# Requirement Elicitation Techniques for Open Source Systems: A Review

Hafiza Maria Kiran
Department of Computer Science and IT
The University of Lahore, 1Km, Defence Road
Lahore, Pakistan

Zulfiqar Ali [1,2]
[1]Department of Computer Science and IT
The University of Lahore, 1Km, Defence Road
Lahore, Pakistan
[2]Department of Computer Science
National University of Computer and Emerging Sciences,
A.K Brohi Road, H-10/4,
Islamabad, Pakistan

***Abstract*—The trend of Open Source Software development has been increased from the past few years. It has gained much attention of developers in the industry. The development of open source software systems is slightly different from traditional software development. In open source software development, requirement elicitation is a very complex and critical process as developers from different regions of the world develop the system so it's really difficult to gather requirements for such systems. A variety of available tools, techniques, and approaches are used to perform the process of requirement elicitation. The purpose of this study is to focus on how the process of requirement elicitation is carried out for open source software and the different ways which are used to simplify the process of requirement elicitation. This paper comprehensively describes the techniques which are available and are used for requirement elicitation in open source software development. To do so, a literature survey of the existing techniques of requirement elicitation is conducted and different techniques are found that can be used for requirement elicitation in open source software systems.**

***Keywords*—Requirements engineering; requirement elicitation; open source system; requirement elicitation techniques***

## I. Introduction

Requirement elicitation is very first and important step in the process of the development of a software system [1]. In requirement elicitation, all the requirements related to the system which is going to be developed are gathered from stakeholders. Although it's a very critical phase to gather requirements, so it also effects on the quality of software. Most of the times, the cause of software system's failure is due to poor requirement elicitation [2].

The open source concept was started in the 1980s with a general public license model of Richard Stallman [3]. According to this model, software should be flexible enough so that it can be modifiable. In many cases, open source systems become distributed systems as the people from all around the world are involved in development. In such cases, the task of requirement elicitation is really crucial. Open source software has gained much attention in the industry. Open source software has changed the scenario in which millions of line of code are accessible to developers and they can read, enhance and improve the source code [4].

The reputation of open source software development has been increased from past few years and the companies and organizations are focusing on the development of such projects frequently. The importance of software requirement engineering rises expressively with the rapid growth of open source software development practices, more time and required resources for development purpose. As the requirements are the critical and important base of software, it required much time and fruitful effort so that the process can be as right as possible.

The purpose of this study is to focus on how the process of requirement elicitation is carried out for open source software and the different ways which are used to simplify the process of requirement elicitation. This paper comprehensively describes the techniques which are available and can be used for requirement elicitation in open source software development. To do so, a literature survey of the existing techniques of requirement elicitation is conducted and different techniques are found that can be used for requirement elicitation in open source software systems.

Section II introduces the requirement elicitation, Section III explains the Open Source Systems, Process of Requirement Elicitation in Open Source Systems is provided in Section IV. Section V provides the review of requirement elicitation techniques for open source systems. Section VI provides the comparative analysis of the various techniques with respected open source software categories and the last Section concludes the survey study.

## II. Requirement Elicitation

The term 'elicitation' means to obtain, gather, collect or identify. In software engineering, requirement elicitation is the process of gathering requirements from stakeholders for software development. The process of the development of every software initiates with the phase of requirement elicitation [5]. Different elicitation techniques are used for the process of requirement elicitation in which analysts note down stakeholder's wants, needs, and desires [6]. In requirement elicitation process, the analyst focus on the understanding of

Higher Education Commission (HEC) of Pakistan.

requirements, vision, and constraints of the system which is going to develop [7]. The scope of the requirements is determined in requirement elicitation phase and broader requirements are defined [8].

The phase of requirement gathering is the earliest and continuous process in software development. These requirements are gathered from different sources which include existing systems, stakeholders, documentation and the problem owners [9]. In requirement engineering process, the activity of requirement elicitations is examined the most critical activity. It is a very complex process and it also involves some other activities. Multiple techniques are available to perform these activities. The process of requirement elicitation is divided into five types: to understand the application area, find out the sources of requirements, exploring stakeholders, selection of appropriate tool and technique, elicitation of the requirements from different sources.

Before starting the activity of requirement gathering, the sources of requirements are identified. These sources include documentation, existing systems interviews, etc. [10], [11].

