

# Analysis of Requirement Engineering Processes, Tools/Techniques and Methodologies

**Tousif ur Rehman**

Shaheed Zulfikar Ali Bhutto Institute of Science & Technology, Islamabad, Pakistan  
rehman.trs@gmail.com

**Muhammad Naeem Ahmed Khan**

Shaheed Zulfikar Ali Bhutto Institute of Science & Technology, Islamabad, Pakistan  
mmak2010@gmail.com

**Naveed Riaz**

Shaheed Zulfikar Ali Bhutto Institute of Science & Technology, Islamabad, Pakistan  
n.r.ansari@szabist-isb.edu.pk

**Abstract**— Requirement engineering is an integral part of the software development lifecycle since the basis for developing successful software depends on comprehending its requirements in the first place. Requirement engineering involves a number of processes for gathering requirements in accordance with the needs and demands of users and stakeholders of the software product. In this paper, we have reviewed the prominent processes, tools and technologies used in the requirement gathering phase. The study is useful to perceive the current state of the affairs pertaining to the requirement engineering research and to understand the strengths and limitations of the existing requirement engineering techniques. The study also summarizes the best practices and how to use a blend of the requirement engineering techniques as an effective methodology to successfully conduct the requirement engineering task. The study also highlights the importance of security requirements as though they are part of the non-functional requirement, yet are naturally considered fundamental to secure software development.

**Index Terms**— Requirement Engineering, Requirement Elicitation, Joint Application Development

## I. Introduction

Requirement engineering (RE) is an initial activity of a software development project and it can affect the entire software development activity if not properly executed. RE is a key to the successful completion of software projects as it serves as its foundation. Requirement engineering is the most critical and complex process for the software development because of the diversification in the obtained requirements and due to rapid changes in the requirement [1]. It is always difficult to develop accurate requirements that remain

consistent in large and complex systems. The studies show that majority of the errors in the software functionality are directly linked to the mistakes done at the time of requirement gathering and elicitation phases. This study evaluates various aspects of RE including its processes, tools (techniques or approaches), methodologies and technologies. The aim of the study is to find pros and cons of requirements engineering processes, tools and techniques with particular reference to evaluating their efficacy in conducting the requirement engineering task amicably. On the basis of the critical analysis conducted for RE methodologies, we suggest the best/effective approach for gathering requirements to complete the requirement engineering phase.

This paper is structured in five sections. This section, being the introduction, provides a brief overview of the RE methodology and processes. An insight into the tools and techniques used for requirement engineering is provided in Section II. Literature review is summarized in Section III followed by a critical analysis of various RE tools and techniques and an abridged discussion on the survey RE tool and techniques is explicated in Section IV. The conclusion and prospective future work to this research is articulated in the last section.

### 1.1 Defining Requirements Engineering

RE is a vital phase of software development that aims at collecting quality requirements, analyzing and documenting them for subsequent implementation in the software code in an appropriate way in order to achieve desired functionality and meet the users' needs [2]. A precise definition of RE is defined by Zave as: "*Requirements engineering is the branch of software engineering concerned with the real world goals for functions of and constrains on the software systems. It*

is also concerned with the relationship of these factors to precise specifications of software behavior and their evolution overtime and across software families” [3].

## 1.2 Requirements Engineering Processes

Requirement engineering is an organized approach in which RE activities encompass the entire system and software development lifecycle. RE process is iterative that targets developing quality product [2]. As requirements elicitation act as foundation stone for all the subsequent software development work, therefore, all the key stakeholders are required to follow the same requirement gathering process. There are two broader categories of RE process [3]: *Requirement Gathering* (eliciting, analyzing, specifying, and validating requirements) and *Requirement Implementation* (executing the requirements in the software development activities).

