RESEARCH PAPER | Engineering | Volume : 4 | Issue : 5 | May 2014 | ISSN - 2249-555X

Impact of Requirement Engineering Processes on Software Development Cost

KEYWORDS: Requirement Engineering, Requirement gathering ,Requirement Implementation, RE Processes, Software Engineering

Rajinder Singh

Associate Professor, Department of Electronics,S.D College, Ambala Cantt

ABSTRACT *21st century has seen many technological developments. One of the industries which have seen tremendous developments is Software Development industry. Lot of research has been done in the field of software engineering to find out better models, processes and practices which can help in developing quality softwares on time and within the anticipated budget. But various studies have shown that high number of projects fails. The common problem faced by all software development companies is the rework which needs to be done after the completion of the project leading to increase in development cost of software. There are many factors which lead to rework but the most important one is the requirement elicitation. Requirements of the clients, if not precisely defined at the start of the project, there are more chances of rework. The aim of this study is to find out those practices and requirement engineering processes which if followed reduces the chances of rework thus leading to more chances of project success in terms of cost. In this study, I have studied the practices of 38 different projects taken from 9 companies in India and analyzed the impact of those practices on rework and software development cost.*

1. INTRODUCTION

21st century has seen many technological developments. One of the industries which have seen tremendous developments is Software Development industry. Lot of research has been done in the field of software engineering to find out better models, processes and practices which can help in developing quality softwares on time and within the anticipated budget. But various studies have shown that high number of projects fails. The common problem faced by all software development companies is the rework which needs to be done after the completion of the project leading to increase in development cost of software. There are many factors which lead to rework but the most important one is the requirement elicitation. Requirements of the clients, if not precisely defined at the start of the project, there are more chances of rework. The aim of this study is to find out those practices and requirement engineering processes which if followed reduces the chances of rework thus leading to more chances of project success in terms of cost. In this study, I have studied the practices of 38 different projects taken from 9 companies in India and analyzed the impact of those practices on rework and software development cost.

1.1 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 success or failure of software projects in practice, in this paper I focus only on requirements engineering.

1.2 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.

1.2.1 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.

1.2.2 Requirement Implementation/ development: The input/output of RE process, devised by Kotonia and Sommervile, 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,

Volume : 4 | Issue : 5 | May 2014 | ISSN - 2249-555X

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 Sommervile, 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.

2. Objective of Study :
Theories says that RE practices has direct impact on the success of any software project. The objective of this study is to explore the impact of requirement engineering practices on rework done in software development. Software industry is devoting large amount of funds towards the development of software thus increasing the cost of final project. Cost incurred due to rework is also included in the development cost which if can be eliminated can lead to reduction in the development cost of software. Knowing the common underlying problems that cause rework and identification of RE practices that reduces scope of rework will help software development teams avoid making those same mistakes over and over and making use of those practices that have more success rate. Researching the causes of several team projects that failed will provide insight for future IT team project development. It is inevitable that history will repeat itself if the history is unknown. This may cause disastrous and costly consequences.

3 Research Methodologies
The aim of this paper is to identify those RE Processes which help reducing the rework and development cost. If such practices are thoroughly studied, they could be used for enabling reduction of cost of the project.

For this paper, I have conducted survey on some of Indian Companies and tried to Put light on the practices leading to eliminate the rework which would provide us with the opportunity to assess the effects of Requirement Engineering over an entire project life cycle To prepare the evidence to check the Requirement engineering tools used in Indian Companies and to see the effect of RE on Rework and cost, a detailed 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). The questionnaire was divided in to four parts :
(1) Details and profile of the company
(2) Profile of the person filling the questionnaire
(3) Details of Project
(4) Details of RE techniques and processes used.

Many closed-ended questions were used to minimize the length of the questionnaire, however participants were offered an "Other-please specify" option to prevent forced answers from occurring.

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

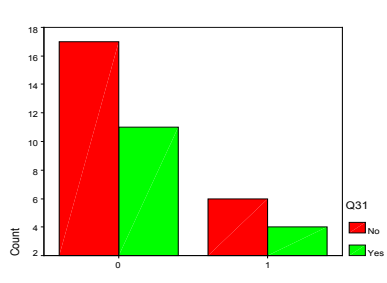
In order to understand the nature of RE Processes, a qualitative as well as quantitative approach is employed. The sample size used in this study involved 38 software development projects from nine companies of Pune, Gurgaon, Chandigarh and Ambala. Due to this reduced sample size, the use of qualitative research methods was preferred. Furthermore, the main aim of this study is to formulate a hypothesis about the relationship between RE process and the development cost.

4. Questionnaire Results & Analysis:
I received completed questionnaires from number of respondents, reporting on 38 distinct projects. As noted earlier, the majority of our respondents were developers or project managers from Pune, Gurgaon, Chandigarh and Ambala based companies. The responses to set of 38 questionnaires described 38 projects, 23 regarded as successful as no rework needed to be done and 15 unsuccessful as rework was done and the cost was increased. The Survey questionnaire had mixed type of questions.

