THE USEFULNESS OF RE PRACTICES IN S/W DEVELOPMENT

Dhirendra Pandey[1*], Ugrasen Suman[2], Ashwani Kumar Ramani[2,3]

[*1]Department of IT, B. B. Ambedkar University, Lucknow (U.P.) –226025, India
prof.dhiren@gmail.com

[2,3]School of Computer Science and Information Technology
Devi Ahilya University, Indore (MP) – India
[2]Ugrasen123@yahoo.co.in, [3]ramaniak@yahoo.co.in

Abstract- Requirement Engineering (RE) research has reported that incomplete requirement specification, uncontrolled changing requirement, unsecure and outdated requirement elicitation, and lack of user involvements are major reasons of software failure. The presented research paper introduces various requirement engineering issues, which can be helpful to develop effective, secure, updated, and quality requirement engineering process for software development. A systematic review on requirement engineering practices along with our published research work including RE process model, Information Requirement Engineering (IRE), Security Requirement Engineering (SRE), and requirement modeling is highlighted, as well as measurement analysis of proposed RE process model is also described in this research article.

INTRODUCTION

Requirement Engineering (RE) is an important as well as challenging practice for software development. It is a process of providing user requirement that can further be used to implement in software development. The systematic execution of RE can be useful for system analysts to discover information and achieve project objectives. [1, 2]. We have observed that the most common problems associated with software development are related to requirement. Analysis on RE has reported that imperfect requirement, changing requirement and specification as well as lack of user interest are the cited factors, which caused project to be challenged [39]. Therefore, a correct, consistent and complete way to collect, understands, specify and verify user requirement is necessary for software development.

In addition, RE is a systematic approach to elicit, organize and documenting the requirement of the system. It is a process that establishes and maintains the relationship between a customer and the project team on the changing requirement of system. RE is a critical step of software development process because errors at this stage inevitably led to the later problems in the software development process. On observation, we have found that detecting and repairing errors in the maintenance stage can be more expensive as compared to RE phase. Thus, the RE is a key process to the success of producing high quality software products. Many researchers are interested in analyzing the RE practices because this area has a great attention and future scope.

This paper explores various requirement engineering aspects such as, requirement elicitation, requirement specification, requirement verification and validation. Also, it analyzes the requirement required for designing, developing, testing, implementing and maintaining the system or software product. Requirement is an important milestone for developing quality software products. In the following of research article, the state-of-the-art and research objectives of presented research are presented in Section 2 and 3 subsequently. Requirement engineering issues, i.e., information RE, security, RE, etc. are described in Section 4 and their subsequent sections. Finally Section 5 presented the concluding remarks. A bibliography in the latest research on RE is also included at the end of the article.

STATE-OF-THE-ART

The purpose of RE is to ensure that the software development team produces software or system that satisfies user needs. Also, the primary goal of RE is to capture the requirement from various stakeholders and implement them in software development activities [39]. Hence, the impact of RE in software development process has very vast research scope. The RE process mainly focuses on two important aspects, namely; requirement development and requirement management. Requirement development includes elicitation, analysis, specification, verification and validation activities. In requirement development, the expected user classes are identified for the product, and received information from the user can be analyzed.

In addition, requirement management is a critical activity that comprises of sub-activities such as, managing requirement over time and across product families [11]. This view of RE for software development is a very significant field of research having good amount of research work and has a great scope of software development with more reliability and satisfaction. The success of software development depends upon the fitness for the needs of its user and business environment [3, 4]. Many researchers have already presented their research work on RE practices. The designer of Information System (IS) and programmers often begin designing and programming the incumbent system without understanding the user or stakeholders' needs properly. In addition, we have observed that the future system to be effective, it has to be balanced between the technical view of designers and programmers as well as social view of user and customers [39]. From literature survey, we have found that the carefully identified and elicited software requirement is a key issue for project

success [6]. At the same time, the cost of correcting a system error after delivering the complete system is higher than the cost of correcting a similar error during the requirement analysis phase [7]. Since requirement often change during development, it is important to manage changing requirement [8]. There exist various software measures for the requirement management activities have also been found in literature surveys. A set of requirement indicators have been observed which are compatible with the measurement practices of the Capability Maturity Model (CMM) for software [9]. Some research works have made state of the practice surveys that are used for developing large information system with the help of RE [29]. Also, various researchers have given a detailed analysis for managing information system for industries. They have also made survey and clarified the fundamental problems in RE practices [10, 11].

The system analyst plays an important role in RE process. There have been various research problems and future research direction related to RE such as, technological enhancement, modeling, requirement reuse, globalization of RE process etc. have been observed in literature survey [12, 13]. On the other hand, the importance of RE practices on classified software packages has also been found in literature [14]. Empirical evidences exist in literature also demonstrate that organizational issues are important in RE process [15, 16]. Some researchers have introduced anti-requirement [17]. It is a requirement of the malicious user that subverts existing requirement. It can be dangerous for every software organizations. Various research finding have already recognized the importance of incorporating customer requirement into the innovation process [18].

In the last decades, many of the researchers investigated and integrated the security analysis and RE practice. Security is one of the most important requirements for a system to produce accurate and secure software product with high quality [19, 20]. Some researchers have also been interested in improving RE practice and defining RE process because of their confidence that RE can be the key to develop a successful system [21, 22]. However, implementation of RE work products throughout the organization and convincing user to apply RE practices in software development process is considerable challenges [23, 24, 25]. For the last some decades the RE has given a significant attention to developers and researchers for designing quality software products. The success of the organization-wide adoption of RE practice depends on human, social, cultural, global, personal, organizational, technological, and economical issues [26, 27, 28].

RESEARCH OBJECTIVES

In order to perform RE, the system requirement must be properly identified, analyzed and reviewed early in the software development process. The RE focuses on discovering, analyzing, documenting and managing system requirement at the early stages of software development process [29, 30, 31]. The research on RE practices indicates that RE process critically influences the success of software development process. In this research article, the following

research objectives have been presented for analysis of various RE issues in software development, thereby, effectiveness of RE can be observed.

a. Analyze the RE issues and practices: The success of system depends upon the RE process and fitness of the stakeholders needs. The objective of the presented research is to investigate and analyze the effectiveness of RE in software development processes. It can be performed by involving the analysis of requirement management and requirement development techniques. Also, the study of interview and social-organization participation difficulties is another important objective of RE research. The impact of security requirement in SRS documents is also included in the research. This research indicates that the quality of RE practices influences the success of software development processes.

b. Design a RE process model: RE is generally accepted to be the most critical and complex process within the development of software systems. It helps to describe a multidisciplinary role of requirement engineering process as well as the patterns for social interaction. Furthermore, an effective requirement engineering process model can be helpful for requirement elicitation, specification, verification and validation and for requirement management. The objective of the research is to introduce a process model for RE and requirement management. The adaption of RE process in software development provides a good impact on the production of quality software product. The RE model can also be used in larger software development process, where the requirement can change over the time.

c. Incorporating security requirement issue in RE: The objective of the research is to incorporate the security requirement issues into the RE process. The security requirement can be used in an organization for designing accurate and secure software products. Also, security requirement have given significant attention in recent years in developing the secure software products. Therefore, security RE approach is necessary to avoid possible threats such as, anti requirement, leakage, wiretapping etc. during establishing the relationship between security requirement and the specification of software performance.

d. Usefulness of information in RE: Analyzing and extracting useful information according to the business and technological needs is always required for decision making. The aim of the research is to present a framework for Information RE (IRE) that collects the information from various sources such as, market survey, stakeholders and competitor organization. The collected information will be processed and finally software will be developed that satisfies the market needs. The objective of IRE approach is to define information requirement that can help software developers to gather information in effective manner from various stakeholders' and user over the years.

e. Measurement analysis of RE process: Measurement analysis of proposed RE process model for designing quality software products is another important objective of the presented research. It presents the measurement based on system dynamics model as well as performance analysis for measuring performance of the various

existing and proposed RE process, in term of quality, cost and schedule adherence. Thus, the proposed RE process model is compared with existing models. Finally, we have observed that neither linear model nor purely iterative model is able to perform every activities of RE process. Also, it is found that the RE process model can be beneficial from a combination of linear and iterative structures.

f. Modeling requirement framework: Most of the requirement documents are often ambiguous because these are written in less formal and imprecise natural languages. Generally, it is observed that these requirement are generally gathered in textual format that may be vague, unclear and sometimes it may be changed by overwrite. Without modeling the requirement documents, the requirement is difficult to understand. The aim of the research is to present requirement modeling framework for providing the clear view of user and stakeholders' requirement, thereby vagueness can be minimized. The model strengthens requirement engineers to understand and elicit the vague and implicit requirement.

