Requirements Clinic: Third Party Inspection Methodology and Practice for Improving the Quality of Software Requirements Specifications

Shinobu Saito, Mutsuki Takeuchi, Masatoshi Hiraoka, Tsuyoshi Kitani
NTT DATA Corporation
Tokyo, Japan
{saitousnb, takeuchimtk, hiraokam, kitanit}@nttdata.co.jp

Mikio Aoyama
Dep. of Software Engineering
Nanzan University
Seto, Japan
mikio.aoyama@nifty.com

***Abstract*—We have been involved in a number of large-scale software development projects, which might lead to loss of millions of dollars if failed. The quality of SRS (Software Requirements Specification) is the key to success of the software development. Review and inspection are common practices for the verification and validation of SRS. However, verification techniques used in projects might be characterized as ad hoc. In this article, we propose requirements clinic, a third party inspection methodology for improving the quality of the SRS. In order to systematically inspect a SRS, we developed a perspective-based inspection methodology based on PQM (Pragmatic Quality Model) of SRS. PQM is derived from IEEE Std. 830 from the perspective of pragmatic quality. To inspect a SRS according to PQM, we identified 198 inspection points, which lead to a quality score between 0 and 100. The requirements clinic advises to the requirements engineering team by a comprehensive quality inspection report including quality score, benchmark and SRS patterns for improvement. Since 2010, we have been practicing the methodology to a variety of development projects, and revealed an average of 10.6 ROI in 12 projects. We also discuss the feasibility of the methodology and lessons learned from the practices.**

***Index Terms*—Requirements verification and validation, Third party inspection, Quality model of requirements specifications.**

I. Background And Challenges

NTT DATA is a global IT solution provider with more than 60,000 engineers specialized in the development of large-scale business information systems (BIS). Failure of the large-scale BIS causes loss of millions of dollars. At NTT DATA, requirements engineering process and successive development processes are conducted by separate teams based on the corporate-wide development standard TERASOLUNA [12]. The quality of SRS (Software Requirements Specification) is the key to success to the entire development and quality of the delivered software systems. Together with all activities in requirements engineering process, V&V (Verification and Validation) is a quality gate of the process [2, 8, 16]. Inspection and review are popular techniques used for requirements V&V [8, 13, 14]. However, a recent survey revealed that most organizations do not correctly apply quality rules in requirements inspection although they do have the rules at either corporate or project level [6, 11]. We observed two major challenges in making requirements V&V work in practice.

First, practical and useful quality model and criteria are still missing. As a standard guideline, IEEE Std. 830 [7] is widely accepted. It defines six characteristics of a good SRS. A more generic quality framework is also proposed [3]. However, both criteria are too generic to inspect SRS written mostly in a natural language. We need a more concrete and workable quality model for specific context and viewpoint. One promising model is pragmatic quality type as defined by Krogstie et al. [10]. By pragmatic, it means a degree of understanding of a certain actor, say, user, designer, or tester to a SRS. Although some works have been conducted in this direction, a practical quality model is still missing.

Second, a practical and organized inspection method for a SRS of large-scale BIS is missing. For software development, an inspection process is rigorously defined and widely applied [5]. There are many literatures advising on inspection and review in requirements engineering [1, 4, 8, 13, 17]. However, SRS inspection largely relies on reading informal natural language documents [9]. It is time-consuming and error-prone. There are substantial researches on automation with natural language processing. However, from pragmatic quality, inspection still relies on human reading and requires skill. Skilled people are short, too. It is necessary to provide a practical reading method, which can be conducted by un-skilled people. One promising way is PBR (Perspective-Based Reading) [13, 15]. It provides a set of questions based on specific perspective, such as view from user, designer, or tester. However, practice of PBR is limited.