Questions & Responses:
Q no 31 is Rework done or not leading to increase in development cost.
Q. System Development Paradigm used :
- Object Oriented
- Function Driven
- Data driven
- Evolutionary , Iterative
- Other

Object Oriented Paradigm

			Q31		Total
			No	Yes	
Q	0	Count	17	11	28
		% within Q3D_A1	60.7%	39.3%	100.0%
	1	Count	6	4	10
		% within Q3D_A1	60.0%	40.0%	100.0%
Total		Count	23	15	38
		% within Q3D_A1	60.5%	39.5%	100.0%

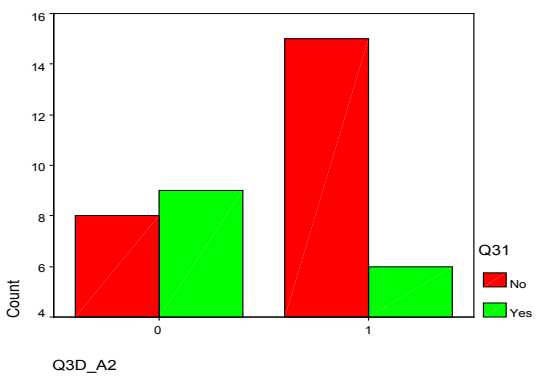


Volume : 4 | Issue : 5 | May 2014 | ISSN - 2249-555X

Analyses:
Out of the 10 projects which selected Object Oriented paradigm, in 6 projects no rework was done and in 4 cases rework was done and the cost was increased. So Object oriented paradigm has 60% success rate.

Function Oriented Paradigm

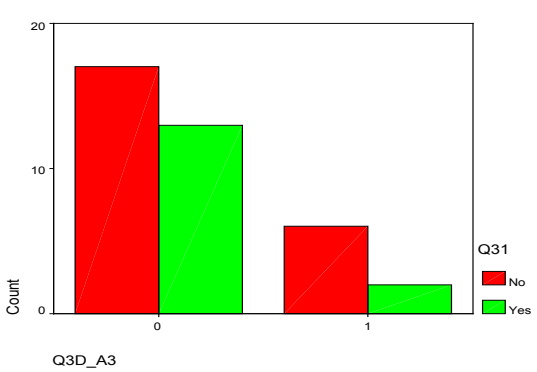
			Q31		Total
			No	Yes	
Q	0	Count	8	9	17
		% within Q3D_A2	47.1%	52.9%	100.0%
	1	Count	15	6	21
		% within Q3D_A2	71.4%	28.6%	100.0%
Total		Count	23	15	38
		% within Q3D_A2	60.5%	39.5%	100.0%



Analyses:
Out of the 21 projects which selected Function driven, in 15 projects no rework was done and in 6 cases rework was done and thus cost was increased. So Function driven paradigm has 71% success rate.

Data driven paradigm

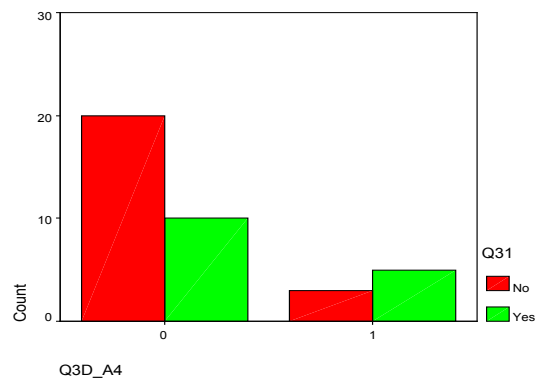
			Q31		Total
			No	Yes	
Q	0	Count	17	13	30
		% within Q3D_A3	56.7%	43.3%	100.0%
	1	Count	6	2	8
		% within Q3D_A3	75.0%	25.0%	100.0%
Total		Count	23	15	38
		% within Q3D_A3	60.5%	39.5%	100.0%



Analyses:
Out of the 8 projects which selected Data driven paradigm, in 6 projects no rework was done and in 2 cases rework was done and thus cost was increased. So Data driven paradigm has 75% success rate.

Evolutionary, Iterative Paradigm

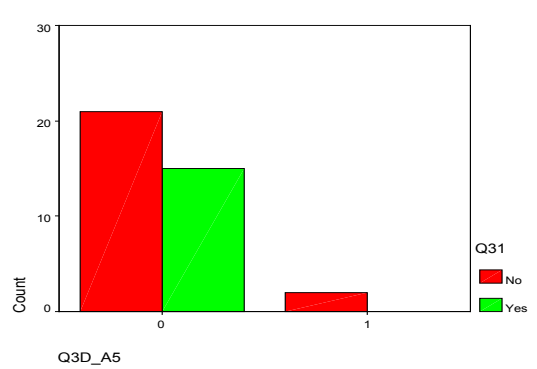
			Q31		Total
			No	Yes	
Q	0	Count	20	10	30
		% within Q3D_A4	66.7%	33.3%	100.0%
	1	Count	3	5	8
		% within Q3D_A4	37.5%	62.5%	100.0%
Total		Count	23	15	38
		% within Q3D_A4	60.5%	39.5%	100.0%



Analyses:
Out of the 8 projects which selected Evolutionary Iterative Paradigm, in 5 projects no rework was done and in 2 cases rework was done and thus cost was increased. So Evolutionary, Iterative Paradigm has 37.5% success rate.

