

**IDENTIFYING FACTORS IN REQUIREMENTS  
ENGINEERING PHASE AND IMPACT THEREOF ON  
SOFTWARE DEVELOPMENT PHASE: A STUDY ON  
SRI LANKAN BANKING SOFTWARE DOMAIN**

Kananke Gamage Wenura Rangana  
(159129V)

Thesis submitted in partial fulfillment of the requirements for the degree of Master of  
Business Administration in Information Technology

Department of Computer Science and Engineering

University of Moratuwa  
Sri Lanka

May 2017

## **Declaration**

“I declare that this is my own work and this thesis does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief, it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my thesis/dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature:

Date:

K.G.W. Rangana

The above candidate has carried out research for the Master’s thesis under my supervision.

Name of the supervisor: Dr. Indika Perera

Signature of the supervisor:

Date:

## **Abstract**

Requirements engineering is an important stage of the software development life cycle. All other phases of development lifecycle are depending on that. Especially the development phase is based on the requirements and therefore requirements should be clearly understood and properly documented. There can be several factors which should be considered within the requirements engineering phase. In this research through the literature review, it has been identified four major factors; they are communication, inadequate/incomplete requirements engineering, requirements prioritization, and requirements documentation. Based on the identified factors, a separate questionnaire has been developed for requirements engineers and software engineers. The questionnaire for requirements engineers is measuring how they are effectively using those factors within the requirements engineering phase and the questionnaire for software engineers is measuring how much impact is there for software development phase by those factors. With the information gathered, the Straussian grounded theory has been used to analyze the data and the final outcome shows that those factors are highly effective in the requirements engineering phase and impacts to the development phase as well.

**Keywords:** Software Development Life Cycle, Software Development, Requirements Engineering, Banking, Finance, Requirements, Requirements Documentation, Communication, Incomplete Requirements, Inadequate Requirements, Requirements Prioritization,

## **Dedication**

I dedicate this thesis to my family, all my teachers for nursing me with affection and love and their dedicated partnership for success in my life.

## **Acknowledgement**

I wish to express my greatest gratitude to all those who have helped me achieve this research on the topic of “Identification of Factors in Requirements Engineering and its Impact on the Software Development Phase: A study on Sri Lankan banking Software Domain” successfully.

# Table of Contents

Declaration .....	i
Abstract .....	ii
Dedication .....	iii
Acknowledgement .....	iv
Table of Contents .....	v
List of Tables .....	ix
List of Abbreviations .....	x
1. INTRODUCTION .....	1
1.1. Chapter Overview .....	1
1.2. Research Background.....	1
1.3. Problem Statement .....	2
1.4. Introduction to Software Development Life Cycle .....	3
1.4.1. Requirements Engineering.....	4
1.4.2. Design .....	5
1.4.3. Development .....	5
1.4.4. Testing.....	6
1.4.5. Maintenance .....	6
1.5. Objectives of the study .....	6
1.6. Significance of the study .....	6
1.7. Limitations of the study.....	7
2. LITERATURE REVIEW .....	8
2.1. Chapter Overview .....	8
2.2. Introduction to Requirements Engineering .....	8
2.3. Introduction to Banking .....	9
2.4. Communication .....	10
2.5. Documentation .....	13
2.6. Incomplete/Inadequate Requirements .....	14
2.7. Prioritization.....	15
3. METHODOLOGY .....	23
3.1. Chapter Overview .....	23
3.2. Introduction to Methodology .....	23

3.3.	Interview Structure .....	24
3.4.	Questionnaire Design .....	25
3.5.	Population and Sampling .....	25
3.6.	Interview Preparation .....	26
3.7.	Interview Execution .....	27
3.8.	Data Collection.....	27
3.9.	Data Analysis Techniques .....	27
4.	DATA ANALYSIS .....	29
4.1.	Chapter Overview .....	29
4.2.	Introduction .....	29
4.3.	Response to the Research.....	31
4.4.	Data Analysis .....	32
4.4.1.	Numerical Calculation .....	33
4.4.2.	Requirements Engineering.....	33
4.4.3.	Software Engineers .....	35
4.5.	Data Interpretation.....	35
4.5.1.	Detail Analysis.....	36
4.5.2.	Data Interpretation Summary.....	39
4.6.	Summary .....	40
5.	CONCLUSTION AND RECOMMENDATIONS.....	41
5.1.	Chapter Overview .....	41
5.2.	Conclusion.....	41
5.3.	Recommendations .....	42
5.4.	Future work .....	43
	References.....	45
	Appendix A: Questionnaire .....	48
	About the organization .....	48
	Professional profile .....	48
	Factor questions for requirement engineers .....	48
	Factor BASED questions for software engineers.....	49
	Appendix B: Interview Request Email .....	50
	Appendix C: Interview Request Letter .....	51
	Appendix D: Interview Request Research Summary .....	52
	Appendix E: Answers Summary – Requirement Engineers .....	53

Appendix F: Answers Summary – Software Engineers.....	58
Appendix G: Interview Answer Script – Requirement Engineers.....	67
Appendix H: Interview Answer Script – Software Engineers .....	78



## **List of Figures**

Figure 1.1: Software Development Life Cycle (SDLC)	: 4
Figure 2.1: Types of Requirements	: 8

## List of Tables

Table 3.1: Interviewees Description	: 26
Table 4.1: Questions for requirements engineers and its dimensions:	29
Table 4.2: Questions for software engineers and its dimensions	: 30
Table 4.3: Summary of organizations	: 31
Table 4.4: In detailed view of results summary	: 32
Table 4.5: Weightage system summary	: 33
Table 4.6: Weightage summary – Requirements Engineers	: 33
Table 4.7: Weightage summary – Software Engineers	: 35
Table 4.8: Relationship of selective coding and...	: 39
Table 4.9: Relationship of selective coding and...	: 40

## **List of Abbreviations**

SDLC	- Software Development Life Cycle
DB	- Database
IT	- Information Technology
BRS	- Business Requirement Specification
SRS	- Software Requirement Specification
RAD	- Rapid Application Development
ATM	- Automated Teller Machines
ICT	- Information Communication Technology
SLASSCOM	- Sri Lanka Association of Software and Service Companies

# 1. INTRODUCTION

## 1.1. Chapter Overview

This research focuses on the requirements engineering and the software development phases of the SDLC. In this research, it tries to identify the factors which make the requirements engineering phase more important and critical in Sri Lankan Banking Software domain as well as the impact to the software development phase due to those identified factors. This chapter summarizes the background to the problem domain as well as the introduction to the topics as well.

## 1.2. Research Background

Sri Lanka is evolving as one of the best destinations in global IT market. Further, the country has been ranked among the top 50 Global Outsourcing destinations by AT Kearney and ranked among top 20 Emerging Cities by Global Services Magazine. Over the past few years, Sri Lanka has won several awards for this quality factor, i.e. “outsourcing destination of the year by National Outsourcing Association UK, successively in 2013 and 2014 (SLASSCOM). So the opportunity to move into more international market will not be far away from us as a country.

To provide the global requirements in the IT industry, the output should be in good quality, cost effective and align in the agreed time frame. To absorb those good qualities into the local IT organizations, there are several methodologies to follow, such as, Waterfall Model, Incremental Model, Spiral Model and V-shaped Model etc. Even though there are many, common phases are there for each and every methodology. That is called software development life cycle SDLC (Kaur, 2015).

There are mainly five phases/stages of the SDLC. They are Requirements Gathering, Design, Development, Testing and Maintenance. SDLC is a globally accepted standard process to follow towards a quality product when developing software applications. It mainly ensures the quality, cost effectiveness and bring into line with the agreed time frame of the product outcome (Barracliffe, et al., 2009).

Among the above mentioned five phases, the first phase, Requirements gathering is the most important and the critical phase (Pandey, Pandey, 2012). Due to the high importance of this phase, there are dedicated teams to face this stage in most of the organizations today. Initially they collect the requirements from the clients through various methods, such as interviews, conference calls, documentations, emails, etc. Then the initially collected information is used to plan the project design and the approach. Not only the design of the project, but also several other important aspects too, such as technology, cost, time frame, quality, security features, etc.

After finalizing the requirements, Design process initiates. Based on the output of the designing phase, development starts. At this stage whatever the implementation completes is based on the built design on collected requirements. If the requirements are not complete enough, the final output of the SDLC process is not what the customer expected or not on the agreed time frame.

In the Banking software domain too, it is very important to follow the SDLC process in a proper manner. So in this research, it has measured the impact to the software development phase from the requirements engineering phase which is not only the requirement gathering but also integration of several functionalities such as identifications of functional and non-functional requirements, validating them and documenting them (Swarnalatha, et al., 2014).

The research is mainly based on identified four factors through the literature review. The identified factors are, Communication, Inadequate/Incomplete requirements, Requirements prioritization and Requirements documentation. So, in the research itself, it has been discussing how the impact of the development phase based on these factors.

### **1.3. Problem Statement**

Requirements engineering, Design, Development, Testing and Maintenance stages are key phases in the process, regardless of the software development methodology used in any organization today.

It is identified that, most of the software development projects fail due to inappropriate and incorrect gathering of requirements (Pandey, Pandey, 2012). There can be many reasons for that, such as communication problems, improper documentations, and inconsistency of requirements. Especially in the Banking domain, gathering correct, complete and consistent requirements are essential.

As mentioned above, requirements engineering become much crucial when we consider the success of the software product.

This research has been planned as a qualitative approach and collected data through interviews. Data had been analyzed for producing the impact of the development phase on requirements engineering phase. So as a summary in this research, it had been identified, selected factors through the literature review and those factors had been tested in the interview process for identifying the impact to the software development phase in the banking software industry.

#### **1.4. Introduction to Software Development Life Cycle**

Software Development Life Cycle is also known as SDLC is a set of predefined phases of developing and managing a software product in a way towards the success. Through this process it is ensured that the output is high in quality, cost effective, satisfy the customer requirement and deliver within the agreed time frame (Barracliffe, et al., 2009).

For the requirements engineering stage, the inputs are the customer requirements for the systems. Based on the requirements, the design phase begins. Taken design as the input to the development phase, software engineers start developing application. The output of the development stage, which is the final product is tested in the testing phase based on the requirements and finally the software product goes under the maintaining stage if it is needed further.

Below picture depicts the chain of each step in SDLC very clearly.

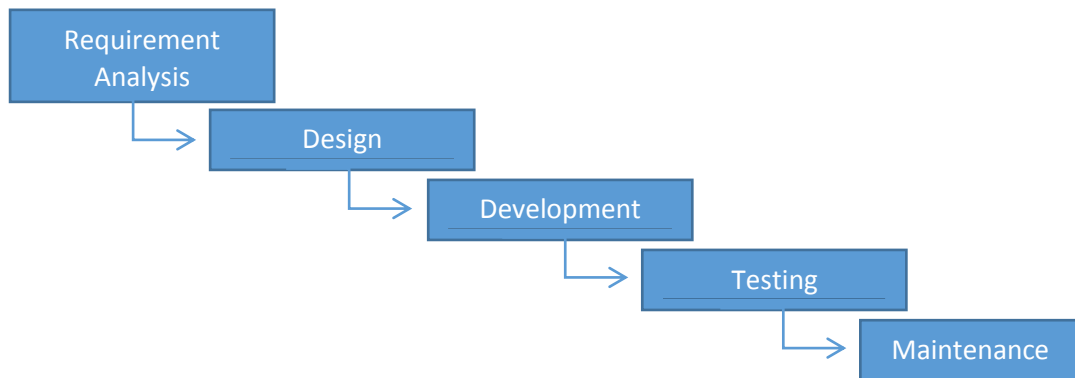


Figure 1.1: Software Development Life Cycle (SDLC)

The five stages of the life cycle are described below.

#### **1.4.1. Requirements Engineering**

Requirements engineering phase is the first phase in the process of the SDLC. In this phase, requirements are collected from the product owner and modes of collecting the requirements could be differ and some of them are,

- One to one interviews
- Conference calls
- Group interviews
- Questionnaires
- Prototyping
- Use cases
- Document analysis
- System redesigning
- Skype calls
- Email

From the IT organization's side, there are different levels of responsible people to participate in the requirements engineering phase. They can be,

- Software architects

- Project managers
- Technical leads
- Business analyst
- Sometimes even Software engineers/Quality assurance engineers, etc.

They are participating based on the availability, organizational standards and structure, experience and the necessity, etc. Among the objectives of this phase,

- Complete the detailed formal set of requirements for the systems.
- Verify the completed requirements with the stake holder, can be considered.

As output of the phase,

- Identify functional requirements.
- Identify non-functional requirements, can be considered.

Finally, after gathering required information from the customer, sign-off is taken before going into the next stage of the SDLC which is design phase.

#### **1.4.2. Design**

There are several important goals of this phase. They are,

- Create system design proposals to customer.
- Ensure the system design is aligned with the standards.
- Obtain approval from the product owner to the proposed design.

Design phase consisted with, UML diagrams, Logical DB design, Forms design and UI design. So, until the approval is taken from the internal and external parties, the phase continues.

#### **1.4.3. Development**

Based on the requirements and the design, real implementation of the system is started. According to the system capacity and the complexity, time to complete the implementation phase may differ. After completion of the phase, systems should be thoroughly tested to ensure that it complies with the requirements.



#### **1.4.4. Testing**

In most of the organizations today, dedicated team is available for testing purpose. It covers all the scenarios of the systems and that could be iterated several rounds until the team ensure that there are no more issues. Applications are tested against the collected requirements.

