ANALYSIS OF REQUIREMENT PHASE USING POST DATA PROCESSING SYSTEM

[1]Ashok Bakthavathsalam, [2]Dr. T. Purusothaman, [3]Dr. R. Ravichandran, [4]Dr. G. Vijaya
[1]Research scholar, Anna University, Chennai
[2]Associate Professor, Govt. College of Technology, Coimbatore
[3]Director, KG College of Arts and Science, Coimbatore
[4]Associate Professor, KGISL Institute of Technology, Coimbatore, India
ashokresearchphd01@gmail.com

Abstract: In software engineering, Requirement engineering (RE) is the first and essential phase in Software development life cycle (SOLC) model. The process of RE is to gather the specific features to meet the user's purpose. Further, the process of this phase is to identify the inadequate, ambiguous, incomplete requirements and then transforms into adequate, unambiguous, complete software that satisfy the user's expectation. These transformations can give the better understanding between the users and developers. And also avoid the misinterpretation about software between users and developers for improving the software quality. In this paper, new technique called Post data processing system for requirements engineering process to analysis better requirement and then develops software.

Keywords: *Requirement engineering, Software engineering, software development life cycle*

1. INTRODUCTION

The most important phase of Requirements engineering is to gather and analysis the requirements from various stakeholders before software can be built. This phase will help to develop the product of software depends on user requirement from the system features. The mutual understanding between the human and machines, to collect the requirements that will help to produce well two types of Software Requirements as follows: [1]. A functional requirements is explains that the system can do the specific action without any human constraints. Non-functional software requirement is to describe that performance characteristic of system such as accessibility, quality, efficiency, security, privacy etc [1]. Generally, Requirement engineering is a complex process system and acts as an important role in software development life cycle. [3, 4, 5], The RE is used to maintain, analyst and develop the features of requirements from different stakeholder that satisfy the client's expectation.

The process of requirement engineering is a complex system because of software product evolution that will satisfy the client's needs and the practitioners consider the various set of software product previously that will demand from the client's side. The systematic approach of Requirement engineering is to collect about the functional and nonfunctional requirement from various Stakeholders and then design well to develop the process of quality software for any software development, The overall scheme of Software development life cycle is covered by the activities of requirement engineering. Requirements management is to managing and controlling the requirement change during a requirement analysis process and also specifies the well-defined requirements [9, 10, 11]. The beginning phase in Software development life cycle is requirements engineering. Practically, RE process it to collect the accurate amount of requirements from various clients that would remain constant throughout the software development life cycle [14]. Thus, SDLC is an incremental process development that the process of requirements analysis can be performed in parallel with other phases of activities consists of design, implementation, testing and documentation. In RE phase, a group of activities and requirements are mainly collected for discovering, analyzing, verification, validating and maintaining the documents for a system.

There are two main set of activities in Requirements engineering as follows:

- Software requirement development phase: to discovering, analyzing, documenting,

verification and validating requirements from group of activities.

- Software requirement management phase: to documenting, tracing, managing and then controlling the change in requirement from different stakeholders.

Software Requirements verification is the process of developing software activities for produce right product to meets its technical specifications and non-technical specification. Software Requirements validation is the process of right product is to developed software activities that will meet its client requirement [7].

One of the challenging activities in software development life cycle is Requirement engineering phase compared with other development phase. The most important tool of Software Requirement engineering phase is to collecting the precise requirement from various clients for discovering, analyzing, developing and documenting the software requirements [8].

2. RELATED WORKS

Zave (2005) explained about Requirements engineering (RE) which is the part of software development life cycle that the goals for, function of and constraints on software systems. It is also concerned with the activities and their relationship. The features of RE process have two attractive reasons as follows: First, to motivate the software system development in real time goals that referred why or what should system do. Second, specification must be precise. The set of activities and specification requirement are provided for analyzing, designing and validation done for deliver the right product.

Laplante (2007) determined objectives, functions and constraints of system hardware and software can be considered by both systems engineering and software engineering disciplines. The activity of feasibility study can start with requirement process that will supervise the feasibility report in some software life cycle models. From the suggestion of feasibility study, the analysis of requirement can starts after the product should be improved. The feasibility studies can be preceded by requirement analysis because of that may leads to do higher capacity before the requirements are finalized.

Lemos, Giese, & Müller(2013) explored that software system is to become more adaptable, dependable, efficiency, recoverable, resilient and self-optimization based on some changes need to system operation, system requirements and environments from the increasing requirement complexity.

Kaur & Singh (2010) presented that the requirement engineers have to know well about software behavior thoroughly because there is still lack a mature science in software engineering A self-adaptive systems should be developed and deployed to well known about the software requirements successfully. It acts in real time entities that requirement engineers have a mature science to specify the required software behavior.

