A Systematic Approach for Requirement Elicitation in Globally Distributed Software Development

Zafar Ul Islam, Nasir Mehmood Minhas
University Institute of Information Technology (UIIT),
Pir Mehr Ali Shah University of Arid Agriculture Rawalpindi (PMASUAAR),
Rawalpindi, Pakistan.

zafar_islam65@yahoo.com, nasirminhas@uaar.edu.pk

ABSTRACT----Globally Distributed Software Development (GDSD) has different characteristics that make it different from traditional software development. These characteristics are directly affecting the requirement engineering process which is most crucial step in software development phase. GDSD trend is rapidly increasing for few years, and demanding an investigation on requirement engineering in multi site organization. In this paper, the goal was to examine different requirement engineering practices with reference to GDSD. Identify that how to improve requirement engineering process in GDSD areas where culture diversity, poor communication, time zone, ineffective knowledge and trust negatively affecting the requirement engineering process, specially requirement elicitation and requirement negotiation process. In this paper we analyzed all those aspects that might cause of problems in GDSD environment and suggested a framework that help to improve requirement elicitation process in GDSD environment.

Keywords- Requirements Engineering; Requirements Elicitation; GDSD

1. INTRODUCTION:

Requirement engineering plays an important role in software development. One of the estimate state that 40 % of the requirements need to rework during the course of the software developments projects Lloyd, W., Rosson, M., Arthur [23]. From few years it has been observed that a major cause of software failure is the problems occurring in requirement engineering phase, as a result the final product don’t meet the user Requirements Miller J, Yin Z [32]. Errors in Requirement engineering have major impact on software cost. It is evident that early defect detection and correction during the requirement phase may resolve many problems later during coding, testing and maintenance, Daniele E. Damian Didar Zowghi [4]. For past few years new trends are coming in our software industry people around are terming these trends as GSD (Global Software Development) and DSD (Distributed
Software Development). There is a very thin line difference between these two concepts, from here onward we will be using term GDSD (Globally Distributed Software Development) for simplicity purpose. GDSD involves geographically distributed software development organizations and stakeholders. There are number of benefits of this environment like access to cheap labor, 24/7 working environment (follow the sun mechanism), exploration of new markets, etc. Beside these benefits, in GDSD environment a series of new problems emerge, particularly when stakeholders are distributed in different geographical location. Fig 1(Inter Organization Globally Distributed Projects) shows the structure of Globally Distributed Software Development projects. Well known problems during the traditional requirement elicitation process those have been extensively analyzed in literature [1, 3, 4, 16, and 19]. These problems have been highlighted in the coming section.

1.1 Geographical Distance:

Geographical distance between different sites, has direct impact on communication process, it introduces barriers for face to face communication between stakeholders. Clear communication is critical to avoid misunderstanding and conflict resolution. Typically requirements elicitation activity requires face to face communication between stakeholders, but due to GDSD environment face to face communication would be reduced severely between various stakeholders.

1.2 Time Zone:

Time zone difference has also effect on communication related issue specially when there is no natural time overlap of working hours for voice or video conversation. Time difference affect both elicitation and negotiation activities of requirement engineering. Requirement engineering is often considered a time consuming activity. If time does not overlap or overlap for a short period then synchronous communication is not possible between different stakeholders and delay may occur in project.

1.3 Culture Diversity:

Requirement engineering process widely depends on the stakeholders’ relationship. This becomes more and more important when we have to deal with the difference in national, ethical tradition,
customs and norms of behavior as well as language. People from different sites in different country may have different tradition. When the native language is not same and level of common language is different, this may be a source of misunderstanding. It is better to detect such types of problems as soon as possible so there consequence can easily prevent.

1.4 Knowledge Management:

In requirement engineering there are huge amount of information from multiple sources need to be share between all stakeholders, but due to distance between different sites make this problem worse as compare to tradition software development. Much of this is form of knowledge remains tacit and undocumented particularly at the early stage of requirement engineering process. This problems is extravagant further when stakeholders are geographical distributed and are not able to communicate one an others to pass the knowledge.

