A Study of the Metrics for Measuring the Quality of the Requirements Specification Document

Dr. Bernard Wong
bernard@it.uts.edu.au
University of Technology, Sydney
Faculty of Information Technology
PO Box 123, Broadway,
NSW 2007, AUSTRALIA.

Abstract

The primary objective of this paper is to present an exploratory study on the measurements used for evaluating the quality of an e-commerce Requirements Specifications document. The paper presents the results of a study, which uses qualitative techniques to investigate the measurements for the requirements phase. The study involved an e-commerce project, and two stakeholder groups, the users and the developers. The study finds that the two groups of stakeholders are very similar in the measurements they choose for evaluating requirements documents. The results also find support for the use of IEEE 802-1993.

These results, whilst still exploratory, are valuable as they highlight the differences and similarities of not just the stakeholder groups, but more importantly the choice of measurements at the Requirements phase.

Keywords: Software evaluation, Software Quality, Metrics, Empirical Software Engineering, Human Factors, Requirements Engineering

1. Introduction

Increasingly sophisticated technology makes it possible to build more complex systems more quickly, however a system is only useful to the customer if it addresses the real requirements. No matter how sophisticated the technology, no matter how quick the development of these complex systems, if it does not address customer and organizational needs, the solution is not accepted. Today this scenario is also the challenge found in e-commerce development. It involves the rapid deployment of leading-edge technology, integration of multiple systems, and is usually subject to a significant set of expectations and constraints concerning performance, scalability, security, flexibility, usability and functionality.

Evidence from many studies has shown that the problems in software development, and in this case e-commerce development, begin right at the start, in the requirements phase. Requirements development and management have been described as critical in the software development process, however little is understood by why the difficulties, and how the requirements phase can be improved.

This paper describes a recent qualitative study, which identifies the measurements used for evaluating the quality of the requirements document for an e-commerce project. The study also studies the measures used by potential end-users and compares them with those used by the developers. The results show similarities between the two stakeholder groups but finds that the motivation, priority and rationale for the measurements differ.

This study is extremely significant for both the software engineering and software quality communities as it shows through empirical studies how the requirements phase in software development utilize measurements and why they differ. It introduces appropriate measurements to evaluate the software at the major milestones of development rather than just the measurements at the end of a project.

2. Quality of the Requirement Document

Software requirements define the required functional requirements (what the software must do), performance requirements (how many and how fast) and interface requirements (with what how and with whom the software must interact).

The IEEE 802-1993 standard [7] describes the recommended practice for software requirements specification. The document lists eight attributes that requirements specifications are expected to exhibit: completeness, consistency, correctness, modifiability, ranking, trace ability, non-ambiguity and verifiability. These attributes are not independent. For example, a specification cannot be correct if it is incomplete.

However it has been stated that these attributes are subjective.

The literature on software development has introduced many models to describe the activities, constraints, and resources involved in the production of software ([1], [8]). Though there are differences between these models, similarities between them also exist. For example, the gathering of requirements, producing requirements documents and requirements specifications will always be at the start of a development project. In some models this activity occurs once at the start, whilst in other models, this may occur many times throughout the development cycle. Another example is in the analysis and design milestone. As is the case with requirements gathering, this activity is a phase that exists in all models. Whilst it may exist as only one milestone, it may be a repeated phase in other models. And the most obvious example is, of course, the final software product, the programs and the databases. The many models may have many other activities, but these three are major milestones in all. How an analyst gathers the requirements, or analyzes or designs, could differ between each model. The issue, which is relevant to this study and the framework, is that at each of these stages, outputs exist which requires evaluation. The software quality literature has continuously promoted the Verification, Validation and Testing approach. The literature emphasizes the importance for verifying and validating between the requirements stage, the design stage and the coding stage. Therefore, the question is raised as to whether the evaluation of requirements documents, requirements specifications, and design documents are the same, and if not, how they differ.

As such, the research question of this paper investigates the metrics used for requirements quality evaluation. The study will compare these characteristics with the attributes from IEEE 802-1993 [7], and also compare two stakeholder groups, the potential end-users and the developers.

