Adama Science and Technology University

School of Graduate Studies

Department of Computing

Software Requirement Registration and Traceability Model in the Context of Organizational Development

Endale Aragu

August 2015

Adama Science and Technology University

School of Graduate Studies

Department of Computing

A Thesis Submitted to the Department of Computing of Adama Science and Technology University in Partial Fulfillment of the Requirements for Degree of Master of Science in Software Engineering

By

Endale Aragu

August 2015

Adama, Ethiopia

Adama Science and Technology University

School of Graduate Studies

Department of Computing

Software Requirement Registration and Traceability Model in the Context of Organizational Development

By

Endale Aragu

Names and Signature of Members of the Examining Board

Desalgn Abebaw

Chair Person, Examining Board **Mesfin Kifle (PhD)**	Signature
Advisor **Tibebe Beshah (PhD)**	Signature
External Examiner **Prof.Dr.Ing Achim Ibenthal**	Signature
Internal Examiner	Signature

DECLARATION

This thesis is my original work and has not been submitted for a degree in any university.

Endale Aragu

August, 2015

ACKNOWLEDGMENTS

Most of all, I would like to thank God, for all his blessings and mercy and also helping me pass all the challenges and those hard times that I couldn‘t pass them by my own.

I owe my deepest gratitude to my advisor Dr.Mesfin KIfle for accepting this thesis work and his time, patience and also helping comments all the way through this study. He really was an inspiration for me. His eagerness and encouragement has always inspired me to towards the completion of the work.

I owe my deepest gratitude to Mr.Amanuel Zewege for his time, patience and also for helping comments all the way through this research . He really was an inspiration for me. His eagerness and encouragement has always inspired me to towards the completion of the work.

I would also like to express my appreciation to my family and friends who have helped me in so many ways.

I showed my deepest gratitude to my friend Mikias Belay who manage and assisted the interview and questioners with different programmers, customer representative and project manager in different organization

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS AND ABBREVIATIONS

ASTU .....Adama Science and Technology University

CBS .....Computer Based System

OT .....Organizational Development

RE .....Requirement Engineering

SDLC .....Software Development Life Cycle

UML.....Unified Modeling language

JAD .....Joint Application Development

RQ.....Research Question

IT.....Information Technology

RAD.....Rapid Application Development

OOA .....Object Oriented Analysis

SA .....Structured Analysis

SRS.....System Requirement Document

MPA.....Main Process Area

SPA .....Sub Process Area

ABSTRACT

Requirement engineering is the first and crucial phase in the development of software. It involves set of activities like system feasibility study, elicitation analysis, specification and management of the requirements.

Poor and uncontrolled requirement engineering processes yield low quality, highly expensive software products. Obviously, organizations are highly dissatisfied with such systems. One of the most critical requirement engineering processes that grossly contribute to this inaccuracy is lack of proper implementation and measuring requirement engineering process, project management.

Most software companies do not have formal registration and management of requirement hence it is difficult to trace and update the requirement whenever change requests come. Every requirement change by itself have an impact in the project therefore the impact of the change must be analyzed before accept and implement it because every requirement change required additional budget and time from initially stated.

In order to overcome such problem, we design a new model that show how to integrate requirement engineering practice with organizational context, project management and practice gap in the requirement engineering.

The new model may be useful for organization in order to improve the requirement engineering practice and it may be used as an input for the researcher for future requirement engineering process improvement. It is designed for requirement engineering process evolution in the organization and the implementation part of this research is important for registration and tracing of the requirement in the software project and the key result is design model that used to evaluate requirement engineering process in the software firm

Keyword: requirement engineering, organizational context, requirement change, requirement traceability

CHAPTER 1

INTRODUCTION

1.1. BACKGROUND

In the current dynamic and changing global situation, organizations have been striving to survive by transforming their service provision and product quality. The introduction of any computer based system (CBS) has the potential to transform any organization into which it is introduced. In general, the introduction of CBSs has become a driver for innovative work practices and new models of management and organization (Isabel Ramos, 2004).

Software development process plays an important role for the success of software in the organizational development

Software development failure occurs, when software could not fit for the solution or the business process and over run of the cost and time from that initially stated. The requirement Engineering (RE) process plays an important role in the software development process.

The objective of RE process should be to develop a set of necessary, verifiable and attainable requirements, which are acceptable to all the relevant stakeholders (Kotonya and Sommerville, 1998). When requirements are poorly defined and RE processes are ad hoc, the end result is nearly always an unsatisfactory product or a cancelled project.

In the requirement engineering process, consider all affected element in the organizational change. It may be organizational business process change, structural or standard change and government policy or standard change

Many software projects have failed because they contained a poor set of requirements (El Emam and Madhavji, 1995). No software process can keep delivery times, costs and product quality under control if the requirements are poorly defined (Sommerville et al ,1998). In order to produce software, which closely matches the needs of an organization, an application domain and the stakeholders, great attention must be given to the RE process (Niazi ,2000).

1.2. MOTIVATION

In light of the rapid increase in demand, fueled by the large scale national projects in the government and public sectors as well as the huge demands resulting from the IT-oriented reform initiatives in the service industry (such as telecom, airlines, banks, insurance companies, etc.), and the increase in presence of foreign business firms in response to the investment opportunities extended, currently the demands and pressures for better and improved software development and support services at the local level are increasing. Furthermore, the national ICT capacity building initiatives envision transforming the country into a preferred outsourcing destination in the software sector. On the other hand, despite such increase in investment and encouraging developments, there is very limited capacity to meet the demands and realize the needs (Biru, 2008). Most of the existing business processes in the organizations are reported to be manual , By enabling CBS, make the organization more productive and improve quality of services in order to satisfy the customers and produce efficient and high quality products . Organizational change or development through CBS can be effectiveness only if the software development process as effective furthermore, one of the major challenges on the software projects are, most projects are not completed on-time and on-budget, with all features and functions as initially specified.

“More than half the errors in a project originate with the requirements and analysis activities done prior to product design. Based on preliminary survey, most projects fail as a result of incomplete requirements, poorly written requirements, or misinterpreted requirements.” Of project failures, 24% were attributed to incomplete requirements and specifications or changing requirements. “It seems clear that requirements deserve their place as a leading root cause of software problems, and our continuing personal experiences support that conclusion.” (Leffingwell, 2003)

In this research inspire what are the major challenges especially on requirement engineering in the software development process by considering organizational context as well as software development firm’s aspects.

1.3. STATEMENT OF PROBLEM

Organizational development needs to automate or implement CBS in the organization in order to improve the productivity and services for the customers. this idea come from higher officials or management and left on them without discussion and participation with others staff members in the organization but any software development project needs organizational staff members support and participation because every requirements of the system are collected from staff members in the organization. Without completed and correct requirement, success of the project is unrealistic furthermore in adequate planning during the tendering and organizational changes are other problems in the software development process in the organization. Most software does not fit for solution and not completed based on budget and on time this will create another problems for the organization

"Most of the companies were observed not to use any 'standard' method in the actual conduct of the software development projects. A commonly employed approach involved the use of cyclical requirements gathering and then programming practices, with a strong tendency to rely on in-house-developed practices rather than industry standard guidelines. Unsatisfied users, overwhelming costs, frustrated developers, etc. are still common characterizations of software development projects in Ethiopia." (Biru, 2008) There is no organizational guideline or standard for documentation of requirement and coding this may be create a problems during system integration and requirement change.

RE is very important and fundamental activity in the software development , based on preliminary assessment , most software companies in Ethiopia highly focus on coding, even most software project managers and team leaders does not consider as a major task and source of errors in the project. if so they could not assign enough budget ,very responsible, skilled , experienced person for this activities just assigned programmers for doing RE activities, this will create several problems during development and submission of the project.

"Given the frequency of requirements errors and the multiplicative effect of the "cost to fix" factor, it's easy to predict that requirements errors will contribute the majority—often 70% or more of the rework costs. Since rework typically consumes 30% to 50% of a typical project

budget (Boehm and Papaccio, 1988), It follows that requirements errors could consume 25% to 40% of the total project budget!" (Leffingwell, 2003) Whenever the requirement changes, it affect other software development phases but currently there is no means that shows and measures the consequence of each requirement change in the project

1.4. RESEARCH QUESTIONS

In order to identify major requirement engineering problems in the software development process, I Distributed questionnaire and discussion held with people involved in software projects in an attempt to answer the following main research questions.

RQ1: What does major requirement engineering practice and challenges, especially software development and organizations context look like?

RQ2: what are the metrics that measure practice of requirement engineering in general in terms of their applicability to the local organizational context in particular ?

The answers of RQ1 shows the RE practice in the industry and what are gaps ,challenges and consequences of RE in the project , the answer to RQ2 ,used to evaluation how the industries are measures requirement engineering practice in order achieve requirement engineering Goals.

1.5. OBJECTIVE OF THE STUDY

The general and specific objectives of the study are the following.

1.5.1. GENERAL OBJECTIVE

The general objective of the thesis is exploring and designing a requirements registration and traceability model in the context of organizational development to support organizational development.

1.5.2. SPECIFIC OBJECTIVES

The following specific objectives are identified.

- Understand the current practices of RE and related software development activities by reviewing previously conducted researches and other literatures (such as books, journal articles, conference papers and the Internet) both local and global,
- To collect information about organizational development for this research
- To identify the challenges of organizational development and software development process
- To identify and analyze requirement volatility in the software development project to design a requirements registration and traceability model
- Design a requirements registration and traceability model
- Evaluate the proposed model and develop a prototype for requirement registration and traceability in the project

1.6. SIGNIFICANCE AND APPLICATION OF THE STUDY

This research is an applied research categorized under software engineering ,it has a major significant for software development companies and any organization who need to automate the existing manual system, it also applied in the software development process how the organizational context considered in the requirement engineering process and other software development phases in order to a software fit for a solution , it address the major important activity on requirement engineering , assessing the requirement activities and identify major the organizational context issues and how consider those issues in the software development .it also how requirements are registered and properly put on the repository for the purpose of tractability.

One of the goals of software development companies getting profit with customer satisfaction. The objective of the software in the organization make them competitive and more productive, this can be achieve through proper implementation and awareness of requirement engineering activities by considering organizational context in the software development.

1.7. SCOPE AND LIMITATIONS OF THE STUDY

In this research identify major organizational context, those are highly related to software requirement engineering process in the implementation of organizational development though CBS, how CBS is fit for solution in the organization, depict requirement engineering practice, problems, cause and solution in the software development process in the organization, measure the requirement engineering practice, develop a prototype that register and trace the requirements in the project. Requirement tractability is the process of finding requirement in the project for the purpose of updating, deleting and also for further analyzing

1.8. METHODOLOGY

In order to meet the objectives of this study, the following techniques are adopted in this research

1.8.1. LITERATURE REVIEW

This thesis touches requirement engineering process and practice, software development process, software development organization practice and capacity, organizational context and development, software development process and organization context, requirement engineering practice and software development Therefore, literatures related to the ideas mentioned above and any other relevant literatures that help to enhance the research will be reviewed and if necessary will be adopted.

1.8.2. DATA COLLECTION

Due to the lack of readily available empirical data on the software industry population in the country, selection of participants was guided by undergraduate student internship office of ASTU this office have a link with different software development company for student internship program purpose and personal selection of software company and customer organization . ,A total of two client organization ,13 software companies,4 ICT software development office of the university and 91 professional are participated , Applicable questionnaires were sent to all (hand delivered as all of the respondents were in Adama and Addis Ababa)from to expedite the process, arrangements were made to physically collect the completed questionnaires A sort of purposive sampling technique was used in selecting participants from the lists. That is, attempts were made to ensure the inclusion of most of the IT Departments within large organizations that practiced in-house development of software's. All private firms that were actively involved in the development and customization of small to large software systems and projects were included. Those software companies whose major business was not software development or customization were excluded. In the selection of individual participants within these institutions, based position working on the project and year of experience, Here as well, care was taken not to exclude software engineers or project managers who had valuable experience on large projects

The question was designed by adapting from "Stuart Anderson and Massimo Felici" and "Tesfaye Biru 2008"and adding additional question from books and observing existing software development practice in the software firms

Based on collected data and interview, analyze the data and design the model which will be improve the requirement engineering process in software development firm

The following data gathering methods can be used to enrich the thesis,

Interview and questioner:

- Questioner: using questioner it is possible to address large number participant in different organization and it is difficult to interview all participants due to time constraints and they are found geographically dispersed.
- Based preliminary investigation assessment the questioners have prepared in order to address concepts of software development process and related activities based on organizational context.
- The questioners have categorized into three parts : general questions that assess the company and employee profile , the software process activities and method or approach , the second part is assess the software engineering process like requirement elicitation, analysis and specification and management of the requirement by listing recommended and common practice of the process. The third part is assessing organization information, participation and support of the project. all the information that is found the questioner . all the above information is the major factors that affect the project success.

  The questionnaire for the software companies was designed based on preliminary survey, that need to obtain information on the company in respect of: software development process in general, project management practice and challenges , the staff profile ,the type of project participated in, requirement engineering process and applied techniques and methods in the software development process. All the above stated

company practice and other related information is important factors that affect the success of the software project.

Software development is an applied discipline and inherently practical. Therefore, quantitative data from the surveys alone may not provide the in-depth understanding and contextual information essential for revealing how software development projects are practiced in reality . Therefor interview and discussion is conducting to get better in depth understanding

Document review: review all software engineering and development process related documents in the organization are reviewed because it shows the process in organization in the software development

1.8.3. EVALUATION AND TESTING

The data collection completed in the case study, furthermore analysis, interpretation and summarization of the data performed and used as an input for main research area. Based on the survey result and literature design a model and finally test the model.