REQUIREMENT ENGINEERING PRACTICES

RE is an essential stage of software development and its success depends upon various social and organizational issues. The various important technical issues of RE are the appropriate use of RE tools and techniques (i.e., graphical editing and traceability, behavioral modeling and databases, and word processing, etc.), effectiveness of various RE dimensions (i.e., requirement elicitation, requirement specification, and requirement verification and management, etc.), and the implementation of RE practices [35]. The proper employment of aforesaid RE issues in software development results in quality software products. Another challenging issue in RE is to conduct interview for requirement gathering in a systematic manner to avoid software project failures. The improper capturing system requirement is the major factor in failure of 90% of large software projects and poor requirement management can be attributed to 71% of software projects failure [36]. The research output indicates that, failed or an abandoned system by aforesaid reasons causes major loss in software industries.

The information related to interview were collected from various online databases such as, Academic Research Library (ACL), i.e., ProQuest, Australian Public Affairs (APA), Full Text (Informit), Business Source Premier (BSP), Communication & Mass Media Complete (C&MAC), i.e., EBSCO, Emerald Management Xtra, i.e., Emerald, Expanded Academic ASAP (Gale), Factiva, ACM library and the IEEE Library. In addition, some of the information is also retrieved from Google Scholar and Google Search. The key references from various online databases and search engines were explored to find out requirement gathering information through interview. At last, we have found that many authors imply to be talking about requirement gathering, but quickly gloss over these issues and then perform research on one of the requirement analysis techniques.

In this research, we have used the interview model and cognitive science for effective interview to collect the clear and complete requirement from various stakeholders [46]. The interview model can help to conduct effective face-to-face discussion between stakeholders. An interview model is presented to illustrate effective requirement gathering process, which is shown in Figure 1. The model can be useful for conduction of effective face-to-face interaction between analyst and user in order to elicit requirement. The model has four divisions, i.e., conspicuous requirement, user knowledge, analyst knowledge and mutual understanding between user and system analysts. The conspicuous requirement can be shared either user or analysts, whereas, the user knowledge and analyst knowledge are required to exchange between these stakeholders for imperative requirement gathering. Also, the mutual understanding between stakeholders can overcome the conflicts that arise in interview process. The segments of the models are described in the subsequent paragraphs. Division A represents the knowledge, which is common to both analyst and user. Repeated interviews would hope to increase the size of this quadrant.

The division C represents the knowledge that the analyst has but end user do not have. It includes knowledge of the analyst skills as well as the knowledge gained from education and training. By making use of this knowledge, analyst will be able to provide the product knowledge. The knowledge that the user has but the analyst does not have, is represented in division B. It includes understanding of the unique business models of the user business. The analysts try to learn such knowledge from user. The division D represents new knowledge that will be created from the interaction of user and analyst.
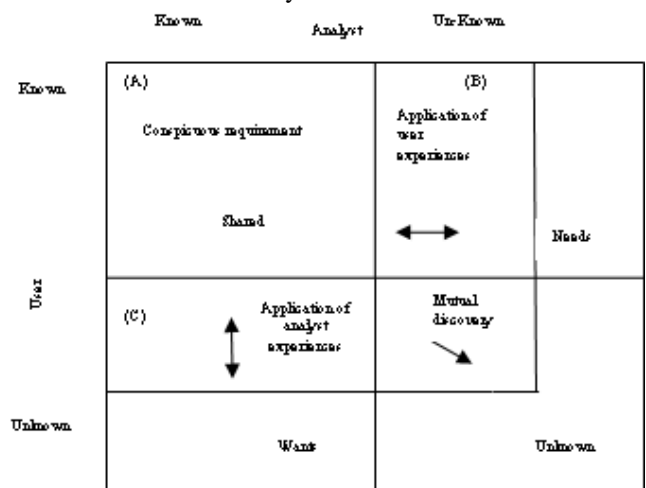

Figure.1. Interview model which shows the elements (user) and (analysts)

We have assumed that the analyst can ask questions for collecting more requirements from the user and user has to answer the questions thereby requirement can be finalized.

The main purpose of cognitive interviewing approach is to evaluate origin of response errors in survey questionnaires through an interdisciplinary effort using survey methodologists and different psychologists.

It focuses mainly on the questionnaires rather than entire survey administration process that the respondents use to

answer survey questions. For conducting the cognitive interview, volunteers are recruited, and interviewed either in a laboratory environment or in some other private location to discover concealed requirement. Cognitive interviews deals with the communication between people, which is hampered by the limitations of human cognition. Especially, such communication problems arise when it needs to be conducted by some specific languages. The communication problems in this process can be classified as the limitations of humans' cognition as information processors, the complex nature of requirement, and the obstacles encountered in user and analyst interaction. The research reported that interaction obstacles in cognitive interview can also be characterized by internal company policies or by the external customer characteristics. There is some research work carried out on language difficulties, which is arising because of varying software development terminology.

The most fruitful outcome can arise by applying theory of individual cognition to improve conversational performance between stakeholders and increase the ability to requirement gathering. An important issue of requirement gathering is the ineffective relationship between customer and developers, which results poor requirement gathering process that causes the software project failures. Some researchers have found that there is absolutely lack of attention among experts about the way they elicit the information or knowledge. An effective relationship can be discovered between behaviors of the stakeholders and high quantity learning outcomes. Teaching approaches were designed to encourage successful learning behaviors. A strong relationship can be found from stakeholder's experiences, the nature of learning, and the value of learning outcomes. Based on the findings of our research reported, we have found that the improved understanding of requirement gathering conversations can be obtained by cognitive interview approach. Finally, the research efforts have originated that interviews are the most efficient way of acquiring requirement, if it is performed resourcefully.

The effective participation between social (i.e., user, clients, etc.), and organizational (i.e., developers, analysts, etc.) elements is another important issue in RE, which can affect the entire requirement gathering process. One of the challenges is that each community interprets things in the light of their own background assumptions. This is especially problematic with non-interactive participation, where there is less opportunity to verify that the reader has interpreted requirement as it was intended. The requirement engineering phase is characterized by the strength and significance of participation activities. During this phase, various stakeholders must be able to communicate their requirement to the analysts and vice versa for their validation. The presented study describes a field investigation into the problems of participation among disparate communities involved in the requirement elicitation and specification activities [37]. In this research, a combination of learning, data gathering and analysis techniques were applied to investigate the participation problems [37]. The literature survey and empirical study have been used for the research on participation difficulty. The cross sections of social science and Computer

Supported Co-operative Work (CSCW) literatures were also surveyed to analyze the empirical results and reasoning about possible causes and consequences of participation difficulties.