Other Paradigms:

			Q31		Total
			No	Yes	
Q	0	Count	21	15	36
		% within Q3D_A5	58.3%	41.7%	100.0%
	1	Count	2	0	2
		% within Q3D_A5	100.0%	.0%	100.0%
Total		Count	23	15	38
		% within Q3D_A5	60.5%	39.5%	100.0%



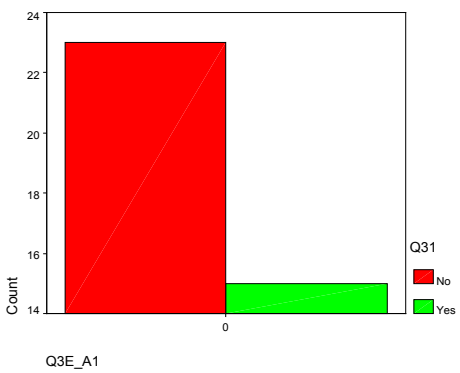
Volume : 4 | Issue : 5 | May 2014 | ISSN - 2249-555X

Analyses:
Out of the 2 projects which selected some other Paradigm, in both projects rework has been done and thus cost is increased. So in other than above mentioned paradigms there is 100% failure.

Q. Life Cycle Model Used :
- Spiral
- Waterfall
- Evolutionary ,Iterative , Prototyping
- Other , if yes mention it : _____

Spiral Life Cycle Model

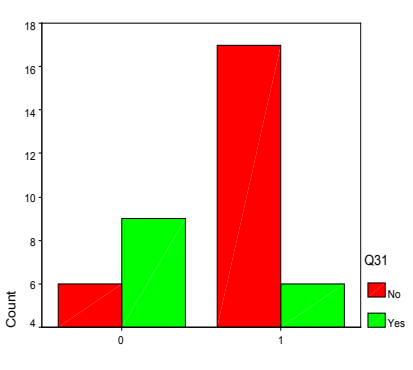
			Q31		Total
			No	Yes	
Q	0	Count	23	15	38
		% within Q3E_A1	60.5%	39.5%	100.0%
Total		Count	23	15	38
		% within Q3E_A1	60.5%	39.5%	100.0%



Analyses:
No one has selected spiral life cycle model. Out of 38 projects , in 15 projects rework was done and thus cost was increased.

Waterfall Life Cycle Model

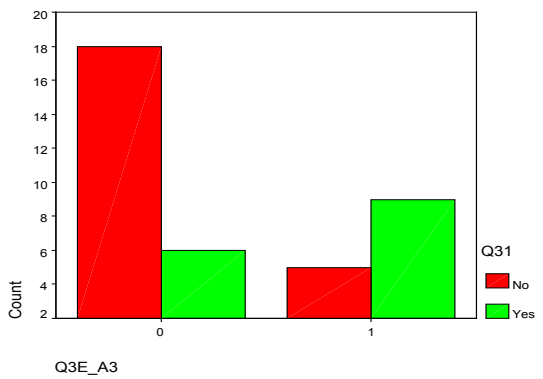
			Q31		Total
			No	Yes	
Q	0	Count	6	9	15
		% within Q3E_A2	40.0%	60.0%	100.0%
	1	Count	17	6	23
		% within Q3E_A2	73.9%	26.1%	100.0%
Total		Count	23	15	38
		% within Q3E_A2	60.5%	39.5%	100.0%



Analyses:
Out of the 23 projects which selected Waterfall Life Cycle Model, in 17 cases no rework was done and cost was as per schedule. Waterfall Life Cycle Model has 73.9% success rate.

Evolutionary, Iterative, Prototyping Life Cycle Model :

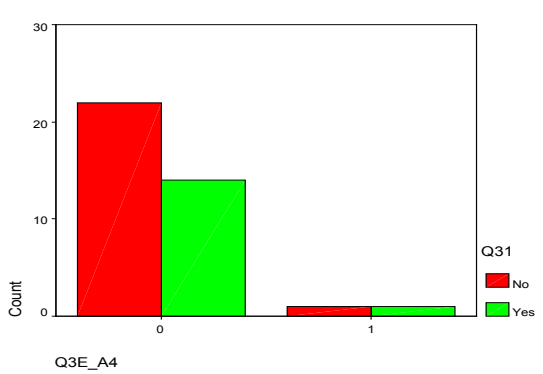
			Q31		Total
			No	Yes	
Q	0	Count	18	6	24
		% within Q3E_A3	75.0%	25.0%	100.0%
	1	Count	5	9	14
		% within Q3E_A3	35.7%	64.3%	100.0%
Total		Count	23	15	38
		% within Q3E_A3	60.5%	39.5%	100.0%



Analyses:
Out of the 14 projects which selected Evolutionary, Iterative, Prototyping Life Cycle Model, only in 5 cases no rework was done. Evolutionary, Iterative, Prototyping Life Cycle Model has 35.7% success rate.

Other Life Cycle Model:

			Q31		Total
			No	Yes	
Q	0	Count	22	14	36
		% within Q3E_A4	61.1%	38.9%	100.0%
	1	Count	1	1	2
		% within Q3E_A4	50.0%	50.0%	100.0%
Total		Count	23	15	38
		% within Q3E_A4	60.5%	39.5%	100.0%



Analyses:
In only 2 projects, others methods were used. In one case rework was done in other not done.