3. CATEGORY OF REQUIREMENTS

The general categorization of Requirements in many ways that meet its technical management as follows: First, the statement of client's requirements is to specify the system expectations in terms and measures of mission objectives, environment, constraints, effectiveness and suitability. Second, the top level function of functional requirements that describes the task identification is done and then action must be performed. Third, Non-functional requirements are to define the system behavior that can be used to judge the system operation. Fourth, Performance Requirements is to degree which a mission or function must be executed and also measured in terms of quantity, quality, coverage etc. Based on the system life cycle factors for success, performance requirements will be interactively developed among all identified functions in requirements analysis and their relationship to other requirements. Fifth, Design Requirements is to develop the requirements for products and execute the requirements for processes expressed in technical data packages and technical manuals. Sixth, Derived requirements are transformed from higher-level requirement. For example, the design requirements may produce the high speed range of requirement for low weight. Seventh, Allocated Requirements is to simplify a high-level requirement into number of lower-level requirements.

4. SOFTWARE REQUIREMENT ENGINEERING PROCESS

- Eliciting Requirements: The beginning step of requirement engineering process is called as requirement elicitation. The definition of elicitation is to collecting the requirements and asking the right question about requirements to avoid the suggestion. In requirement elicitation, gather the information for interpreted, analyzes, modeled and validated before the requirement engineers can be collected the sufficient amount of system requirements. The other activities in requirement engineering are mostly related to requirements elicitation. The design model is used to driven the techniques of requirements elicitation and vice versa.

- Modeling and Analyzing Requirements: Modeling is the basics activity of RE process. The analysis of modeling requirements is to develop the description of concepts. Most of the textbook in RE is to focus on the requirements of modeling and analysis techniques. During the process of RE, the entire range of product can be represented by modeling techniques. More in details, requirements elicitation tools are applied to different modeling techniques for collecting the appropriate information. The modeling approach of requirement engineering can be classified with an example and then develop the useful information by some analysis techniques from the model.
- Communication Requirements: The process of effective interaction between the various stakeholders about requirements. So, RE process is not only acts as identification and specifying requirements. The main role for requirements in documentation can be read, analyzed and validated. The purpose of requirements documentation is to define the languages such as formal, semi-formal and informal languages suggested .The capabilities of expressions and reasoning based on different natural languages are obtained.
- Agreeing Requirements: The problem of all stakeholder agreement can be maintained with their different goals based on requirements elicitation and modeling requirements. Providing the requirements for stakeholder must be accurate and validated that the developing requirement process and models. Clearly, the required conditions are not only provided for validating requirements, it is also applied to solving the problems among all stakeholders. The approach of SCR tool in formal method used to check automatically that the model is consistent and complete. Other techniques can be solved the real world problems based on prototyping method and use of scenarios. The two main reasons for requirements validation to be difficult. First, philosophical in nature and consider the fact question and what is predictable. Second, obtaining agreement among stakeholder with their different goals is too difficult.
- Evolving Requirements: The developed (or changed) in both system operation and stakeholder requirements can give successful environment in software systems. Thus, one of the basis activities in RE process is change management. Because of changes in requirements documentation need to be managed. Providing various techniques and tools are used to monitor and control the changes in various kind of documentation. Suppose, requirement must be added or deleted depends on changes in requirements specifications. Based on changes in stakeholder needs that the requirements are added or not. Usually, based on cost and scheduling that the requirements are deleted during development process.

5. PROPOSED FRAMEWORK

Software requirement phase mainly distinctness on examining and gathering particular requirements and objectives for the system from different points of views i.e. customer, users, constraints, system's operating environment, trade, marketing and standard. The basic process of requirement gathering is shown in figure 1.

- Again collect requirement from the client that solves above stated problems

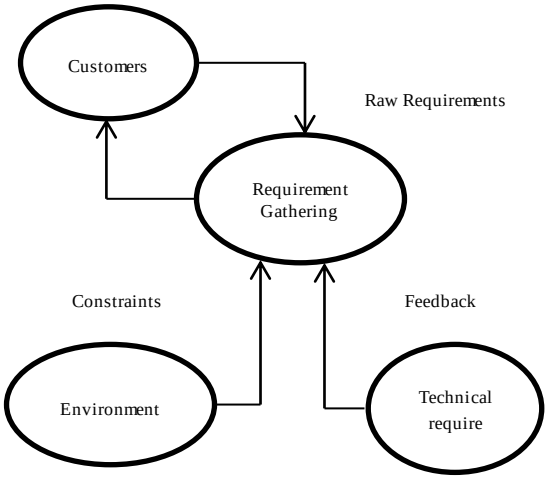

Figure 1: Requirement Collection

Software requirements phase begins with identifying clients of the system and collecting raw requirements consists of functional and non-function requirements from different view of people. Raw requirements are requirements that have not been clearly analyzed and have not yet been noted down in a well-formed software requirement notation. Frequently, customer requirements gets fail to interpret because the system analyst may fail to misunderstand the user's requirement. In this paper, post data processing system is proposed to find and detect the problem after raw collection in requirement which is listed as ambiguous Requirement, partial Requirement, Contradictory Requirement, Unconcluded, and Unable to identify and project. This phase mainly focuses to collect various requirements from various points of view that are business requirements, user requirements, information and security requirements, and some standard requirement. This problem can be done by three steps in requirement collection.