1.9. THESIS ORGANIZATION

The thesis is organized as follows; chapter two contains review of literature that shows the research direction and problems area that will be solved by this research.

Chapter three contains related works that was identified based on literature review and preliminary survey and continued chapter four an industrial survey and assessment of the organizational practice in the software development project.

The Fifth chapter design the model based on the industrial survey and literature review and the test and experiment of the model are discussed in chapter six. Finally, the conclusion made on the thesis result, the contribution of this research work and recommendation on possible future work related to the thesis are presented in chapter seven.

CHAPTER 2

LITERATURE REVIEW

In this chapter, I attempted to address detailed review on the concepts related to organizational development or change, software development and organizational development, major reason of software failure, requirement engineering and software project success, measurement of requirement engineering process and any other related literatures that support this research

2.1. SOFTWARE DEVELOPMENT

The software environment in Ethiopia has a long way to go to reach the minimum maturity level expected to meet the demands from users and the industry. According to the findings, the software development situation so far is mostly dominated by expected improvements in business efficiency and value as a result of the software introduction that never materialized; substantial budget and time overruns far beyond expected; delivery of unfriendly and poor and thus unused quality software products; difficulty on the part of the users to effectively utilize the newly deployed software systems (Biru, 2008). In the software development process requirement engineering plays a vital role for the project success because requirement engineering a foundation of any software development and most stated problems in the above paragraph related to requirement engineering process.

The process of eliciting, analyzing, specifying, validating and maintaining requirements is known as Requirement Engineering (RE). The main goal of requirement engineering is to meet the degree of end user's or stakeholder's satisfaction in minimum cost and time.

The Standish survey pointed to software project practices that needed improvement. According to the respondents, the most frequently stated factors that influenced project success were user involvement, executive management support, and a clear statement of requirements. These factors show that requirements engineering is crucial to achieve project success.

2.1.1. COMPANY STANDARD

(Biru, 2008)most of the companies were observed not to use any 'standard' method in the actual conduct of the software development projects. A commonly employed approach involved the use of cyclical requirements gathering and then programming practices, with a strong tendency to rely on in-house-developed practices rather than industry standard guidelines on this reason for projects are delayed by because Changing requirements ,Difficulty of users to clearly articulate their requirements and high staff turnover All stated are highly related with requirement engineering process,

2.1.2. STAKEHOLDER INVOLVEMENT

Stakeholders are individuals who affect or are affected by the software product and therefore have some level of influence over the requirements for that software product. The requirements engineering process provides the best opportunity to consider all of the various stakeholder's interest in context with one another. There are three main categories of stakeholders: the acquirers of the software product, the suppliers of the software product, and other stakeholders.

(Muneera Bano, 2013)Involving users in software development life cycle (SDLC) has been suggested to improve the quality, accuracy and completeness of requirements , that should result in users' satisfaction. Users are meant to have a great deal of domain knowledge about the tasks they perform, work practices, context of the system use and their behavior and preferences. Typically a user is someone who would ultimately use the system being developed or would be in some way affected by it. When you select user or user representative he/she must be enthusiastic to express the domain knowledge or business process very well in this condition the user participation has great contribution for project success.

The acquirer type stakeholders can be divided into two major groups. First there are the customers who request, purchase, and/or pay for the software product in order to meet their business objectives. The second group is the users, also called end-users, who actually use the

product directly or use the product indirectly by receiving reports, outputs, or other information generated by the product.

The suppliers of the software product include individuals and teams that are part of the organization that develops the software product or are part of the organizations that distribute the software product or are involved in other product delivery methods (for example, outsourcing).

IDENTIFYING STAKEHOLDER

The first step in identifying the stakeholders is to make sure that one considers all of the potential stakeholders.

The second step in identifying the stakeholders is for the practitioners to decide how they are going to deal with these conflicts. They accomplish this by determining which stakeholders have higher priorities based on their contribution to the success of the software product. This allows practitioners to make appropriate trade-offs when conflicts occur. (Weinberg, 1989)

The third step is to decide who will represent each stakeholder group that one has designated as friendly or unfriendly. There are three main choices:

Representative Select a stakeholder champion to represent the group of stakeholders. For example, if there are multiple testers who will be testing the product, the lead tester might be selected to represent this stakeholder group. The lead tester would then participate in the requirements engineering activities and be responsible for gathering inputs from other testers and managing communication with them.

Software practitioners must determine when the stakeholder group needs to participate in the requirements engineering activities. Are they going to participate throughout the entire process or only at specific times?

The final decision is to establish the priorities of the stakeholder team members based on their relative importance to the success of the software product

2.2. REQUIREMENT ENGINEERING PROCESS

2.2.1. PLANNING PHASE

Planning phase is the first phase of RE, Four types of stakeholders are identified during planning phase, i.e. the executive sponsor, the analyst, the domain experts (DEs) and the user representatives (URs). The executive sponsor is the manager or executive who is responsible for making executive level decisions and commitments. The analyst is responsible for different tasks of RE. A domain expert is a person who can provide detailed information on a narrow, well-defined topic. They have the best available view of a particular domain area.

It should also answer the following

- Does the overall objectives of the organization is satisfied by the system?
- Can the system be developed within the proposed budget and timeline?
- Can the proposed system made to co-exist with the existing systems?

2.2.2. REQUIREMENT ELICITATION

“Requirements elicitation is defined as a process to understand a problem and its application domain. The goal of requirements elicitation is to identify as many requirements as possible to prepare several alternate solutions for the stated problem. The requirements will be written in a user requirements document (URD) as the output of the requirements elicitation activity” (Kasirun, 2005)

“Elicitation techniques are tools of finding & exact understanding. The goal of Elicitation technique is to find out as many problems as possible so that it could become easier for stakeholder to get the best suitable application according to the requirements (Int) (Shadab Khan[1], 2010)

TYPES OF ELICITATION APPROACH

There are different ways to get the required information and approach problem. One is direct approach, second is indirect approach. First one classifies the methods by whom we interact with the domain expert and second one classifies them by what type of information is obtained.

DIRECT APPROACH

In direct approach the purpose is to enhance the understanding of the problems of system that is currently in used. Most common techniques used are Interviews, case study, Prototyping. With these tools a comprehensive and comprehensive analysis of total procedure can been done. In this loom it is good to get the more knowledge about system and genuine data. In order for these methods to be victorious, the domain expert has to be reasonably coherent and willing to share information (E. Burge, 2009).

INDIRECT APPROACH

Indirect methods are used in order to obtain information that cannot be easily articulated directly. Questioners, documents analysis are its examples. Important thing in this approach is, how thing are clarify by using figures and statistics. In it a large quantity of data can be gathering from analyzing the documents. The results acquire from this type of investigation are easy to measure and an applicable test suggestion can be driven from them.

CATEGORY OF ELICITATION TECHNIQUES

CLASSIC REQUIREMENTS ELICITATION TECHNIQUES

These requirements elicitation techniques have been used for a long time. These are tested and proven methods. The different classic requirements elicitation processes are, Interview, questioner and social analysis or observation

MODERN REQUIREMENTS ELICITATION TECHNIQUES

These requirement elicitation techniques have been used before few years and they are better than classic approach when the stakeholder could not articulate clearly the real need or requirement during requirement elicitation phase , Proto typing, scenario , requirement reuse and brainstorming are modern requirement elicitation techniques

2.2.3. REQUIREMENT ANALYSIS

Stakeholders analyzed the requirements for completeness. Completeness means that no requirements that are needed have been omitted, i.e. whether the elicited requirements have covered all of the needs and objectives of the organization, application domain and stakeholders? (Kotonya and Sommerville 1998)

Stakeholders analyzed the requirements to see if the elicited requirements contribute to the business goals of the organization, i.e. whether the elicited requirements satisfy the needs and objectives of the organization, application domain and stakeholders. It is also analyzed to ascertain whether the elicited requirements are in fact necessary and solve the specific problems.

At the foot of the analysis checklist the stakeholders have to provide general comments about the initial requirements statement. In general comments the stakeholders give their point of view about the elicited requirements and mention whether or not they agree with the elicited requirements, or want further modification. If they want further modification then that modification is specified.

During analysis the requirements engineer aims at understanding how the requirements will be implemented by the software system , how they will be considered in the development plan and how they will be used for the testing of the system . Requirements analysis typically results in one or more prototypes, a definition of project scope or release plan, and a requirements specification for the system. (Samuel A. Fricker, 2015)

2.2.4. REQUIREMENT SPECIFICATION

A software requirement specification is a document containing a complete description of what the software will do without describing how it will do it and it also clearly and precisely describes each of the essential requirements (functions, performance, design constraints, and quality attributes) of the software and the external interfaces.

2.2.5. REQUIREMENT MANAGEMENT

Requirements management according to Sommerville and Sawyer, is “the process of managing changes to a system’s requirements” effective requirements management as: Managing changes to agreed requirements ,Managing the relationships between requirements, Managing dependencies between the requirements document and other documents produced during the systems and software engineering process.

2.3. REQUIREMENT ENGINEERING PRACTICE

2.3.1. REQUIREMENT ENGINEERING AND PROJECT COMPLEXITY OR SIZE

"The cost of the requirement engineering depends on the size and the type of the system that is being developed. For large systems it wills costs 15% of the total budget only for formal requirement specification, for smaller systems it varies from 8 to 10 percent" (Gunda, 2008)From the survey conducted and discussions held with practitioners, A project size and complexity has an impact on the requirement engineering process and in the case of large scale project RE process become major task in the project hence it needs large resource , efforts and experts in order to understand and properly specified the business level, user/process level and product level requirements refer figure . However, It does not mean that small scale projects has not affected by requirement engineering process but the extent of the effect is less because requirement is not complex and large hence eliciting and specifying the requirement is not a such difficult but still there is problems that related to requirement engineering process .from the survey, most organization assign less than 2% of the total cost for requirement engineering process because it does not considered as major tasks and sources of errors in the project

Major requirements are most software practitioners just talk about "the requirements." However, by recognizing that there are different levels and types of requirements, as illustrated in Figure 1 adapted from (Wiegers, 2004), practitioners gain a better understanding of what information they need to elicit, analyze, specify, and validate when they define their software requirements.

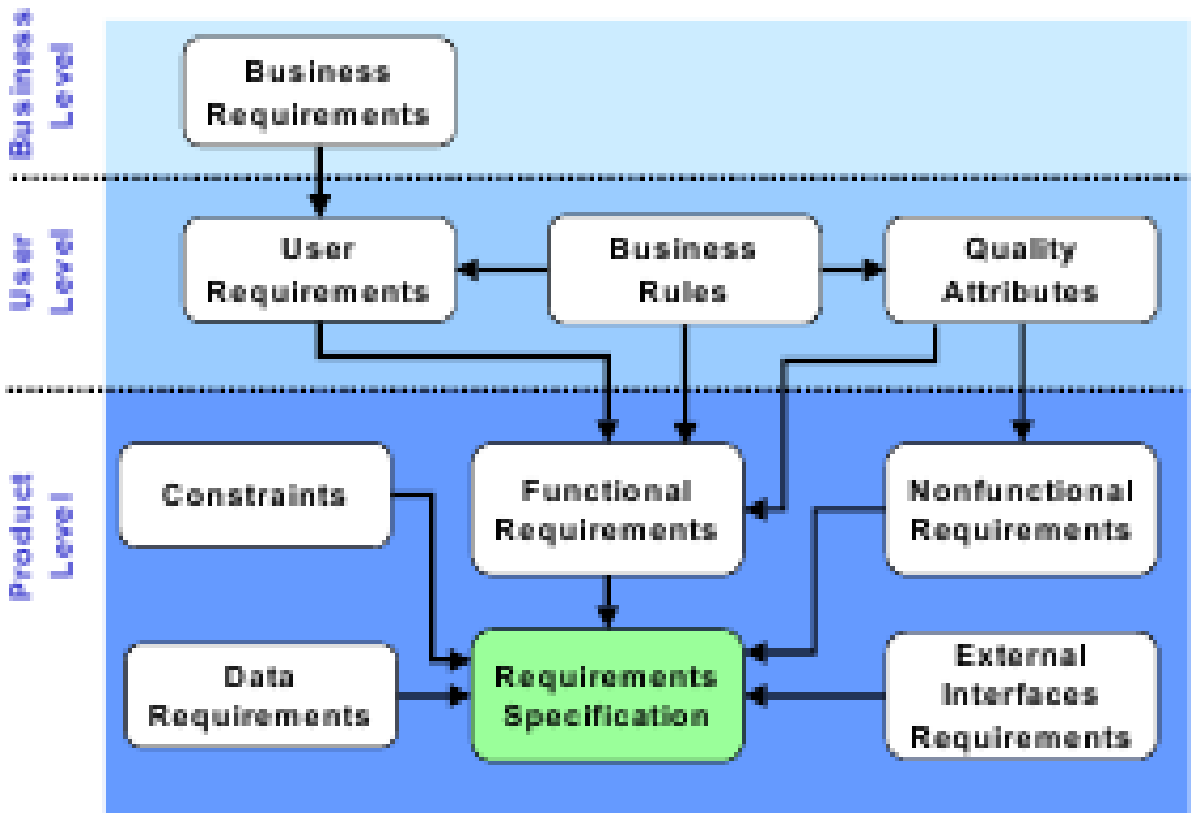


Figure 2-1: Type and Level of Requirement

TYPE OF REQUIREMENT

Business level requirement: define the business problems to be solved or the business opportunities to be addressed by the software product. In general, the business requirements define why the software product is being developed. Business requirements are typically stated in terms of the objectives of the customer or organization requesting the development of the software.