1.5 Trust:

Inequality between sites, temporal separation, cultural diversity, language difference all contribute to the challenges of building atmosphere of trust. Most researchers agree that building trust is a great challenge in creating successful global teams. Trust called glue of global software development. Charles handy state that” Most of the organization tend to arranged the assumption that people cannot be trusted or relied upon, even in a tiny matter”

1.6 Technical issue:

Several technical aspects also affect requirement elicitation process in GDSD environment. The requirement engineering process depends upon the coordination and control mechanism that can reduce the impact caused by team distribution. The main factor found is pattern, process and configuration management.

Many projects have fail while the number of professional could not attend the growing demand major problems in communication and coordination activity that is critical during early phase of strategic planning, requirement gathering, analysis and negotiation Dildar Zowghi [4]. The geographical and
temporal distance between stockholders, increase the difficulty in developing the requirement engineering process Gabriela N [1]. The geographical and temporal distance between the industries has rapidly adopted these practices owing to their many economic advantages, such as possibility to saving cost by locating the software development in where salaries are lower. However graphical dispersion over multiple sites also has a negative effect upon the teams. The most important challenges that GDSD must confront in this respect is the lack of face to face communication, in addition to others factor such as culture diversity, time difference, which are also worthy of consideration at [2].

In this paper the author analyzed various existing approaches which are being used today in GDSD along with challenges and a problem faced due to the distributed nature of GDSD and proposed framework called Requirements engineering in Globally Distributed Software Development (RE_GDSD).

Rest of the paper is organized as follow: section II background, section III Requirements Engineering in globally distributed software development, section IV Our Proposal, Section V and VI is Comparison and Conclusion.

2. BACKGROUND:

Prior research reveiles, that the traditional requirement engineering process is totally different from GDSD process [1, 2, 3, and 4]. As in GDSD inadequate communication, ineffective Knowledge, time difference, culture diversity and trust is a big issue. Requirement engineering required face to face communication are remove existing conflict through negotiation [2, 5, 7, 16, and 22]. Where in GDSD the scenario is totally different where Client, user and developer are distributed in different geographical location. The requirement engineering is not machine oriented process it is a human oriented process [27]. A survey of the literature reveals a large volume of research work in this area of GDSD.

In software development process, Requirement elicitation is the phase which investigates the stockholder needs and their requirements to encompass them to product or to update the previous
range of products. To make success software development process, requirement engineering plays a

crucial role as the outcome of this phase will affect all others phases. The information gathers should

be related to stockholder needs and should be measurable, testable and actionable. Mahemoff at [61]
presents his work about how culture diversity affects the requirement elicitation process in GDSD

environments. The main contribution is form of classifying culture factor that impact in requirements,
based an overt and covert factors. It also purpose that the classification could from the basis for a

repository of culture information access by the developer. The author has described requirement
elicitation and validation with the perception of the real words scene. In [6] author purposed an

approach which capture the usage of the currents system using multi medium including picture, audio,

and video and then the observation has been analyzed related to the original requirement. In [24]
authors describe different requirements engineering process for a global software development

process outlining some preliminary suggestions on what such process model would include. Research
was conducted through a field study where is conclude that some of the fundamental problems
associated with the activity of the requirement engineering process are exacerbate when software
development team globally distributed. The study describes briefly the impact of global software
development team in requirement engineering process and argues that there is a need to investigate
and develop a requirement engineering process to supports. Lloyd’s researches [41] present an