This article reports on our experience to solve the challenges above mentioned. The technical contributions of this article include:

(1) An inspection process conducted by an independent inspection team at requirements clinic,
(2) A PQM (Pragmatic Quality Model) based on IEEE Std. 830,
(3) A systematic PBR method based on PQM,
(4) A practical description of a comprehensive inspection report including quality score, benchmark and SRS patterns, and
(5) Evaluation and lessons learned from experiences with large-scale BIS projects for three years.

The structure of this article is as follows. Chapter II illustrates entire inspection process at requirements clinic. Chapter III describes the pragmatic quality model, method, and reporting of requirements inspection. Chapter IV reports the experiences of the proposed inspection method at NTT DATA. Chapter V discusses the assessment, evaluation and lessons learned from our experience. Finally, conclusion and future work follow in Chapter VI.

II. Approach: Requirements Clinic

We propose a structured inspection methodology for SRS based on PBR and PQM derived from IEEE Std. 830. The proposed methodology is conducted by an independent inspection team, which we call requirements clinic since the independent inspection team works like a medical doctor for the requirements engineering team. Figure 1 illustrates an overview of our approach.

The entire inspection process consists of four steps:

(1) Checking our SRS: The SRS is checked out from a requirements engineering team to an independent inspection team in the clinic.

(2) Inspection: The independent inspection team conducts inspection to the SRS, and compiles an inspection report.

(3) Feedback: The clinic submits the inspection report and improvement advices with a set of appropriate SRS patterns to the requirements engineering team.

(4) Improvement: The requirements engineering team receives the inspection report from the clinic and takes action to improve the quality of the SRS.

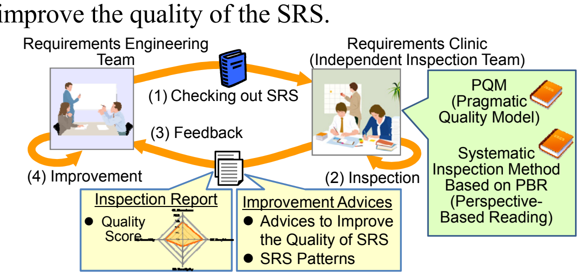


Figure 1. Inspection Process at Requirements Clinic

III. Inspection Methodology

A. PQM (Pragmatic Quality Model)

At NTT DATA, a requirements engineering process and successive development processes are conducted by separate teams. The SRS should be comprehensive to developers of the successive development teams. The inspection to the SRS should address not only syntactic and semantic quality, but also "pragmatic quality". By pragmatic quality [10], we mean the quality of SRS from the perspective of software developer, who read the SRS for architecture and functional design without knowing the background of the specific context of customer and business. It is also necessary that the quality criteria should be practically "inspectable" by the inspection teams of little knowledge on the requirements.

From the prerequisite, we developed PQM based on IEEE Std. 830. It provides a high-level definition on the quality characteristics. However, it is too generic to inspect SRS without human factors of inspectors. Furthermore, we assume inspection teams have little knowledge on the background of project and application domain. To enable third party inspection, we needed to derive concrete and evaluable quality characteristics from IEEE Std. 830.

As illustrated in Figure 2, PQM consists of the following four quality characteristics. Each characteristic is further refined to several sub-characteristics. It is based on the perspective of developers in subsequent development team.

(1) CTG (Conform To Goal)

CTG is derived from "Correct" and "Ranked for importance and/or stability" in IEEE Std. 830. CTG evaluates the degree of conformance of the descriptions of a SRS to the system goals. We assume if the goals are clearly described, the developers can judge the "correctness" and "rank for importance" of the descriptions of SRS.

(2) Coverability

Coverability is derived from "Complete" of IEEE Std. 830. It is a pragmatic completeness of a SRS from the perspective of developers. Coverability is evaluated with whether a SRS is completely described in accordance with all the contents of our SRS guideline.

(3) Comprehensiveness

Comprehensiveness is derived from "Unambiguous" and "Verifiable" of IEEE Std. 830. From the perspective of developers, use of standard description method, templates and terms lead to "Unambiguous" and "Verifiable".

