Turkish Online Journal of Qualitative Inquiry (TOJQI)
Volume 12, Issue 5, July 2021: 3331-3364

Research Article

A Framework for Software Customization in Global Software Development (GSD)

[1]Muhammad Nadim, [2]Rana M. Amir Latif, [3]Khalid Hussain, [4]NZ Jhanjhi, [5]Mehedi Masud, [6]Saleh Yahya Alyahyan, [7]Mamoona Humayun

Abstract

Global Software Development (GSD) is an aspect of software engineering that requires the joint efforts of professional software from separate locations. The market is motivated to start GSD due to a lack of local software developers, inadequate resources, fixed budget, and time. Many companies have shifted to software development outsourcing because of the high availability of resources, which cost one-third less than onshore vendors, skilled and professional developers. Vendors improve their skills through outsourced project experience and learn new ways to satisfy client needs. However, many risks are involved in an outsourcing process. GSD becomes the latest and popular development in software engineering, brings a new opportunity to getting resource mobility, market speeding time, obtaining additional information, and also helps to increase operational efficiency. In this paper, we have presented a framework for software customization in global software development. We have divided our approach into two parts. In the first part, we have analyzed the current literature work on the requirement change management process to find the limitation, gaps, and issues of Requirement Change Management (RCM). Our focus is on proposing a framework for software customization in a distributed environment in the second part. In our framework, we overcome the gap between the client and teams and, with the help of a requirement graph, increase the understanding of change requests at the development site. We use the knowledge repository to save the record of a change request. To evaluate our research work, initially, we conducted a case study to evaluate our framework in a real-life context. We have applied the proposed framework on a case study of "Finance Automation" to prove the applicability of our proposed approach. Then we evaluate our framework with the help of the survey. In the survey, we get feedback from the people of different sectors. Lastly, we analyzed the feedback of the survey that is shown as graphed.

[1]Department of Computer Science PMAS-Arid Agriculture University Rawalpindi, Campus
[2]Department of Computer Science, COMSATS University Islamabad, Sahiwal Campus
[3]National Superior Institute of Science and Technology, Islamabad.
[4]School of Computer Science and Engineer, SCE, Taylor's University, Malaysia
[5]Department of Computer Science, College of Computer and Information Technology, Taif University, PO Box. 11099, Taif 21994, Saudi Arabia
[6]Department of Computer Science, Community College in Dwadmi, Shaqra University, Saudi Arabia
[7]Department of Information Systems, College of Computer and Information Sciences, Jouf University, Sakaka , KSA
E-Mail: muhammadnadim096@gmail.com

Keywords: Global Software Development (GSD), Requirement Change Management (RCM), Case Study, Revenue Cycle Management (RCM), Survey

1 Introduction

Requirement engineering identifies the context in which the software system will be used. Requirement engineering works as a bridge to overcome the users, customers, and other stakeholders affected by the system and provides opportunities through software technology. The success of a project largely depends on requirements quality. Many factors help stakeholders achieve customer satisfaction and quality; Software Requirement Specification (SRS) plays an important role [1]. When it is badly written requirements specification, development teams often face challenges in getting the required knowledge at the right moment. Another challenge in the GSD environment is requirements management. Managing customer requirements is a principal factor in the development process for the marketplace [2]. Although it is difficult to manage these requirements locally, it is more difficult to distribute these requirements to multiple teams and communicate them over organizational boundaries. This difficulty increases in GSD projects and projects that have many stakeholders over organizational and cultural boundaries [3].

Over the last two decades, the number of publications has increased in engineering and management. Customized software products are now distributed within organizational boundaries and multi-site organizations. We believe that managing customize requirements in the GSD context is important and difficult. Also, customized software products are now distributed over organizational boundaries and multi-site organizations [4]. The process of customizing software and customization requirements in GSD environments should be investigated to understand the challenges in the context of distributed software development projects. This research investigates the problem faced during the RCM process in the GSD environment [5]. Many researchers pointed out that managing requirement change is still weak. The collective guidance for RCM is still weak and needs to develop systematic practices for requirement change management. Many researchers provide different solutions for requirement engineering implementation in GSD, but there is a lack of details about the process. It is difficult to face the challenges of GSD without any suitable process of Requirement Engineering (RE), especially the RCM process [6].

Due to the novelty of the GSD, this study aims to explore the RCM process for better understanding and improvement of the RCM process. We believe that overcoming the difficulties in the RE process in GSD is an important step towards achieving success in GSD. Due to the GSD demands, suitable and proper methods, processes and models can increase the efficiency and effectiveness of GSD work. We are trying to build up a small contribution in this area by proposing our framework for requirement change management for a distributed environment [7]. Another problem that the GSD team faces is related to changing requirements and their management as compared to onshore development. Due to cultural, temporal, and geographical differences, it is an arduous task to manage and record every changing requirement in a multi-site development [8]. Software teams mutually develop a software project in different places, as shown in Figure 1. Due to time zone difference, teams work 24 Hours constantly, which increases the efficiency rate of GSD. Skilled and qualified developers are hired from all over the world who increase the success rate of software projects. The major purpose of developing software to increase the market value, and decrease development cost, hiring skilled and professional developer of a diverse background.

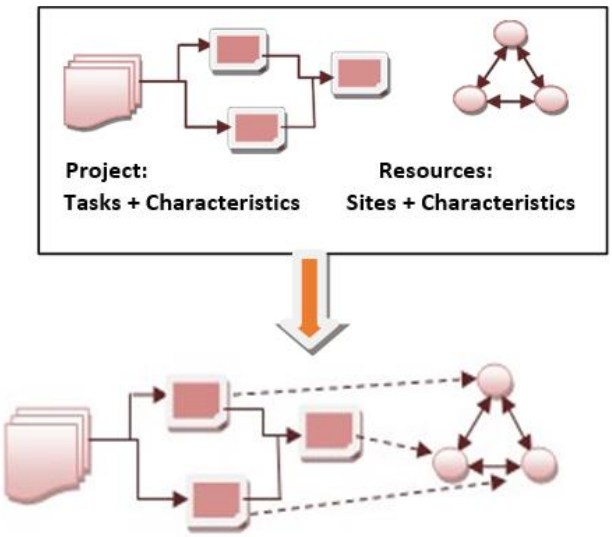


Figure 1 Global Software Development [9]

Many GSD Projects have failed because of a deficiency of face-to-face communication and coordination, problems related to information security, cultural differences, and infrastructure issues, and so on shown in Figure 2.

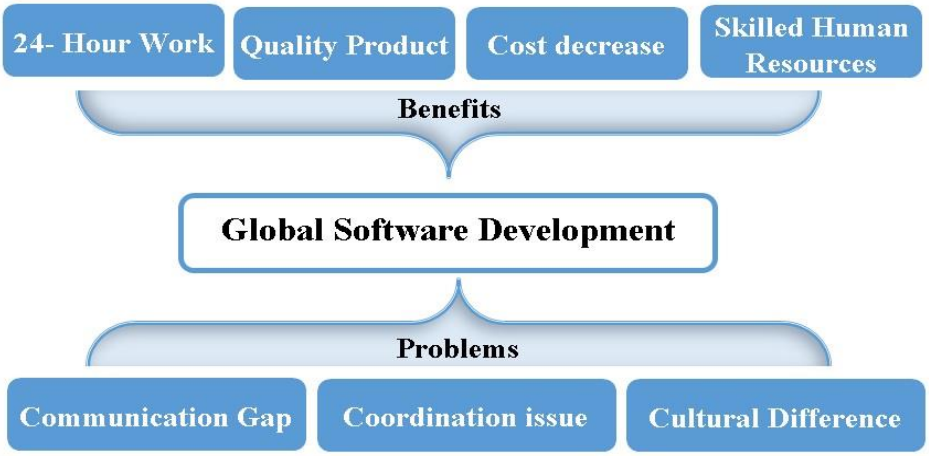


Figure 2 GSD Benefits and Problems [10]

The objective of this study is to examine the exiting RCM process to identify the limitation and problems. Through a literature review, it is intended to identify the method that is used to reduce communication risk. Also, to improve the change management process for a GSD environment.

1.1 Research Questions

- RQ1: Which factors affect the change management process in global software development?
- RQ2: How can we manage the change requirement across organizational boundaries in a distributed environment?

- RQ3: How can we reduce the communication gap between the team and the client?

2 Literature Review

Effective control of development software plays an essential part in gathering and enforcing specifications for all software systems. In the sense of global software development (GSD), its relevance is growing with stakeholders worldwide. In GSD, obstacles such as cultural gaps and time zone discrepancies establish major hurdles throughout the selection of specifications. Therefore, the need for successful project management grows steadily to overcome GSD obstacles. This paper explores potential strategies and methods for successful global project management of apps. The Systematic Literature Review (SLR) describes 25 activities [11]. These activities can help tech providers handle GSD development ventures more efficiently.

Global software development (GSD), due to difficulties such as geographical distance, linguistic hurdles, time zones, and cultural diversity, remains a demanding activity for software vendors. GSD specifications are too challenging to gather and enforce and may impact product consistency and project performance. In much previous research in conjunction with GSD, regular and productive negotiations and dialogue during efficient requirement collection and execution are critical success factors. This present research analysis discusses and explores the value of regular and useful discussions utilizing the Comprehensive Literature Review (SLR) [12]. The study is often carried out based on the software firm's scale, regions, period, and testing methods.

The sophistication of computing grows as industrial economies rely heavily on large-scale information structures that progressively function inside a continuously accessible environment. Global software development (GSD) vendors are trying to decompose this uncertainty through the breakdown of the desired consumer product through different in-house, manufacturing, or commercially buying components (COTS). These modules are eventually assembled into a final job object. However, there is a shortage of clear procedures in the product development process for GSD suppliers. Our previous analysis described a set of nine Essential Success Factors (CSFs) for product integration for GSD vendors. The author performed a recent Systematic Literature Review (SLR) to support GSD vendors in adopting defined CSFs, classify procedures, and verify the findings using a sample survey [13]. The author defined 116 practices/solutions to incorporate applications to enable GSD vendors to reduce their application integration phase complexities.

Due to their agile existence, agile methods are regarded as a significant appeal of global software development (GSD) ventures. In addition to its considerable benefits to GSD, few obstacles impede its adoption across the global tech industry. This research aims to establish a comprehensive literature review to examine the main factors that affect agile adaptation worldwide. Twenty-eight empirical studies (2015-2019) have been reported and analyzed. Scrum and Intense Programming (XP) are the most common agile techniques applied to the applications and the organizational framework. Also described are five types of software, i.e., modeling, condition elicitation, data monitoring tools, etc., typically utilized during agile implementation. The key results of this research suggest that these agile methodologies are well modified because of their iterative process and fast code creation [14]. Still, fundamental problems such as low consumer engagement and lack of documentation significantly impact their development worldwide.

Organizations for global software creation (GSD) also promoted better product consistency applications. The GSD using state-of-the-art technologies in one location across the website. Contact, teamwork, and regulation are the key obstacles confronting GSD. These challenges

generate uncertainty and incompleteness in prioritizing criteria and traceability due to multiple stakeholders and human resources. The author, therefore, introduced a system to enhance the priority criteria and traceability mechanism utilizing artificial intelligence technology [15]. The system was experimentally tested and contrasted to current strategies. Results defined that the proposed system enhanced prioritization and traceability criteria dramatically, with less human intervention in-GSD challenges.