The research work on participation difficulty was carried out in two stages. The first consisted of informal interaction between stakeholders and observations to establish some knowledge about practices and methodologies of both developers and their customers. These interaction or interview were concentrated mainly on the participation channels between stakeholders participating in software development project, as well as on the problems that can be attributed to the ineffectiveness of those participation channels. Other management and technical issues such as, place and schedule of interview etc. were also discussed. Most of these interviews were recorded for further analysis and reference. The second stage of this work was based on two questionnaires; one for stakeholders and another for software developers. These questionnaires were designed to receive a quantitative evaluation for the various aspects of the participation activities during requirement gathering. Questionnaires were sent to some Indian software companies and their responses represent organization profile of the companies that were targeted.

The result of the study are discussed in term of their relation to three major participation barriers, i.e., effectiveness of the existing participation channels, restrictions on expressiveness imposed by management as well as social and organizational barriers. There are two standard approaches to resolve aforesaid problems. The first approach emphasizes on development of better notations and effective use of electronic repositories. The second emphasizes on the importance of interaction between development team and other stakeholders that has given rise to practices such as, end-user participation and ethnographic techniques. Each of these approaches has its own set of limitation, and neither directly addresses the question of facilitating appropriate and effective participation over restricted channels. The study indicates that the practitioners should adapt a compromised form of the two approaches for aforesaid participation barriers such as, by enriching notations with natural language descriptions and utilizing the personal contact of face-to-face discussions [37].

Further, the impact of security requirement in Software Requirement Specification (SRS) documents are also identified and discussed. We have observed that the security segment is normally untouched or partially included in SRS documents. Therefore, the safety and security requirement can be an important part in non-functional requirement section of SRS document. For example, requirement, which is concerned with possible loss, damage, or harm, can be included in safety requirement specification. This section also specifies the safeguards or actions taken against the risks, as well as all the necessary actions that can be used to prevent the various risks. The security or privacy issues such as, user identity authentication requirement and physical security requirement should also be included in SRS documents. In addition, it refers some external policies or

regulations containing safety and security issues that can affect the software design or coding [38].

In short, the presented research has discussed some of the important issues of RE such as, appropriate use of RE tools and techniques, role of interviews, and participation difficulties between customer and developers. The incorporation of security requirement has a great impact in SRS documents, which is also reported in this section.

RE Process Model:

The systematic RE process can help system analyst to improve software development process. There exist various effective RE process models, i.e., Kotonya, Maculay, and Shams-Ul-Arif RE process model [19, 45]. These existing models are limited to cover only few RE dimensions such as, requirement elicitation and requirement specification. Also, these process models are unable to incorporate their phases with software development process in the right manner. It is generally accepted that the requirement can change over the time and during software development process, which causes the unexpected results. Hence, controlling and managing the continuously changing requirement is become more necessary for quality software development. Therefore, we have presented a different RE process model for proper software development and requirement management which is shown in Figure 2.

The proposed model has mainly four phases, namely; requirement elicitation and development, documentation of requirement, validation and verification of requirement, and requirement management and planning. Requirement elicitation and development phase includes requirement analysis and allocation and flow down of requirement. Documentation of requirement includes identification of requirement, and software and system requirement specification. Validation and verification of requirement phase is concerned with conforming the documented requirement. The requirement management and planning phase is executed independently for an effective management of requirement and controls the continuously changing requirement [39].
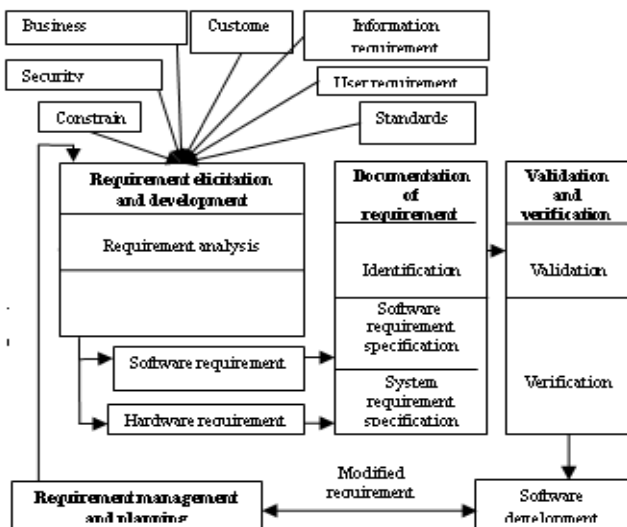

Figure.2. RE process model depicts (requirement elicitation and development) (documentation of requirement) (validation and verification of requirement) and (requirement management and planning)

The proposed model has mainly four phases, namely; requirement elicitation and development, documentation of requirement, validation and verification of requirement, and requirement management and planning. Requirement elicitation and development phase includes requirement analysis and allocation and flow down of requirement. Documentation of requirement includes identification of requirement, and software and system requirement specification. Validation and verification of requirement phase is concerned with conforming the documented requirement. The requirement management and planning phase is executed independently for an effective management of requirement and controls the continuously changing requirement [39].

The proposed model integrates all the important aspects of RE into software development process in order to find out useful requirement from various sources. The requirement management and planning phase is incorporated in proposed RE process model and associated with every software development process. The RE activities in presented model such as, elicitation, documentation of requirement, verification and validation, and requirement management are tightly interconnected to the software development phases. Also, it can help to recover and modify the requirement when requirements change during development and maintenance phase. Software developer can design the software system, which will be able to avoid and manage changing requirement with the help of proposed model.

Requirement Elicitation and Development:

Requirement elicitation and development phase mainly focuses on examining and gathering desired requirement and system objectives from different perspectives (e.g., customer, user, constraints, system's operating environment, trade, marketing and standard etc.). Requirement elicitation phase begins with identifying stakeholders of the system and collecting raw requirement from various viewpoints. The raw requirement is a requirement that have not yet been analyzed and written in the final requirement specification documents. The elicitation phase aims to collect requirement from different perceptions such as, business requirement, customer requirement, user requirement, constraints, security requirement, information requirement, standards etc. Typically, the specification of system requirement starts with observing and interviewing people.

Furthermore, user requirement are often misunderstood because the system analyst may misinterpret the user's needs. Hence, the understanding of requirement is necessary for requirement development. In addition to requirement gathering, standards and constraints play an important role in requirement development. The development of requirement may be contextual to the system development. The pictorial form of development of requirement is shown in Figure 3, in which the system analyst collects raw requirement from user and then performs detailed analysis on requirement and receives the technical feedback from developers. At the same time, the environmental constraints from organization are also analyzed. The final outcomes are compared with the technicality of the system, i.e., conversion of users' requirement into technical requirement

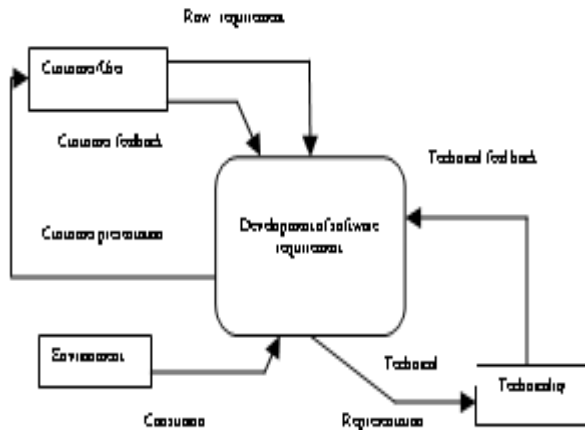and produces good and necessary requirement for software development.

Figure.3. Development of requirement

Requirement Analysis:

The analysis of requirement is the basic activity of any organization in building quality software products. These requirements are then rigorously analyzed within the context of business requirement. We have also observed that the identified raw requirement may be conflicting in nature. Therefore, the negotiation, agreement, communication and prioritization of the raw requirement become important activities of requirement analysis. The analyzed requirement needs to be documented to enable communication with stakeholders and future maintenance of requirement. The software requirement analysis process covers the complex task of eliciting and documenting the stakeholder's requirement. Documenting, analyzing and modeling user's requirement can be a basis for system design. Software requirement analysis and documentation processes are critical to software project development. Also, it can be used to refine the allocation of requirement for the system to allocate requirement in various system or sub-systems elements.

