EMITTER International Journal of Engineering Technology
Vol. 9, No. 1, June 2021, pp. 13~30
DOI : 10.24003/emitter.v9i1.558

Addressing Communication, Coordination and Cultural Issues in Global Software Development Projects

[1]Sami-Ul-Haq, [2]Muhammad Naeem Ahmed Khan, [3]Aamir Mehmood Mirza, [4]Saif Ur Rehman, [5]Raja Asif Wagan, [6]Imran Saleem

[1,2]Independent Research Scholar, Islamabad, Pakistan
[3,5]Balochistan University of Information Technology, Engineering and Management Sciences, Quetta, Pakistan
[4]University Institute of Information Technology, PMAS-University of Arid Agriculture, Rawalpindi
[6]University of Management and Technology, Lahore, Pakistan
E-mail: [1]grtsami48@gmail.com, [2]mnak2010@gmail.com,
[3]mirza.aamir@buitms.edu.pk, [4]Saif@uaar.edu.pk,
[5]raja.asif@buitms.edu.pk, [6]imran.saleem@umt.edu.pk

Received September 8, 2020; Revised October 13, 2020; Accepted December 7, 2020

Abstract

The field of Global Software Development has been an active area of research for the last two decades due to its enormous benefits such as lower labor cost, faster development and easy access to the skilled labor pool. Apart from these benefits, it faces some challenges like communication, coordination, trust and configuration management etc. These challenges arise primarily due to physical, cultural and time zone differences. The empirical studies highlight that the existing Global Software Development solutions do not fully meet the user needs as there are still several gaps in these solutions. Therefore, to fulfill these gaps, there is a need to develop novel frameworks that address outstanding issues. In this paper, we have attempted to address the aforesaid GSD challenges. The practitioners can benefit from our proposed framework during the execution of GSD projects. The proposed framework mainly focuses on the root causes of the two principal challenges namely the communication and cultural differences. We believe that if the team members of a software project can communicate effectively and show considerations for others by imparting due reverence to the cultural norms, then the other residual issues can easily be reduced and minimized.

Keywords: Action-Based Research Methodology, Communication, Coordination, Global Software Development, Trust.

1 INTRODUCTION

Global Software Development (GSD) has attracted much interest during the last two decades due to its enormous benefits such as faster delivery of services, moderately cheaper labor cost and open access to skilled manpower around the globe. Aranda et al. [1] reported that all these factors have eventually materialized in the form of GSD which is a coordinated activity of multisite software development. GSD is defined as a bunch of environments through which a company, generally known as a client, outsources some or entire software development tasks to another organization, known as a vendor, which agrees to provide the required services on payment [2]. Various aspects of GSD are valuable to understand the real strengths and weaknesses of GSD encompassing the overall global scenario. In this connection, the merits of outsourcing software projects in terms of quality of services, availability of skilled manpower, low cost of development and round the clock working operations are discussed in [3]. Whereas, Conchuir et al. [4] identified six main influencing factors of GSD include low-cost of software development, affluent use of time zone effectiveness for providing 24/7 services, cross-site modularization of development activities where software is developed at one site and is tested at another site, access to skilled workforce around the globe, significant benefits of exploiting innovations made by various researchers and sharing best practices and close proximity to the customers.

The key challenges faced by GSD are linked with several other learning domains such as requirement engineering, configuration management and knowledge management. It is not possible to fully exploit GSD benefits without putting in place effective communication mechanisms [5]. Requirement engineering is a vital activity in any software development lifecycle, particularly when development teams are scattered across multiple locations [6]. Cultural differences generate a lot of challenges and project managers need to have sound awareness about different cultures and norms of the team members [7]. Romero et al. [5] argue that cultural differences in the GSD project can be overcome by enhancing communication and mutual coordination. Another challenging task is the selection of proper communication media [6]. Using collaborative tools and effective communication strategies can minimize risks associated with the communication gaps [8]. Communication has a direct impact on trust and asynchronous media for communication like emails have been criticized in [9] as a limited amount of information can be shared through these media; therefore, synchronous communication is preferred over asynchronous communication. However, major barriers like time zone, language impediments and cultural differences pose serious challenges in such type of communication. To this end, agile software methodologies can be useful as these methodologies emphasize improving communication rather than sharing documents [10]. Coordination is another challenging task in GSD. Collaboration models and tools can certainly improve the coordination process and help minimize the risks associated with these issues [8]. Similarly, configuration management becomes critical due to lack of communication, trust and time zone gaps [11]. Knowledge management is one of the most frequent problems that organizations routinely face [12].

Software development activities comprise three layers: human actors, technology and information. The existing literature on GSD focuses more on human issues than technological perspective [14]. Technology has multiple facets that pose serious challenges in GSD projects. For example, the selection of development platforms, bug tracking systems, programming languages, knowledge repositories and communication/collaboration tools necessitate employing sensible methodology [15]. A list of tools used in different GSD knowledge areas is provided in Table 1.

Table 1: List of few tools used in different GSD knowledge areas.

Knowledge Area	Tools
Communication	**Team Collaboration**: MS-SharePoint, Apache Wave, IBM Lotus etc. **Email and Messengers:** Gmail, Yahoo, Hotmail, Outlook etc. Video **Conferencing**: Skype, Zoom, WebEx, etc. **VoIP:** Microsoft NetMeeting, WebEx, WorkSpace3D etc.
Requirement Engineering	**Requirement Management**: IBM Rational DOORS, Abacus, Modern Requirements by eDevTECH, etc. **Requirement Repository**: Rational Requirements Composer etc. **Agile Project Management**: GatherSpace **Project Management**: Microsoft Project, Primavera, Trello Microsoft Team Foundation Server etc.
Testing	**Functional Testing**: TestComplete, Selenium, JUnit, HttpUnit, .TEST, GUITAR etc. **Non-functional Testing**: WebLOAD, LoadRunner, JMeter, OpenSTA, AppLoader, DB Stress etc. **Quality Assurance**: TestLink, KlarosTestmanagement etc.

