REQUIREMENT RELIABILITY METRICS FOR RISK ASSESSMENT

Amir Akhter Jamili
PAF KIET
amirjamili@yahoo.com

Abstract

There are a number of criteria for measurement of quality. Two of the most important of these are Validity and Reliability [1]. Software Reliability is not as well defined as Hardware reliability, but now Software Engineers are working to promote the concept of Reliability in software products. This paper presents the requirement reliability metrics for risk assessment. These metrics could help to quantify requirements using quality attributes impacting on reliability that help to analyze possible risk in software.

Keywords: Reliability Metrics, Risk Assessment, Metrics, Requirement Measurement.

1. INTRODUCTION

The success of a Project is highly dependent on the requirement and it's interpretation to the different phases of SDLC. This is a benchmark concept in Software Engineering discipline. There is plenty of research being conducted to make requirement cycle more manageable and predictable for handling associated risks. Software metrics play a key role in this domain and provide good support for decision-making implies support for risk assessment and reduction [2, 3, 5]. The term software metrics is a collective term given to the measurement of the various attributes concerning software engineering. Software metrics have been used as indicators of software quality [9]. The most important issues in measuring quality are; reliability and validity, and their impact on measurement errors. Validity defines whether the measurement or metric really measures what it should be measuring. In other words, it refers to the extent to which an experimental base measurement reflects the real meaning of the concept under consideration**.** An obvious way to estimate the reliability is statistical techniques to analyze data [4]. Reliability refers to the consistency of a number of measurements taken using the same measurement methods on the same subjects. If repeated measurements are highly consistent or even identical, then there is a high degree of reliability with the measurement method or the operational definition. If the variations among repeated measurements are large, then reliability is low [1]. Requirement reliability is an inverse proportion of risk if reliability is high then risks associated with project are low and if the requirement reliability is low then risks associated with project are high. There were several works in the analysis of software reliability elements, and most of them were the by-products of general study for software quality [10].

Traditional metrics approach does not cover the evolution of requirement in the earlier stager of the SDLC.

Because of this inability to identify flaws in the early requirement gathering phase, later stages of the SDLC suffer from unpredictable occurrence and behaviors. A sound and reliable requirement gathering phase will ensure that later stages of the production will produce outcomes at par with expectations and results ensuring both validity and reliability of the product. Conventional Software engineering processes try to access the risks evolved by basing their predictions on past experience, which may or may not reflect the nature of the project at hand. A good example of this is the common practice of extraction of risk associated with time is based on previous behavioral observations and its effects on the team and critical issues faced by project management.

2. Methodology

Number of Quality attributes related to requirement specifications is subjective and their aspects are measurable by their indicators [6]. Requirement Reliability metrics use four categories of Quality indicators for measuring requirement reliability, these are 1: Completeness: Requirements that have all related

0-7803-8871-2/04/$20.00 ©2004 IEEE

attributes and fully defined and most probably no changes will occur. 2: Weak: Requirement is not seized and not fully defined most probably changes will occur. 3: Complex: If the Interaction is complicated from technical, design, development points of view then this is known as a complex requirement**.** 4: Ambiguous: Requirements that have two or more interpretations

In Project XY the entire requirement is treated as a unit, in metrics database requirement is classified into their Quality Indicator. Each Iteration of requirement document has an individual place in this database with its priorities. Priorities are defined in three levels 1: Mandatory: Requirements that have the highest priority, and must be met at all costs. 2: Evident: Requirement that have a lower level of priority 3: Frill: Requirement that have the lowest priority.

Reliability expressed in terms of the size of the standard deviations of the repeated measurements. We use Index of Variation to compare these iterations. Index of Variation (IV) is simply a ratio of the standard deviation to the mean:

IV= Standard deviation/mean

If repeated measurements are highly consistent or even identical, then there is a high degree of reliability with the measurement method or the operational definition. If the variations among repeated measurements are large, then reliability is low.