#### **1.4.5. Maintenance**

Some systems do not need maintenance phase, but some are essential. In this phase, it covers mostly the change requests, issues, data related tasks such as backups, etc.

So, from the section 1.4.1 to 1.4.5, it has been discussed that main five phases of the SDLC. But for this research, it considers only the requirements gathering and development phases.

### **1.5. Objectives of the study**

There are several objectives that are set for this research. They are,

- Identify important factors in the requirements engineering.
- To identify how those factors are applied in the requirements engineering phase of the software organizations.
- To identify how those factors impact to the development phase of the SDLC.
- To build an association between those factors and the impact to the requirements engineering and development phases.

### **1.6. Significance of the study**

Requirements engineering phase is more crucial due to the high importance of it in the SDLC. The entire software product based on the requirements from the stakeholder. So, it is very important to ensure Consistency, Completeness, and Correctness of the requirements. If, the requirements does not have above mentioned values, that will be a major factor for failing the product.

The identified factors through the literature are not unique to the banking domain but common factors to be considered for the requirements engineering phase in any domain. So, this study is not valuable only for the banking software sector, but for the other application domains as well.

### **1.7. Limitations of the study**

There are numerous limitations set for the research. They are as follows.

- The study does not include all the software development organizations in the Banking software domain.
- The study does include organizations; only develop Banking software as a product, not as a project.
- The study focuses only on the requirements engineering and the development stages of the SDLC.
- The study only targets the software organizations perspective not from the client's perspective.

## 2. LITERATURE REVIEW

### 2.1. Chapter Overview

The research will be a qualitative approach and it will be based on four identified factors. They are, Communication, Documentation, Incomplete/Inadequate requirements and Requirements prioritization. Factors were identified through a very comprehensive literature review.

### 2.2. Introduction to Requirements Engineering

Mainly the requirements can be categorized into two broad categories. They are as follows.

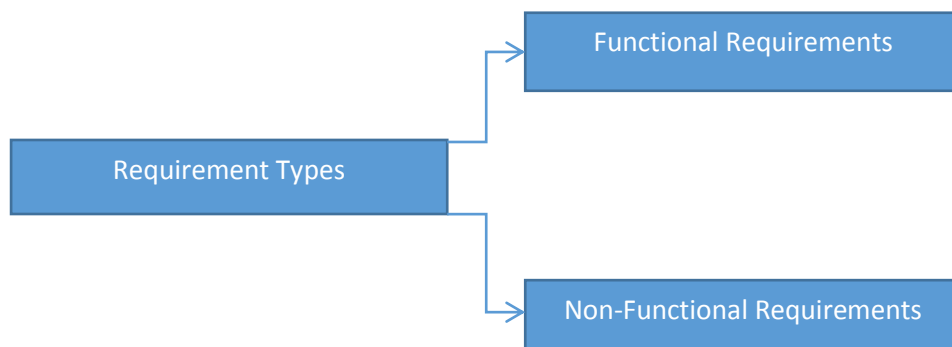


Figure 2.1: Types of Requirements

Functional requirements are software requirements which specify what the system should do and non-functional requirements specify how the system should work. Identifications of functional and non-functional requirements, validating them and documenting them are mainly considered as requirements engineering (Swarnalatha, et al., 2014).

Requirements engineering is not a standalone process, but it is an iterative process itself, because, after gathering initial requirements, there could be several iterative process to verify existing requirements or collect new requirements. Identified requirements should be clearly defined and documented. There could be several types of documents, preparing while in the requirements engineering process. One of the main documents is Software Requirement Specification (SRS) (Swarnalatha, et al., 2014).

### **2.3. Introduction to Banking**

The banking industry is specially based on the information. So the importance of the information management is very important. So, the IT plays a major role in the banking industry. Not only the commercial banking sector in the country, but also the public sector too uses the IT for competitive advantage (Suraweera, et al., 2014). To name few IT driven banking functions are,

- Online banking
- Mobile banking
- ATM
- Telephone banking
- Internet payment gateway
- Customer management

Apart from the above main options, there are many ways of integrating IT into the banking sector.

The banks use the technology to serve their customers in a secure, reliable, and affordable manner, and sustain competitive advantage over the other banks. The introducing of technology to the banking sector, would lead to the benefits such as,

- Greater productivity
- Profitability
- Efficiency
- Faster service
- Customer satisfaction
- Convenience
- Flexibility
- 24x7 operations
- Space
- Cost savings

Likewise, ICT, together with banking industry has integrated much and IT is becoming an essential part of the Banking industry (Ravi, 2008).

#### **2.4. Communication**

In the development phase, it refers documents, prepared in the requirements engineering phase. So, based on the requirements in the relevant documents, the application builds up. Therefore documents should be consisted with what the stakeholder needed. Communication is highly necessary at that point of negotiating with the client. This process also involves the movement of data to the meaningful information which is software requirements documents and can be considered as a negotiation process with the stakeholder for an agreement on what is to be developed. It is known that over the last 20 years, issues on requirements elicitation are important and becoming critical in developing software (Ravi, 2008). This is because the lack of requirements elicitation will cause failure of the whole project. In order to make the requirement gathering process more clearly on those issues various techniques have been introduced. They are,

- Interviews
- Document analysis
- Group work
- Ethnography
- Prototyping
- Questionnaires
- Scenarios
- Viewpoint

The most frequent technique used in the industry is interview technique. This technique requires direct interaction between the interviewer and the respondent. The qualities of the information obtained are closely related to interviewer's skill. Apart from that, there are few other techniques as well. They are focus group, workshop, etc. Each technique has its own quality and purpose, like prototype technique is mostly used in user interface verifications (Ravi, 2008).

To build up an effective software development phase, it is highly essential to have a strong requirements engineering phase (Ahmad, et al., 2012). Sometimes not only a single stakeholder communicates to the requirements engineers but multiple. In that case, the communication becomes a very complex topic because the requirements might vary from person to person. The paper discusses three different modes of communication which are Text based communication, Face to face communication and Rich media. The research had been conducted using six projects and those modes applied to them such as one mode for two projects and then the outcome was monitored and further, there were three dependent variables had been used to come up with a measurable result. The variables are satisfaction, comfort and perceived engagement. With the final outcome they have proved that the face to face communication has the highest level of satisfaction, comfort and perceived engagement during the negotiation and elicitation phases (Ahmad, et al., 2012).

Requirements are the blueprint of the software and they are collected through various methods, such as negotiation, documentation, etc. finally the collected requirements are used for the development phase. There could be situations like project fails in terms of cost, schedule, quality, failure to deliver due to inappropriate collection of requirements (Huzooree, Ramdoo, 2015).

Requirements engineering is consisted with mainly four phases. They are Requirements elicitation, Requirements negotiation, Requirements specification and Requirements validation (Huzooree, Ramdoo, 2015). The research has been done based on those four factors to identify how they impacts to the requirements engineering phase. The final outcome of the research indicates that, Issues mostly related to confusion of project scope, inadequate stakeholders' involvement, communication and negotiation skills, ineffective techniques, time constraints, improper documentation and lack of requirements management, un-prioritized requirements, ambiguous and inconsistent requirements. However, with the analysis of those situations, it is shown that the base is the communication problem. Anyhow, solving these issues at some stage is necessary for the successful development of the software implementation of requirements.

Communication is not an ideal process of the requirements engineering, it continues throughout the whole life cycle. In the initial stage, most of the requirements are gathered and it should be strongly validated and communicated to the developers. Anyhow initial requirements could be changed. So, In order to make them reflected in the application, the development team should be notified immediately. Since the incorrect and inefficient communication led to the unsuccessful software applications, communication is one of the major factors to be considered for successful software development (Bjarnason, et al., 2013). The research has been conducted in an organization with working employees around 5000 and in the industry of embedded system development. Even though the communication is a challenge itself, there were good communications among the teams. The research was conducted on four identified factors which causes of several communication gaps and they are scale, common views, and temporal aspect and decision structures.

Due to the complexity and the size of the organization which is the scale factor, it cause for the communication gaps. When the size is becoming higher the complexity of communication becomes also higher comparatively. The factor, common views, which is different roles in the organization have different domain knowledge and different perspectives. Due to communication gaps, it could fail to identify the tacit knowledge by not communicating or ineffective communication. This would reflect throughout the requirements engineering process as well as the development phase.

They have identified that the requirements engineering should continue in the full life cycle of the project. The requirements are dynamic and changeable before they implement and even after the implementation. This comes under the factor, temporal aspect. The cause, week vision of overall goal covers the factor of decision structure. When there are no proper requirements to be implemented, the development team by itself in a trouble of which requirements to be included.

As focusing the results of the research, communication has become one of the major factors for the requirements (Bjarnason, et al., 2013). The study highlights that communication gaps can have serious and expensive results in terms of wasted effort and quality issues, as well as, not meeting the customers' expectations.

## **2.5. Documentation**

Mainly software requirements could be categorized into two parts which is Functional and Nonfunctional requirements. Software requirements specification is the document where it contains all the requirements for the system and it contains both the Functional requirements which are defined, what the system should do and Nonfunctional requirements which are defined how the system should work. So, the document is highly important when it comes to the development phase (Pa, 2011). Without having a proper documentation for the development phase, it could impact the project in many ways, such as

- Cost rework
- Budget overruns
- Poor quality systems
- Stakeholders' dissatisfaction
- Projects failure

Proper documentation solves those big issues and to have such accurate set of documents, both the communications and documentation should function properly (Pa, 2011).

Requirements engineering is all about collecting requirements from the stakeholder and keeps them in proper documents for future reference. The end product of the requirements engineering process is SRS. SRS is used in the system design and development. SRS contains that the technical approach to solve the problem. So, in order to make the application such a way that what the customer expected a better documentation process is a must (Michael, Boniface, 2014).

Documents are knowledge databases of any given industry, especially in the software industry (Falbo et al., 2012). It expresses what the system does and how it does that. In the software industry, documentation happens most of the time within the requirements engineering phase. There are several types of documents generated in the process of requirement engineering. These documents are normally read by the software engineers. In this research, it discusses the use of semantic annotations in



requirements documents, in order to make information regarding the links between requirements and other software artifacts, such as other requirements, use cases, classes and test cases, interpretable by computers. To do that, the paper extend a semantic document management platform to the requirements domain, and explore the conceptualization, established by the Software Requirements Reference Ontology in order to provide features to support some activities of the Requirements Engineering Process, namely: prioritizing requirements, analyzing impacts of requirements changes, tracing requirements through traceability matrices, and verifying requirements using checklists (Falbo et al., 2012).

## **2.6. Incomplete/Inadequate Requirements**

The application is based on what the requirements are. So, the importance of correct, complete and well documented requirements is really concerned (Michael, Boniface, 2014). There are many issues arising due to the inadequate requirements engineering and there are many reasons for that also. The research has identified below are few identified reasons for the inadequate requirements engineering.

- **Lack of End User Involvement:** It is highly important to participate in the requirements engineering phase by the end users, because they are people who knows the required functions than any other. So, their involvement would definitely cause for the completeness of the requirements.
- **Poor Planning and Estimation Processes:** It is too risky to have poor planning for a project. Without having a proper plan, it is too difficult to ensure proper estimation, clear resources management, risk mitigation budget and deadline.
- **Failure to Effectively Manage Changes to Scope:** Sometimes the previously proposed scope would not fit in with the current requirements.
- **Inadequate Man Power Resources:** If the developers are do not fit to the required skill level, it is too difficult to drive to the success.
- **Too long or Unrealistic Time Scales:** If the development phase becomes too long, that would lead to a product which is no longer valuable to the

customer. On the other hand, if the product owner expecting the delivery too early, that would lead to cutting down some necessary important features of the system.

- Failure to Adequately Identify and Document Requirements: If the design is kicked off too early without the core requirements having been adequately identified, documented, and agreed on by the users, the application is having a chance of producing the wrong product

In order to facilitate proper requirements engineering phase to eliminate unnecessary incompleteness below steps should be taken into consideration.

- Requirements Elicitation: In this phase, requirements are gathered initially.
- Requirements Analysis: The requirements are analyzed in order to identify inconsistencies, defects and omissions
- Requirements Documentation: This is the next phase, which is the end product of requirements elicitation and analysis. The documentation is very important as it would be the foundation for the design of the software.
- Requirements Review: To improve the quality of the SRS this review process is carried out.

The research has brought to the front the major factors responsible for poor software development among up-coming software developers in developing nations as incomplete requirements specifications as a result of poor involvement of the users (Michael, Boniface, 2014).

## **2.7. Prioritization**

Requirements prioritization is not a simple process. It is based on multiple decision criteria. Mainly, it should be processed by identifying highly important requirements having the discussion with stakeholder. Some of the other criteria would be, cost, quality, available resources and deliver time. Hence, requirements prioritization is

one of the main factors for the success of the SDLC outcome. It frees some obstacles of releasing the software applications in a single release (Achimugu, et al., 2014).

Stakeholder, prioritizing the requirements based on their comparative prominence through a scoring system. The comparison becomes more complex with the number of requirements. Prioritization is not only for the new requirements but also for the existing requirements too. There are several limitations for existing requirements prioritization techniques. They are,

- Scalability: Less support for manageability when number of requirements becomes higher than planned.
- Computational complexity: Hard to calculate the score for the priority list
- Rank update: Automatically update the score when inclusion new requirements or excluding existing requirements
- Communication among stakeholder: Most of the prioritization techniques do not support stakeholder communication
- Requirements dependencies: There could be requirements based on another requirements
- Error proneness: Prioritization techniques are not robust
- Lack of fully implemented requirements prioritization systems: Existing techniques has not been defined for real life scenarios

In order to make the prioritization process more productive, these challenges should be overcome (Achimugu, et al., 2014).