In this paper, we have highlighted the importance of GSD challenges mainly communication, coordination, trust and configuration management, and we have proposed a framework to amicably address these challenges. The proposed framework is based on theoretical grounds and we have analyzed the performance of our framework through comparison and Root Mean Square Percentage Error (RMSPE). The software industry, research community and GSD practitioners can benefit from our framework during the execution of large-scale projects. Our framework emphasizes that GSD team members working on a software project can communicate effectively and show considerations for others by imparting due reverence to the cultural norms, then the other residual issues can easily be minimized to a greater extent.

2 RELATED WORKS

Setamanit et al. [17] present a framework to help project managers in the planning phase of GSD. The proposed model mainly addresses human resources, manpower allocation, productivity, quality assurance and effects of interaction on the productivity rate. The proposed model is primarily meant for senior management and also does not address other key challenges like requirement engineering, configuration management, communication, coordination and technology-use mediation plans. Vizcaíno et al. [26] propose a game called GSD-Aware to train students in GSD. The authors observed that just after 50 minutes of playing the game, the students were aware of the significant influence of lack of trust, lack of coordination, lack of face-to-face, cultural differences and informal communication, time difference, and lack of team spirit. Stapel et al. [18] proposed a Flow Mapping mechanism for planning and managing communication in the GSD environment. The characteristics of flow mapping are visualization of project participants, documents and information flow to improve awareness in GSD. Aranda et al. [13] discuss communication strategies to abate misunderstandings among GSD team members. To subdue the issue of cultural differences, it is suggested to organize relevant training, employ cultural mediators and practice virtual mentoring. However, to select an appropriate technology a preference-based system is required to decide which groupware tools are more appropriate to use in a given scenario.

A shared and common understanding of the database views is a challenging task for GSD projects. Bartelt et al. [19] discuss an orchestration model for GSD that emphasizes how two companies can work together having their project teams, processes and data storage schema. To work on a GSD project, both the teams would have only a few shared processes and data artifacts that would be visible to both the teams. In this regard, the dashboard is a commonly used tool for managing shared artifacts. Similarly, Ylikotila and Linna [12] propose a collaboration model for GSD that comprises of three phases: initiation, planning and collaboration. There are several actors and tasks associated with each phase of the model and the role and responsibilities of each actor are clearly defined. Palacio et al. [20] also propose a Collaborative Working Spheres tool to facilitate team collaboration. The proposed tool offers features to manage various activities, events, actions and resources in a GSD project. The proposed framework monitors activity information and indicates a suitable time to start collaboration but does not facilitate GSD process improvement. Physical distances and lack of effective communication are major factors that reduce the performance of GSD team members [21]. Wiredu et al. [15] propose a conceptual framework for managing human resource, processes and their interconnections. The proposed framework resolves information ambiguities, manages dependencies among process, determine conflicts among processes. Jalali et al. [22] propose a model for trust dynamics in the GSD project lifecycle. The proposed model entails the static and dynamic trust evolution phases for trust-building. According to the proposed model, the client sets the expected behavior for the trustee. As long as the trustee adheres to the expected behavior, the required trust level is achieved. If the actual and expected trust level is not met, it means that distrust among teams prevails. A comprehensive discussion about various tools used in different GSD process areas is provided in [23] and [24]. Smite et al. [16] stated that effective communication can help build trust but efforts in trust attainment by employing effective communication strategy are considerably less. Most of the studies present a problem-specific strategy, but a comprehensive analysis of the available GSD solutions is a grey area. The empirical studies on GSD [11,16] highlight the need for appropriate frameworks and process models in GSD.

We conclude the following research gaps that specifically relate to the unavailability of appropriate process models and frameworks for GSD.

- The existing efforts are not adequate, thus exhaustive analyses of the solutions are required.

- Communication, coordination, trust, requirement engineering and technology selections are challenging tasks in GSD and need further research.
- Effective management strategies are required to be devised for better project execution.
- Deficiency of frameworks and process models demands devising novel approaches to improve GSD processes.

In this connection, we have proposed a framework for GSD that mainly focuses on the communication issue.

3 ORIGINALITY

In this study, we focus on various GSD issues such as communication, coordination, trust, requirement engineering, configuration management, knowledge management, technology-use mediation, and cultural differences. Nevertheless, the main emphasis of this research is on the key issues of communication, coordination, time zone and cultural differences. The central reason for studying these issues is to identify the root causes and impact of communication over other GSD processes. We have identified open research areas and formulated the following specific research questions:

RQ 1. What are the key challenges of GSD?
RQ 2. What are the existing gaps in GSD research and how these gaps can be addressed?}
RQ 3. What are the root causes of the identified challenges?}
RQ 4. What is the impact of communication on other knowledge areas?
RQ 5. What are the limitations of existing frameworks and process models?

4 SYSTEM DESIGN

In this study, we have employed an Action-based research methodology hence it is important to briefly describe this methodology. Action-based research aims at addressing the real concerns of team members. Strengths of action-based research methodology and grounded theory serve as a motivation to undertake research in GSD. In action-based research, a group of professionals identify a problem and chalk out a strategy to resolve it [25]. Action-based research is a persistent effort to identify the root causes of organizations' problems and improve their quality and performance. Action-based research has four phases: plan, act, observe and reflect. In the planning phase, the actual problem is identified, research questions are formulated and a research plan is developed followed by gathering the required information to find answers for the research questions.