empirical study of how a groupware tool can be used to minimize the problems in Distributed
Software Development. Damian [38] presents conclusion from case study in two different global
Distributed software development organizations. Author model how distance affects the requirement
elicitation activity in distributed Software Development Environments. Author tries to minimize the
problem which occurs in requirement elicitation due culture diversity, time difference, inadequate
communication and knowledge management. D. Zowghi [16, 37, 24] the author main research work
was a case study. It is possible to justify used of quantity since it is involved the study in the system
development in its actual circumstance with concise explanation and the understanding the methods
in those situation where carry out precedes the theory. Davis [53] the author describes effectiveness of
elicitation technique in distributed software Development environments and systematically reviews all
these technique in his work. The author have used aggregation analysis and found that structured
interview are the most effective technique in eliciting requirements Daniela [4] author purpose a set of feature of awareness in GDSD. The author discuss a research project in which he attempt to study the feature of such awareness supports the author discuss a research project in which he attempting to study the feature of such awareness supports. Gabriela N. Aranda1, Aurora Vizcaíno2, Alejandra Cechich1, Mario Piattini [3]. The author attempts to detect in advance possible sources of problems that might take place in a GDSD project, and suggest strategies to minimize them. In the first Phase of their work he tries to investigate as much as possible about the environment and the people that will be the parts of the requirement elicitation process along with the domain. In the second phase he defines strategies to minimize the problems that may appear during the requirement gatherings. Gabriela N. Aranda1,*., Aurora Vizcaíno2 and Mario Piattini [2] the author discuss various issues which today affect our global software development process. These issues are inadequate communication, inefficient knowledge, language diversity, time zone culture diversity and much more, he suggests various techniques to minimize these problems. His strategy is: How to minimize problems related to high and intermediate degrees of cultural and language diversity, by means of training in the cultural differences that may appear. How to minimize problems related to high and intermediate degrees of language diversity, by means of ontology as a communication facilitator. How to minimize problems related to communication by means of the study of the cognitive nature of people and the characteristics of their environment Gabriela N. Aranda1,*., Aurora Vizcaíno2 and Mario Piattini [3] the author proposed another approach to minimize the global software development issue, his purposed approach consists of a framework for requirement elicitation process in global environment called RE_GSD. His framework is a basis for adjusting or extending co-located requirement elicitation methodologies to GDSD environment. It not only takes into account the characteristics of GDSD projects but also their stakeholders’ cognitive characteristics with the goal of making them feel more comfortable in the elicitation environment. Fig 2 (Requirements – Software Failure) show a ratio of software failure due to various levels of requirement (according to literature review about this research)
2. REQUIREMENT ENGINEERING IN GLOBALLY DISTRIBUTED SOFTWARE DEVELOPMENT

Requirement engineering becomes more difficult activity when stakeholders are distributed in different geographical location. If we do not perform efficient requirement engineering process it become more difficult in local as well as global software development. Most of the organization wants to outsource their project, hence leading to GDSD. We face a lot of challenges in this area like travel cost, communication issue, time constrain, availability of the staff, culture difference, knowledge management, trust and so on.

Many software fail due to poor understanding of the requirement. Communication and collaboration is most important activity in requirement elicitation. But in globally distributed software development we face a communication problem due to distance; distance is an important factor in Requirement engineering for globally distributed software development environment. Culture, trust and time zone is an others important issue in distributed software development. Table 1 (Affected Requirement Elicitation Activity Due to GDSD Problems) show which requirement elicitation activity is affected due to GDSD challenges.

4. OUR PROPOSAL

Our proposal consists of the framework for requirement elicitation for globally distributed software development environment, called RE-GDSD. The existing approaches of requirements elicitation have some limitation when they used in GDSD for requirement gathering. Author has study various model like liner requirement engineering model, this is a basic model and it is just used for a simple projects. It cannot be use large and complex type of the projects. This model provide functionality to others models. But there are many problems associated with this model like no iteration, no user feedback, and freezing of requirements. The next model the author has study is liner iterative model. This model solve some of the problems which analyst face in Liner model such as requirement validation and freezing of requirements which is a good thing for this model but this model also has some drawback such as no reverse engineering is possible in this models. The next model is spiral model which is very high cost models as compare to others models, and demand highly specific expertise to analysis any type of risk in this models. This model is not also fit for small projects. The
next model which author has study is iterative model. This method address many problems associated with the other models. But in iterative model people get stuck in a loop when they are working in a project. Each phase in iteration is rigid with no overlap at minimum change whole process repeat again and again. The most important thing is delay in one phase can affect all others phase.

