

# Change Management in a Global Distributed Environment (Process & Challenges)

<sup>1</sup>Saman Tariq, <sup>2</sup>Aasma Khan

<sup>1</sup>*Department of Software Engineering, University of Management and Technology (UMT), Sialkot, Pakistan*

<sup>2</sup>*Department of Software Engineering, Iqra National University (INU), Peshawar, Pakistan*

**Abstract- Context:** in a distributed environment changing requirement results in project failure, changes come from different sites by multi-site stakeholder involvement. by considering change requests from a different site and managing stakeholder issue change requests can handle efficiently. Here the objective is to find and manage the change process and suggest a solution to the stakeholder issues. **Methods:** we have done a literature review to find limitations in the change request process model and issue that stakeholders faced in a distributed environment. **Results:** We have identified activities missing in the change request process and change management process. We have also highlighted the role and responsibility for the activities to carry out a process of change management effectively. Our research also highlights stakeholder issues related to change management in a distributed environment and suggest an appropriate solution to the faced challenges. **Conclusion:** Our framework process not only defines a good balance change process but also identifies change activity in each process, role and responsibility to carry out the process and artifact efficiently. But there is a need to find more stakeholder issues related to managing the change request and their appropriate solution also need to find process activities in detail so that there must be no misunderstanding related to task assignment.

**Keywords:** Requirement Change, Change Management Process, Change Process, Global Software Development (GSD), Distributed Teams, Distributed Software Development, Global Challenges, Framework for Change Management.

## I. INTRODUCTION

The process of understanding, managing, locating, controlling and authenticating the changes in the requirements are basically called as Requirement Change Management (RCM). For the success of the globally distributed projects, it is very necessary to achieve and fulfill the needs of the stakeholders, that's why making changes in the requirements is a big challenge. Global Software Development (GSD) is gaining more attraction to the software industry throughout the world. During the 1990s, the differences in development cost, the limited pool of workforce, the necessity to

---

<sup>1</sup>Samantariq786@gmail.com

<sup>2</sup>Leo199343@live.com

get closer to the customer and round the clock development, force the organizations to outsource their projects [1].

Change may refer to add new functionality in a system or modification of an existing system. To change the system, the system must be capable to add new functionality in the system or can accept required modification [2]. So the system must be designed in such a way that provides flexibility to the developer for change (Saleh, Mark, and Jordan; Cardinet).one term use for change is ECM (Engineering Change Management) involve a change in a component of a system or change in a whole system where change analyze, accept, change impact are identified and then it passes through to CCB(change control board) for acceptance and in last execution of change is performed [3]. It's not preferable to freeze the requirement at the start of the development phase. The changing requirements are not problematic; the real problem is how to manage these changes [4], [5].

The primary objective of this paper is to study the causes of change in a distributed environment and Input that gives rise to change request by conducting a systematic literature review. This paper also highlights challenges and their solution that stakeholders face in a distributed environment when managing the change request.

This research paper addresses the following questions:

- 1) The process of change request comes in a distributed environment?
- 2) Challenges involved for acceptance of change requests on the multi-site stakeholder?
- 3) A solution suggested for the discussed challenges in a distributed environment.

The rest of the paper is organized in the following section. Section 2 describes the background and motivation of Requirement Change Management. Section 3 describes the related work based on the requirement change framework. Section 4 describes the proposed framework steps for change requests and management processes. Section 5 describes the results and discussion based on our suggested work also highlight solutions to the proposed literature issues related to stakeholders in a global environment and in last conclude our work.

## II. LITERATURE REVIEW

Global software development plays an important role in increasing productivity where people of different time zone are actively involved. Some important GSD benefits are reduction in cost, skilled force, increase productivity and utilization of 24 hours working across the global environment.

Different challenges are involved in GSD due to differences in geographical, cultural, and temporal distances.

Among those communication issues are on the top of the list, these communication issues give rise to change request [6]. Change request comes due to some external and internal factors [7]. External factors involved like a bug report, changing business needs functionality and technical lacking of customer background. Internal factors involved like financial resources, equipment, copyright, and human resources. This change request arises because of geographical and socio-cultural differences in a globally distributed environment. Requirement change management (RCM) is not a simple process; it's a complex process that involves valid requested requirements, detailed planning, the reasoning behind the change, the influence of change to the existing requirement, scheduling cost estimation, propagation of change, proper decision making, approval from Change Control Board (CCB), verification and validation of the change request and change outcome [8].