Implementing specifications is a challenging software engineering process. Global Software Creation (GSD) renders things more challenging. Progress variables must be discussed in the sense of GSD during Requirement Engineering (RE). Success factors are identified and then evaluated in this study by Systematic Literature Review (SLR). The variables established are analyzed by numerous study techniques such as case reports, interviews, questionnaires, surveys, and tests. The variables found are evaluated in various sub-continents, the scale, and the length of tech firms. The SLR findings will allow manufacturers to enforce improved specifications [16].

The author intends to provide in-depth observations into the effective implementation of agile GSD practices between 1999 and 2016 and define the most commonly used agile strategies and delivery scenarios. The author is now looking to find study possibilities and weaknesses in the context of agile GSD [17].

Global Software Creation (GSD) has many advantages. It has many problems, on the other side, including its effect on software quality. Enhancing the consistency of software goods will enhance the operation of software processes such as Functionality Maturity Model Integration (CMMI). However, CMMI has seldom been addressed as a process management model in the GSD sense. A comprehensive literature analysis was carried out using Kitchenham's approach to address this void. Between 2013 and 2018, GSD analysis from many datasets was examined. There are 12 papers picked at the end. The results demonstrate that CMMI is crucial to the progress of GSD ventures. However, several customizations and combinations with other specifications are needed. Concept Process Areas (PA) and CMMI sophistication are used for various techniques in the latest Business Process Improvement (SPI) specifications. The findings of this study can contribute to the creation of custom CMMI to improve the success rate of GSD projects [18].

The current study aims to recognize key barriers/challenges that obstruct the process of incorporation for various types and sizes of projects at every point in the GSD setting [19].

As a consequence of digital and communication technology (ICTs), app production has moved from local to multinational software creation (GSD). Despite the advantages obtained from GSD, manufacturers experience difficulties when combining products produced independently by multinational teams. This present study aims to evaluate the list of essential challenges/barriers (CBS), which at all times impede the integration phase. The author also carried out a comprehensive literature review (SLR) to accomplish the goal and retrieved data from 88 articles in six public libraries. A total of 16 obstacles were listed, ten of which are categorized as CBS [20]. Four of the largest rates of obstacles are "lack of coordination," "lack of sufficient paperwork," "lack of usability," and "lack of design."

Graph analytics and large data have captivated the interest of academics and scientists worldwide. When biological data is exposed to graph analytic techniques, it can be interpreted in an interpretable manner. This study aims to conduct an analytical assessment of three tools: the MEME set, DMINDA, and Neo4j. According to the findings of this analysis, visualizing the similarity matrix of repeated patterns reflects the most and least related patterns in the sequences depicted in the conclusion graph [21].

Graph analytics' relevance cannot be overstated. When biological data is exposed to graph analytic techniques, it can be interpreted in an interpretable manner. This article discusses two distinct methods for visualizing a graph. According to the findings of this analysis, visualizing the similarity matrix of repeated patterns reflects the most and least related patterns in the series. Chart analytics and big data have captured the interest of academics and scientists from all around the world. It can yield significant insights in a variety of domains, including life sciences, industry, computer sciences, and engineering. Neo4j and the Cypher query language were used in the research [22].

This paper reports on our concept of Guidelines for global software development (GSD) management that incorporate the basic practice of Capability Maturity Model Integration (CMMI) level 2 management of requirements. The guidelines provide a Change Management and Traceability Paradigm, facilitating the relevant CMMI Level 2 criterion. Also, the Project Management Body of Information (PMBOK) method community (PG) adaptation for project life-cycle activities is established to facilitate the efficient management of system engineering processes [23]. The author is adding a cloud-based reactive middleware to handle GSD initiatives to control transition and to track. Global software development [24-26] plays a vital role in speedy software development by making this possible where contributors can be from several different geographical locations. At the same time the improving the sharing and quality improve the software development process [27-29]. Developer can use different best suited development methodologies.

3 Materials and Methods

We have also discussed the research approach and the researcher's role, research background, ethical issues, and case description. This section aims to explain the process of data collection and data analysis. We use a case study and survey to evaluate our proposed work in a real-life context.

3.1 Research Methodology

Selection and decision of the research methodology are related to a research project and how the researcher's plan includes data collection and data analysis and includes how to implement a research project. Software engineering researcher has a sufficient range of available research methods. Most of these five classes are identified in research methods related to software engineering research:

- Controlled Experiment
- Survey
- Action Research
- Case Study
- Ethnography

Every research method has some strengths and weaknesses. So, no single research method is generally applicable.

3.1.1 Research Methods

The selection of research methods is the main concern of the research topic and research questions and considers essential factors to be considered, such as resource, time, and accessibility issues. Based on the research topic, the objective of this study to investigate and understand RCM issues in GSD and mitigate these issues with the help of the proper approach. Therefore, for exploratory and confirmatory investigation research method was necessary. There are two main reasons for survey methodology that offer difficulties in the research objective. First of all, a survey is not able to answer the 'why' research question. Secondly, the survey tries to deal with the context, but the ability is quite limited. The reliability of a survey depends on different factors related to the respondent. Also, the survey method has a limited number of questions to draw satisfactory responses and only provides one data source. Different methods are used for collecting data for research purposes, but due to various benefits, strengths, and advantages, the survey method is preferred by many researchers. The survey provides a facility to represent many populations, provides a convenient way for data collection, and gives a precise result.
Compared to the survey, controlled experiments answer the 'why' and 'how' types of research questions. Control experiments are not suitable for current research objective because it is conducted when the investigator holds the situation with precise, direct and systematic manipulation of the studied phenomenon. Another reason is that it ignores contextual factors like political/ social and organizational factors and is extremely time-consuming. The controlled experiment cannot be applied when the context is important, and you cannot separate phenomena from context. Another research method is action research that tries to solve a real-world problem while the researcher is simultaneously studying the experience of solving this issue. As an empirical method of action research, the biggest challenge is its immaturity. But the use of action research improves the specific practice. Action research is based on implementation, evaluation, and critical analysis of the collected data to improve the relevant practices.
Ethnography is a research method in which research focuses on field observation. In software engineering, ethnography can help understand how communication strategy practices to build a culture of technical communities, enabling them to perform technical work collaboratively. However, it is based on fieldwork, and it requires the researcher's involvement for a long period in the GSD phenomenon. In this case, ethnography is not appropriate for this study. The case study helps to investigate a contemporary phenomenon within its real-life context when the boundaries are not clear between context and phenomenon. In software engineering, it is a suitable research method. Also, predefined phenomena are investigated without explicit control of variables using a case study. The author identifies the case study use when:

- "Why' or 'How' type of research questions is being faced
- Investigators want to have little control over events
- Focus is a contemporary phenomenon within its real-life context

Consequently, we chose action research, case study, and survey as a research method.

3.2 Action Research

Action Research provides an investigation process that brings actions and research at the same time. Through action research, researchers and practitioners can identify the problem and deal with

the difficulties and develop a solution related to the problem. Action research is a conceptual research framework in which researchers and practitioners are actively involved to make changes and improvements in the project.

3.2.1 Steps in Action Research

The action research consists of the following phase, as shown in Figure 3.

- Identification of Problem area
- Collection and organization of data
- Analysis of data
- Act on evidence
- Evaluate result

3.2.1.1 Problem Identification

Due to the novelty of the GSD, this study aims to explore the Requirement change management process for better understanding and improvement of the RCM process in global software development.

3.2.1.2 Gather Data

We conduct literature to study different research works and found out how much work has been done on these topics and which topics are still pending and how we can improve them. We have collected data through:

- IEEE Xplore Digital Library
- Google Scholar
- ACM Digital Library
- Springer
- Science Direct

3.2.1.3 Analysis of Data

After data collection, we have identified the factors that negatively impact in GSD environment and challenges that the GSD team faces with the help of literature. Also, we critically analyze the previous research to find the gaps and their limitations.

3.2.1.4 Act of Evidence

After a detailed analysis of data, we proposed a solution to overcome the change management problems in a globally distributed environment that are identified through the literature review.

3.2.1.5 Evaluate Result

We have conducted a case study to check our framework in a real-life context. After proposing a solution, we evaluate our framework with the help of a case study and survey method.

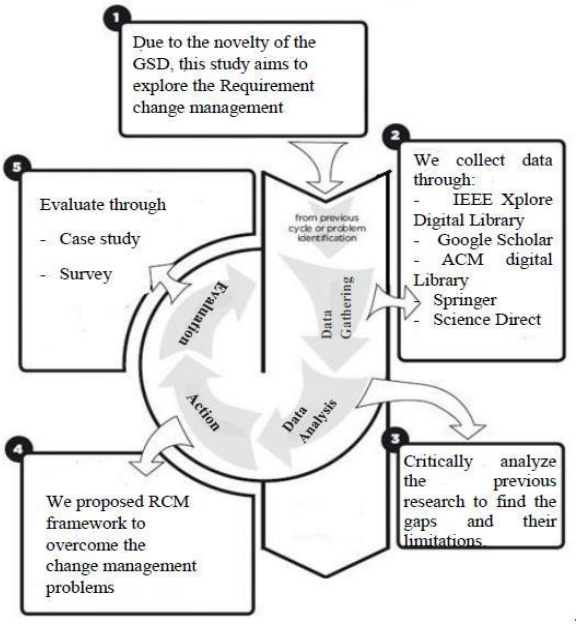


Figure 3 Action Research Cycle

3.3 Proposed Framework

We started our study by acquiring background knowledge about GSD and requirement change management in a GSD environment. In this research, we are investigating the problems faced during the change management process in the GSD environment. Many researchers pointed out that managing requirement change is still weak. Due to the novelty of the GSD, this study aims to explore the Requirement change management process for better understanding and improvement of the RCM process. In this section, we discussed our proposed approach briefly about software customization in global software development.

3.3.1 Revenue Cycle Management (RCM) Framework Description

In this framework, R means "Role" and S means "Site" to show the work distribution in multiple sites. RS1 means stakeholder role at site one who can initiate change. Similarly, RS2 means stakeholder at site 2. This framework is extendable to any number of development sites where RSN means any stakeholder at any site. We use CRF (Change Request Form) to gather change requirements from clients. SRS document (Software Requirement Specification) document and change request is the input of the RCM process. CRF document divide into three sections. 1st section is related to Submitter, who is used to submitting the change request. In the 2nd section, the Project manager fills this section after identifying an impact analysis of the change request, as shown in Figure 4.