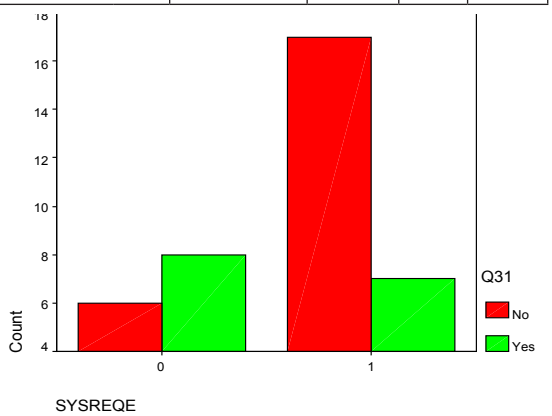
Volume : 4 | Issue : 5 | May 2014 | ISSN - 2249-555X

Q. Development Team Structure used :

- System Analysis ,Requirement Engineering
- Testing
- Technical Writing
- No specialists , all are developers

System Analysis, Requirement Engineering

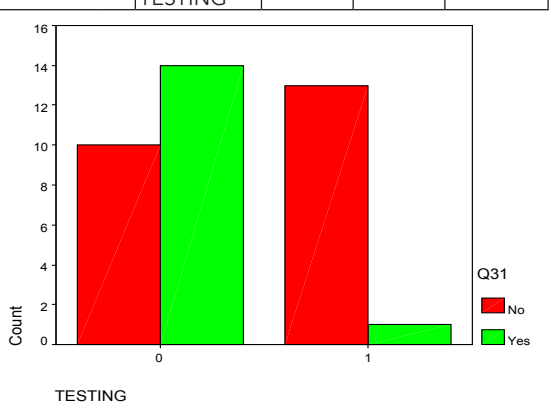
			Q31		Total
			No	Yes	
SYSREQE	0	Count	6	8	14
		% within SYSREQE	42.9%	57.1%	100.0%
	1	Count	17	7	24
		% within SYSREQE	70.8%	29.2%	100.0%
Total		Count	23	15	38
		% within SYSREQE	60.5%	39.5%	100.0%



Analyses:
Out of the 24 projects which selected System Analysis, Requirement Engineering as Development Team Structure , only 7 projects need to be reworked leading to success rate of 70.8%.

Testing as Development Team Structure:

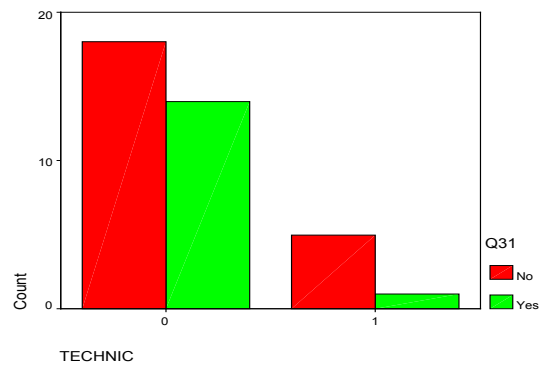
			Q31		Total
			No	Yes	
TESTING	0	Count	10	14	24
		% within TESTING	41.7%	58.3%	100.0%
	1	Count	13	1	14
		% within TESTING	92.9%	7.1%	100.0%
Total		Count	23	15	38
		% within TESTING	60.5%	39.5%	100.0%



Analyses:
Out of the 14 projects which selected Testing as Development Team Structure only 1 project need to be reworked leading to success rate of 92.9%.

Technical Writing as Development Team Structure

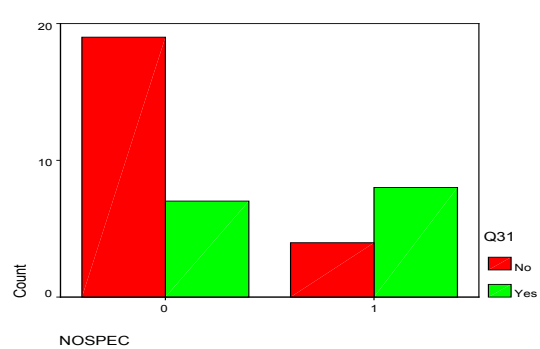
			Q31		Total
			No	Yes	
TECHNIC	0	Count	18	14	32
		% within TECHNIC	56.3%	43.8%	100.0%
	1	Count	5	1	6
		% within TECHNIC	83.3%	16.7%	100.0%
Total		Count	23	15	38
		% within TECHNIC	60.5%	39.5%	100.0%



Analyses:
Out of the 6 projects which selected Technical Writing as Development Team Structure , in 5 projects rework was done leading to success rate of 83.3%.

No specialists, all are developers

			Q31		Total
			No	Yes	
NOSPEC	0	Count	19	7	26
		% within NOSPEC	73.1%	26.9%	100.0%
	1	Count	4	8	12
		% within NOSPEC	33.3%	66.7%	100.0%
Total		Count	23	15	38
		% within NOSPEC	60.5%	39.5%	100.0%



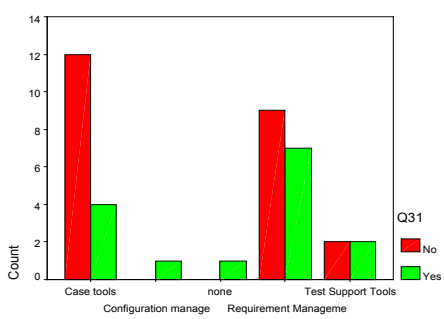
Analyses:
Out of the 12 projects which selected No specialists, all are developers, Development Team Structure only 8 project need to be reworked leading to success rate of 33.3.9%.