2.1 Completeness

	Mandatory	Evident	Frill
Iteration1	45	25	30
Iteration2	40	20	40
Iteration3	42	19	39
Iteration4	48	28	24
Iteration5	40	26	34
Iteration6	45	25	30
Iteration7	49	29	22
Iteration8	47	29	24
Iteration9	41	30	29
Iteration10	38	21	41
Std	4	4	7
Mean	44	25	31
IV	8.75	15.85	22.29

Table 1.0: Complete Requirement

Completeness = **Mandatory + Evident + Frill**

IV= Standard deviation/mean

STD	9
Mean	33
IV	28

Table 1.1: Index of variation in Complete Requirement

2.2 Weak

	Mandatory	Evident	Frill
Iteration1	40	22	38
Iteration2	41	21	38
Iteration3	49	28	23
Iteration4	50	29	21
Iteration5	47	27	26
Iteration6	41	25	34
Iteration7	39	19	42
Iteration8	38	29	33
Iteration9	41	30	29
Iteration10	38	21	41
Std	4.53	4.04	7.47
Mean	42	25	33
IV	10.68	16.10	22.99

Table 2.0: Weak Requirement

Weak = **Mandatory + Evident + Frill**

IV= Standard deviation/mean

STD	9
Mean	33
IV	27

Table 2.1: Index of variation in Weak Requirement

2.3 Complex

	Mandatory	Evident	Frill
Iteration1	39	29	32
Iteration2	48	23	29
Iteration3	46	27	27
Iteration4	41	27	32
Iteration5	38	28	34
Iteration6	37	21	42
Iteration7	50	19	31
Iteration8	52	25	23
Iteration9	41	30	29
Iteration10	38	21	41
Std	6	4	6
Mean	43	25	32
IV	12.83	15.20	18.34

Table 3.0: Complex Requirement

Complex = **Mandatory + Evident + Frill**

IV= Standard deviation/mean

STD	9
Mean	33
IV	27

Table 3.1: Index of variation in Complex Requirement

2.4 Ambiguous

	Mandatory	Evident	Frill
Iteration1	39	28	33
Iteration2	35	29	36
Iteration3	42	20	38
Iteration4	48	28	24
Iteration5	46	19	35
Iteration6	41	18	41
Iteration7	42	26	32
Iteration8	49	23	28
Iteration9	50	17	33
Iteration10	38	21	41
Std	5	5	5
Mean	43	23	34
IV	11.76	19.79	15.79

Table 4.0: Ambiguous Requirement

Ambiguous = **Mandatory + Evident + Frill**

IV= Standard deviation/mean

STD	10
Mean	33
IV	29

Table 4.1: Index of variation in Complex Requirement

AB = Ambiguous Requirement, CO = Complete Requirement

WK = Weak Requirement, CP = Complex Requirement

OV= Overall (AB+CO+WK+CP)

	AB	CO	WK	CP	OV
STD	10	9	9	9	9
Mean	33	33	33	33	33
IV	29	28	27	27	28
Risk Factor	71	72	73	73	72

Table 5.0: Index of Variation through out the project

In the table above, requirements are classified into number of iterations, in metrics database all these requirement iterations are classified according to their definitions. These definitions are based on quality attributes impacting reliability. The results of the assessment of requirement iterations are based on consistency among them. If the consistency between these iterations are low and variation among them is high it means the quality attributes impacting on reliability are lacking and if the consistency between them is high and variation is low then quality attributes impacting on reliability is high influence. (See fig 1.0 and 2.0).