Requirements are demands or needs and requirements elicitation is all about understanding stakeholder needs (Chomal, Saini, 2015). Mainly requirements engineering includes

- Domain analysis
- Elicitation
- Specification
- Assessment
- Negotiation

- Documentation
- Evolution

The main issue of the requirements could be the quality of the requirements because incomplete or inadequate requirements would lead the software system for full of unnecessary implementations. It is highly recommended that the frequent communication with the stakeholder is important since misunderstandings, conflicts; uncertainty of requirements could be eradicated (Chomal, Saini, 2015). To overcome this situation, well organized requirements engineering process has been introduced by the research, that is,

- Domain Understanding: Domain understanding means, understanding of application domain to be developed.
- Requirement Collections: It is the course of action in which interaction between customers and users is done to discover the requirements of the system.
- Requirement Classification: After the requirements are gathered, they are clustered and classified.
- Conflict resolution: Many times, when the multiple clients are concerned requirements will conflict. Conflicts need to be determined keeping in mind the overall objectives.
- Prioritization: Here, requirements are identified and listed according to their importance.
- Requirements Validation: Check and validate the gathered requirements to see if they are complete, correct and sound

There are several factors affects for the requirements prioritization. With the constraints of cost, resources, or several other reasons, not all the requirements could be implemented at once. Instead, selected set of requirements are considered initially (Hassan, Awan, 2012).

The research was conducted with participating 17 different organizations. They have different types of application domain, such as Processing, Non Processing and IT professional. To get the current procedure, there were several interviews had been carried out by the researchers. There were several factors considered in this research to identify how the prioritization affected to the application development. They are stakeholder designation, age, leadership experience in years and qualification of the analyst.

The research found that these factors mitigate the risk of requirements prioritization. By practicing these factors within the organization, it can solve, relationship and communication problem. Any development which is practicing these factors can get the maximum out of the requirements engineering and would lead to a cost effecting solution (Hassan, Awan, 2012).

Due to some constraints, it is very difficult to decide which requirements should be implemented first. There are few constraints, such as resource availability, cost and time (Khan et al., 2015). If the set of requirements are higher, all can't be implemented at once. So, there should be discussions happen in between the organizations and the stakeholder to make the priority list. When the number of stakeholders becomes higher, and then again setting the priority is also a much difficult task. The research has identified several methods of requirements prioritizations and they are,

- Analytic network process (ANP),
- Analytic hierarchy process (AHP),
- Hierarchy AHP
- Spanning tree matrix
- Bubble sort
- Binary search tree
- Priority groups

After analyzing the set of above methods of prioritizing the requirements, it is shown that the AHP is the most reliable and fault tolerant mechanism of prioritizing the requirements because, based on a scale it gives weightages to the tasks. The requirements are ordered according to those given weightages (Khan et al., 2015).

During the requirements prioritizing process, there should be a way of finding the core business requirements to be implemented since all the requirements are not capable of facilitating once due to certain circumstances. Requirements prioritization is more important when it comes to the success of the project output. With the available time, budget the prioritization is highly important. Not only that, there are other factor also. They are risk, dependency with other tasks, technical challenges, customer need, etc. Selecting set of requirements out of the entire task is much challenging. In this research, it has been evaluated the previously mentioned frameworks of requirements prioritization and finally the research suggest a new framework that will rank the requirements by the relative level of value, cost, effort and threat associated with each requirements (Goel, Thakur, 2013).

When it consider the incremental delivery of software applications, requirements prioritization is more important. Choosing a set of requirements from a large set of tasks list is not an easy mechanism. Due to that, there should have a series of negotiation rounds to prioritize requirements (Gupta et al., 2013). The software industry is more dynamic, due to that quality of the industry; requirements are also going to be dynamic. What had been prioritized earlier is not always going to be the set of requirements to be implemented by software engineers. So, it always needs to be considered re-prioritization of requirements.

The research by Mr Gupta proposed a framework which should be considered when re-prioritization occurs. The proposed approach involves considering the three cases as given below.

- a) For new requirements:
- b) For Already Implemented Requirements:
- c) Delayed Requirements:

For all these scenarios, the priority level is calculated by an equation. The research has been done in a case study method. The considered system is a library management system. Before the system begins, all the requirements are list down and based on the initial priority level implementation starts. To start the second phase there are several considerations taken into account as below.

- No new change request has emerged in the meantime.
- No new requirements have emerged.
- Values of heuristic parameters i.e. parameters of adjustment factors are collected.
- The average density is computed for each category.

Hence, this paper proposes the multilayered dynamic prioritization method that could perform the task of re-prioritization and prioritization of software (Gupta et al., 2013).

It is set that the requirements prioritization methods are very important in the current software industry (Batra, Bhatnagar, 2016). Due to the importance of this topic many researchers are conducting in this area of study to produce some better results than existing. This research is fully based on analysis of previous researches and produce to do list of future works. It is below as mentioned in the research.

- Literature review depicted that incorporation of fuzzy logic in requirements prioritizing generates very effective results.
- The researchers might propose an approach that deals with prioritization of non-functional requirements as non-functional requirements are equally important.
- Researchers might work in the area of requirements re-prioritization.
- Researchers might propose a methodology that includes fuzzy logic for prioritizing requirements that produces more effective and reliable results.

- Researchers might proposed a single method that addresses all the factors necessary for prioritization process i.e. cost, benefit, risk, effort, easy to learn, easy to use, confidence, understandable to non-experts, reliable, efficient, scalable and flexible etc.
- Researchers might work for developing prioritization techniques that work well in large scale and dynamically changing requirements.
- Researchers might propose an approach to select an appropriate prioritization method for the particular application. This approach may be employed for real life applications.

As found in the previous researches, there are few parameters identified based on the current literature to be further analysis of future works. They are, cost, effort, risk, understandability, reliability, efficiency, scalability and flexibility. (Batra, Bhatnagar, 2016).

Security is also a major factor when prioritize the requirements. Today, security is not a factor which should be excluded when implementing software applications. Security factor should be considered in all most all the phases of the SDLC. CERT is also reporting the rising of software vulnerability cases in the past few years. So that the software engineers should be highly concerned regarding this factor and should be facilitated high security measures to their systems (Gulati et al., 2012). As the research mentioned the importance of the security concerns should be discussed in the requirement engineering process.

Requirements are always uncertain (Voola, Babu, 2012). In addition to the other basic approach to prioritize the requirements, the requirements uncertainty prioritization approach should also be considered. The introduced prioritization approach within this research, express, how incomplete, uncertain, imprecise and conflicting goals of stakeholders are transformed into an ordered list of requirements. The approach is shown as a sequence of activities as described below in 5 steps.

1. Identify relevant stakeholders for the project under consideration and assign weights.



2. Obtain inputs from stakeholders with Extensive Numerical Assignment.
3. Aggregate inputs collected applying Interval Evidential Reasoning Algorithm.
4. Use Utility theory to compute ranks.
5. Distribute outputs to the stakeholders, obtain degree of satisfaction and reach consensus.

Each step has its own responsibilities and functions. First step highlights on the issue of having one stakeholder in the project. It is said that the importance of a group of stakeholder involvement to the requirements prioritization phase is well recognized as the quality of the final software product is determined verily by the accurate identification of stakeholders and their needs. The next step is to get input from stakeholders for prioritization of requirements. Numerical Assignment (NA) is the most common, simple and fast prioritization technique but may not provide precise and reliable results. It is based on ordinal scales where requirements are grouped into some priority groups. Then the thinking is to transform Numerical Assignment into Extensive Numerical Assignment (ENA) with a rich set of inputs, flexible enough to accommodate imprecision. Then through a very complex mathematical algorithm is used to compute ranks for the each task (Voola, Babu, 2012).

## **3. METHODOLOGY**

### **3.1. Chapter Overview**

The purpose of the research is to find the impact to the software development phase due to four identified factors in the requirements engineering phase. The research is a study in the Sri Lankan banking software industry. Four factors were identified through a literature review and those factors were tested via interviews with requirements engineers and software engineers. Throughout the chapter the research design and the approach to the final result has been illustrated via several topics and subtopics.

### **3.2. Introduction to Methodology**

The research was conducted through a very comprehensive literature review with the selected research articles from year 2011 to 2015 and they are based on different problems addressed in related to the requirements engineering stage. Among the selected titles, there are research topics on security testing of SDLC, Emerging trends for an SDLC, requirements engineering models, the role of requirements engineering in SLDC, impact of requirements management, impact requirements engineering on software development cost, security measures of software requirements engineering, Inadequate Requirements Engineering Process, etc.

Based on the above research papers, four factors were identified and to be proved that those factors are highly impact to the software development through requirements engineering a qualitative questionnaire was developed. The questionnaire was developed separately to requirements engineers and software engineers, including questions which measure the effectiveness of those four factors. Using the grounded theory as the analysis techniques, answers were measured and analyzed finally produced the final outcome.

The target populations for this research are requirements engineers and the software developers working in the software organizations who are developing software for banking domain as a product in Sri Lanka. Out of the companies functioning in that

domain, filtered out only the organizations develop software for banking domain as a product not as a project. Out of the two main methods of research methodologies which are quantitative and qualitative, the most appropriate way for this research is qualitative. One of the main reasons for selecting the qualitative method is that the target population is comparatively low. Apart from that, the research is factor based; data collected via set of interviews was also a reason for that decision.

### **3.3. Interview Structure**

It is known that the interview is one of the best options to secure the essential information, especially for qualitative research. But, on the other hand, it is somewhat challenging and time consuming process when the detail questions are answered. So, the questionnaire should be planned and structured, focusing only on the relevant topic and fact, not to unnecessary detail (Alshenqeeti, 2014).

When the research is planned, it has been considered below points,

Communication mode/method: *The selected communication method is the face to face interviews but, based on the availability of the few interviewees there were few interviews happened over the phone.*

Conducting interview: *Interview process starts after giving a brief introduction to the topics and the introduction to the questionnaire as well. They were informed that the answers are anonymous, which is no organization or personal detail disclosed to a third party or include in the report.*

Length of the interview: *The total length for an interview limits maximum 20 minutes.*

Reliability: *Make sure the interviewees should have better knowledge about the problem domain.*

Design the questionnaire: *To cover up the full problem domain, a literature review was the base for to build up the questionnaire.*

Analysis of the data: *Chose the qualitative analysis since the research is factor based and the grounded theory is used as the method of data analysis.*

### **3.4. Questionnaire Design**

To design the questionnaire, there were two main sections considered. They are requirements engineering and software development. So the target audience was mainly two separate groups. Apart from that, basic introduction to the organization and questions to gather interviewee data was also included.

Section 1 - About the organization: This question brings the starting point to the other sections. This will help to find the basic information of the organizations such as, Clients, projects, number of employees etc.

Section 2 - Professional profile: Since, the requirements engineering is the highest priority of this research, it is highly necessary to acquire the background of the requirements engineers such as, their professional life, experience, etc.

Section 3 - Factor based questions for requirements engineers: This set of questions used to collect the research data which is relate to the requirements engineering.

Section 4 - Factor based questions for software engineers: This set of questions used to collect the research data which is related to the software engineers and this will gather how the factors affect to them after the requirements gathering.

### **3.5. Population and Sampling**

There were 11 organizations selected for the research. They were selected based on several criteria. They are as, registered companies in the Sri Lanka Association of Software and Service Companies (SLASSCOM), from the market analysis and gathering information from Sri Lankan public and private banks.

Out of the 11 selected companies, 8 organizations participated in the research and 3 of them did not willing to participate for the research. Out of the participated organizations, 5 organizations have certifications, either, CMMI level 3 or ISO 9001 but, 3 organizations do not have any certification with them.

The research is mainly based on requirements engineering and software development stages. So, the interview process divided into two parts. One is interviewing requirements engineers, and the second part is interviewing software engineers. For

the research, there were 18 interviews for requirements engineers and 27 interviews for software engineers.

Below is the summary of the interviewees.

Table 3.1: Interviewees Description

Organization	Job Category	No of participants
Organization 1	Requirements engineers	3
	Software engineers	4
Organization 2	Requirements engineers	3
	Software engineers	5
Organization 3	Requirements engineers	2
	Software engineers	2
Organization 4	Requirements engineers	3
	Software engineers	4
Organization 5	Requirements engineers	3
	Software engineers	5
Organization 6	Requirements engineers	2
	Software engineers	2
Organization 7	Requirements engineers	1
	Software engineers	3
Organization 8	Requirements engineers	1
	Software engineers	2

### 3.6. Interview Preparation

Right after the finalization of the questionnaire, interview preparation was started. As already mentioned 11 organizations were selected based on the criteria and 8 organizations were agreed to participate in the interview process. At the very first time the informal request was made over the phone and initial formal interview request email sent (Appendix B) along with the summary of the research (Appendix

D) and with the approval letter from the university (Appendix C). Hereafter several discussions were going on over the phone and a date was fixed based on the availability of the both parties. Only 2 organizations participated in the research over the phone and 1 organization participated via paper work and all the other 5 organizations participated in the interview in their premises.

There were several constraints had to be overcome in the interview process, such as, availability of both parties, organization policy to espouse of the internal information, project deadlines and time for an interview.

### **3.7. Interview Execution**

As mentioned in a previous section, there are four main sections in the research questionnaire. First two sections, regarding the, basic information about the organization and the interviewee. About the organization sections just to understand the organization and build the relationship between both parties. It was conducted via the HR or the Administrative departments of the particular organization. Then the second section of the questionnaire is to have the background of the professional profile of each interviewee.