As every model has some advantage and disadvantage so authors suggest a combine approach for requirement elicitation process in GDSD environment call RE_GDSD framework. The main characteristics of RE_GDSD framework include:

4.1 PHASE 1: Stakeholder Identification

Stakeholder identification process is a parts of requirement elicitation process. Stakeholders play critical roles as source of data and gatekeeper of the information. Accurate identification of the stakeholders is essential to the successful of the projects. Stakeholder specification is an important aspect in requirement engineering, because stakeholders hold valuable knowledge and have an active participant in requirement elicitation process. Stakeholder identification process consists of following four steps. These steps are shown in fig 3(Stakeholder Identification Process).

4.1.1 Identify Stakeholder Type

In this step we specify the types of the stakeholder which participate in Requirement elicitation process in our GDSD environment. The identification of the stakeholder is based on some criteria (function, knowledge, hierarchical level)

4.1.2 Identify Stakeholder Roles

This step identifies the role that will be including in the projects. This step is performed simultaneously with the step 1.

4.1.3 Stakeholder Selection

This step is based on result of step 1 where we specify different stakeholder base on some criteria. The project manager will analyze each profile and select those stakeholders which satisfy the given criteria.

4.2 PHASE 2: Stakeholder Data Collection According to GDSD Problems

The main goal of this phase to discover as much as possible about the environment and about the stakeholder who is selected in phase 1, which will be the parts of the requirement elicitation process.
In this phase we collect GDSD problems related data by focusing on the most common problems referred to literature related this topic. To collect such types of information of stakeholder we designed a set of forms which is presented in [3].

4.3 PHASE 3: Problems Identification:

The goal of this phase is to analyze the data gather in phase 2, Identify the source of problems that may cause of future problems. To do this we analyzed the information gather during second phase and detect which factor affect our requirement elicitation process by keeping in mind the most common problems which we study during the literature review about this topic. Different factor which can be measured in globally distributed team have been defined. They are culture diversity, time difference, knowledge management, geographical distance. We use different linguistic tags and we shall explain how value of different linguistic tag can be obtained of each problem. The problems identification process show in figure 4 (Problem Identification Process).

4.3.1 Geographical Distance:

In tradition software development process developing software development team as a challenging task, but this task become more complex when team are distributed in different geographical location. Distance directly impact in team communication, coordination and control. Communication is exchange of information between team members, whether formally or informally, coordination between team members is also important to attain overall objective while control is the process of adhering objective and policy the impact of geographically distance diagrammatically show in fig 5 (Impact of Distance on GDSD).

4.3.2 Different Time Zone:

Time differences are calculated by analyzing the information gathers during the second phase and then convert them into Greenwich Mean Time. In order to define tag high, Medium, low for overlap factor we purpose a formula for calculating time difference between various stakeholders base on n working hour which is presented in the [6]. for example we have three virtual sites, site A, site B and site C site A in UK site B in Pakistan and site C in Malaysia. Since time zone between Pakistan and UK which is +5 Pakistan and Malaysia is +3 Malaysia is 3 hours a head from Pakistan and time
difference between UK and Malaysia is +8 hours according to GMT we calculate time table for GMT standard as fellow: This information is shown in Fig 6 (Time Table Overlap Evaluation).

4.3.3 Culture difference:

To analyze the culture difference between various stakeholders we use Hofstede mode which is frequently used to measure culture difference in GDSD projects. There are five dimension of Hofstede mode model: Power distance, individualism, uncertainty, masculinity, and long terms orientation. To obtain value of culture difference we used a formula proposed to calculate a culture difference between two countries which is presented in [6]

$$D_{A,B} = \sum_{i=1}^{5} |v_i(A) - v_i(B)|$$

Where ‘I’ is the dimension of Hofstede model and vi (x) is the value of ith dimension of the country x.

Suppose we calculate a culture difference between Australia and Brazil by applying the above formula, we got the total cultural difference value is 158, other values are shown in Table 2 (Culture Difference between Brazil and Austria).

