International Journal of Computer Trends and Technology
ISSN: 2231-2803 / https://doi.org/10.14445/22312803/IJCTT-V67I3P105
Volume 67 Issue 3, 19-21, March 2019

Original Article

New Approach toward Requirements Analysis at Elicitation Phase in Software Engineering

Qutubud Din[1], Shah Faisal[2], JiongYu[3]

[1] *School of software Engineering, Xinjiang University, Urumqi 830008, PR China.*
[2,3] *School of information science and Engineering, Xinjiang University, Urumqi 830046, PR China.*

Abstract *- Requirement elicitation is a critical job in the requirement engineering process with the help of which client can easily convey his requirements for a project, and engineers can easily get information about an upcoming project. Fruitful retrieval of requirements from a client is a big challenge for requirement engineers due to poor communication, which leads to delaying and a high cost for a project. A recent project proposed improving requirement elicitation in a different way, in which requirements elicitation will be done in three phases: pre-requirements, post requirements, and final requirements. With the help of these three additional parts in the requirements elicitation phase, projects will lead toward fruitful results against to client's requirements.*

Keywords *- requirements Engineering, Elicitation, pre, post, final-requirements.*

I. INTRODUCTION

Requirements elicitation is the most condemnatory phase in the software development life cycle.
Prosperous requirement analysis always conducts a successful result in the end, but patchy requirements always show a failed result. Requirement engineering is based on five major steps [1, 2].

- Requirements elicitation
- Requirements Analysis
- Requirements specifications
- Requirements validation
- Requirements Management

In these five phases, we directly involve our client in the first phase, which is requirements elicitation means that we give 60% importance to our clients in this phase but bypass the remaining four phases and give 40% direct involvement to our clients. So its means that requirements gathering from the user is based on the first phase of the requirements engineering process. If we successfully execute our first phase, gathering valuable requirements from the user in the first phase leads to a fruitful result in the dreaming phases. But it is impossible to collect irreplaceable requirements in one attempt through direct communication with the client.

Fig.1indicate direct involvement with the user between requirements gathering.

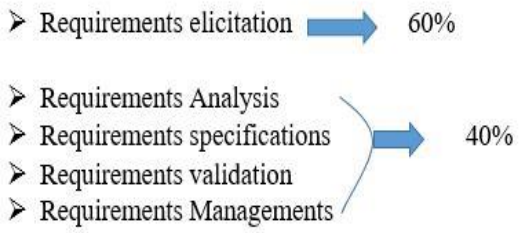


Fig. 1 Direct involvement with users between requirements gathering.

Requirement elicitation is a critical phase in the software development process, which draw the line between the success and failure of the product. Fredrick Brooks shows how requirements engineering crucially plays an important role in his work. It (Requirement elicitation) is a critical part of software development phases in which it is important to know about "what to build ."Other parts are not so crucial, which play a major part, and also, no other part than requirement elicitation is so difficult for rectification at the end [3-5]. According to Lutz, 60% of errors occur due to requirement elicitation in critical systems. So this research will add some modifications to requirement engineering. Which will show good performance in risk management, effective cost, and timely delivery of product according to user required system.

II. REQUIREMENTS ENGINEERING MAIN PILLARS

A. Requirements elicitation

Requirements elicitation is the first step in Requirements engineering in which requirements are gathered from users, clients, and stakeholders. In

Requirements elicitation, it is impossible to gather all requirements against the future product. Requirements elicitation practices include interviews, questionnaires, prototyping, brainstorming, user observation, workshops, etc.

1. Problems in Requirements elicitation

a. problem of range

For customers, there are undefined boundaries for a system that confuses them because of clarification of overall system objectives.

b. The problem of poor communication

Actually, before starting the project, it is difficult for customers that what they need because of poor communications and less practical involvement with computing atmospheres, which provides ambiguous information to engineers.

c. Problem of volatility

Over time, in most cases, requirements also change their face. So in most cases, engineers ignore the volatility of requirements and become limited to users' requirements [6].