The different models proposed in the literature for Requirement Change Management (RCM). According to capability maturity model change must be tracked and store in a database after that it must be analyses with the existing requirement to check the feasibility of requested change [9], [10]. Another model proposed for RCM which consist of three steps, determine the changes, approval for acceptance of that change and perform that change [11]. In order to understand the requirement need-based requirement model is proposed (NRM). This model is based on 4w (Who, what, why, where).in this model V&V method (verification and validation) used for continuously viewing of change requests in the system [12]. When multiple sites are involved change will propagate in such a way that change in design at one site may reflect the change in code at another site, this may reflect a delay in work because of cross-site communication issues [13]. One method for managing requirement changes in to establish a repository for a list of change requirements then use the Requirement Traceability Matrix (RTM) for linking it to the other requirement then communicate it to the stakeholder and in the last manage the change[14]. Table 2.1 explained below provides a brief introduction about the models that are commonly used for the required change management process. All models have some advantages as well as some drawbacks as none of them completely covered the change activity.

*Table 2. 1: Comparison of different Change Models*

<b>RCM Model</b>	<b>Objective</b>	<b>Input</b>	<b>Pros</b>	<b>Cons</b>	<b>Covered change activity</b>	<b>Ref.</b>
Dean Leffingwell and Widrig model	Actors provided by a framework to carry out their tasks.	Customer, developer, end-user, CCB & Maintenance organization	Clearly defined entry and exit condition	No verification performed after change implementation	Overall change management	[9], [15], [1]
Olsen's Model	Treat all activity that is involved in a life cycle as change	Customer, developer & CCB	Life cycle independent so can be applied for maintenance	Change assignment to a resource is missing	All activity involved in the development	[1], [9], [15], [16]
V-like model	This model describes the technical activities for implementing a change.	Maintenance organization	Solutions are obvious and low cost.	Estimation of the cost required to implement the change is missing as well allocation of resources	Activity related to technical modification	[1][9] [15]
Ince's Model	The model focuses on software configuration	Customer, developer, and CCB	The same process followed for all change Request	No decision making about change input and output	Change management activity	[1][9] [15]
Spiral model	The process of change management is divided into cycle also solve & record Changes	No input	Easy to use	No testing and verification performed to check that the change is met or not	Same activity of V-model	[1][9] [15]
NRM model	Help analysts to understand accurately	Customer and developer	Efficiently manages continuous changes in the system requirement.	Decision making is missing about change whether the change should accept or reject	Verification and validation	[1], [9], [15]
AMES model	Help in decision making of change activity	Developers, CCB, maintenance organization	This model manages change throughout the life cycle of software	It does not cover Configuration management.	Decision making and technical modification	[1], [9], [15]

S.A. Bohner Model	This model is used for Change implementation, Understanding the problems, For Solution specification And Regression testing	No actors involved for input	This model is silent about the impact analysis, that's why it finds out the scope of the change and it also helps in improved budget estimation	Provide no information about activity acceptance and rejection as well as its documentation. In short actors and key artifacts are completely missing.	Scope of change and budget estimation after the change	[1], [9], [15]
CHAM Model	In the CHAM model impact of the change is estimated based on cost	Customer developer and end-user	Good in estimating the impact on the resources and assess whether change detail is enough or its impact on the effort.	This model does not perform any testing activities e.g. regression testing, acceptance testing, etc.	Checks the change impacts	[1], [9], [15]
S.A Ajila model	This model analysis the impact of the change on the functionality	No actors involved for input	Good in checking the impact of change on functionality	Negotiation after the change is missing and does not analyze the impact on cost, effort, etc.	Checks the change impact on the functionality	[1], [9], [15]
Simon lock model	This model analysis the change at the initial stage and hence saves the cost of software	Customer, developer, and end-user	Save the software cost as change can be stopped at the initial stage	This model is missing the activity at the initial stage that can help in understanding the change	Saves the change cost	[1], [9], [15]

While managing the change requests stakeholders face different problems in a distributed environment. Some of the common issues related to stakeholders are due temporal, geographical [17] and communication distances. To manage the change request effectively, we need to understand the issues in a distributed environment. Table 2.2 explained below highlights issues related to stakeholders and managing the change request in a distributed environment.