Volume : 4 | Issue : 5 | May 2014 | ISSN - 2249-555X

Q. Development Tools used:

- Test support tools
- Case tools
- Configuration management tools
- Requirement management tools

			Q31		Total
			No	Yes	
Q	Case tools	Count	12	4	16
		% within Q5	75.0%	25.0%	100.0%
	Configuration management tool	Count	0	1	1
		% within Q5	.0%	100.0%	100.0%
	none	Count	0	1	1
		% within Q5	.0%	100.0%	100.0%
	Requirement Management tool	Count	9	7	16
		% within Q5	56.3%	43.8%	100.0%
	Test Support Tools	Count	2	2	4
		% within Q5	50.0%	50.0%	100.0%
Total		Count	23	15	38
		% within Q5	60.5%	39.5%	100.0%



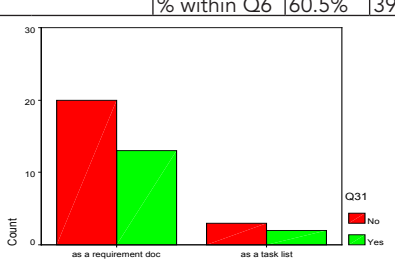
Q5

Analysis:
Case Tools has the maximum success rate as 75% and Requirement management tools are on second number with 56.3% success rate.

Q. Requirement document created

a) as a requirement document
b) as a task list

			Q31		Total
			No	Yes	
Q	as a requirement document	Count	20	13	33
		% within Q6	60.6%	39.4%	100.0%
	as a task list	Count	3	2	5
		% within Q6	60.0%	40.0%	100.0%
Total		Count	23	15	38
		% within Q6	60.5%	39.5%	100.0%



Q6

Analysis:
Requirement document created as requirement document reduces the rework as out of 33 projects which created requirement documents, in 13 projects rework is done leading to success rate of 60.6%.

Q. Requirement Document Structure

- Standard
- Flixible
- Simple , consistent , and concise

			Q31		Total
			No	Yes	
Q	Flexible	Count	18	5	23
		% within Q7	78.3%	21.7%	100.0%
	Simple	Count	2	6	8
		% within Q7	25.0%	75.0%	100.0%
	Standard	Count	3	4	7
		% within Q7	42.9%	57.1%	100.0%
Total		Count	23	15	38
		% within Q7	60.5%	39.5%	100.0%

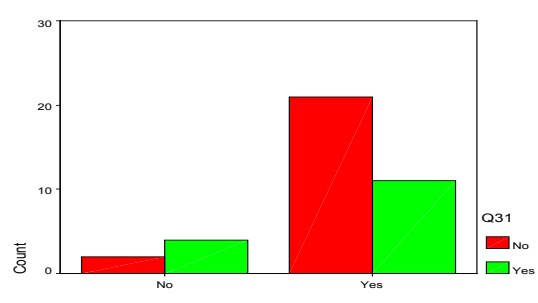
Analysis:
Requirement Document Structure flexible has maximum success rate of 78.3% as only 5 projects out of 23 needed to be reworked and cost was increased.

Q . General Guidelines

- Requirements management policies defined
  Yes No
- Document validation checklists defined
  Yes No
- Requirement analysis checklists defined
  Yes No
- Process defined
  yes No
- Problems analysed
  yes No

Requirements management policies defined

			Q31		Total
			No	Yes	
Q	No	Count	2	4	6
		% within Q10_S1	33.3%	66.7%	100.0%
	Yes	Count	21	11	32
		% within Q10_S1	65.6%	34.4%	100.0%
Total		Count	23	15	38
		% within Q10_S1	60.5%	39.5%	100.0%



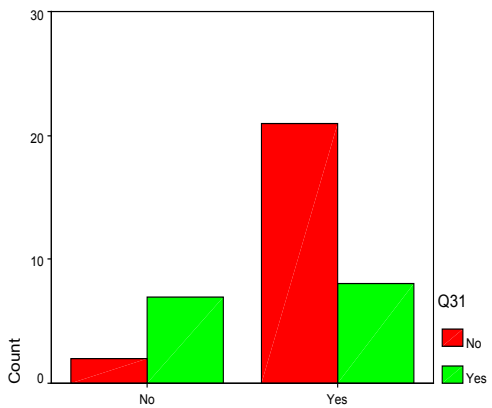
Q10_S1

Volume : 4 | Issue : 5 | May 2014 | ISSN - 2249-555X

Analysis:
Requirement management policies if defined leads to maximum success rate of 65.6%

Document validation checklists defined

			Q31		Total
			No	Yes	
Q	No	Count	2	7	9
		% within Q10_S2	22.2%	77.8%	100.0%
	Yes	Count	21	8	29
		% within Q10_S2	72.4%	27.6%	100.0%
Total		Count	23	15	38
		% within Q10_S2	60.5%	39.5%	100.0%

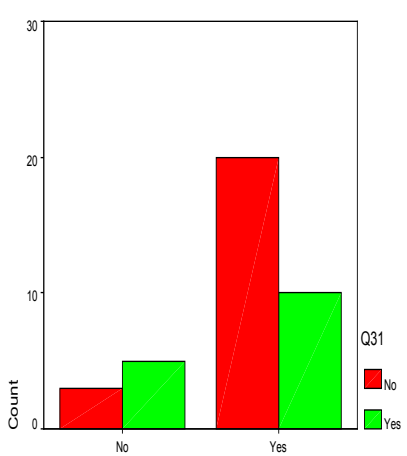


Q10_S2

Analysis:
Document validation checklists if defined leads to maximum success rate of 72.4%