(4) Internal Traceability

Internal traceability is derived from "Consistent", "Traceable", and "Modifiable" in IEEE Std. 830. From the perspective of developer, we reorganize the three characteristics into "internal traceability". The internal traceability requires that the items and relationship between them should be clearly identified in a SRS.

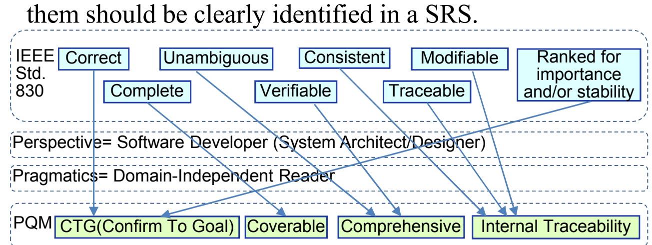


Figure 2. PQM Derived from IEEE Std. 830

B. SRS Guideline

As a guideline for writing a SRS, we developed a SRS guideline based on the IEEE Std. 830 and a bunch of SRS previously developed. The SRS guideline consists of two parts: description guideline and SRS patterns.

(1) Description guideline explains how to describe a SRS. It defines contents of chapters and sections as shown in Table 1, together with templates of sections.

(2) A SRS pattern provides a good example of a SRS. It includes examples of chapters and sections of Table1 with standard notations including UML, ER, and DFD. For example, a concrete goal graph is illustrated for Section 2.1, and a list of concrete NFR definitions is provided for Section 5.1.

TABLE I. TABLE OF CONTENTS OF STANDARD SRS

Chapter	Section
1. Introduction	1.1 Purpose of SRS
	1.2 Intended Reader
	1.3 Structure of SRS
	1.4 References
2. Overview of System	2.1 Goal of System
	2.2 Business and Scope of System
	2.3 Constraints
	2.4 Terms
3.Items Causing Change or Unspecified	3.1 Items Causing Changes
	3.2 Items Unspecified
4. Functional Requirements	4.1 Business Flow
	4.2Functions
	4.3Data Model Definition
5. Non-Functional Requirements	5.1 NFR Grade
	5.2 Requirements to System Architecture
	5.3 Requirements to Migration
	5.4 Requirements to Service Provisioning

C. *Systematic Inspection Method*

1) Inspection Matrix: To conduct inspection correctly and assure the quality of a SRS, we developed inspection matrix by matching PQM and TOC (Table Of Contents) of standard SRS. As illustrated in Figure 3, each row represents sub-characteristic of PQM and each column represents section of standard SRS. Therefore, each cell indicates a specific inspection point if applicable. In Figure 3, X indicates there is at least one inspection point applicable. For Section 2.1 "Goal of System", there are seven inspection points except for C3-2 and C3-3.

2) Inspection Guideline: To make inspectors clearly identify the inspection points, we developed inspection guideline based on the inspection matrix. Figure 4 illustrates parts of the inspection guideline. Since each row of the inspection guideline represents a sub-characteristic of PQM, a set of specific inspection guidelines with respect to the sub-characteristic is described in each row. As a whole, we identified 198 inspection points. It enables inspectors to systematically examine each section of a SRS against specific sub-characteristic. Therefore, people of little knowledge on the system and domain can inspect a SRS with the rigorous inspection guideline.

3) TQS (Total Quality Score) : At each inspection point, the inspector is required to score the SRS: assign +1 if no error, and assign 0 if any errors found. By summing up the score of +1 or 0 over 198 inspection points, and dividing the sum by the total number of the inspection points, 198, we can get a TQS of a SRS between 0 and 100. We allocate an equal value of +1 to all the inspection points, and the TQS indicates an inspection pass ratio as a basic quality measurement of a SRS.

