Global Journal of Engineering Science and Research Management

SW REQUIREMENT ENGINEERING IN PRACTICE

Smita Raj*
* C-204, Shiksha Niketan, Vasundhara, Sec-5, Ghaziabad 201012

DOI: 10.5281/zenodo.199474

KEYWORDS: Requirement, Requirement engineering, process models, Requirement Elicitation, Requirement Management, Requirement Validation.

ABSTRACT

Requirements engineering is the engineering discipline of establishing user requirements and specifying software systems. It refers to the process of formulating, documenting and maintaining the SW requirements and to the subpart of SW Engineering pertaining to this process.

Requirement engineering continues through the lifetime of the system. Requirement engineering is very critical to the success of software project or product. Requirement should be capable of being measured, testable, traceable, action oriented w.r.t. the business need and opportunities. Besides, Requirement should be properly furnished with details which are sufficient for system design. To derive the requirements user need to have clear and thorough understanding of the product(s) to be developed. Preparation of requirements can be done after detail discussion with team members and end user.

INTRODUCTION

A system's users seldom know exactly what they want and cannot articulate everything they know. Even if requirements are stated, many details can only be discovered during implementation. Requirement engineering provides the appropriate mechanism for understanding what the customer wants, analyzing need, assessing feasibility, negotiating a reasonable solution, specifying the solution unambiguously, validating the specification and managing the requirements as they are transformed into an operational system.

Misunderstanding the requirement leads to SW developers solving the wrong problem or attempting an infeasible problem which can turn into chaotic development process.

The processes used for RE vary widely depending on the application domain, the people involved and the organisation developing the requirements.Requirement engineering and management has a very vital effect on development cost and quality of software.

Requirement analysis method should be incremental, systematic, customer oriented and efficient enough to improve the schedule, budget and product quality.

There are numerous generic activities applicable to most of the process as follows:

1. Requirements elicitation
2. Requirements analysis and negotiation
3. Requirements validation
4. Requirements management

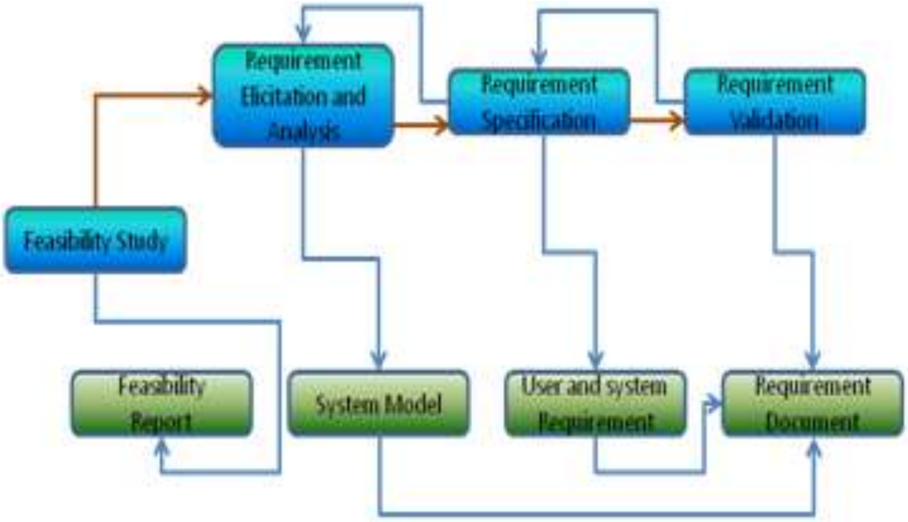

Figure 1: Requirement engineering Kickoff

With each new system, requirement engineering starts with feasibility study which decided whether the system is worthwhile or not. The study contains alignment of the system with organizational objectives, be engineered using current technology and within budget and integration with other systems that are used.

Requirements Engineering Process Modeling

Requirements engineering [3] is an iterative process, often the purpose of going through the exercise is to define the problem itself, and in the process, refine requirements. That is, an initially fuzzy requirement gets clarified through discussion, leading to other, more specific requirements. While the model has been motivated by requirements engineering, its components are generic enough to be applicable across various phases of the life cycle.