Requirement analysis checklists defined

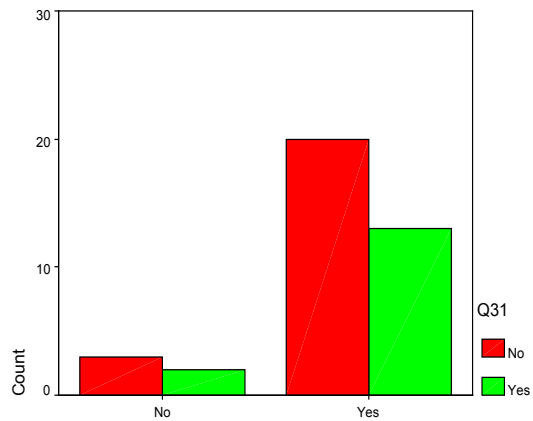
			Q31		Total
			No	Yes	
Q	No	Count	3	5	8
		% within Q10_S3	37.5%	62.5%	100.0%
	Yes	Count	20	10	30
		% within Q10_S3	66.7%	33.3%	100.0%
Total		Count	23	15	38
		% within Q10_S3	60.5%	39.5%	100.0%



Q10_S3

Process defined

			Q31		Total
			No	Yes	
Q	No	Count	3	2	5
		% within Q10_S4	60.0%	40.0%	100.0%
	Yes	Count	20	13	33
		% within Q10_S4	60.6%	39.4%	100.0%
Total		Count	23	15	38
		% within Q10_S4	60.5%	39.5%	100.0%

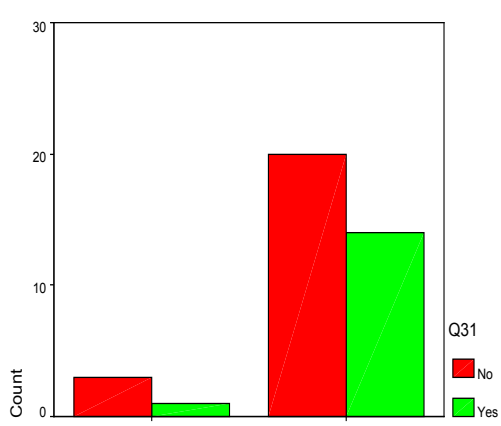


Q10_S4

Analysis:
Processes if defined leads to maximum success rate of 60.6%

Problems analyzed

			Q31		Total
			No	Yes	
Q	No	Count	3	1	4
		% within Q10_S5	75.0%	25.0%	100.0%
	Yes	Count	20	14	34
		% within Q10_S5	58.8%	41.2%	100.0%
Total		Count	23	15	38
		% within Q10_S5	60.5%	39.5%	100.0%



Q10_S5

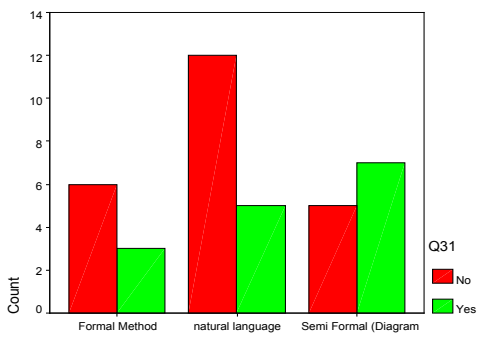
Analysis:
Problems if analyzed leads to maximum success rate of 58.8%

Volume : 4 | Issue : 5 | May 2014 | ISSN - 2249-555X

Q. Methods used

- Formal methods
- natural language
- Semi –formal (diagrams, psudocodes)

			Q31		Total
			No	Yes	
Q	Formal Method	Count	6	3	9
		% within Q11	66.7%	33.3%	100.0%
	natural language	Count	12	5	17
		% within Q11	70.6%	29.4%	100.0%
	Semi Formal (Diagrams, psedocode)	Count	5	7	12
		% within Q11	41.7%	58.3%	100.0%
Total		Count	23	15	38
		% within Q11	60.5%	39.5%	100.0%



Q11

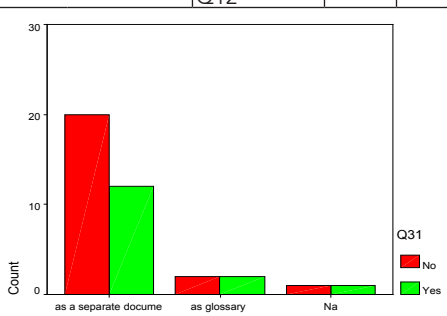
Analysis:

Natural Language processing has less chances of rework.

Q. Data Dictionary created

- as glossary
- as a separate document

			Q31		Total
			No	Yes	
Q	as a separate document	Count	20	12	32
		% within Q12	62.5%	37.5%	100.0%
	as glossary	Count	2	2	4
		% within Q12	50.0%	50.0%	100.0%
	Na	Count	1	1	2
		% within Q12	50.0%	50.0%	100.0%
Total		Count	23	15	38
		% within Q12	60.5%	39.5%	100.0%



Q12

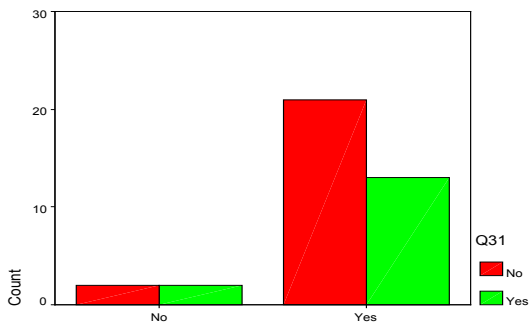
Analysis:

Data Dictionary created as a separate document has success rate as 65.6%.

