

Clarifying Non-Functional Requirements to Improve User Acceptance – Experience at Siemens

Christoph Marhold¹, Clotilde Rohleder^{1,2}, Camille Salinesi², Joerg Doerr³

¹Siemens PL (DE) GmbH, Lina-Ammon-Strasse 22, D-90471 Nuremberg

²CRI - Université Paris 1, 90, rue de Tolbiac, F-75013 Paris

³Fraunhofer Institute Experimental Software Engineering, D-67663 Kaiserslautern
{christoph.marhold, clotilde.rohleder}@siemens.com, camille.salinesi@univ-paris1.fr,
joerg.doerr@iese.fraunhofer.de

Abstract. [Context and motivation] The starting point for software development is usually the system requirements. The requirements, especially non-functional requirements specified in a document are often incomplete and inconsistent with the initial user needs and expectations. [Question/problem] Experience at Siemens showed us that programmers working on software development often have trouble interpreting under-specified non-functional requirements, resulting in code that does not meet the users' quality expectations and contains "quality faults" that can only be detected later through expensive user acceptance testing activities. [Principal ideas/results] In this problem statement paper, we investigate the need for clarifying non-functional requirements in software specifications to improve user acceptance. In particular we focus on establishing the role of non-functional requirements on user acceptance. [Contribution] Our contribution is that we emphasize the need for a systematic empirical study in this area. We propose a possible set-up where a number of hypotheses have been developed that a systematic experiment will help to validate. Our work is based on industrial experiments at Siemens, in the particular context of the installation of a Product Lifecycle Management (PLM) system.

Keywords: User Acceptance, Non-Functional Requirements.

1 Introduction

It is crucial for implementers that the systems they develop are finally accepted by their users. New systems must therefore meet explicit and implicit criteria for acceptance. Experience shows that user non-acceptance originates more often from inadequate non-functional requirements (NFR) more than from problems with functional requirements [6]. There are many NFR-related causes of user non-satisfaction: for example ambiguous, incomplete, wishful thinking, inconsistent, unusable, silent, under-specified or under-sized NFR. From users' point of view, NFRs are qualities that the system must show. In practice, users are unsatisfied because the system does not achieve their expectation in terms of quality: the perceived quality does not match the expected quality. Generally errors detected during user acceptance tests are reported as bugs. But in fact, non-acceptance can arise from many other issues than bugs. In this paper, we refer to quality as defined in the ISO9126 standard [13] that classifies NFRs as Software Quality tree (Fig. 1), and distinguishes between the internal/external qualities of systems, and quality in use.

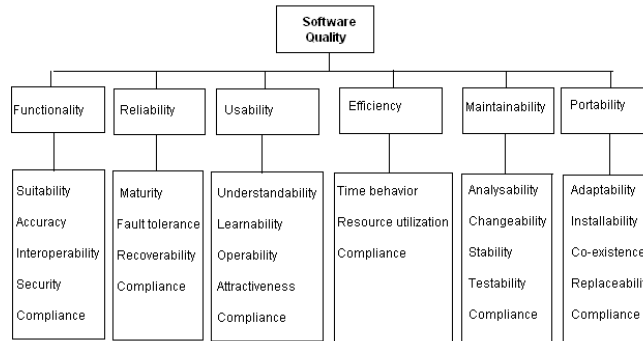


Fig. 1: ISO9126 Software Quality Tree

One major hypothesis underlying our work is that the latter is influenced by the former. Our vision is to significantly increase user acceptance of complex systems in better implementing quality goals (in the shape of an NFR). To our knowledge, there is surprisingly little research to date on the mechanisms, technologies and attitudes that address the impact of non-functional requirements on user acceptance. We have therefore decided to undertake a research work on this topic.

This paper is the first of a series of investigations of the impact on user acceptance of a methodology for clarification of non-functional requirements using an empirical approach. The paper has two parts: In the first part, we introduce the domain of PLM, and report Siemens's experience and view about critical factors of user acceptance while installing PLM systems, and about the perception of the importance of NFR in these projects. The second part of the paper uses this analysis of past experience at Siemens to draw the basis for a series of systematic experiments about the correlation between NFRs and user acceptance. The originality of the paper is twofold (a) it is an industry experience based exploration of a fundamental RE problem, and (b) it explores an issue that has received so far little attention in RE research. This paper reports early results from our analysis in the industrial context of PLM. In the long term, we hope to be able to develop a method that helps predict user acceptance for any kind of complex system based on an analysis of NFR specifications.

The paper is organized as follows: Section 2 describes the research problem in the PLM context. Section 3 presents the need for an empirical study. Related works and conclusions are given respectively in Section 4 and Section 5.

