correspondence with research questions. Based on this phase 51 papers are selected for final data extractions.

6. DATA EXTRACTIONS

In this phase the required data as shown in Table 1 in order to answer the research questions.

Table 1: Data Extraction Form

S. No	Data to be extracted
1	Date of Review
2	Title of the Paper
3	Country of the Author
4	Study Strategy
5	Agile Methodology
6	Company Scope (Local, Global)
7	Challenges of Agile Methods in GSD
8	Challenges of Agile Methods in LSD

7. DATA SYNTHESIS

In this process the identified issues/challenges from finally selected papers are combined to groups. The issues in GSD using the agile methods are shown in Table 2, while the issues confronted in LSD using the agile methods are shown in Table 3. In both tables i.e. Table 2 and Table 3, frequency represents the number of occurrences of a particular issue/challenge in final selected papers.

Table 2: Data Synthesis of “Agile Global Software Development Issues/Challenges”.

S. No	Issues/Challenges	Paper Id	Frequency	Percentage N = 51
1	Geographical Distances	S1,S12, S15, 19, S21,S26, S30,S41, S51	9	18 %
2	Socio-cultural Distances	S4,S5,S, S12,S13, S15,S28, S36,S37, S41,S44, S45,S55, S59	14	28 %
3	Temporal Distances	S4,S12,S13, S15, S20,S30,S36,S37,S40,S44,S57, S58	12	24 %
4	Language Differences	S1,S6,S, S8,S12,S13, S14, S26,S44, S48, S51	11	22 %
5	Ineffective Communication	S1,S5,S, S9, S13, S21,S25, S26,S28,S30,S35,S36,S37, S40,S44, S45,S48,S51,S53, S55,S57, S59	22	43 %
6	Insufficient Documentation	S5, S14, S20, 22, S44,S48,	6	12 %
7	Lack of Trust Among the Agile Teams	S6,S13,S25,S37,S40, S44, S52, S57	8	16 %
8	Lack of Motivated Agile Team	S9,S13,S14, S26, S36,S45,S51,S58, S59	9	18 %
9	Lack of Customer’s Involvement	S13,S14,S22, S53	4	8 %
10	Lack of Requirements Understanding	S6, S13, S19	3	6 %

11	Insufficient Infrastructure for Software Development	S7,S8,S3, S44	4	8 %
12	Architectural Challenges	S1,S6,S2, S51	4	8 %
13	Management Challenges	S1,S8,S2, S26,S28, S30,S37,S44, S49, S51	10	20 %
14	Lack of Appropriate Technology	S1,S14, 19, S26, S36,S49, S58	7	14 %
15	Data Privacy Issues	S5, S30, S44	3	6 %
16	Lack of Agile Training	S20,S35	2	4 %
17	Invisible Project Goal	S5,S55	2	4 %
18	Limited Support for Distributed Development	S2, S3, S11, S17, S56	5	10 %
19	Software Integration Issues	S3, S11, S42,S46	4	8 %

Table 3: Data Synthesis of “Agile Local Software Development Issues/Challenges”.

S. No	Issues/Challenges	Paper Id	Frequency	Percentage N = 51
1	Insufficient Systems’ Documentation	S2,S13, S17,S47	4	8 %
2	Lack of Formal Communication	S2,S3,S24	3	6 %
3	Degraded Software Quality	S3,S10, S11, S42	4	8 %
4	Resistance to Agile Adoption	S11, S47, S54	3	6 %
5	Frequent Changes in Requirements	S11,S16, S17	3	6 %
6	Lack of Team Collaborations	S38,S46	2	4 %
7	Agile Implementation Issue	S11,S17, S32	3	6 %

8	Lack of customers' Presence	S2, S16, S24	3	6 %
9	Lack of Business Knowledge	S4, S32, S54	3	6 %
10	Lack of effective Project Plan	S24, S42	2	4 %
11	Lack of Technical Expertise	S4, S47	2	4 %
12	Lack of Proper Agile Development Training	S3, S4	2	4 %

8. CONCLUSION

This paper presents a systematic literature review protocol for identifying the issues and challenges in adopting the agile methodologies in GSD and LSD. This paper also presents the preliminary result (issues/challenges of agile methods in GSD and LSD) of our SLR protocol. Agile methodologies are applied both in Local Software companies and Global Software companies. However there are a lot of issues and challenges when applied it in these companies. We intend to highlight these issues and challenges by executing the SLR protocol. While applying the designed search string in different search venues, we found N=115 papers out of which N=51 are selected for final extraction of the required data based on our predefined selection criteria. We have almost finalized the data extraction of final selected publications by elicitation of the required data using an extraction form, as shown in Table 3. We have initially synthesized the extracted data in the form of risk factors as our preliminary results, as shown in Table 2 and Table 3. These risk factors will assist the agile software developers to add avoidance techniques/practices for these identified risks for effective software development.

9. REFERENCES

[1] Shrivastava, Suprika Vasudeva. "Distributed agile software development: A review:1006.1955 (2010).
[2] Paulk, Mark C. "Agile methodologies and process discipline." Institute for Software Research (2002): 3.
[3] Herbsleb, James D., and Deependra Moitra. "Global software development." *Software, IEEE* 18.2 (2001): 16-20.

[4] Turk, Dan, Robert France, and Bernhard Rumpe. "Limitations of agile software processes 1409.6600 (2014).
[5] Jammalamadaka, Kiran, and V. Rama Krishna. "AGILE SOFTWARE DEVELOPMENT AND CHALLENGES."
[6] Kamaruddin, Nina Kamarina, Noor Habibah Arshad, and Azlinah Mohamed. "Chaos issues on communication in Agile Global Software Development." *Business Engineering and Industrial Applications Colloquium (BEIAC), 2012 IEEE. IEEE, 2012.*
[7] Holmström, Helena, et al. "Agile practices reduce distance in global software development." *Information Systems Management* 23.3 (2006): 7-18.
[8] Florin, Dumitru, Gabriela. "Issues and strategy for agile global software development adoption." *Recent researchers in Applied Economics* (2011): 37-42.
[9] Layman, Lucas, et al. "Essential communication practices for Extreme Programming in a global software development team." *Information and software technology* 48.9 (2006): 781-794.
[10] Hossain, Emam, Muhammad Ali Babar, and Hye-young Paik. "Using scrum in global software development: a systematic literature review." *Global Software Engineering, 2009. ICGSE 2009. Fourth IEEE International Conference : (2009).*
[11] Korkala, Mikko, and Pekka Abrahamsson. "Communication in distributed agile development: A case study." *Software Engineering and Advanced Applications, 2007. 33rd EUROMICRO IEEE conference: 2007.*
[12] Jalali, Samireh, and Claes Wohlin. "Agile practices in global software engineering-A systematic map." *Global Software Engineering (ICGSE), 2010 5th IEEE International Conference: 2010.*
[13] Akbar, Rehan, Muhammad Haris, and Majid Naeem. "Agile framework for globally distributed development environment (the DAD model)." *Proceedings of the 8th Conference on Applied Informatics and Communications, 2008.*
[14] C. Okoli and K. Schabram, "A guide to conducting a systematic literature review of information systems research," 2010.
[15] S. Keele, "Guidelines for performing systematic literature reviews in software engineering," Technical report, EBSE Technical Report EBSE-2007-01 2007.
[16] Rashid, Nasir, and Sifat Ullah Khan. "Green Agility for Global Software Development Vendors: A Systematic Literature Review Protocol." *OF THE PAKISTAN ACADEMY OF SCIENCES* (2015): 301.