Volume 3, No. 3, March 2012
Journal of Global Research in Computer Science
ISSN-2229-371X
REVIEW ARTICLE
Available Online at www.jgrcs.info

PROCESS SUPPORT FOR REQUIREMENTS ENGINEERING

Onkar Nath Pandey [*1], D.B.Ojha[2]
* Research Scholar, CMJ University
pandeyonkarnath@yahoo.co.in [1]
[2]Department of Science& Technology, Mewar University, Rajasthan, INDIA
ojhabrat@gmail.com[2]

Abstract: This paper presents reasons of the project failures. Various elements of system are discussed. Application of RE tools are explained to resolve the problems. Subsequently, RE tool development and evaluation is also explained.

INTRODUCTION

Information systems development is a complex activity which comprises operating information, allocating material and human resources and managing all of them usually with a computerized software system. Kotonya and Sommerville (1998) define an information system as primarily concerned with processing information which is held in some kind of databases. Such systems are usually implemented using computer hardware and are built on top of commercial operating systems [1]. There are several arguments for distinguishing between software and systems. However every large project involves hardware, networks, people and procedures to follow, in other words systems of some kind (Alexander and Stevens, 2002). In most projects the system requirements engineering primarily focus on software requirements engineering [2].

Requirements define what the system is required to do and the circumstances under which it is required to operate. Requirements engineering (RE) is considered to be one of the most important stages in the whole system and software development process [3]. Errors done during the RE process could be very expensive in the later development stages and during the system maintenance and use.

DEFINITIONS AND PRELIMINARIES

Project Failure Analysis:

Alexander and Stevens (2002) and Leffingwell and Widrig (2000) refer to the Standish Group and EXPITI (European Software Process Improvement Training Initiative) studies which identify three most common reasons for the project failure [4] are:

a. lack of user involvement,
b. incomplete requirements and requirements specifications, and
c. changing requirements and specifications.

The techniques to support and to improve the process could be characterized in three different ways, like work avoidance (reuse of methodology and techniques in development and reuse of software artefacts [5], e.g., requirements and requirements specifications), working smarter (apply new RE methods to execute the process, develop techniques for training users, support maintenance of the software products, work out techniques for requirements negotiation, development of standards both for RE process and documents), and working faster (use software tools, e.g., using targeted RE tools to support the RE process) [6]. This work addresses RE process support and improvement by emphasizing working faster and suggesting a methodology to acquire RE –tools in order to automate the RE process. Boehm (1999) discusses that introduction of new software tools could improve the productivity by 8 percents (in comparison to application of new methods – 17 percents and reuse of prepared artifacts – 47 percents). Eight percent of improvement is already a significant result for the RE process improvement. On the other hand introduction of new tools require organizational changes, and in most cases it means adoption of new engineering methods. Furthermore, software tools facilitate reuse of already developed artefacts across related domains by suggesting knowledge repositories and linking them across different projects [7].

RE Tools:

Requierments engineering (RE) tools are software tools which provide automated assistance during the RE process and support the RE activities (Matulevicius, 2004b) [8]. The literature and vendors of these tools usually call these do support different requirements engineering activities such as requirements elicitation, requirements documentation and analysis, requirements validation are requirements specification. Therefore they are call as RE-tool in this work. The need for automated support may vary in different projects; and if a company does not have a mature RE process, automation won't necessarily help us there are other basic process improvements that should be done first. On the other hand if the company deals with system requirements specifications containing many requirements which need to evolve over time, RE-tool support could clearly be useful (Kotonya and Sommerville, 1998; Kaindl et. Al 2002, Matulevicius, 2004c) report that the mainstream RE practice relies on office tools(e.g. text editors, drawing and modeling tools) rather than targeted RE-tools (e.g. DOORS, CalierRM, Requisite Pro, DOORS and CORE) provided by various companies or research groups. Reasons for not using RE-tools include financial causes, like high RE-tool price low return on investment. Companies consider it to be difficult to adapt RE tools to their organizational needs. Many software

companies are not aware that there might be significant gains in taking up advanced tool support. A part of the reason might be that it is difficult to evaluate the available RE tools. Hofmann and Lehner (2001) stress that a lack of well defined RE process and a lack of team training in the selected tools caused the insufficient support for the RE activities.