User level Requirement: User requirements look at the functionality of the software product from the user's perspective. They define what the software has to do in order for the users to accomplish their objectives. Multiple user-level requirements may be needed in order to fulfill a single business requirement.

Product level requirement: define the software functionality must be built into the product to enable users to accomplish their tasks, thereby satisfying the business requirements. Multiple functional level requirements may be needed to fulfill a user requirement.

2.3.2. REQUIREMENT ENGINEERING AND ORGANIZATIONAL CAPACITY

From the survey conducted and discussions held with practitioners, most organizations are very young and limited capacity such that they have less number of professional staffs and participating in the small scale projects , the requirement engineering activities are almost ignored except few of them are represented the requirement using use-case diagram. Use-case diagram is not complete and it represents only functional requirement of the system, the remaining requirement is not represented and specified, it become incomplete requirement this will lead project rework and extending the schedule even though the cost and effort of rework is not such big relative to large project. Very few organizations are large capacity and participating in large projects which means the project has large number of functionality and high costs and they have large number of professional and experienced staffs but still they don't have plan, standards and managing requirement engineering in the software development process. Even they don't have Requirement engineering experts or analyst and there is no capacity building training on requirement engineering process which help the system analyst completely specify requirement of the new system.

2.3.3. REQUIREMENT ENGINEERING AND PROJECT SUCCESS

When requirements are poorly defined and RE processes are ad hoc, the end result is nearly always an unsatisfactory product or a cancelled project.

Deficient requirements are the single biggest cause of software project failure. From studying several hundred organizations, Capers Jones discovered that RE is deficient in more than 75 percent of all enterprises. In other words; getting requirements right might be the single most important and difficult part of a software project (Jones, 1996) Successful projects were those that completed on time and budget and produced a software product with all features and functions as initially specified.

2.3.4. REQUIREMENT ENGINEERING AND PROJECT SCOPE

A well-defined scope sets expectations among the project stakeholders. It identifies the external interfaces between the system and the rest of the world. The scope definition helps

the project manager assess the resources needed to implement the project and make realistic commitments (Wiegers, 2006) .One of the Goal of requirement engineering is a clear scope of the project

In this chapter, an attempt was made to reviews of literature that support this research. The literatures focus on the following areas, organizational context component and development in the software development perspective, software development practice and stakeholder's involvement and requirement engineering process and practice in with respect to software development.

2.4. ORGANIZATIONAL CONTEXT AND DEVELOPMENT

2.4.1. ORGANIZATIONAL CONTEXT

No software development process exists in isolation. It derives its meaning from context or it is influenced by the context (Biru, 2008) . Each dimension of work, the human actions and interactions that can be relevant to requirements bits of knowledge that should be sought during requirements elicitation for a CBS. The following show each dimension of work in the organization

STRUCTURAL

Relevant organizational goals, objectives, and strategies ,Tasks, processes, rules, regulations, and procedures Communication channels and exchanged information, Coordination and control ,Formal roles, How authority is distributed and Needs of system support to work Relevant organizational and technological knowledge to be able to perform tasks are the major structural issues in the organization

SOCIAL

Shared goals and objectives, Personal knowledge and its impact on work concepts, practices and relationships, Fostered participation in decision making, Use of individual and group skills are some of social issues in the organization that relate with RE process

POLITICAL

Conflict of interests and Negotiation processes: concepts, and practice are some political issues that must be consider in the RE process

Form of power held: organizational authority, control of scarce resources, control of the definition of formal arrangements, restricted access to key information, control of organizational borders, control of core activities, member of a strong coalition, charisma.

The RE process structure is the result of the joint effort of system's stakeholders and elicitors for emancipation, fairness, and community empowerment. Its shape is situational, i.e., it varies with organizational history and culture and resources involved.

2.4.2. ORGANIZATIONAL DEVELOPMENT

Organizational development (OT) is the process of fundamentally changing an organization's processes in order to allow it to better meet new challenges. OT is usually accompanied by deployment of so-called enabling CBSs, the OT triggers or is triggered by the deployment of the enabling CBSs, and sometimes the causality is in both directions. Certainly, the OT and the enabling CBSs share many requirements, particularly the higher level ones. RE for either of them is effectively RE for both of them. (Isabel Ramos, 2004)

An organization does OT and deployment of its enabling CBSs to achieve a variety of goals, some of them are: to increase its income, to improve its market penetration, to reduce its reaction time, and to reduce costs, all of which allow it to better meet new challenges. Triggers for OT and deployment of its enabling CBSs include an organization's desires to: implement a new management or business model, adopt a new best practice, satisfy clients better, create a new internal or external image, promote a new social order, obey environmental rules, foster collaborative practices, etc.

From the survey conducted and discussions held with practitioners, most organization OT(specially CBSs) initiatives came from higher officials and management. Main Objective of enabling CBSs in the organization: increase the productivity, improve its market penetrations, easy to manage the process and product information and make the communication between section or unit and management easy and effective. even though the CBSs has so many

advantages there is also some challenge that accepting positively organizational change through CBSs this will have greater impact on the software development activities specially on requirement engineering activities some of the problems are unwilling to express business process requirement ,give incorrect or unclear information ,unable to show all necessary document and other . In order to avoid such problems the higher officials as well as system analyst create an awareness about new software system objective and ensure that no impact on them. This will reduce problems on requirement engineering process and make the requirement process simple and effective this will have greater contribution of the project success.

CHAPTER 3

RELATED WORKS

In the previous chapter, an attempt was made to provide a picture of software project success in the organizational development by considering the following major issue: organizational context, software development process and practice based on literature. Furthermore the following literatures are selected in order to get a full idea of a project success , the relationship between requirement engineering and software project success, requirement engineering best practice and how requirement engineering practice improves are reported in this chapter

3.1. REQUIREMENTS PRACTICES, CUSTOMER AND USERS, AND PROJECT MANAGEMENT

(June Verner, 2005)When requirements are poorly defined and RE processes are *ad hoc*, the result is nearly always an unsatisfactory product or a cancelled project. A Standish Group Report revealed that three of the top ten reasons for challenged projects or outright project failure were lack of user involvement, unstable requirements and poor project management.

Requirements practices, the sponsor and customers/users, and project management are the major parameters in order to assess the project success in the research. In the .Although good project management necessitates that requirements are complete and consistent, gathering requirements with a specific methodology (R1) was not significantly correlated with project success. However, in 49% of projects respondents did not know what requirements methodology was used. For the ones that did know, four projects used prototyping and eleven used JAD sessions with prototyping; for the remainder of projects, interviews and focus groups were the main requirements gathering method. Thirteen of the 15 projects using prototyping and/or JAD were successful. Eight U.S. projects used UML to document requirements; only four were successful. Nearly half the projects began with incomplete requirements (R2). It is therefore not surprising that the scope was changed for many projects.

According to (June Verner, 2005) look the table below

Table 1: Percentage "Yes" Responses to Questions

ID	Question	Success[1] % Yes	Failure[2] % Yes	Total[3] % Yes
C0	Were the stakeholders committed and involved?	66	57	63
C1	There was a high level of customer/user involvement	80	47	69
C2	There was a high level of customer/user confidence in the development team	70	29	56
C3	There was a low level of customer/user turnover	73	55	65
C4	Senior level project sponsorship lasted right through the project	80	50	70
C5	You were affected by large numbers of customers/users	29	33	30
R1	Were requirements gathered using a specific method?	53	50	52
R2	Were requirements complete and accurate at project start?	47	25	40
R3	If not complete at start were requirements completed later?	80	23	56
R4	Overall, were the requirements good?	81	28	66
R5	Did the project have a well defined scope?	81	46	69
R6	Did the scope increase during the project?	61	74	66
R7	Customers/users made adequate time available for requirements gathering?	80	42	66
R8	Was there a central repository for requirements?	77	44	66
R9	Did requirements result in well defined deliverables?	79	37	64
R10	Did the size of the project have a negative impact on requirements?	31	52	38
M1	The requirements were managed effectively	86	35	64
M2	Was the project manager experienced in the application area	69	69	69
M3	Was a defined development methodology used?	73	50	66
M4	Was the methodology appropriate for the project?	81	46	65
M5	Was delivery decision made with appropriate requirements information?	67	20	51
M6	The project manager was able to choose the development methodology	41	25	34

According to research result, the best predictor of project success was that the requirements were good together with the requirements were managed effectively (93% of projects were predicted correctly). This survey shows that effective project management is fundamental to effective requirements engineering.

Finally, managing the project size in terms of functionality, applying appropriate development methodology which fit with requirement methodology, scope is well defined when it creeps it means that when you decide the scope you never miss necessary features ; a project manager who manages requirements effectively, completing the requirements at some stage during the

project and a central repository for requirements. All above stated are major success factors of the project

3.2. PROJECT SUCCESS INFLUENCING FACTORS

According to (Samuel Fricker, 2015),the most frequently stated factors that influenced project success were user involvement, executive management support and clear statement of requirement

These Clearly stated requirements contribute to a shared understanding between the project team and the software product's users, management, and other stakeholders. The shared understanding reduces the risk of unsatisfactory outcome and rework of project results

User involvement is critical for building software that will be understood by the users, that will be used appropriately, and that creates joy. Management support is critical to align the software with the strategic goals of the organization. Clearly stated requirements contribute to a shared understanding between the project team and the software product's users, management, and other stakeholders

One of the early influential works describes requirements engineering as inquiry .During an inquiry the requirements engineer asks questions about a future software product to stakeholders and turns the obtained answers into a specification. While doing so, new questions emerge that are posed again to the stakeholders to initiate the next inquiry.

Today, a requirements engineer is expected to elicit needs and expectations from stakeholders, to model and analyze the impact of these inputs on the system together with the development team and to check proposed implementations for acceptance by the stakeholders

During elicitation the requirements engineer aims at understanding the project vision and constraints, the context that the product will be deployed into, and the stakeholders that will need to accept the product. Such requirements elicitation results in an overview of users,

external systems, and other stakeholder viewpoints and a description of their respective background, interests, and expectations

During analysis the requirements engineer aims at understanding how the requirements will be implemented by the software system , how they will be considered in the development plan, and how they will be used for the testing of the system (June Verner, 2005). Requirements analysis typically results in one or more prototypes, a definition of project scope or release plan, and a requirements specification for the system.

During requirements checking, the requirements engineer checks that the right approach has been selected for fulfilling the vision and achieving the system goals and that the system will be accepted by the stakeholders. Requirements checking initiate a new inquiry cycle if the checked requirements turn out to be not good enough.

From the research finding, most projects elicited the requirement to do with stakeholder workshops, by studying existing systems, or by reusing specifications. In the Project planned the product to be developed, often by prioritizing requirements. Often a mix of planning techniques was used, analyzed requirements. Often a mix of informal modeling, prototyping, and object-oriented analysis was used; project specified requirements a majority specified functionality, quality, use scenarios, and user interfaces of intended solutions but functional requirements dominated, for specifying the requirements, natural language dominated as the notation. Natural language was often complemented with UML diagrams. In the Project stored requirements. Requirements documents were the most common type of storage. In the Project checked requirements. Projects tended to prefer manual requirements checking, preferably with rigorous inspections

REQUIREMENTS ENGINEERING SUCCESS

The most important requirements engineering goals were shared understanding between the project team and its stakeholders and good quality of the requirements specification. These two objectives of requirements engineering were often complemented with the need for a clear scope for the development, for using little time and resources for requirements engineering, for satisfying the users, and for delivering the requirements engineering work

results in time. The most common goal pursued by the software products that were specified with the requirements was productivity improvement

Scenarios are exemplary sequences of system usage. In requirements engineering, they are used to describe concrete stories of how users and external systems interact with the system under consideration to achieve goals that are of value to the user. Scenarios make the functionality of the system concrete and thus enable users to judge whether they feel to be able to use the system meaningfully and whether they like it. Scenarios also allow capturing interaction design knowledge from user experience experts. A common format used to document scenarios is the use case template (June Verner, 2005).

Business cases are used to document predicted financial results and other business consequences for one or multiple alternative ways of how a product is built, deployed, and maintained (June Verner, 2005) The business case planning work and results that are obtained with it are determinant for selecting what is in the product scope and what not and for evaluating whether a chosen scope is attractive for the customer of the project. The understanding of a business case allows to stop work on a product that does not make sense or to re-scope the product to make it more attractive.

Workshops create an efficient, controlled, and dynamic setting for quickly eliciting, prioritizing, and agreeing on requirements. The discussion of requirements by the critical stakeholders makes a requirements workshop to be one of the most efficient techniques to perform inquiry and to achieve shared understanding. No other technique allows exposure and resolution of conflicts between stakeholders so efficiently. The same applies for discovery and resolution of misunderstandings.