4.3.4 Language difference:

The most observable communication barriers within distributed environment are language. If we choose a common language for communication between various stakeholders, and then there was a problem for those stakeholders which being use common language as second language and need more time to respond and must simplify their terminology and sentence formation. At the other end the receiver who is using this common language as a first language may translate the message incorrectly. At other end people who share mother language from usually from different country or region may have difference regarding to pronunciation and word convention.

4.3.5 Trust:

Due temporal, culture and language difference between different sites may be challenges of an atmosphere of trust. Trust is an important antecedent of effective work. So this issue is true not only face to face group but also virtual team. In face to face team trust can build from informal meeting, face to face interaction between team members at everyday but this type of interaction is not possible
in distributed environment. We use “Be a STAR” approach to minimize the trust issue in our distributed environment.

4.3.6 Misunderstanding:
Due to different culture background and different communication style that stakeholders carry with them in distributed environment can cause misunderstanding in other’s behaviors from others countries. Misunderstandings also reduce the level of trust between team members.

4.4 Phase 4: Strategy Recommendation to Minimize These Issues:
This phase is most crucial phase of our research, after observing all factors which found during the analysis of the previous phase, here author propose a strategy to minimize these factor to improve requirement elicitation process.

4.4.1 Strategy A: Use of software tool for EPM
This tool facilitates the project manager to monitor and record any action between various stakeholders in most convenient way than the tradition software development. There are many such types of tool for EMP. By utilizing of this tool every project member action can be recorded and forward to project manager even email, instance message and VoIP. Due to use of EMP, project manager can aware at project teams members action at any time.

4.1.2 Strategy B: Use of Synchronous or Asynchronous Communication Technology:
When time difference is high between team members we recommend synchronous communication and when time difference is medium we recommend asynchronous communication and when time difference is low or no overlap we recommend compromise between team members and change their job into shifts.

4.1.3 Strategy C: Software Interpreter Related to language issue
Langue issue is most common and crucial issue in GDSD environment. To overcome this issue the author recommend software language interpreter. If distributed team use a common language, each members of the team have different way to understand and speak the language. Moreover in addition to actual human interpreter, the project team can use the language translation software which makes it possible to for dispute form different site.

4.1.4 Strategy D: Culture Interpreter Related to Misunderstanding
Different culture background between various stakeholders can be cause of misunderstanding between each other’s behaviors; we cannot ignore this crucial issue during GDSD and need to be developing strategy to control such types of problems. Author suggests that if we hire a person with in our distributed environments with a specific role such as culture interpreter and they are clearly aware of their responsibility. This interpreter are interfere when any stakeholder tense and fail to communicate with others stakeholders.

4.1.5 Strategy E: “Be STAR” Approach
Lack of trust breakdown of coordination efforts between distributed team. When there is a lack of trust between team members there is a lack of willingness to communicate. We use “Be a STAR” Approach to build trust among globally distributed stakeholders.

4.5 PHASE 5: Requirements Gathering
When problems identify and strategy has been defined against each problem then we carry out our requirement elicitation process. As shown in figure 8. Requirements gathering phase begin with the choosing a set of requirement elicitation technique that is applicable with the current strategy defined during previous phase and select one of the best technique to elicit the requirement.

5. COMPARISON BETWEEN RE-GDSD FRAMEWORK AND EXISTING FRAMEWORK:
To compare RE_GDSD we have studied different framework for requirement elicitation in GDSD environment and selected iterative, christle, and RE_GSD framework as a basis of our proposal. The selection is due to fact that all these frameworks are generic view and fit for our intension. Table 3 show the comparison between RE_GDSD Framework and existing Frameworks.

6. CONCLUSION
Requirement engineering plays an important role in software development. Form few years it has been observed that problem in requirement engineering phase is a major cause of the software failure. In GDSD environment we are facing new challenges, like Cultural difference, Time Zone difference, Knowledge Management and Trust.
In this paper we studied the existing approaches which are being used for requirement elicitation in GDSD environment. Existing techniques of requirement elicitation have been reviewed critically and a new approach has been suggested with the name RE-GDSD. Finally a comparison was made between the existing techniques and the proposed one. Results showed that the proposed technique is batter in all respects as compared to existing ones. Future work of this research is to fuse the two stages of RE-GDSD into one stage framework. An automated system will be built on the basis of RE-GDSD framework.