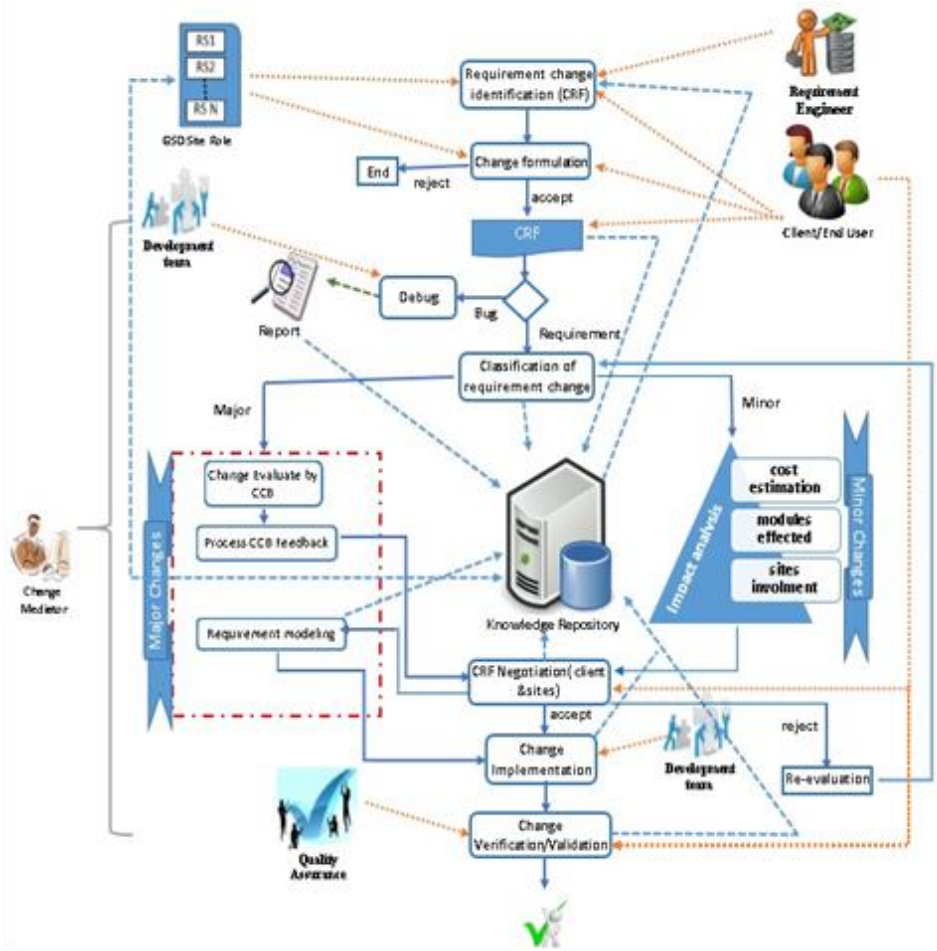


Figure 4 Revenue Cycle Management (RCM) Framework

3.3.1.1 Phase 1: Change Identification

The Requirement change management process initiates when clients or stakeholders at any site request for change. The client communicates with the stakeholder in any change due to a problem encounter or new requirement functionality. The client sends the change request through email or using the tool. The change request form (CRF) is generated with the help of the Change Mediator (CM) when the client requests any changes. A change Mediator (CM) is a person who is responsible for managing the client's request and is actively involved in the change management process. Requirement engineer gathers these requirements from clients or stakeholders.
In the next step, this change is discussed between the client and the requirement engineer/change mediator involved in the project. This discussion is conducted through an online video conference call that helps define, formulate and understand the identified change. If the change is in the scope of the project, then this change is further processed. But if the change is out of the project scope, it will be rejected. After change formulation, the change mediator assigns the ID to the change request. Then the client fills up the 1st section of CRF related to the Submitter that is individually used to submit the change request. This CRF is saved in Knowledge Repository. Now the change mediator identifies the type of request with the help of CRF. The request may be an error/bug, new requirement, or modify any requirement. In case of a bug/error, the development team directly

debugs this error and generates the debug report. This report is saved in a knowledge repository and also sent to the client. If identified, request related to the new requirement, modify any requirement and then go to the next step.

3.3.1.2 Phase 2: Type of Change Requirement

We divide the change request into two types: major changes requirement and minor changes requirement.
Major changes are the requirements that have a large impact on other requirements. These requirements are:

- **Add a New Requirements Relation**

In this type of requirement, add a new relation between two requirements Ri and Rj.

- **Delete Requirements Relation**

In this type of requirement, delete a relationship is between two requirements Ri and Rj.

- **Update Requirements Relation**

In this type of requirement, update a relation between two requirements Ri and Rj.

- **Add a New Requirement**

In this type of requirement, a new requirement R is added to the system.

- **Delete Requirement**

In this type of requirement, delete an existing requirement R in the system. Minor changes are the requirements that are related to updating any existing requirement. These requirements are:

- **Add Property to Requirement**

Add a new property ‘p’ in the existing requirement R.

- **Add Constraint to Property of Requirement**

Add a new constraint ‘c’ of the property ‘p’ of the requirement R.

- **Change Property of Requirement**

Change any property ‘p’ to ‘p1’ of the requirement R.

- **Change Constraint of Property of Requirement**

To change the constraint ‘c’ to the ‘c1’ of the property ‘p’ of the requirement R.

- **Delete Property of Requirement**

Delete property ‘p’ in the existing requirement R.

- **Delete Constraint of Property of Requirement**

Delete a constraint ‘c’ of the property ‘p’ of the requirement R.

In this step, Using CRF, the change mediator (CM) identifies the type of requirement. There are two main classifications of a change request. Major changes include new requirements and deleting any requirements. At the same time, the other classification minor change updates any requirement. We divide these requirements because major changes need more evolution than minor changes. So, the Change Control Board (CCB) evaluates the major changes in detail level, and a project manager or change mediator evaluates the minor changes. The Change Control Board (CCB) members are the project manager, program/site manager, senior developer, and client contact person. CCB does not involve every change request because CCB needs more time to evaluate each request. In minor requirements, the project manager evaluates the requirement related to any minor change on existing requirements. The project manager estimates costs and identifies which modules and sites are involved during the change. It helps to evaluate more than one requirement in the case of different categories at the same time.

3.3.1.3 Phase 3: Major Changes

If client/stakeholders need to add a new requirement or delete any requirements, the following steps are performed, shown in Figure 5.

Step 1

The change control board (CCB) evaluates this change request within a specific time under the policies and service level agreement. It conducts impact analysis to check the feasibility of the proposed change within the time, budget, and project scope. CCB estimates cost and required time, identifies the affected modules, and checks the feasibility of this requirement. CCB also identifies sites that implement the change request, estimate time zone, and other differences between those sites that are involved during this change request. CCB evaluates the major changes just because it needs more evaluation. After evaluation, every member of CCB gives their feedback related to the change requirement. If the CCB members are present at different locations, they will negotiate with each other through an online video conference call.

Step 2

If CCB members do not give their feedback within a specific time, then threshold time is increased to get feedback from CCB related to change. Members of all CCB have the same priority, so every CCB member must decide either to implement the change or not.

Step 3

Change mediator or project manager process the CCB feedback related to change request, which helps to finalize CRF. Then CRF section 2 and 3 is filled.

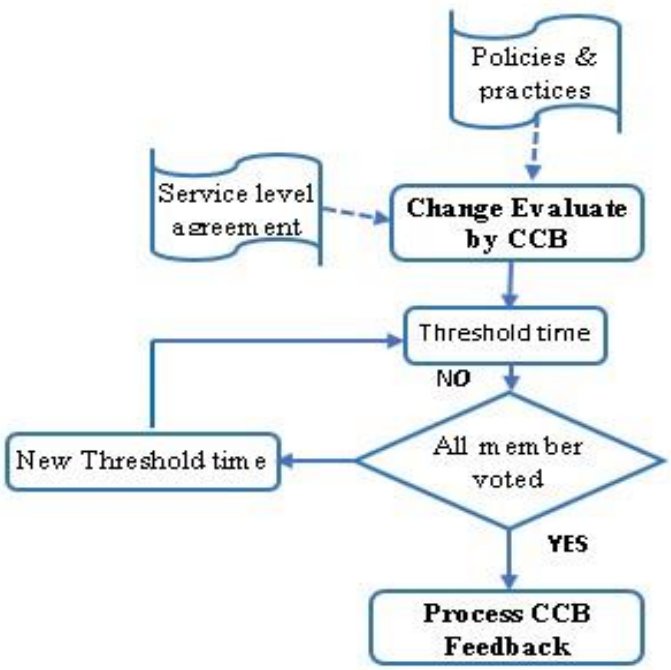


Figure 5 CCB Change Evaluation

3.3.1.4 Phase 4: Minor Changes

Suppose the client wants to modify any requirements like a change in interface. In that case, the Project Manager (PM)/Change Mediator (CM) conducts impact analysis to check the feasibility of the proposed change. In impact analysis, CM or PM estimates the possible cost for change implementation and identifies the affected modules and sites that implement the change request. After impact analysis, the change mediator fills the CRF.

3.3.1.5 Phase 5: Implementation and Verification

After evaluation of the change request, CRF negotiates with the client or stakeholders. When CRF finalizes between client and Change mediator, then change request is ready for implementation and saved this CRF in knowledge Repository. In the case of Major changes, requirement modeling is performed to generate requirement models with the help of SRS. The requirement model helps to identify which requirements are effective in delete or add new requirements. It also provides different ways to implement these change requirements effectively. With the help of the requirement model, the method engineer or senior developer selects the best way for implementation. With the help of the requirement model, the development team can easily understand the change request. Track the record of the required model in the knowledge repository for future use. In the next step, tasks are assigned to the relevant development teams at different GSD Sites. Development teams implement major and minor changes according to the client requirements. After implementation, the quality assurance team and clients validate and verify these requirements. After successful implementation, the final report is to be saved in the knowledge repository.

3.4 Evaluation of Our Framework

We have evaluated our proposed framework with the help of a case study. We have selected the "Finance Automation" project of inbox technology as a case study to check the effectiveness and efficiency of the proposed framework.

3.4.1 Case Study

A case study is a research method that is appropriate in real-life context for investigation and answers why and how questions when the investigator has slight control over the events. The investigation in a real-life company provides an opportunity to study where the relevant behaviors in the contemporary event were not manipulated. Some case study is used as a longitudinal depth examination of a case or event. The longitudinal examination is to collect data, analyze information, observe events, and systematically provide the results over a long period. For example, a researcher conducts a case study to examine the process of reading for one subject over some time. The case study is a unique method to observe any natural phenomenon. Unique means to examine only a small number of subjects of interest in detail. In quantitative analysis, the data pattern is observed at the macro level, but data is observed at the micro-level in the case study. The case study is the best approach to investigate the RCM process, which helps assess an activity, program, event, and process.
The case study research method is widely used in IS research, but it also allows the researcher to use several approaches to collect data and data analysis. So, we use the case study approach to evaluate our proposed work. Our case study uses different data collection sources that include questionnaires, documentation, observation, and interviews. The use of multiple sources of data collection allows an in-depth study of the phenomenon from different angles and increases the validity of data and research results. Also, the method of data analysis didn't include quantitative procedures. In the same way, we use the case study approach to evaluate our purposed work because it allows a researcher to use several methods for data collection and analysis, focus on the contemporary phenomenon with its real-life context, and answer the 'how' and 'why' type of research questions.