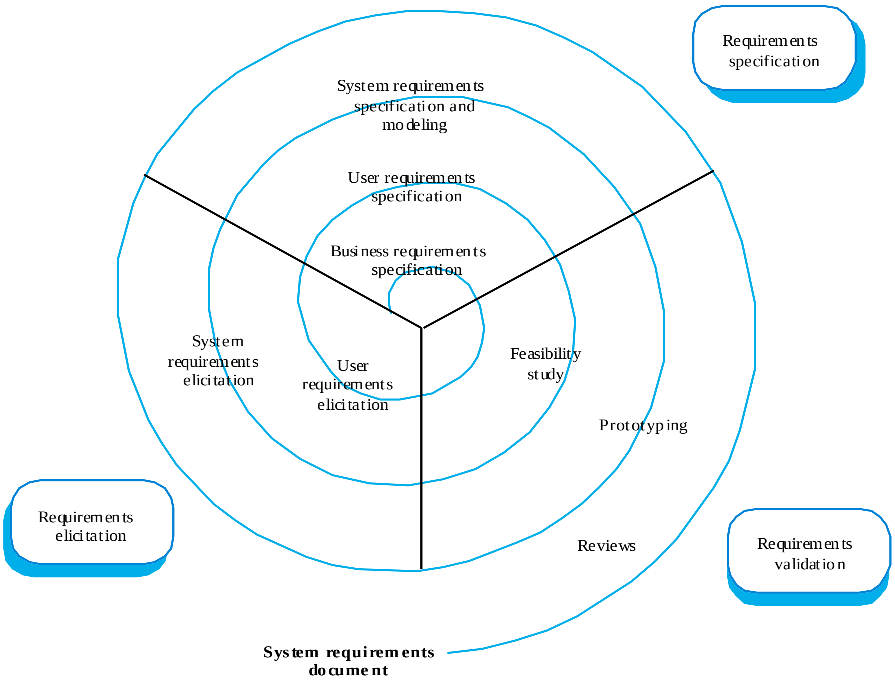
Requirements specification
System requirements specification and modeling
User requirements specification
Business requirements specification
System requirements elicitation
User requirements elicitation
Feasibility study
Prototyping
Reviews
Requirements elicitation
Requirements validation
System requirements document