The third section of the questionnaire only for the requirements engineers and the fourth section of the questionnaire are only for the software engineers.

### **3.8. Data Collection**

Data was collected from the period of 18<sup>th</sup> of January 2017 to 24<sup>th</sup> of February 2017. The data collection approach was qualitative and it contains detailed answers. All the data would be anonymous and it would be organizational perspective and personal perspective too. Data was collected in a three ways, such as, interview, papers and over the phone. Collected data were summarized in a way that could be easily interpreted to meaningful information.

### **3.9. Data Analysis Techniques**

Since the research is based on the factor analysis, a qualitative approach was selected as the data analysis technique which is the Grounded theory.

Based on the grounded theory, proposed by Strauss and Corbin in 1990 the process of analyzing the data in a qualitative research consisted with organizing the data, breaking it down into descriptive smaller sections, searching for similar patterns, etc.

The idea behind the theory is to build the codes (Open, Axial and Selective) based on the collected data from the interview process.

Data analysis divided into three main steps such that, open coding, which is categorized collected data into conceptual categories. Then the axial coding, categorized into more narrow categories which is derived from the open coding. From the axial coding, it summarizes the more prominent category called selective category which describe the identified factors (Lawrence, Tar, 2013).

## 4. DATA ANALYSIS

### 4.1. Chapter Overview

This chapter expresses the data collection and it produces the detailed output of the analysis. For the data analysis, the research uses the grounded theory which is useful in qualitative data analysis.

### 4.2. Introduction

Since the research is going to be a qualitative and to test the reliability of the questions two organizations were selected initially and conducted a preliminary interview. Based on the given answers and further analysis of the questions, the questionnaire was refined and it contains mainly four sections which are, About the organization, Professional profile, Factor based questions for requirements engineers and Factor based questions for Software engineers. With the refined questions, the next round of interview process started.

Questions were based on four identified factors, they are, Communication, Incomplete/Inadequate requirements, Requirements prioritization and Documentation. To be tested the each factor and how it is being affected to the software development phase, questions were consisted separately to the requirements engineers and software engineers.

Below two tables show how the questions were focused on each factor in each job role.

Table 4.1: Questions for requirements engineers and its dimensions

<b>Factor/Variable</b>	<b>Dimensions</b>	<b>No of questions used to measure the factor</b>
Communication	Communication Language and Method	2
	Understandability	2
Inadequate/Incomplete	Requirements verification	1



Requirements	Requirements clarification	1
Requirements Prioritization	Criteria of prioritization	1
	Responsibility of prioritization	1
	Change of the prioritization	1
Documentation	Types of documents	1
	Preparation of documents	1
	Responsibility of document preparation	1
	Document verification	1
	Change of documents	1

Table 4.2: Questions for software engineers and its dimensions

<b>Factor/Variable</b>	<b>Dimensions</b>	<b>No of questions used to measure the factor</b>
Communication	Clearness of the requirements.	2
	Understandability	1
Inadequate/Incomplete Requirements	Time to understand	1
	Requirements clarification/verification	2
	Change of requirements	2
	Change management	4
Requirements Prioritization	Order of tasks	1
Documentation	Understandability of the content	1
	Preparation of documents	1

### 4.3. Response to the Research

As mentioned in the chapter 3 total sample for the research was 45, out of 45 there were 27 software engineers and 18 requirements engineers. As mentioned previously in chapter 3, 5 organizations having industry certification and 3 do not. Out of the total requirements engineers' interviews, it can be divided requirements engineers from certified organizations were 14 and others were 4, and software engineers from certified organizations were 20 and others were 7.

Table 4.3: Summary of organizations

<b>Category</b>	<b>Number of Respondents</b>	<b>Percentage of Respondents (%)</b>
Respondent organizations	8	72.72
Not respondent organizations	3	27.27

Below table displays in detailed view about the respondent.

Table 4.4: In detailed view of results summary

<b>Category</b>	<b>Number of respondents</b>	<b>Total percentage of respondents (%)</b>	<b>Percentage of respondent from the same category</b>
Requirements engineering (Certified Org)	14	31.11	77.77
Software engineers (Certified Org)	20	44.44	74.07
Requirements engineers (Not Certified Org)	4	8.88	22.22
Software engineers (Not Certified Org)	7	15.55	25.92

#### **4.4. Data Analysis**

There were 13 questions raised on software engineers and another 13 questions raised on requirements engineers. Based on the given answers and with the analysis of data, results were produced. Appendix E contains the data that are related to the requirements engineers and the Appendix F contains the data that are related to the software engineers. In this chapter data of the Appendix E and Appendix F referred for data analysis. To give a numerical value to the each category, a weightage system introduced and below is the scale level table.

Table 4.5: Weightage system summary

Weightage Description	Rank
Strongly Disagree	1
Disagree	2
Neither Agree Nor Disagree	3
Agree	4
Strongly Agree	5

#### 4.4.1. Numerical Calculation

If an organization ranked above or equal to 3 (Neither Agree Nor Disagree), that is considered as positive remark for the calculation and if an organization ranked below or equal to 2 (Disagree) that is considered as negative remark in the particular category. So, based on the rank total positive value is taken and divided by 8 which is the total number of organizations and multiplied by 100 to take the percentage of that particular value. I.e.

Number of organizations with positive impact (rank above or equal 3) = 6

Number of total organizations = 8

Total Percentage (P) =  $(6/8) * 100 = 75\% (> 50\%)$

So, that factor should be considered as a positive factor to the requirements engineering because the percentage is greater than 50%.

#### 4.4.2. Requirements Engineering

Below table summarize the data and factors, collected from requirements engineers.

Table 4.6: Weightage summary – Requirements Engineers

Selective Coding	Category	Org 1	Org 2	Org 3	Org 4	Org 5	Org 6	Org 7	Org 8	Percentage (%)
Offline communication	Communication	5	5	5	5	5	5	5	5	100
Online communication	Communication	5	4	4	4	5	2	2	3	75
Direct communication	Communication	5	5	5	5	5	5	5	5	100

ion										
Indirect communication	Communication	5	4	4	4	5	2	2	3	75
Paper based documents	Documentation	5	5	5	5	5	5	5	5	100
Internal communication	Communication	5	5	5	5	5	4	4	4	100
External communication	Communication	5	5	5	5	5	5	5	5	100
Online documents	Documentation	5	4	4	4	5	3	3	3	100
External documents	Documentation	5	5	5	5	5	4	4	4	100
Internal documents	Documentation	5	5	4	4	4	3	3	3	100
Re-communicate	Incomplete/Inadequate requirements	5	5	5	5	5	5	5	5	100
Re-collect requirements	Incomplete/Inadequate requirements	5	5	4	5	5	5	5	5	100
Re-complete requirements	Incomplete/Inadequate requirements	5	5	4	5	5	5	5	5	100
Re-analyze requirements	Incomplete/Inadequate requirements	5	5	4	5	5	5	5	5	100
Requirements ranking	Requirement prioritization	4	4	4	5	5	3	4	3	100
Re-order ranking	Requirements prioritization	4	4	4	4	5	3	4	3	100
Prepare technical documents	Documentation	5	5	5	5	4	2	2	3	75
Prepare non-technical documents	Documentation	4	4	5	5	4	3	3	3	100
Offline documents	Documentation	5	5	5	5	5	3	4	3	100

As in the Table 4.6: (Weightage summary – Requirements Engineers) there are 19 selective coding identified from the collected information on requirements engineers. In the column 2 it has been categorized into the relevant category which has been a factor in this research. It has been ranked from 1 to 5 based on that selective coding

and how it relates to the each organization. In the final column total calculation taken based on the formula in Chapter 4.4.1

### 4.4.3. Software Engineers

Below table of data summarizes the collected data for software engineers.

Table 4.7: Weightage summary – Software Engineers

Selective Coding	Category	Org 1	Org 2	Org 3	Org 4	Org 5	Org 6	Org 7	Org 8	Percent age (%)
Refer documents	Documentation	5	5	5	5	5	4	5	5	100%
Offline communication	Communication	5	5	5	5	5	5	5	5	100
Understand the documents	Documentation	5	5	5	5	5	3	3	3	100%
External communication	Communication	5	5	5	5	4	4	4	5	100%
Internal communication	Communication	5	5	5	5	5	5	5	5	100%
Miscommunications	Communication	4	4	5	5	4	5	5	4	100%
Reevaluate documents	Incomplete/Inadequate requirements	5	4	4	5	4	4	4	4	100%
Requirement reprioritization	Requirements prioritization	4	5	5	4	5	3	3	4	100%
Reassert requirements	Incomplete/Inadequate requirements	4	5	5	4	5	2	2	3	75%
Reintegration	Incomplete/Inadequate requirements	3	3	4	5	5	2	2	3	75%
Retesting	Incomplete/Inadequate requirements	4	4	4	4	5	3	2	3	87.5
Redevelopment	Incomplete/Inadequate requirements	4	4	5	5	4	4	4	4	100%

### 4.5. Data Interpretation

There were totally 45 interviews of requirement engineers and software engineers in 8 organizations. Questions were mainly based on four identified factors. Throughout

the research, it has been identified that those factors should be highly considered in requirements engineering. It has been clearly elaborated below with the extraction from the interview answers (Appendix G and H) by different interviewees.

#### **4.5.1. Detail Analysis**

There were few questions asked from the interviewees based on the communication factor, separately from the requirements engineers and the software engineers. For the first question which was to understand what languages and methods they used to communicate to the client, the most common answer was that they use Sinhala and face to face interviews. Sometimes the language was depending on the client, it shows from the answer from organization 3 which is:

[“Language is based on the client and it will be face to face most of the time.”]

Most of the organizations use face to face interviews and apart from that they use Skype calls, conference calls, referring existing documents, email and phone calls. As a continuous question to the first questions, the second question was asked. It was to monitor how they gather initial requirements of a particular project". The answer was face to face to interviews. As a percentage of that answer, it was 100%. First two questions show that all the organizations used communication as a practice and if that is divided into sub categories, it can be seen like online or offline communications and direct or indirect communication.

As proved in the requirements engineering, software engineers also used communication factor within their process. In the first question for software engineers shows that most of the organizations use offline communication for the development of the software application. Most of the answers were that they are referring some prepared documents from the requirements engineering process which is SRS. Other than the SRS, they refer memos and any notes as a referring material. So the betterment of the software engineers in the development phase, requirement engineers should be having better communications with the stakeholders and then collect requirements and make sure to arrange them in a proper order.

One of the other main factors is incomplete requirements/Inadequate requirements. Question 6 – “When do you need to talk to the client again for the same requirement” was mainly focusing on that factor. Among the few answers from different organizations following can be listed.

[“If the requirement is not clear”]

[“We talk to the client to get missed points”]

[“Is the requirement is not enough for time estimation”]

[“When the requirements are not clear to implement or develop”]

Question 9 – “When do you need to change the priority” for requirement engineers was also measured this factor. One of the interviewees of organizations 6 mentioned that they had to change the priority due to that they have gathered incorrect requirements. One of the tasks in the requirements engineering is documenting them. Sometimes the completed documents need to be changed due to an incomplete or incorrect gathering of requirements. Answers for the question 13 in the requirements engineering section capture that some of the changes may due to that factor.

Question 5 in the section of software engineers prove that, they have to request for the verifications due to several reasons. Among them below could be emphasized.

[“When the requirement is not clear”]

[“If the requirement is not clear enough to develop”]

[“If no one can understand the requirements”]

[“To understand the missed things”]

So that they need to spend additional time for the changes and sometimes they wait for the verifications. So, it seems that this factor is more important in requirements engineering.

Requirement prioritization is also measured in this research. Questions 7 and 9 of the requirement engineering section dedicated to the gathering of information regarding



this factor. Almost all the organizations consider this factor. It is proved that through answers from each organization, that they consider customer requirements to make the priority order. Other than the customer requirements, they consider technical aspect, the size of the task and resource availability. So in order to make the requirement engineering process more smooth requirement prioritizations factor should be highly concerned.

The last factor that has been selected for the research is documentation. Documentations are much more important in requirement engineering since the completed documents are referred by the software engineers for the development phase. Documentations are also important to the requirement engineers while they are in the process of gathering requirements from the stakeholder. As the answer to the question 3 one of the interviewee in the organization 3 mentioned that they refer to existing documents to analyze the requirements.

Not only using existing documentation for the requirement engineering, but also as an output of the phase, requirement engineers produces several documents, such as SRS, BRS, Mock-ups and flow charts.

Preparing documents is not an individual task and there are several people who involved in this process, such as project managers, developers, technical teams and QA. After completing the documents, there are separate processes to verify them, like discussion with project managers, meetings with technical teams, involve stakeholder for the discussions, and internal discussions within the QA and development teams.

Mainly the completed documents are referred by the software engineers. For the implementations they use SRS, functional specifications, UML diagrams, etc. Even though they do not get adequate time to read those documents, they get enough time to read and start the initial developments.

With the analysis of the answers to those questions by requirement engineers and software engineers it is recommend that those factors should be considered within

the requirement engineering process and there is a high impact of those factors to the software engineering phase of the SDLC.

#### 4.5.2. Data Interpretation Summary

According to the Table 4.6 there are 19 selective coding under four main factors. The importance of all these selective coding are greater than 50%, which means for the requirements engineering they should be considered. Selective coding for each factor can be expressed as below table.