### 1.2.1 Requirement Gathering:

In the requirement gathering stage, the effective RE process modal consist of the following four phases [1]:

- a. Requirement elicitation and development
- b. Documentation of requirements
- c. Validation and verification of requirements
- d. Requirement management and planning

The methodologies selected by the requirement engineer during the requirement elicitation drill ordinarily depend on the available resources, time constraint and the type of information being sought. There are five categorizes of elicitation techniques [4]. *Traditional techniques* are generic data gathering techniques that involve questionnaires, surveys, interviews, task analysis, domain analysis and Introspection. *Cognitive techniques* pertain to the knowledge/requirement gaining used to collect and prioritize requirements. Repertory grids, card sorting, laddering and protocol analysis are some of the cognitive techniques. *Group elicitation techniques* aim at eliciting better understanding of requirement through the involvement of team or groups of software engineers. Group works, brainstorming, JAD requirement workshops and protocol analysis relate to group elicitation techniques. *Prototyping* is also categorized as *Modern elicitation technique* which is used for elicitation purpose when requirements are not clear or when urgent stakeholders’ feedback is required to proceed further. *Contextual techniques* comprise ethnography, conversation analysis and observations/social analysis that serve as an alternative to the traditional cognitive techniques.

### 1.2.2 Requirement Implementation/development:

Requirement engineering processes comprise multiple activities in requirement development and as

such no single process is available that suffice all the activities; however, a good requirements engineering process model can be defined through intuition and background knowledge of the system to be developed. The RE process models are limited in number and have their specific area of application.

*The input/output of RE process*, devised by Kotonya and Sommerville, intake the following five inputs:

- existing system information
- stakeholder needs
- organizational standards
- regulations
- domain information

It also generates three outputs, namely agreed requirements, system specification and systems models. This process is general and flexible as for all the organizations only the requirements can differ, but these inputs and outputs always remain fixed [3,5].

*Linear Requirements Engineering Process Model*, envisaged by Linda Macaulay, bears lesser complexity and is primarily meant for administering small projects. This model entails five activities in sequences which are: concept, problem analysis, feasibility and choice of options, analysis and modeling, and requirement documentation [3].

*Linear Iterative Requirements Engineering Process Model*, conceived by Kotonya and Sommerville, emphasizes on accurate specifications for the system and validation of RE multiple times from the stakeholders. The model is iterative that lasts until the final requirements are attained and stakeholders get satisfied.

*Iterative RE Process Model*, formulated by Loucopoulos and Karakostas, is performs requirement engineering in several iterations and is suitable for those software development projects which are released version after version. The model consists of three simple phases elicitation, specification and validations.

*Spiral Model of RE Process*, suggested by Kotonya and Sommerville, performs RE process in spirals (or coil), where each spiral twists represent complete version of the requirements on the basis of which the system is expected to be developed. Each spiral is further divided into four quadrants namely, specification elicitation, requirements analysis and negotiation, requirements documentations and requirements validations. The model is capable to handle risks can increase project cost and compromise quality, such as specification delay, requirements change, low ROI etc.

## II. An Insight into Requirements Engineering Tools and Techniques

In RE, selection of pertinent tools and techniques in accordance with the type and complexity of the project is fundamental to eliciting requirement. This section outlines prominent RE tools and technique along with their role. These techniques are by and large classified into four main categories namely, *classic/traditional techniques*, *cognitive techniques*, *modern and group elicitation techniques* and *contextual techniques*. Each of these categories consists of a set of various techniques that are grouped together on the basis of their common characteristic and peculiarities.

### 2.1 Classic/Traditional Techniques

*Interviews:* Interviews is the common and popular method used by the requirement engineers to elicit system requirements and comprehend objectives of the system through verbal conversation with the stakeholders [6]. Interviews could be *structured or closed* (i.e., in the form of predefined questions), *semi-structured* (i.e., a blend of predefine and unplanned questions) and *unstructured or open* (i.e., an informal interview that does not involve predefined questions). The first two approaches largely aim towards acquiring quantitative data, whereas the later approach attributes to understand user expectations through open discussions with the stakeholders and acquire qualitative data [3].

*Surveys:* The survey techniques are used to get large set of requirements from a larger population that may scattered on disparate geographical locations. Surveys collect information from large number of users and it is quite economical and rapid to analyze the data through planned surveys.