B. Requirements Analysis

In the requirements analysis phase, engineers analyze those requirements that will fit a future product. This phase makes sure documentation, validation, and management of the upcoming product.

1. Problems in Requirements Analysis

a. Confusion of customers about their actual needs

Before starting a project, customers are unsure about their actual needs, so they provide ambiguous information about their desired product because of less awareness about existing products. And then analyst analyses that information and starts projects according to that information which unsatisfied customers [7, 8].

b. Short timeline for completion of the project

In more cases, customers define the short interval for completion of the project as are unreasonable schedule for completion. Because of less time, analysts cannot analyze customers' requirements properly, which affects the quality of the product.

C. Requirements specifications

Requirements specification describes the project that differentiates functional and nonfunctional requirements and provides use cases that define customers' instruction with software.

1. Problems in Requirements specifications

a. Ambiguous information

In some cases, customers want something else, but engineers portray that information differently, which is documented as ambiguous requirements and provides double meanings for developers [9-11].

b. Too technical

Requirements are not defined in the client's language, which causes a problem for developers.

D. Requirements validation

Requirements validation is a process that ensures that software meets all specifications. It is an iterative process practiced throughout the life cycle and checks the validity of provided information in the analysis, elicitation, and specification phases.

1. Problems in Requirements validation

a. Stated unclear requirements

Before starting a project, clients are not familiar with how to provide information for a project. They stated unclear information to requirement engineers who provide a path toward failure results at the end of the project.

b. Deficiencies in elicitation and analysis phases.

Requirements processed during the elicitation and analysis phase have some deficiencies, which creates bad space in the requirement validation phase [12, 13].

E. Requirements Managements.

Analyzing, documenting, prioritizing, tracing, and agreeing on requirements is the requirements management phase where requirement engineers control changes and communicate with concerned clients [14].

1. Problems in Requirements management.

a. Maximizing project cost

Because of incomplete requirements in the elicitation, analysis, specifications, and validation phase, errors in requirements are inherited in the management phase, which increases cost and delays the project.

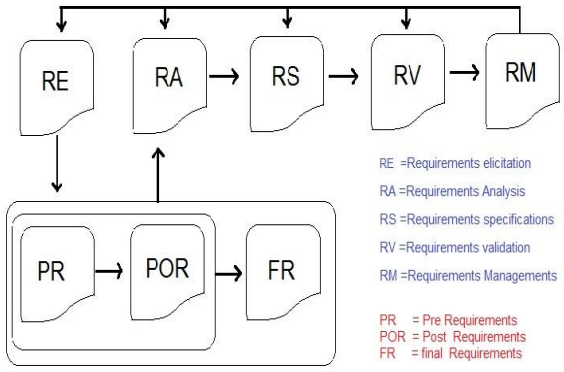


Fig. 2 Proposed parts in elicitation phase-in requirements engineering model.

2. Details Discussion Of Additional Parts In Requirement Elicitation Phase

The model is modified to pre-requirements, post-requirements, and final requirements, which is the child part of the requirements elicitation phase, which involves the user conveying his/her requirement to the requirement engineer face to face. The requirements of the engineering model and clients will discuss the problem in a better way, which is shown in Fig.2, which is discussed below**.**

a. Pre requirement

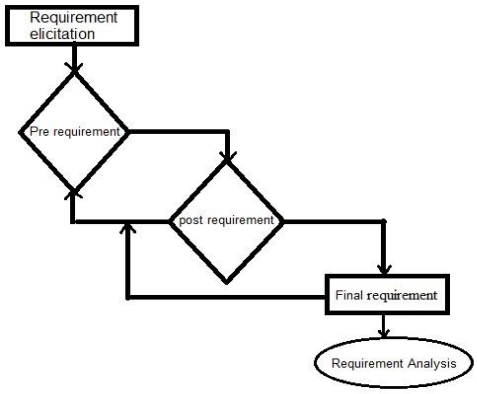
In the pre-requirement part, the user and clients will start a conversation about a new project. Here client will provide some basic information about their problem. After that requirement, the engineer will provide some information about their project in the form of prototype, paperwork, flowcharts, brainstorming, existing projects, etc. this part will be face-to-face or virtual conversation.