## III. Open Source Systems

The open source concept was started in the 1980s with a general public license model of Richard Stallman [3]. According to this model, software should be flexible enough so that it can be modifiable. In many cases, open source systems become distributed systems as the people from all around the world are involved in development. In such cases, the task of requirement elicitation is really crucial. From the developer's point of view, open source is a blend of two imperative properties [4]. The first property is the visibility and access to the source code. And the second one is the authority to make changes or enhancements to the source code. In open source software development, organizations do not pay much attention to software engineering activities like detailed requirement gathering, testing of system etc. The design of software which is going to be enhanced is totally based on the elements of existing software. Open source software development is a new, unique and different from traditional software development and the resulting product of open source software development is not the property of a single organization [6]. The importance of why to adopt open source practices is independence in the context of the price of the product and the licensing conditions. The advantages of adopting open source software development are that it ensures a high level of security, maximum stability, independence from vendors of major software. The reason of why open source software is different from traditional software is because of licensing. In open source software systems, the license fee is not required. Its lower cost is the key factor in its adoption in small businesses [3]. Open source software is flexible enough to modify it for enhancement and improvements. In terms of reliability and quality, open source software is much better than traditional software. But there are some limitations of open source software. Open source software has lack of personalized support and this software do not come with a warranty. The types of open source systems include office automation, web design, communications, E-Commerce, content management systems and operating systems.

## IV. Process of Requirement Elicitation in Open Source Systems

This is absolutely true that the process of requirement elicitation is applicable in OSS. But in the context of open source software development, the activity of requirement elicitation carried out in a different way as in traditional software development process. In OSS, requirements are presented in natural language text format rather than in a formal template [12]. Why the process of requirement elicitation is different in OSS because of the nature of the project that is distributed and a huge amount of participants like users, stakeholders, developers, and customers. Moreover, it's also different because of the informal nature of documentation and communication.

In traditional software development, there are so many contexts in which requirements can be exposed. For example, there is face-to-face requirement elicitation in the form of interviews and workshops. Moreover, there is a concept of recycling old manuals and specifications. In case of open source software development, use of these requirement elicitation techniques is slightly different. In this scenario, requirements are gathered by discussions, through email or messaging or by communication over the internet. This is the reason that these requirements are usually informal and unstructured.

## V. REQUIREMENT ELICITATION Techniques for OPEN SOURCE SYSTEMS

Requirements for open source systems come from multiple sources and these requirements are of different nature in some aspects as compared to traditional software systems [13]. As discussed in the previous sections that there multiple techniques for requirement elicitation in developing a software system [10], [11]. There are some requirement elicitation techniques which can be as it is used for open source systems as they are used for traditional systems for example [13]. But which requirement elicitation techniques we can use for open source software development, is a question. The elicitation techniques which are commonly used are mentioned in Fig. 1 and described briefly in further sections. Fig. 1 is shows the requirement elicitation techniques which are used for requirement elicitation of open source software systems.

### A. Groupware Tools

In the 1980s, the term collaborative software is being initiated to selected as groupware [14]. Groupware is basically a software application which is designed with the help of those people who are engaged in a common task for the achievement of their desired goal. In 2002, a study was conducted by researchers Rosson and Lloyd in which they tried to pay attention to find the importance and effectiveness of the process and activities of requirement elicitation in open source software development [15]. According to this study, the method used for the process of requirement elicitation was Groupware Tools. Requirements for OSS were gathered or elicited by communication among groups of several stakeholders using groupware tool.

### B. Web Survey

In the 1930s, the survey was a typical and standard way for the research purpose in different fields like marketing, social sciences, etc. [16]. These surveys are used for the purpose of data collection from a sample of persons. A number of methods are involved in data collection via surveys [17]. The first method in survey data collection which was paper and pencil interviewing has been changed into computer-assisted interviewing. And the other methods like face to face survey, telephonic survey, and mail survey are progressively replaced by web surveys. In 2015, a study was conducted by researchers named as Kuriakose and Parsons in which the technique which they used for the purpose of requirement elicitation was Web Survey. This study also focused to figure out the requirement engineering practices which the developers are following the process of requirement elicitation in open source software development. The study concluded that although the practice of requirement elicitation is much important and helpful for development process according to the survey which they conducted, the usage of RE practices in the development of OSS is low. This technique is effectively used when the target population is spread over the large geographical area [18]. Surveys should be designed in such a way that they must be clear and contain domain knowledge. Proper attention and pre-planning are required in order to make this technique successful and quick. There are some pros and cons of adopting this technique. For example, this technique is easy to gather requirement because there are multiple choice questions in the survey and same questions are asked of multiple people. This technique can only be used for general purpose software. In this technique, sometimes useful feedback is not received because of ambiguities in questions.