PQM				TOC of Standard SRS			
Character-istics		Sub-Characteristics		2.1 Goal of System	2.2 Business & Scope of System	2.3 Const-raints	2.4 Terms
ID	Name	ID	Name				
C1	COG	C1-1	Clarity of project goals	X			
		C1-2	Correspondence to project goals	X			
C2	Covera-bility	-	-	X	X	X	X
C3	Compre-hensive-ness	C3-1	Templates usage	X	X	X	
		C3-2	Standard description usage		X		
		C3-3	Preparation of glossary				X
C4	Internal Tracea-bility	C4-1	Presence of listed artifacts	X	X	X	
		C4-2	Presence of Identifier	X	X	X	X
		C4-3	Identifiability	X	X	X	X

Figure 3. Inspection Matrix

D. *Inspection Report and Improvement Advices*

1) Inspection Report:

a) Meta-model of Inspection Report: Figure 5 illustrates a meta-model of inspection report. The inspection report presents the results of inspection from the two perspectives: PQM and TOC of standard SRS. The report also presents benchmark for SRS quality with other projects. The executive summary states a short and concise summary of the inspection results. It includes diagnostic from the quality score and benchmark for quality.

Compiling an inspection report can be rather straightforward along with the inspection matrix and guideline. After compiling, the inspector and his/her peers review the inspection report and carefully write executive summary. The executive summary suggests key findings and insights into the root causes of lower quality scores of a SRS. It motivates managers of requirements engineering team to take action to improve the quality of SRS. Peer review is particularly important to make an inspection report fair and valuable to requirements engineering team.

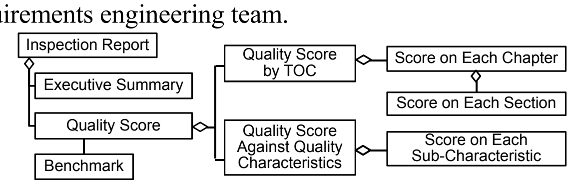


Figure 5. Meta-Model of Inspection Report

PQM				Example of Inspection Criteria for a Sub-Characteristic to Inspection Points	No. of Inspection Points
Characteristics		Sub-Characteristics			
ID	Name	ID	Name		
C1	COG	C1-1	Clarity of project goals	One project goal should be described in one sentence. (project goals should be decomposed to atomic sentences)	2
		C1-2	Correspondence to project goals	All business & system requirements should correspond to more than one project goal.	3
C2	Coverability	-	-	All contents of SRS should be described.	54
C3	Compre-hensiveness	C3-1	Templates usage	Template of SRS should be used.	36
		C3-2	Standard description usage	Standard description (notation) of SRS should be used.	6
		C3-3	Preparation of glossary	Glossary of SRS should be created.	3
C4	Internal Traceability	C4-1	Presence of listed artifacts	All artifacts that be stated in the artifact lists should be created.	16
		C4-2	Presence of Identifier	All artifacts and certain elements of the artifact should have identifier.	32
		C4-3	Identifiability	All artifacts and certain elements of the artifact are identifiable using identifier.	46
Total					198

Figure 4. Inspection Guideline

b) QS (Quality Score): Figure 6 and 7 illustrate two graphs on distribution of the QS by chapter in an inspection report.

Figure 6 presents three bar charts: Max (Maximum score among all the SRS), Baseline (Adjusted Max), and Score of the SRS, by chapter by chapter. The score of Baseline is adjusted by taking into account of the specific conditions of a system under inspection. For example, if some parts of a SRS is reused from a bunch of SRS which were inspected and developed previously, we need to exclude the parts from the evaluation so that we can evaluate the parts of newly specified. In Figure 6, Baseline is downward adjusted by 2.9% for "Functional Requirements" and by 1.3% for "Non-Functional Requirements". The total of Baseline is 95.1%.

As illustrated in Figure 6, we scored the quality of the whole SRS to 57.4% by summing up score of each chapter.