*Table 2. 2: Stakeholder issue in GSD related to RCM*

<b>Problem</b>	<b>Description</b>	<b>Ref.</b>
Pb1	Misunderstandings, duplicate change requests & vague requirements.	[18]
Pb2	Lack of communication medium.	[18]
Pb3	Lack of changing awareness among all stakeholder	[19]
Pb4	No standard format followed at different sites.	[18]
Pb5	Different processes followed at different sites.	[1]
Pb6	No uniform tool usage by the development team.	[1]
Pb7	Lack of awareness of change needs.	[19]
Pb8	No sharing of updated change documents.	[1]
Pb9	Lack of understanding of change propagation	[19]
Pb10	Impact of new requirements on existing ones.	[19]
Pb11	Co-ordination issues due to different temporal distances.	[1]
Pb12	Artifacts are produced in different formats.	[15]
Pb13	Lack of decision making during traceability among artifacts.	[15]
Pb14	Unassigned role and responsibility of change request activity.	[15]
Pb15	Un updated and wrong selection of tool.	[1], [20]
Pb16	Missing and unawareness of change validation at the end of the development phase.	[15]
Pb17	Unawareness of Regression testing after change implementation.	[15]
Pb18	No verification of implemented changes to check either the specifications are met or not.	[16]
Pb19	Unable to perform the change configuration process by the stakeholder.	[15], [20]
Pb20	Inexperience selection of cost and schedule estimation.	[21]

### III. RELATED WORK

Different frameworks are introduced for handling the change process but the flow of work in them is not managed effectively. A global framework introduced by the author Niazi in which he improved the process of the requirement for a distributed environment [22]. Kumar & Kumar proposed the process of change according to types, relationship and interrelated activities of change. But they have not explained the communication of repository according to global context [23]. Sinha proposed EGRET (Eclipse-based requirement tool) for the understanding of the change request [21] and managing those changes in the global environment [24]. Lai proposed a repository

for change requirement but requirement change activities are missing as well process for managing change requirement is not complete. In case of rejection of requirement from the shared repository, no alternate scenario is described [14]. Hussain proposed the process of a change request from initiation to a selection of change requests, analyze the change and the most important part in a change is that of the impact of change on the already existing requirement. But unable to explain the validation and verification of change after change implementation to check that the required specifications are met or not [25]. None of them explained the change process from the verification perspective and none of them assigned the role and responsibility to the activities involved in the change process.

#### IV. METHODOLOGY

As the objective of this paper is to study the change management in the distributed environment [26] and to find the processes and challenges related to it and reasons that are giving rise to change requests so for this we conducted a literature review. This paper also highlights problems that stakeholders and interacting sites faced in a distributed environment during managing the change request. This research paper addresses the following questions:

- 1) The process of change request comes in a distributed environment?
- 2) Challenges involved for acceptance of change requests on the multi-site stakeholder?
- 3) Did the solution suggest the discussed challenges in a distributed environment?

To address the above questions we conducted a literature review and found different models that are used to manage changes in the change request process in these models we have identified processes used to manage the change requests, activities involved in the change process and the arrival of change requests in a distributed environment. We have also searched for the main problems the stakeholders in a distributed environment and interacting sites have faced and then integrated them into a table and after proposing a framework we have provided a solution for all the mentioned problems.