Figure 2: Requirement Engineering [5] Spiral Model

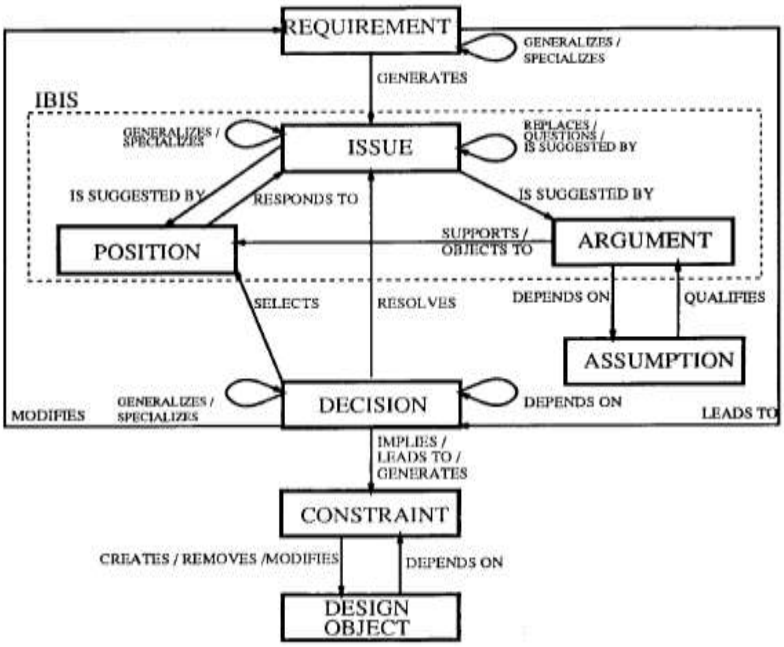

Figure 3: Requirement Engineering Conceptual Model

Requirement Elicitation

A comprehensive and complete discovery of any business and system requirement is very rarely available at the analyst's tips. Most of business or technical requirements is not documented anywhere. It rests in the minds of stakeholders, in feedback that has yet to be obtained from end users and from a study of flowcharts, surveys that have yet to be created. And so requirements must be elicited [2], or drawn out, and the methodology in doing so, must be logical and meticulous. The importance of elicitation cannot be overstated, for it is the linchpin to any requirements project. As one scholarly article notes: “Mistakes made in elicitation have been shown many times to be major causes of systems failure or abandonment and this has a very large cost either in the complete loss or the expense of fixing mistakes.” Adequate study and preparation for elicitation can go a long way to preventing these types of errors. The purpose of requirements elicitation, therefore, is to thoroughly identify the business needs, risks, and assumptions associated with any given project.

Goal of Requirement Elicitation

The sole purpose of requirement elicitation is about seeking information about the software, the system, and the business. The following information can be sought from the customers and the users:

- What are the objectives of the software product?
- What is to be accomplished by the software?
- How does the software fit into the needs of the business?
- How is the software to be used on a day-to-day basis?

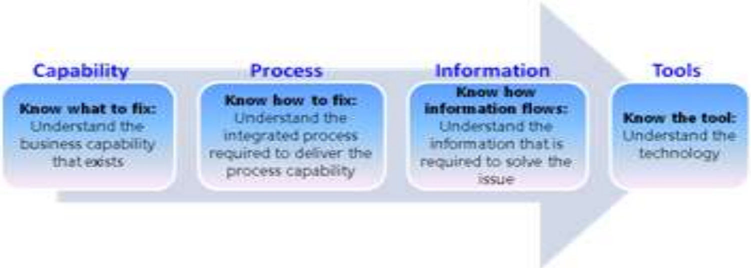
Capability
Know what to fix:
Understand the business capability that exists
Process
Know how to fix:
Understand the integrated process required to deliver the process capability
Information
Know how information flows:
Understand the information that is required to solve the issue
Tools
Know the tool:
Understand the technology

Figure 4: Goal of Requirement Elicitation

Requirement Elicitation process flow

The process starts with the domain understanding which is required to details the clear understanding about the requirement. It facilitates to collect and validate the customer requirement. Sometimes requirement validation and collection takes multiple round of discussion with end user. Once requirement collection is over, it goes to the next level of classification. The requirement classification can be functional, non-functional, user interface and scalability etc. On the other side, after requirement validation, prioritization of the requirement should be done.

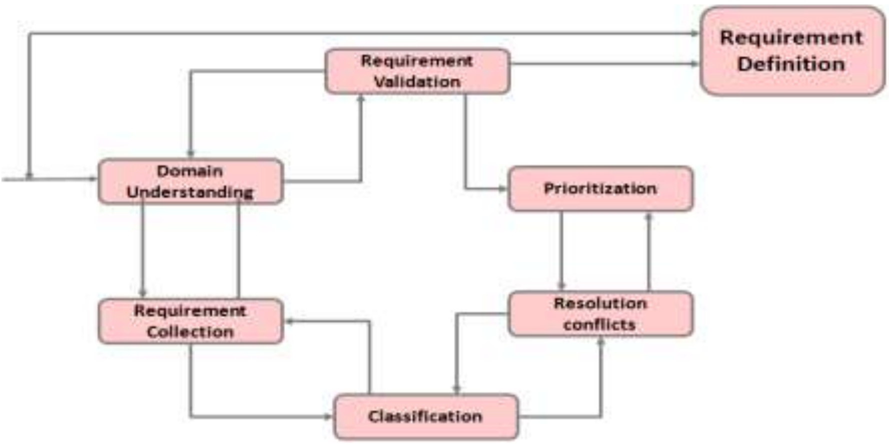

Figure 5: Requirement Elicitation Process Flow

Many time requirement conflicts occur due to the non-clarity of requirement classification and un-managed prioritization. To address this situation continuous discussion with customer or end user should be done regularly.