Adoption:

In order to adopt a tool, an infrastructure must be set to support the tool. A company must be willing to invest in putting such an infrastructure in the organization (E1 Emam and Madhavji, 1995). This includes personnel training, tool support groups, funding for the tool implementation. However, the management of software companies usually have unrealistic expectations, as, for example, immediate pay-off. Because of their limited use in practice it is difficult to evaluate RE-tools in terms of their impact on an organization's processes. Similarly, it is difficult to examine tools in an experimental situation, as it is difficult to control for the variation in system developers capabilities. Moreover, RE tools provide the greatest benefit for large projects with stakeholders who frequently change their minds about requirements, while a controlled experiment normally requires prescribed tasks of a fairly limited size.

Evaluation:

It would be hard to create experimental tests that would provide a realistic evaluation of the tools and for small and medium-size organizations the cost of thus evaluating several RE-tools empirically might be prohibitive [9]. There is also a need for a cheaper kind of evaluation that can be done analytically rather than empirically. For instance, RE tools can be evaluated from a theoretical point of view – using information provided by vendors. They can be tried out on some realistic examples, but without the rigour of a controlled experiment. A potential problem of such evaluations, however, is that they easily become ad-hoc and subjective [10]. Hence, to support the completeness and effectiveness of such evaluations, they should be grounded in sound evaluators. The objective of this work is to develop an RE-tool acquisition method, which would help to elicit the environment needs in order to adapt the RE-tool(s) in an inexpensive way and in a short amount of time.

RESULT AND DISCUSSION

Methodology:

Mathematics, Science and engineering have special historical relationships with computing, so different research methods may be used. Denning (2000) classifies three major research approaches: theory, experimentation and design. All three research approaches constantly interact in the process of research. Theoretical approach involves building of conceptual frameworks and notations for understanding relationships among objects in a domain and the logical consequences of axioms and laws. Theory characterizes the analytical method when a formal theory is proposed and then compared with empirical observations [11]. Experimentation is a process of exploring models of systems and architecture within the given application domains and testing whether those models can predict new behavior accurately [12]. Experimentation describes an empirical method when a model is proposed and evaluated through empirical studies, for example, case studies and experiments. Design characteristics constructing of systems that support work in given organizations or applications domains. Design describes and engineering method, which studies the solutions and evaluates them [13]. Generally the research process involves three fundamental questions to be raised:

a. What is the problem that is being focused on, and why is this important? The question explains a need to search for understanding, for a sense of having found a satisfying explanation of some aspects of reality.
b. What is the best way to approach a solution to this problem? The question describes how the understanding is achieved by means of statements of general laws and principles – laws applicable to the widest possible variety of phenomena.
c. How can the proposed solution be validated (i.e., was the problem solved)? The question analyses if the laws or principles can be tested experimentally.

The research of this work addresses RE process improvement. The research objective targets the development of the RE-tool evaluation approach which contributes to selection of the RE-tools in order to improve both the RE process and product. In order to analyze the defined research questions, both descriptive and prescriptive research methods are used; and they include literature study, survey (Dillman, 2000), case studies, and experiment (Wohlin et al., 2002), leaving aside field study and action research (Sankran, 2001). One could argue that application of the RE-tool assessment method especially in an organization that is planning to acquire an RE-tool would be valuable and useful; however the analytical literature study and the empirical survey (Matulevicius, 2004a) of software development organizations revealed some major problems in conducting field studies and/or action research. It was difficult to find an industrial organization, which would be interested in participating in such an investigation. The organizations are not interested and not willing to automate the RE process. A part of the problem is that often their RE process is immature and needs further consideration. Therefore, all together the research phases in this work include:

a. An extensive analytical literature study. Post-mortem analysis of software development projects with focus on RE is followed with a survey research (Dillman, 2000) in targeted geographical areas. The analysis highlights the existing RE problems and the RE process difficulties and shows weak automated support of the RE process.
b. Both analytical and descriptive studies contribute to the proposal of the conceptual framework for evaluating the RE-tools before acquisition to the environmental settings, theoretical approach to the RE tool assessment which is summarized to the RE-tool evaluation approach (R-TEA).
c. The validation of the proposed method comprises both experimental empirically-based (Wohlin et al., 2002) and design research approaches. A number of case studies in order to explore different conceptual elements of R-TEA method. Building a prototype and testing the

proposed R-TEA method in the scope of Chapter 8. Two prototypes are implemented in order to target validity issues from a design point of view.