According to result of the research, only three techniques (Scenarios, Business cases and Workshops) correlated with requirements engineering success. None of the other techniques correlated significantly with success, and no technique correlated negatively

3.3. KEY PROCESS AREA IN REQUIRMENT ENGINEERING

TO SUPPORT A GOAL-BASED APPROACH IN THE RE

(Mythili, 2014) Goals are useful for organizing and justifying requirements. Goals have been introduced into requirements engineering for a variety of reasons, i.e. requirements acquisition, relating requirements to the organizational and business context, clarifying requirements, documenting requirements, dealing with conflicts, assisting the management of change and driving the initial design (Mylopoulos, 1998) and (Kasirun, 2005) Normally it is difficult for the stakeholders to fully understand the requirements of the organization or application domain but with clear goals a good understanding can be obtained. (Lamsweerde, 2001)

TO SUPPORT THE INCREMENTAL AND CYCLICAL BEHAVIORS IN THE RE PROCESS

(Potts et, 1994) Have proposed a cyclical model, called the Inquiry Cycle that consists of three iteratively repeated activities: expression, discussion and commitment

The incremental behavior is regarded as the most realistic approach to software development for large-scale systems (Pressman, 1997)

TO ENCOURAGE STAKEHOLDERS INVOLVEMENT IN THE RE PROCESS

Stakeholders' involvement in the RE process is one of the most important factors that contribute to the success of the project and reduce rework of the project because creating shared understanding between stakeholder and developer team

In most case the concerned stakeholders are not involved in the RE process and their real needs are not considered in the system (al, 1997). Involving the stakeholders in the development process can reduce their fear for example that the development of a software system will result in loss of jobs.

TO SUPPORT THE MANAGEMENT OF RE PROCESS

During the RE process new requirements emerge and existing requirements change at all stages of the system development process. It is often the case that more than 50% of system's requirements will be modified before it is put into service (Sommerville, 1998).

The RE process is a learning process, and ideas generated at one point may change at another point. This evolution of requirements throughout the whole software development life cycle has to be managed in order to ensure high-quality specifications. The management includes issues such as information storage, organization, traceability and documentation. Requirements management may look like an overhead in the RE process, but it is usually rewarded by better customer satisfaction and lower system development costs.

TO DEFINE A PLANNING PHASE FOR THE RE PROCESS

Normally the RE process is started without any planning and the requirements engineers inevitably wish to start very quickly. The RE process will be an unproductive exercises if started haphazardly and without planning. Particular attention should be paid to the planning of the RE process

Based on preliminary survey and literature review the following four parameters are selected

Requirement engineering practice is foundation of every software development process within the software company.

User involvement or participation: is very important factor of project success, it address when how and why users are participated in the project

Organizational context: it address business process representation and managing business process change in the software development process, resolving conflicts of interests and checking organizational goal.

Table 3-1 Summary of Related Works

Authors	RE Practice	User Involvement	Organizational Context
(Muneera Bano, 2013)	✓	✓	
(Isabel Ramos, 2004)		✓	✓
(Muhammad Waqas Boota, 2014)		✓	
(Mythili, 2014)		✓	
(Samuel A. Fricker, 2015)	✓	✓	
(June Verner, 2005)	✓		

As can been seen from the above table Most authors identify, user involvement are important factors in the software project success and three of them identify requirement engineering practice Is a fundamental and crucial parts in the software development, even though requirement engineering practice and user involvement is important, the skill or knowledge of Practitioner or analyst and organizational context has an impact on project success.

Participating user in the software development is not the only success factors rather when, how and who participate in the development and how the users are treated in order to get accurate and complete requirement is the main issues ,this will be achieved when the analyst have enough knowledge and skill on the requirement engineering concepts, methods and tools . There is gap between theory and practice this may have different reason some of them are lack of skills, knowledge, awareness and others factors

In the software requirement engineering practice, most software firms do not have an industry standard and lack of applying the scientific knowledge in the software development process as a result of this it become difficult to enable CBS for organizational development.

In the software development process, requirement engineering process is fundamental and highly related with organizational context therefore the requirement engineering process should be align with organizational context because the main source of a new system requirement in the organization is users, if users are participating and can be expressing the requirements very well this will creates shared understanding between stakeholder and development team and reduce rework and change request during the submission of the project

As can been see, in the table 3.1, most authors did not see requirement engineering practice, the organizational context and practitioner or analyst gap issues in the project success. When we look organizational context can be seen from customer and vender sides ,from customer side the organizational context : Structure of the organization, organizational change such as structural , business process, standards or polices from vender side: guideline on RE activities , assessing RE activities that has an impact on design and architecture process. Every organization has its own goals, structure, business process, polices and standard, the software must be fit with organizational context.

Finally (Muneera Bano, 2013) identify requirement engineering practice and user involvement has an impact on the project success But he does not mention how, when and where involved (Isabel Ramos, 2004) Describe user involvement is major affecting factor in the software development but he does not see other related factors that affect the software project.

In general measuring the requirement engineering process by considering organization context is very important factor that make the software project success full.

CHAPTER 4

SOFTWARE INDUSTRY SURVEY

In this chapter, an attempt was made to provide the current software development practice in the software firms (Addis Ababa) and software development section of ICT office in different universities and organizational, different organizations from customer's side in Ethiopia are participated in this research survey.

BACKGROUND OF ORGANIZATIONS

In the industrial case study ,A total of two client organization ,13 software companies,4 ICT software development office of the university and 91 professional are participated , Applicable questionnaires were sent to all (hand delivered as all of the respondents were in Adama and Addis Ababa)from to expedite the process, arrangements were made to physically collect the completed questionnaires. Each questionnaire was accompanied by an explanatory covering letter that stated the purpose of conducting the survey. An overall response rate of 93% ,87and 91% was obtained from software companies, ICT office and software development practitioners, respectively.to physically collect the completed questionnaires .

The interviews and discussions conducted for collecting data for this study included structured one-to-one interviews with selected developers, designer, project manager managers of IT Departments in selected organizations and software development companies.

4.1. CASE STUDY

The in-depth understanding and contextual information essential for revealing how software development projects are practiced in reality. (Biru, 2008)

During the primary survey and discussion held with practitioners, try to see different software development organizations found in Addis Ababa participating in the software development and classify them into two groups (local standard), large number of organizations are small in size and less capacity which means they have less than 5 IT/computing graduated staffs and participated in small scale projects which means less number of application package and they

are mostly private-owed organization. The other category is middle in size and better capacity than the rest and very few in numbers. They are private and government owed organizations, large companies are engaging into large projects where as small companies mostly engaging in small scale projects.

When a given organization are participating in large scale projects, requirement engineering process become complex and large due to this management and specification of Requirement engineering process become difficult and the effect of requirement engineering problems also become high. By considering this issues selected four organizations (A, B, C, D) for the analysis and further discussion in this research. For ethical reasons, the company identities and ownerships of the cases reported are deliberately kept anonymous .Organization A,B,C was in software development industry and organization D was in manufacturing industry, organization A was privately-owned, they have experienced staffs and participated in large projects ,organization B is government and developing large and a variety of software's for most government organization ,they have large number experienced staffs. Organization C small in size and less capacity found in the government university and develop software for automating the internal manual operation of the universities and sometimes they have also customization of the project for other universities and organizations, Organization D was in the manufacturing industry they have hundreds of employee and large number of customers. Two projects were investigated in this organization, one project is completed and the other is partially completed. This organization selected in order assess from customer perspectives

4.2. ORGANIZATION OF THE QUESTIONNAIRE AND INTERVIEW

The interviews and discussions conducted for collecting data for this research included structured one-to-one interviews with selected programmer, analyst and project manager of software development companies and also discuss with managers of IT Departments from customer organization. In particular, selected individuals from those that had already participated in the questionnaire survey were given more chance in the interviews to qualify their responses through providing explanations or examples .finally you can see the result of the questioner in the next topics

4.3. DISCUSSION ON CASE STUDY AND LITERATURE REVIEW

4.3.1. DISCUSSION ON CASE STUDY

ORGANIZATIONAL CONTEXT

The implementation of CBS in the organizational development is successful only if the software's fit for the solution. From the survey conducted and discussions held with practitioners, try to see organizational context from both sides, from customers and venders.

From customers side, conducted interview and collect questioner from two customers of software development organizations those are selected for this industrial survey ,the following issues are identified, inability of users articulate the real needs, organizational business process change, lack of follow up from higher official/decision makers. Important persons of the project leave the organization or change the positions and structural change like merging and expansion of department/section in the organization and lack of experts to specify and requesting expected deliverable that help to manage or control project progress. Business process may change because of market competition, technology change standards and policies or rule of the organization as well as government changes

From software vendor aspect, the following major issues are identified and important in the project success ,organizational standards and guidelines of the project, staff turn-over, evaluation and management of deliverable ,skill or knowledge of staffs ,understanding of project scope and complexity at the beginning of the project and training of staffs, major factors that affecting the software development activities

Table 4-1 Respondents View on Staff Profile

No of IT/computing Graduates	% From total company
<=5	63%
>5 and less than 10	27%
>10	10%

In order to assess software firms and software development section in the organization, considering basic issues such as: software development practice, staff profile and type of project participated in and requirement engineering practice. From this survey data most organization gives only high emphasis on coding and the rest of software development activities has very less emphasis as a result of this problem on shared understanding between stakeholders this will lead continues change request and high rework this survey finding match with the literature (Biru, 2008), almost major reason for delay is highly related with requirement engineering

As can be seen most organization (73%) have less than <= 5 IT/computing it imply that they have less capacity .it is difficult to participate on large scale project with this numbers of staffs.

Based on finding on the preliminary survey, try to see requirement engineering practice in the selected three software companies. These three companies participating large projects and they have basic requirement engineering practices but there is a problems related to RE process then prepare questioner that assess the practices in the organization

Based on respondents, Waterfall process model is the most commonly used process models followed by incremental, agile, RAD, evolutionary prototyping, unified process models. When you use waterfall model the requirement must be well-known on upfront otherwise it is difficult to accept and implement changes, regarding scope 53% respondent show scope instability in the project , it means that lack of shared understanding between developers and stakeholders, scope of the project are expected clearly identify during requirement analysis phase but if the project has large number of application package, it may extend until the end of system design .the project requirements are not properly elicit and analysis during the first phase of project scope instability always happen in the project.

Different organization have different guideline, based on the respondent 12% of the organization have requirement, design and coding guideline and most software companies does not have any guideline in the organization , the main purpose of guideline is creating common language that makes communication and integration become easy in the organization The new staff joins the organization, he/she can easy to understand and work with the existing project with the help of guideline. Based on respondent data, when you see completed project and compare with original tender document there is major variation because inability of user to articulate requirements, lack of requirement engineering skills and changes in the organization since commencement of the project, Requirement analysis document, design document and operational manuals are deliverable in addition to the software. Based on question number 15, There is no standard format to initiate, discuss and approve changes rather management, users and developers are discuss and decide about changes based on Q16 project delay because of Poor planning because of unclear / incomplete requirement and followed by continues requirement change because of lack of user involvement and poor requirement engineering process and staff turnover are the major reasons

Table 4-2 Reason for Delay

Reason for Delay	%
Poor planning because of unclear / incomplete requirements	51
Continuous Requirement change request because of Poor requirement engineering process	48
Continuous Requirement change request because of lack of user involvement	42
High project staff turnover	31
Lack of cooperation from users or stakeholders	27
Problems within the development team	20
Organizational instability or business process change	17
Poor planning because of lack of project management skill	10
Lack of proper progress monitoring and control	10
Inadequate budget to finance additional work due to changes in User requirements	8
Lack of use of appropriate software development methodology	18

Based on survey result on the above table, major reason for projects delayed are incomplete requirement , poor requirement engineering process , lack of users/stakeholders participation and staff turnover on the project. In this finding the continues requirement change due to lack of user involvement and lack of emphasis on requirement engineering is major factors when we compared to other related works

Based discussion held with project managers and team leaders, requirement engineering has got less emphasis and budget. According to “The cost of the requirements engineering depends on the size and the type of the system that is being developed. For large systems it wills costs 15% of the total budget only for formal requirement specification, for smaller systems it varies from 8 to 10 percent” (2000)

Based on survey, the requirement engineering process assessed and the following are major findings

From response of different professionals: during requirement elicitation process, document analysis and interview are the dominant methods followed by questioner and workshops.

Table 4-3 Stakeholder Participation

Stakeholders Participation	100 %
At the beginning of the project	78
At the end of the project during acceptance testing	63
Only the time of requirement elicitation	40
Whenever necessary	24
Scheduled based	16

Table 4-4 Requirement Elicitation Techniques

Requirement Elicitation Techniques	% of Techniques
Document Analysis	79
Interview	77
Questioner	32
Workshop	18
Prototyping	11
Requirements reuse	21
Scenarios	14
Brainstorming	4
User Centered Design	1

User participation is very critical success factors of the project." Involving users in software development life cycle (SDLC) has been suggested to improve the quality, accuracy and completeness of requirements"] (S. Kujala, 2003).When I observe table 4.3, participation of users are mostly at the beginning and end of the project therefore it is difficult to find out what the users really wants from the system ,this will create high rework and users dissatisfaction .If you observe table 4.3 document analysis is and interview requirement elicitation techniques are dominated and followed by questioner .but prototyping and scenario are more prefer than

other requirement techniques because they can easy to represent the users and system interaction .

Table 4-5: Requirements Analysis

Requirement Analysis Practice	100 %	Modeling Techniques	100 %	Validation Practice	100 %
Model the requirements	83	Object Oriented OOA	87	check that requirements document meets your standards	57
Allocate requirements to subsystems	45	Structural SA	12	organize formal requirements inspections	23
Define system boundaries	41	Prototyping	1	Demonstrate requirements for validation	12
Use checklists for requirements analysis	36			define validation checklists	8
Classify requirements	21			Use multi-disciplinary teams to review requirements	7
Prioritize requirements	19				
Perform any risk analysis on requirements	10				
Use interaction matrices to find conflicts and overlaps	1				

Requirements analysis involves refining the requirements to ensure that all stakeholders understand them and scrutinizing them for errors, omissions, and other deficiencies .During

analysis the requirements engineer aims at understanding how the requirements will be implemented by the software system, how they will be considered in the development plan, and how they will be used for the testing of the system. Requirements analysis typically results in one or more proto types, a definition of project scope or release plan, and a requirements specification for the system.