Allocation and Flow-Down of Requirement:

The major purpose of requirement allocation and flow-down is to ensure that all system requirement are fulfilled by a sub-system or set of subsystems that are performed together to achieve system objectives. Top-level system requirement should to be organized hierarchically that can assist to view and manage information at different levels of system. The requirement are decomposed down to the system level at which the requirement can be designed and tested. Thus, allocation and flow-down may be performed for several hierarchical system levels. The level of detail increases as the work proceeds down in the hierarchy. System-level requirement are general in nature. The requirement at low-level in the hierarchy are very explicit. The top-level system requirement defined in the system requirement development phase, which is the main input for the requirement allocation and flow-down phase. As the system-level requirement is being developed, the elements which are defined in the hierarchy should also be considered for requirement analysis.

Documentation of Requirement:

A formal requirement document is prepared after collecting requirement, which contains a complete description of the external behavior of a software system. Requirement development process is the activity of determining, which functionality of the system will be performed by software. Non-functional requirement are combined together with functional requirement into the software requirement specification with the help of flow-down and allocation of requirements. The documentation of requirement includes requirement identification along with requirement specification. A software requirement specification (SRS) is established for each software subsystem, software configuration item, or system component.

Requirement Identification:

Requirement identification practices focus on the assignment of a unique identifier for each requirement. These unique identifiers are used to refer requirement during product development and management. Requirement identification process consists of three sub-activities, namely; basic numbering, identification, and automotive support. The basic numbering activity includes significant numbering and non-significant numbering, whereas, identification activity includes labeling, structure based identification and symbolic identification. The automotive support is used to support and automate the management of system items, which includes dynamic renumbering, database record identification and baselining requirement.

Requirement Specification:

The SRS is produced after the successful identification of requirement. It is an effective document for requirement specification, which is a complete description of the behaviour of the system or software to be developed. Generally, SRS is a comprehensive description of the intended purpose and environment for software under development. The SRS minimizes the time and effort required by developers to achieve desired goals and also helpful in minimizing the development cost. A good SRS defines how an application will interact with system hardware, other programs and user in a wide variety of real-world situations. the parameters such as, operating speed, response time, availability, portability, maintainability, footprint, security and speed of recovery from adverse events are also evaluated in SRS.

Requirement Verification and Validation:

Once the entire requirements are specified in the SRS, the requirement agreements are finalized by involving different parties. Validation and verification activities include validating the system requirement against raw requirement and verifying the correctness of system requirement documentation. The most common techniques for validating requirement are requirement reviews with the stakeholders, and prototyping. Software requirement are validated against system-level requirement. Verification of SRS ensures correctness, consistency, unambiguousness and understandability of requirement. In requirement verification and validation, requirement traceability mechanism can generate an audit trail between the software requirement and finally tested code. Traceability should be

maintained at system-level requirement and architectural work products. The outcome of the software requirement development phase is a formal document including a baseline of the agreed software requirement.

Requirement Management and Planning:

The requirement management and planning phase controls and tracks the changes of agreed requirement, relationships between requirement, and dependencies between the requirement documents etc. The requirement management planning is a continuous and cross-section process that begins from identification and change control during requirement development process. In addition, it is a continuous activity, which can be performed after final development and during maintenance. Requirement traceability is a part of requirement management, which describes and follows the life of requirement and its relations with other development tools in both forwards and backwards direction. The continually changing requirement is required to be managed, because these are often changed time to time, which causes excessive damage. The changing requirement can be managed by applying requirement change management process. It refers to the ability to manage changes to requirement throughout software development lifecycle. Change management includes identifying, analyzing, deciding on whether a change will be implemented and the possibility of implementing and validating requirement change requests.

Requirement management is sometimes considered as the most multifaceted RE process. A requirement change can have a large impact on the developing system, which is very difficult to estimate. For every change, costs and re-development work must have to be considered before approving the change. After refinement of requirement, the requirement can further be incorporated into software development processes. Requirement management tools have been developed to manage unstable requirement and large amount of data collected during requirement engineering process. Requirement management tools collect system information together with the system requirement in a database or repository and provide a range of facilities to access the information about the requirement. Software requirement management tool supports secure and concurrent co-operative work between members of a multidisciplinary development team that support standard systems, modelling techniques and notations. It is necessary to maintain an audit trail of changes, archive baseline versions, and engage a mechanism to authenticate and approve change requests. One should also maintain a comprehensive data dictionary of all project components and requirement in a shared repository and produce predefined and ad-hoc reports. Documents should also be generated that can comply with standard industrial templates.

The process model is implemented in various software projects by the post-graduate students who were working as work group in their final project. In addition, the model is also applied in a various management systems. Finally, we have observed that after adopting this model, development cost and time is relatively reduced. Also, the successful implementation of proposed RE process can have a good impact on the production of quality software product. The proposed RE model can be used in larger software development process for continuously changing requirement. The organization can allow user for any further change in requirement during software development process. The SRS can be modified accordingly and this changed requirement can be re-implemented in software development process.

Security Requirement Engineering (SRE):

Designing a secure software system is challenging issue for the software designers. The malicious user always tries to break the systems and in response, software vendors started providing security as a necessary feature for their products and systems. The various powerful techniques have been developed to solve a wide array of security problems in the past years. While choosing security measures, the security system designer consider the design of the entire system, and they do not incorporate security technologies at random. Generally, developers apply security mechanisms such as, password protection, firewalls, virus detection tools, etc. at the implementation stage of software development. These security mechanisms are rarely considered as requirement at the beginning. As a result, security requirement that are specific to the system and that provide protection of essential services and assets are often neglected.

Although, security requirement have given significant attention in recent years but it is lacking the context for operation. Therefore, security RE process is presented in Figure 4, which incorporates the concept of security requirement in RE. The RE aspect includes functional and non-functional goals as well as requirement constraints, whereas security RE covers risk analysis and management, which has four dimensions, namely; security requirement, security goals, threat, and assets. This process includes security mechanism in non-functional requirement during establishing the relationship between security requirement and the specification of software performance. The ultimate goal of proposed process is to secure assets from threat, which can harm the assets. Threats can be inverse of security goals, which are operationalised by the security requirement. The aim of this research is to integrate security RE into RE process, so that secure software product can be produced [40].

Further, we have found that managing risks in security RE is a great challenge in RE research. One of the key to a good alignment between business domain and security of IT structures is to keep the focus on the assets of the business and keep secure them from various risks. Therefore, a security engineering process is proposed as shown in Figure 5, which incorporates threat modelling in RE [40]. Threat modelling involves understanding the complexity of the system and identifying all possible threats to the system, regardless of whether or not they can be exploited. The proposed security engineering process comprises of three activities, namely; risk identification and modelling, selection of security requirement, and development of security mechanism.