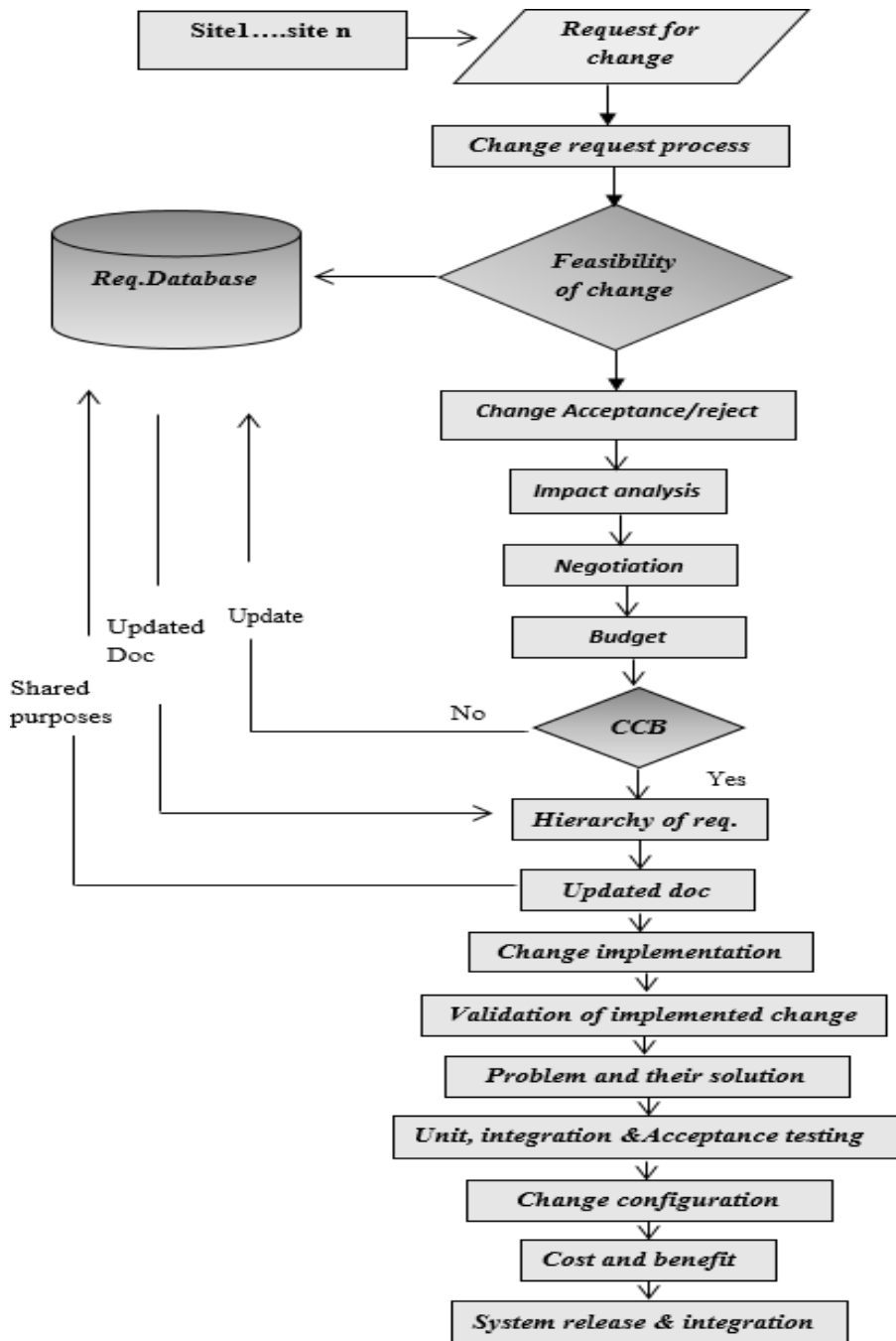


Figure 1: Framework steps for Change Management

## V. RESULTS AND DISCUSSIONS

The common elements in the above mention models are that none of them have explained the change configuration and change verification process after the change implementation. Models describe the process of change but provide no information about the challenges that the stakeholders faced when managing the change request. Same case with the framework proposed for requirement change management. As a framework for Requirement change management provide information about the understanding of change requests and actively involved in the change request process but they provide no information about the role and responsibility to carry out those activities and have little focus on the resulting artifact.