Table 4.6 Requirement Specification and Techniques

Requirement Specification Practice	**100%**	**Specification Techniques**	**100%**
Have standards templates / documents for describing requirements	15	Natural Language	`73
Have guidelines how to write requirement	12	UML	80
Uniquely identify each requirement	22	Structured analysis Diagram	12
Use diagrams appropriately	23		

As can be seen from the above table, natural language and UML are the most commonly used specification techniques for specify the requirement in project, but when you consider specification practice, most companies does not have standard or template document in the organization and it is difficult to change and trace the requirement the document, adapt SRS template the document Define a standard template for documenting software requirements in your organization. The template provides a consistent structure for recording the functionality descriptions and other requirements-related information. Rather than inventing a new template, adapt an existing one to fit the nature of your projects. Many organizations begin with the SRS template described in IEEE Standard 830-1998 (IEEE 1998b),it must be uniquely identify and identify source of requirement in order to ensure that all stakeholders know why every requirement belongs in the SRS and to facilitate further clarification, trace each

requirement back to its origin. Allocate requirements into subsystem; this will become an input for design of the system

Table 4.7 Requirement Management

Requirement Management Practice	**100%**
Baseline and control versions of requirements	37
reuse requirements over different projects	34
have defined policies for requirements management	13

As can been seen from the above table , very few organizations has Baseline and control versions of requirements and have defined policies for requirements management but most organizations only reuse requirement over different project and whenever change request come, the deal with customer, managers and programmer's then agree on change , this will lead the project delay because there may be continues change request unless there is no defined standard or policies for accepting and implementing requirement changes in the organization .

Finally ,based on survey and discussion held with practitioner in the software companies, some of the major problems are : lack of awareness and knowledge on RE process, difficult to trace requirement, difficult to integrate sub-system or modules, much errors found in the system and fixing errors become difficult, there is no standard and project delay .

Table 4-8 RE Practice Gap Analysis

Organization	size of Project	Req. Elicitation	Req. Analysis	Req. Specification	Req. Management
A1	High	2/5	3/10	3/9	2/10
A2	High	3/5	4/10	3/9	2/10
A3	Medium	2/5	2/10	2/9	1/10
B1	High	3/5	4/10	2/9	2/10
B2	Medium	2/5	2/10	3/9	2/10
C1	Medium	2/5	2/10	2/9	1/10

Note: the project size (local standard), high more than 4 application module and project cost greater than 6 million birr and medium size project is less than 4 module and cost is less than 6 million and greater than 1 million birr

As can be seen from the above summery of table all requirement activities performed below average this will have greater contribution for change request of the project, project delay and user dissatisfaction.

Based on finding, most organization tender document did not fully express the organization requirement in the software system because professional or consultant did not participate during preparation of tender document, this will create a conflict between customer and vender at the end of the project, most software companies give high emphasis on coding and very less emphasis on requirement engineering process as a result of this large number of change request and rework available during the lifetime of a project, this is the major reason for project delay . The organizational context like structural, social and political issues has an impact on the software development process especially in the requirement engineering process. The software should be fit with organizational solution otherwise it could not get an acceptance

CHAPTER 5

REQUIREMENT ENGINEERING AND ORGANIZATIONAL CONTEXT

In the previous chapter, an attempt was made to provide a picture on problems on requirement engineering process and organizational context in the software development

5.1. ORGANIZATIONAL CONTEXT TOWARD REQUIREMENT ENGINEERING

According to survey result , Organizational context is one of influential factors that affects the requirement engineering process, organizational structure (the organization goal and business process) is used to identify business level requirement , social aspect(shared understanding and goal between stakeholder) used to identify user level requirement based on business level requirement , political aspect(conflicts of interest).

From survey and discussion held with practitioner, there is a problem on customer and vender organization with respect to requirement engineering process , from customer sides poor tender document preparation ,lack of project follow up and management ,on time decision and delivery of resource, from vendor side there is no standard in the organization it become difficult to integrate and communicate with in the organization, assign less budget and less emphasis on requirement engineering process and assign less experienced staff, lack of knowledge and skill on requirement engineering concepts, tools and methods

Requirement engineering process must incorporate the organizational context in order to achieve success full organization development which means the new CBS system must fulfill the organizational goal and objective by improving business process and productivity. the RE process always consider organizational context that is stated in the 5.1 section .

5.2. PRACTICE GAP AND REQUIRMENT ENGINEERING

There is always a gap between the research and practice because researchers do not have practical knowledge about the environment, practitioner capability and other issue and on the other hand practitioner could not have an experience of implementing and updating their scientific knowledge and skill from research and training institutes. This will creates a gap between practitioner and researcher, Most practitioners does not have enough knowledge and skills about the RE engineering techniques and tools. This will create a problem. Based on survey and discussion held with practitioner most software companies gives high emphasis for coding ,RE is considered as simple and very less important activities in the SDLC therefore they do have enough knowledge and skills of RE concepts, tools and models this will create a problems in the software companies . in order to solve this problem there must be training for practitioner about scientific knowledge and skills this will improve the capacity of practitioner and improve the RE process in the SDLC.

5.3. PROJECT MANEGMENT AND REQUIRMENT ENGINEERING

Project management has a contribution of RE process success, based on survey and discussion held with practitioner, most project manager are missing their roles in the requirement engineering process planning: prepare training plan, work plan and deliverable plan of the RE process. **Controlling**: manage requirement risk, renegotiate with customers and track requirement effort. **Scheduling** spilt project into task and assigner required resource. **Project Follow-up and Management**: prepare deliverable plan for RE process and evaluate the deliverables.

5.4. EVALUATION MODEL

The new requirement engineering process evaluation model is designed. The model considers: organizational context ,project management, practice gap and main areas corresponding to requirement engineering process, in hierarchical ways, It has three major parts: The first part ,main process areas (MPA), it contain major processes such as organizational context , project management ,practice gap ,requirement elicitation ,

requirement analysis, requirement specification and requirement management , second parts, Sub-process area (SPA) , it is the building block of MPA , that create better understanding of MPA and the last part of the model is Action ,bottom part of the model it contains a set of activities in the SPA.

5.5. MODEL STRUCTURE DESCRIPTION

The model structure has designed in the following hierarchy; The model hierarchy has three levels, namely Main process area (MPA), Sub-process area (SPA) and Action. On the top level of the model, there are five Main process areas corresponding to requirements engineering main activities and organizational context . Each MPA is further broken down into several SPAs, which facilitates better understanding. On the bottom level, an action, a smallest unit, denotes a certain activity that should be done or a certain item that should be present.

5.5.1. MAIN PROCESS AREA (MPA)

On the top level of the model, a main process area represent a cluster of related practices in one main requirements engineering activity such as Elicitation. There are five MPAs in the model: Organizational context , Requirement Elicitation, Requirements Analysis, Requirements Specification, and Requirements management

5.5.2. SUB-PROCESS AREA (SPA)

Sub-process area contains closely related actions, which help to achieve a bigger goal. Each sub-process area classified based actions characteristics. In this model 13 sub-Process areas under 5 main process areas

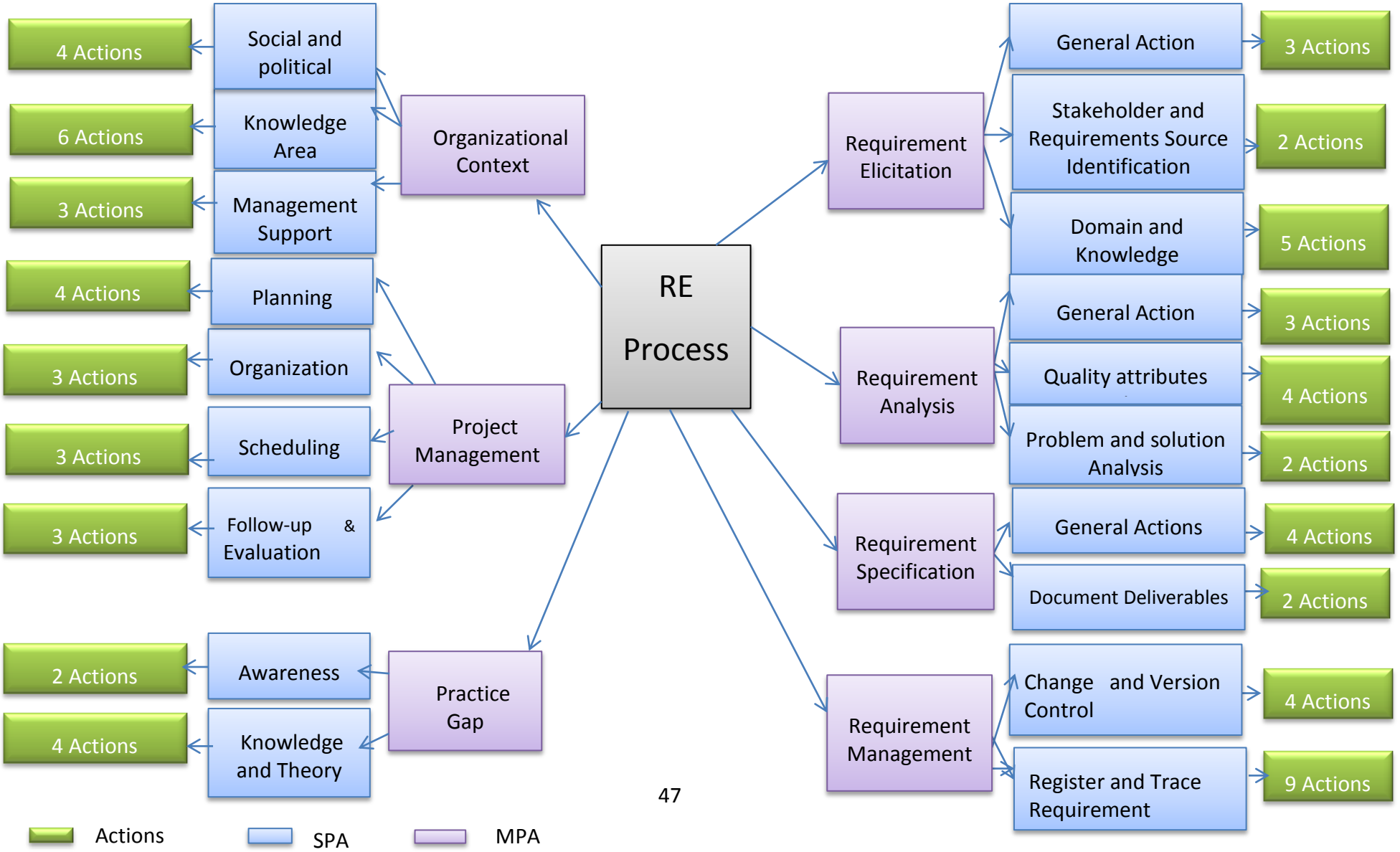
5.5.3. ACTION

The smallest unit in the model is called “action” showing a specific good practice. By performing the action, the organization can improve their process and gain certain benefits. Each action is assigned a certain level depending on its implementation difficulty and essentiality for the requirements engineering process .in this model a total of 56 actions under 13 sub-process area .

Finally ,based on preliminary assessment of software development in the organization and questioner result the following requirement process evaluation model is designed in order to identify the current requirement engineering process and it show the gap in the process this will support the improvement the organization in the requirement engineering process in the project .

5.6. MODEL DESIGN

Figure 5-1: Proposed Model



5.7. HOW THE MODEL WORKS

The model is designed and works in a hierarchical way that improves software development activity. The input of the model is organizational context and requirement engineering parameters which are called Actions, the goal of each SPA depend on those actions performed, if those actions classified and put in each SPA. Actions are best practice of the those parameters in the bottom layer and an input for middle layer and In the MPA there are 13 SPA, for each 13 sub-process area, it has the corresponding actions, in order to achieve the goal of the sub-process area, performing the actions with in that SPA .if you have a good achievement for each SPA then achieved goal of MPA and it has a large contribution for the success of your requirement engineering goals. Furthermore before you proceed to the other SPA, you should achieve at least 50 % action in the SPA otherwise you are not guarantee achieve the goal of the requirement engineering in the software development

Table 5-1 How the Model Work (Example)

MPA	SPA	Total Available Actions	No of Action(s) Performed
Organizational Context	Knowledge Area	7	7
	Management Support	3	1
	Social and Political	4	3
Total		14	10

In the above table, organizational context are achieve 71 % from the three SPA, this imply that it Organizational context MPA is performed in good status but it does not mean that all SPA performed in good status, in this example Management support SPA is performed below average(33%) so you need again see the actions which are not performed in this Management support SPA and try to improve it.