2 User acceptance and NFR specification in the PLM context

2.1 Challenges of PLM systems acceptance by users

Product lifecycle management (PLM) is “the process of managing the entire lifecycle of a product from its conception, through design and manufacture, to service and disposal. PLM integrates people, data, processes and business systems and provides a product information backbone for companies and their extended enterprise” [2]. Installing a PLM system implies –like with other complex COTS such as ERPs –

some kind of matching between users' requirements and the requirements that the system is able to satisfy [20]. Fig. 2 shows that PLM is a strategic business approach that applies a consistent set of business solution in support of collaboration creation, management, dissemination, and use of product definition information across the extended enterprise from concept to end of life [2].



Fig. 2. Overview PLM Product Lifecycle Management [9]

The worldwide PLM applications market in 2007 amounted to \$ 8.7 billion. There is no doubt that PLM is now broadly valued by large manufacturers as well as by small and medium-sized business. Cost, effort, time and complexity of implementing a PLM system can be compared to those for implementing an ERP system. Siemens's experience of implementing its PLM Teamcenter solution showed that the user acceptance tests had best results when non-functional requirements on response time or ease of use had been specified prior to implementation. PLM experts reported bad user acceptance in projects where the specification of non-functional requirements had been focused only on technical non-functional requirements or where user oriented non-functional requirements [14] had been insufficiently specified. Time and efforts spent in projects for correcting the final software product to improve user acceptance (projects of type A) compared to time and efforts spent in projects where more attention was paid to specifying sufficiently all NFRs to ensure user acceptance (projects of type B) was evaluated to be around 100. Moreover, we observed that the level of user acceptance reported in projects of type A has never reached the same level than the user acceptance reported in projects of type B. PLM experts at Siemens report that there is a correlation between specifying non-functional requirements and improving user acceptance. Developers and technical project stakeholders with direct contact with customers are asking for more formality in NFR specification, whereas project managers responsible for budgets want less NFR formalization.

2.2 State of the Practice on NFR specification for PLM system installation

In practice, although one could assume that some care is taken when writing down the initial requirements including the NFRs while implementing such a complex system as a PLM, the role of a clear specification of the NFRs is usually undermined.

Besides, user acceptance test documents tend to be little more than bug reports. We observed that in fact, it is often assumed that a lot of quality information is obvious at this time of the project or is agreed verbally. The main purpose of a PLM system is to enable collaboration among users. PLM systems handle a large collection of collaborative data (requirements specifications, simulation data, design 2D files, 3D models, bill of material, production plans, sales and marketing data, logistics, etc) from the early stage of product development until the maintenance phase. As a result, PLM systems have extremely diverse kinds of users: requirements engineers, CAD designers, CAE and CAM engineers, ERP users, maintenance technicians, etc. Each have specific expectations with respect to the PLM tool, not only in terms of functionality, but also in terms of ergonomics, performance, interoperability with other systems, and ability to support business goals. Besides, PLM systems must handle extremely different fields of application which results in extremely different NFR priorities [19]. For example in OEM Automotive Supplier sector, NFR "Portability" has "absolute" priority because of multi CAD systems (NX, ProE, CATIA, etc.) on different platform infrastructures (Windows, SUN, IBM, HP) used by key PLM users, whereas in the field of Defense the NFR "Security" comes first.

3. Empirical Investigation – Design and Result

3.1 Research questions of Interest and Hypotheses

We decided to undertake an empirical investigation of the relationship between NFRs and user acceptance for several reasons. Besides (i) to assess the usefulness of formal NFR specification models, we are interested in (ii) evaluating whether meeting NFRs achieves better results than acceptance tests and (iii) determining the cost, in terms of extra efforts, to get user acceptance if NFRs are not adequately considered. Overall, the purpose of our empirical investigation is to determine the influence of NFR related factors on the acceptance by users of enterprise-wide systems such as PLM, ERP, etc. The investigation is split in two phases: an initial study, and a series of interviews. The initial study aims at exploring the relationship between NFR specifications and user acceptance in a qualitative way. The final result expected from this study is a series of hypotheses that can be quantitatively validated. This study is qualitative. It is mostly based on expert feedback and lessons learned from PLM installation experiences. The purpose of interviews is to validate quantitatively the hypotheses specified during the empirical study phase, and based on the results, to develop a systematic method that would help predict user acceptance from NFRs. Interviews during this second phase will be undertaken both with users and developers of the PLM system. The questions that will be asked will be about their perception of system in relationship with NFRs, under the same perspective as defined in our hypotheses. Both the initial study and the questions raised during interviews have to be designed to deal with the following two research questions: RQ1: what are the non functional-related factors that influence user acceptance of enterprise-wide systems such as PML? RQ2: what are the influence flows from early requirements specifications down to user acceptance?