In the action (or act) phase, effective communication is established with the collaborators to share findings established in the planning phase. This way we discover different perspectives about the employed processes. In the reflection phase, meticulous evaluation is performed to observe the impacts of findings made in the previous phase on the overall working of the organization. This necessitates stalking the required change to improve processes. The goal of grounded theory is used to generate an inductive formal theory. Constructivist epistemology is a core concept of grounded theory that emphasizes the need for subjective interrelationships among researchers and participants. Grounded theory is the only option when no other theory is being practiced. However, these methodologies require active and periodic data collection and reporting from collaborators and practitioners. The physical, cultural and time zone differences prevent GSD teams (i.e., clients and vendors) to communicate and coordinate effectively. Thus, the activities in project planning, designing, execution, testing, maintenance and management become cumbersome. Hence, various technology tools are used to perform various project activities such as coordination, requirement engineering, configuration and knowledge management. Given this, we propose a novel framework to provide technological support to the clients and vendors working on GSD projects.

The main objective of this study is to fill research gaps in different processes of GSD. We propose that both clients and vendors are required to be involved in the planning and management of GSD projects. Our framework involves four key process areas including project initialization, planning and management, execution and reflection. The interaction of these processes is shown in Fig 1. A detailed description of the framework is provided below.

4.1 Key process areas

1) *Project initialization:* In this process area, a GSD project is initiated undersigning of an agreement or contract. We propose to organize a web-based orientation or introductory session in which modus operandi to undertake the project as well as cultural norms will be discussed to devise an acceptable communication strategy. Such sessions are helpful to build trust among the GSD team members.
2) *Planning and Management:* In this process area, both vendor and client nominate a focal person for liaison, discuss meeting plans and decide upon various tools to be used for communication, coordination and other knowledge areas.
3) *Execution:* In this process area, the vendor team executes the project. All the key activities such as requirement gathering, system/software development, configuration management, testing (in case if it is required to be performed at the development site) are carried out in this process area. Our proposed framework follows iterative philosophy i.e. when a project build is released, it is sent to the client for inspection and reviews. The client in turn provides feedback and suggestions either for the next build or required changes in the existing build.
4) *Reflection:* In this process area the acceptance testing, whether on-site or factory testing, is performed and the final product is released followed by preparing a lesson learned report and sharing of project-related documents with the client.

4.2 Key features

1) *Shared Management System:* The proposed framework involves a shared management system; both client and vendor work together to create working plans for the project. It means that the client cannot impose its decisions unilaterally as the vendor can put forth its organizational strategy independently. The shared management system creates site-specific plans for the execution of the GSD project.
2) *Orientation:* Orientation is an introductory session in which cultural norms are discussed. It helps resolve communication and trust-building issues.

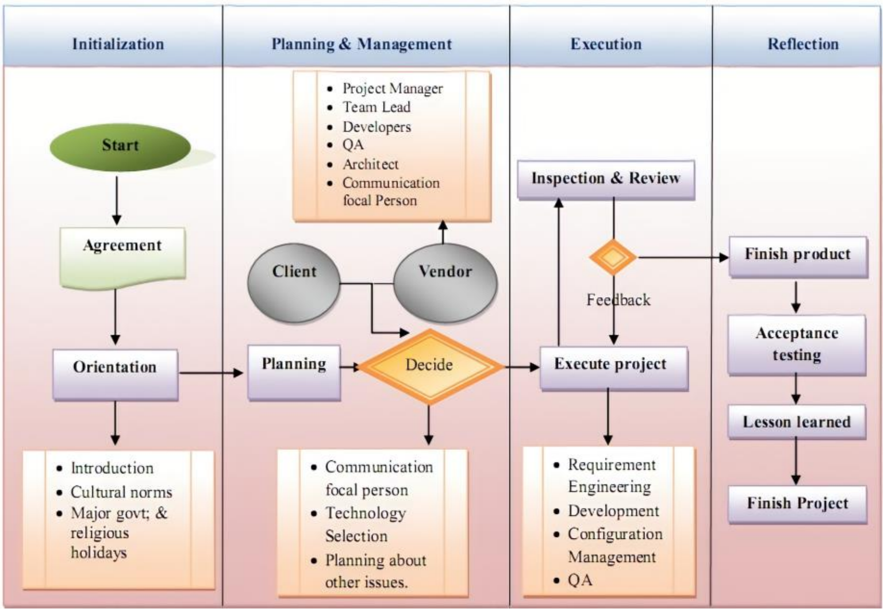

Figure 1: Interaction of GSD Process Areas in the Proposed Framework.

3) *Iterative Philosophy:* Iterative methodology is an agile practice that divides the actual work into multiple parts and puts more emphasis on communication. The iterative philosophy addresses requirement engineering and configuration management issues as well as facilitates defect detection at an earlier stage of a GSD project. An important consideration is that a mature communication process indirectly facilitates building trust between both parties.
4) *Focal Person:* A focal or liaison person proposed in our framework will act as a dedicated communication facilitator at the vendor's side. The focal person is required to be proficient in targeted natural languages (i.e., natural languages spoken by vendor and client). For effective communication, sound technological comprehensions and cultural understandings are mandatory for the focal person. The focal person will act as an interpreter and liaison with the client to resolve different issues through the effective use of communication methods and theories.

5 EXPERIMENT AND ANALYSIS

Our framework is comprised of four key process areas (KPA). In the first KPA, an orientation session has been organized over the internet. For the second KPA, a focal or liaison person as proposed in our framework is nominated for communication facilitator at the vendor's side. The vendor and client worked together and developed plans for the execution of the GSD project. The third KPA is about the execution of the plan and the vendor team executes the project. For the last KPA, acceptance testing is conducted and lesson learned report is developed and project-related documents shared with the client.

5.1 Empirical validation

To validate our proposed framework, we used two datasets discussed in [8] (termed as Dataset-I in this study) and [16] (termed as Dataset-II in this study).