During the formation of security requirement, threats can be analyzed based on their complexity and a decision is made whether to mitigate the threat or accept the risk associated with it. Each stage of security engineering provides feedback to the preceding stage and all earlier stages. Feedback allows designers to catch mistakes made in early stages without letting their effects cascade. Threat modelling and security requirement provide the foundations upon which the rest of the security system is built. Identifying threats helps to develop realistic and meaningful security requirement. The proper identification of threats and appropriate selection of countermeasures reduces the ability of attackers to misuse the system. The security RE process is further explored in Figure 6 to involve the organisational models i. e., business model and structural model. Business model incorporates company assets such as, customer information, user account management, etc. Whereas, the structural model integrates the IT assets such as, hardware, software, server, etc.
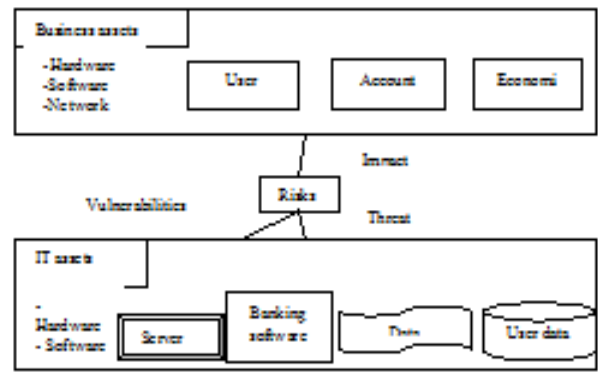

Figure. 4. SRE framework depicts the two major elements such as,(requirement engineering concepts) and (risk analysis and management)

Risk is potential problem that can come from the business process and can threaten or damage the IT assets of organisational structural model [40]. Security engineering system plays an important role to mitigate risk by providing some security measures during the structural modeling of business organization [41].

Figure. 5. Threat modeling framework includes (threat identification and modeling)(security requirement) (develop security mechanism)

Figure 6. Risk management framework involves (business assets), (it assets) and (risks)

Information Requirement Engineering (IRE):

Information is one of the important requirements for software development to generate quality and updated software. Typically, the specification of information commences with observing and interviewing people at different levels. The collected information flows across the management spectrum in decision making for software development. But, there is still non existence of such models in RE, which can provide such kind of support to organization. Therefore, we have presented a new approach for Information Requirement Engineering (IRE), which can assist software developers to gather proper and valuable information for decision making. The proposed IRE approach incorporates some perception such as, sources of information, information processing actors, and their responsibilities which is shown in Figure 7. The presented IRE approach has already been described in more detail in our previous research article [42].
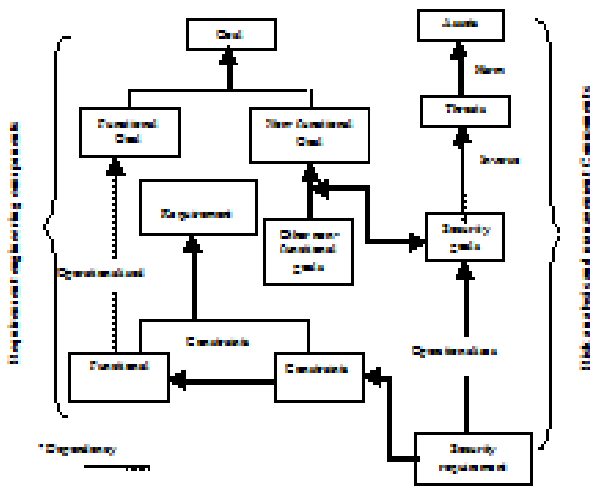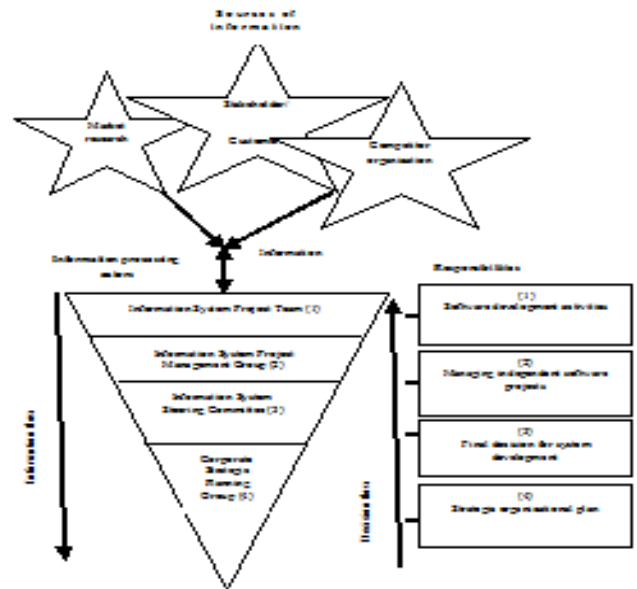

Figure. 7. IRE approach incorporates (information processing actors) and their (responsibilities)

The resources of information are repositories of facts that can be used by the information processing actors. The actors are responsible user of organization having the organizational responsibilities of information processing and decision making. They can process and analyze the information according to the requirement of management

and different user groups or stakeholders. The requirement of analyzed information is an important pre-requisite for software development. The management constantly decides about development of the system on the basis of analyzed information. Also, the management can decide the most important system is developed first, and then the less important subsystems with the help of proposed IRE.

The process of analyzing obtained information at different level of management spectrum is examined, which may lead to positive change/ modifications in the organization. The management spectrum for information engineering comprised of various information processing actors such as, project team, project management group, steering committee, and strategic planning group. These actors assist management to take decisions on software development. The main objective of project team is to collect information from various sources and allocate it to the next level i.e., information system designers. Information system project management group is responsible for managing independent projects. The information system steering committee of senior management is mainly authorized for system development. Finally, the key actor in our approach is corporate strategic planning group, who design strategic plans and make decisions. Also, they provide overall strategic direction to project team and explain the importance of information system to project stakeholders.

Measurement analysis of RE process:

The measurement of RE process can increase the user participation and also improve the quality of software products. The presented research analyzes the performance of proposed requirement engineering process model with the existing models. In literature, the Maculay and Kotonya - Summervillay RE process models have already been proposed to gather quality requirement. The Kotonya-Summervillay model has suggested a conceptual linear RE process model, which indicates iterations between some activities such as, elicitation and specification [19]. Also, the activities in this model are overlapped each other and often performed iteratively. The Maculay model provides a purely linear RE process, which is unable to indicate the overlapping or iterations of activities [45].

Although, the activities are performed step by step, but the order of the activities becomes irregular. Also, the proper management of requirement is not performed in aforesaid models. The proposed model depicts that the RE process is linear and iterative in nature [39]. It consists of various RE phases without overlapping. In the proposed model, any of its phases is started when the previous phases is completely performed. Structured interviews were conducted with the help of qualitative questionnaire to find out the performance measurement between proposed model and the existing models. For this, questionnaires were selected in order to collect the large amount of data required to gain a comprehensive understanding of the RE process. The questionnaire consists of the background details of the participant, company and project details, the RE process, and RE techniques (Table 1 and Table 2).

Many closed-ended questions were also used to minimize the length of the questionnaire. Open-ended questions were used, where it was important for participants to answer in their own words. The participants were asked to complete the questionnaire and add any additional verbal information. The use of interview sessions allowed participants to clarify meanings of the terms and questions used, ensuring that they had a clear understanding of what was being examined. The duration of the interviews was about 30 minutes with the differences in time resulting from the amount of additional information added verbally by the participant and the use of contingency questions in the questionnaire.

Table 1 Survey of projects in company A and company B

Project	Project member	Effort per month	Technical background of project members	Special quality requirement	Customer access	Non-RE tool support
α1	30-115	621	60%	High performance	Easy	Med.
α2	15-29	28	35%	Legal compliance	Easy	Low
α3	5-15	5	78%	High performance	Easy	Low
β1	10-15	114	95%	System availability	Easy	Med.
β2	5-15	12	65%	High performance	Normal	Low
β3	10-15	26	95%	System availability	Easy	Med.

The interviews took place at prearranged times in private meeting rooms at the companies' premises to maintain a quiet, undisrupted locations, which was consistent for every interview. A total of 10 interviews have been conducted at 2 different companies, operating in the manufacturing and financial industries.