3.2 Possible set-up of an empirical study

We should be able to determine the correlation between the clarity of NFRs and user acceptance in setting a number of hypotheses that the interviews will help to validate. Based on the aforementioned return on experience, we drew a series of hypotheses of potential interest. For feasibility reasons, we reduced the list to 3 hypotheses as follows.

H1: Quality of non-functional requirements specification influences user acceptance.

Experience: One customer specified following requirement concerning PLM Graphical User Interface: “GUI should be very simple and attractive for informational users”. Developers needed to know if a complete new GUI had to be developed or if the current ones could be customized to be accepted by the informational users. The requirements had to be specified more precisely: how could the GUI be more attractive for the informational users? The quality of specifications of non-functional requirements seems to help programmers better understand the non-functional requirements in comparison with textual or semi structured descriptions.

H2: User acceptance increases when users are involved in the NFR prioritization.

Experience: in the preliminary phase of specifications –Prove of Concept- the users are committed in the weighting of non-functional requirements. In that phase, we pre-align the prioritizations of non-functional requirements according to the expectations.

H3: Improving user acceptance is not a continuous function of satisficing NFRs.

Experience shows that user acceptance has a maximum level to be reached. There is a maximum limit of efforts that should be invested in satisficing NFRs to get maximized user acceptance. If the effort goes further, the user acceptance will never be better.

These hypotheses shall be checked in a systematic experiment. We need to compare efforts and time spent in getting the mile stones user acceptance test within different projects having different quality of specifications. We aim to work in test-driven development and search how much this emphasizes the contribution of testing and test cases, when their construction anticipates the actual development of the code. The strategy of empirical experiment is still in discussion. The advantage of performing case studies with focus groups would provide detailed feedback and valuable information. But we have to check if the results are able to be generalized to other PLM groups. If this study proves the importance of meeting the NFRs to improve the user acceptance, we should develop methods within PLM delivery methodology that advocates an expressive documentation form of the NFRs.

4. Related works

Improving user acceptance is an important issue in industry. The issues met to install PLM systems may not be very different from the ones met to maintain an Information System, to personalize ERP or to create new market product. The literature proposes different research approaches to address user acceptance. Some approaches consist in surveys with the goal to assess system acceptance. For example, [16] defines a system acceptance indicator (SAI) and [18] and [5] propose similar approaches. The activity

of interest [18] is the integration of an extant soft-system (human workforce) and the modified hard-system (IT system) [5]. Our approach is new because we adopt a requirement-driven point of view, whereas the aforementioned approaches focus on social and psychological and technical aspects such as HC interaction within system implementation. The importance of clarifying non-functional requirements is also a topic of interest in different areas of Requirements Engineering. For example, [10], [11] proposes to represent the NFRs as clouds. [15] used similar goal representation to visualize quality components. [3] [4] propose to represent the non-functional requirements by goals according to the decomposition methods of [3] and [1]. [14] focussed on user oriented non-functional requirements but did not mention user acceptance. Potts [21] used the term 'fitness for use' to designate the concept. 'Fitness' has been studied under different perspectives and was considered as a concept per se that can be modeled by [20]. Melnik's study [17] focused on the use of tables called Fit user acceptance tests for specifying functional requirements. Our approach is different because we are not starting from the user acceptance test to clarify functional requirements. We wish to get enough expertise to know how to specify requirements, in particular non-functional requirements, such that they will favourably impact user acceptance. We could also refer to methods like the AMUSE methods [6], [7], [8] where the results are not tailored to the medical domain or discussions on the nature of NFRs like in [12]. In [7] the authors show an approach that was designed to appraise and measure the users' (future) satisfaction in the requirements engineering phase. In this phase little systematic guidance exists on how to evaluate the effects of features on user satisfaction early on, and on how to contribute the results to the development process. The AMUSE approach [7], [8] claims to close this gap. It helps requirements engineers and product managers to select the most promising features, i.e., the ones that will satisfy the user the most, already in the requirements phase. The approach does not differ between functionality and non-functionality but deals with both in the general term feature. The approach uses a standardized user satisfaction measurement device (questionnaire), a feature appraisal and prioritization methodology, and some tool support. For evaluation of user acceptance, the standardized questionnaire could be used in the future to evaluate the effect of NFRs on user acceptance.