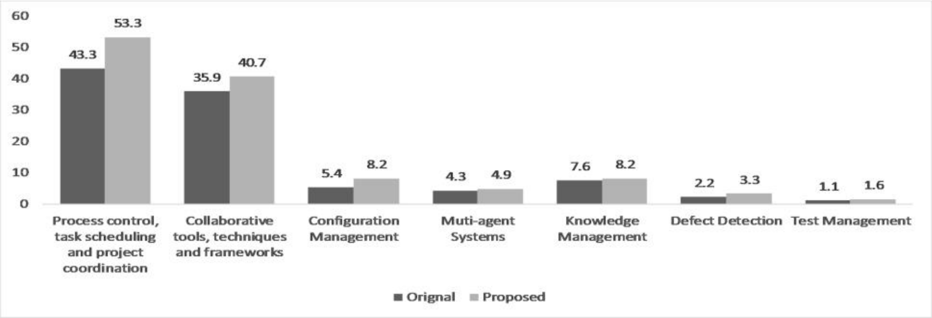

Figure 2: Comparison based on Dataset-I

We have used these datasets as these are the standard datasets used for quantitative measures about different GSD knowledge areas. We took specific threshold values against each knowledge area to evaluate the current level of usefulness of various GSD attributes.

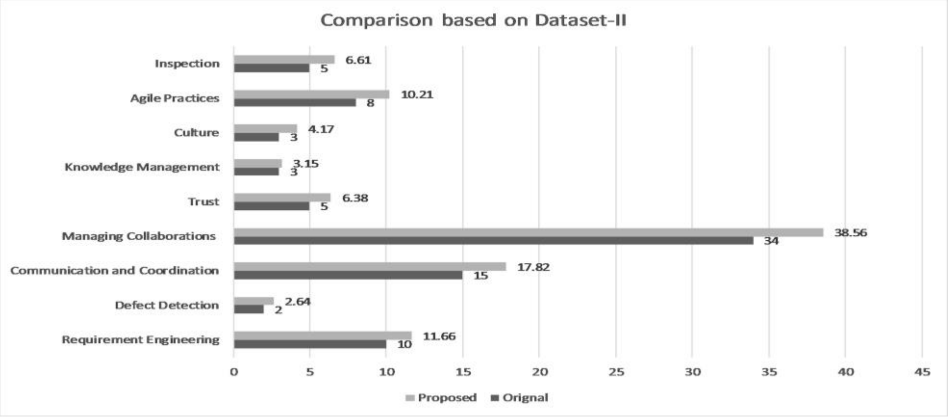

Figure 3: Comparison based on Dataset-II

As new research theories are evolving and practitioners are regularly adopting new technologies, therefore, we have assigned threshold values to each GSD attribute to find the effectiveness of our framework. For this purpose, we compute variance and prediction error.

For comparison purpose, we have used standard root mean square prediction error (RMSPE) (equation 1) to calculate the accuracy of the data.

$$\text{RMSPE} = \sqrt{BIAS^2 + VARIATION^2} \qquad (1)$$

It is worth mentioning that the least values of RMSPE represent better accuracy of results. We obtained variation, biasness and RMSPE as **3.437936, -2.91635 and 4.358899** respectively for Dataset-I; and **1.304357, -1.7961 and 2.219759** respectively for Dataset-II. RMSPE signifies the fitness of a model and lesser values for RMSPE indicate that the fitness of the model is better.

Table 2: Comparison based on Dataset-I (Data compiled from reference [8]).

Parameters	Value in the Year 2009	Threshold Value	%age increase in 2010	Value in the Year 2010	%age increase in 2011	Value in the Year 2011	%age increase in 2012	Value in the Year 2012
Process control, task scheduling and project coordination	43.3	7	3.15	46.6	3.26	49.8	3.49	53.3
Collaborative tools, techniques and frameworks	35.9	4.3	1.53	37.4	1.60	39	1.66	40.7
Configuration Management	5.4	15	0.81	6.2	0.93	7.2	1.07	8.2
Muti-agent Systems	4.3	4.7	0.20	4.5	0.21	4.7	0.22	4.9
Knowledge Management	7.6	2.5	0.19	7.7	0.19	7.9	0.19	8.2
Defect Detection	2.2	15	0.33	2.5	0.37	2.9	0.43	3.3
Test Management	1.1	15	0.16	1.2	0.18	1.4	0.21	1.6

Using the RMSPE formula, we estimate biasness by taking the sum of prediction error divided by the total number of observations. Prediction error shows the difference between the actual and predicted values. The results of our study based on Dataset-I and Dataset-II are shown in Table 2 and Table 3. The corresponding bar graphs for the study results are illustrated in Fig 2 and Fig 3 respectively.

Table 3: Comparison based on Dataset-II (Data compiled from reference [16].

Parameters	Value in the Year 2010	Threshold Value	%age increase in 2011	Value in the Year 2011	%age increase in 2012	Value in the Year 2012
Requirement Engineering	10	8	0.8	10.8	0.86	11.66
Defect Detection	2	15	0.3	2.3	0.34	2.64
Communication and Coordination	15	9	1.35	16.3	1.47	17.82
Managing Collaborations	34	6.5	2.21	36.2	2.35	38.56
Trust	5	13	0.65	5.65	0.73	6.38
Knowledge Management	3	2.5	0.075	3.07	0.07	3.15
Culture	3	17	0.51	3.51	0.59	4.17
Agile Practices	8	13	1.04	9.04	1.17	10.21
Inspection	5	15	0.75	5.75	0.86	6.61

5.1.1 Case study - questionnaire-based survey

To further validate our framework, we conducted a survey in which experts and practitioners from the leading IT companies that work on GSD projects and provide offshore services participated. This survey aimed to determine possible solutions to different GSD key challenges from practitioners' perspectives. The survey findings reveal that our proposed framework can overcome several GSD challenges.