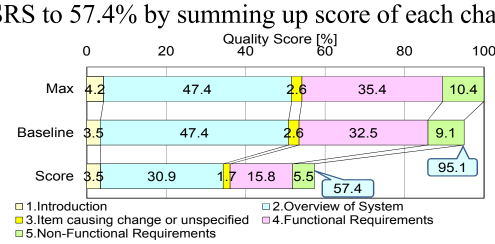


Figure 6. Setting Baseline for TOC View

Figure 7 illustrates the QS of each chapter against the Baseline. It visualizes quality of the SRS quantitatively, and helps to find weakness of the SRS. For example, the QS of "Functional Requirements", 48.6%, is relatively low.

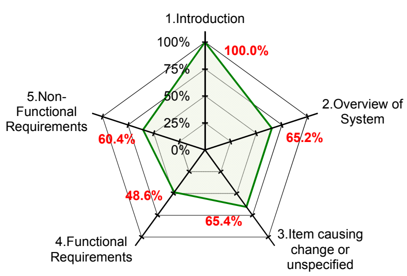


Figure 7. Quality Score by TOC against Baseline

c) Diagnostics with Benchmark: Figure 8 illustrates distribution of the QS and benchmark of a real SRS. It also illustrates QS distribution of 15 percentile and 85 percentile by bar chart of all the projects. It is particularly useful for project manager to understand the relative quality of the SRS of his/her project among other projects.

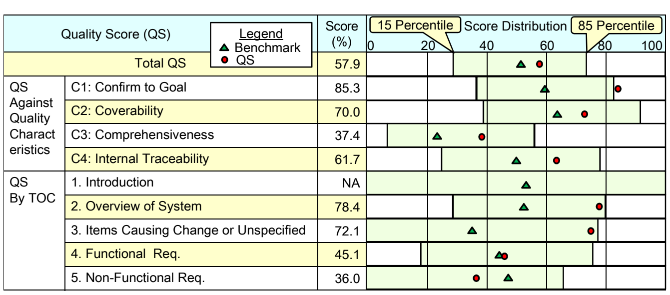


Figure 8. Benchmark

d) Detailed Quality Analysis at Section Level: We can see the details of the QS by looking into the QS of each section in a chapter, say Chapter 4. On the other hand, by walking through the inspection points along with sub-characteristics, we can see which quality sub-characteristic is good or bad. Like this, the inspection report can help us to understand the quality of a SRS from multiple perspectives.

2) Improvement Advice: The inspection team provides improvement advices and selected SRS patterns. Figure 9 illustrates an example of improvement advice and a SRS pattern.

Inspection team compiles advices to the sections of:

(1) Relatively lower QS among other sections, and,

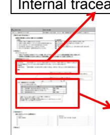
(2) Lower QS against the baseline in benchmark.

Advices suggest improvement with respect to four quality characteristics of PQM. For concrete advices, a SRS pattern suggests a good practice and its example by section by section. The SRS patterns are distilled from a bunch of SRS developed previously. The inspection team chooses right SRS patterns and attaches them to the inspection report.

With improvement advices, we intend all the members of requirements engineering team to share the opportunities of quality improvement of a SRS, and motivate themselves to take any necessary actions to improve the SRS.

A Part of Advices for 2.1.3 (The purpose of the project)

Quality Characteristic	Advice
Conform to project goal	Project goals are itemize by one sentence. The relationship between project goals and business requirements should be described.
Coverability	The background of the project should be described.
Comprehensiveness	—
Internal traceability	Each project goal should have identifier.