**Table 2.3: Change process, role & responsibility with their key artifacts**

Process	Activity	Description	Role and Responsibility	Artifact
Change request	identify, validate, analyze control, action, and close	Title, Requested by, Reason, type of change, priority & outcome of change	Project manager	Requirement change document
Change Acceptance/reject	initiation, analysis, approve/reject and funding	check the impact and priority of change and also checking the funds for the change	project manager CCB	updated requirement change document
Impact analysis [27]	understand change, identify document & identify a task	impact on requirement functionality, impact on cost and schedule and impact on the stakeholder	CCB & developer	impact analysis report
Negotiation	the objective of negotiation, prepare, conduct and review negotiation	acceptable agreement between client and development team	end-user, contractor and project manager	negotiation record, change order checklist, cost estimates

Budget allocation	create a budget, budget approval, allocate budget, utilize budget and reconcile budget	feasibility of change according to funds and resources provided	finance manager, sales manager, and marketing managers	Budget report
Change implementation	implement the change and monitor them	if desired output not met back out the changes	developer & testers	report on change implementation
Validation of implemented change	develop control & procedure, clarify and review system	check deliverable met the desired criteria or not also review system for validation accuracy	testers and developers	validation and lesson learned report
Unit, integration & Acceptance testing	create a test script, execute test cases and evaluate them	test cases, test criteria, test coverage, and type of testing	PM, test lead, test team, analyst and customer	test script, test log & test report
Change configuration	plan change configuration, deliver them, monitor configuration status and manage CR	baseline change request management according to workspaces	configuration manager, CCB & integrator	change Request, workspaces and configuration management plan
Cost and benefit analysis	define and identify cost elements and cost drivers, analyze risks and generate results	in this, we use different methods and technique for estimation and accuracy of the result	project manager, finance manager maintenance & engineers	cost and benefit report
System release planning	identify bugs, user guidance, installation manuals, and required documentation	exit criteria of a product	project manager, testers and marketing team	reports, manuals, and documentation

Frameworks provide information about the process to carry out the task effectively but they have less focus on the challenges [28] involved by the stakeholder during managing the change request.

Here we have proposed solutions for the above-mentioned problems that are faced by stakeholders in distributed environments .mainly we have to overcome the temporal, geographical and communication distances. The table below is giving a possible solution to overcome these problems.

**Table 2. 4: Recommended Solutions for Stakeholder problems in GSD**

<b>Problem</b>	<b>Solution</b>	<b>Reference</b>
p1	Keep expecting and do planning for requirements that are going to change throughout the development process, prioritize requirement or reprioritize if required according to needs.	[29], [30], [8]
p2	to avoid this problem we need to know our audience and need to communicate to them in ways that are easy for them to understand e.g. using diagrams, storyboards, prototypes, etc. and then get feedback from them	[29], [30], [8], [10]
pb3	to solve this problem stakeholders need to be informed as soon as a change occurs in a way that, get input from them to prioritize it and identifying the reason to prioritize it	[29], [30], [8]
pb4	standards should be followed throughout the interacting sites that will help to complete tasks effectively and on time	[29], [30], [8]
pb5	to solve this problem inform all the interacting sites about the process to follow this is going to save time and can make the interaction better	[29], [30], [8]
pb6	This is just like using the same processes among different sites so is to use the same tools among distributed interacting sites so this is also going to save time as there will then be no need to translate the information or data exchanged.	[29],[30], [8]
pb7	For this, there is a need to identify all the key stakeholders including the end-user as well, and there should be a skilled facilitator as well to keep all the activities on track	[29], [8]
pb8	For this, there should be one representative from each group of stakeholders how is going to keep them aware on regular basis and make it easy for all the involved stakeholders to share their feedback so that everyone gets updated.	[29], [30], [8]
pb9	For this, there is a need to draw feedback from all of your stakeholder representatives and keep sharing it among the interacting sites and stakeholders	[29], [30], [8]
pb10	For this, we need to test requirement changes at initial stages so that it can cost us less which is otherwise costly to tackle them in later stages	[29], [30], [8]
pb11	For this, there is a need to follow standards, uniform tools, processes and to keep stakeholders informed regularly so, in this way, coordination problems should be solved to some extent.	[29], [30], [8]