5.1.2 Survey context

We collected data from 94 GSD experts and practitioners by sending them a questionnaire. We divided the organizations that participated in the survey into four types based on the experience of working on GSD projects. The four types of organizations were determined based on the experience as follows: Novice (1 to 2 years), Advanced (3 to 4 years), Matured (5 to 10 years) and Recognized (10+ years). The participation, in percentage, of the Novice, Matured, Advanced and Recognized organizations were 21%, 19%, 36% and 24% respectively. The size of each company that participated in the survey was more than 50 professionals. The roles of the survey participants included managers, team leads, software engineers and quality assurance personnel each having at least five years of professional experience of working on GSD projects. The eight major questions of the survey questionnaire are listed in Table 4.

Table 4: Key GSD questions of Survey Questionnaire.

Sr	Question
1	What are the most critical issues and challenges in GSD?
2	What are the root causes of GSD challenges?
3	What are the research gaps in GSD that need to be addressed?
4	Can we classify GSD activities into the following four phases? (a) Project Initialization (b) Plan and Manage (c) Execution (d) Reflection?
5	Whether client-vendor mutual coordination and communication (i.e., Shared Management System) for decision making is required for GSD?
6	To what extent orientation, iterative philosophy and communication affect project success?
7	Communication challenge is more important for which phase of SDLC?
8	What are the most critical challenges in GSD viz-a-viz your experience level in GSD?

5.1.3 Survey results

- The survey results for the question **"What are the most critical issues and challenges in GSD?"** are shown in the form of a bar graph in Fig 4.

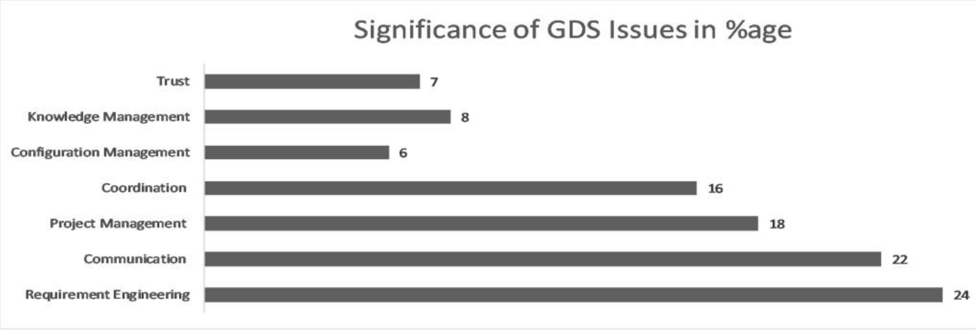

Figure 4: Weights of major issues in GSD.

The survey results indicate that majority of the practitioners considered the requirement engineering (24%), communication (22%), project management (18%) and coordination (16%) as the major issues in GSD.

- The survey results for the question **"What are root causes of GSD challenges?"** are shown in the form of a bar graph in Fig 5. The survey participants identified cultural variations (35%), time zone differences (31%) and lack of communication (29%) as the prominent root causes of different GSD challenges.

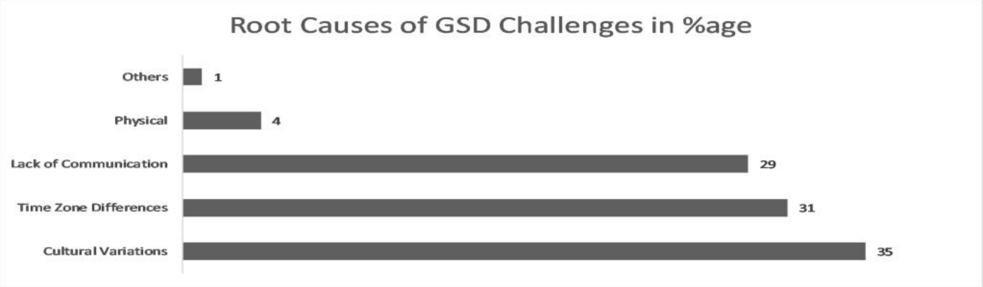

Figure 5: Prominent root causes of GSD challenges

- The survey results for the question **"What are research gaps in GSD that need to be addressed?"** are shown in the form of a bar graph in Fig 6. According to the survey participants, the most important research gaps in GSD are the inadequacy of techniques (41%), best practices (34%) and state-of-the-art tools (25%).

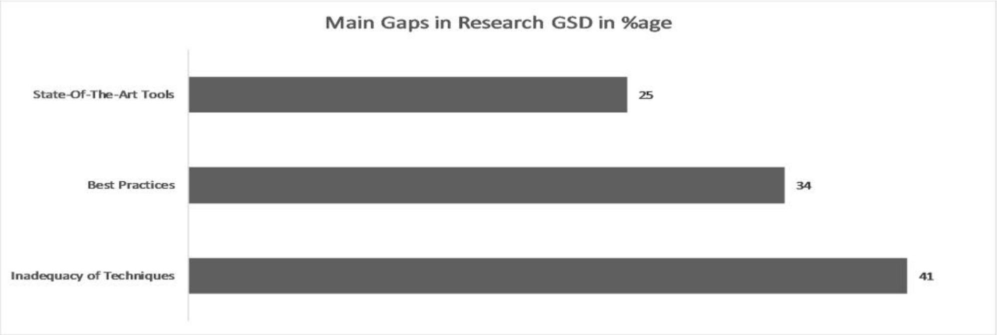

Figure 6: The foremost research gaps in GSD.

- The survey results for the question "Can we classify GSD activities into the following four phases? (a) Project Initialization (b) Plan and Manage (c) Execution (d) Reflection?" are shown in the form of a pie graph in Fig 7. A vast majority of the survey participants (83%) agreed to the suggested classification of GSD activities.