REFERENCES

1. Gabriela N. Aranda, Aurora Vizcaíno 2010 A framework to improve communication during Global Software Development. Received: 10 July 2009 / Accepted: 21 April 2010 / Published online: 21 May 2010 _ Springer-Verlag London Limited 2010

2. Gabriela N. Aranda, Aurora Vizcaíno 2010 Analyzing and evaluating Main factor that challenges in Global software Development GIISCo Research Group, Computing Sciences Department, Universidad Nacional del Comahue, Buenos Aires, 1400. 8300 Neuquén, Argentina

3. Gabriela N. Aranda1, Aurora Vizcaíno2, Alejandra Cechich1, Mario Piattini 2008 Universidad Nacional del Comahue, Departamento de Ciencias de la Computación,

4. Daniele E. Damian Didar Zowghi 2007 Requirements Engineering Challenges in Multi site Software Development


7. Al-Salem Lana S, Abu SamahaAla Eliciting web application requirements- and industrial case study (2007)


9. MRomeroll, A. vizcaino, M. piattini, using virtual agent for teaching of requirements elicitation in GSD publishers springer Berlin / Heidelberg 2008 , pages (s) 539-540


13. Herela D, Greenberg S , using groupware for distributed requirements using components . technical Reports 160, international Journal review in computing © 2009-2010 IJRIC.


47. Jeju Enhancing Requirements Engineering Activities through the Use of Mobile Technology Devices and Tools, December (2008)


49. Herbsleb JD, Grinter RE (1999) Splitting the organization and integrating the code Conway’s law revisited In 21th International conference on software engineering (ICSE/99). New York, pp 85–95


57. LarsMathiassen et al, Management Requirements engineering Risks , an analysis and synthesis of literature , November 2004


59. Perini Anna ,RiccaFilippo, Susi Angelo, Tool Supported requirements prioritization: Comparing theAHP and CBRank methods,(2009)


61. Mahemoff ,M, Johnston , 1998 software internationalization, implications for requirements engineering processing of third Australian workshop on requirements engineering
Fig 1: Inter Organization Globally Distributed Projects

Fig 2: Requirement – Software Failure
Table 1: Affected Requirement Elicitation Activity Due to GDSD Problems

<table>
<thead>
<tr>
<th>PROBLEMS</th>
<th>CHALLENGES</th>
<th>RE-ACTIVITY AFFECTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>CULTURE DIVERSITY</td>
<td>- Difference in customer culture and business</td>
<td>- Elicitation</td>
</tr>
<tr>
<td></td>
<td>- Trusting in working relationship</td>
<td>- Negotiation</td>
</tr>
<tr>
<td></td>
<td>- Common understanding of requirements</td>
<td></td>
</tr>
<tr>
<td>INADEQUATE COMMUNICATION</td>
<td>- Appropriate of the user</td>
<td>- Elicitation</td>
</tr>
<tr>
<td></td>
<td>- Trusting working relationship</td>
<td>- Negotiation</td>
</tr>
<tr>
<td></td>
<td>- Manage conflict and for open discussion of interest</td>
<td>- Examining current system</td>
</tr>
<tr>
<td></td>
<td>- Common understanding of requirements</td>
<td>- Manage uncertainly</td>
</tr>
<tr>
<td></td>
<td>- Effective meeting</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- delay</td>
<td></td>
</tr>
<tr>
<td>KNOWLEDGE MANAGEMENT</td>
<td>- Trusting working relationship</td>
<td>- Negotiation</td>
</tr>
<tr>
<td></td>
<td>- Common understanding of requirements</td>
<td>- Manage uncertainly</td>
</tr>
<tr>
<td></td>
<td>- Effective meeting</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- delay</td>
<td></td>
</tr>
<tr>
<td>TIME DIFFERENCE</td>
<td>- Manage conflict and for open discussion of interest</td>
<td>- Elicitation</td>
</tr>
<tr>
<td></td>
<td>- Common understanding of requirements</td>
<td>- Negotiation</td>
</tr>
<tr>
<td></td>
<td>- Effective meeting</td>
<td>- Examining current system</td>
</tr>
<tr>
<td></td>
<td>- delay</td>
<td>- Manage uncertainly</td>
</tr>
<tr>
<td></td>
<td>- Elicitation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Negotiation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Examining current system</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Manage uncertainly</td>
<td></td>
</tr>
</tbody>
</table>