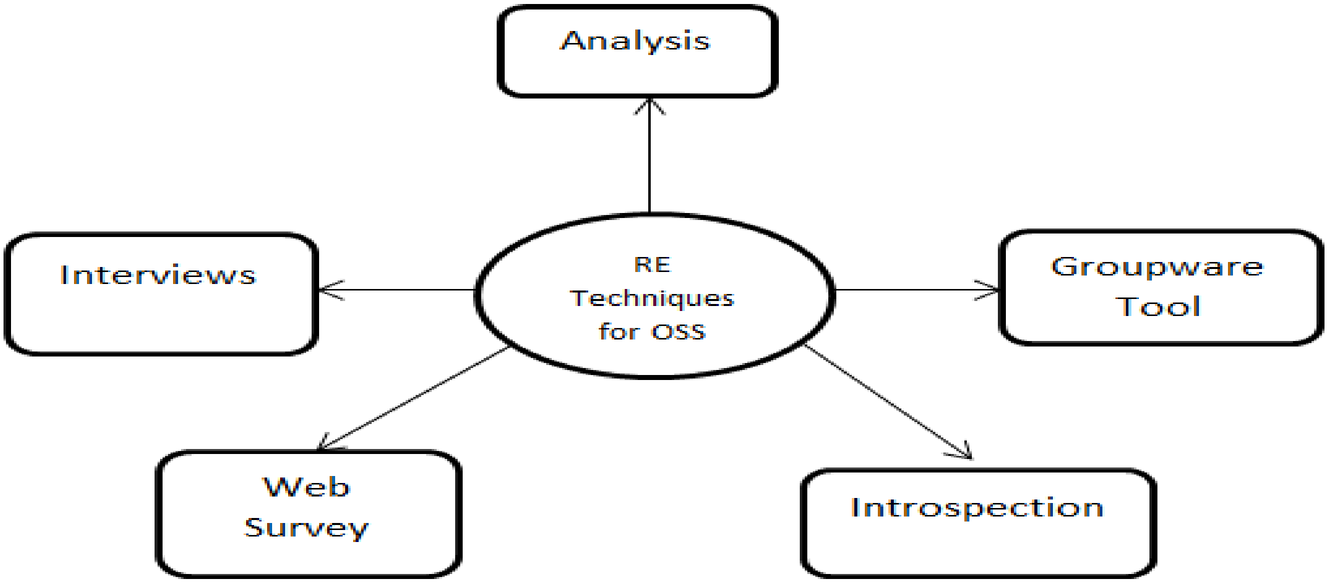

Fig. 1. RE Techniques for OSS.

### C. Interviews

In requirement elicitation process, the most common and traditional technique used is interview [19]. With the help of this technique, a huge amount of data can be collected quickly and efficiently. The feedback of using this technique highly depends on interviewer's skills. The three primary types of interviews are structured interviews, unstructured interviews, and semi-structured interviews.

Structured interviews include a collection of pre-defined questions which are asked to stakeholders [9]. This technique results in quantitative data and considered as an effective technique in requirement elicitation process. On the other hand, a semi-structured interview is a mixture of some pre-defined and some unplanned questions. It is actually a combination of structured and unstructured interviews. Whereas un-structured interviews aka open interviews include totally unplanned questions [18]. Qualitative data is produced as a result of this technique. In this technique, stakeholders and analysts discuss software system and finalize requirements. This technique is more effective when a specific issue is going to be resolved and focus is on a deep understanding of the particular issue.

There some pros and cons of interview technique, discussed in [20]. For example, this technique is really effective for complex topics and provides the overview of the whole system. The chances of no responsiveness are much low. A lesser number of people involved in this technique but still it is much time-consuming.