Goals of requirement engineering are shared understanding, specification quality, clear scope, efficiency, user satisfaction, timelines, fit for solution, estimation reliability, Architecture Quality, Cost/Benefit Analysis

5.8. MODEL DESCRIPTION

Description of models adopted from Empirical Evaluation of a Universal Requirements Engineering Process Maturity Model (Nguyen, 2010)

Table 5.2 Model Description

ID	Title	Category
	Organization Context	**MPA**
	Knowledge Area	SPA
OCKA 1	Define tender document with the participation of experts	ACTION
OCKA 2	Create an Organization-wide Glossary of Terms	ACTION
OCKA3	Educate user reps and managers about requirements	ACTION
OCKA4	Train Analyst in Requirement Development and Management Processes	ACTION
OCKA5	Train Developers in Application Domain	ACTION
OCKA6	Prepared Organizational Standard (Coding or Documentation)	ACTION
OCKA7	Prepare Pair programming approach	ACTION
	Management Support	SPA
OCMS1	On Time Decision	ACTION
OCMS2	Assign user representative based on analyst preference	ACTION
OCMS3	On-time delivery of required resource	ACTION
OCSP	**Social and Political**	SPA
OCSP1	Shared Goal and objectives drives from organizational goal	ACTION

OCSP2	Identify Individual and group skill	ACTION
OCSP3	Resolving Conflict of interests and Negotiation processes	ACTION
OCSP4	Identify Power Held	ACTION
PM	**Project Management**	MPA
PMP	**Planning**	SPA
PMP1	Select Appropriate Methodology	ACTION
PMP2	Base plans on requirements	ACTION
PMP3	Review past lessons learned	ACTION
PMP4	Plan a training and workshop by referring knowledge/ skill gaps	ACTION
PMC	**Control**	SPA
PMC 1	Manage requirements risks	ACTION
PMC2	Renegotiate commitments	ACTION
PMC3	Track requirements effort	ACTION
PMS	**Scheduling**	SPA
PMS1	Split project into tasks and estimate time and resources required to complete each task	ACTION
PMS2	Organize tasks concurrently to make optimal use of workforce.	ACTION
PMS3	Minimize task dependencies to avoid delays	ACTION
	Project Follow-up and Management	SPA
PMPFR1	Prepare deliverable plans for the project progress	ACTION
PMPFR2	Evaluate the deliverables	ACTION
PMPFR3	Participation on change and its impact	ACTION
OCPG	**Practice Gap**	MPA
OCPGA	**Awareness**	SPA

OCPGA 1	Create an awareness on required RE skill and knowledge	ACTION
OCPGA 2	Create an awareness on the RE problems and consequence	ACTION
OCPGKT	**Scientific Knowledge and Theory**	SPA
OCPGKT1	Train analyst about RE methods and techniques	ACTION
OCPGKT2	Select methodology/techniques depend on the situation	ACTION
OCPGKT3	Assign experienced staff in the RE process	ACTION
OCPGKT4	Assign enough budget for RE process	ACTION
RE	**Requirement Elicitation**	**MPA**
	General Action	SPA
REGA1	Consider social ,structural and political issues in the organization	
REGA2	Elicit Quality Requirements	ACTION
REGA3	Qualify and Quantify Quality Requirements	ACTION
REGA4	Let Business Concern Guide Focus of Elicitation	ACTION
REGA5	Use Appropriate Elicitation Techniques according to Situation	ACTION
REGA6	Use Artifacts to Facilitate Elicitation	ACTION
REGA7	Create Elicitation Channels for Requirements Sources	ACTION
REGA8	Reuse Requirements	ACTION
	Stakeholder and Requirements Source Identification	SPA
RESRSI1	Identify and Involve Relevant Stakeholders	ACTION
RESRS2	Identify Other Requirements Sources	ACTION
	Domain Consideration and Knowledge	SPA
REDCK1	Elicit Information about System Domain Restrictions	ACTION
REDCK2	Elicit Information about System's Technical Infrastructure	ACTION

REDCK3	Elicit Information about System's Business Process	ACTION
REDCK4	Elicit Information about System's Operational Domain	ACTION
REDCK5	Elicit Information about System Boundaries	ACTION
RA	**Requirements Analysis**	**MPA**
	General Actions	SPA
RAGA1	Perform Requirements Risk Analysis	ACTION
RAGA2	Perform Systematic Requirements Prioritization at Project-level	ACTION
RAGA3	Analyze Requirements Relations	ACTION
RAGA4	Identify Irrelevant Requirements for Early Dismissal (in/out scope OR Triage)	ACTION
RAGA5	Analyze the Strength of Relations between Requirements	ACTION
RAGA6	Perform refinement and abstraction of requirements	ACTION
	Quality Attributes Analysis	SPA
REQAA1	Analyze for Missing and Double Requirements	ACTION
REQAA2	Analyze for Ambiguous Requirements	ACTION
REQAA3	Analyze for Correctness of Requirements	ACTION
REQAA4	Analyze for Testability of Requirements	ACTION
	Problems and solutions analysis	SPA
REPS1	Validate requirements with relevant stakeholders	ACTION
REPS2	Use Checklist to Ensure Quality of Requirements	ACTION
REPS3	Review Requirements	ACTION
REPS4	Create Prototype	ACTION
REPS5	Perform Systems Modeling	ACTION
RS	**Requirements Specification**	**MPA**
RSGA	**General Actions**	SPA
RSGA1	Establish Standardized Structure	ACTION
RSGA2	Define Requirements Attributes	ACTION

RSGA3	Define Requirements States	ACTION
RSGA4	Document Requirements Rationale	ACTION
RSGA5	Record Rationale for Rejected Requirements	ACTION
RSDD	**Documentation Deliverables**	SPA
RSDD1	Define User Documentation Deliverables	ACTION
RSDD2	Define System Documentation Deliverables	ACTION
RSDD3	Define Management Documentation Deliverables	ACTION
RM	**Requirement Management**	**MPA**
RMGA	**Change and version control**	SPA
RMGA1	Define and Maintain Requirements Development and Management Processes	ACTION
RMGA2	Define a Process for Managing Change and Evolution	ACTION
RMGA3	Define and record status of requirement	ACTION
RMGA4	Establish Effective Communication With Requirements Issuers	ACTION
RMGA5	Baseline Requirements	ACTION
RMGT	**Registration and Traceability of Requirement**	SPA
RMGT1	Measure requirement volatility	ACTION
RMGT2	Manage Versions of Requirements	ACTION
RMGT3	Track change requests	ACTION
RMGT4	Requirements Traceability Policies	ACTION
RMGT5	Uniquely Identify each Requirement	ACTION
RMGT6	Document Requirements' Source	ACTION
RMGT6	Define link to other requirement	ACTION
RMGT8	Document Impact of Requirement on Other Artifacts	ACTION
RMGT9	Defining links to other system elements	ACTION

SELECTED ACTIONS ELEMENT DESCRIPTION

Identify user class (RESRSI1): One of the major challenges in the requirement elicitation process identifying the potential stakeholders, the following checklist can help in identifying potential stakeholders: it is very critical factors for project success

- What types of people will use the software product?
- What business activities are supported by the software product and who performs, is involved in, or manages those activities?
- Whose job will be impacted by the introduction of the new software product?
- Who will receive the reports, outputs, or other information from the software product?
- Who will pay for the software product?
- Who will select the software product or its supplier?
- If the software product fails, who could be impacted?
- Who will be involved in developing, supporting, and maintaining the software product?
- Who knows about the hardware, other software, or databases that interface with this software product?
- Who established the laws, regulations, or standards governing the business activities supported by the software product?
- Who should be kept from using the software product or from using certain functions/data in the software?
- Who does this software product solve problems for?
- Who does this software product create problems for?
- Who does not want the software product to be successful?

Draw a context diagram (DCK): The context diagram is a simple analysis model that shows how the new system fits into its environment. It defines the boundaries and interfaces between the system being developed and the entities external to the system, such as users, hardware devices, and other information systems.

Model the requirements (**QAA**) : A graphical analysis model depicts requirements at a high level of abstraction, in contrast to the detail shown in the SRS or the user interface view that a prototype provides. Models can reveal incorrect, inconsistent, missing, and superfluous requirements.

Adopt an SRS template (RSGA1): Define a standard template for documenting software requirements in your organization. The template provides a consistent structure for recording the functionality descriptions and other requirements-related information.

Establish a baseline and control versions of requirements documents (RMGA5): The baseline consists of the requirements that have been committed to implementation in a specific release

Maintain a history of requirements changes (RMGT9): Record the dates that requirements specifications were changed, the changes that were made, who made each change, and why.

Track the status of each requirement. Establish a database with one record for each discrete functional requirement. Store key attributes about each requirement, including its status (such as proposed, approved, implemented, or verified), so that you can monitor the number of requirements in each status category at any time.

Measure requirements volatility (RMGT1): Record the number of baseline requirements and the number of proposed and approved changes (additions, modifications, deletions) to them per week/per month.

CHAPTER 6

TESTING AND ANALYSIS

6.1. TESTING AND ANALYSIS OF THE MODEL

To assess the applicability of a RE evaluation model, the users basically perform a mapping from the actions present in the model to the activities in a real process using the checklist. The checklist is actually a direct development of the model into question form. A snapshot of the checklist is shown in table 5.1. The whole checklist can be prepared in this way and evaluate by users and experts in different organizations.

The checklist follows the same structure as the model with questions grouped according to the MPA and SPA. For each action in the model, there is a corresponding question or group of questions to verify if the action is done or not. The Action ID which links the question(s) to the associated action in the model helps the users in case they need to locate the item for further information or clarification. It is adopted from Empirical Evaluation of a Universal Requirements Engineering Process Maturity Model (Nguyen, 2010)

When answering the questions, the users may encounter one of the following situations:

- The action was deemed vital but was performed partially or not at all in this RE Process s. It should be marked as “Partially applicable” (PA)

- The action was completed in this RE process. It should be marked as “Applicable” (A) The action was not necessary or possible to be performed in this process. It should be marked as “Inapplicable” (IA)

Table 6-1: Checklist of Evaluation Model

Action Id	Question	PA	A	IA	Reason/comment if it is inapplicable
	Organizational Context				
OCKA	**Knowledge Area**				
OCKA1	Do you define tender document with the support of experts or consultant				
OCKA2	Do you have a product-wide glossary of terms to ensure that the key concepts in the domain are Properly understood by all parties?				
OCKA3	Do you educate user representative and manger about requirement engineering				

Based on the applicability of the model, it can be used to evaluate the major process areas in the organizational context, project management, practice gap and main area of requirement engineering process. This will help an organization to evaluate the current practice of RE with respect to organizational context, project management and practice gap. This will make the software engineering process more effective in the organizational development.

Testing the applicability of the model According to more than 45 experts in the area, those are project manager, system analyst, users, customers, academicians and all other stakeholders' response. The following average results of each main process area prepared.

Table 6-2: Testing Applicability of the Model

MPA	% Partially Applicable (PA)	% Applicable(A)	% Inapplicable (IA)
Organizational Context	5	93	2
Project Management	3	90	7
Practice Gap	10	81	9
Requirement Elicitation	11	79	10
Requirement Analysis	9	88	3
Requirement Specification	5	90	5
Requirement Management	13	75	12

As can be seen from the above table, the model can be used to evaluate the RE process and other related main process areas in the organization. From the result in the above table requirement management main process area is inapplicable, because they stated that it is highly depends on individual capability and experience in that process.

6.2. MODEL COMPARISONS

In order to identify the improvement of this model in the requirement engineering process I tried to compare with other model which is Empirical Evaluation of a Universal Requirements Engineering Process Maturity Model (Nguyen, 2010) in the software development organizations

In the new model organizational context, project management and practice gap are the new component that will not available in other the requirement process evaluation model. Some of new model component are described in the table 6.3 below.

Table 6-3: Model Comparison

Major New model Component that will not available in the existing model	Type of Improvement	Testing organization
Educate user representative and managers about requirements	Create a willingness filling and getting cooperation during requirement elicitation	A,B&D
Train Analyst in Requirement Development and Management Processes	Getting better skill about requirement elicitation methods and approaches that make the requirement engineer more efficient and filling the gaps	B&C
Prepared Organizational Standard (Coding or Documentation)	If the organization has standard then integration and testing process become simple. if new staff join the project it makes simple to understand and work on	A,C&D

	it	
Prepare Pair programming approach	Assign more than one person in the module parts this help to reduce the challenge when professional staff leave or sick during critical time the organization	A,B&C
Identify Individual and Group Skill and behavior	It help to assign the right person in the right place in the project d based on individual skill and behavior	B,C &D

Finally, when you apply the model in the organization, can get a better improvement of practice in the main process areas that are mentioned in the model. This makes the RE process efficient in the organizational development , in general, if the result of each MPA more than 70% your process in better status and CBS organizational development can be achieved successfully .

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1. CONCLUSION

In order to produce quality software you should give high emphasis and enough budgets for requirement engineering and critically working on requirement engineering because it is the foundation of every software project

Organizational development is very important issues now a days in order to improve the service and the product this will be supported by CBS ,the effectiveness of CBS in the organizational development measures how much fit with organizational development goal this will be achieved by proper implementation of RE and related process .

The requirement registration traceability approaches considers organization context in the software development process this will improve implementation of the software in the organizational development process.

The proposed approach of requirement engineering has a greater contribution for most software companies by creating awareness and creating industry standard on requirement development and management practice that improves software requirement engineering practice in. According to survey most software project delayed because of lack of emphasis and improper requirement engineering practice.

During the process of the research I face some limitations during the testing and as a result it also affects some of the test scenarios.

The approach was not test using entire software requirements because it is very large, time taking and some of the organization requirement needs secret.

7.2. RECOMMENDATION

Software organization must give high emphasis on requirement engineering and prepared a staff upgrading training on requirement process management in order to get better result in the requirement engineering phase because it is the foundation of any software project. Based on survey result many projects are delayed and budget and time over run from initially stated. In order produce high quality software with less cost and less time you must assign enough budgets and experienced staff for RE process and you should have an organizational standard this will make communication and integration become simple in large project, by assign two persons in one model you can reduce problems related to staff turnover in the project.