3.4.1.1 Interview Criteria

We have conducted interviews for data collection from the team members of the Pakistan site. The following team members were interviewed:

- Project Manager
- Development Team
- Program/Site Manager
- Requirement Engineer

The selection criteria of team members were; at least two years' experience in distributing development and over four years in software development projects. The project manager has three-year experience in a distributed environment and has over five years in software development projects. Similarly, other team members meet the selection criteria. In this project, many change requests come from the client and are written in English. The Project Manager provides

information about several people involved in this project and a description of the development and change management process. Quality Assurance Manager provides the information about quality assurance, and the development team helps to understand the development-related activities and issues.

3.4.1.2 Case study 1

We use the "Finance automation" project of Inbox Business Technology as a case study to evaluate our method to check and validate its usefulness. First, we present a description of the organization as shown in Table 1.

Table 1 Case Study Execution

Phases	Description	Roles and Artifacts
Phase 1: Change Identification	The client wants to Book Invoices, and Purchasing Requisition also performs automatically. The client sends the request. The requirement engineers and change mediator communicate with the client to understand this change through online video calls and then fill the CRF related to submitter and change request information. Change mediator assigns the CR.2028 id to change request. This change request is saved in the Knowledge Repository.	Roles: • Client, • Requirement Engineer, • Change Mediator Artifact: • Change Request Form (CRF), • Project Document or SRS
Phase 2: Type of Change Requirement	After the CRF, the change mediator identifies the type of change request. This change request is related to a major request because the client wants to add new functionality. The change mediator sends this request to CCB for change evaluation.	• Change Mediator
Phase 3: Major Changes	The CCB estimates the cost, the time required and finds which module and sites are involved. CCB conduct meeting through online video call because of Amsterdam's site manager and requirement engineer (Netherland). In this case, the Islamabad site develops this change, and financial and supply chain modules are affected. The CCB approves the change request, and the change mediator fills the CRF section related to the CCB decision. And negotiate with the client to inform the cost and time for the change implementation.	Roles: • Change mediator, • CCB (Project Manager, Site Manager, Senior Developer) • Client

		Artifact: • CRF
Phase 4: Implementation	After finalization, the requirement engineer develops the requirement graph with the help of project documentation; in this requirement graph, the Supply chain and financial module change according to the change request that helps the development team implement. Then Change mediator sends this change to the Islamabad site for development. After implementation, the quality Assurance team and client validate this change. Then the final report is saved in the Knowledge repository.	Roles: • Change mediator, • Development Team, • Client, • Quality Assurance Team, • Requirement Engineer. Artifact: • Project Document, • Final Report

3.4.1.2.1 The Software Organization

We have selected "Inbox Business Technology." It is a software consultancy firm that is located in Pakistan. Business models in the digital age need to be agile enough to have fluidity adapt to a shifting environment in which strategy and execution aim to hit a moving target. Inbox Business Technology gives its customer that agility with its digital services portfolio. The aim of the Inbox is "providing agility for the digital age." Inbox started its operation in Karachi and then made a branch office in Lahore, Islamabad, and Singapore. Now, the Inbox branch office is in Islamabad and has been completing several offshore projects. The major development work takes place in Islamabad, Lahore, and the head office in Karachi. Approximately 1000 employees are working in this company. The company provides different services to its clients. These services include:

- Digital security and intelligence
- Enterprise management services
- Citizen services and customer
- Cloud and converged systems

The client of inbox technology is:

- AkzoNobel
- Bank Alfalah
- Coca-Cola

- Dawood Hercules
- Dell
- Engro
- IBA
- ICI
- Jazz etc.

The major development work in Islamabad, Lahore, and head office in Karachi. The company deals with local and foreign clients to provide quality assurance, software development, and customer contact operations. We have selected Inbox technology because it develops GSD projects in Pakistan.

3.4.1.2.2 The Studied Projects

In this research work, projects are related to the development of "Finance Automation" for "XYZ" telecom industry client that is situated in Amsterdam (Netherland). Inbox Technology provides the services to automate their finance operation. The study is done in the Inbox technology on the Islamabad site, one of the three sites used for the software development project. On the Islamabad site, about 30 employees were involved in software development and support activities. The team consists of one Project Manager, one Site manager, 17 developers, and other team members include a requirement engineer and quality assurance manager. The other sites involved in software development are in Amsterdam (Netherland) on the client-side. The team in Amsterdam had one Site manager, three developers, two requirement engineers, and a requirement analyst, who was responsible for requirement analysis, requirement specification, initial software design, and partial module development for this project. Onsite senior developer and analyst were responsible for creating work in coordination with the client and the team at Islamabad.

3.4.1.2.3 Introduction of The Case Company

Client "XYZ" is a telecommunication company that was founded more than 20 years ago. The clients want to automate their finance operations to handle their financial performance that increases day by day. According to the client, the following features are necessary for the Finance Automation system.

- Enable the client to easily purchase different products, track purchase order and validate purchase order
- Enable the client to manage their stock
- Clients can use asset Tracking
- Client easily Maintain their account payable and receivable
- Enable clients to manage vendor contract
- The vendor tax management
- User-friendly environment

The major module for this application was:

a) Supply Chain Module

- Purchasing Module
- Inventory
- Asset Tracking

In this module, the client wants to maintain their purchase information. This information is related to purchase items, vendor information, and the cost of purchase items. They also want to manage their stock and keep track of their assets. If an employee or branch office needs any items, provide these items from stock and maintain their record.

b) Financial Module
- Account Payable
- Account Receivable
- General Ledger

In this module, the client wants to manage their accounts, maintain invoices, and keep the debit and credit records of these invoices.

c) Vendor Contracts module
- Vendor Contract Management
- Vendor Tax Management

The client wants to maintain vendor information, vendor contract management and also handle "how much the vendor and the company pay the percentage of the tax?"

3.4.1.2.4 Application of Our Method to The Case Study

In this section, we execute the change request of the "Finance Automation" project with the help of the proposed framework to check how the framework efficiently and effectively manages the change request in a distributed environment, as shown in Figure 6.

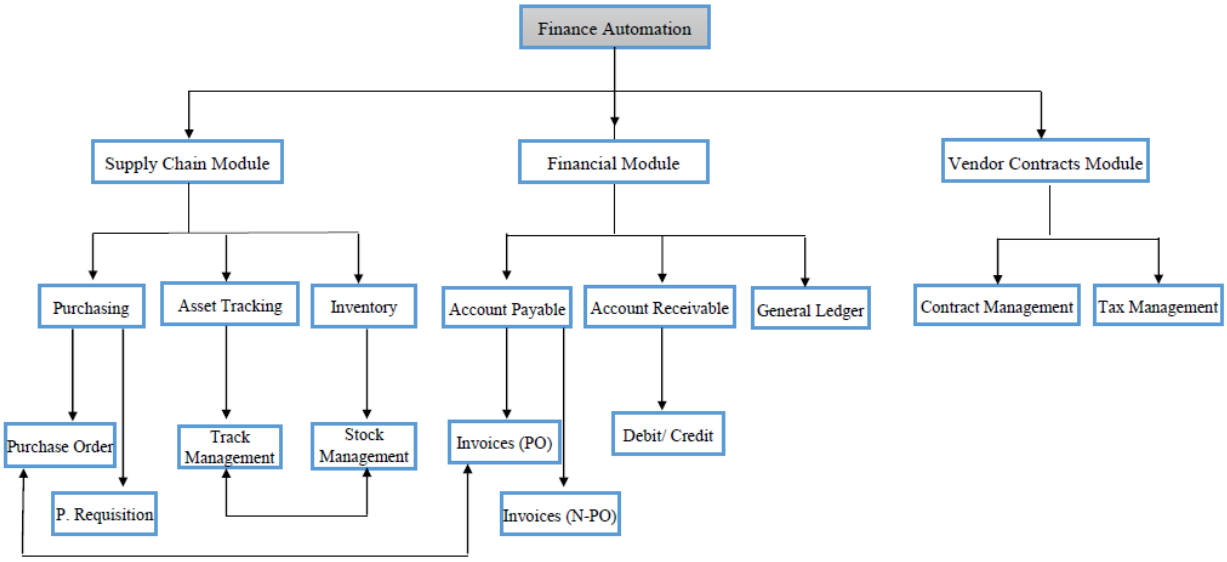


Figure 6 Requirement Graph for Finance Automation before making change

3.4.1.3 Case Study 2

For the last several years, Client XYZ involved in the merchandising business that sells its products in the Australian market in Australia. The customer need is very important for the client, and they want to develop a positive relationship with them. The client's motive is "to provide quality and reliable products at a reasonable price to abroad range of customer." In the merchandising field, the client holds a strong position because of their teamwork, clear focus, devotion, and well-defined marketing strategies. Now the client wants to expand their business through an online shopping system. According to the client, the following features are necessary for an online shopping system.

- Enable customer to purchase different products easily, track order, showing information about sellers' and secure a platform for payment
- Enable XYZ sells various products, manages customer and wholesale seller's information, and manages customer order and track record of the sold and available product.
- User-friendly environment

As thing clear from the client, now the client requests some related changes to the payment module. In the initial requirements, and only a credit card can use for payment, but no client wants to provide different options for customers. So, the client wants to add a PayPal feature in the payment module.

3.4.1.3.1 Application of Our Method

In this section, we execute the change request of the "Online Shopping System" project with the help of a proposed framework.

Phase 1: Change Identification
The client sends the change request. The requirement engineers and change mediator communicate with the client to understand this change through online video calls and then fill CRF related to submitter and change request information. Change mediator assigns the CR.001 id to change request. This change request is saved in the Knowledge Repository.

Phase 2: Type of Change Requirement
After the CRF, the change mediator identifies the type of change request. This change request is related to a minor request because this request has a minor impact on the payment module. The change mediator sends this request to the project manager for impact analysis.

Phase 3: Minor Changes
The Project manager estimates the cost, the time required and finds which module and sites are involved. After evaluation, the project manager approves this change request, and the change mediator fills the CRF section related to change evaluation. And negotiate with the client to inform the cost and time for the change implementation.

Phase 4: Implementation
After finalization, the change mediator sends this change to the Islamabad site for development. After implementation, the quality Assurance team and client validate this change. Then the final report is saved in the Knowledge repository.

4 Results and Discussion

We have evaluated our framework with the help of a survey and a case study. The survey provides a facility to represent many populations, a convenient way for data collection, and gives a precise result. The purpose of the survey is to get to know about different people's opinions or to know about their thought about our proposed framework by filling a questionnaire. The case studies are also used to evaluate our work. Its results are observed to study that if a certain kind of project is going to be built where stakeholders and development are dispersed worldwide, then how this framework becomes more useful.

4.1 Result of Case Study 1

We give our framework to the Inbox technology for the change management process. After using our proposed framework, participants have filled a questionnaire that is related to my framework. We prepared a questionnaire in survey crust and sent it to 23 participants. A total of 17 employees filled this questionnaire, shown in Table 2. The result of the questionnaire is shown in Figure 7 and Table 3.