*Questionnaires:* The questionnaire is a method of requirement elicitation which is simple and requires lesser time and cost. To get precise results, the questionnaire should be clear, concise and structured to obtain genuine user requirements, objective and constraints [3,6]. However, this technique lacks in the mechanism to seek users' clarification on the topic.

*Task Analysis:* This technique entails constructing top-down tasks hierarchy of the system to find out the knowledge used or required in the development of the system. Using this hierarchy, the task and sub-tasks are placed at different levels in a tree structure.

*Domain Analysis:* Domain analysis is used to gather early requirements and capture a bird eye view of the domain knowledge by investigating the existing applications and related documentation [7]. Usually this technique is used by the domain experts to study the domain area thoroughly. It is helpful in eliciting requirement from design documents, instruction manuals, templates and forms either used in the existing system or in the current business processes. Domain

analysis also encompasses the domain knowledge and its reusable concepts and components. Mostly, this technique is used when project involves replacement or enhancement in the existing legacy system.

*Introspection:* Introspection is a preparatory step in requirement elicitation where requirement engineers use their experience and expertise to acquire requirements of the stakeholders in terms of their expectations towards the new system. However, this technique mainly necessitates requirement analysts to have a massive experience in this area. It is very effective when analysts are well-known of the domain and goal of the system as well as experts in business processes that users ordinarily perform [3,7].

### 2.2 Cognitive Techniques

*Card Sorting:* In this technique, set of cards are sorted according to the name of domain entities by the customer/stakeholders along with the description of the criterion according to which the requirements elicitation cards are sorted. Card sorting helps in prioritizing most important requirements by ordering the cards. To make this technique more effective, it is important that all the essential entities are included in the process and requires that both the analyst and participants have sufficient knowledge of domain; otherwise, this technique produces wrong results. If domain knowledge is not well-known then group work is relatively more effective than the card sorting technique [3,7].

*Class Responsibility Collaboration (CRC):* It is a derived technique of card sorting and is used to represent software requirement in the form of classes where each class has its corresponding assigned responsibility to process the user requirements. CRC demonstrate relationship among classes and provides high level of abstraction. However, CRC cards are limited in delineating details about the software elicitation [3].

*Laddering:* Laddering technique aims at collecting clear answers for a series of questions from the stakeholders followed by arranging them in hierarchical order which is easy to understand and useful to prioritize the stakeholders' needs. The domain information of the stakeholder plays a very important role for the success of this technique. If requirements are too large then it becomes complex and hard to perform modifications like adding or deleting requirements anywhere in the ladder [3,7].

*Repertory Grids:* In repertory grid technique, the stakeholders are requested to build attribute and assign appropriate values to the set of specific domain entities on a grid and store requirements in the form of matrix. It involves element categorization, ordering the defined categories and assigning suitable variable with their corresponding values. Repertory grid is useful to identify similarity and differences between different information domains. Traceability also becomes easy

through this technique. Since repertory grid is more meticulous than card sorting and lesser than laddering, therefore, the efficacy of repertory grids is inadequate to delineate specific distinctiveness for the complex requirements [3,7].

### 2.3 Modern and Group Elicitation Techniques

*Group Work:* This technique is used to elicit the requirements of the system by inviting different stakeholders in a group meeting. This technique is effective to elicit requirements and resolving conflicts among the stakeholders by discussing all aspect of requirements with proper suggestions by the group members in a cooperative environment. However, it requires a lot of effort to conduct such meeting as it is always difficult to get hold of all the stakeholders at the same time [3,7].

*Brainstorming:* Brainstorming is used to generate numerous ideas in a shorter time span regardless of focusing on a specific issue through informal discussions amongst the participant of different stakeholder groups. It is mostly used in innovative type of projects where participants share their ideas on the basis of their experience and personal research about the project. The key disadvantage of brainstorming is that it cannot be effectively used to resolve major issues [3,7].