pb12	For this one standard should be produced and then it should be circulated to all the distributed interacting sites and stakeholders and try to keep all related artifacts in a central repository so that it is accessible by the team.	[29], [30], [8]
pb13	For this consistency should be maintained, creating manuals and then maintaining these manuals can also help to solve this problem	[8]
pb14	There should be a facilitator who can keep track of every activity, representatives from each group of stakeholders so that managing change becomes easy	[8]
pb15	For this try to invest in tools that efficiently gather and distribute requirements change information among different sites and stakeholders	[8]
pb16	Validation should be included in the followed standards so that all sites and stakeholders are aware of this activity to minimize errors and get better results	[8]
pb17	For this development team should provide advanced tool support and simulators and emulators should be provided	[8]
pb18	For this verification should also be included in the followed standards so that all sites and stakeholders are aware of this activity to minimize errors and get better results	[8]
pb19	Change configuration process should be followed and try to keep all stakeholders informed for this automatic notification can help to pass on this information	[8]
pb20	For this, there is a need for an experienced and skilled employee in cost and schedule estimation	[8]

## VI. CONCLUSION AND FUTURE RESEARCH DIRECTION

Different change models and frameworks proposed to manage the change process effectively and to understand the challenges involved by the stakeholder due to differences in geographical and temporal distances. Change models do not focus on verification of change requests and change configuration after implementation of a change request. The global requirement change framework deals with the change process effectively but unable to assign role and responsibility to the change process activities also unable to discuss challenges involved by the stakeholder due to geographical and communication distances. Our framework process not only defines a good balance change process but also identifies change activity in each process, role and responsibility to carry out the

process and artifact as well. Also, provide the solution to the challenges that are faced by the stakeholder in a distributed environment while managing the change request.