Table 2 RE activities at company A and company

Project	Project creation	Elicitation	Negotiation	Verification and validating	Change management	Tracing
α1	No	Explicit	Implicit	Explicit	Explicit	Implicit
α2	Explicit	Explicit	Implicit	Implicit	Implicit	Implicit
α3	Implicit	Explicit	Implicit	No	No	No
β1	Implicit	Explicit	Explicit	No	Implicit	No
β2	Explicit	Explicit	Explicit	Explicit	Implicit	Implicit
β3	Implicit	Explicit	Explicit	Explicit	Implicit	Explicit

The participants were primarily in a project management or business analyst role, with responsibility for RE activities on the relevant project. Thus, the proposed RE process model is

compared with existing models and we have observed that proposed model emphasizes on every RE process. Also, it is observed that neither linear model nor purely iterative model is able to perform every activities of RE process (Table 3). In this way, the presented RE process model can benefit from a combination of linear and iterative structures [43]. The outcome is based on the result from case study of three different projects, which is executed in two aforesaid companies.

Table 3 Performance Measurement of RE Processes,i.e., Kotonya(K*), Macullay (M*) and Pandey(P*)

Project	RE in project lifecycle	Model structure	RE model used	Performance
α1	Continuous	Iterative	(K*)	Average performance
α2	Start	Linear	(M*)	Average performance
α3	Continuous	Iterative	(P*)	Good performance
β1	Continuous	Linear then iterative prototype	(K*), (P*)	Good performance
β2	Start of each iteration	Linear	(K*), (M*)	Average performance
β3	Continuous	Linear then iterative prototype	(K*), (P*)	Good performance

Requirement Modelling Framework:

Most of the requirement documents are written in ambiguous natural languages, which are less formal, and imprecise. Also, the requirement are generally gathered in textual format that may be vague, indistinguishable, and sometimes, it may be changed by overwrite. Without modeling the requirement documents, the acquaintance of the requirement is difficult to understand. The lack of suitable framework for modeling requirement is one of the important issues in this research. In academic community, researchers have proposed various detailed and focused requirement development models. But most of these models resulting from academic research are too typical for practical application and solve only some specific issues. Therefore, a requirement modeling framework is depicted in Figure 8 for providing the clear view of user and stakeholder's requirement. It consists of five major phases, namely; feasibility study, requirement collection and specification, analysis of business requirement, system requirement modeling, and system design. Further, analysis of business requirement includes business conception and association, business object life cycle, and business tasks and methods. The system requirement modeling incorporates actors, use cases and their scenarios [44].

Feasibility Study:

Feasibility study aims to describe the strengths and weaknesses of the existing business or proposed system. It also defines the resources required to the proposed system and the prospects for its success. There are two main criteria to judge the system feasibility, i.e., the cost requirement and another is the value to be attained. As such, a well designed feasibility study should provide a historical background of

the business or project, description of the product or service, accounting statements, details of the operations and management, marketing research and policies, financial data, legal requirement, and tax obligations. Generally, feasibility studies precede requirement development and project implementation.

Requirement Elicitation and Specification:

Requirement elicitation and development phase mainly focuses on examining and gathering desired requirement. Also, it focuses on objectives for the system from different perspectives [224]. Requirement elicitation phase begins with identifying stakeholders of the system and collecting raw requirement from various stakeholders. The elicitation phase aims to elicit the requirement from various viewpoints and business domain such as, business requirement, customer, user, constraints, security, and business standards etc. Furthermore, the specification of system requirement starts by observing and analyzing requirement by interviewing with the stakeholders. The user requirement are often misunderstood because the system analyst sometimes unable to understand the user's needs.

In addition, standards and constraints are also play an important role in systems development and their specification. Therefore, we have observed that the main objective of RE process is to collect requirement from customer and business environment in a systematic manner. The system analyst collects the unprocessed requirement and performs detailed analysis based on system requirement. Thereafter, these outcomes are then compared with the system architecture for final specification. The SRS document is used to specify the user requirement. It describes the system to be developed rather than its procedure. In addition, it includes a set of use cases that describe all the interactions that user will have with the system or software.

Analysis of Business Requirement:

Many organizations have already established their procedures and methodologies for conducting business requirement analysis, which may have been optimized specifically for the business organization [21]. However, the main activities for analyzing business requirement are identifying business conception and association, determining business object life cycle, and identifying business tasks and methods.

Business Conception and Association:

There exist various methodologies for business conception and association techniques but still these are conflicting in initialization of the process. In the study, we have analyzed that the starting point should be business requirement analysis and their relationships with other system elements. In the business conception and association model, system analyst can apply simple organizational working model using UML classes with names and without more detailed information, associations with stakeholders name and their role. Such models are discussed by business analysts and domain experts, who are usually unfamiliar with object-oriented analysis and design. Hence, all the other elements of the model such as, aggregations, compositions, generalizations, interfaces, enumerations etc., should not be

used for conceptual analysis. It enables UML novices to understand the process after getting a little explanation. Additionally, analyst can provide textual descriptions for each of these concepts and generate printable or navigable domain vocabularies. It is therefore believe that it should be the root artifact of the process, which should be used for defining other requirement model elements, use-cases etc.

Business Object Life Cycle:

Requirement models can be used either in requirement gathering process or during systems analysis. Eliciting requirement is considered as a separate activity, or a part of systems analysis. The importance of correct requirement must be a high priority for system development. Building accurate requirement models means the guarantee of the correctness of requirement. Requirement models can also be used to discover the functional and data requirement for software and business systems [44]. Additionally, the requirement model can be used as specifications for the designers and developers of the system. Organizations have their own business rules for managing business objects. In many cases, business rules regulate how important business objects change states. Requirement modeling can be one of the important tools to understand these changes [44].

The states also serve as a part of terminology, which can be used in other business and requirement models. State machine diagrams should be created only for those business concepts that have dynamic states. Business modelers should define triggers on all transitions in state diagram. In business modeling for transition triggers, most people use informal signals that in most cases correspond to actions of business roles. Also, time and property change triggers are used to express various states changes according to time or data based business rules. Therefore, there is possibility to define inner triggers that happen inside one state, and it does not alter the transition [44].
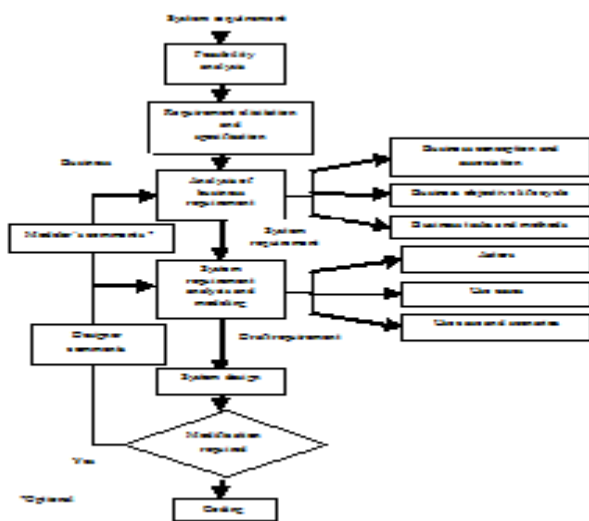

Figure 8 Requirement modeling framework depicts (requirement elicitation and specification) (analysis of business requirement)(System requirement analysis and modeling)(System design)

Business Tasks and Methods:

After learning domain terminology and business rules concerning lifecycles of business objects, analyst can identify business tasks and methods, and associate roles to

processes in which they are involved [45]. In this process, designers need to visually separate it from system actors and use cases. The business roles association to business processes can perform best within the specialized use case diagram or editable relationship matrix. The first step in moving from business analysis to requirement definition is use case analysis [44].