5 Conclusion

Experience with user acceptance of PLM system shows the need of meeting first the quality requirements. We believe that the user acceptance may depend on how well the NFRs are developed. We believe empirical research is necessary to evaluate the impact and efficiency of expressing NFRs in a more formal way. Also an in-depth research is needed to increase understanding of the interplay between NFRs and user acceptance. We recommend the development of a process model (methodological framework) for development of NFRs. Another important issue is how to effectively guide the elicitation of non-functional requirements and check their correctness in view of user acceptance. The research should ensure maximum user acceptance of the PLM implementation.

References

1. Castro, J., Kolp, M., Mylopoulos, J.: Towards Requirements-Driven Software Development Methodology: The Tropos Project, Information Systems (2002)
2. CIMdata, <http://www.cimdata.com> (2003)
3. Chung, L., Nixon B.A., Yu E., Mylopoulos J., Non-Functional Requirements in Software Engineering, Kluwer Academic Publishers, Boston (2000)
4. Chung, L., Nixon B.A., Yu E., Dealing with Change: An approach Using Non-Functional Requirements, Proceedings of the Second International Symposium on Requirements Engineering, York, England, Springer Verlag London Limited, Requirements Engineering Journal, p. 238-260 (1996)
5. Checkland, P.: Systems Thinking, Systems Practice – Includes a 30 year retrospective, John Wiley and Sons, Chichester, USQ, ISBN 0 471 98606 2 (1999)
6. Doerr, J., Kerkow, D.; Landmann, D.: Supporting Requirements Engineering for Medical Products - Early Consideration of User-Perceived Quality, In: Association for Computing Machinery (ACM): 30th International Conference on Software Engineering. ICSE'2008. Los Alamitos : IEEE Computer Society, pp. 639-647 (2008)
7. Doerr, J.; Hartkopf, S.; Kerkow, D. ; Landmann, D.; Amthor, P.: Built-in User Satisfaction - Feature Appraisal and Prioritization with AMUSE, In: Sutcliffe, Alistair (Ed.), Jalote, Pankaj (Ed.) ; IEEE Computer Society: 15th IEEE International Requirements Engineering Conference. Los Alamitos : IEEE Computer Society, Proceedings 101-110 (2007)
8. Doerr, J.; Kerkow, D.; Koenig; T., Olsson; T.; Suzuki, T.: Non-Functional Requirements in Industry - Three Case Studies Adopting an Experience-based NFR Method, 13th IEEE International Requirements Engineering Conference. (2005)
9. Eigner, <http://vpe.mv.uni-kl.de/cms/index.php?id=274> (2005)
10. González-Baixauli, B., Sampaio do Prado Leite, J.C., Mylopoulos, J.: Visual Variability Analysis for Goal Models. Requirements Engineering Conference, 198-207, (2004)
11. González-Baixauli, B., Laguna, M.A., Sampaio do Prado Leite, J.C.: A Meta-model to Support Visual Variability Analysis, First International Workshop on Variability Modelling of Software-intensive Systems, Limerick, Ireland (2007)
12. Glinz, M., On Non-Functional Requirements, International Conference on RE (2007)
13. ISO/IEC 9126-1: Software Engineering - Product Quality - Part 1: Quality Model (2001)
14. Keller, R.K., Schauer, R. Design Components: Towards Software Composition at the Design Level, Proceedings of International Conference on Software Engineering, Kyoto, Japan, pp. 302-311 (1990)
15. Lapouchnian, A., Yu, Y., Mylopoulos J., Liaskos S., Sampaio do Prado Leite J.C.: From stakeholder goals to high-variability software design, <ftp.cs.toronto.edu/csrg-technical-reports/509>. Tech. rep., University of Toronto (2005)
16. Lehane, P., Huf, S.: Towards understanding system acceptance: the development of an assessment instrument and work practice, Proc. of OZCHI, Canberra, Australia (2005)
17. Melnik, G., Read, K., Maurer F.: Suitability of fit user acceptance tests for specifying functional requirements: Developer perspective. In Extreme programming and agile methods - XP/Agile Universe 2004, pages 60–72 (2004)
18. Mwanza, D., Towards an Activity-Oriented Design Method for HCI research and practice, Knowledge Media Institute, The Open University, Walton Hall, United Kingdom (2002)
19. Rohleder, C.: Visualizing the Impact of Non-Functional Requirements on Variants– A Case Study, International Conference on Requirements Engineering, REV'08, Barcelona (2008)
20. Salinesi, C., Rolland, C., Fitting Business Models to Systems Functionality Exploring the Fitness Relationship., Proceedings of the 15th Conference on Advanced Information Systems Engineering (CAiSE'03), Klagenfurt/Velden, Austria (2003)
21. Potts, C.: Fitness for Use: The System Quality that Matters Most, International Workshop on Requirements Engineering: Foundation of Software Quality, Barcelona (1997)