Every software development organization should be evaluate the requirement engineering process using this model and identify the gap from different perspective and try to improve the gap, this will reduce rework, error, time and resource

Finally for future work I proposed the following work needs additional investigation

- Requirement elicitation techniques and RE
- The effectiveness tools that r represents the requirement engineering process
- The relationship between software Maturity level of the organization and RE
- The relation between Requirement engineering with process model

REFERENCES

Biru Tesfaye Reflective Steps: A Collaborative Learning Oriented Approach to Software Development and Process Improvement. - [s.l.] : universty of hamburg, 2008.

Dr. Sohail Asghar Mahrukh Umar Requirement Engineering Challenges in Development of Software [Book]. - [s.l.] : International Journal of Software Engineering (IJSE), Volume (1): Issue (2), 2010.

Gunda Sai Ganesh. Requirements Engineering: Elicitation Techniques. - [s.l.] : University West, Department of Technology, Mathematics and Computer Science, artment of Technology, Mathematics and Computer Science, , 2008. - p. 40.

Gunda Sai Ganesh. Requirements Engineering: Elicitation Techniques [Journal]. - [s.l.] : University West, Department of Technology, Mathematics and Computer Science, 2008.

Isabel Ramos Daniel M. Berry and Joa˜o A´. Carvalho Requirements Engineering for Organizational Development . - [s.l.] : Elsevier Science, 2004.

Johnson R. and Foote B. Designing reusable classes [Journal] // Journal of Object-Oriented Programming. - June/July 1988. - 2 : Vol. 1. - pp. 22-35.

Jones C. Applied Software Measurement: Assuring Productivity and Quality. - [s.l.] : McGraw-Hill, 1996.

June Verner Karl Cox Requirements Engineering and Software Project Success [Journal]. - [s.l.] : Australasian Journal of Information Systems, 2005.

June Verner Karl Cox,Steven Bleisteinand Narciso Cerpa Requirements Engineering and Software Project Success An Industrial Survey in Australia and the U.S // 2008.

Kotonya and Ian Somerville: “Requirements engineering process and techniques”,Bashar Nuseibeh, Steve Easterbrook: “Requirements engineering: A road map”, [Journal]. - [s.l.] : ACM, 2000.

Muhammad Waqas Boota Naseer Ahmad, Abdul Hye Masoom Requirement Engineering Issues and Their Solutions [Journal]. - [s.l.] : International Journal of Engineering and Technical Research (IJETR), 2014. - Issue-11 : Vols. ISSN: 2321-0869, Volume-2, .

Muneera Bano Didar Zowghi Users' Involvement in Requirements Engineering and [Journal] // IEEE. - 2013.

Muneera Bano Didar Zowghi USERS' INVOLVEMENT IN REQUIREMENTS ENGINEERING AND SYSTEM SUCCESS. - [s.l.] : University of Technology, Sydney, Australia, 2013.

Mythili Sathiya and K. Implementation of Five Key Process Areas to Improve the Requirement Engineering Process [Journal]. - [s.l.] : International Journal of Computer Applications (0975 – 8887), May 2014. - Vols. Volume 93 – No 5.

Nguyen Mai Empirical Evaluation of a Universal Requirements Engineering Process maturity model. - [s.l.] : Blekinge Institute of Technology , 2010.

Pino Robinson E. [Book Section] // Network Science and Cybersecurity. - [s.l.] : Springer Science and Business Media, 2013.

Pressman [Journal]. - 1997.

Requirements engineeringas a success factor in software projects. [Book]. - [s.l.] : IEEE Software,18(4):58–66,, 2001.

S. Kujala 'User involvement: a review of the benefits and [Journal]. - 2003.

Samuel A. Fricker Rainer Grau and Adrian Zwingli Requirements Engineering: Best Practice [Journal] // Springer International Publishing Switzerland . - 2015. - p. 15.

Samuel Fricker Rainer Grau and Adrian Zwingli Requirements Engineering: Best Practice [Journal]. - [s.l.] : Springer, 2015.

Savolainen T. [et al.] Positioning of Modeling Approaches, Methods and Tools [Book Section] // Computers in Industry. - 1995. - Vol. 25.

Shadab Khan[1] Aruna B Dulloo[2] and Meghna Verma[3] Systematic Review of Requirement Elicitation Techniques [Journal]. - [s.l.] : International Research Publications House , 2010.

Sommerville Kotonya and [Journal]. - 1998.

Wang'ang'a George Njehia Effective Requirements Managemen // Effective Requirements Managemen. - [s.l.] : Blekinge Institute of Technology, 2002.

Wiegers Karl E. Testing the Requirements. - [s.l.] : Microsoft Press, , 2006.

APPENDICES

APPENDIX – 1: SURVEY QUESTIONNAIRES

Adopt the questioner from "Stuart Anderson and Massimo Felici" and "Tesfaye Biru 2008"

General Information

Name (Optional)	
Position in the organization	
Field of Study	
Year of Experience	
Org .Name/Section	
No of project participated	

Part I: General

1. **Tick the popular methods which are used in your companies for software process models (multiple selections allowed)?**

☐ 1.1. Water fall
☐ 1.2. Spiral
☐ 1.3. Evolutionary Prototyping
☐ 1.4. RAD
☐ 1.5. Agile
☐ 1.6. Incremental /Iterative
☐ 1.7. Unified

2. **Do you have seen the difficulty of defining a clear scope of the project during the life time of the project**

☐ 2.1. Yes ☐ 2.2. No

3. **Based on plan which of the following is/are mostly achieved in your organization**

☐ 3.1. On time
☐ 3.2. on Budget
☐ 3.3. Work as needed
☐ 3.4. User satisfaction

4. Do you have seen an efficient support from management

☐ 4.1. Yes ☐ 4.2. No

5. During the submission of the project ,Do you have seen a request for large Modification /a new features /functionality in the project

☐ 5.1. Yes

☐ 5.2. No

6. Which type of change have you seen in the organization and impact on the project

☐ 6.1. Structural

☐ 6.2. Business process

☐ 6.3. Polices and standard

☐ 6.4. No change

☐ 6.5. Other, please specify_____

7. Most customers or users raised what type of requirement(s) are not fulfill during the submission of the project

☐ 7.1. Business level requirement

☐ 7.2. User level requirement

☐ 7.3. Product level requirement Other_____

8. Which of the following guidelines are available and in use at your workplace?

☐ 8.1. Planning guideline

☐ 8.2. Change management guidelines

☐ 8.3. Requirement documentation guidelines

☐ 8.4. Design documentation guideline

☐ 8.5. Coding guideline

☐ 8.6. No guideline

9. From your experience, which aspects of software development are challenging?

☐ 9.1. Planning

☐ 9.2. Requirement gathering

☐ 9.3. Analysis and design

☐ 9.4. Coordination with the team

☐ 9.5. Communication with users

☐ 9.6. Use of tools

☐ 9.7. Coding

☐ 9.8. Testing

10. Based on change request on the requirement ,which of the following mostly affected

☐ 10.1. Cost plan of the project

☐ 10.2. Schedule plan of the project

☐ 10.3. Developers effort , by requesting reworking or modification of the task

☐ 10.4. other, please specify _____

11. which type of training prepared by the organization

☐ 11.1. Project Domain and business process related

☐ 11.2. Development tools related

☐ 11.3. Organizational standards and guideline related

☐ 11.4. No training

☐ 11.5. If any other ,please specify_____

12. For completed projects, how do you compare the alterations/changes in the requirement specified in the original tender document and actual implementation?

☐ 12.1. Almost the same

☐ 12.2. Similar but with minor variation

☐ 12.3. Major variation

☐ 12.4. Totally different

13. If your response to question 18 is "major variation" or "totally different", what do you think are the major reasons?

☐ 13.1. Inability of the user to adequately articulate requirements at the beginning

☐ 13.2. Lack of requirement engineering skills of developers

☐ 13.3. Inadequate planning during the tendering

☐ 13.4. Changes in the organization since the commencement of the project

☐ 13.5. Other, please specify:_____

14. **What are the deliverables usually expected of projects (in addition to the software)?**

☐ 14.1. Inception report

☐ 14.2. Project management plan

☐ 14.3. Requirement analysis document

☐ 14.4. Design document

☐ 14.5. Test plans and results

☐ 14.6. Project progress report

☐ 14.7. Operation and training manual

☐ 14.8. Other, please specify _____

15. How do you handle changes in projects

☐ 15.1. There is a standard format to initiate, discuss and approve change

☐ 15.2. The change process is communicated to all stakeholders timely

☐ 15.3. Only management participates in the decision process related to change

☐ 15.4. Both management and users participate in the decision process related to change

☐ 15.5. The development team also participates in the decision process

☐ 15.6. On the basis of approved changes, the project plan and contract are formally revised and properly documented

☐ 15.7. Additional resources required are provided immediately

16. In relation projects already completed (please fill in the following table for at least three most current or recent project)

Project Name	Planned		Actual		Select Reason for delay and incomplete :write the letter(s)
	Start	Finished	Start	Finished/ Expected to Finished	

A 16.1. Poor planning because of unclear / incomplete requirements

B 16.2. Poor planning because of lack of project management skill

C 16.3. Continuous Requirement change request because of lack of user involvement

D 16.4. Continuous Requirement change request because of Poor requirement engineering process

E 16.5. Lack of proper progress monitoring and control

F 16.6. High project staff turnover

G 16.7. Lack of use of appropriate software development methodology

H 16.8. Inadequate budget to finance additional work due to changes in User requirements

I 16.9. Lack of cooperation from users or stakeholders

J 16.10. Problems within the development team

K 16.11. Organizational instability or business process change

17. Which of the following SDLC has got high emphasis in the organization

☐ 17.1. Requirement engineering
☐ 17.2. Design and architecture
☐ 17.3. Coding
☐ 17.4. Maintenance and testing

Second part: This part tries to assess Requirement Engineering Activities

Part I: Requirement elicitation

18. Which of the following activities implemented in your organization

☐ 18.1. Feasibility study before starting a new project?
☐ 18.2. Reuse requirement
☐ 18.3. Identify use cases
☐ 18.4. Identify user classes
☐ 18.5. List out organizational policies and standards

19. When user or stakeholder involve during life time of the project

☐ 19.1. At the beginning of the project
☐ 19.2. Whenever necessary
☐ 19.3. At the end of the project during acceptance testing
☐ 19.4. Only the time of requirement elicitation
☐ 19.5. Scheduled based
☐ 19.6. Other, please specify _____

20. Which requirement techniques is/are used

☐ 20.1. Interview
☐ 20.2. Questioner
☐ 20.3. Workshop
☐ 20.4. Prototyping
☐ 20.5. Requirements reuse
☐ 20.6. Scenarios
☐ 20.7. Brainstorming
☐ 20.8. Document Analysis
☐ 20.9. User Centered Design

☐ 20.10. Others ,please specify _____

Part II .Requirement Analysis, Negotiations and validation

21. Which of the following requirement analysis practice is available in your organization

☐ 21.1. Define system boundaries
☐ 21.2. use checklists for requirements analysis
☐ 21.3. use checklists for requirements analysis
☐ 21.4. Prioritize requirements
☐ 21.5. Use interaction matrices to find conflicts and overlaps
☐ 21.6. Perform any risk analysis on requirements?
☐ 21.7. classify requirements
☐ 21.8. Allocate requirements to subsystems
☐ 21.9. Model the requirements
21.10. check that requirements document meets your standards

22. which of the analysis or modeling techniques are used in your organization

☐ 22.1. Domain-driven development
☐ 22.2. OOA (Object Oriented Analysis)
☐ 22.3. Prototyping
☐ 22.4. SA(Structured Analysis)
☐ 22.5. Informal Modeling
☐ 22.6. Others, please specify_____

23. Which of the following requirement validation practice is available

☐ 23.1. check that requirements document meets your standards
☐ 23.2. organize formal requirements inspections
☐ 23.3. use multi-disciplinary teams to review requirements
☐ 23.4. define validation checklists
☐ 23.5. demonstrate requirements for validation

Part II: Requirement Specification and Management

24. which requirement specification practice is available

☐ 24.1. have standards templates / documents for describing requirements
☐ 24.2. have guidelines how to write requirements
☐ 24.3. produce a summary of the requirements
☐ 24.4. The requirements document easy to change
☐ 24.5. use diagrams appropriately
☐ 24.6. Record business rules
☐ 24.7. Specify quality attributes
☐ 24.8. uniquely identify each requirement