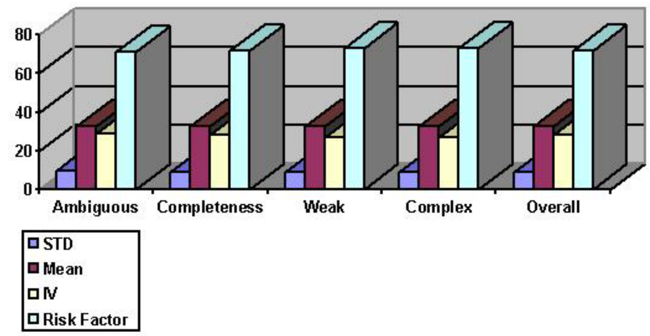

Figure 1.0: Requirement Reliability

Risk = Index of Variation – Reliability

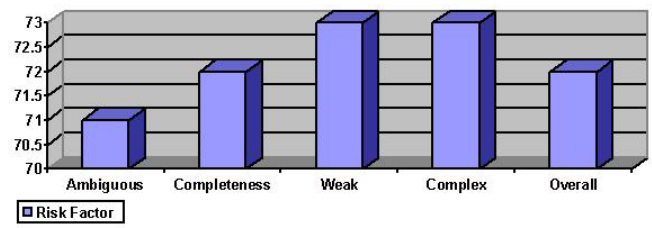

Figure 2.0: Risk Factor

In metrics database requirement are classified in to four different parts according to their definition in above graphs (fig 1.0 and fig 2.0) shows not only the impact of consistency individually but also overall as a unit, In fig 2.0 risk factor is displayed after subtraction of index of variation from reliability and shows risk associated with requirements.

3. CONCLUSIONS

In this paper I have presented Software Reliability metrics for Risk Assessment. Requirement specifications are subjective and their aspects are measurable by their indicators. Traditional metrics do not support the risk measurement of requirements in the earlier phases. This lack of defined measurement may cause serious unpredictable effects on the entire project management. Existing software engineering practice for evaluating software risk is base on calculating probability from past experience and extrapolation of these values for future project planning. According to my work I have presented an alternative to the traditional method of risk assessment instead of basing our requirement analysis on past experience, requirement reliability metrics focuses on the gathering of data from the project at hand. Requirement reliability metrics using four categories of quality indicator impacting on reliability these are completeness, weak, complex and ambiguous. These indicators refine and classify requirement metrics database. In Requirement reliability metrics reliability refers consistency to a number of measurements taken using the same measurement method on the same subject. If variations among them are high it means reliability is low and probability of risk is high and if variation among these variables are low then reliability is high and probability of risk is low.

References

[1] Stephen H. Kan "Metrics and Models in Software Quality Engineering", Second Edition 1995.

[2] Norman Fenton & Martin Neil, "Software Metrics: Road Map" International Conference on Software Engineering - Ireland, ACM, Pages 357-370, 2000.

[3] Tom Mens, Sege Demeyer, "Future Trends in Software Evolution Metrics", WPSE - Vienna- Austria, ACM, Pages: 83-86, 2001.

[4] Dr. Linda Rosenberg, Ted Hammer, Jack Shaw "Software Metrics and Reliability", 9th International Symposium - Germany, Nov 1998

[5] Norman E Fenton, "Software Metrics: Successes, Failures and New Directions", SM/ASM Conference, 1999.

[6] IEEE Standard 830-1998 Recommended Practice for Software Requirements Specifications.

[7] Willa K. Ehrlich, John P. Stampfe and Jar R. Wu "Application of software reliability modeling to product quality and test process "International Conference on Software Engineering Proceedings of the 12th international conference on Software engineering", Nice, France Pages: 108 – 116, 1990.

[8] Nachiappan Nagappan "Toward a Software Testing and Reliability Early Warning Metric Suite", International Conference on Software Engineering Proceedings of the 26th International Conference on Software Engineering, Pages: 60-62, 2004.

[9] Basili V., Briand, L., Melo, W,"Validation of Object Oriented Design Metrics as Quality Indicators. IEEE" Transactions on Software Engineering, Vol. 22, Pages: 751 – 761, 1996.

[10] Isao Miyamoto "Toward an effective software reliability evaluation", International Conference on Software Engineering Proceedings of the 3rd international conference on Software engineering, Atlanta, Georgia-United States, Pages 46 - 45, 1978.