The business processes are usually modeled in two ways, i.e., "as is" and "to be" [44]. "as is" represents current situation, and "to be", represents target situation that should be reached after automation or refactoring [10]. For software developers, identification of various parts of target business, which is to be implemented at user's site, is much important to define business tasks and method.

System requirement analysis and modeling:

Requirement modeling is an important activity in the process of designing and managing requirement structures. The requirement modeling helps requirement engineers to understand structure and assist in analyzing the way of business requirement, which are related to information technology requirement and vice versa thereby they can facilitate the business-IT alignment in an organization. Basically, it includes actors, use cases, and scenarios, which is discussed in following sub-sections.

Actors:

An actor is a responsible user or external entity, which can interact with the system or products. At the same time, an actor is external to a system that interacts with the system. An actor may be a human user or another system, and has some goals and responsibilities to satisfy in interacting with the system. It is also necessary to generate actor, who has given compact overview of the whole model. In addition, preparing the requirement specification model is also necessary that incorporates the package details diagram showing package use cases, their associations with actors and, relationships between use cases. For modeling good requirement, system engineer prepares activity diagrams, which can visualize the scenarios of complex use cases. In requirement model, the activities should be nested within appropriate use cases and assigned as their behaviors. Finally, various actors involved in requirement modeling can be described in use cases according to pre-defined templates, i.e., rational unified process, use case document, and actors function etc.

Use case and scenarios:

The use case is a description of a potential series of interactions between software requirement model and external stakeholders. A use case diagram in the UML is a type of behavioral diagram defined and created from use case analysis. The purpose of use case is to present a graphical overview of the functional requirement, provided by a system in terms of actors, their goals, and any dependencies between use cases. Also, it is useful to describe the system performance to each actor. Generally, requirement models are used to captures functional requirement that the end-user needs from the system. The other requirement such as, non-functional requirement or detailed functional requirement is unable to capture in standard requirement modeling diagrams. The simplest way

is to describe these requirement in simple textual format and include references to use cases, their scenarios etc [44].

Another approach is to create specific requirement modeling to introduce stereotypes for important requirement along with tags consisting of requirement specific information and defines types of links for tracing requirement such as, derive, satisfy, and support. The data structure definition, on which system analysis has been performed, can also be included in requirement modeling. The object diagrams can also be used in class diagram for defining samples for explanation or testing of data structure. Unlike conceptual analysis, some more elements are also used in use case scenario such as, attributes and association, final specifications, enumerations, and generalization. Although such model is considered to be part of design, which can often created and maintained by system analysts. For data-centric applications, it is important to perform data flow diagrams, which is showing information flows between different classifiers, i.e., system-context diagram. It indicates information flows from system to outside world entities, i.e., actors or external system that need to be integrated [44].

The previous requirement modeling artifacts for which system analyst might be responsible were user interface prototypes. The prototype itself can theoretically be mapped to UML composite structure diagram. However, while focusing on separate screen prototypes, people sometimes loose the screens, which can be used by each actor and there is the possibility to navigate from each screen to the other screens. For capturing this information, GUI navigation map can also be created. Finally, we have emphasized that the requirement analysis work can be presented in iterative and incremental models thereby ambiguity in stakeholders requirement can be eliminated. Also, the requirement modeling tasks might be different, using different approaches [44].

System Design:

After the successful completion of requirement analysis and modeling phase, the draft requirement may be provided to the design team. Design team checks the validity of the draft requirement and starts to design the system or software model. Basically, system design is the process of designing, developing and implementing the proposed system as per the requirement obtained during the analysis of existing system. The main objective of the system design is to develop the best possible design as per the requirement from user and working environment for operating the software system. System design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirement. The UML in object oriented analysis and design is widely acceptable for the modeling software requirement and it is increasingly used for designing non-software systems. After designing of the system model, designer evaluates the efficiency of the design model. If any modification remains in the model, designer again checks the validity of requirement and asks for correction with comments. The process will stopped until the clear cut clarification is not received by the design team.

Many organizations already have developed their own procedures and methodologies for conducting business requirement analysis, which may have been optimized specifically for the business organization. In the aforementioned framework, the main activities for analyzing business requirement are identifying business conception and association, determining business object life cycle, and evaluating business tasks and methods. Similarly, the factors such as, identification of stakeholders, capturing the stakeholder's requirement, categorizing, interpreting, and recording the requirement are helpful to create requirement models. In addition, the implementation of requirement modeling for various requirement analysis purposes, and mapping of conventional requirement model into system elements is presented.

DISCUSSION

We have presented various requirement engineering approach, i. e, effective requirement engineering process for requirement gathering and management, SRE, IRE, etc. , which can be beneficial for software industries, for developing secure, updated and exciting software products. By applying these approaches, and keeping in view of aforementioned RE prctices, industries can overcome their technical problems related to the requirement gathering, requirement management and requirement security.

CONCLUSION

RE is an imperative activity of software development process for designing and developing the quality software products. In this paper, the social, organizational and technical issues are discussed, and at the same time, impact of security in SRS document is also observed. The effective RE process model is proposed, which incorporates iterative philosophy for better requirement analysis and processing. The proposed process model is helpful for later maintenance as well as for requirement prototyping with more user input. Further, the Security Requirement Engineering (SRE) process is designed and illustrated with risk management to develop secure software products. In addition, Information Requirement Engineering (IRE) approach has been developed for better information gathering and decision making. Furthermore, the performance measurement of proposed RE process model is presented with their usefulness in the business organizations and for the community of information system user. Also, a framework for requirement modeling is designed to understand the diverse views of stakeholders thereby vagueness in requirement can be minimized.

In conclusion, the aforesaid RE prctices can be used to make effective RE process, which can be useful in software development process to produce the quality software products.

REFERENCES

[1].   P. Jalote, An integrated approach to Software Engineering, Second Indian Edition, Narosa Publishing House, New Delhi, 2010.

[2]. S. Wahono, Analyzing Requirement engineering Problem, IECI Japan workshop, Japan, (2009) 55-58.

[3]. B. Nuseibeh and S. Easterbrook, Requirement engineering: a roadmap, In Proc. of the IEEE Int. Conf. on Soft. Eng. ICSE (2010) 35–46.

[4]. D. L. Parnas, Software engineering is not computer science programs, Ann. Soft Eng., 6(1) (2009) 19– 37.

[5]. Williams and Kennedy, A framework for improving the Requirement engineering process effectiveness, School of Computing, Information System and Management, London, 2007.

[6]. The Standish Group. The CHAOS Report. Denish, MA: The Standish Group, 2004.

[7]. S. L. Pfleggeer, Software engineering Theory and practices, Prentice Hall, New Jersey, 2008.

[8]. D. J. Reifer, Requirement Management: The search for Nirvana", IEEE Software (2010) 45-47.

[9]. Baumert and Whinney, Software Measure and the Capability Maturity Model, Software Engineering Institute Technical Report, 2002.

[10]. Cutis et. al, A field Study of the software Design Process for large system design, Communication of the ACM, Vol. 31, No. 11 (2003) 1268-1287.

[11]. Ehman, Khaled and Madhavji, A field Study of Requirement Engineering Practices in Information System Development, Proc. of the Second IEEE international symposium on RE, York, England (2004) 68-80.

[12]. Siddiqi et. al., Requirement engineering: The emerging wisdom, IEEE software, vol. 13, No. 2 (2010) 15-19.

[13]. Nuseibh, Bashar and Steve, Requirement engineering: a road map, Proc. of the 22nd International conference in advanced Information Engineering, New York, ACM (2010) 35-36.

[14]. Sawyer and Pete, Package Software: Challenges for RE, Proc. of the 12th International Conference in advanced Information System Engineering, Sweden, Springer (2011) 137-142.

[15]. Eman and Madhavji, A field study of RE Practice in Information System development, Proc. of RE'05, New York (2005) 68-70.

