



POSITIVE TRENDS IN REQUIREMENT ENGINEERING PRACTICES FOR HIGHER SOFTWARE QUALITY

Dr. Rajinder Singh*

Abstract : *In this competitive world, customer satisfaction is the utmost important thing for any organization to survive. Customer satisfaction merely depends upon the quality of the product delivered to him. Software industry has seen a tremendous growth in last few years. Success of Software industry depends upon the cost, schedule and quality of the software being produced. Many of the software projects fail due to being delayed, over budget or due to non-conformity to quality standards. Quality products can be produced only if requirements of the client is understood rightly and at right time. Understanding of the requirement of the client is very crucial for any organization. A lot of research has been done to find out effective requirement engineering processes so as to produce quality products on time and on budget. Theories say that quality of any software directly depends upon how effectively RE processes have been followed by the software development team. If requirements are identified clearly at initial stage then it leads to high standard of quality software project. So requirement engineering processes and software quality are co-related and go side by side.*

The aim of this study is to check how much RE processes affects software productivity by conducting a survey in nine different software development companies and taking evidence from the software developers who are actually using these practices practically in their jobs.

Keywords : Requirement Engineering, Requirement Implementation, RE Processes, Software Quality

*Department of Electronics, S.D College, Ambala Cantt



1. INTRODUCTION

In this competitive world, customer satisfaction is the utmost important thing for any organization to survive. Customer satisfaction merely depends upon the quality of the product delivered to him. Software industry has seen a tremendous growth in last few years. Success of Software industry depends upon the cost, schedule and quality of the software being produced. Many of the software projects fail due to being delayed, over budget or due to non-conformity to quality standards. Quality products can be produced only if requirements of the client is understood rightly and at right time. Understanding of the requirement of the client is very crucial for any organization. A lot of research has been done to find out effective requirement engineering processes so as to produce quality products on time and on budget. Theories say that quality of any software directly depends upon how effectively RE processes have been followed by the software development team. If requirements are identified clearly at initial stage then it leads to high standard of quality software project. So requirement engineering processes and software quality are co-related and go side by side.

The aim of this study is to check how much RE processes affects software productivit by conducting a survey in nine different software development companies and taking evidence from the software developers who are actually using these practices practically in their jobs. For this study , I have asked the questions to **23** respondents from **9** software development companies of Ambala , Gurgaon ,Pune, and Chandigarh . All these respondents are engaged in software development as developer or manager.

This paper is divided in five sections. Section I gives introduction, Section II describes Requirement Engineering , Section III gives introduction of Software Quality , Section IV states the objective of study, Section V gives details of Questions ,results and their analysis followed by conclusion.

2. REQUIREMENT ENGINEERING:

RE can be simply defined as identifying a problem's context, locating the customer's requirements within that context and delivering a specification that meets customer needs within that context. There are many requirements methodologies that purport to do this, for example, soft systems methodology [1], scenario analysis [2], and UML [3]. Sometimes they work, sometimes they do not. The implication of such requirements methodologies, if



we can label at least aspects of them as such, is that the application of 'x' method will produce the right requirements irrespective of the problem's characteristics.

This is conventional wisdom and unsurprisingly, the creators and vendors of requirements methodologies claim, with one exception [4] that their approach is a hammer and all problems are nails. While there are many factors other than just application of a requirements methodology that influence the quality of software projects in practice, in this paper I focus only on requirements engineering.

2.1 Processes of Requirement Engineering:

Requirement Engineering Process is divided in to two main categories [5]:

- a. **Requirement Gathering** : Requirement gathering is divided in four phases :
 1. Requirement Eliciting
 2. Analyzing Requirements
 3. Specifications of Requirements
 4. Validating requirements
- b. **Requirement Implementation**: Requirement implementation is actual execution of the requirements in the software development phases.

2.1.1 Elicitation of requirements can be done using five different techniques [6]

- i) **Traditional techniques:**
Traditional methods comprise of technique of gathering data using questionnaires, surveys, interviews, task analysis, domain analysis and Introspection.
- ii) **Cognitive techniques** : In this technique requirement engineer collect and prioritize requirements. Some of the cognitive techniques are Repertory grids, card sorting, laddering and protocol analysis.
- iii) **Group elicitation techniques** : This technique involves eliciting requirement through the involvement of team or groups of software engineers. Group works, brainstorming, JAD requirement workshops and protocol analysis relate to group elicitation techniques.
- iv) **Prototyping** is the technique which is used for elicitation purpose when requirements are not clear or when urgent stakeholders' feedback is required to proceed further.



- v) **Contextual techniques** involves ethnography, conversation analysis and observations/social analysis that serve as an alternative to the traditional cognitive techniques.