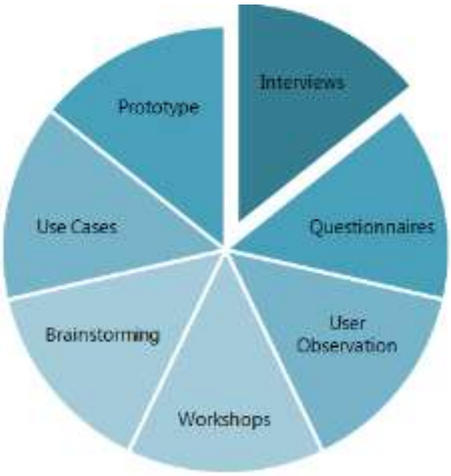

Figure 6: Requirement Elicitation Activities

It is non-trivial because that can never be sure to get all requirements from the user and customer by just asking them what the system should do. Before requirements can be analyzed, modeled, or specified they must be gathered through an elicitation [6] process. Commonly used elicitation processes are the stakeholder meetings or interviews.

Requirement Elicitation stages

- Organization Objective setting
- Background knowledge acquisition
- Knowledge organization
- Stakeholder knowledge collection.

REQUIREMENTS ANALYSIS AND NEGOTIATION

What is Requirement Analysis?

Requirement analysis is a software engineering task that bridges the gap between system engineering and software design. The needs of the end users are collected and analyzed for their validity with possibility of implementation. The requirements being gathered are split up into various categories like functional, nonfunctional and User Interface requirements. The aim of the task is to generate a Requirements Specification Document that can be used in the next phases of the development model.

How is SW Requirement Analysis done?

The SW requirement analysis task initiates with the study of the high level needs as received. This information in plain text is researched upon to get to the details of the user requirements. Existing similar implementations, at times, are referred. There might be needs for user interviews or questionnaires to decipher and elicit the actual user needs Any possible assumptions made, dependencies identified and constraints (related to both software and hardware) are analyzed and elicited.

The possible use cases of the system and the flow of each of the identified requirement is done. Use of UML notations is a common practice to carry out this step. For each of the identified functional requirement, the use cases are generated to elicit the:

- Pre-conditions
- Basic and alternative flows
- Failure modes and their error handling strategy
- Post conditions and qualification provisions
- Relationship and dependency on other requirements.

The non-functional requirements are also elicited by thinking on the user's goals, values and concern points, any existing constraints, industry and market trends and/ or competitor's system analysis.

SW Requirement Analysis process flow

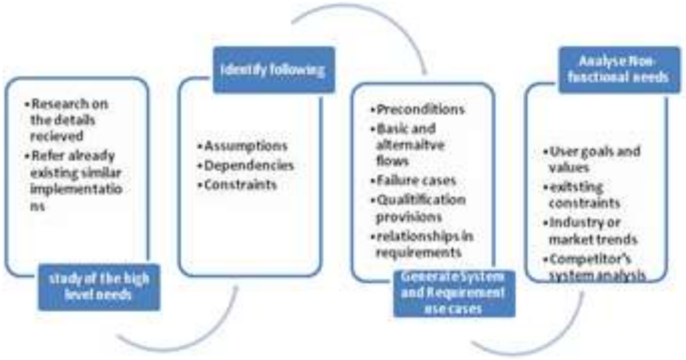
•Research on the details recieved
•Refer already existing similar implementatio ns
study of the high level needs
Identify following
•Assumptions
•Dependencies
•Constraints
•Preconditions
•Basic and alternaitve flows
•Failure cases
•Qualitification provisions
•relationships in requirements
Generate System and Requirement use cases
Analyse Non-functional needs
•User goals and values
•exitsting constraints
•Industry or market trends
•Competitor's system analysis

Figure 7: Requirement Analysis

Since the intent of the SW requirement analysis activity is to elicit the needs of the user, get them reviewed and approved by the stakeholders, we need to ensure that the specifications are written in such a way that they can be easily read and reviewed.