*Joint Application Development (JAD):* JAD is a business analysis approach for rapid decision making and to solve a problem quickly where a large number of stakeholders are engaged through open discussion. It is an agile approach used to elicit most of the requirements and their changeability. JAD is a structured approach where all steps, actions and roles of participants are defined for the session. Since the main goals of the system are already established before the stakeholders participate in discussion, therefore it differentiates from the brainstorming. The major focus of JAD session is to focus on the needs and desires of the business and users, but not on the technical issues. Due to its agile nature, sometimes it is incapable to validate the requirements. Also, it requires requirement engineers to possess vast experience and expertise [3,7].

*Requirements Workshops:* Requirement workshops are organized to elicit requirements for the project from the stakeholder. As compare to brainstorming and group meetings, the requirement workshops are able to provide a complete set of requirements but are relatively slow in the process of elicitation. As requirements elicited from this technique are collected after a multiple sessions, therefore, the resulted requirements are unchangeable. It is considered as a cost effective technique in terms of time and money and is feasible for only large and complex type of projects [3].

*Protocol Analysis:* In protocol analysis, the participants perform an activity to discuss the customer

requirements while talking loudly. This technique facilitates active participation of all the key stakeholders. For targeted system, the protocol analysis can provide specific information and rationale of the processes to the analyst. Sometime talking through operation, this technique may not provide true picture of requirements and unable to completely represent the real processes [3,7].

*Prototyping:* Prototype is an early product version which is launched so that customers can get experience with it and can suggest their required requirements for the next version. This response/feedback is considered as additional requirements and helps to further investigate the possible solutions. Prototyping is a useful technique to develop novel applications and to build GUI interface. This technique is used with the combination of other requirement engineering techniques like interviews and JAD. Conversely, potential hazards in prototyping are that the user often resist changes if they had become used to a specific kind of the system as well as it is also expensive in terms of time and cost [3,7].

*Use cases:* This technique intends at defining the requirements by portraying complete flow of events to the stakeholders in the form of a story telling style. Use cases are informal and easy to use that help understanding the requirements and validating them with stakeholders.

*Scenarios:* Scenarios are used to find and prepare the narrative and details descriptions of current and future processes required for developing the software project. Scenarios are commonly used after collecting the initial requirements. Scenarios also define the actions and interactions between user and the system. Scenarios are useful to validate requirements and develop test cases [6,7].

### 2.4 Contextual Techniques

*Ethnography:* In social context, ethnography is the study how people understand their problems and perceive their solutions. In requirement engineering context, the ethnography is a technique to find out how people discern their needs to be met from the software. This technique is useful for the collection of quality attribute like usability and efficiency from the peoples and these attributes are essentials for the success of project [3].

*Observation/Social Analysis:* Observation (also known as social analysis) is one of the types of ethnographic technique in which requirement engineer visits and observes the customer's environment where software services are required to be performed [6]. Alongside, the software engineer also observes the existing processes and it makes this technique more authentic as requirement engineers directly visits and observers the entire environment and verifies and

validates the requirements. Observations are usually coined with other requirement engineering tools such as interviews and task analysis. However, observations become much expensive technique when huge travelling costs are involved [3,7].

### III. Literature Survey

Pandey et al. [1] propose requirements engineering process model that consists of the following four phases:

- a) Requirement elicitation and development
- b) Documentation of requirements
- c) Validation and verification of requirements
- d) Requirement management and planning

Whence these four phases are incorporated in a RE process model, it helps produce quality requirements for software development because this approach introduces all important and hidden aspects of RE process. The proposed model helps obtain requirements that relate to the quality of the software products as well as handles the requirements changes over the time. The existing RE processes are limited to wrap few dimension of RE like requirement elicitation, requirement specification and requirement verification and validation. It is also stressed to link requirement engineering process phases with software development process in order to produce quality products.