3 Data Collection and Analysis

Eight subjects were interviewed singly and in-depth at a distribution organization during late 2002. All respondents were involved with the e-commerce project being evaluated, either as a user of the software or as a developer supporting the software. Four of the subjects were users and four were developers. The organization is an international distribution company of wireless voice and data products and a premier supplier of outsourced services with over 20,000 customers worldwide. The organization recently installed a new financial system, and aims to implement an e-commerce solution to improve and automate their supply-chain management and to introduce added value to customer service, like on-line customer order tracking, on-line product catalogs, and in-house management reporting. The study was conducted after the first phase of development was released. This first phase included a catalog system, and an automatic pricing system, which would calculate in real time appropriate pricing of products for each customer.

The people surveyed came from different jobs and backgrounds in this organization. All the respondents were familiar with the business and the objectives of the e-commerce project. We surveyed two programmers, a technical support leader, and an operations coordinator from the development side. The development team all had a minimum of 4 years experience in the I.T industry. We also surveyed 2 sales managers, financial controller, who "owns" the system, and the marketing manager from the user side.

The interviews aimed to focus on the respondent's perception of the quality of the requirements document, which consisted of both the requirements description and specification, and later the quality of the first release of the e-commerce system. There were no hints, nor guidelines used during the interview, which would influence the subjects to give any particular result. Most subjects were involved with both the evaluation of the requirements document and the e-commerce system. However, one sales manager resigned at the beginning of the year, and so another person became involved in the project and contributed to the second survey.

To identify the full set of linkages connecting means to ends, users were given a laddering task ([4], [6], [9], [10]). The laddering procedure consists of a series of directed questions based on mentioned distinctions the individual has with respect to the quality of the software being evaluated. The purpose of the laddering is to force the user or developer up the "ladder of abstraction" to uncover the structural aspects of user knowledge as modeled by the means-end chain. The questioning procedure was designed around the unique demands of the laddering procedure. They were based on prior answers, the interpretation of the answers, and focused on "pushing the participant up" the characteristic-consequence-value hierarchy. Very often, further clarification of the answers was sought before introducing another question. The laddering method has been widely used in the consumer research and psychology disciplines. It has become an accepted method to gain insights regarding sources of value that are perceived to be, or could become, motivationally important ([2], [3], [11], [12], [15], [17]).

In the interviews warm-up questions were used to set the tone for the interviews. Subjects were asked about their interest in computers, in particular software, the web, e-commerce. Different types questions were used between the two sessions, but the aim was the same, that is, to relax the subject in order to obtain more detailed and spontaneous answers.

Questions were asked regarding their desired e-commerce project. How much had they been involved in the preparation of the requirements document? How do they rate the finished document? These questions allowed for the "laddering" approach to then take over. The questions, which followed, then asked why they rated the document in that way, and what contributed to that assessment. The interview would then focus on each reply, with continual questions based on each answer given. The questions and answers created a chain starting from the characteristic used in the quality evaluation, with links to the desired consequences, until finally ending with the value sought. This process would be repeated continuously for each identified characteristic, resulting in a number of "ladders" being created for each respondent.

An example of the use of this laddering approach is given in an earlier study of Wong & Jeffery [15], where an excerpt of one of the respondents is given and described. It must emphasized, that each question is carefully formulated to not bias the results. At times throughout the interview, clarifications of the replies are made to ensure correct interpretation of the dialogue. Whilst this may appear to bias the study, it should be pointed out that what is clarified, is as a result of what had already been said. Unfortunately, the limited size of this paper prohibits giving examples of the interviews.

After performing the interviews, the transcripts were analyzed. The first step in the analysis was to conduct a thorough content analysis of all the elicited concepts. All the responses at the characteristics level were considered first, so that terms close in meaning could be grouped together. The goal here was to reduce the fragmentation of responses that occurred when respondents were using their own language or terminologies, without losing meaning, by grouping elements with widely divergent meanings into the same category. This procedure was repeated at the consequence and value levels. All laddering responses were then expressed in a set of standard concepts. The aggregate set represented the content component of the respondents' quality evaluation structure. The results are then represented in tables, and then modeled using a structured chart, one chart for the users, and one for the developers.

4 Results

Any instance in which a subject links at least two elements together in an asymmetrical fashion (A causes, produces, or leads to B) is defined as a ladder. In all, 17 ladders were elicited from the subjects for the requirements document, the shortest ladder having a length of two and the longest ladder a length of four. The typical ladder was comprised of from three to five elements, although there were many two-element ladders (that is, when a number of characteristics lead to the same consequence).