Table 2 Number of Participant

Participant	No of Participant
Project Manager /Site Manager	2
Requirement Engineer	8
Developer	7
Total	17

Table 3 Result of Questionnaire

	Agree	Strongly Agree	Satisfactory	Disagree	Strongly Disagree	Satisfaction
Increase Client Involvement (AF-1)	8	2	3	4	0	76.47%
Effective Communication (AF-2)	5	2	4	4	2	64.71%
Sharing of Change Information (AF-3)	7	2	4	3	1	76.47%
Understanding of Change Requirement (AF-4)	6	1	5	4	1	70.59%

Improve Decision Making Process (AF-5)	7	2	4	3	1	76.47%
Reduce change Evaluation time (AF-6)	6	1	5	3	2	70.59%

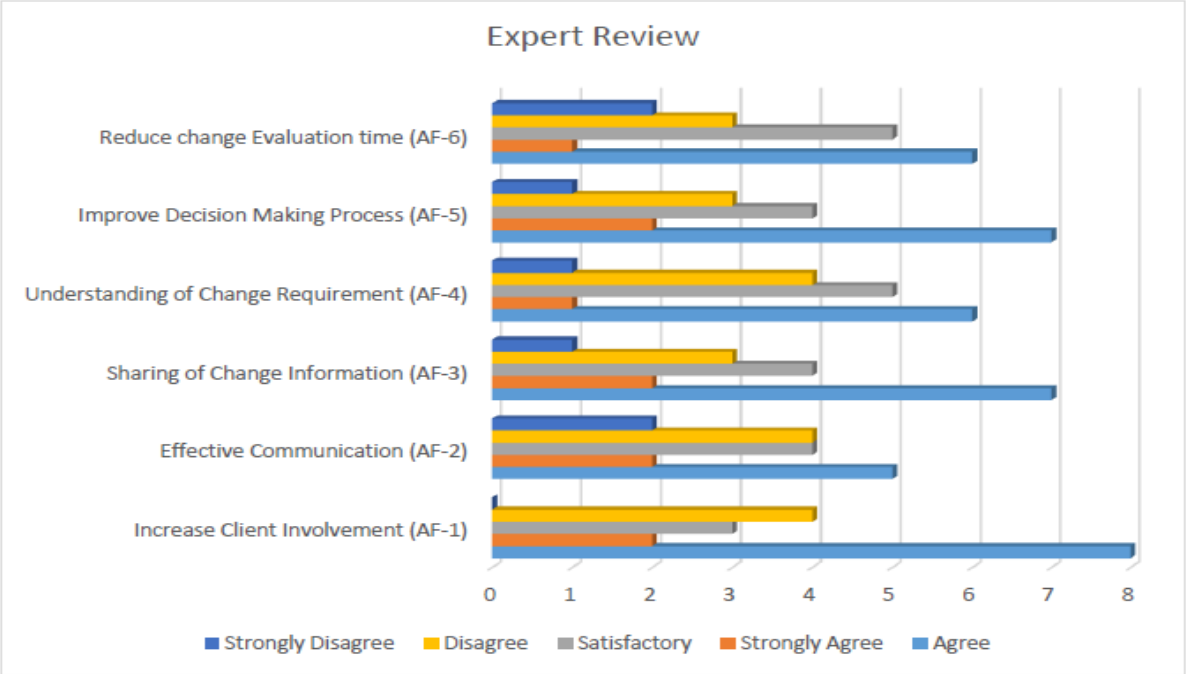


Figure 7 Expert Review

4.1.1 Increase Client Involvement and Communication

Previous research shows that inadequate communication creates many challenges in a distributed environment. In our framework, increase the communication between team and client. In this case, the client and requirement engineer are in Amsterdam (Netherland) whereas, the change mediator and other requirement engineers are in Pakistan. The client sends the email for the change request to the requirement engineer. Requirement engineer and change mediator communicate with a client through an online video conference call, which helps to understand the client's requirements. Throughout the change management process, clients are involved. At first, the client fills the Change Request form with the help of a change mediator. After implementation, the change request is verified by the client. An increase in client involvement reduces the chances of project failure.

4.1.2 Sharing of Change Information

Due to the distributed team, limited sharing of information or poor recording about requirement information can create conflict between the distributed team and create difficulty understanding

requirements. We can solve this problem through the knowledge repository. When client "XYZ" fills CRF for requirement changes, save it in the knowledge repository. Also, the decision about change requests is saved in the knowledge repository. All distributed teams, Islamabad team, and Netherlands team use knowledge repository to confuse requirements. Also, bugs, evaluations, implementation, and reports of the Quality Assurance Team are stored in the knowledge repository.

4.1.3 Understanding of Change Requirement

In a distributed environment, the development team faces difficulty understanding the change requirement. With the help of the requirement model, we overcome this problem. After decision-making, we generate a requirement model based on changes. The requirement model is a way to show a pictorial representation of change requirements that can help to understand the change requirement. Requirement Engineer generates a requirement model and sends it to the development team at Islamabad, Pakistan. The requirement model provides a better understanding of the change required, which helps in implementation.

4.1.4 Decision Making Process

Another gap identified in the literature is poor decision-making, which creates problems related to project budget and time. In our framework, we evaluate our change request through CCB and minor changes through change mediator. In this case, CCB evaluates the time difference between the Netherlands and Pakistan, also estimates change cost and time required for implementation. CCB also finds "supply chain" and "Financial Module" will be effective, and the Islamabad team implements this change.

4.1.5 Detail about Factors of Impact Analysis

In literature, there is a lack of detail about impact analysis factors. In our framework, we mention the complete detail of the impact analysis factors. Impact analysis factors include: cost estimation, estimated time, the time difference between GSD teams, affected module, affected requirements, and identify which GSD team is directly or indirectly affected. All of these factors help in the decision-making process.

4.2 Result of Case Study 2

Past researchers used student groups in a university environment to play the roles of stakeholders to conduct experiments on GSD. We create a virtual environment within Arid University. We select 19 students from the University Institute of Information Technology (UIIT), PMAS Arid University. Then we divided into five teams that are shown in Table 4. Basic information about the case study, roles and their responsibility, and detail about the proposed framework. We interviewed the student team after completing the change management process for both given methods. We asked questions related to the method's usefulness, frequency of conflict that occurred during the change management process, and time spent in the change management process. The student team rates the used method between 1 to 4. Where 1 for Not Useful, 2 for

Less Useful, 3 for Useful, and 4 for Very Useful. Finally analyzed the data obtained during an interview is shown in Table 5.

Table 4 Number of Participant

Team	No of Participant
Analyst	2
RE	3
Development Team in Pakistan	5
Development Team in China	4
Client	5

Table 5 Result of Case Study

		N	Proposed Framework	Other Framework
Usefulness	Mean	19	4.75	3.42
Frequency of conflict		19	1.5	1.84
Time spent	Time(m)	19	80	150

4.2.1 Usefulness

The result shows that our framework is more useful than other RCM methods. The reason behind why our framework is useful is the use of a knowledge repository, more client involvement that increases customization, and the use of a requirement graph.

4.2.2 Frequency of Conflict

We asked questions related to the frequency of conflicts. The student team rates our method between1 to 4. Where 1 for rarely, 2 for occasionally, 3 for less frequently, and 4 for very frequently. The result shown in the table, low “mean” value indicates less conflict has occurred.

4.2.3 Time Spent

The result shows that the student team spent 80 minutes during the change management process using the proposed framework.

4.3 Survey Questionnaire

We conducted a survey based on eight questions. With the help of this survey, we check the effectiveness of our framework. Our targeted audience has a different level of educations; they can be researchers or professionals.

4.3.1 Occupation of Population

Our targeted researchers work on GSD, and professionals are requirement engineers, developers, and project managers. In this survey, the total number of participants was 80, as shown in Table 6 and Figure 8.

Table 6 Number of Participant

Participant	No of Participant
Student	15
Requirement Engineer	8
Developer	11
Total	34

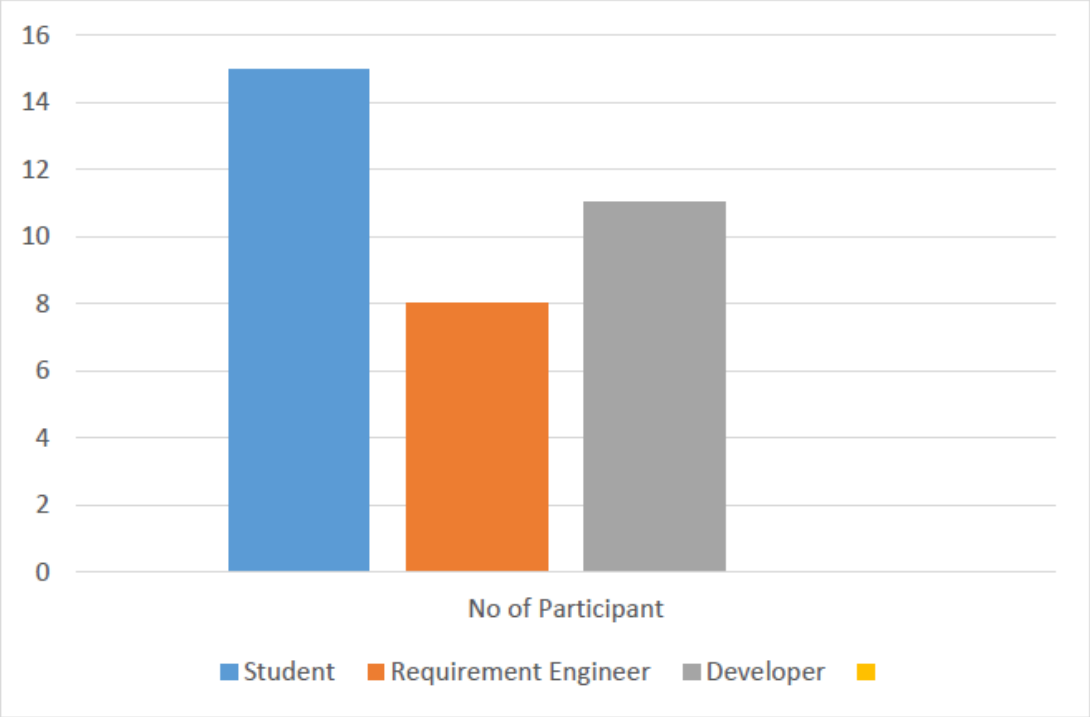


Figure 8 Occupation of Participant

4.3.2 Result of Questionnaire

Figure 9 shows the results of every question. It shows the views of the audience's thinking or feedback. The result of the survey shows that many participants were satisfied, whereas a few participants disagreed.

Q1. Is this framework reduce change evaluation time?
Q2. Is the requirement graph help to understand the change
Q3. Are requirements verify from a client?
Q4. Are change request and problems reports are properly

Q5. Is the quality assurance team help to improve the project quality?
Q6. Is this framework give the expected output?
Q7. Is the change required is properly evaluated?
Q8. Is this framework applying to every type of project?