Davey et al. [2] explain that mostly errors occur due to difficulties in communication between requirement engineers and customers or if the needs of an organization change with the passage of time. Some theories highlight that organizations always have dynamic behavior and new requirements may arise, therefore, considering that the requirements have become stable could be misleading. Personal construct theory, cognitive theory and education theory can be used to improve communication. A methodology phenomenography is also used to match the nature of conversations collected through elicitation.

Arif et al. [3] elaborate that RE being the initial step in software development activities is directly linked to the success of project because all the subsequent development activities depend on it. Wrongly perceived requirements may results in cost overrun, late delivery of the system, end user dissatisfaction and producing unreliable system. ESPI survey 1995 shows that 40-60% of the errors introduced in a project are traced back to the requirement elicitation stage. Standish Group Study published in 1994 highlights that 21.9% of the projects fail due to wrongly obtained and misleading requirements as well as rapid changes in requirements. However, by using best practices, iterative processes, standard tools and technologies in the requirement elicitation phase can help overcome these problems. Artificial Intelligence based modern automated technologies like Neural Impulse Actuator (NIA) and

web-based mobile technology to automate the process of requirement engineering could be quite useful to precise requirement gathering.

Nuseibeh & Easterbook [4] suggest that modeling and analysis phase of RE cannot be undertaken in isolation from organizational and social context where the newly developed computer systems are to be used. To implement the above aspect, participant observation and ethno-methodology can be benefitted. Moreover, the optative properties of the users and the environment should also be accounted for during RE modeling. Similarly, the inconsistencies in the RE modeling also lead to the development of imprecise systems. Such inconsistencies should be resolved through effective negotiations with the stakeholders that eventually end up developing a consensus on the desired model.

Gunda [6] states that each elicitation methods have its own strength but the important factor is to select appropriate method for specific application. A comparison of different methods is made by gathering process discussed in literature and experimental case studies. It can help requirement engineers to select the best elicitation method according to the situation and type of application as well as on the basis of characteristics and effectiveness of the method to elicit the best requirements.

Paetsch et al. [8] stress to benefit the agile software development methodology in requirement engineering. The approaches used to deliver software quickly - like agile methodology, adaptive software development and eXtreme programming etc. - follow some common rules like accepting change requirement, improved customer satisfaction, frequently delivering working software and close teamwork of both the developer and business people. These rapid software development methodologies can also be adopted for requirement gathering. However, highly skilled people are required in such agile methods due to time constraints and lack of documentation in agile methodologies.

Vijayan et al. [9] argue to use paper prototype for requirements elicitation to circumvent errors that occur in the systems due to lack of communication between users and requirement engineers. Studies show that many users face difficulty in spelling out the requirements completely and correctly until they are shown some prototype model exhibiting how the system is expected to look like. The prototype can either be created through a graphics application or hand drawn. However, it is suggested that a prototype be developed in the form of a user interface to facilitate users to comprehend the shape and structure of the system. The proposed prototype approach is divided into five steps: domain knowledge acquisition, system understanding, requirements elicitation, prototype validation, requirements stabilization. However, such an approach is suitable for small or medium size projects and results in additional cost and time to develop prototype model of the project.

Hadavi et al. [10] emphasize that tremendous growth in computer networks necessitates much importance to be assigned to security requirements which are generally classified as confidentiality, authorization, authentication, integrity, non-repudiation, intrusion detection and auditing. The current research on security requirements is focused on different aspects of software development like eliciting security requirements for SDLC processes, security requirements modeling and security threat modeling. For this purpose, UMLsec can be used in requirements elicitation as it contemplates access control policies. Though security requirements cover certain part of software development work but are highly essential to secure software from unauthorized use.

The success or failure of requirement elicitation depends on the selection of appropriate technique [11]. Many of the experts use scenarios and use-cases for eliciting requirements. Fifty percent of software products fail to satisfy the user requirements due to inexperienced analysts. By improving the ability of analyst in proper elicitation techniques selections can improve the success rate of the products. Hickey et al. [11] focus on technique selection on the basis of situational assessments. It necessitates that a meticulous attention be paid to anomalies, prerequisite skills, creation of models and modeling notations.