Three tables for users and three tables for developers are represented here. Each table lists not only the measurements used in the evaluation of the requirements document but also compares these measures with the attributes of IEEE 802-1993 [7].

It would be trivial to just address the issue of stakeholder differences between users and developers, as this difference is obvious. However, it is of interest to identify what the desired values and consequences are for each stakeholder, and to determine whether they are the influence for the choice of characteristics and measurements in their evaluation of requirements document and e-commerce system.

The first tables, table 1 and table 2, list the metrics collected from the respondents during the interviews. Content analysis was used to simplify this list, reducing any redundancies. The metrics were grouped using headings found in other quality models and found in software engineering literature.

GROUPING of the METRICS	Measurements for Requirements Document	IEEE 802-1993
Economic Perspective	Estimated cost of development	Completeness
Institutional Perspective	Reputation of staff putting together document	Consistency
Functional Perspective	Outlines clearly what the developers need to do and why	Non-Ambiguity
	Required reports are accurately described	Non-Ambiguity, Completeness, Correctness
	Functionalities are clearly mapped to business objectives	Non-Ambiguity
Usability Perspective	Describes proposed screen layouts	Non-Ambiguity
	Document is easy to understand	Non-Ambiguity
	Document is complete	Completeness

	Document is consistent	Consistency
Operational Perspective	Description of web environment	Correctness
	Description of testing procedures	Verifiability
	Description of risk areas and options	Traceability, Verifiability

Table 1 Content Analysis of characteristics from the Users

Table 1 shows the list of metrics and characteristics obtained from the set of users. As can be seen, the number of elements obtained is lengthy. The metrics elicited by the users for evaluating the requirements document were related to the characteristics economics, functionality, usability and operational issues. The users did not mention any measurements relating to the characteristics support nor technical issues.

The characteristics such as functionality and usability were also mentioned continuously. In fact, functionality was a major issue for the evaluation of the requirement document. Usability is important, only in that the requirements document is easy to understand, consistently written, and easy to follow. This is a strong support for the IEEE 802-1993 attributes [7].

Other characteristics were raised, such as cost or value for money, brand name, and reputation. Not much discussion was given to these characteristics, though the users all stated that these characteristics led to having a system, which delivered better quality of life and better job security.

GROUPING of the CHARACTERISTICS	CHARACTERISTICS for Requirements Document	IEEE 802-1993
Technical Perspective	Use of modeling tools Use Case diagrams	Completeness, Consistency, Correctness, Modifability, Ranking, Traceability, Non-Ambiguity, Verifiability
(Portability & Maintainability)	Description of the environment, the configuration, the hardware, the network	Completeness, Correctness
	Good design of architecture	Correctness
Functional Perspective	Outlines clearly what the developers need to do and why	Non-Ambiguity
	Required reports are accurately described	Completeness, Correctness
	Functionalities are clearly mapped to business objectives	Non-Ambiguity
Usability Perspective	Describes proposed screen layouts	Correctness
	Document is easy to understand	Non-Ambiguity, Consistency
	Document is complete	Completeness
	Document is consistent	Consistency
Operational Perspective	Description of web environment	Correctness
(Reliability, Efficiency)	Description of testing procedures	Verifiability
	Description of risk areas and options	Traceability, Verifiability

Table 2 Content Analysis of characteristics from the Developers

Table 2 shows the list of metrics and characteristics obtained from the set of developers interviewed. Like the list in figure 1, the number of elements obtained is lengthy. It would appear that there are a lot more similarities between the users and developers when evaluating the requirements document.

Unlike the users, technical characteristics played a very important role. This supported the findings of earlier studies ([13], [14]). Much of the developers' focus, were on development and programming design and approaches, the development process, program documentation and tools. The use of appropriate modeling tools, seemed to be the developer's focus for consistency, completeness, non-ambiguous requirement documents. The discussions centered round the problems faced when lack of adequate processes, documentations and tools. Frustration was raised when discussion moved towards maintenance and enhancements of poor quality software. Similar comments from a number of the developers highlighted the lack of enjoyment in their job when appropriate programming practices were not followed.

With the requirements document quality, the developers gave no consideration to characteristics such as cost, value for money, brand name, or reputation. Though small amounts of interest came from the users, it is suspected that perhaps the manager would have more interest.

5 Conclusion

The goal of this study was to investigate the measurements used for evaluating the quality of

requirement specification documents. The research question proposed.