### D. Introspection

In this technique, the requirement analyst tries to develop all the requirements on the basis of needs and wants of stakeholders for a particular system [10]. This introspection technique for requirement elicitation is more efficient when the analyst is well aware of the domain and outcome of the project also has expertise in business processes which are performed by the user. This technique is used with a combination of other techniques to initiate the process of requirement elicitation [20]. This technique is basically the practice of observing the thoughts of stakeholders. In this technique, the system analyst should be experienced and domain expert.

As they focus on how the design of the system should be. This requirement elicitation technique can be a useful technique but stakeholders and experts are from diverse fields and they may not easily understand one another. There are some pros and cons of introspection technique. For example, this technique is quite easy to implement and there is no implementation cost required for this technique. This technique can be an initial step to start the process of requirement elicitation. Stakeholders and analyst should be aware of the domain. There is no chance of discussion between stakeholders and other experts. It is difficult for the requirement analyst to understand or imagine the situation in which new system will work.

### *E. Analysis*

The process of dividing a complex task into smaller chunks so that it can be easily understandable in a better way is called analysis. In software requirement engineering, the term requirement analysis is used to determine the needs and wants of stakeholders for a new system.

In this technique, information which comes from existing documents is gathered and then analyzed [20]. This requirement elicitation technique is successfully used in order to initiate the process of requirement elicitation. The collected information in this technique may diverge because of availability of documents and interaction with humans. This technique is mostly used when there is a need to have the detailed domain information studied by a domain expert. In this technique, experts usually analyze design documents, manuals and templates of existing systems. When there is a need to enhance the existing system or to replace it with another system, then this technique is mostly adopted.

There are some pros and cons of adopting this technique. For example, this technique is more helpful in a scenario when there is no availability of users and stakeholders. This technique helps us by providing some previous historical data about the system. This technique is also used when there is need for requirement reuse and it is an inexpensive technique. This technique is little bit time consuming as it is a difficult and time taking process to find out information from a big amount of documents and the information gathered from existing documents may be incomplete. The information which is gathered can be invalid.

## VI. Discussion

In 2002, a researcher claimed that the process of requirement elicitation in open source software development is not same as in traditional software development [21]. The requirement elicitation process in open source software development is actually a combination of the social and technical process which involves the positive social relationships development and social agreements or contracts which are negotiate informally. The techniques which are used in the process of requirement elicitation for open source software systems are mentioned in this paper. There are following five techniques mentioned in the paper which are groupware tools, web surveys, interviews, introspection, and analysis. These requirement elicitation techniques are used in different scenarios for the development of different open source software depending upon the nature and category of the software.

A comparison of these mentioned techniques is given in the paper which clearly shows that which requirement elicitation technique best matches with which type of software.

Table I show the relationship between requirement elicitation techniques and different scenarios in which they are used for open source software development. For the development of very complex software, the requirement elicitation technique which suits the process of requirement gathering is an *interview.* This technique is really an easy and effective technique for stating requirements, needs and wants of the user and stakeholder. The requirement analyst can have detailed specifications with the help of using different variants of this technique like structured, unstructured and semi-structured interviews regarding particular software which is going to be developed. The more questions asked by the interviewer from the stakeholder, the more it will be clear to state requirements.

If there is a scenario in which the availability of stakeholder or user is not possible then the requirement elicitation technique which suits best is *Analysis.* Requirement analysts study previous documentation and manuals regarding specific software so that they can understand the system and gather requirements for system development.

TABLE. I. Requirement Elicitation Techniques which suit best for major categories of OSS

| Type of S/W<br>RE Techniques | Complex Software | Stakeholder & User not available | Large no. of responses required within short time | No cost to RE process | Engagement of multiple stakeholders in common task |
|---|---|---|---|---|---|
| Groupware Tool | | | | | * |
| Interviews | * | | | | |
| Introspection | | | | * | |
| Analysis | | * | | | |
| Web Survey | | | * | | |

For the development of that software in which the requirements for a system are gathered from a large population, *Web Survey* technique of requirement elicitation is best for this scenario. Requirement analyst simply prepares an online survey and spread it to the intended audience for a response. Same questions are asked in web survey from multiple people.

For the development of software in which there is a low budget for development and the requirement analysts do not want to spend cost for requirement elicitation, the technique which best matches for this is *Introspection.* This technique is very cheap and no implementation cost is required for this technique. It is a simple and easy technique and can be adopted as an initial step to start the process of requirement elicitation.