SW Requirement Negotiation

As the SW requirements are analyzed, it leads to the identification of any missed user needs, possible inconsistencies, constraints and/or conflicts in the desired results. Any, of these could often happen when there are multiple stakeholders (like users, admin, maintainers, customers etc.) of the intended system. This would lead to further analysis of these requirements to either gather more information or to refine the existing but incomplete information. Collaborative software analysis by involvement of the various stakeholders is the way forward for smooth negotiations on the software requirement analysis.

REQUIREMENT VALIDATION

The objective of requirement validation [7] is to certify that the requirements document is acceptable in terms of completeness, consistency, conformance to standard, requirement conflicts, technical error and ambiguous requirement. It denotes checking whether input, performed activities and created output (requirement artifact) of requirement engineering core activity fulfill defined quality criteria.

The difference between requirement analysis and requirement validation is that analysis works with raw requirements as elicited from the system stakeholder whereas validation works with a final draft of the requirements document i.e. with negotiated and agreed requirements.

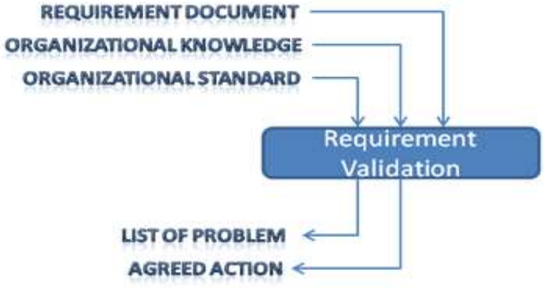

Figure 8: Requirement Validation

Requirement document should be complete and baseline version of the document, not an unfurnished draft. It should be formatted and organized in-line with organizational standards.

Developer should have knowledge of organization which may be used to judge the realism of the requirement. Organizational standards should be clearly defined and written so that developer can refer the same, to organize the requirements document.

Often, manual validation techniques, which are also known by the general term review, are used for requirements validation. Three major types of reviews can be differentiated:

- Commenting
- Inspections
- Walk-through

Along with reviews, the following three techniques have proven themselves to be useful for requirements validation:

- Perspective-based reading
- Validation through prototypes
- Using checklists for validation

Prior to applying any of these techniques, preparatory steps need to be taken as needed, such as identifying and inviting the right stakeholders or organizing suitable rooms and supplies.

REQUIREMENT MANAGEMENT

The purpose of requirements management [4] is to ensure that an organized documents, verifies and meets the needs and expectations of its customers and internal or external stakeholders. Requirements management begins with the analysis and elicitation of the objectives and constraints of the organization. Requirements management further includes supporting planning for requirements, integrating requirements and the organization for working with them (attributes for requirements), as well as relationships with other information delivering against requirements, and changes for these.

The traceability thus established is used in managing requirements to report back fulfillment of company and stakeholder interests, in terms of compliance, completeness, coverage and consistency. Traceability also support change management as part of requirements management in understanding the impacts of changes through requirements or other related elements (e.g., functional impacts through relations to functional architecture), and facilitating introducing these changes.

Requirements management [8] involves communication between the project team members and stakeholders, and adjustment to requirements changes throughout the course of the project. To prevent one class of requirements from overriding another, constant communication among members of the development team is critical. For example, in software development for internal applications, the business has such strong needs that it may ignore user requirements, or believe that in creating use cases, the user requirements are being taken care of.