What are the characteristics used for requirement document quality evaluation? Are these characteristics similar to those proposed by the IEEE 802-1993 document? How do different stakeholder groups compare with regards the characteristics used in the measurement?

Though the study is exploratory, and the findings require further empirical work, the results propose possible answers.

It is evident from the results that the users and the developers differ. Measurements identified in the requirements phase appear to be similar which differs from the measurements used for evaluating the finished software product [16]. It is also evident from the results that IEEE 802-1993 [7] is supported. Though the stakeholders differ in how they prioritize these attributes, the measurements for the quality of the requirements documents follow a similar focus.

Software engineering literature, have identified many characteristics, in the many models of quality. Though some are similar, many of the models differ in the characteristics, which define quality, with no explanation of what determines the choice of characteristic. The research conducted in this study and in previous studies in the development of the Software evaluation framework has introduced a model, which gives the explanation for the use of a measurement. The goals of an individual are important if the measurement is to be applied in context.

These findings are valuable to software quality practitioners and software engineers. The results help to clarify areas of measurements, and the goals sought by the stakeholders. It helps to explain the differences and the similarities but more important, why the differences. Activities such as inspections, walkthroughs, reviews, and testing can be more focused on measurements identified as important to the stakeholder and prioritized. Estimates of impact on the consequences and values can be potentially calculated, even from the requirements document phase, and can be recalculated at each milestone as a means of tracking a software project quality. However, the study has only investigated users and developers of an e-commerce application. There are many more stakeholders. Further studies are required to address these areas.

References

[1] Avison, D. & Wood-Harper, A.T.: Multiview: An Exploration in Information Systems Development, Blackwell Scientific Publications, Oxford, 1990.

[2] Bagozzi, R. and Dabholkar, P.: Discursive psychology: an alternative conceptual foundation to means-end chain theory, Psychology & Marketing, 17, July 2000, pp. 535-586.

[3] Bagozzi, R.: Goal-directed behaviours in marketing: cognitive and emotional, Psychology & Marketing, 14, Sept 1997, pp. 539-543.

[4] Gutman,J. & Reynolds, T.J.: An Investigation of the Levels of Cognitive Abstraction Utilized by Consumers in Product Differentiation, *in Attitude Research Under the Sun*, J Eighmey, ed, Chicago: American Marketing Association, 1979, pp. 128-50.

[5] Gutman,J.: A Means-End Chain Model Based on Consumer Categorization Processes, Journal of Marketing, 46 (Spring):1982, pp. 60-72.

[6] Gutman.J.: Means-End Chains as Goal Hierarchies, Psychology & Marketing, 14 (6):1997, pp. 545-560.

[7] IEEE 802-1993, Recommended Practice for Software Requirements Specifications, December 2, 1993.

[8] Pfleeger, S.: Software Engineering Theory and Practice, Prentice Hall, 1998.

[9] Reynolds et al: Laddering Theory, Method, Analysis, and Interpretation, *Journal of Advertising Research*, Feb/Mar 1988, pp.11-31.

[10] Reynolds, T.J. and Gutman, J.: Developing Images for Services through Means-End Chain Analysis, in *Emerging Perspectives on Service Marketing*, 1983, pp. 40-44.

[11] Rokeach, M: Beliefs, Attitudes and Values, San Francisco: Jossey Bass, 1968.

[12] Valette-Florence, P.: A Causal Analysis of Means-End Hierarchies in a Cross-cultural Context: Metholodogical Refinements, Journal of Business Research, v 42 No 2 June 1998, pp. 161-166.

[13] Wong, B. & Jeffery, R.: Quality Metrics: ISO9126 and Stakeholder Perceptions, Proceedings of the Second Australian Conference on Software Metrics, 1995, pp.54-65.

[14] Wong, B. & Jeffery, R.: A Pilot Study of Stakeholder Perceptions of Quality, Technical Report, CSIRO, 1996.

[15] Wong, B. & Jeffery, R.: Cognitive Structures of Software Evaluation: A Means-End Chain Analysis of Quality, Proceedings of the Third International Conference, PROFES 2001, 2001, pp. 6-26.

[16] Wong, B.: A Study of the Metrics applied to the Software Evaluation Framework 'SEF', Proceedings of the Third International Conference on Quality Software, QSIC 2003.

[17] Yankelovich, D: New Rules, New York: 1981 Random House.