2.1.2 Requirement Implementation/ development:

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

- a) Existing system information
- b) Stakeholder needs
- c) Organizational standards
- d) Regulations
- e) Domain information

It also generates three outputs, namely agreed requirements, system specification and systems models. This process is general and flexible as for all the organizations only the requirements can differ, but these inputs and outputs always remain fixed [5,6].

Linear Requirements Engineering Process Model, envisaged by Linda Macaulay, is a simple model, primarily used for administering small projects. This model is composed of five tasks in sequences :

1. Conceptualization
2. Problem analysis,
3. Feasibility study,
4. Analysis and Modeling,
5. Requirement documentation [5].

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

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



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

3. SOFTWARE QUALITY:

In the context of software engineering, software quality measures how well software is designed (quality of design), and how well the software conforms to that design (quality of conformance),[7] .

As stated in Wikipedia Software quality may be defined as conformance to explicitly stated functional and performance requirements, explicitly documented development standards and implicit characteristics that are expected of all professionally developed software.

The three key points in this definition:

1. Software requirements are the foundations from which quality is measured.
Lack of conformance to requirement is lack of quality.
2. Specified standards define a set of development criteria that guide the manager in software engineering.
If criteria are not followed lack of quality will usually result.
3. A set of implicit requirements often goes unmentioned, for example ease of use, maintainability etc.
If software confirms to its explicit requirement but fails to meet implicit requirements, software quality is suspected.

4. OBJECTIVE OF STUDY :

Theories say that RE practices helps producing quality software projects. The objective of this study is to explore the impact of requirement engineering practices on Software quality. Software industry is devoting large amount of funds towards the development of software thus increasing the cost of final project. Loss to the software company due to non-conformity of quality standards may be direct (financial) or indirect (loss of fame , goodwill or business). This loss can be eliminated if RE processes are followed and requirements are



clear, no rework is done and enough time has been spent to gather and analyze requirements of customer. Aim of this study is to take the evidence from the IT professionals about the impact of RE processes on Software quality.

RESEARCH METHODOLOGIES

The aim of this paper is to check whether the RE practices lead to better software productivity or not. If such practices are thoroughly studied, they could be used for enabling production of quality projects.

For this paper, I have conducted survey on some of Indian Companies and tried to find out the relation between RE practices and Software quality.

To prepare the evidence to check the impact of Requirement engineering tools on Software quality , a Questionnaire is prepared and is filled by the authorized employees of the companies such as Infosys, Cognizant Technology (Pune),Market RX(Gurgaon), One World Technology(Ambala), Ameotech Informatics(Chandigarh), GENPACT, GTech Informatics, Automatic Data Processing India Pvt. Ltd, Silex Softwares Pvt. Ltd.(Ambala).

After collecting the data from these companies, analysis of the data is done using frequency tables and graphs tools of SPSS Software.

The sample size used in this study involved **23** software development projects from **nine** companies of Pune, Gurgaon, Chandigarh and Ambala. Due to this reduced sample size, the use of qualitative research methods was preferred. Furthermore, the main aim of this study is to formulate a hypothesis about the relationship between RE process and the Software Quality.

5. Questionnaire Results & Analysis:

I received completed questionnaires from number of respondents, reporting on **23** distinct projects. As noted earlier, the majority of our respondents were developers or project managers from Pune, Gurgaon, Chandigarh and Ambala based companies.The Survey questionnaire had mixed type of questions.

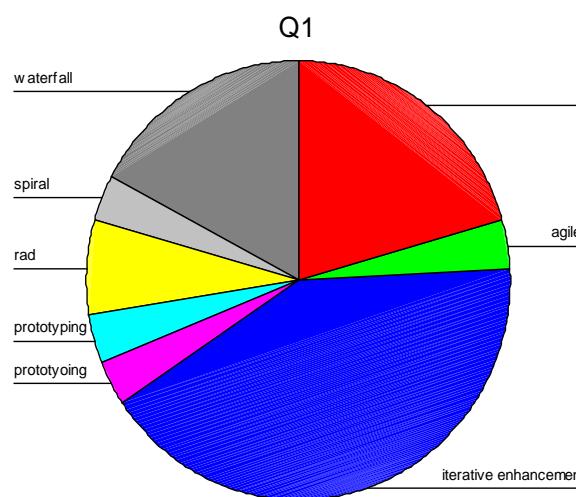
5.1 Questions & Responses:

Q1 which of the following process model is being followed by your company?