Improvement Advices

A Part of SRS Pattern for 2.1.3 (The purpose of the project)

ID	Project Goal	Corresponding Business Requirements ID.
P01	Deliver various service and product quickly to a customer.	BR02, BR03
P02	Manage various service and product efficiently.	BR04
P03	Provide service and product at lower price.	BR01, BR04

Figure 9. Sample of Improvement Advices

IV. Practice

At NTT DATA, we initiated the requirements clinic in 2010. Since then, we have been inspecting 30 to 40 sets of SRS annually. The size of the projects inspected varies. Figure 10 illustrates the distribution of project size in terms of person-month. Note that about one fourth is large-scale project of more than 500 person-months. In practice, there are two types of drivers, who ask requirements clinic to inspect the SRS.

(1) PMO (Project Management Office)

PMO is a corporate-wide management team responsible to monitor and advise to projects. PMO picks up risky and/or large-scale projects and ask the requirements clinic to inspect the SRS.

(2) Project Manager

A project manager takes an initiative to inspect the SRS of his/her project.

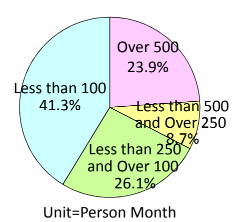


Figure 10. Project Size Distribution

The effort and duration required for inspecting a SRS depends on the size of it. However, as a standard, we allocate one inspector for three weeks; one week for reading and understanding a SRS, one for inspection, and one for compiling and reviewing an inspection report. To ensure the quality of the inspection, a few members join the review of inspection report. After returning the inspection report, the inspection team continues follow-up monitoring including improvement activities and collecting quality statistics.

V. Assesment, Evaluation, And Lesson Learned

To validate the proposed method, we conducted feasibility assessment and evaluation. We also collected lessons learned and feedback from development teams.

(1) Feasibility Assessment prior to Practice

Before applying to real projects, we conducted a proof of concept of the proposed inspection method.

(2) Evaluation from the Practices

We demonstrate the effect of the proposed inspection method with 12 samples from the practices.

A. Feasibility Assessment on the Proposed Method

1) Assessment Goal: Before applying the proposed method to real projects, we conducted an empirical study with two teams on real SRS.

2) Empirical Assessment Method: We grouped four engineers into the following two teams.

Team 1: Two senior engineers with more than six years of experience

Team 2: Two junior engineers with less than two years of experience

Each team is asked to inspect three sets of SRS, SRS_A, SRS_B, and SRS_C, along with the proposed inspection method. The numbers of pages of SRS_A, SRS_B, and SRS_C are respectively, 15, 92, and 180.

3) Results: Figure 11 illustrates the average inspection performance (quality score) of each team, and the average of two teams. Team 1 of senior engineers always outperforms Team 2 of junior engineers. However, unlike other human factors such as program productivity, the maximum deviation is rather small. The maximum deviation is 5.7 (20.8% more) from the average 27.3 for SRS_A.

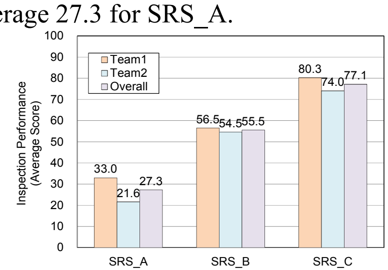


Figure 11. Performance Comparison by Team

Figure 12 illustrates individual inspection performance. By individual performance, senior engineers may not always outperform junior engineer, and the deviation among the engineers are a little wider. The maximum deviation is 8.3 (30.4% more) from the average 27.3, for SRS_A by Inspector 1.

Due to the small number of participants, it is still too early to conclude the assessment. However, it is reasonable to say that the proposed inspection method makes the deviation among the inspectors of widely different experiments small enough. It indicates that the proposed method enables an independent inspection team to inspect a SRS at a level of reasonably well.