Q. Templates for requirements exists.

- Yes
- No

			Q31		Total
			No	Yes	
Q	No	Count	2	2	4
		% within Q13	50.0%	50.0%	100.0%
	Yes	Count	21	13	34
		% within Q13	61.8%	38.2%	100.0%
Total		Count	23	15	38
		% within Q13	60.5%	39.5%	100.0%



Q13

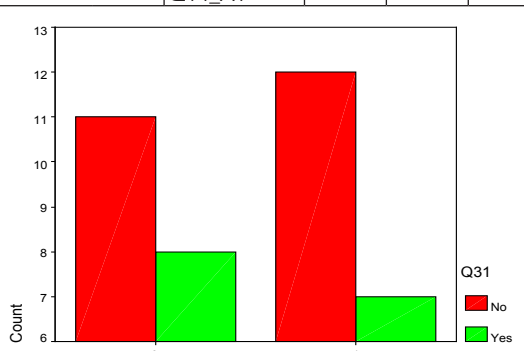
Analysis:

Templates if created has success rate of 61.8%

Q Which requirement elicitation method was used ?

- Questionnaires
- Focus groups
- Design reviews
- Interviews
- Participatory Design
- Other
- None

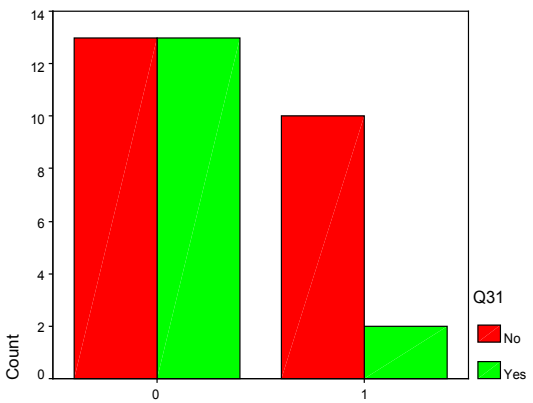
			Q31		Total
			No	Yes	
Q	0	Count	11	8	19
		% within Q14_A1	57.9%	42.1%	100.0%
	1	Count	12	7	19
		% within Q14_A1	63.2%	36.8%	100.0%
Total		Count	23	15	38
		% within Q14_A1	60.5%	39.5%	100.0%



Q14_A1

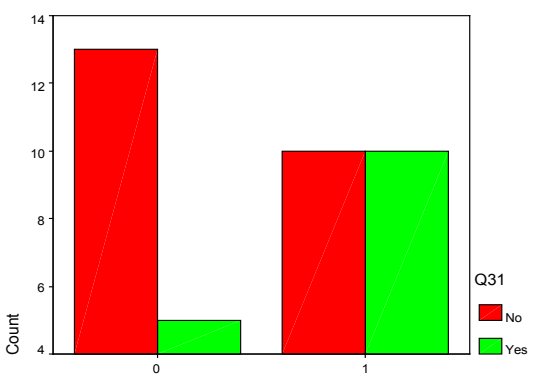
Volume : 4 | Issue : 5 | May 2014 | ISSN - 2249-555X

			Q31		Total
			No	Yes	
Q	0	Count	13	13	26
		% within Q14_A2	50.0%	50.0%	100.0%
	1	Count	10	2	12
		% within Q14_A2	83.3%	16.7%	100.0%
Total		Count	23	15	38
		% within Q14_A2	60.5%	39.5%	100.0%



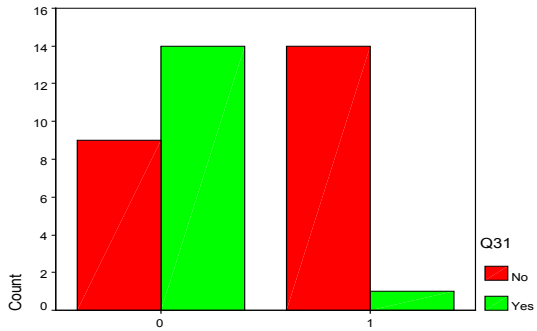
Q14_A2

			Q31		Total
			No	Yes	
Q	0	Count	13	5	18
		% within Q14_A3	72.2%	27.8%	100.0%
	1	Count	10	10	20
		% within Q14_A3	50.0%	50.0%	100.0%
Total		Count	23	15	38
		% within Q14_A3	60.5%	39.5%	100.0%