Carrizo et al. [12] considers that despite the verity of elicitation techniques, interviews are still considered in most of the cases; and determines set of project attributes that influence elicitation technique's effectiveness. The attributes/characteristics that can influence the technique selections and recommendation on suitable techniques for certain values of those attributes are mandatory in RE.

Jain et al. [13] elaborate that poorly specified requirements pertain to inconsistency in the selection of requirements elicitation techniques, the level of details and missing requirements on standard security solutions. Including security specifications during the requirements phase will not only reduce defects but also assist in reducing security breaches. Security requirements gathering allows comprehending information about the sensitive part of the environment and decides how security breaches can be nullified. Software Security Requirements Gathering Instrument (SSRGI) helps gather security requirements from the various stakeholders and can thus help avoid serious flaws in the resultant software product. SSRGI allows stakeholder involvement where security requirements are elicited through customer interaction; and enables authentication and authorization in LAN based client/server system. SSRGI ensures systematic consideration of security aspects during development process and can help avoid serious flaws in the resultant software product.

Zhang [14] categorizes requirement techniques in four methods and differentiates them on the basis of

communication. The four methods are: conversational methods (interviews, workshop, focus groups, and brainstorming), observational methods (social analysis, observation, ethnographic, protocol analysis), analytic methods (requirement reuse, documentation studies/content analysis, laddering, card sorting, repertory grid) and synthetic methods (scenarios, passive storyboards, prototyping, interactive storyboards, JAD/RAD, contextual inquiry). Requirements engineering is a complex social interaction process, therefore, analysts should use a proper and contextual means to perform this process.

Sadiq et al. [15] explains that the Requirement elicitation is the initial and critical knowledge intensive activity where elicitations are done by many techniques like use cases, brainstorming, other elicitation method and JAD. JAD is considered as the basic technique used to elicit and prioritize requirements. Requirements can be mathematically prioritized after eliciting through existing methods like JAD, ontology and so on. Further, requirements can easily be ranked and by prioritizing the requirements, best quality products that match with the customer needs and preferences can be produced.

A lot of challenges like communication and coordination can impact the effectiveness of requirement engineering. Lloyd et al. [16] conducted an empirical study to investigate the effectiveness of requirements engineering in distributed environment using role-play methodology. Distributed requirements engineering is more effective when stakeholders participate actively in synchronous activities of the requirements process.

#### IV. Critical Analysis

A critical review of the various tools, techniques and methodologies for RE proposed in the contemporary research is presented in Table I.

Table 1: Critical Review

Ref	Proposed Methodology/Issue Discussed	Strengths	Limitations/Weaknesses
[1]	A requirements engineering process model consisting of requirement elicitation, documentation, validation and verification and requirement.	Obtains quality requirements. Trans-communication amongst various processes. Accommodates changes in an effective manner.	-
[2]	Method to discover missing requirement elicitation.	Suggest three methods to make effective communication to get best requirements to improve success factor.	Incomplete requirements from conversation lead to the failure.
[3]	Modern automated technologies to be used in the RE process. A combination of RE tools/techniques are more effective	Help obtain accurate user requirements. Help reduce project failure rate.	-
[4]	Using participant observation and ethno-methodology in RE.	Incorporation of optative properties of the environment helps in successful deployment of the system.	A review of the existing RE practices is provided.
[6]	Selection of elicitation technique.	Helps to select appropriate technique for specific application.	Experts required who are familiar with their effectiveness.
[8]	Incorporate agile methodologies in RE.	Helps save time and cost in RE phase.	Highly skilled and experienced professionals are required to undertake such tasks.
[9]	Use of paper prototype for requirements elicitation to enhance communication between users and requirement engineers.	Helps obtain requirements completely, precisely and correctly.	Approach is only suitable for small or medium size projects. It also results in additional cost and time to develop prototype model of the project.
[10]	The need to address security requirements in SDLC processes, security requirements modeling and security threat modeling are highlighted. UMLsec is proposed as a possible solution to be employed to elicit security requirements.	The proposed methodology helps ensure auditing, confidentiality and integrity of software.	-
[11]	Selection of appropriate elicitation technique by experts.	Helps to get precise requirements to satisfy users and for product success.	Analysts must be expert / experienced.
[12]	Framework for developers to decide/choose that which elicitation technique is best.	Made easy to select appropriate technique selection.	Requires more research for its effectiveness.
[13]	Discuss instrument to gather security requirement.	Helps to avoid serious mistake in the resultant software product.	-
[14]	Comparison and categorization of requirements elicitation techniques.	To get best requirement method by understanding the environment.	-
[15]	Techniques for ranking and prioritization of software requirements.	Helps in prioritizing the requirement to satisfy users and produce quality product.	Difficulty in large and complex projects.
[16]	Effectiveness of elicitation techniques in distributed environment.	-	Wrong technique may lead to very serious consequences like increase in cost and defective product.