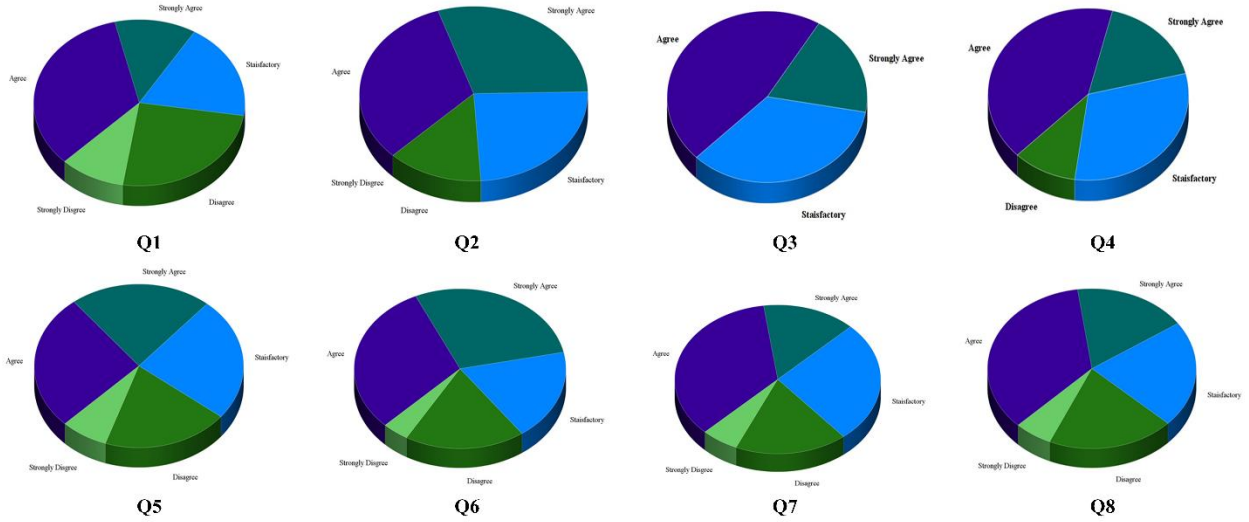


Figure 9 The evaluation of the survey questions

4.4 Working Prototype

The purpose of the proposed framework is to improve the customization process in global software development. After proposing a detailed framework, a tool is developed to manage change requirements. It can help to get a change request from the client and then evaluate this change. It also sends a notification to the client after acceptance of the change request.

4.4.1 User Registration

Figure 10 showcases when the user will click on to register the link button on the home screen of our application, and the user will enter information all click on the Register button.

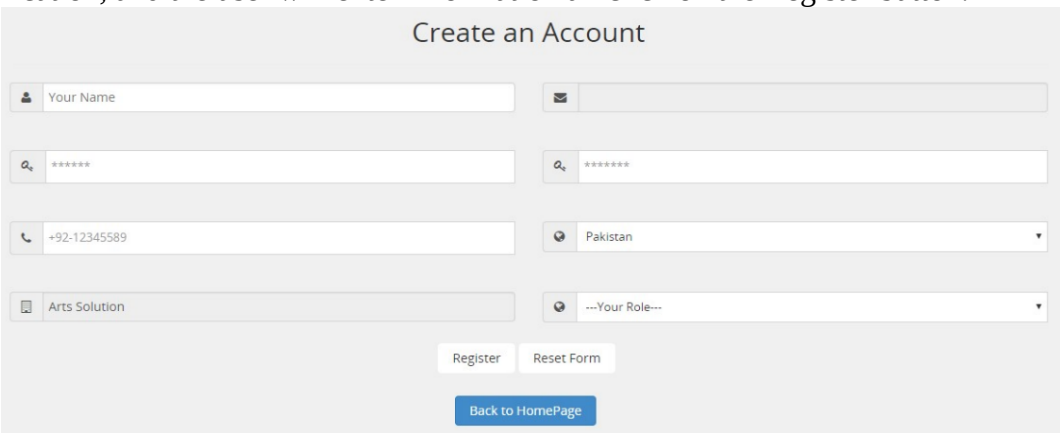


Figure 10 User Registration

4.4.2 User Login

Figure 11 shows when the user will click on to login link button on the home screen of our application, and the user will enter his email and password and click on the Sign-in button.

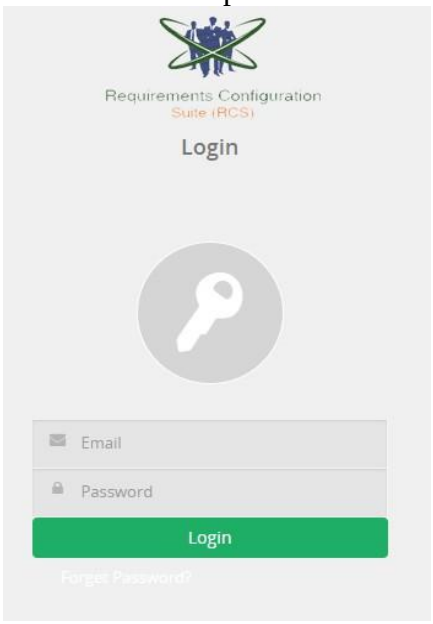


Figure 11 Login

4.4.3 Register Stakeholders

Figure 12 is the screen that appears in the case when Admin Will Invite stakeholders by entering their email addresses. Once the Invite Stakeholders button is clicked, an email invitation is sent to the stakeholder.

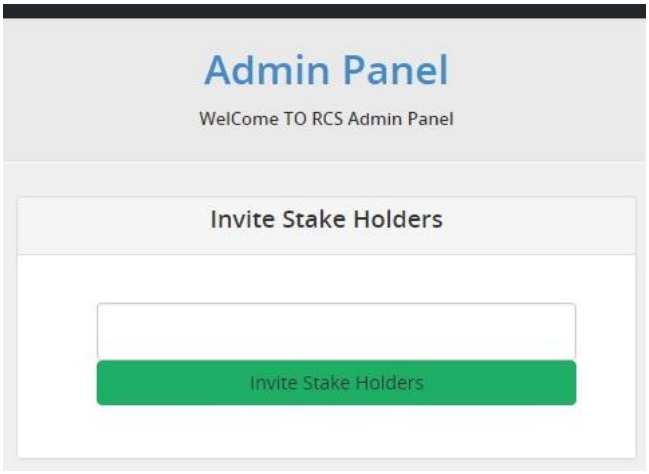


Figure 12 Invite Stakeholder

4.4.4 Add Project

Figure 13 is the screen that appears in the case when the admin will add a project by entering the appropriate information and click on Create Project Button. Admin Can Also Update Delete and View the project.

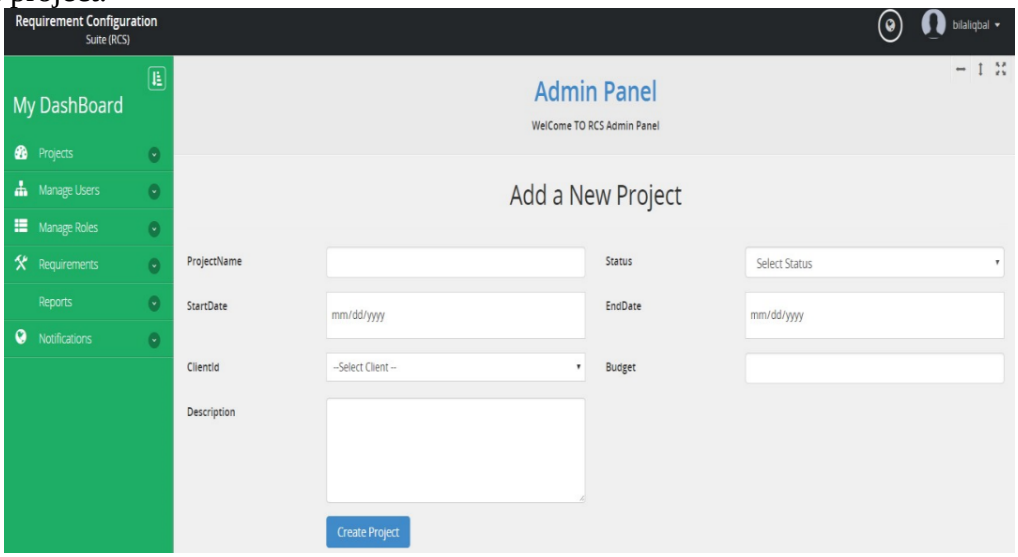


Figure 13 Add Project

4.4.5 Add Project Requirement

Figure 14 is the screen that appears in the case when the admin will add project Requirements by entering the appropriate information and click on Add Requirement Button.

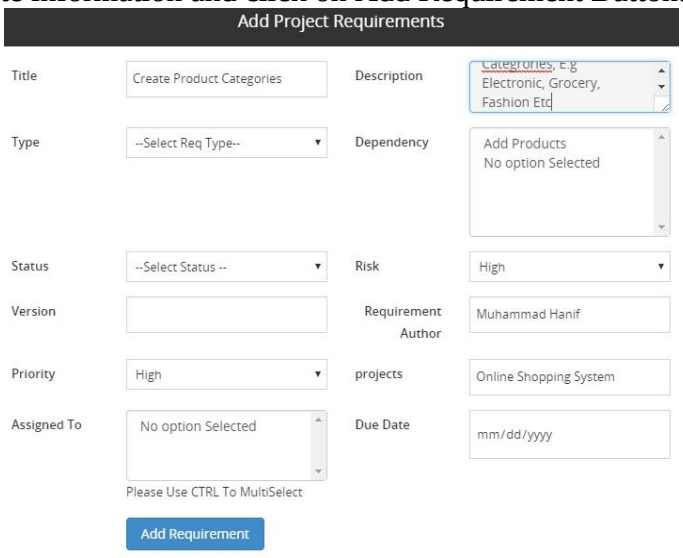


Figure 14 Add Project Requirement

4.4.6 View Project Requirements

Figure 15 is the screen that appears in the case when Admin Can View the project requirements by selecting the particular project. After the selection of the project, the project requirements will be displayed to the admin.

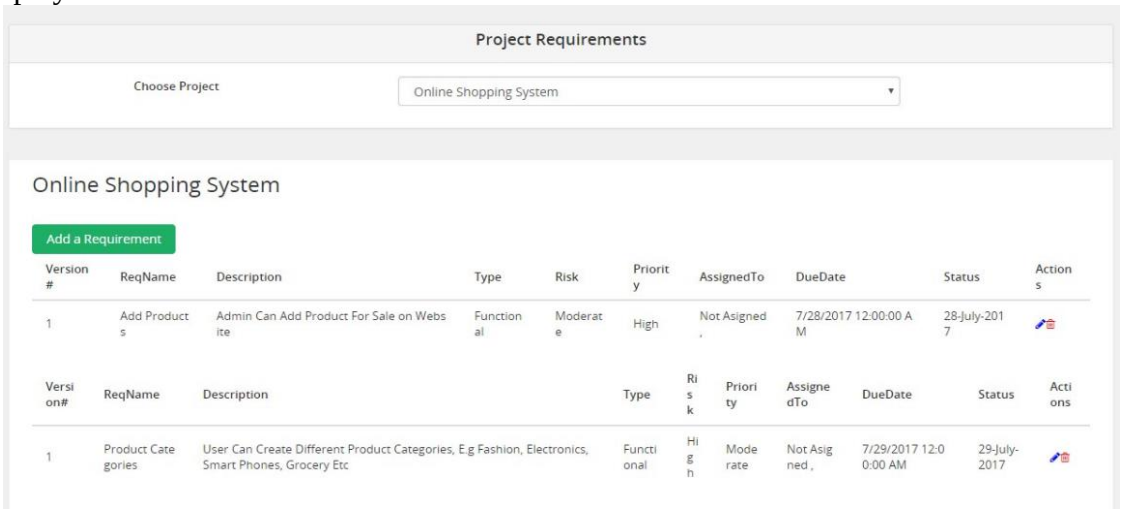


Figure 15 View Project Requirement

4.4.7 Perform Impact Analysis

Figure 16 and Figure 17 is the screen that appears in the case when the admin can perform Impact Analysis by selecting the particular requirement. The system will show the impact analysis result to the user.