Table 4.8: Relationship of selective coding and factors – requirements engineers

Selective coding	Factors
Offline communication	Communication
Online communication	
Direct communication	
Indirect communication	
Paper based documents	
Internal communication	
External communication	
Online documents	Documentation
External documents	
Internal documents	
Prepare technical documents	
Prepare non-technical documents	
Offline documents	
Re-communicate	Incomplete/Inadequate requirements
Re-collect requirements	
Re-complete requirements	
Re-analyze requirements	
Re-analyze	
Requirements ranking	Requirements prioritization

Re-order ranking	
------------------	--

Below table displays how the each selective coding relates to the factors for software engineers.

Table 4.9: Relationship of selective coding and factors – software engineers

Selective coding	Factors
Offline communication	Communication
External communication	
Internal communication	
Miscommunications	
Refer documents	Documentation
Understand the documents	
Reevaluate documents	Incomplete/Inadequate requirements
Reassert requirements	
Reintegration	
Retesting	
Redevelopment	
Requirements reprioritization	Requirements prioritization

Each selective coding in both the Table 4.8 and 4.9 got importance greater than 50%, which means they are very much considerable to the requirements engineering and affects to the software engineering as well.

#### 4.6. Summary

As identified via the literature review, four factors are very crucial to the SDLC, especially to the requirements engineering and implementation phases. Further, with the analysis of data based on those facts it recommends that that they are very much important to the banking software domain too. Since the selected factors are not common to the banking software domain, they can be considered to the other application domains as well.

## 5. CONCLUSION AND RECOMMENDATIONS

### 5.1. Chapter Overview

This research is mainly focusing on the requirements engineering and how it affects to the software development phase of the SDLC. Above chapters have gone through different topics of research and finally this chapter summarizes the content in a more abstract way.

### 5.2. Conclusion

The research was conducted on eight organizations which are in the software industry, developing application into banking and finance domain as a product. Five organizations out of the eight are having one or more industry certifications, but other three does not have any.

Research was conducted to identify the effectiveness of four identified factors which is Communication, Incomplete/Inadequate requirements, Requirements prioritization and Requirements documentation to the development stage from the requirements engineering phase. Through a literature review those factors were identified and tested through a questionnaire separately to requirements engineers and software engineers.

In the previous chapter, the collected data were analyzed and showed results on them. Based on the results, it is proved that those factors are very much important in the requirements engineering phase and they would effect on the development phase as well.

There are many critical factors in the requirements engineering. Among them, this research has identified Communication, Incomplete/Inadequate requirements, requirements prioritization and documentation factors are much more important in the requirements engineering phase of the software SDLC. The outcome of those factors affects to the development phase of the SDLC.

Most of the software organizations have identified that those factors are key to their success, but few of organizations still not much consider them in their process. Table 4.6 indicates that how the factors are referred by requirements engineers and the Table 4.7 depicts that how those factors affected by software engineers. I.e. in Table 4.7, research found that the “Reevaluate documents” is 100% due to incomplete/inadequate requirements engineering phase. So the time of both the requirements engineers and software engineers is unexpectedly exceeded due to such scenarios.

Similarly, Table 4.6 and Table 4.7 build a good relationship among those factors and ensure that they are the few of the key factors of the requirements engineering.

Because of that, the ultimate objective of this research which to find important factors of requirements engineering and how those factors are related to the success/fail of the development phase has been identified and proved.

### **5.3. Recommendations**

There are several factors which should be considered when it comes to the requirements engineering. Among them Communication, Inadequate/Incomplete requirements engineering, Requirements prioritization and Requirements documentations get much attention due to its high importance. Through this research it has been proved that those four factors are much important when addressing to application development.

Communication is a very sensitive factor of the requirements engineering. There are no restrictions on this factor, but managing the way of communicating, language, method of communicating would further improve the quality of the application development.

The study identified that the gathering correct, complete and consistent requirements is also much important for the requirements engineering. Communication factor is also the base to improve the quality and correctness of the requirements.

Requirements prioritization is also a very crucial factor when consider the success of the product. Not all the requirements are needed at once or in the initial stage.

Therefore, the task list should be ordered such a way that, application should be developed without any privatization issue. The order of the task list may depend on the, client's requirements, resource availability, dependency of the tasks, etc. So, all the factors should be considered when the process of requirements prioritization.

The research also found that the proper documentation process would also be an important factor for the success of the software development. Software engineers refer the documentation for their development process. So, in order to make the development process more accurate and clear, it is beneficial to have a good way of managing a proper documentation process

#### **5.4. Future work**

It is known that the topic is a broad subject that needs to be explored further. However, the study only focuses on the Sri Lankan Banking software domain only. But there might have international organizations on the same domain too. By conducting a research on them could produce a better result and that could be further help on the local organization too.

This research was not included all the software organizations due to some reasons such as do not like to reveal their data outside, according to their business domain and organizations polices, they were not willing to participate for the outside research works etc. So in order to gather more information on this software domain, it is better to get them to join and do the research in a very broad way.

There might have local or international software organizations situated in Sri Lanka who is developing banking and financial software applications as a project work. But, in this research, it has been considered only the organizations, which are developing applications as a product. Involving them to the research might also produce different results. It will be another path to study further as a future work.

As a limitation, the study only focuses on the software development stage impacted from requirements engineering phase. But, other three phases also would be affected on the requirements engineering phase. As a future work, that would be fine to conduct research on those phases also.

Finding facts on the client's perspective also much important area of study. This research is on the software organizations' perspective. A Stakeholder is also having much impact due to improper implementation due to issues in the requirements engineering phase. Because of that, the path is open to further analysis.

## References

- Achimugu P, Selamat A, Ibrahim R, Mahrin M,N. (2014). A systematic literature review of software requirements prioritization research. *Information and Software Technology*.
- Ahmad R, Tahir A, Kasirun Z.M. (2012). An empirical assessment of the use of different communication modes for requirement elicitation and negotiation using students as a subject. *IEEE Symposium on Computers & Informatics*.
- Alshenqeeti H. (2014). Interviewing as a Data Collection Method: A Critical Review. vol(1927-6028). ISSN: 1927-6028.
- Barraclyffe M, Gardner L, Hammond J, Duncan S. (2009). *Systems Development Life Cycle (SDLC) Methodology*. Information Technology Services.
- Batra M, Bhatnagar A. (2016). Requirements Prioritization: A Review. *International Journal of Advanced Research in Science, Engineering and Technology*, vol(3), ISSN: 2350-0328.
- Bjarnason E, Wnuk K, Regnell B. (2013). Requirements Are Slipping Through the Gaps. A Case Study on Causes & Effects of Communication Gaps in Large-Scale Software Development.
- Chomal V.S, Saini J.R. (2015). On Prioritizing and Freezing of Software Requirements. *International Journal of Innovative Science, Engineering & Technology*, vol(2), ISSN: 2348 – 7968.
- Falbo R.A, Eduardo C, Machado B.N. (2012). *Semantic Documentation in Requirements Engineering*.
- Goel L.B, Thakur S. (2013). Requirement Prioritization: A Study and Proposed Framework. *International Journal of Emerging Technology and Advanced Engineering*, vol(3), ISSN: 2250-2459.



Gulati A, Sharma S, Mehmi S. (2012). PROPOSING SECURITY REQUIREMENT PRIORITIZATION FRAMEWORK. International Journal of Computer Science, Engineering and Applications (IJCEA), vol(2), DOI : 10.5121/ijcsea.2012.2303.

Gupta V, Chauhan D.S, Dutta K, Gupta C. (2013). Requirement Prioritization: A Study and Proposed Framework. Requirement Reprioritization: A Multilayered Dynamic Approach, (Vol. 3, pp.55-64).

Hassan S.U, Awan S.A. (2012). Software Engineering: Factors Affect on Requirement Prioritization. Global Journal of Computer Science and Technology Software & Data Engineering, vol(12), ISSN: 0975-4350.

Huzooree G, Ramdoo V.D. (2015). A Systematic Study on Requirement Engineering Processes and Practices in Mauritius. International Journal of Advanced Research in Computer Science and Software Engineering, vol(5), ISSN: 2277 128X.

Kaur, S. (2015). A Review of Software Development Life Cycle Models. International Journal of Advanced Research in Computer Science and Software Engineering, vol(5), ISSN: 2277 128X.

Khan J.A, Rehman I.U, Khan Y.H, Khan I.J, Rashid S. (2015). Comparison of Requirement Prioritization Techniques to Find Best Prioritization Technique. I.J. Modern Education and Computer Science. DOI: 10.5815/ijmecs.2015.11.06v.

Lawrence J, Tar U. (2013). The use of Grounded Theory Technique as a Practical Tool for Qualitative Data Collection and Analysis. Software Engineering : An International Journal (SEIJ) (Vol. 11 pp 29-40). ISSN: 1477-7029.

Michael K.A, Boniface K.A. (2014). Inadequate Requirements Engineering Process: A Key Factor for Poor Software Development in Developing Nations: A Case Study. World Academy of Science, Engineering and Technology International Journal of Computer, Electrical, Automation, Control and Information Engineering, vol(8).

Pa N. C, Zin A.M. (2011). Requirement Elicitation: Identifying the Communication Challenges between Developer and Customer. International Journal on New Computer Architectures and Their Applications, vol(1), ISSN: 2220-9085.

Pandey D, Pandey V. (2012). REQUIREMENT ENGINEERING: AN APPROACH TO QUALITY SOFTWARE DEVELOPMENT. Journal of Global Research in Computer Science, vol(3), ISSN: 2229 371.

Ravi V. (2008). Introduction to Banking Technology and Management. Institute for Development and Research in Banking Technology, India.

Swarnalatha, K. S., Srinivasan, G. N., Dravid, M., Kasera, R., Sharma, K. (2014). A Survey on Software Requirement Engineering for Real Time Projects based on Customer Requirement. International Journal of Advanced Research in Computer and Communication Engineering, vol(3), ISSN: 2278-1021.

Suraweera T., Kahingala S., Batepola A., Punchihewa M.G., Senevirathne K., Kahandawaarachchi C. (2014). IT Driven Banking Services in Sri Lanka: Customer Acceptance and Service Quality.

SLASSCOM. The Sri Lankan IT-BPM Industry. Retrieved from <https://slasscom.lk/sri-lanka-advantage>.

Voola P, Babu V. (2012). Requirements Uncertainty Prioritization Approach: A Novel Approach for Requirements Prioritization. Software Engineering : An International Journal (SEIJ), vol(2).

## Appendix A: Questionnaire

### ABOUT THE ORGANIZATION

---

1. Brief summary about the organization?

### PROFESSIONAL PROFILE

---

2. How many years do you have experience in the IT industry?
3. What is your current designation and experience (years) in that designation?
4. What is your highest educational qualification (i.e. BSc in Information Technology, MSc in Computer Science, MBA in Business Analyst, Dip in Human Resource, etc.)?
5. How many years you have been involving in requirement engineering process for banking software domain?

### FACTOR QUESTIONS FOR REQUIREMENT ENGINEERS

---

6. What languages/methods (I.e. Phone call, Interview, Skype, E-mail, Chat message, other) do you use to communicate to the client and which method is easier/difficult for you?
7. How do you gather initial requirements?
8. How do you clearly understand the requirements?
9. How do you verify the requirements and with whom?
10. How do you suggest alternatives to the client?
11. When do you need to talk to the client again for the same requirement?
12. How do you prioritize requirements?
  - a. What criteria do you use to prioritize requirements?
  - b. With whom do you discuss when prioritizing requirements?
  - c. When do you need to change the priority?
13. What documents do you prepare before the developments?

14. How do you prepare documents for the collected requirements?
  - d. With whom do you prepare the requirement documents?
  - e. How do you verify the prepared documents and with whom?
15. For what reasons do you have to change the completed documents?

## FACTOR BASED QUESTIONS FOR SOFTWARE ENGINEERS

---

16. How do you get clear/complete requirements before the development begins?
17. How clear you understand the given requirements?
18. How do you refer/understand requirements/documents for the development?
  - f. How do you get adequate time to read/understand those requirements/documents?
19. How do you verify the requirements/documents with the requirement engineers?
  - g. For what reasons, do you have to request for requirement verifications?
  - h. How do you spend the time until you get verification?
20. For what reasons do you get requirement changes?
  - i. How do you manage your time with requirement changes?
21. How do you get additional time for changes?
22. How do you extend the deadline for requirement changes?
23. How do you explain the re-development cycles due to not getting the right requirements?
  - j. How do you test your existing functions due to redevelopment?
24. Any real situation of success/failure, due to requirement engineering?
25. Any additional thing you have to express for the impact due to requirement engineering?

## **Appendix B: Interview Request Email**

Dear Sir,

I, K.G.W. Rangana (841493664V), a student of University of Moratuwa at Department of Computer Science and Engineering, currently enrolled in Master of Business Administration 2015 batch under the index number of 159129V.

I am conducting my research on the topic of “IDENTIFYING FACTORS IN REQUIREMENT ENGINEERING PHASE AND IMPACT THEREOF ON SOFTWARE DEVELOPMENT PHASE: A STUDY ON SRI LANKAN BANKING SOFTWARE DOMAIN” under the supervision of Dr. Indika Perera.

In order to carry out my research, there is a need of real data gathering regarding my topic mentioned above.

I, kindly request to arrange an interview for me (at least with two requirement engineers and two software engineers) to collect data related to this topic from your organization as much as possible to make this research a success.

Here with, I have attached the research summary, signed approval letter and the questionnaire for your reference.

I really appreciate if you can allocate me a convenient time for a quick interview.

Thank You & Best Regards,  
Wenura Gamage.