b. Post requirement

After pre-requirements, the requirement engineer will start a brief analysis of user-provided pieces of information in the pre-requirement part. After his analysis lack of information or ambiguous information's from the user side requirement engineer will schedule the next meeting with clients to fulfill that space. Here is a loop between pre-requirements and post requirements which will be repeated repeatedly to tell the requirements engineer is not satisfied that the requirement is 60% good to forward to the next part.

c. Final requirements

After completing the pre-requirement and post-requirement parts, the filtered requirement will be forwarded to the final parts. The requirement will be organized technically and properly and then forward to the requirement analysis phase shown in scheme 1.



Scheme 1. Proposed mechanism for requirement elicitation phase.

IV. CONCLUSION

This paper has added child parts to the requirement elicitation phase. The paper focuses on the requirement elicitation phase, which is modified to pre-requirement, post-requirement, and final requirements to ensure the accuracy and meaningful retrieval of information from clients and save time and cost for the project. These additional parts to requirement election will ensure the effective requirement for the project. This approach will effectively work in small and medium projects and be further tested for industrial projects.

ACKNOWLEDGEMENT

The School of Software Engineering, Xinjiang University, Urumqi, China, is highly acknowledged for providing research facilities.

REFERENCES

[1] Standish Group. CHAOS Demographics and Project Resolution, 2004.
[2] Lauesen, S “Software Requirements: Styles and Techniques” Addison Wesley, New York, 2002.
[3] Hickey, A.M., Davis, A.M “Elicitation technique Selection: how do experts do it?” Proceedings of the 11th IEEE International Requirements Engineering Conference, P169 - 178, 2003.
[4] Lobo, L.O., Arthur, J.D “An objective-driven process for selecting methods to support requirements engineering activities" In 29th Annual IEEE-NASA Software Engineering Workshop, P118-130, April 2005.
[5] D. Fitton and B. Bell, “Working with teenagers within hci research: Understanding teen-computer interaction,” Proceedings of HCI 2014, Sep 2014.
[6] Nuseibeh, B. and Easterbrook, S. “Requirements engineering: a roadmap” Proceedings of the Conference on The Future of Software Engineering, ACM Press, P35 - 46, 2000.
[7] Viviane Laporti, Marcos R.S. Borges and Vanessa Braganholo, Athena: “A collaborative approach to requirements elicitation” Computers in Industry 60 P 73-80, 2009.
[8] SuranjanChakraborty, SaoneeSarker, and SuprateekSarker, ”An Ex- ploration into the Process of Requirements Elicitation: A Grounded Approach," Journal of the association for information systems, Volume 11, Issue 4, P 212-249, April 2010.
[9] S. Sandiwarno, “Design of Information System Model for the Help Desk” International Journal of Computer Trends and Technology, Volume 67 Issue 1 – Jan 2019.
[10] Prasad Rajagopal, Roger Lee, Thomas Ahlswede, Chia-Chu Chiang, Dale Karolak, A New Approach for Software Requirements Elicitation, SNPD/SAWN05, IEEE, 2005.
[11] Alan Davis, Oscar Dieste, Ann Hickey, Natalia Juristo, Ana M. Moreno, Effectiveness of Requirements Elicitation Techniques: Empirical Results Derived from a Systematic Review, IEEE-RE, 2006.
[12] Bill Davey and Chris Cope, Requirements Elicitation What's Missing?, Issues in Informing Science and Information Technology Volume 5, 2008.
[13] InahOmoronyia, GuttormSindre, Tor Stlhane, Stefan Biffl, Thomas Moser, and WikanSunindyo, A Domain Ontology Building Process for Guiding Requirements Elicitation, REFSQ 2010, Springer LNCS, P 188-202, 2010.
[14] D. Fitton and J. C. Reed, "Primed design activities," Proceedings of the 9th Nordic Conference on Human-Computer Interaction – NordiCHI ’16, Oct 2016.