[16]. Lubas, Potts and Richter, A review of the state of the practice in requirement modeling, Proc. of RE'93, San Diego (2008) 2-14.

[17]. S. Sindre, and A. L. Opdahl, Templates for misuse case description, Proc. of the 7th International workshop on RE, Switzerland (2011), 23-31.

[18]. Kusiak and Tang, Innovation in requirement life cycle framework, Proc. of the 5th int. symposium on intelligent manufacturing sys, London (2010) 61- 67.

[19]. G. Kotonya, and S. Ian, Integrating safety analysis and Requirement engineering, Proc. on the 4th Asia pacific software engineering and international computer science conference, China (2011) 259-271.

[20]. IT- Software process assessment– part 7: Guide for use in process improvement, Technical Report, Switzerland, 2009.

[21]. K. Wiegers, Software Requirement, Microsoft Press, USA, 2009.

[22]. Kauppinen et. al., Implementing Requirement engineering process throughout Organizations", success factor and challenges, Elsevier Science, reprinted with permission from information and Software Technology (2004) 937-953.

[23]. B. Curtis, Software process improvement: methods and lessons learned, Proc. of the 9th International Conference on Software Engineering, (ICSE 1997), China, (2009) 624–625.

[24]. Diaz, M., and Sligo, J., How software process improvement helped Motorola, IEEE Software, Vol. 14, No.5 (2011) 75–81.

[25]. B. McFeeley, IDEAL: a user's guide for software process improvement, Handbook MU/SEI-96-HB-001, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PE, USA, 2006.

[26]. S. Zahran, Software Process Improvement: Practical Guidelines for Business Success, 5th Addison-Wesley, Harlow, England, 2008.

[27]. K. Wiegers, Software Requirement, Microsoft Press, Redmond, WA, USA, 2009.

[28]. P. Zave, Classification of research effort in RE, ACM Computer Survey (2007) 313-314.

[29]. C. Kuloor and A. Eberlian, Requirement engineering for software products, The University of Calgary, Canada (2011) 1-12.

[30]. Palayger Bhavani, Measuring and influencing requirement engineering process quality in Organization, information and communication sciences, Mcqurie University, Australia, 2009.

[31]. K. Wiegers, The real world of requirement engineering, Requirenautics Quarterly (2004) 50-58.

[32]. H. Hecht and M. heccht, How reliable are requirement for reliable software?, Microsoft Press, Redmond, WA, USA, 2004.

[33]. Sommerville and G. Kotonya, Requirement engineering process and Techniques, 1st ed., EnglandWiley, 1998.

[34]. C. S. Kuehl, Improving system requirement quality through application of an Operational Concept Process: an essential Element in System Sustainment, Microsoft Press, Redmond, WA, USA, 2000.

[35]. Dhirendra Pandey, Ugrasen Suman, and A. K. Ramani, Issues and Impact of Requirement Engineering Practices in Designing Quality Software Products, Proc. of National Conference on Architecturing Future IT System, DAVV, Indore (MP) (2011) 44-46.

[36]. Dhirendra Pandey, Ugrasen Suman, and A. K. Ramani, Role of Interviews in Requirement Gathering, Proc. of National Conference on Intelligent Computing & Information System, D. P. Vipra College, Bilaspur (CG) (2009) 26-31.

[37]. Dhirendra Pandey, Ugrasen Suman, and A. K. Ramani, Social-organizational Participation Difficulties in Requirement Engineering Process, Software Engineering Journal, BioInfo Publication, Mumbai, Vol. 1, No. 1, ISSN: 2229–4007 & ISSN: 2229–4015 (2010) 01-05.

[38]. Dhirendra Pandey, Ugrasen Suman, and A. K. Ramani, Design and Development of Requirement Specification Documents for Making Quality Software Products, Proc. of National Conference on Intelligent Computing & IS, D. P. Vipra College, Bilaspur (CG) (2011) 43-47.

[39]. Dhirendra Pandey, Ugrasen Suman, and A. K. Ramani, An Effective Requirement Engineering Process Model for Software Development and Requirement Management, Proc. of International Conference on Advances in Recent Technologies in Communication and Computing, ARTCom' 2010, IEEE Computer Society, 2010, ISBN: 978-0-7695-4201-0, Kerala (2010) 287-291.

[40]. Dhirendra Pandey, Ugrasen Suman, and A. K. Ramani, Security Requirement Engineering Framework for Developing Secure Software, International Journal of Computational Intelligence and Information Security, IJCIIS, Australia, Vol. 1 No. 8, ISSN: 1837-7823 (2010) 55-65.

[41]. Dhirendra Pandey, Ugrasen Suman, and A. K. Ramani, Security Requirement Engineering Issues in Risk Management, International Journal of Computer Applications, Published by Foundation of Computer Science, USA, ISBN: 978-93-80747-89-4, Vol. 17, No. 5 (2011) 11-14.

[42]. Dhirendra Pandey, Ugrasen Suman, and A. K. Ramani, "An Approach to Information Requirement Engineering", Proc. of IEEE International Conference on Information Security and Application, ICISA, 2011, ISBN: 978-1-4244-9223-7/11, South Korea, 2011.

[43]. Dhirendra Pandey, Ugrasen Suman, and A. K. Ramani, "Performance Measurement of Different Requirement Engineering Process: A study", International Journal of Computer Engineering and Technology, IJCET, Tamilnadu, ISSN: 0976-6375, Vol. 1, No. 2 (2010) 286-300.

[44]. Dhirendra Pandey, Ugrasen Suman, and A. K. Ramani, "Framework to Modelling Software Requirement", International Journal of Computer Science Issues, IJCSI, Malaysia, ISSN: 1694-0814, Vol. 8, No. 3 (2011) 164-171.

[45]. L. A. Macaulay, Requirement Engineering, Springer-Verlag, 2006.

[46]. J. H. Flavell, Metacognition and cognitive monitoring: A new area of cognitive-developmental inquiry, American Psychologist Journal (2011) 906–911.

Short Bio Data for the Auhtor
Dhirendra Pandey is a member of IEEE and IEEE Computer Society. (Born on June 28, 1982 at Bilaspur, India) Mr. Dhirendra has received his MPhil Degree in Computer Science from Madurai Kamraj University, Madurai, Tamilnadu, India. Presently, Mr. Dhirendra perusing PhD in Computer Science from School of Computer Science & Information Technology, Devi Ahilya University, Indore (MP). The major research area of Mr. Dhirendra is Software Engineering, Data Mining and Warehousing and Artificial Intelligence.

He is working in Babasaheb Bimrao Ambedkar University, Lucknow as Assistant Professor in the Department of Information Technology. He has more than 7 year teaching and 5 year research experience. Prior to this job he worked at various institutions as Assistant Professor, i.e, SSBAM College, Ambikapur, india, PNS college, Bilaspur, India.he has published more than 15 research papers, which are indexed in IEEE, DOAJ, etc.

He is approaching to investigate the effectiveness of requirement engineering process in software development process. He is the active member of the many National and International professional bodies viz. IEE, CSI, IAENG, UPCST, Member Board of Studies in Computer Scienec, Surguja University, Surguja, etc

Dr. Ugrasen Suman has received his PhD degree from School of Computer Science & Information Technology (SCSIT), DAVV, Indore. Presently, he is a Reader in SCSIT, Devi Ahilya University, Indore (MP). Dr. Suman is engaged in executing different research project in SCSIT. He has authored more than 30 research papers.

Professor (Dr.) A. K. Ramani (is no more) has received his ME and PhD Degree from Devi Ahilya Vishwavidyalaya, Indore (M.P.). Dr. Ramani has authored more than 100 research papers and executing several major research projects. Presently, he is the Head of the Department in SCSIT, Devi Ahilya University, Indore (MP).