The first prototype tool supports the proposed R-TEA method itself. The second prototype describes an experimental-design environment for analysis of poorly supported features of existing commercial RE-tools.

CONCLUSION

It requires organizational changes, and in most cases it means adoption of new engineering methods. Furthermore, software tools facilitate reuse of already developed artifacts across related domains by suggesting knowledge repositories and linking them across different projects.

REFERENCES

[1] Hands K., Peiris D. R., & Gregor P. (2004). Development of a Computer-Based Interviewing Tool to Enhance the requirements Gathering Process, Requirements Engineering, 9, 204-216..

[2] Harrison W., Ossher H., & Tarr P. (2000). Software Engineering Tools and Environments; A Roadmap. A.Finkelstein (Ed.) The Future of Software Engineering. ACM Press.

[3] Hatley D.J., & Pirbhai I.A. (1998). Strategies for Real-Time System Specification. Dorset House.

[4] Hirschheim R. A., & Klein H.K. (1989). Four Paradigms of Information Systems Development. Communications of the ACM, 32 (10), 1199-1219.

[5] Hofmann H.F., & Lehner F. (2001). Requirements Engineering as a Success Factor in Software Projects. IEEE Software, 58-66.

[6] Hoffmann M., Kuhn N., Weber M., & Bittner M. (2004). Requirements for Requirements Management Tools. Proceedings of the IEEE Joint International Conference on Requirements Engineering (RE'04). Kyoto, Japan, 301-308.

[7] Humprey W.S. (1989). Managing the Software Process. Addison – Wesley. IEEE (1998) IEEE Recommended Practice for Software Requirements Specification, IEEE Std 830-1993 (Revision of IEEE Std 830-1998).

[8] INCOSE (2002). INCOSE: Tools Survey: Requirements Management (RM) Tools by International Council on Systems Engineering (INCOSE) [online]. Available: http://www.incose.org/.

[9] ISO/IEC (1991). Information Technology – Software Product Evaluation – Quality Characteristics and Guide lines for their Use. ISO/IEC IS 9126, Switzerland.

[10] James L. (1996). What's Wrong with Requirements Management Tools? Requirements Engineering, 1, 190-194.

[11] Kaindl H., Brinkkemper S., Bubenko jr. J., Farbey B., Greenspan S. J., Heitmeyer C.L., do Prado Leite J. C. S., Mead N.R., Mylopoulos J., & Siddiqui J. (2002). Requirements. Engineering and Technology Transfer: Obstacles, Incentives and Improving Agenda. Requirements Engineering, 7 (3), 113-123.

[12] Haywood E., & Dart P. (1997). Towards Requirements for Requirements Modelling Tools. Proceedings of the 2nd Australian Workshop on Requirements Engineering, Australia, 61-69.

[13] Kaindl h. (2004). Active Tool Support for Requirements Engineering through RETH. Proceedings of the 12th IEEE International Requirements Engineering Conference (RE2004),262-263.

Short Biodata of All the Author

Dr.D.B.Ojha: Ph.D from Department of Applied Mathematics, Institute of Technology, Banaras Hindu University, Varanasi (U.P.), INDIA .The degree field is Optimization Techniques In Mathematical Programming. The major field of study is Functional Analysis. He is working at Mewar Universit, Rajasthan as Professor. He has the authored/coauthored of more than 200 publications in technical journals and conferences.

 Mr.O.N.Pandey did his B.Tech.(IIT, Kanpur)1968, ME(IIT,Roorkee)1970. His specialization in Instrumentation, Control and Industrial Automation under UNIDO Fellowship from USA, UK, ITALY and JAPAN and also membership of UNIDO fellowship, Numerical Control Society.