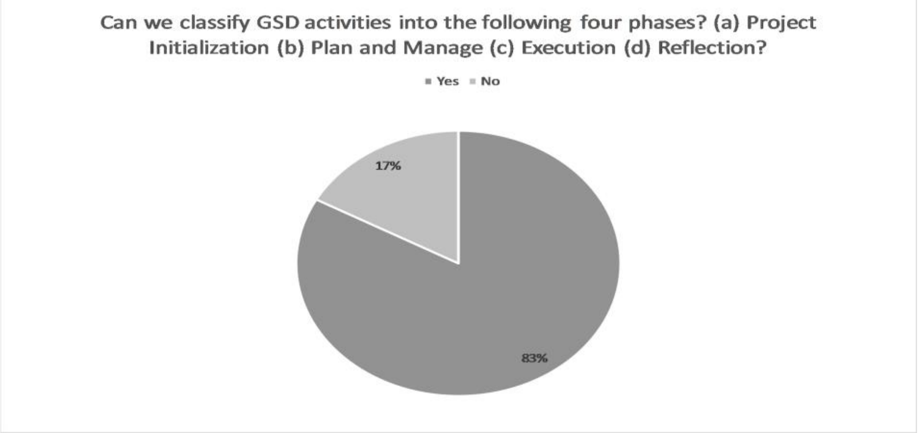

Figure 7: The foremost research gaps in GSD.

- The survey results for the question **"Whether client-vendor mutual coordination and communication for decision making are required for GSD?"** are shown in the form of a pie graph in Fig 8. As evident from the pie graph, 95% of the practitioners agreed that close client-vendor coordination is imperative for decision making in the GSD environment.

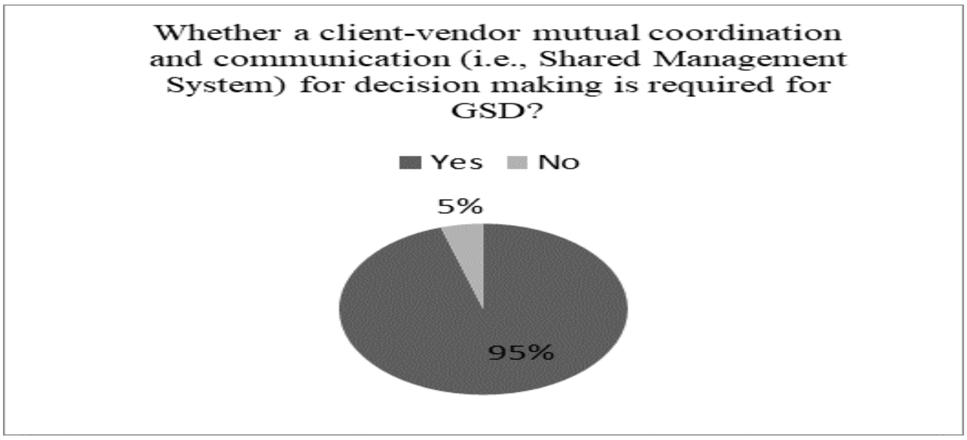

Figure 8: Weights of major issues in GSD.

- The survey results for the question **"To what extent orientation, iterative philosophy and communication affect the project success?"** is shown in the form of a column graph in Fig 9. The survey results point out that an effective communication mechanism is necessary for the success of a GSD project.

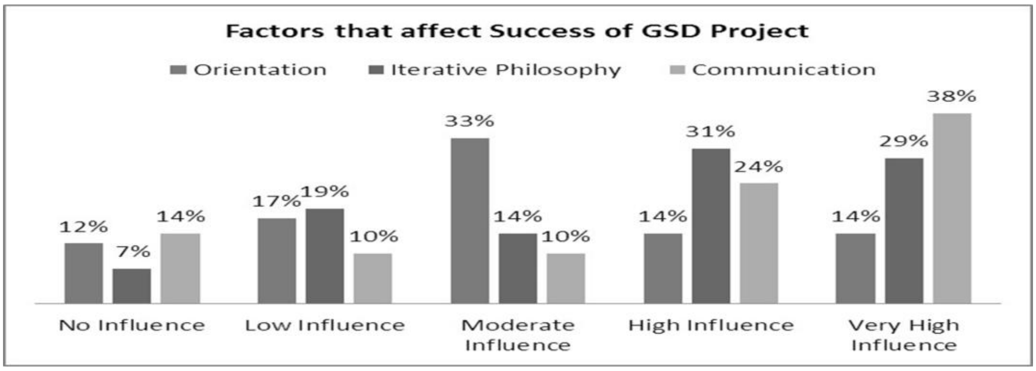

Figure 9: The effects of orientation, iterative philosophy and communication on GSD project success.

- The survey results for the question **"Communication challenge is more important for which phase of SDLC?"** are shown in the form of a pie graph in Fig 10. According to the survey results, communication is important for all phases of SDLC in general and the development phase in particular.

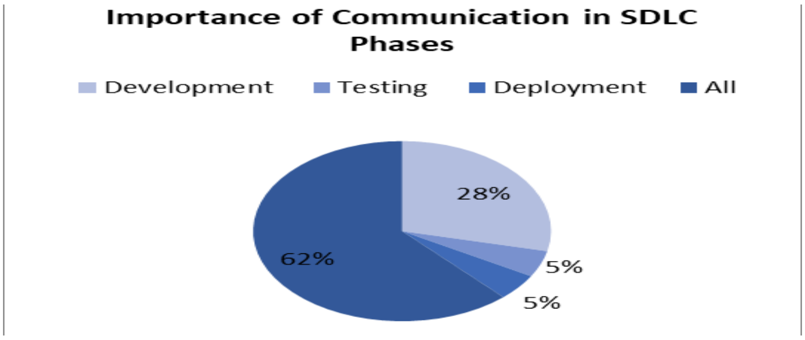

Figure 10: Importance of communication in the SDLC phase.