- Waterfall
- Prototyping
- Iterative enhancement

- Spiral model
- RAD model
- CBD model

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	agile development	1	4.3	4.3	4.3
	iterative enhancement	12	52.2	52.2	56.5
	Prototyping	2	8.7	8.7	65.2
	Rad	2	8.7	8.7	73.9
	spiral	1	4.3	4.3	78.3
	waterfall	5	21.7	21.7	100.0
	Total	23	100.0	100.0	



Analysis:

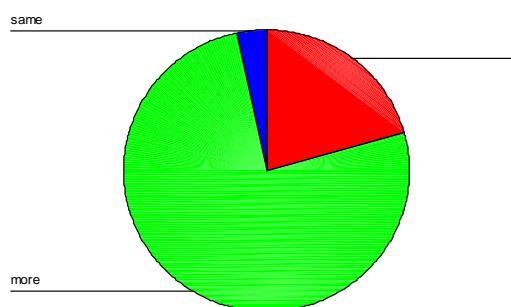
52.2% respondents have selected iterative enhancement model, 21.7% respondents have selected waterfall model, 8.7% have selected prototyping, 8.7% have selected RAD, 4.3% have selected agile and 4.3% have selected spiral model.

Q2 How the RE process affected requirements tracing of the project.

- Far More
- More
- Same
- Less
- Far Less

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid more	22	95.7	95.7	95.7
same	1	4.3	4.3	100.0
Total	23	100.0	100.0	

Q2

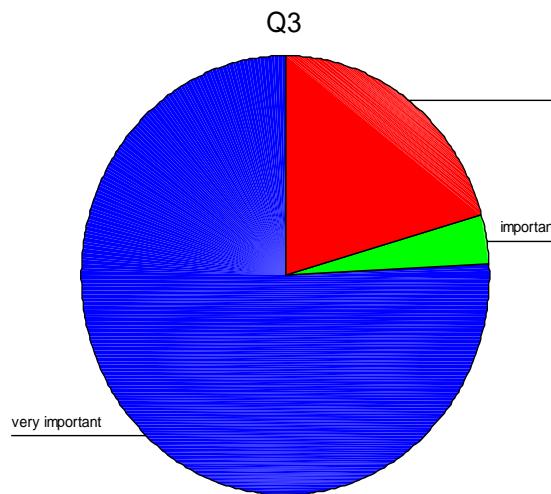
**Analysis:**

95.7% respondents have selected more option. It means 95.7% respondents believe that RE process affected requirements tracing of the project.

Q3 How important is the use of RE Processes in improving SW productivity?

- Very Important
- Important
- Unsure
- Not really Important
- Not Important at All

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid important	1	4.3	4.3	4.3
very important	22	95.7	95.7	100.0
Total	23	100.0	100.0	

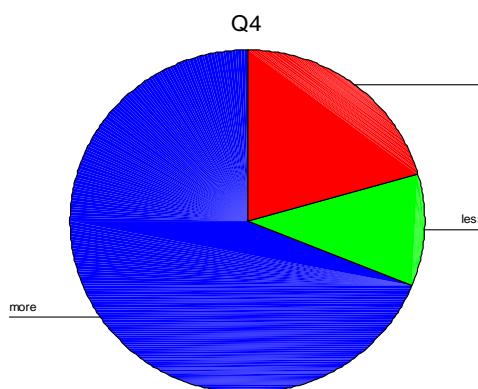
**Analysis:**

95.7% respondents have selected very important and 4.3% have selected important so 100% respondents believe that RE Processes plays an important role in improving SW productivity.

Q4 How do you believe the communication inspired by the requirements Analysis sessions improved or deteriorated productivity?

- Far More
- More
- Same
- Less
- Far Less

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	less	3	13.0	13.0	13.0
	more	20	87.0	87.0	100.0
	Total	23	100.0	100.0	



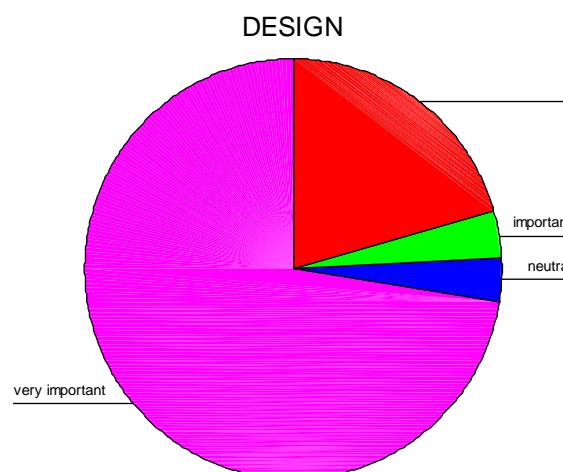
Analysis:

87% respondents have selected more option so 87% respondents believe the communication inspired by the requirements Analysis sessions improved or deteriorated productivity.

Q5 In your design, coding, testing or documentation activities, how important is it to understand the effect of RE processes on SW productivity.