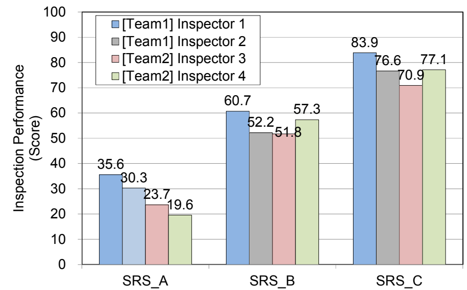


Figure 12. Performance Comparison by Individual

B. Evaluation of the Effectivness from the Practice

1) Evaluation of ROI: We inspected 12 projects in the first year we initiated the requirements clinic. To evaluate the effectiveness of the clinic, we collected statistics of inspection reports and opinions of project managers from the 12 projects. Table II summarizes statistics of the 12 projects.

TABLE II. Practial Cases

Project ID	Domain	Type of Development	No. of Pages	
			SRS	Advices
A	Finance	New	648	19
B	Finance	Extension	374	47
C	Public	New	847	31
D	Finance	Extension	49	30
E	Public	Extension	13	40
F	Public	Extension	10	40
G	Public	Extension	72	40
H	Public	Extension	15	40
I	Public	Extension	7	40
J	Public	Extension	24	40
K	Public	Extension	140	40
L	Finance	Replacement	1,000	79
Total			3,199	446

For evaluating the cost effectiveness of the inspection method, we collected the following statistics.

(1) Cost to correct a SRS based on its inspection report

(2) Estimated cost saved by the inspection in the subsequent design process, i.e. cost to fix errors caused by the SRS at the design process if not identified by the inspection. We asked each project manager to estimate it based on the inspection report when completing design process.

Effectiveness of inspection in terms of ROI:

$$\text{ROI} = \frac{\text{Cost Saved by Inspection [Person-Days]}}{\text{Effort to correct SRS [Person-Days]}}$$

Figure 13 illustrates distribution of ROI of 12 projects. The maximum is 75.0, and minimum is 2.0. 10 percentile and 90 percentile are respectively 2.0 and 42.5. An average of all the ROIs is 10.6. This result well supports the effect of the inspection. It also supports be-spoken rule of thumb that an average of ten times efforts is needed if bugs in a SRS are passed to the subsequent design process and fixed.

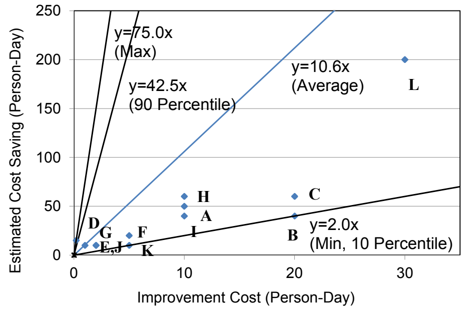


Figure 13. ROI of Inspection

2) Impact of Quality of SRS to Delay of Project: To evaluate the effect of the proposed PQM, we statistically analyzed the correlation between quality sub-characteristics and project performance. Among our findings, Figure 14 illustrates the correlation between "C1-1: Clarity of Project Goals" and the delay of project, i.e. difference of actual complete date from its scheduled date. Figure 14 also illustrates linear approximation and its correlation coefficient of -0.73. This analysis indicates that we can predict the risk of project from quality sub-characteristics of SRS.

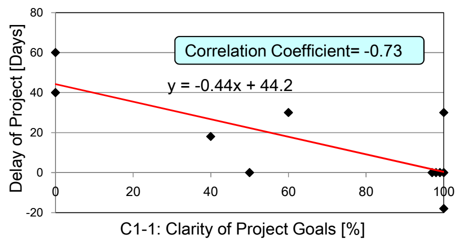


Figure 14. Impact of Quality of SRS to Delay of Project

C. Lessons Learned and Feedback from Practices

We collected some 250 opinions from the requirements engineering teams. Among the opinions, leaders of requirements engineering team gave us many positive feedbacks. The followings are some of major opinions.

(1) Advices teach not only what we should write, but also what we need not write,

(2) Graphical representation of QS including radar chart and benchmarking helps to intuitively grasp overall health conditions of the SRS and requirements engineering team,

(3) Advices help to share the knowledge on how to improve a SRS and to convince the member of requirements engineering team to do, and,

(4) SRS patterns are effective to teach writing SRS to newcomers to requirements engineering team.