#### REFERENCES

- [1] N. M. Minhas and A. Zulfikar, "An Improved Framework for Requirement Change Management in Global Software Development," *J. Softw. Eng. Appl.*, vol. 7, no. August, pp. 779–790, 2014.
- [2] E. C. Y. Koh, N. H. M. Caldwell, and P. J. Clarkson, "A technique to assess the changeability of complex engineering systems," *J. Eng. Des.*, vol. 24, no. 7, pp. 477–498, 2013.
- [3] V. Kocar and A. Akgunduz, "ADVICE: A virtual environment for Engineering Change Management," *Comput. Ind.*, vol. 61, no. 1, pp. 15–28, 2010.
- [4] S. McGee and D. Greer, "A software requirements change source taxonomy," *4th Int. Conf. Softw. Eng. Adv. ICSEA 2009, Incl. SEDES 2009 Simp. para Estud. Doutor. em Eng. Softw.*, pp. 51–58, 2009.
- [5] K. Pierce, C. Ingram, B. Bos, and A. Ribeiro, "Experience in managing requirements between distributed parties in a research project context," *Proc. - IEEE 8th Int. Conf. Glob. Softw. Eng. ICGSE 2013*, pp. 124–128, 2013.
- [6] A. A. Khan, S. Basri, P. D. D. Dominic, and F. E. Amin, "Communication risks and best practices in global software development during requirements change management: A systematic literature review protocol," *Res. J. Appl. Sci. Eng. Technol.*, vol. 6, no. 19, pp. 3514–3519, 2013.
- [7] M. W. Bhatti, F. Hayat, N. Ehsan, A. Ishaque, S. Ahmed, and E. Mirza, "A methodology to manage the changing requirements of a software project," *2010 Int. Conf. Comput. Inf. Syst. Ind. Manag. Appl. CISIM 2010*, pp. 319–322, 2010.
- [8] M. Mäkäräinen, "Software change management process in the development of embedded software," *Vtt Publ.*, 2000.
- [9] M. Niazi, C. Hickman, R. Ahmad, and M. Ali Babar, "A model for requirements change management: Implementation of CMMI level 2 specific practice," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5089 LNCS, pp. 143–157, 2008.
- [10] CMMI Product Team, "Capability Maturity Model Integration (CMMI) for Systems Engineering / Software Engineering / Integrated Product and Process Development / Supplier Sourcing, Version 1.1," no. March, p. 729, 2002.
- [11] X. Luo, K. Kar, S. Sahu, P. Pradhan, and A. Shaikh, "On improving change management process for enterprise IT services," *Proc. - 2008 IEEE Int. Conf. Serv. Comput. SCC 2008*, vol. 2, pp. 341–348, 2008.
- [12] a. Kobayashi and M. Maekawa, "Need-based requirements change management," *Proceedings. Eighth Annu. IEEE Int. Conf. Work. Eng. Comput. Based Syst. 2001*, pp. 171–178, 2001.
- [13] J. D. Herbsleb, A. Mockus, T. a Finholt, R. E. Grinter, B. Labs, and L. Technologies, "Distance, Dependencies, and Delay in a Global Collaboration," *Cscw*, pp. 319–328, 2000.
- [14] R. Lai and N. Ali, "A Requirements Management Method for Global Software Development," *Humanpub.Org*, vol. 1, no. March, pp. 38–58, 2013.
- [15] S. Ramzan and N. Ikram, "Requirement change management process models: Activities, artifacts and roles," *10<sup>th</sup> IEEE Int. Multitopic Conf. 2006, INMIC*, pp. 219–223, 2006.
- [16] S. M. Ghosh, H. R. Sharma, and V. Mohabay, "Analysis and Modeling of Change Management Process Model," vol. 5, no. 2, pp. 123–134, 2011.
- [17] H. Holmstrom, E. Ó. Conchúir, P. J. Ågerfalk, and B. Fitzgerald, "Global Software Development Challenges: A Case Study on Temporal, Geographical and Socio-Cultural Distance," *Int. Conf. Glob. Softw. Eng.*, pp. 3–11, 2006.
- [18] A. A. Khan, S. Basri, and P. D. D. Dominic, "Communication Risks in GSD during RCM: Results from SLR," 2014.
- [19] M. Bano, S. Imtiaz, N. Ikram, M. Niazi, and M. Usman, "Causes of requirement change - a systematic literature review," *16th Int. Conf. Eval. Assess. Softw. Eng. (EASE 2012)*, pp. 22–31, 2012.
- [20] S. U. Khan, M. Niazi, and R. Ahmad, "Critical barriers for offshore software development outsourcing vendors: A systematic literature review," *Proc. - Asia-Pacific Softw. Eng. Conf. APSEC*, pp. 79–86, 2009.
- [21] J. M. V. Fernandes, "Requirements Change in Complex Product Development: Understanding Causes, Managing Uncertainty and Planning for Change," 2015.
- [22] D. E. Damian, "Requirements Engineering challenges in multi-site software development Didar Zowghi Departments of Software Engineering University of Technology, Sydney," *Requir. Eng.*, vol. 8, no. 3, pp. 149–160, 2007.
- [23] S. A. Kumar and T. A. Kumar, "Study the Impact of Requirements Management Characteristics in Global Software Development Projects: an Ontology Based Approach," *Int. J. Softw. Eng. Appl.*, vol. 2, no. 4, p. 107, 2011.
- [24] V. Sinha, B. Sengupta, and S. Chandra, "Enabling collaboration in distributed requirements management," *IEEE Softw.*, vol. 23, no. 5, pp. 52–61, 2006.
- [25] S. Hussain, N. Ehsan, and S. Nauman, "A Strategic Framework for Requirements Change in Technical Projects: Case Study of a R & D," *Int. Conf. Comput. Sci. Inf. Technol.*, pp. 354–358, 2010.

- [26] W. Hurssain, "Requirements Change Management in Global Software Development: A Case Study in Pakistan," 2010.
- [27] S. Lehnert, Q. U. A. Farooq, and M. Riebisch, "A taxonomy of change types and its application in software evolution," *Proc. - 2012 IEEE 19th Int. Conf. Work. Eng. Comput. Syst. ECBS 2012*, no. Section III, pp. 98–107, 2012.
- [28] A. M. Pettigrew, R. W. Woodman, and K. S. Cameron, "Studying organizational change and development: Challenges for future research," *Acad. Manag. J.*, vol. 44, no. 4, pp. 697–713, 2001.
- [29] S. M. Ghosh, H. R. Sharma, and V. Mohabay, "A Study of Software Change Management Problem," no. 3, pp. 39–48, 2011.
- [30] REN17 and T. Editors, "Getting requirements right: avoiding the top 10 traps ..," *Ibm White Pap. Ser.*, no. October, 2009.