## V. Conclusion and Future Work

Requirements elicitation is more of an art than a science due to the kind of skills and experience required to undertake this task on a varied nature of the software development projects. RE processes needs to be iterative that ensure maximum user participation and also facilitate change requirements from time to time. Requirement management and project management best practices can be amalgamated to prepare the SRS (the final outcome of the RE process) that is flexible enough accommodate and implement modified and changed

requirements. Due to relative merits and limitations of the existing methods, it is always suggested to use an assortment of different techniques to attain perfection in the requirements elicitation phase. In fact, many of RE techniques intrinsically necessitate to be used in conjunction with other techniques due to their commonality in complementary attributes.

The survey of the contemporary literature reveals that group workshops, interviews, observation and scenarios techniques along with their variations are still the most commonly used techniques in requirement elicitation.

Prototype based models have also been proposed in the literature, but such methodologies suffer from additional cost and time to complete the project and also bear limitation of their applicability to small or medium size projects.

The selection of tools, techniques and approaches to elicit requirements largely depends on several factors like application domain, type of the system being developed and current status/stage of the development etc. In this paper we have surveyed various tools, techniques and methodologies for requirements elicitation process coupled with their selection procedure to choose the best and more appropriate elicitation techniques. Based on the analysis and comparison of different techniques, we conclude that every elicitation technique and methodology has certain pros and cons. Some techniques are good at eliciting the basic requirements; some are meant for actual needs; some suitable are for understanding requirements; some are appropriate for quick elicitation; and some are pertinent to carry out improvements. Only few techniques are popular among project managers, stakeholder and the software engineers. However, none of these techniques can solely serve the purpose; therefore, it is suggested to use a mixture of the techniques discussed in this paper keeping in view type of the project, development environment, time constraints and specific needs of the customer. Selection of right techniques carries much importance as it can lead the software project to the successful completion. Moreover, the task of requirements elicitation should be carried out by the experienced analyst.

Considering the aforesaid factors, we intend to propose an efficient requirements elicitation technique that amalgamates key features of different techniques, so that it is capable to help build integrated tool for capturing, modeling and analyzing security requirements through standard procedures.