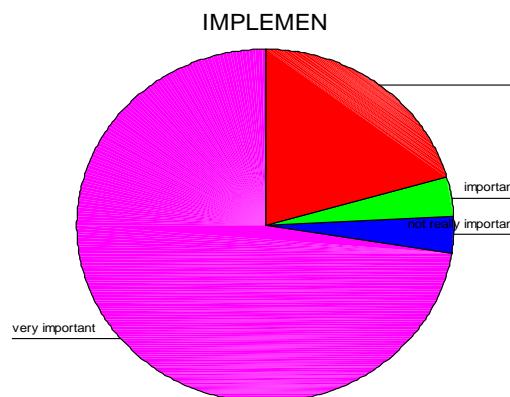
Design

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	important	1	4.3	4.3	4.3
	neutral	1	4.3	4.3	8.7
	very important	21	91.3	91.3	100.0
	Total	23	100.0	100.0	



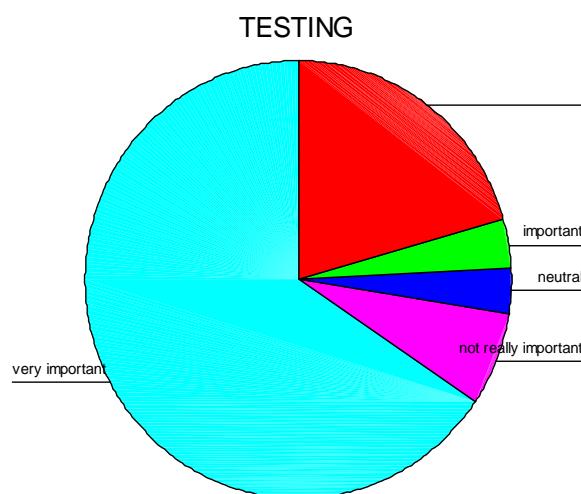
IMPLEMENTATION

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	important	1	4.3	4.3	4.3
	not really important	1	4.3	4.3	8.7
	very important	21	91.3	91.3	100.0
	Total	23	100.0	100.0	



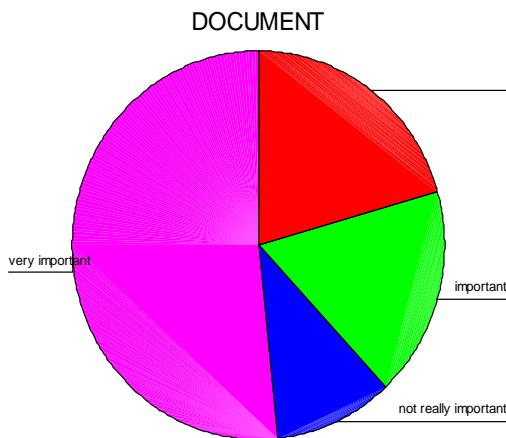
TESTING

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid				
important	1	4.3	4.3	4.3
neutral	1	4.3	4.3	8.7
not really important	2	8.7	8.7	17.4
very important	19	82.6	82.6	100.0
Total	23	100.0	100.0	



DOCUMENT

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid				
important	10	43.5	43.5	43.5
not really important	3	13.0	13.0	56.5
very important	10	43.5	43.5	100.0
Total	23	100.0	100.0	



Analysis:

In Design phase, 95.6% respondents considered RE processes Important and very important for Software quality.

In Implementation phase , 95.6% respondents considered RE processes Important and very important for Software quality.

In Testing phase , 86.9% respondents considered RE processes Important and very important for Software quality.

In Documentation phase , 87% respondents considered RE processes very important and important for SW quality.

CONCLUSION:

From above study, it is concluded that practically also most of the Software developers believe that if RE processes followed properly help produce quality projects. There is a positive relationship between RE processes and Software quality.

REFERENCES:

- [1] Checkland, P., Systems Thinking, Systems Practice, Wiley Publications, 1981.
- [2] Carroll, J. (ed.), Scenario-Based Design: Envisioning Work and Technology in System Development, Wiley Publications, 1995.
- [3] Booch, G., Rumbaugh, J., Jacobson, The UML User Guide, Addison Wesley, 1999.
- [4] Jackson, M., Problem Frames, Addison Wesley, 2001.
- [5] Arif.S., Khan. Q. & Gahyyur. S.A.K., (2009-2010). Requirement Engineering Processes, Tools/Technologies, & Methodologies, International Journal of Reviews in Computing (IJRIC), ISSN: 2076-3328, Vol.2
- [6] Nuseibeh, B. & Easterbrook, S., 2000. Requirements engineering: a roadmap C. Ghezzi, M. Jazayeri, & A. L. Wolf, eds. Context, 1(258), p.35-46.
- [7] Pressman 2005, p. 746