## Appendix C: Interview Request Letter

K.G.W. Rangana,  
102/2, Nathuduwa,  
Kelaniya,  
(841493664V)

Dear Sir/Madam,

I, K.G.W. Rangana (841493664V), a student of University of Moratuwa at Department of Computer Science and Engineering, currently enrolled in Master of Business Administration 2015 batch under the index number of 159129V.

I am conducting my research on the topic of “IDENTIFYING FACTORS IN REQUIREMENT ENGINEERING PHASE AND IMPACT THEREOF ON SOFTWARE DEVELOPMENT PHASE: A STUDY ON SRI LANKAN BANKING SOFTWARE DOMAIN” under the supervision of Dr. Indika Perera.

In order to carry out my research, there is a need of real data gathering regarding my topic mentioned above. I, kindly request to provide data related to this topic from your organization as much as possible to make this research a success.

Thank You.

-----

Name: Dr. Indika Perera

(Research Supervisor)

Date:

-----

Name: K.G.W. Rangana

(Research Student)

Date:

## Appendix D: Interview Request Research Summary

**Research Title:** IDENTIFYING FACTORS IN REQUIREMENT ENGINEERING PHASE AND IMPACT THEREOF ON SOFTWARE DEVELOPMENT PHASE: A STUDY ON SRI LANKAN BANKING SOFTWARE DOMAIN

**Names of Researchers:** Dr. Indika Perera, Mr. K. G. W. Rangana

**Researchers Introduction:** Dr. Indika Perera is a senior lecturer, attached to the department of computer science and engineering at University of Moratuwa, Sri Lanka. He got his PhD from the University of St Andrews. Mr. K. G. W. Rangana is a post graduate (Master of Business Administration in Information Technology – 2015) student attached to department of computer science and engineering at University of Moratuwa. He is also working as Senior Software Engineer for private software organization in Sri Lanka.

**Research Summary:** Software Development Life Cycle (SDLC) consisted with mainly five stages. They are, Requirement engineering, Designing, Development, Testing and Maintenance. If the first stage having some issues it directly reflects through the whole process and produce a failed output. Through this research, it will be identifying some critical factors affect to the requirement engineering and the impact of them to the software development phase in the banking software domain.

**Research methodology:** Data is collected through several interviews and collected data will be used to build up research scenarios.

**Questionnaire:** Questioner is consisted in four parts. They are, Overview of the organization, General questions to the requirement engineers, Factor based questions to the requirement engineers and Impact based questions to the software engineer.

**Anonymity and confidentiality:** Questionnaire does not contain questions to identify any specific information about the interviewee or the organization. The questionnaire will remain anonymous after submission too.

**Acknowledgement:** Thank you very much for your time and help in making this study possible. Your participation in the research would be greatly appreciated.

**Contact:**

Research supervisor:

Researcher:

*Dr. Indika Perera (PhD (St Andrews), MBS (Colombo), MSc (Moratuwa), BSc Eng. (Hons) (Moratuwa), PGDBM (Colombo), MIE (SL), CEng) indika@cse.mrt.ac.lk*

*Mr. K.G.W. Rangana, No, 102/2, Nathuduwa, Kelaniya. 841493664V 0779646046 wenura.15@cse.mrt.ac.lk*

## Appendix E: Answers Summary – Requirement Engineers

Question	Open coding	Axial coding	Selective coding	Category
Q1. What languages/methods (I.e. Phone call, Interview, Skype, E-mail, Chat message, other) do you use to communicate to the client and which method is easier/difficult for you?	Both Sinhala and English used as the communication language. Mode of communication would be face to face interview, phone calls, email, conference calls, Skype voice calls, previous document reading.	Face to face interview, phone calls, Sinhala, English, Email, Skype, Conference Calls, Document Analysis	Offline communication, Online communication, Communication language	Communication
Q2. How do you gather initial requirements?	Gather initial requirement Through interviews, discussions, workshops, case studies, face to face interviews	Interviews, discussion, workshops, case studies, face to face interviews	Offline communication and Online communication	Communication
Q3. How do you clearly understand the requirements?	By conducting interviews with the stakeholders, reading the existing documentations, asking the questions. Draw flow charts based on the requirement, By UML design, By reading the collected requirement several times, preparing the documents,	Interviews, read documents, draw flow charts, UML design, preparing the documents, verify with the customer, analyzing the requirements	Direct communication, Indirect communication, Paper based communication	Communication/Documentation



	verify with the customer, draw flow of the requirement, analyzing the requirements.			
Q4. How do you verify the requirements and with whom?	Initially designing some prototypes, samples. Verifying with the project manager and the product owner, do the initial draft, prepare documents, prepare working prototypes, design UI diagrams, design mockups, contact customer and verify with them	Prototypes, samples. verifying with the project manager, product owner, initial draft, prepare documents, prepare working prototypes, UI diagrams, mockups, contact customer and verify with them	Internal communication, External communication, Online documents, Paper based documents, External documents, Internal documents	Communication/ Documentation
Q5. How do you suggest alternatives to the client?	Suggest the best option most of the time, If the required thing is not technically capable as requested, suggest another things, Provide few best suggestions among several alternatives and ask customer to select one of them	Convincing , Proving or Presenting the suggestions, the best options are, provided, ask customers to select preferred one	External communication, Offline communication and Online communication, Paper based communication	Communication/ Documentation
Q6. When do you need to talk to the client again for the same requirement?	If the requirement is not clear enough to design/implement/estimate time/confirm the requirement to customer. To get	Requirement is not clear enough to design/implement/estimate time/confirm the requirement to customer. Get missed points.	Re-communicate, Re-collect requirements Re-complete requirement , Re-analyze requirement	Communicate/ Incomplete requirements/Inadequate requirements

	the missed points. Find out the changes. Clear doubtful areas.	Find out the changes. Clear doubtful areas.	s	
Q7. What criteria do you use to prioritize requirements?	Stakeholder requirements, dependency, resource availability, technical difficulty, Requirement of the client. Task size, time to finish, resources need to finish the task, Urgent task list, technical problems, dependency among tasks, resource utilization, Customer feedback	Requirement, dependency, resources, technical difficulty, task size, time to finish, customer feedback	Requirement ranking	Requirement prioritization
Q8. With whom do you discuss when prioritizing requirements?	Initially with the project manager. Then with stakeholder, with customer, with development team	With the project manager, stakeholder, customer, development team	Internal communication, External communication	Communication
Q9. When do you need to change the priority?	If some high priority task comes in-between. If there are some technical blockers, Based on the priority level changes. Urgent tasks comes accidentally, If the client need to change it, business rules	High priority task comes in-between. Technical blockers, priority level changes, Urgent tasks, client changes, business rules changes, unavoidable situations, issues in the collected requirements	Requirement ranking, Re-order ranking, Re-analyze	Requirement prioritization/ Incomplete requirements/Inadequate requirements

	are changed due to unavoidable situation, there might have incorrect requirements.			
Q10. What documents do you prepare before the developments?	There are several types of documents prepare for the help of implementation. They are SRS, BRS, Functional Speck, Requirement prioritization Document, UML, Flow charts	SRS, BRS, Functional Speck, Requirement prioritization Document, UML, Flow charts	Prepare technical documents, prepare non-technical documents	Documentation
Q11. With whom do you prepare the requirement documents?	With the help of project manager and document writers, with technical leads or systems architects, with the help of requirement engineers, with developers	Project manager, document writers, technical leads or architect, requirement engineers, developers	Internal communication, External communication, Prepare documents	Documentation/Communication
Q12. How do you verify the prepared documents and with whom?	With the project manager and stakeholder. For the technical documents, with Architects or technical leads, sometimes with client, sometimes with developers. Verify with the team by analyzing, asking questions. Sometimes change the priority level, while verifying	With project manager, stakeholder, architects, technical leads, client, developers. Verify the documents by analyzing	Internal communication, External communication/Online documents/Offline documents	Communication/Documentation

<p>Q13. For what reasons do you have to change the completed documents?</p>	<p>If the requirement changes comes in. If the business process changes, If the new requirements comes in, when the business requirements change, If the technology changes comes in, For requirement changes, Errors in the documents</p>	<p>Requirement changes, Business process changes, New, Business requirements change, Technology changes, Errors in the documents</p>	<p>Requirement changes, Business process changes, Technology changes, errors in documents</p>	<p>Incomplete requirements/Inadequate requirements/Documentation</p>
---	--	--	---	--

## Appendix F: Answers Summary – Software Engineers

Question	Open coding	Axial coding	Selective coding	Category
Q1. How do you get clear/complete requirements before the development begins?	Reading through SRS, Flow diagrams, UML diagrams, Functional Specifications. Read through the provided requirement notes and memos, internal discussions	SRS, Flow diagrams, UML diagrams, Functional Specifications, requirement notes and memos, internal discussions	Refer documents, Offline communication	Documentation/Communication
Q2. How clear you understand the given requirements?	Time is limited to read and understand the requirement before the developments begins, however developers get time to understand the technical docs. Read the documents for initial development begins. Full picture of the requirement is not 100% clear before begins the developments. Sometimes while developing the application requirements are changed. Try to understand the	Read documents and understand it. Try reading technical and non-technical documents before and while implementations going on. But ultimately the time is not enough to clearly read and understand them.	Refer documents, Verify requirements	Communication/Documentation

	<p>given requirement clearly as much as possible and start the initial implementation based on that, Reading the functional specification is much clear the requirements. Time is given to understand the UML, but that time is not enough. sometimes, Prepare UML based on the requirements and try to go with it</p>			
<p>Q3. How do you get adequate time to read/understand those requirements/documents?</p>	<p>Normally the time is limited to read and understand the whole picture before the development starts, but to start it initially get enough time. Initially read the documents and understood, but rest of the content understand when the implementation going on. While developing, go through the content of the materials, Read the specifications</p>	<p>Time to read and understand the documents, read the document while implementation going on, read and draft the initial steps,</p>	<p>Refer documents, Understand the documents</p>	<p>Documentation</p>

	<p>first and draft the initial work, then based on that refer documents and implement. Functional specification is read before the development start because to build the architecture, design. Normally no additional time to read the documents. If the requirement is clear enough to start the coding, it is started, While developing is going on, read the documents. Not getting much extra time to understand them</p>			
<p>Q4. How do you verify the requirement/documents with the requirement engineers?</p>	<p>Ask questions, have discussions, writing test cases with the team. Have meetings with them. First with in the development team, documents are referred then find the points where further verification needed, then raise the issues</p>	<p>Questioning, Test case verifications, document referring, communicate to the customer</p>	<p>Internal communication /Refer documents/external communication</p>	<p>Communication/ Documentation</p>

	<p>to business analyst. Identify the grey areas, then discuss with the BA team, if not go for the customer. Initially both the BA and SE team gather and discuss the prepared documents. Ask doubtful areas.</p>			
<p>Q5. For what reasons, do you have to request for requirement verifications?</p>	<p>If the requirement is not clear enough to developments. Some requirements are not clear when it comes to the development stage, and then ask for some verification or clarification. To understand the requirement further and further. Documents are not that much clear about the tasks. If the requirement is not clear to start the development, if it blocks some task. To understand the missing parts, changed parts. To find out</p>	<p>Not clear to development begins. Clear the doubtful area. Clear to technical difficulty. Understand the missing parts. Prepared documents are clear. Requirement reprioritization.</p>	<p>Reevaluate documents/Miscommunications, Requirement reprioritization</p>	<p>Incomplete requirement/Inadequate requirements/Requirement prioritization/Communication/Requirement prioritization</p>



	<p>more detail on the requirements. If the given requirement is not clear to develop the system, if it has some technical difficulty. Sometimes the tasks listed for a particular sprint is not properly ordered when it comes to the implementations so that needs to be changed.</p>			
<p>Q6. How do you spend the time until you get verification?</p>	<p>Move to another task if that is possible and not depend on the hold one. If not, Do a Research part of the project. If some other tasks are assigned, worked on that. Sometimes does not need to get verified from the client but get instant verifications form the business analysis team, Sometimes need to wait till customer respond.</p>	N/A.	N/A.	N/A.
<p>Q7. How do you manage your time with requirement</p>	<p>Working on understanding other tasks,</p>	N/A.	N/A.	N/A.

changes?	working on research. Working on some other tasks, RnD. Working on something else. If no tasks assigned, read some technical stuffs. Do non project work such as technical findings.			
Q8. How do you get additional time for changes?	Normally we do not get extra additional time for small changes. If it is somewhat large we get time. Have discussion with the leads and get allocated time if it is highly necessary. If the change is big pass that to next sprint or if it needs within this sprint, pass some other task to next sprint.	Get additional time large change. Need to request from the leads and proofs are needed. Pass the change to next sprint or exchange it with a task with this sprint.	Reassert requirements.	Communication/ Documentation/Incomplete requirements/Inadequate requirements
Q9. How do you extend the deadline for requirement changes?	Ask from the project manage. Talking with leads, Provide scenarios to them and make them aware. Ask from the management with samples.	Communicate to leads; communicate to the customer indirectly (through project manager or team leads). Need to prove	Reassert requirements.	Communication/ Documentation

	Communicate to the PM and customer (not directly). By talking with the leads.	from extra time request. Need to prepare small notes on them.		
Q10. How do you explain the re-development cycles due to not getting the right requirements?	There are several times we have to work on the same task. There might have several iterations on some functions due to changes. We will have to do some rework because of some issues in the requirements. Need re-developments or changes in existing implementations if it got changed. Re-development happens due to changes to the existing functions or issues in the existing functions.	Work in the same task again and again sometimes. Sometimes not getting the correct requirement or complete requirement need to do iterations on the same function.	Redevelopment , Retesting, Reintegration	Incomplete/Inadequate requirements
Q11. How do you test your existing functions due to redevelopment?	We have test whole process if some part got changed. It depends on the change, if the change affects to the existing functions, need to test them too. Unit testing are necessary, then	Test whole process if some part got changed. If the change affects to the existing functions, need to test them too. Unit testing, integrations testing.	Redevelopment , Retesting, Reintegration	Incomplete/Inadequate requirements

	do some integrations testing.			
Q12. Any real situation of success/failure, due to requirement engineering?	Most of the projects are success. Due to requirement engineering we did not have any project of failure. Very little failure due to issues in the requirements. There was not having total project failures. But some sprints but not totally. Most of the cases were success stories. Sometimes there were situations, that project got delayed due to waiting time till the customer respond to verifications. There were some occasions where deadline extended to meet the client requirement due to some major changes, but anyhow we could deliver the product.	N/A	N/A	N/A
Q13. Any additional thing you have to express for the impact due to	Understanding the requirement is essential before the	N/A	N/A	N/A

<p>requirement engineering?</p>	<p>developments start.  Organization must allocate much time to Requirement engineering and Design.  100% agree with the requirement engineering affects the final outcome of the project.  Business analysis team has to more concern on the requirement engineering.  Requirement engineering is the highest important thing in the SDLC.</p>			
---------------------------------	---	--	--	--

## Appendix G: Interview Answer Script – Requirement Engineers

In this answer script each row in front of the each organization reflects the separate answers by each requirement engineer in the organization. Organization 2 participated for the research via e-mail and organization 4 and 5 participated for the interview via phone call. Therefore, most of the time answers were redundant of those organizations.