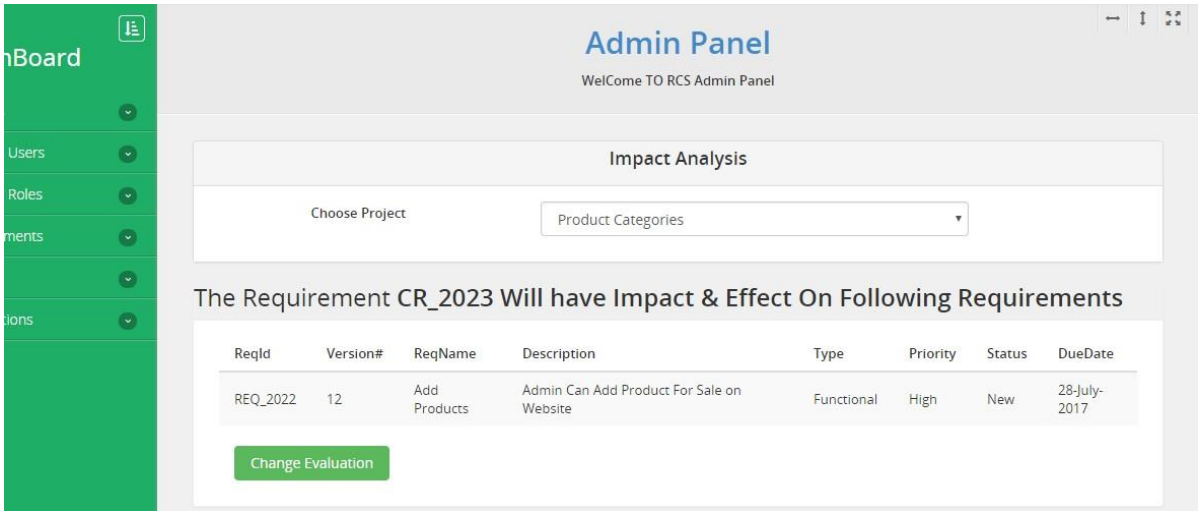


Figure 16 Impact Analysis

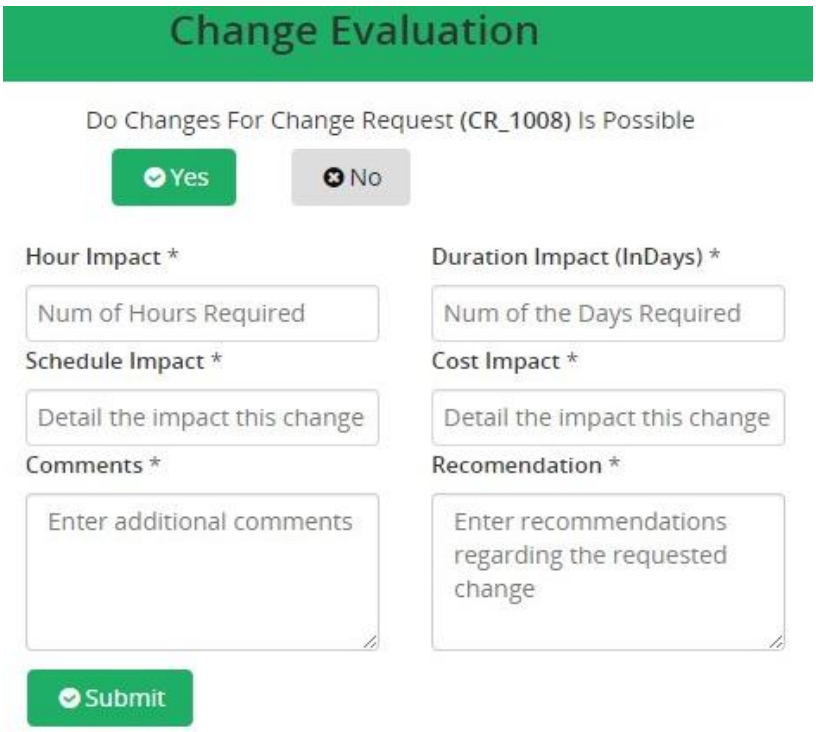


Figure 17 Change Evaluation

4.4.8 Generate Reports about Requirements

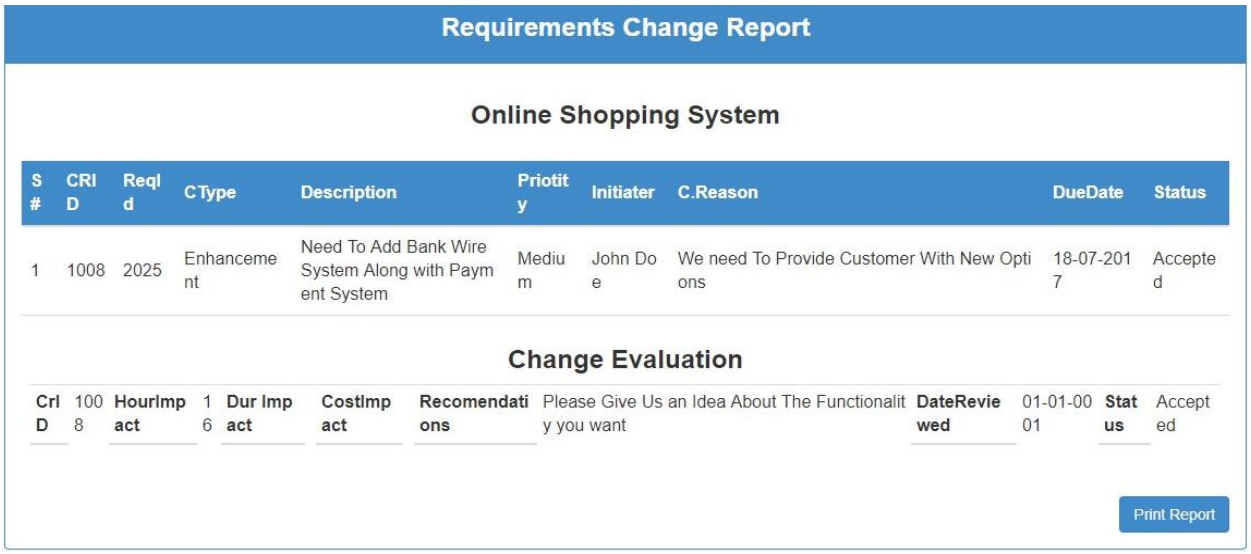
Figure 18 and Figure 19 is the screen that appears in the case when an admin wants to generate reports about project requirements or requirements change requests.

Requirements Report

Online Shopping System

S #	Req Id	ReqName	Description	Type	Risk	Priority	AssigneeTo	DueDate	Status
1	2022	Add Products	Admin Can Add Product For Sale on Website	Functional	High	High	Not Assigned ,	28-July-2017	New
S #	Req Id	ReqName	Description	Type	Risk	Priority	AssigneeTo	DueDate	Status
2	2023	Product Categories	User Can Create Different Product Categories, E.g Fashion, Electronics, Smart Phones, Grocery Etc	Functional	High	Moderate	Not Assigned ,	29-July-2017	New
S #	Req Id	ReqName	Description	Type	Risk	Priority	AssigneeTo	DueDate	Status
3	2024	Shopping Cart	Need a Shopping Card for Product Where User can Add item	Functional	Moderate	High	Not Assigned ,	31-July-2017	New
S #	Req Id	ReqName	Description	Type	Risk	Priority	AssigneeTo	DueDate	Status
4	2025	Add Payment Option	Need To Integrate A payment Gateway For Payments	Business	High	High	Not Assigned ,	03-August-2017	New

Figure 18 Requirement Report



Requirements Change Report

Online Shopping System

S #	CRID	Reqld	CType	Description	Priotity	Initiater	C.Reason	DueDate	Status
1	1008	2025	Enhancement	Need To Add Bank Wire System Along with Payment System	Medium	John Doe	We need To Provide Customer With New Options	18-07-2017	Accepted

Change Evaluation

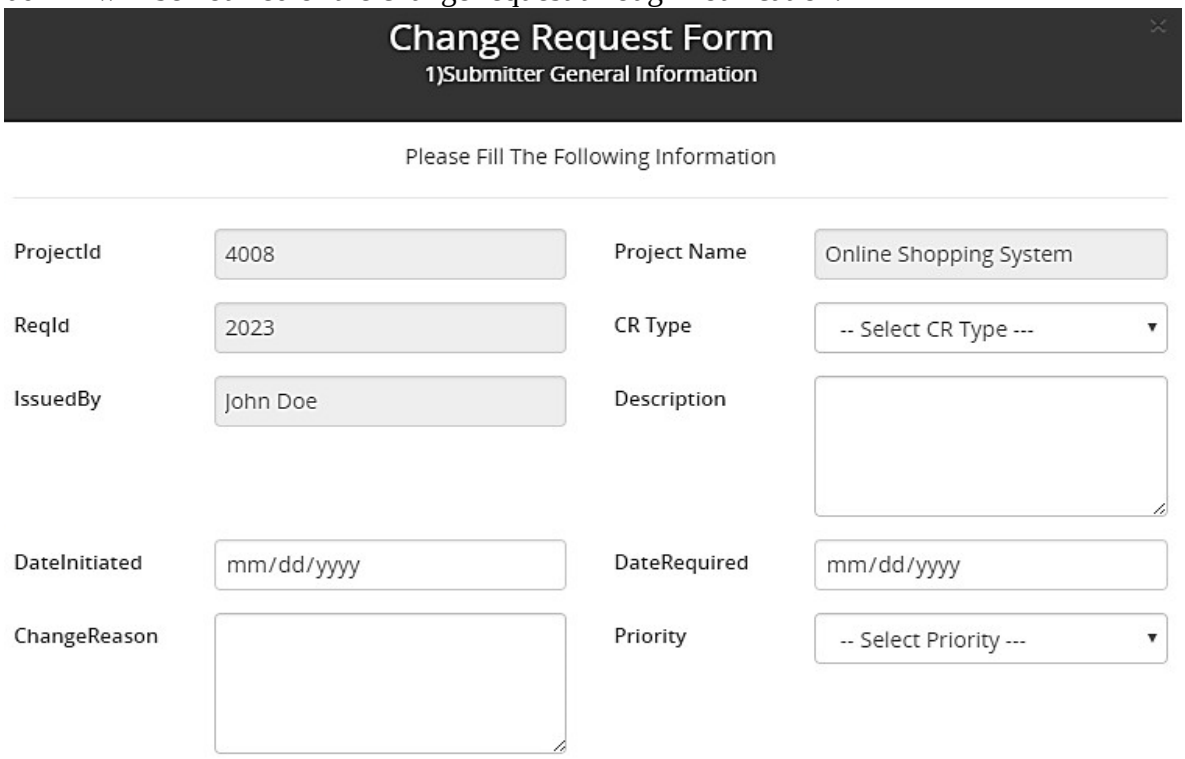
CrlD	1008	HourImpact	16	Dur Impact	CostImpact	Recomendations	Please Give Us an Idea About The Functionality you want	DateReviewed	01-01-0001	Status	Accepted

Print Report

Figure 19 Requirement Change Report

4.4.9 Initiate Change Request

Figure 20 is the screen that appears in the case when the client clicks on the Initiate Change Request button. The client will fill the CRF and submit the change request. Upon successful submission, an admin will be notified of the change request through notification.