- The survey results for the question **"What are the most critical challenges in GSD viz-a-viz your experience level in GSD?"** are shown in the form of a column graph in Fig 11.

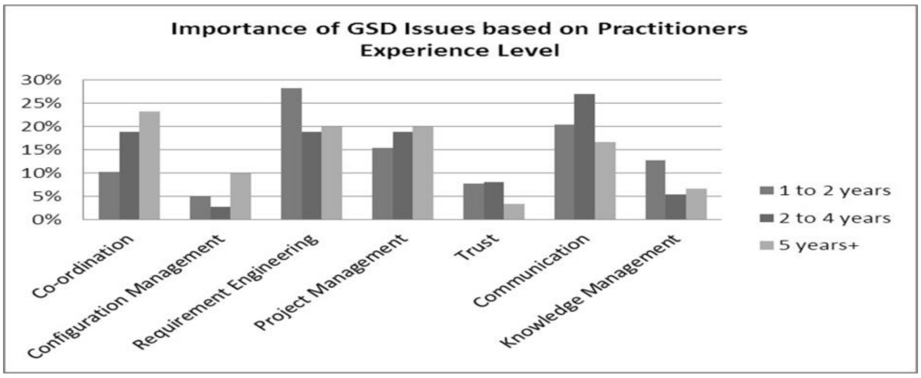

Figure 11: Importance of communication in the SDLC phase.

According to the seasoned GSD practitioners having 5+ years of working experience, requirement engineering, communication and project management are the most important issues in the GSD environment. The survey results for this question were in total conformance with the results of Q1. Overall, the survey results were also in conformity with our proposed framework.

5.2 Results and discussion

Communication, time zone and cultural diversity are the main factors that affect planning, requirement engineering, configuration management, quality assurance and technology selection in GSD. Our proposed framework offers a wide range of support for handling these challenges and allows the participation of different actors in GSD. We compared our framework with nine existing frameworks described in Table 4. We observe that existing frameworks are mostly problem specific. Only one model proposed by Ylikotila and Linna [12] offers broad features and support for GSD development, but it lacks a well-defined mechanism for technology selection. Our framework addresses the aforesaid issues in the following ways:

- In the initialization phase, we suggest holding an orientation activity that creates cultural awareness among team members and helps build trust.
- All team members should be involved in creating a management plan. This approach facilitates trust-building and improves coordination.
- Schedule of regular meetings, communication mechanism and tools to be used should be pre-planned. The required tools should be selected following the mutual consent of the vendor and client.
- There would be a focal person to meet the communication needs of GSD teams. It is the best practices reported in the contemporary literature as well.
- GSD project is executed by following the iterative philosophy of agile methods. This way, communication, coordination and quality of requirement engineering processes are improved.

- Throughout the GSD project lifecycle, our framework focuses on effective communication and coordination among the teams.
- Target users of our framework are all the actors involved in GSD, i.e., vendor, client and project managers.

Based on the methodology adopted in our framework, it can be observed from Table 5 that it meets all the key attributes of a GSD environment. Furthermore, in this study, we have been able to find an answer to the research questions formulated in the fourth section.

RQ1. What are the key challenges of GSD?

This study identifies that communication, coordination, trust, requirement engineering, project management, configuration management, knowledge management and technology selection are the key challenges faced during the execution of a GSD project.

RQ2. What are the existing gaps in GSD research and how these gaps can be addressed?

This study reveals that existing solutions do not address all the GSD challenges and there is a need for augmented frameworks to address GSD issues. These research gaps can be filled by developing and adopting new tools, techniques, methodologies, and frameworks to efficiently execute GSD projects. However, for this purpose, effective management strategies are required.

RQ3. What are the root causes of the identified challenges?

Physical, cultural, time zone differences are the root causes of the aforementioned challenges. Moreover, communication itself causes several other challenges.

RQ4: What is the impact of communication on other knowledge areas?

Ineffective communication affects other knowledge areas, processes and activities such as coordination, trust, requirement engineering, knowledge management and configuration management. It creates a lot of hurdles for smooth project execution and makes project management cumbersome.

RQ5: What are the limitations of existing frameworks and process models?

This study identifies that effective communication can facilitate trust-building but practitioners do not make significant efforts to employ effective communication strategies. Secondly, contemporary studies present problem-specific solutions only to manage communication needs. We put forward the following recommendations for undertaking a GSD project.

- Arrange an orientation session.
- Nominate a focal person who has proficiency in the targeted languages.
- Schedule for regular meetings, communication media and tool support should be decided at the outset.
- Execute the project based on iterative philosophy.

- Maintain a centralized knowledge base to support knowledge management activities. Whenever a project activity is updated, it should be instantly shared with the entire team.
- Each team should maintain its configuration files to avoid integration problems.

Table 5: Comparison with contemporary studies

Ref #	Actor Support			Process/Activity Support							
	Client	*Vendor*	*Project Manager*	*Communication*	*Coordination*	*Trust*	*Configuration Management*	*Requirement Engineering*	*Project Management*	*Technology Selection*	*Quality Assurance*
[12]	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
[13]	✓	✓		✓							
[15]	✓	✓	✓		✓						
[17]			✓						✓		✓
[18]			✓						✓		
[19]	✓	✓	✓						✓		
[20]	✓	✓		✓	✓						
[21]	✓	✓			✓					✓	
[22]	✓	✓				✓					
[26]											
Our Framework	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

6 Conclusion and future work

In this study, we proposed a novel framework that identifies and eliminates the root causes of major challenges of GSD such as communication and cultural differences. We believe that if team members can effectively communicate with each other and have awareness about different cultural norms, the effects of other GSD issues are automatically minimized to a major extent. This study accounts for communication, coordination and cultural issues in GSD as these are the prime issues in GSD. The proposed framework is based on theoretical grounds and we have analyzed the performance of our framework through comparison and RMSPE. The proposed framework can effectively manage different GSD processes and activities. Both the industry and research community can benefit from our framework. The industry can benefit from it during the execution of GSD projects and academia/researchers can use it to further enhance the proposed methodology. As a future dimension to this work, we intend to conduct a case study that will account for periodic reporting and data collection to reflect the current state of affairs in the domain of GSD and comparing the results of the case study with latest dataset.