Question 1	What languages/methods (I.e. Phone call, Interview, Skype, E-mail, Chat message, other) do you use to communicate to the client and which method is easier/difficult for you?
Organization 1	We use Sinhala mostly as the language and face to face and phones calls are used as the communication mode.
	We communicate in Sinhala and I mostly use face to face interviews.
	Normally we use Sinhala and use face to face interviews.
Organization 2	Sinhala/English and face to face
	Sinhala/English and face to face
	Sinhala/English and face to face
Organization 3	Language is based on the client and it will be face to face most of the time.
	We use both Sinhala and English and face to face interviews
Organization 4	Mostly in English through face to face interview and documentation
	It is most of the time in English but sometimes in Sinhala and face to face
	Face to face interview in Sinhala and English
Organization 5	Meetings with clients in both Sinhala and English
	All clients are Sri Lankans and we use Sinhala
	Face to face meetings with clients in Sinhala

Organization 6	Sinhala is the convenient language for us and normally we use face to face interviews but sometimes we use conference calls, Skype voice calls and refer existing documents also.
	We use Sinhala and English, and then we initially use face to face interviews and then phone calls, emails.
Organization 7	Sinhala and face to face interviews.
Organization 8	Sinhala and gather requirements from interviews, phone calls and email

Question 2	How do you gather initial requirements?
Organization 1	Through face to face interviews.
	Face to face interviews or discussion rounds
	Face to face interviews
Organization 2	Face interviews
	Face interviews
	Face interviews
Organization 3	From face to face interviews and with few clients we had used workshops and case studies.
	Face to face meetings with customers
Organization 4	Face to face interview.
	Face to face interview
	Face to face interview
Organization 5	Through interviews.
	It is face to face
	Most of the time face to face but sometimes we had case studies also
Organization 6	Face to face interview.
	Face to face interview.
Organization 7	Face to face interview.
Organization 8	Face to face interview.

Question 3	How do you clearly understand the requirements?
Organization 1	Through the interviews and meeting points I understand the requirements clearly
	Normally we understand the requirements by participating to interviews and asking questions
	By intervening the stakeholders, reading the existing documentations, asking the questions and sometimes draw flow charts based on the requirements.
Organization 2	Draw UML
	Draw diagrams
	Draw diagrams
Organization 3	By intervening the stakeholders at the interviews then analyze the results
	We have to read the requirements several times to understand.
Organization 4	I need to analyze it further
	I need to go through the requirements
	I need to refer requirement and analyze them.
Organization 5	I understand it like a story.
	I understand it by reading the collected requirements several times
	I need to refer requirement again and again.
Organization 6	I normally documenting them and verify with the customer
	I'm practicing it like draw of flow diagrams.
Organization 7	What I normally use is try to draw the flow of the requirements and try to verify it with the customer.
Organization 8	Normally I need to go through the requirements and analyze them.

Question 4	How do you verify the requirements and with whom?
------------	---



Organization 1	We use prototypes to verify with customer
	I am using prototypes to verify with customer
	Normally we use prototypes and verify with customer
Organization 2	Use prototypes with PM
	Discuss with project manager
	Verify with project manager
Organization 3	As a practice I use prototypes with project manager or sometimes with product owner.
	We do some initial drafts and verify with the customer
Organization 4	Preparing documents and verify with project manager
	Prepare diagrams like prototypes and verify with project manager
	Prepare diagrams like prototypes and verify with project manager
Organization 5	Verify through designs, working prototypes.
	By designing UI and confirm them with customer.
	Sometimes through mockups
Organization 6	We develop working prototypes and share with customer.
	We use working prototypes and mockups.
Organization 7	I usually use diagrams with project manager
Organization 8	Design prototypes first and confirm with the client

Question 5	How do you suggest alternatives to the client?
Organization 1	We suggest the best option most of the time.
	If there are some other options we suggest them
	We suggest them if the requirement is not technically capable as requested.
Organization 2	Sometimes
	Sometimes
	Sometimes
Organization 3	As a practice we suggest some other things
	We suggest another things

Organization 4	Ask customer to go with the best option
	Suggest alternatives
	Suggest alternatives
Organization 5	Explain the best option to do the same task.
	Suggestions coming from us
	Provide list of alternatives
Organization 6	We provide best options available
	We ask from customer to select from list of alternatives
Organization 7	We provide suggestions or alternatives and ask customer to select one of them
Organization 8	Our team show the alternative paths to the project manager

Question 6	When do you need to talk to the client again for the same requirement?
Organization 1	If the requirement is not clear enough to design.
	When requirement is not clear to implement or develop
	Is the requirement is not enough for time estimation
Organization 2	If it is not clear
	If it is not clear
	If it is not clear
Organization 3	Sometimes the given requirement is not enough when we start the implementation
	We request when it is not clear
Organization 4	If that is not clear
	When it is not clear to developers
	When it is not clear
Organization 5	To clear some areas
	To clear the requirements
	When do we need some verifications
Organization 6	We talk to the client to get missed points

	To find out the changes we have to contact the customer
Organization 7	Our team meet customer to clear up the unclear areas.
Organization 8	We discuss with client to clear doubtful areas

Question 7	What criteria do you use to prioritize requirements?
Organization 1	Normally we consider stakeholder requirements, dependency, and resource availability.
	We think of Requirements and availability of the team
	We consider technical requirements and business requirements
Organization 2	Customer requirements and technical difficulty
	Customer requirements and technical things
	Customer requirements
Organization 3	We consider most of the time requirements of the client
	We are considering on like customer requirements, size of the task
Organization 4	Customer requirements
	Customer requirements
	Customer requirements
Organization 5	Customer priority list
	Customer priority list
	Customer priority list
Organization 6	We are looking at mostly customer feedback, resource utilizations, task dependency
	It's like customer requirements, technical problems; resources need to finish the task.
Organization 7	We are thinking of client's requirements, time to finish the task, resource availability
Organization 8	Most of the time we focus on customer urgent list

Question 8	With whom do you discuss when prioritizing requirements?
------------	--

Organization 1	With project manager
	With project manager
	With project manager
Organization 2	With project manager
	With project manager
	With project manager
Organization 3	Initially with PM and then with stakeholder.
	Initially with PM and then based on the requirement with customer
Organization 4	With the customer
	With the customer
	With the customer
Organization 5	With project manager and customer
	With project manager and customer
	With project manager and customer
Organization 6	Initially with the development team
	With the development team
Organization 7	With developers
Organization 8	With the customer

Question 9	When do you need to change the priority?
Organization 1	If some high priority task comes in-between. If there are some technical blockers we have to change the priority list.
	Sometimes we need to change the priority list if urgent task comes or for any other blocks
	We have to do like that if the customer requests to do so.
Organization 2	If some other task needs high priority
	Based on the priority of the other tasks
	Based on the customer requirements
Organization 3	Sometimes there can be cases like, change of the priority level.

	Sometimes some urgent task comes in so we need to focus on that.
Organization 4	Client changes the order
	That will happen if the customer need any changes
	Normally it is based on the customer requirements
Organization 5	Client change that accordingly
	If the customer need to change it
	Sometimes customer may change it
Organization 6	Sometimes the business requirement may change so that the order needs some changes
	We have to change the order once because some technical difficulties and we might have gathered incorrect requirements
Organization 7	Sometimes there may be cases like unavoidable situations.
Organization 8	Most of the time customer needs to change and sometimes we have to change the order.

Question 10	What documents do you prepare before the developments?
Organization 1	Like SRS, BRS
	SRS
	SRS
Organization 2	SRS
	SRS
	SRS
Organization 3	SRS, BRS, functional specifications
	SRS,BRS
Organization 4	SRS, UML diagrams
	SRS, UML diagrams
	SRS, UML diagrams
Organization 5	SRS including UML diagrams, functional documents
	SRS including UML diagrams, functional documents

	SRS including UML diagrams, functional documents
Organization 6	SRS, flow charts
	SRS, flow charts
Organization 7	SRS, UML, requirement prioritization document
Organization 8	SRS, flow diagrams

Question 11	With whom do you prepare the requirement documents?
Organization 1	We get the help of project manager.
	Normally we discuss with our project manager
	We are having internal discussion with our PM.
Organization 2	With project manager
	With project manager
	With project manager
Organization 3	We are having meetings with project managers and document writers within the organization
	Normally as a team we discuss, team means like, technical leads, architects
Organization 4	By our own team
	Within the team
	Within the team
Organization 5	With PM and Technical teams
	With project manager
	With project manager
Organization 6	With the team including QA, developers
	With people who participated to requirement gathering
Organization 7	Normally with developers and QA
Organization 8	With project managers

Question 12	How do you verify the prepared documents and with whom?
Organization 1	Normally we are having discussion with project manager

	In general with project manager but sometimes we involve our stakeholders too.
	We involve our project managers and sometimes technical teams like TL, project architects, developers to discuss on technical documents.
Organization 2	With PM and stakeholder
	With PM and stakeholder
	With PM and stakeholder
Organization 3	Normally within the organization, having discussions like questing, analyzing documents within the team
	Normally we have discussion in our teams sometimes while discuss we many change the priority level too.
Organization 4	With the PM first and then with the client
	With the PM first and then with the client
	With the PM first and then with the client
Organization 5	With project manager and customer
	Discuss with project manager
	Have meetings with project manager
Organization 6	We conduct meetings with project manager and customer
	We arrange meetings within the team but sometimes with our client too
Organization 7	We have discussions with developers and QA
Organization 8	We are having meeting with the development team and project manager

Question 13	For what reasons do you have to change the completed documents?
Organization 1	Sometimes requirement changes comes in and we need to change the prepared docs
	When changes are coming we need to change them

	Sometimes requirement is going to be change, business process going to be change so that we change the existing documents
Organization 2	If the new requirements comes in and existing ones are going to be changed
	When requirement are changed
	When requirement are changed
Organization 3	When the business requirements change, new requirements comes in and technology changes are coming to the project.
	Mostly if new requirements are coming and then sometime we experienced we had issues in documentations.
Organization 4	For the new requirements
	For the new requirements
	For the new requirements
Organization 5	Requirement changes
	Requirement changes
	Requirement changes
Organization 6	We rarely get such types of changes but it can be like customer make changes.
	Customer make changes to the requirements so documents should reflect this
Organization 7	Most of the time customer makes changes and we also have issues in documentations.
Organization 8	Sometimes customer make changes to the existing requirements and new requirements are coming in



## Appendix H: Interview Answer Script – Software Engineers

In this answer script each row in front of the each organization reflects the separate answers by each software engineer in the organization. Organization 2 participated for the research via e-mail and organization 4 and 5 participated for the interview via phone call. Therefore, most of the time answers were redundant of those organizations.

Question 1	How do you get clear/complete requirements before the development begins?
Organization 1	Normally we refer SRS.
	Other than referring SRS, we use flow diagrams, UML diagrams.
	We refer SRS.
	We read SRS
Organization 2	Refer SRS
	Reading through SRS and other documents
	We use SRS
	We use SRS
	By referring SRS
Organization 3	Requirement comes in SRS and sometimes we refer other technical documents too
	We refer SRS and technical docs
Organization 4	Through SRS and Functional Specification
	We get the information from SRS
	Mainly from SRS
	We read through SRS
Organization 5	Read through provided requirements
	Read documents like SRS
	Most of the time refer SRS before development begins
	We get very little time to read SRS

	Read SRS but time is not enough
Organization 6	Other than SRS, we don't refer other documents.
	SRS but the allocated time is not much enough
Organization 7	We get complete thing in SRS, UML diagrams, functional specs,
	Mainly read SRS and memos, we keep notes while internal discussions going on
	We refer SRS but definitely the time is not enough
Organization 8	We are given SRS, diagrams, UML to refer
	We can read SRS and UML

Question 2	How clear you understand the given requirements?
Organization 1	Time is limited to read and understand the requirements before the developments begins, but however we get time to understand the technical docs
	Time is very limited but anyhow we have to manage that
	Time is not enough to understand most of the time
	We start developing the task sometime before understanding the requirements
Organization 2	We understand the requirements
	We understand the requirements
	We understand the requirements
	We understand the requirements
	We understand the requirements
Organization 3	Full picture of the requirements is not clear before begins the developments.
	Full understanding is not there for to start the developments sometimes
Organization 4	Read the documents for initial development begins and we understand them

	Read for the initial developments and start it
	Read and understand the requirements
	We understand the requirements
Organization 5	Initial requirement is not totally understandable sometimes
	For some projects we understand the initial requirements clearly
	While developing the application requirements is going to be changed
	Understandable but not fully
	Read documents to clear the requirements
Organization 6	Try to understand the given requirements clearly as much as possible and start the initial implementation based on that.
	By reading the functional specification is somewhat clear the requirements
Organization 7	Time is given to understand the UML, but that time is not enough sometimes.
	Since the time is not enough requirement is not totally understandable
	While developing it is becoming clear to us more
Organization 8	I normally prepare UML based on the requirements and try to go with it
	I try to clear the requirements as much as possible and I mean the requirement is not clear 100%.