Requirements management is concerned with:

- Understanding the goals of the organization and its customers
- Transformation of these goals into potential functions and constraints applicable to the development and evolution of products and services

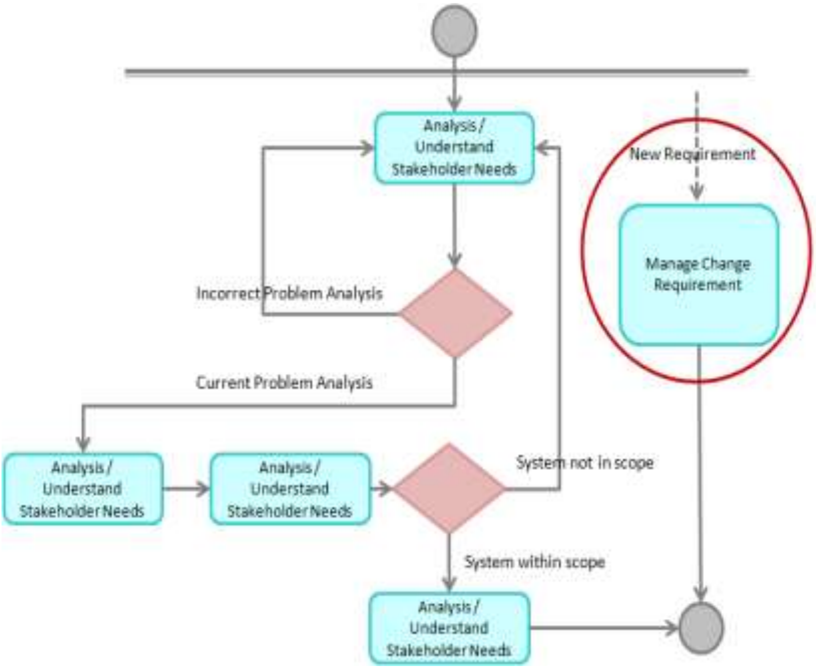
Analysis / Understand Stakeholder Needs
New Requirement
Manage Change Requirement
Incorrect Problem Analysis
Current Problem Analysis
Analysis / Understand Stakeholder Needs
Analysis / Understand Stakeholder Needs
System not in scope
System within scope
Analysis / Understand Stakeholder Needs

Figure 9: Requirement Management

Requirements need to be managed because:

- Change is an inevitable part of a project and in fact desirable at times
- By organizing and tracking your requirements the team improves its chances of completing the project:
  - on time
  - within budget
  - meeting client expectation
- Changing requirements and failure to coordinate these changes to stakeholders is the most common reason for project failures.
- It ensures that the developed product is in conformance with the requirements that have been outlined by the customer.
- It ensures that the product is delivered on time and within budget.
- Managing requirements helps in establishing and maintaining an agreement between the customer and the project team on the changing requirements of a system.

Below table represents the requirement management activities.

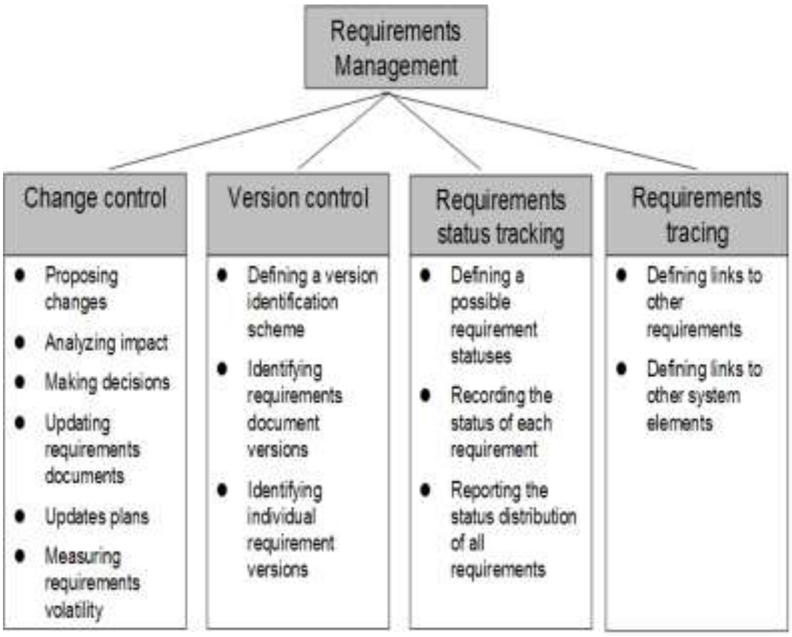
Requirements Management
Change control
Proposing changes
Analyzing impact
Making decisions
Updating requirements documents
Updates plans
Measuring requirements volatility
Version control
Defining a version identification scheme
Identifying requirements document versions
Identifying individual requirement versions
Requirements status tracking
Defining a possible requirement statuses
Recording the status of each requirement
Reporting the status distribution of all requirements
Requirements tracing
Defining links to other requirements
Defining links to other system elements

Figure 10: Requirement Management Activities

The expectation of requirement management is to identify the individual requirement, traceability of highest level requirement to implementation, impact assessment on proposed changes, control process to current project information, change control and deployment of required tool support.