REFERENCES

[1] Aranda G N, Vizcaíno A, Piattini M (2010). **Analyzing and evaluating the main factors that challenge global software development**. Open Software Engineering Journal, 4(1), 14-25.

[2] Babar M A, Verner J M, Nguyen P T (2007). **Establishing and maintaining trust in software outsourcing relationships: An empirical investigation**. Journal of Systems and software, 80(9), 1438-1449.

[3] Ramingwong S, Ramingwong L (2010). **The Paradoxical Relationships of Risks and Benefits in Offshore Outsourcing of Software Projects.** Open Software Engineering Journal, 3, 35-38.

[4] Conchúir E Ó, Ågerfalk P J, Olsson H H, Fitzgerald B (2009). **Global software development: where are the benefits?**. Communications of the ACM, 52(8), 127-131.

[5] Romero M, Vizcaíno A, Piattini M (2009). **Teaching requirements elicitation within the context of global software development.** In Computer Science (ENC), 2009 Mexican International Conference on (pp. 232-239). IEEE.

[6] Sabahat N, Iqbal F, Azam F, Javed M Y (2010). **An iterative approach for global requirements elicitation: A case study analysis.** In Electronics and Information Engineering (ICEIE), 2010 International Conference On (Vol. 1, pp. V1-361). IEEE.

[7] Deshpande S, Richardson I, Casey V, Beecham S (2010). **Culture in global software development-a weakness or strength?.** In Global Software Engineering (ICGSE), 2010 5th IEEE International Conference on (pp. 67-76). IEEE.

[8] Jiménez M, Piattini M, Vizcaíno A (2009). **Challenges and improvements in distributed software development: A systematic review.** Advances in Software Engineering, 2009.

[9] Casey V (2010). **Developing trust in virtual software development teams**. Journal of theoretical and applied electronic commerce research, 5(2), 41-58.

[10] Korkala M, Pikkarainen M, Conboy K (2010). **A case study of customer communication in globally distributed software product development.** In Proceedings of the 11th International Conference on Product Focused Software (pp. 43-46). ACM.

[11] Pilatti L, Audy J L N, Prikladnicki R (2006). **Software configuration management over a global software development environment: lessons learned from a case study.** In Proceedings of the 2006 international workshop on Global software development for the practitioner (pp. 45-50). ACM.

[12] Ylikotila T, Linna P (2011). **A collaboration model for global multicultural software development.** Information Modelling and Knowledge Bases XXII, 225, 321.

[13] Aranda G N, Vizcaíno A, Piattini M (2010**). Analyzing and evaluating the main factors that challenge global software development.** Open Software Engineering Journal, 4(1), 14-25.

[14] Clear T, MacDonell S G (2011). **Understanding technology use in global virtual teams: Research methodologies and methods.** Information and Software Technology, 53(9), 994-1011.

[15] Wiredu G O (2006). **A framework for the analysis of coordination in global software development.** In Proceedings of the 2006 international workshop on Global software development for the practitioner (pp. 38-44). ACM.

[16] Šmite D, Wohlin C, Gorschek T, Feldt R (2010). **Empirical evidence in global software engineering: a systematic review.** Empirical software engineering, 15(1), 91-118.

[17] Setamanit S O, Wakeland W, Raffo D (2006). **Planning and improving global software development process using simulation.** In Proceedings of the 2006 international workshop on Global software development for the practitioner (pp. 8-14). ACM.

[18] Stapel K, Knauss E, Schneider K, Zazworka N (2011, August). **FLOW mapping: planning and managing communication in distributed teams.** In Global Software Engineering (ICGSE), 2011 6th IEEE International Conference on (pp. 190-199). IEEE.

[19] Bartelt C, Broy M, Herrmann C, Knauss E, Kuhrmann M, Rausch **A, Schneider K (2009). Orchestration of global software engineering projects-position paper.** In Global Software Engineering, 2009. ICGSE 2009. Fourth IEEE International Conference on (pp. 332-337). IEEE.

[20] Palacio R R, Morán A L, González V M (2010). **CWS: an awareness tool to support starting collaboration in global software development.** Open Software Engineering Journal, 4(1), 38-51.

[21] Redmiles D, VanDerHoek A, Al-Ani B, Hildenbrand T, Quirk S, Sarma A, Trainer E (2007). **Continuous coordination-a new paradigm to support globally distributed software development projects.** Wirtschafts Informatik, 49(1), 28.

[22] Jalali S, Gencel C, Šmite D (2010). **Trust dynamics in global software engineering.** In Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement (p. 23). ACM.

[23] Portillo-Rodríguez J, Vizcaíno A, Piattini M, Beecham S (2012). **Tools used in Global Software Engineering: A systematic mapping review.** Information and Software Technology, 54(7), 663-685.

[24] Lanubile F, Ebert C, Prikladnicki R, Vizcaíno A (2010). **Collaboration tools for global software engineering.** IEEE software, 27(2).

[25] O'Brien R (1998). **An overview of the methodological approach of action research.** University of Toronto, Canada.

[26] Vizcaíno, F. García, I. G. R. D. Guzmán, and M. Á. Moraga, **Evaluating GSD-aware: A serious game for discovering global software development challenges**, ACM Transactions on Computing Education (TOCE), vol. 19, no. 2, pp. 1–23, 2019.