All the discussion and suggestion of requirement elicitation techniques are based on the literature and previous studies in this domain. So according to different researchers, following techniques which are mentioned in the paper are used for the process of requirement elicitation in open source software development.

## VII. Conclusion

In this paper, a detailed review of the techniques for the process of requirement elicitation in open source software development is presented. The paper described what open source software is and how it is different from traditional software. According to literature survey, all the techniques which are mentioned in this paper have some plus points and some negative points. Some techniques are applied at early stages of requirement gathering, and some techniques are applied later according to need. Every technique is specifically suitable for a particular software system in particular scenario.

## Acknowledgment

We are thankful to the Higher Education Commission (HEC) of Pakistan as this research work is sponsored by the HEC of Pakistan in the form of Scholarship for the Ph.D. program.

## References

[1] H. Kitapci, and B. W. Boehm, "Formalizing informal stakeholder decisions--A hybrid method approach." pp. 283c-283c, 2007.

[2] B. Davey, and K. R. Parker, “Requirements elicitation problems: a literature analysis,” Issues in Informing Science and Information Technology, vol. 12, pp. 71-82, 2015.

[3] T. Jaeger, and A. Metzger, “Open Source Software,” Rechtliche Rahmenbedingungen der Freien Software, vol. 2, pp. 51, 2011.

[4] D. Spinellis, and C. Szyperski, “How is open source affecting software development?,” IEEE Software, vol. 21, no. 1, pp. 28, 2004.

[5] M. Christel, and K. Kang, Issues in Requirements Elicitation. Software Engineering Institute Technical Report CMU, SEI-92-TR-12. Carnegie Mellon University, 1992.

[6] A. Davis, O. Dieste, A. Hickey, N. Juristo, and A. M. Moreno, "Effectiveness of requirements elicitation techniques: Empirical results derived from a systematic review." pp. 179-188, 2006.

[7] S. A. Fricker, R. Grau, and A. Zwingli, "Requirements engineering: best practice," Requirements Engineering for Digital Health, pp. 25-46: Springer, 2015.

[8] M. K. Niazi, “Improving the requirements engineering process through the application of key process areas approach,” AWRE02, 2002.

[9] D. Zowghi, and C. Coulin, "Requirements elicitation: A survey of techniques, approaches, and tools," Engineering and managing software requirements, pp. 19-46: Springer, 2005.

[10] J. A. Goguen, and C. Linde, "Techniques for requirements elicitation." pp. 152-164, 1993.

[11] A. M. Hickey, and A. M. Davis, "Requirements elicitation and elicitation technique selection: model for two knowledge-intensive software development processes." p. 10 pp., 2003.

[12] I. Alexander, “Does requirements elicitation apply to open source development?,” OSSG, 2009.

[13] B. Massey, "Where do open source requirements come from (and what should we do about it)," 2002.

[14] L. Richman, and J. Slovak, “SOFTWARE CATCHES THE TEAM SPIRIT New computer programs may soon change the way groups of people work together--and start delivering the long-awaited payoff from office automation. fouttoune,” FORTUNE Magazine, 1987.

[15] W. J. Lloyd, M. B. Rosson, and J. D. Arthur, "Effectiveness of elicitation techniques in distributed requirements engineering." pp. 311-318, 2002.

[16] V. Vehovar, and K. L. Manfreda, “Overview: online surveys,” The SAGE handbook of online research methods, pp. 177-194, 2008.

[17] J. Bethlehem, and S. Biffignandi, “Handbook of Web Surveys (Wiley Handbooks in Survey Methodology),” 2011.

[18] Y. Zhang, and B. M. Wildemuth, “Unstructured interviews,” Applications of Social Research Methods to Questions in Information and Library Science, pp. 000-060, 2016.

[19] R. Agarwal, and M. R. Tanniru, “Knowledge acquisition using structured interviewing: an empirical investigation,” Journal of Management Information Systems, vol. 7, no. 1, pp. 123-140, 1990.

[20] M. Yousuf, and M. Asger, “Comparison of various requirements elicitation techniques,” International Journal of Computer Applications, vol. 116, no. 4, 2015.

[21] W. Scacchi, “Understanding the requirements for developing open source software systems,” IEE Proceedings-Software, vol. 149, no. 1, pp. 24-39, 2002.