## References

- [1] Pandey, D., Suman, U. & Ramani, A.K., 2010. An Effective Requirement Engineering Process Model for Software Development and Requirements Management. 2010 International Conference on Advances in Recent Technologies in Communication and Computing, 0, p.287-291.
- [2] Davey, B. & Cope, C., 2008. Requirements Elicitation – What 's Missing ?Issues in Informing Science and Information Technology, 5(1), p.53-57.
- [3] Arif.S., Khan. Q. & Gahyyur. S.A.K., (2009-2010). Requirement Engineering Processes, Tools/Technologies, & Methodologies, International Journal of Reviews in Computing (IJRIC), ISSN: 2076-3328, Vol.2
- [4] Nuseibeh, B. & Easterbrook, S., 2000. Requirements engineering: a roadmap C. Ghezzi, M. Jazayeri, & A. L. Wolf, eds. Context, 1(258), p.35-46.
- [5] Kauppinen, M. et al., 2004. Implementing requirements engineering processes throughout organizations: success factors and challenges. Information and Software Technology, 46(14), p.937-953.
- [6] Gunda S. G., (2008). Requirements engineering: elicitation techniques. Trollh ätan;
- [7] Zowghi. D., & Coulin. C., (2005). Requirements Elicitation: A Survey of Techniques, Approaches, and Tools, Engineering and Managing Software Requirements, Aurum, A & Wohlin, C (Eds.), Springer, USA,
- [8] Paetsch, F., Eberlein, A. & Maurer, F., (2003). Requirements engineering and agile software development. WET ICE 2003 Proceedings Twelfth IEEE International Workshops on Enabling Technologies Infrastructure for Collaborative Enterprises 2003, 2(3), p.308-313
- [9] Vijayan, J. & Raju, G., 2011. A New approach to Requirements Elicitation Using Paper Prototype. Science and Technology, 28, p.9-16.
- [10] Hadavi M. A., Hamishagi V. S, Sangchi H. M., Security Requirements Engineering; State of the Art and Research Challenges. Proceedings of the International Multi Conference of Engineers and Computer Scientists 2008 Vol I, IMECS 2008, 19-21 March, 2008, Hong Kong.
- [11] Hickey, A.M. & Davis, A.M., 2003. Elicitation Technique Selection : How Do Experts Do It ? In M. D. Alan, ed. Requirements Engineering. IEEE Computer Society, p. 169.
- [12] Carrizo, D., Dieste, O. & Juristo, N., 2008. Study of Elicitation Techniques Adequacy. In 11th Workshop on Requirements Engineering WER2008. pp. 104-114.
- [13] Jain, S. & Ingle, M., 2011. Software Security Requirements Gathering Instrument. IJACSA International Journal of Advanced Computer Science and Applications, 2(7), p.116-121.
- [14] Z. Zhang. 2007. Effective Requirements Development - A Comparison of Requirements Elicitation Techniques. In Proceedings of the 15th Software Quality Management Conference: (SQM '07), pages 225–240, Tampere, Finland, British Computer Society
- [15] Sadiq M., & Shahid. M., 2009. Elicitation and Prioritization of Software Requirements, International Journal of Recent Trends in Engineering, Vol 2, No. 3

- [16] Lloyd, W.J., Rosson, M.B. & Arthur, J.D., 2002. Effectiveness of elicitation techniques in distributed requirements engineering. Proceedings IEEE Joint International Conference on Requirements Engineering, 0, p.311-318.

### Authors



**Tousif ur Rehman:** Student of MS-Software Engineering in Shaheed Zulfikar Ali Bhutto Institute of Science and Technology (SZABIST), Islamabad, Pakistan. Research interests are in software engineering. Working as Software Engineer in Shaheed Zulfikar Ali Bhutto Institute

of Science and Technology (SZABIST), Islamabad, Pakistan.



**Muhammad Naeem Ahmed Khan:** Obtained D. Phil. degree in Computer System Engineering from the University of Suusex, Brighton, England, UK. Presently, he is affiliated with Shaheed Zulfikar Ali Bhutto Institute of Science and

Technology (SZABIST), Islamabad, Pakistan. His research interests are in the fields of software engineering, e-governance, cyber administration, digital forensic analysis and machine learning techniques.



**Naveed Riaz:** Obtained his PhD in Computer Engineering from Graz University of Technology, Austria. He is affiliated with Shaheed Zulfikar Ali Bhutto Institute of Science and Technology (SZABIST), Islamabad, Pakistan. His main research areas are

focused around software debugging, formal verification and software engineering.

**How to cite this paper:** Tousif ur Rehman, Muhammad Naeem Ahmed Khan, Naveed Riaz, "Analysis of Requirement Engineering Processes, Tools/Techniques and Methodologies", International Journal of Information Technology and Computer Science(IJITCS), vol.5, no.3, pp.40-48, 2013.DOI: 10.5815/ijitcs.2013.03.05