- collection of raw requirement from client
- After collection, post data processing system is to find and detect the problem during requirement analysis

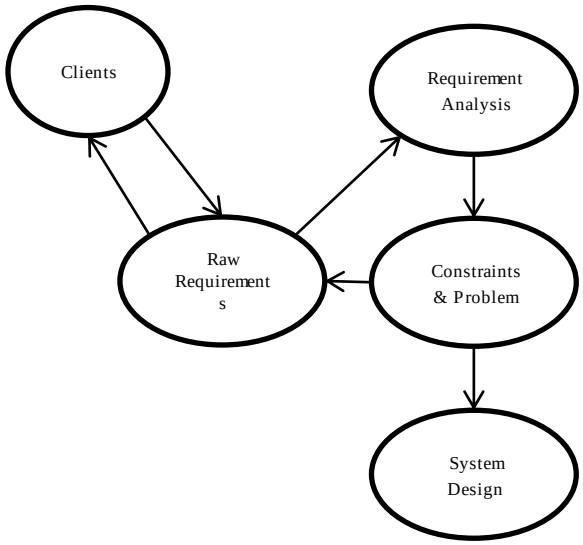

Figure 2: Post data processing system of requirement engineering

5.1 ambiguous Requirements

In several cases, organizations are unable to identify the problem and their needs and requirement. They do not know what they are exact demanding from the development process. In this the development process would be suffer and next process will blocked.

5.2 partial Requirements

Partial requirement can be stated as the organization " people do not know the exact what they are going to solve. It is very difficult process to find the things which is unknown related to development system. In this case, proposed framework has responsibility to know the exact needs and requirement of the development.

5.3 Contradictory Requirements

No system can be development on behalf of the contradictory method. The developers must be avoided contradiction during development process. To detect this problem the developers must to analyze the functional and non-functional requirements whether it is related to systemand development process.

To avoid the confusion about the task and development process, the developers must be know the nature of the organizations and identify the needs and requirements of the organizations.

6. CONCLUSION

In this paper, identified various factors of requirements is intended to refine the actual needs and requirements of the customers in enterprises or organizations. There are various parameters which are essential in requirements process, in case if the customer is unable to decide, unidentified, or uncompleted their requirement. This paper would be benefited for organizations or enterprise.

REFERENCES

[1] P. Jalote, An Integrated Approach to Software Engineering, 3rd edition, Narosa Publishing house, India, 2005.

[2] Lemos, Giese, & Müller "Software Engineering for Self-Adaptive Systems: A Second Research Roadmap" Volume 7475 of the series Lecture Notes in Computer Science, springer, pp 1-322013

[3] Pandey, U. Suman, A. K. Ramani, "Social-Organizational Participation difficulties in Requirement Engineering Process- A Study", National Conference on Emerging Trends in Software Engineering and Information Technology, Gwalior Engineering College, Gwalior,2009.

[4] N. Juristo, A. M. Moreno & A. Silva, "Is the European Industry Moving Toward Solving Requirements Engineering Problems"IEEE Software, 2002, pp. 70-77.

[5] S. Komi-Sirvio & M. Tihinen, "Great Challenges and Opportunities of Distributed Software Development – An Industrial Survey", Fifteenth International Conference on Software Engineering and KnowledgeEngineering, SEKE2003, San Francisco, 1-3 July 2003.

[6] J. Siddiqi, "Requirement Engineering: The Emerging Wisdom", IEEE Software, 1996, pp.15-19.

[7] G. Kotonya & I. Sommerville, Requirements Engineering: Process and Techniques. John Wiley & Sons, 1998.

[8] J. D. Sailor, System Engineering: An Introduction. IEEE System and Software Requirements

[9] Engineering, IEEE Software Computer Society PressTutorial. IEEE Software Society Press, 1990.

[10] Pandey, U. Suman, A. K. Ramani, "Design and Development of Requirements Specification Documents for Making Quality Software Products" National Conference on ICIS, D.P. Vipra College, Bilaspur, 2010.

[11] R. H. Thayer & W. W. Royce, Software Systems Engineering, IEEE System and Software Requirements Engineering. IEEE Software Computer Society Press Tutorial. IEEE Software Society Press. Los Alamos, California, 2012.

[12] W. W. Royce, "Managing the Development of Large Software Systems" Proceedings of IEEE Wescon, Reprinted in Proceedings 9th International Conference Software Engineering IEEE Computer Society Press, Los Alamitos. California. USA, Nov 2012, pp. 328-338.

[13] Phillip A. Laplante (2007) What Every Engineer Should Know about Software Engineering. Page 44.

[14] S. W. Ambler, Process Patterns. Building Large- Scale Systems Using Object Technology. Cambridge University Press, 1998.

[15] P. Parviainen, H. Hulkko, J. Kaariainen, J. Takalo & M. Tihinen, "Requirements Engineering", Inventory of Technologies, VTT Publications, Espoo, 2003.

[16] Kaur, R., & Singh, T, Analysis and Need of Requirements Engineering. International Journal of Computer Applications, 7(14), 27–32, 2010 Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?rep=rep1&type=pdf&doi=10.1.1.206.2789