Fig: 3 Stakeholder Identification Process

Fig4: Problem Identification Process
**Fig 5:** Impact of Distance on GDSD

**Fig 6** Time Table Overlap Evaluation

**Table 2:** Show the Culture Difference between Brazil and Austria

<table>
<thead>
<tr>
<th></th>
<th>PDI</th>
<th>IVD</th>
<th>UAI</th>
<th>MAS</th>
<th>LTO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brazil</td>
<td>69</td>
<td>38</td>
<td>49</td>
<td>76</td>
<td>65</td>
</tr>
<tr>
<td>Australia</td>
<td>36</td>
<td>90</td>
<td>61</td>
<td>51</td>
<td>31</td>
</tr>
</tbody>
</table>

Culture Diff 33 52 12 25 36

D Brazil, Australia (33+52+12+25+39) = 158
**Fig 7:** Recommend Strategy according to GDSD Challenges

**Fig 8:** RE-GDSD Framework

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Stakeholder Identification Process</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Identify stakeholder on the basis of some criteria and the identify the roles of stakeholder which they play during the project and finally select all those stakeholder which satisfy the given criteria</td>
</tr>
<tr>
<td>Stakeholders Data Collection</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td>Design set of form of for Data Collection of All Stakeholders</td>
</tr>
<tr>
<td>Stakeholder Data Collection</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td>Design set of form for Data Collection accord to GDSD Challenges which study during literature review and collect data only those Stakeholders which we select in ph 1</td>
</tr>
<tr>
<td>Problems identification</td>
<td></td>
<td>Identify Problems</td>
<td>Identify Problems</td>
<td>Identify Problems</td>
<td>Identify Problems</td>
</tr>
<tr>
<td>Identify Source of Problems</td>
<td></td>
<td>1) inadequate Communication 2) Time overlap 3) Culture Difference 4) Knowledge Management 5) Trust 6) Misunderstanding</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strategy to Minimize Problems</td>
<td></td>
<td>Select Elicitation tool/ technology</td>
<td>Requirement Elicitation</td>
<td>Requirement Negotiation</td>
<td>Requirement Documentation</td>
</tr>
<tr>
<td>Strategy</td>
<td></td>
<td>Use of ontology 2) tra about Cult 3) Recommend a- synchronous communication for time difference 4) syn chromcsem or divide job into shift 5) Star Approach to build trust 6) culture Training</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Elicitation Approach</td>
<td>Elicitation Approach</td>
<td>Elicitation Approach</td>
<td>Elicitation Approach</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------</td>
<td>----------------------</td>
<td>----------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Iterative Approach</td>
<td>Iterative Approach</td>
<td>Iterative Approach</td>
<td>Combine Approach</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Disadvantage</strong></td>
<td><strong>Disadvantage</strong></td>
<td><strong>Disadvantage</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>People stuck in loop, min changes in whole process repeat again and again, delay in one phase can effect in all phase, delay mean increase both time and cost and result chance of software failure</td>
<td>People stuck in loop, at minimum changes in whole process repeat again and again delay in one phase can effect in all phase, delay mean increase both time and cost and result chance of software failure</td>
<td>People stuck in loop, at minimum changes in whole process repeat again and again delay in one phase can effect in all phase, delay mean increase both time and cost and result chance of software failure</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 3:** Comparison between RE, GDS and Existing Framework