Question 3	How do you get adequate time to read/understand those requirements/documents?
Organization 1	Normally the time is limited to read and understand the whole picture before the developments start, but to start it initially we get enough time
	Time is not enough actually
	Time is very limited with the deadline

	Time is limited but can manageable
Organization 2	While developing we go through the content of the materials
	We go through the documents while developing
	We go through the documents while developing
	We go through the documents while developing
	We go through the documents while developing
Organization 3	Initially we read the documents and understood, but rest of the content understand when the implementation is going on
	We read the specifications first and draft the initial work, then based on that we refer documents and implement
Organization 4	Not getting adequate time
	Not getting adequate time
	Not getting adequate time
	Not getting adequate time
Organization 5	Get reasonable time but it is better if we can get more
	Get reasonable time but it is better if we can get more
	Get reasonable time but it is better if we can get more
	Get reasonable time but it is better if we can get more
	Get reasonable time but it is better if we can get more
Organization 6	Functional spec read before the development start because to build the architecture and design
	Normally we don't get additional time to read the documents but if the requirement is clear enough to start the coding we start it
Organization 7	While developing is going on we read the documents
	We are getting adequate time to read and plan the development
	I normally refer the specs before the development stage begins
Organization 8	Not getting much extra time to understand them
	We read and develop things at the same time

Question 4	How do you verify the requirements/documents with the
------------	---

	requirement engineers?
Organization 1	We ask questions; have discussions, writing test cases with them.
	We have discussion and try to understand them
	We have meetings with them
	We conduct meetings with them
Organization 2	We are having meetings with the development team first
	We will to have discuss within the own team
	We need to have meetings within the team
	Need to discuss the reqiirements
	We need discussions
Organization 3	As a team we identify the grey areas, then discuss with the BA team, if not go for the customer
	First we need to refer documents, then need to find points where further verifications needed then need to raise those issues from business analyst
Organization 4	Have meetings
	Have meetings
	Have meetings
	Have meetings
Organization 5	Ask doubtful areas.
	Need to ask questions for verifications
	Questing the business analysts
	Need to have meetings with the BA
	Have meetings with the BA
Organization 6	Initially both the BA and development team gather and discus the prepared documents.
	Need to refer documents and then we need to verify things
Organization 7	Need to ask some doubtful areas from the leads
	We should have discussions within the development and BA teams
	We need to verify things from the BA

Organization 8	We as a team should have internal meetings
	Developers ask questions in the meetings

Question 5	For what reasons, do you have to request for requirement verifications?
Organization 1	If the requirement is not clear enough to developments
	Some requirements are not clear when it comes to the development stage and then ask for some clarification or verification from the requirement gathering team
	When can't we understand the requirement clearly we request for verifications
	When do we need some extra information from the leads
Organization 2	When the requirement is not clear
	When do we need further clarifications
	To understand the requirement
	When it is not clear
	When it is not clear
Organization 3	We do so to understand the requirement further and further and the documents are not that much clear about the tasks
	Since the requirements are not clear for us we ask for verifications sometimes
Organization 4	If the requirement is not clear to start the development and if it blocks something.
	When we need more information
	Sometimes we need further details of the tasks
	If the requirements are not clear to me
Organization 5	If no one can understand the requirements
	If we need to know some parts of the documents or whatever in the requirements we need verifications
	To understand the missed things

	To get clear things
	We need to clear things
Organization 6	If the given requirement is not clear to develop the system and if it has some technical difficulty definitely we ask for verifications
	I personally request it to understand the missing parts and changed parts and sometimes to find out more details on the requirements.
Organization 7	To find out more detail on the requirements our team ask for the verifications sometimes
	Sometimes the tasks listed for a particular sprint is not properly ordered when it comes to the implementations so that needs to be changed
	Normally we may need some clarifications to the development
Organization 8	If we can't understand the given task we need to talk back to them
	We might need some clarifications of some parts before the development stage.

Question 6	How do you spend the time until you get verification?
Organization 1	Move to another task if possible and I'm not depending on the hold one.
	I do a research part of the project.
	If some other tasks are assigned to me I'm working on that.
	I move to some research parts of the project
Organization 2	None
	None
	None
	None
	None
Organization 3	Sometimes we get instant verifications form the BA team so no

	need to wait
	Sometimes we need to wait till customer respond
Organization 4	We don't have such situations
	We don't have such situations
	We don't have such situations
	We don't have such situations
Organization 5	No such experience
	No such experience
	No such experience
	No such experience
	No such experience
Organization 6	We move to some other task.
	I get some part form other developer and do it
Organization 7	We don't have to wait much time
	No need to wait much time since most of the time our leads helps us
	Sometimes we have situations like that. so that time we move to some other task
Organization 8	To my experience that is very rare but if that happens we definitely move another task.
	May be we move to something else

Question 7	How do you manage your time with requirement changes?
Organization 1	We need to understand the new change and apply them
	If we have already done the coding we need to change that
	We have to rewrite the coding sometimes
	Normally we don't get much time for changes
Organization 2	Anyway we have to manage the changes but it is much difficult
	We have to work on that as soon as possible
	We need to manage that anyhow



	We have to manage the change and work on that
	We have to do both agreed list of tasks and the changes
Organization 3	Sometimes we get more time for changes but sometimes not
	We get extra time for that change
Organization 4	We ask for extra time and sometimes we get sometimes we don't
	Estimated time is not enough then so we need additional time
	We have to re-estimate the time then
	We need to re-evaluate the effort
Organization 5	Sometimes hold the change till we complete existing things
	Hold the change for few days or weeks
	We have to consider the change in later stage
	We have to work extra sometimes
	It's going to be extra work
Organization 6	Need to have extra work then
	We have to consider that change
Organization 7	We normally ask for extra time and if that is not given we have to do extra work
	No option we need to attend on it
	We have to put extra effort to complete it
Organization 8	We discuss within the team and split them among the team
	First we need to have time estimation for it and then go for the developments

Question 8	How do you get additional time for changes?
Organization 1	Normally we do not get extra additional time for small changes.
	If the change is somewhat large we get time.
	We do not get extra time for very little changes but for large changes we get
	We get time for large changes only
Organization 2	Normally we don't get additional time, but sometimes we get

	Sometimes we get additional time
	We get additional time
	We request for time
	We have to ask for extra time
Organization 3	Our process is like move that task to next sprint if that is much bigger
	If the change is big pass that to next sprint or if it needs within this sprint pass some other task to next sprint
Organization 4	We should have to discuss within the team
	We need to have discussion before come to a decision
	Need discussions
	Need discussions
Organization 5	Leads allocate extra time
	From leads we ask for time
	It's up to the leads
	Sometime our team lead give us time
	It is based on the size of the requirement whether we get or not getting time
Organization 6	We normally ask for extra time and most of the time we get that
	We request additional time for changes
Organization 7	We need to have discussion with our team leads first
	We discuss with our leads for additional time
	If the task is big we have to have extra time so we request for extra time
Organization 8	We are having discussion with the leads and get allocated time if it is highly necessary
	Need to have discussion with the team leads for asking extra time

Question 9	How do you extend the deadline for requirement changes?
Organization 1	Normally we do not get that but ask from the project manage.

	We directly communicate to the project manager
	We need to communicate to the project manager
	We talk to our leads first
Organization 2	We need to have discussions with PM
	We need to have discussions with PM
	We need to have discussions with PM
	We need to have discussions with PM
	We need to have discussions with PM
Organization 3	Ask from the management so we need to show them samples that we need additional time to complete.
	We need to talk to leads definitely
Organization 4	Communicate to the leads and sometimes to project manager
	We need to get approval from the PM
	Normally we have to get it form the PM
	We should communicate this to PM
Organization 5	By talking with the leads.
	Talk with leads
	Talking to team lead
	Talking to leads
	Talking to leads
Organization 6	As I said earlier we request for extra time
	If we need some additional time we request that.
Organization 7	We communicate to the leads first
	We discuss with our leads for this
	We request for extra time
Organization 8	Talking with leads, Provide scenarios to them and make them aware
	As I said, we are talking to leads for extra time

Question 10	How do you explain the re-development cycles due to not getting
-------------	---

	the right requirements?
Organization 1	There are several times we have to work on the same task.
	Sometimes again and again we need to do the same task.
	It happens like we have to work on the same task again and again
	It does not happen frequently but sometimes so we need to work on the same task.
Organization 2	We had such situations but very few
	No we don't
	None
	We rarely get such experience
	Normally we don't have that kind of situations
Organization 3	We will have to do some rework because of some issues in the requirements.
	Normally we have to focus on same task few times
Organization 4	We have to redo the same sometimes
	We need to work on same development several times
	We will have to work on same task sometimes
	There may be some redo tasks because of that issue
Organization 5	-
	-
	-
	-
	-
Organization 6	There might have several iterations on some functions due to changes
	Sometimes we don't get correct requirements so we need to have work on same task several times
Organization 7	We need re-developments or changes in existing implementations if it need some changes
	We have to redo the same task if that is going to be changed
	Sometimes in our projects we do that kind of things

Organization 8	To my experience re-development happens due to changes to the existing functions
	As I feel because issues in the existing functions we had to face for rework the same

Question 11	How do you test your existing functions due to redevelopment?
Organization 1	We have to test whole process if some part got changed.
	We need to test the full functionality
	Sometimes we may need to test the full life cycle
	Most of the time we will have to test whole
Organization 2	We need to test the whole function
	We need to test the whole function
	We need to test the whole function
	We need to test the whole function
	We need to test the whole function
Organization 3	We need to test whole process if some part got changed.
	Better we test the full function
Organization 4	We do some integration testing
	Normally we have to do full process testing
	Anyway we test the existing function and do the integration testing too
	We need to test full process
Organization 5	We do full testing and then integration testing.
	We do full function testing after that do the integration testing.
	I usually do integration testing.
	We have to complete integration testing.
	Before do the integration testing we do the testing for changed function
Organization 6	We do full testing on changed functions and test the other one also

	We normally do unit testing and then do some integrations testing.
Organization 7	It depends on the change, if the change affects to the existing functions, need to test them too
	We do full testing
	Sometimes we do full testing
Organization 8	It is better if we can test the full function
	What we normally do is unit testing again

Question 12	Any real situation of success/failure, due to requirement engineering?
Organization 1	Most of the projects are success due to requirement engineering and we did not have any project of failure yet
	We don't have failure stories yet
	I don't have such experience like project failure in the past history
	I have no project failure experience
Organization 2	No failure stories in the past
	No failure stories in the past
	No failure stories in the past
	No failure stories in the past
	No failure stories in the past
Organization 3	As I can remember there were no total project failures but some sprints I guess
	Most of the cases were success stories in the past except few
Organization 4	There was no failures
	No total project failure was in the past
	I can't remember failure situations
	We did not have total project failures
Organization 5	We have very few failure stories in the past.

	I think we didn't have project failures
	We don't have full project failure but some tasks I think
	Normally we don't experience such situations
	In our organizations we don't have any
Organization 6	Very little failure due to issues in the requirements
	We had in the past but not fully may be few tasks like that.
Organization 7	Our organization did not have those project failures but most of the projects were success
	In my team I can't remember any of the total project fails but may be few tasks
	There were some occasions where deadline extended unusually to meet the client requirement due to some major changes, but anyhow we could deliver the project
Organization 8	Sometimes there were situations like the project got delayed due to waiting time till the customer respond to verifications.
	I can't remember such a situation like project failure

Question 13	Any additional thing you have to express for the impact due to requirement engineering?
Organization 1	I think that the understanding the requirement is essential before the development start.
	I believe that the requirement understanding is much needed
	Requirement engineering is the initial phase and it is much important to developers
	We as a team think that the requirement engineering is really help for our development stage
Organization 2	-
	-
	-
	-

	-
Organization 3	I recommend that the BA team has to more concern on the RE.
	As I feel that it is very important to software development
Organization 4	That is important
	I think that is an important to us
	We have to spend more time on that phase
	I see the advantage of that phase
Organization 5	I have experienced the value of the requirement engineering
	Our organization tries to improve the quality of the requirement engineering
	We need to practice it more and more
	I guess it should be there in all the time
	Some organizations doesn't care about this phase but I know that is not correct and we have to consider this.
Organization 6	Organization must allocate much time to requirement engineering and design.
	I think we need to more concern and allocate more time on requirement engineering phase.
Organization 7	100% agree with requirement engineering affects the final outcome of the project.
	We definitely see the advantage of the requirement engineering phase
	That is a very important phase actually
Organization 8	Requirement engineering is the highest important thing in the SDLC.
	I guess that it is more important to success of the project