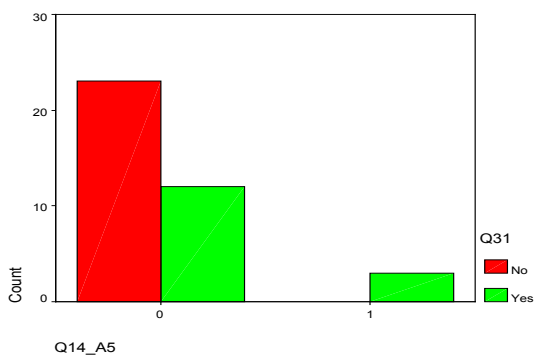
Q14_A3

			Q31		Total
			No	Yes	
Q	0	Count	9	14	23
		% within Q14_A4	39.1%	60.9%	100.0%
	1	Count	14	1	15
		% within Q14_A4	93.3%	6.7%	100.0%
Total		Count	23	15	38
		% within Q14_A4	60.5%	39.5%	100.0%



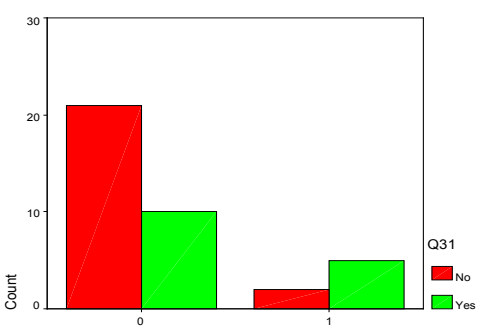
Q14_A4

			Q31		Total
			No	Yes	
Q	0	Count	23	12	35
		% within Q14_A5	65.7%	34.3%	100.0%
	1	Count	0	3	3
		% within Q14_A5	.0%	100.0%	100.0%
Total		Count	23	15	38
		% within Q14_A5	60.5%	39.5%	100.0%



Q14_A5

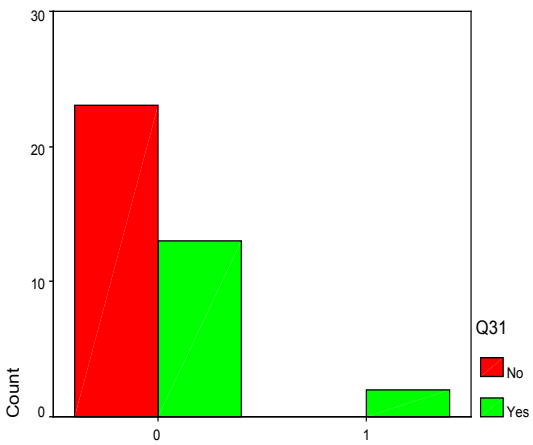
			Q31		Total
			No	Yes	
Q	0	Count	21	10	31
		% within Q14_A6	67.7%	32.3%	100.0%
	1	Count	2	5	7
		% within Q14_A6	28.6%	71.4%	100.0%
Total		Count	23	15	38
		% within Q14_A6	60.5%	39.5%	100.0%



Q14_A6

Volume : 4 | Issue : 5 | May 2014 | ISSN - 2249-555X

			Q31		Total
			No	Yes	
Q	0	Count	23	13	36
		% within Q14_A7	63.9%	36.1%	100.0%
	1	Count	0	2	2
		% within Q14_A7	.0%	100.0%	100.0%
Total		Count	23	15	38
		% within Q14_A7	60.5%	39.5%	100.0%



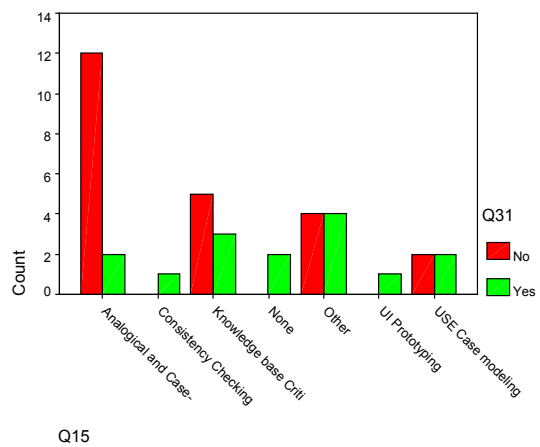
Analysis:

Questionnaires	63.2%
Focus groups	83.3%
Design reviews	50.0%
Interviews	93.3%
Participatory Design	0.0%
Other	28.6%
None	0.0%

Q Which requirement analysis method was used ?

- Requirements Animation
- USE Case modeling
- Automated Reasoning
- Analogical and Case-based Reasoning
- Knowledge based Critiquing
- Consistency Checking
- UI prototyping
- Other
- None

			Q31		Total
			No	Yes	
Q	Analogical and Case-based Reasoning	Count	12	2	14
		% within Q15	85.7%	14.3%	100.0%
	Consistency Checking	Count	0	1	1
		% within Q15	.0%	100.0%	100.0%
	Knowledge base Critique	Count	5	3	8
		% within Q15	62.5%	37.5%	100.0%
	None	Count	0	2	2
		% within Q15	.0%	100.0%	100.0%
	Other	Count	4	4	8
		% within Q15	50.0%	50.0%	100.0%
	UI Prototyping	Count	0	1	1
		% within Q15	.0%	100.0%	100.0%
	USE Case modeling	Count	2	2	4
		% within Q15	50.0%	50.0%	100.0%
Total		Count	23	15	38
		% within Q15	60.5%	39.5%	100.0%



Analysis:
Analogical and case tools have success rate as 85.7%.

Conclusion: From the above discussion it is clear that projects which used one or the other analysis technique had higher success rate compared to those which did not use any RE Above responses and graphs shows there is positive relationship between RE processes and software development cost.

REFERENCE [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. |