Change Request Form

1)Submitter General Information

Please Fill The Following Information

ProjectId	4008	Project Name	Online Shopping System
ReqId	2023	CR Type	-- Select CR Type ---
IssuedBy	John Doe	Description	
DateInitiated	mm/dd/yyyy	DateRequired	mm/dd/yyyy
ChangeReason		Priority	-- Select Priority ---

Figure 20 Change Request Form

4.4.10 Change Request Status

Figure 21 and Figure 22 show the screen that appears when the client is notified about the change request he has initiated. Upon click on the detail's client will get detailed information about the change request, whether it is accepted or rejected.

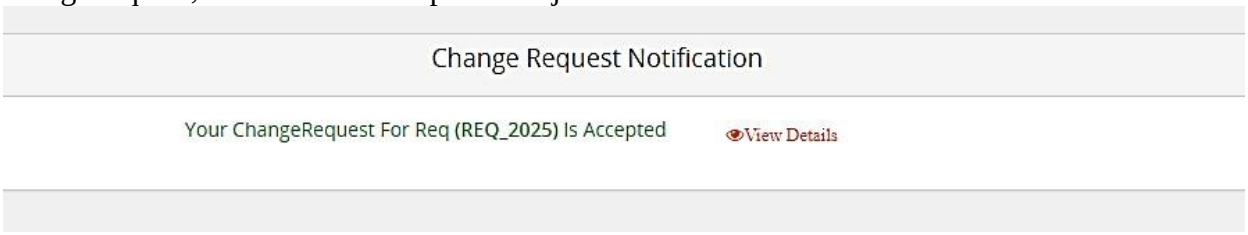


Figure 21 Notification

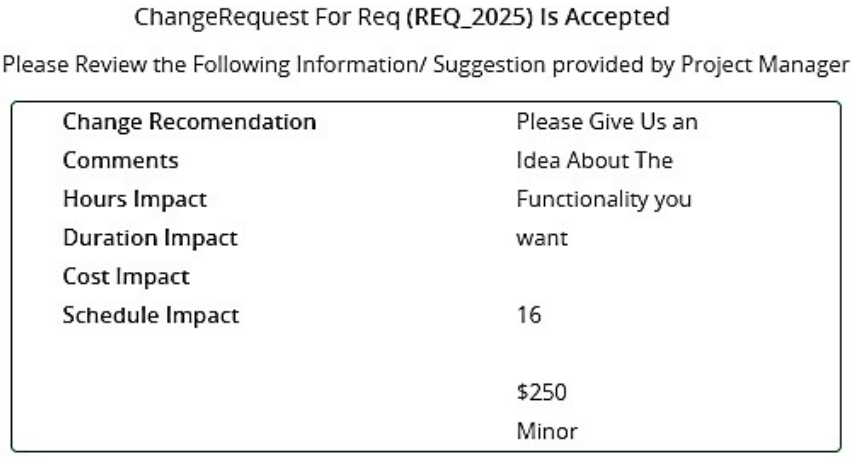


Figure 22 Change request detail

5 Conclusion

It is a difficult thing to manage the customer requirements locally and even more difficult in the case of a distributed environment. In this research, we worked on requirement change management for Global Software Development. We proposed a framework for requirement change management in GSD. In this framework, we increase the client involvement and share the information of requirements between distributed teams. We also use a knowledge repository to save the records of customers' requirements. In the proposed framework, we generate the Requirement model for the development team to understand change requests better. All reports generated in the required change process are saved in a knowledge repository for future use. We evaluate the proposed framework through a case study and survey questionnaires. We have applied the proposed framework on the "Inbox technology" project of "Finance Automation" to prove the applicability of our proposed approach. We can further improve the change management process for a distributed environment.

References

[1] P. Holtkamp, I. Lau, and J. M. Pawlowski, "How software development competences change in global settings—an explorative study," *Journal of Software: Evolution and Process*, vol. 27, no. 1, pp. 50-72, 2015.

[2] M. Marinho, A. Luna, and S. Beecham, "Global software development: practices for cultural differences," in *International Conference on Product-Focused Software Process Improvement*, 2018, pp. 299-317: Springer.

[3] S. Mahmood, M. Niazi, and A. Hussain, "Identifying the challenges for managing component-based development in global software development: Preliminary results," in *2015 Science and Information Conference (SAI)*, 2015, pp. 933-938: IEEE.

[4] R. Akbar and S. Safdar, "A short review of global software development (gsd) and latest software development trends," in *2015 International Conference on Computer, Communications, and Control Technology (I4CT)*, 2015, pp. 314-317: IEEE.

[5] S. Mahmood, S. Anwer, M. Niazi, M. Alshayeb, and I. Richardson, "Identifying the factors that influence task allocation in global software development: preliminary results," in *Proceedings of the 19th International Conference on Evaluation and Assessment in Software Engineering*, 2015, pp. 1-6.

[6] Y. Y. Jusoh, R. N. H. Nor, B. A. Mahmood, M. T. Wafeeq, M. A. Ali, and M. N. B. Jusoh, "Communication management in global software development projects," in *2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP)*, 2018, pp. 1-7: IEEE.

[7] A. Iftikhar, S. Musa, M. Alam, M. M. Su'ud, and S. M. Ali, "A survey of soft computing applications in global software development," in *2018 IEEE International Conference on Innovative Research and Development (ICIRD)*, 2018, pp. 1-4: IEEE.

[8] M. Yaseen, S. Baseer, S. Ali, and S. U. Khan, "Requirement implementation model (RIM) in the context of global software development," in *2015 International Conference on Information and Communication Technologies (ICICT)*, 2015, pp. 1-6: IEEE.

[9] E. Ó. Conchúir, P. J. Ågerfalk, H. H. Olsson, and B. Fitzgerald, "Global software development: where are the benefits?," *Communications of the ACM*, vol. 52, no. 8, pp. 127-131, 2009.

[10] E. Ó Conchúir, H. Holmstrom Olsson, P. J. Agerfalk, and B. Fitzgerald, "Global software development: never mind the problems–are there really any benefits?," 2006.

[11] M. Yaseen and M. Awan, "Practices for Effective Software Project Management in Global Software Development: A Systematic Literature Review," *vol*, vol. 177, pp. 1-6.

[12] M. Yaseen, S. W. Kamal, M. Bacha, A. Khan, and S. U. Zaman, "Frequent and Proper Negotiations as Success Factor in Global Software Development."

[13] M. Ilyas, S. U. Khan, and N. Rashid, "Empirical Validation of Software Integration Practices in Global Software Development," *SN Computer Science*, vol. 1, pp. 1-23, 2020.

[14] A. Altaf, U. Fatima, W. H. Butt, M. W. Anwar, and M. Hamdani, "A Systematic Literature Review on Factors Impacting Agile Adaptation in Global Software Development," in *Proceedings of the 2019 7th International Conference on Computer and Communications Management*, 2019, pp. 158-163.

[15] W. Haider, Y. Hafeez, S. Ali, M. Jawad, F. B. Ahmad, and M. N. Rafi, "Improving Requirement Prioritization and Traceability using Artificial Intelligence Technique for

Global Software Development," in *2019 22nd International Multitopic Conference (INMIC)*, 2019, pp. 1-8: IEEE.

[16] M. Yaseen and Z. Ali, "Success Factors during Requirements Implementation in Global Software Development: A Systematic Literature Review," *International Journal of Computer Science and Software Engineering*, vol. 8, no. 3, pp. 56-68, 2019.

[17] R. Vallon, B. J. da Silva Estacio, R. Prikladnicki, and T. Grechenig, "Systematic literature review on agile practices in global software development," *Information and Software Technology*, vol. 96, pp. 161-180, 2018.

[18] A. Hidayati, B. Purwandari, E. K. Budiardjo, and I. Solichah, "Global Software Development and Capability Maturity Model Integration: a Systematic Literature Review," in *2018 Third International Conference on Informatics and Computing (ICIC)*, 2018, pp. 1-6: IEEE.

[19] M. Ilyas and S. U. Khan, "Software integration in global software development: challenges for GSD vendors," *Journal of Software: Evolution and Process*, vol. 29, no. 8, p. e1875, 2017.

[20] M. Ilyas and S. U. Khan, "Software Integration Challenges for GSD Vendors: An Exploratory Study Using a Systematic Literature Review," *JCP*, vol. 12, no. 5, pp. 416-422, 2017.

[21] R. Tallat, R. M. A. Latif, M. Farhan, A. N. Zaheer, S. U. A. Shah, and F. Ijaz, "Empirical Evaluation of Visual Graph Analytic Techniques," in *2019 2nd International Conference on Communication, Computing and Digital systems (C-CODE)*, 2019, pp. 92-97: IEEE.

[22] R. Tallat, R. M. A. Latif, G. Ali, A. N. Zaheer, M. Farhan, and S. U. A. Shah, "Visualization and Analytics of Biological Data by Using Different Tools and Techniques," in *2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, 2019, pp. 291-303: IEEE.

[23] D. E. Adjepon-Yamoah, "Towards dependable change management and traceability for global software development," *arXiv preprint arXiv:1608.05981*, 2016.

[24] Humayun, M., & Jhanjhi, N. Z. (2019). Exploring the relationship between GSD, knowledge management, trust and collaboration. Journal of Engineering Science and Technology (JESTEC), 14(2), 820-843.

[25] Hamid, M. A., Hafeez, Y., Hamid, B., Humayun, M., & Jhanjhi, N. Z. (2020). Towards an effective approach for architectural knowledge management considering global software development. International Journal of Grid and Utility Computing, 11(6), 780-791.

[26] C. Diwaker et al., "A New Model for Predicting Component-Based Software Reliability Using Soft Computing," in IEEE Access, vol. 7, pp. 147191-147203, 2019, doi: 10.1109/ACCESS.2019.2946862.

[27] Waheed, S., Hamid, B., Jhanjhi, N. Z., Humayun, M., & Malik, N. A. (2019). Improving knowledge sharing in distributed software development. IJACSA) International Journal of Advanced Computer Science and Applications, 10(6).

[28] Saeed, S., Jhanjhi, N. Z., Naqvi, M., & Humayun, M. (2019). Analysis of Software Development Methodologies. International Journal of Computing and Digital Systems, 8(5), 446-460.

[29] Alsaade, F., Zaman, N., Hassan, M. F., & Abdullah, A. (2014). An Improved Software Development Process for Small and Medium Software Development Enterprises Based on Client's Perspective. Trends in Applied Sciences Research, 9(5), 254.