Benefits of requirement management:
A good requirements management allows to avoid many design mistakes and to reduce failures during the development process. The cost of fixing errors or changing functionalities increases exponentially with the development phases. In software development, studies showed that the repair cost of an error is about 10 times higher during the tests as during the requirements management It is thus worthwhile to ensure that non error goes on undiscovered and core need (of end user) remains uncovered past the requirements phase. It is also a fact that the main error causes come from bad or wrong requirements. Therefore an improved requirements management can lead subsequently to a reduction in the product failures and product development cost. It is one of the most important parts of a product development. If it is done with success, the benefits can be numerous and allow faster execution at reduced cost. Thus, although requirements management is a quite difficult task, Software projects should try to improve it as much as possible

CONCLUSION

Requirement engineering is often employed to optimize a given requirement with respect to throughput and reliability. This improves organizations throughput, reduces the number of errors and decreases overhead costs by delivering the error free product. It involves various methods to understand the requirement prior to implementation.

This activity tended to occur across multiple phases.
When Requirement Engineering was seen as a continual task throughout the project, the Requirement Engineering process model was more iterative. The Requirement Engineering activities tended to occur across multiple phases, making the process models appears iterative. When RE occurred at the start of the project or each project increment the process was more linear. When multiple prototypes were used, the process was highly iterative.
Therefore, the point at which the Requirement Engineering process occurs in the project appears to affect the nature of the Requirement Engineering process model.

FUTURE WORK

As mentioned earlier, software requirement engineering provides a basis for establishing the functionality of a system in the software life cycle. The benefits of establishing a "standard" against which the end user needs are validated can have a significant impact on minimizing problems. The future of software requirement engineering lies in the research on the techniques and tool advancements.

- Research is required to have unified techniques that fit the needs of all. The requirement engineering should be flexible enough to cater the technological advancements and the introduction of new business verticals. Considerable efforts should be made in requirement engineering to produce qualified products.
- Optimum utilization of various audio and video technologies for better end user interactions need to be done.
- Non-functional requirement signifies the quality attribute of a system in place. There are needs to identify better techniques to capture and analyze non-functional requirements.
- To optimize the requirement engineering phase, requirement model reuse should be the point of focus. This would reduce the efforts in developing requirement models from scratch.
- To have impactful requirement engineering phase, consideration should be put on social and technical skills of the requirement analysts.

REFERENCE

1. Prasad Rajagopal, Roger Lee1, Thomas Ahlswede,"A New Approach for Software Requirements Elicitation ", Proceedings of the Sixth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing and First ACIS International Workshop on Self-Assembling Wireless Networks (SNPD/SAWN'05)
2. A New approach to Requirements Elicitation Using Paper Prototype
www.sersc.org/journals/IJAST/vol28/2.pdf
3. en.wikipedia.org/wiki/Requirements_engineering
4. http://en.wikipedia.org/wiki/Requirements_management
5. Requirements engineering processes, tools technologies, & Methodologies
www.ijric.org/volumes/Vol2/6Vol2.pdf
6. Requirements Elicitation
http://www.modernanalyst.com/Resources/Articles/tabid/115/articleType/ArticleView/articleId/1427/An-Overview-of-Requirements-Elicitation.aspx
7. Requirements Validation
http://www.sie.arizona.edu/sysengr/publishedPapers/FamousFailures.pdf
8. Requirements Management
9. http://files.itslearning.com/data/ntnu/open/co33719/1424184.pdf?