☐ 24.9. specify requirements quantitatively

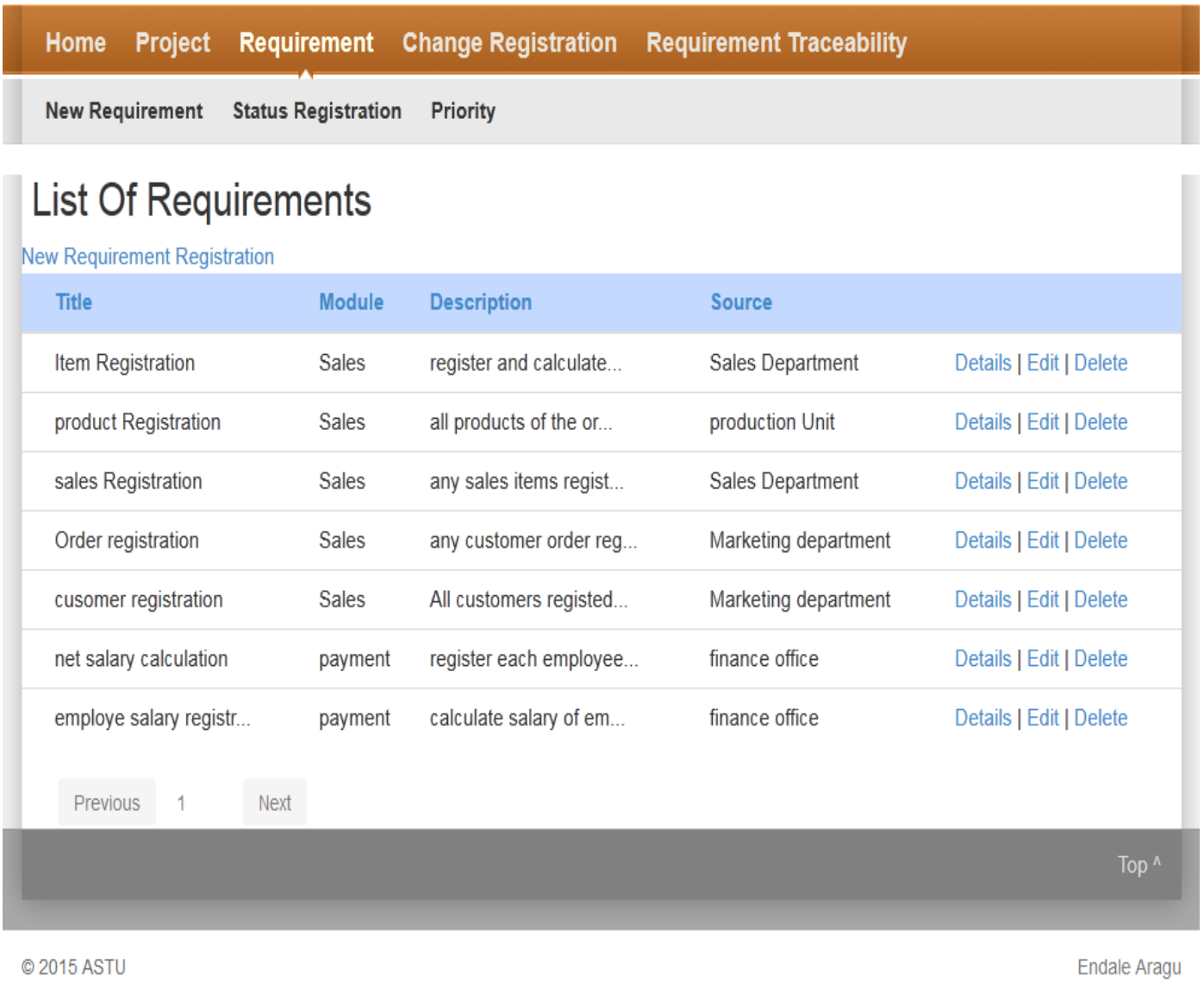
25. which requirement specification techniques are used

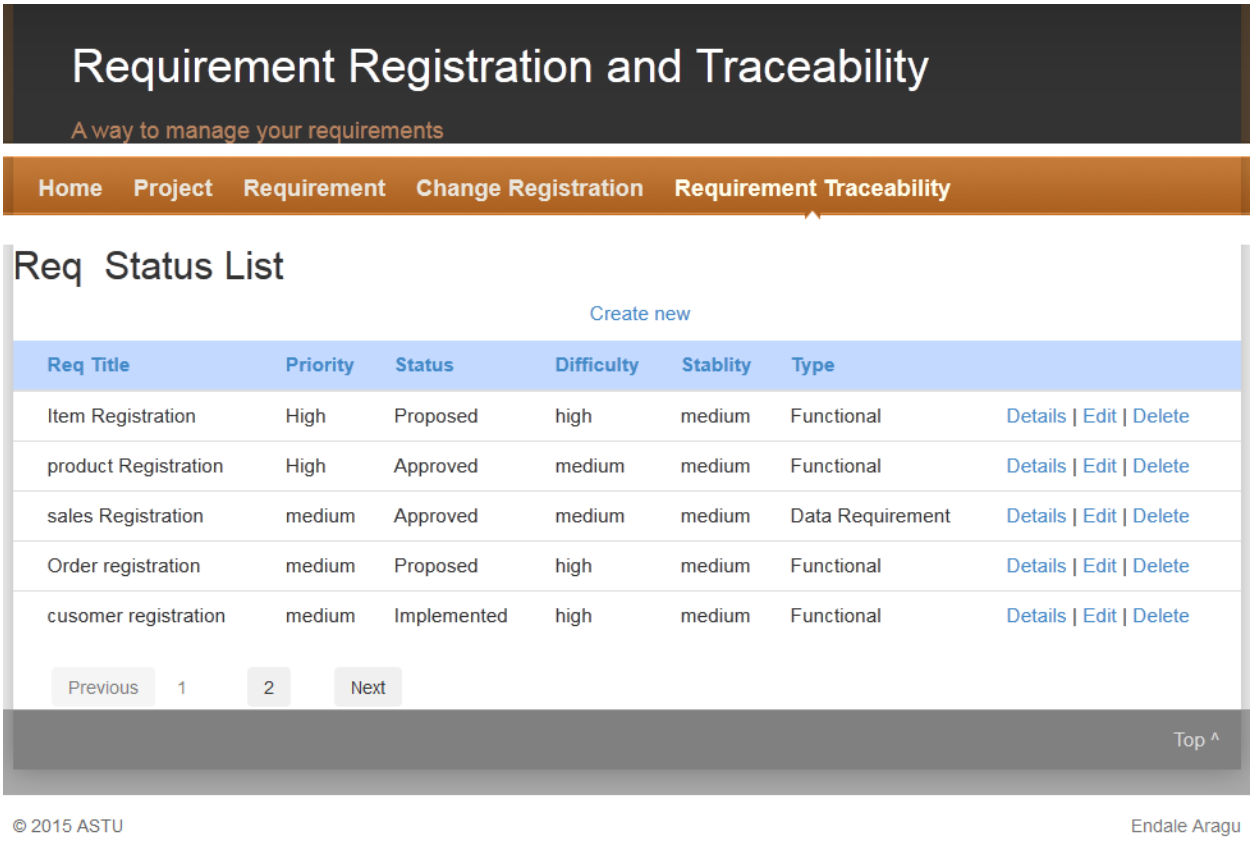
☐ 25.1. Natural language
☐ 25.2. UML diagrams
☐ 25.3. i* or KAOS
☐ 25.4. Structured analysis diagram
☐ 25.5. Tables
☐ 25.6. User screens

26. Which Requirement management practice available

☐ 26.1. have defined policies for requirements management
☐ 26.2. record requirements traceability from original sources
☐ 26.3. maintain traceability manual
☐ 26.4. Identify and measure volatile requirements
☐ 26.5. Baseline and control versions of requirements
☐ 26.6. Identified the areas of greatest expected change
☐ 26.7. reuse requirements over different projects
☐ 26.8. Perform change impact analysis
☐ 26.9. Maintain change history
☐ 26.10. Use a database to manage requirements

Appendix -2 Sample Code






```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.Entity;
using ThesisImplimentation;

namespace ThesisImplimentation.ReqProjects
{
    public partial class Default : System.Web.UI.Page
    {
          protected ThesisImplimentation.ReqTraceabilityEntities
_db = new ThesisImplimentation.ReqTraceabilityEntities();

        protected void Page_Load(object sender, EventArgs e)
        {
        }
```

```
        public IQueryable<ThesisImplimentation.ReqProject>
GetData()
        {
            return _db.ReqProjects;
        }
    }
     }
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.Entity;
using ThesisImplimentation;

namespace ThesisImplimentation.ReqRegs
{
    public partial class Default : System.Web.UI.Page
    {
          protected ThesisImplimentation.ReqTraceabilityEntities
_db = new ThesisImplimentation.ReqTraceabilityEntities();

        protected void Page_Load(object sender, EventArgs e)
        {
        }


        public IQueryable<ThesisImplimentation.ReqReg> GetData()
        {
            return _db.ReqRegs.Include(m =>
m.ReqModule).Include(m => m.ReqStatu);
        }
    }
     }
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.Entity;
using ThesisImplimentation;

namespace ThesisImplimentation.ReqStatus
{
    public partial class Default : System.Web.UI.Page
    {
          protected ThesisImplimentation.ReqTraceabilityEntities
_db = new ThesisImplimentation.ReqTraceabilityEntities();

        protected void Page_Load(object sender, EventArgs e)
        {
        }


        public IQueryable<ThesisImplimentation.ReqStatu>
GetData()
        {
            return _db.ReqStatus.Include(m =>
m.Difficulty).Include(m => m.ReqPriority).Include(m =>
m.ReqReg).Include(m => m.ReqRisk).Include(m =>
m.RStatu).Include(m => m.Stability).Include(m => m.ReqType);
        }
    }
     }
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.Entity;
using ThesisImplimentation;

namespace ThesisImplimentation.ReqModules
{
    public partial class Default : System.Web.UI.Page
    {
          protected ThesisImplimentation.ReqTraceabilityEntities
_db = new ThesisImplimentation.ReqTraceabilityEntities();

        protected void Page_Load(object sender, EventArgs e)
        {
        }

        // Model binding method to get List of ReqModule entries
        // USAGE: <asp:ListView SelectMethod="GetData">
        public IQueryable<ThesisImplimentation.ReqModule>
GetData()
        {
            return _db.ReqModules.Include(m => m.ReqDesign);
        }
    }
     }
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.Entity;
using ThesisImplimentation;

namespace ThesisImplimentation.ReqModules
{
    public partial class Default : System.Web.UI.Page
    {
          protected ThesisImplimentation.ReqTraceabilityEntities
_db = new ThesisImplimentation.ReqTraceabilityEntities();

        protected void Page_Load(object sender, EventArgs e)
        {
        }

        // Model binding method to get List of ReqModule entries
        // USAGE: <asp:ListView SelectMethod="GetData">
        public IQueryable<ThesisImplimentation.ReqModule>
GetData()
        {
            return _db.ReqModules.Include(m => m.ReqDesign);
        }
    }
}
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.Entity;
using ThesisImplimentation;

namespace ThesisImplimentation.ReqDesigns
{
    public partial class Default : System.Web.UI.Page
    {
          protected ThesisImplimentation.ReqTraceabilityEntities
_db = new ThesisImplimentation.ReqTraceabilityEntities();

        protected void Page_Load(object sender, EventArgs e)
        {
        }
        public IQueryable<ThesisImplimentation.ReqDesign>
GetData()
        {
            return _db.ReqDesigns.Include(m => m.ReqProject);
        }
    }
public partial class Insert : System.Web.UI.Page
    {
          protected ThesisImplimentation.ReqTraceabilityEntities
_db = new ThesisImplimentation.ReqTraceabilityEntities();

        protected void Page_Load(object sender, EventArgs e)
        {

        }


        public void InsertItem()
        {
            using (_db)
            {
                var item = new ThesisImplimentation.ReqDesign();
```

```
                TryUpdateModel(item);

                if (ModelState.IsValid)
                {

                    _db.ReqDesigns.Add(item);
                    _db.SaveChanges();

                    Response.Redirect("Default");
                }
            }
        }

        protected void ItemCommand(object sender,
FormViewCommandEventArgs e)
        {
            if (e.CommandName.Equals("Cancel",
StringComparison.OrdinalIgnoreCase))
            {
                Response.Redirect("Default");
            }
        }
    }
public partial class Edit : System.Web.UI.Page
    {
          protected ThesisImplimentation.ReqTraceabilityEntities
_db = new ThesisImplimentation.ReqTraceabilityEntities();

        protected void Page_Load(object sender, EventArgs e)
        {
        }


        public void UpdateItem(int  DesignID)
        {
            using (_db)
            {
                var item = _db.ReqDesigns.Find(DesignID);

                if (item == null)
                {
                    // The item wasn't found
```

```
                    ModelState.AddModelError("",
String.Format("Item with id {0} was not found", DesignID));
                    return;
                }

                TryUpdateModel(item);

                if (ModelState.IsValid)
                {
                    // Save changes here
                    _db.SaveChanges();
                    Response.Redirect("../Default");
                }
            }
        }

       public ThesisImplimentation.ReqDesign
GetItem([FriendlyUrlSegmentsAttribute(0)]int? DesignID)
        {
            if (DesignID == null)
            {
                return null;
            }

            using (_db)
            {
                return _db.ReqDesigns.Find(DesignID);
            }
        }

        protected void ItemCommand(object sender,
FormViewCommandEventArgs e)
        {
            if (e.CommandName.Equals("Cancel",
StringComparison.OrdinalIgnoreCase))
            {
                Response.Redirect("../Default");
            }
        }
    }
        protected void Page_Load(object sender, EventArgs e)
        {
        }
```

```
        public void DeleteItem(int DesignID)
        {
            using (_db)
            {
                var item = _db.ReqDesigns.Find(DesignID);

                if (item != null)
                {
                    _db.ReqDesigns.Remove(item);
                    _db.SaveChanges();
                }
            }
            Response.Redirect("../Default");
        }

        public ThesisImplimentation.ReqDesign
GetItem([FriendlyUrlSegmentsAttribute(0)]int? DesignID)
        {
            if (DesignID == null)
            {
                return null;
            }

            using (_db)
            {
                 return _db.ReqDesigns.Where(m => m.DesignID ==
DesignID).Include(m => m.ReqProject).FirstOrDefault();
            }
        }

        protected void ItemCommand(object sender,
FormViewCommandEventArgs e)
        {
            if (e.CommandName.Equals("Cancel",
StringComparison.OrdinalIgnoreCase))
            {
                Response.Redirect("../Default");
            }
        }
    }
}}
```