VI. Conclusion And Future Work

In this article, we proposed "requirements clinic", an independent inspection team, and an inspection methodology based on the PQM and PBR. The four quality characteristics of PQM are derived from IEEE Std. 830 by considering pragmatic quality. To make the inspection practically workable, we identified 198 inspection points at a specific part of SRS with respect to a specific quality sub-characteristic. We developed a comprehensive inspection report to advise concrete improvement guidelines including quality score from various viewpoints and SRS patterns. To demonstrate the feasibility and effectiveness of the method, we conducted the feasibility assessment, and evaluation of cost effectiveness with of ROI. The feasibility assessment revealed little sensibility of the inspection method to the skill level of inspectors. The evaluation revealed an average of 10.6 ROI, the ratio of cost saving in the successive design process against cost for correcting SRS.

For future work, we are conducting the followings. First, we can provide richer information on writing a SRS depending on the development style and context. Second, we can predict risk based on the inspection. We are continuing the practice and improving the methodology.

References

[1] I. F. Alexander and R. Stevens, Writing Better Requirements, Addison-Wesley, 2002.

[2] B. W. Boehm, "Verifying and Validating Software Requirements and Specifications," IEEE Software, Vol. 1, No. 1, Jan./Feb. 1984, pp. 75-88.

[3] A. Davis, et al., "Identifying and Measuring Quality in a Software Requirements Specification," Proc. of the First Int'l Software Metrics Symposium, IEEE Computer Society, May 1993, pp. 141-152.

[4] C. Denger and T. Olsson, "Quality Assurance in Requirements Engineering," in Engineering and Managing Software Requirements, A. Aurum and C. Wohlin, Eds. Springer, 2005, pp. 163-185.

[5] M. E. Fagan, "Advances in Software Inspections," IEEE Trans. Software Engineering, Vol. SE-12, No. 7, Jul. 1986, pp. 744-751.

[6] G. Fanmuy, et al., "Requirements Verification in the Industry," Proc. of CSDM 2011, Springer, 2012, pp. 145-160.

[7] IEEE Std. 830-1998, IEEE Recommended Practice for Software Requirements Specifications, IEEE, 1998.

[8] JISA, Requirements Engineering Body Of Knowledge (REBOK), Ver. 1.0, Kindaikagakusha, 2011 (In Japanese).

[9] E. Kamsties, et al., "Detecting Ambiguities in Requirements Documents Using Inspections," Proc. of WISE'01, 2001, pp. 68-80.

[10] J. Krogstie, et al., "Towards a Deeper Understanding of Quality in Requirements Engineering," Proc. of CAiSE '95, LNCS Vol. 932, Springer, 1995, pp. 82-95.

[11] C. J. Neill and P. A. Laplante, "Requirements Engineering: The State of the Practice," IEEE Software, Vol. 20, No. 6, Nov./Dec. 2003, pp. 40-45.

[12] NTT DATA, http://www.nttdata.com/global/en/services/adt/.

[13] K. Pohl, Requirements Engineering, Springer, 2010.

[14] A. Porter and L. G. Votta, "An Experiment to Assess Different Defect Detection Methods for Software Requirements Inspections," Proc. of ICSE 1994, IEEE Computer Society, pp. 103-112.

[15] F. Shull, et al., "How Perspective-Based Reading Can Improve Requirements Inspections," IEEE Computer, Vol. 33, No. 7, Jul. 2000, pp. 73-79.

[16] F. Salger, et al., "An Integrated Quality Assurance Framework for Specifying Business Information Systems," Proc. of CAiSE Forum 2009, Jun. 2009, pp. 25-30.

[17] I. Sommerville and P. Sawyer, Requirements Engineering: A Good Practice Guide, Wiley, 1997.