Exploring Perceived Quality in Software Organizations

Jussi Kasurinen[1], Ossi Taipale[1], Jari Vanhanen[2] and Kari Smolander[1]

[1]Software Engineering Laboratory
Lappeenranta University of Technology
Lappeenranta, Finland
jussi.kasurinen | ossi.taipale | kari.smolander@lut.fi

[2]Software Business and Engineering Institute
Aalto University
Espoo, Finland
jari.vanhanen@hut.fi

Abstract— **Software projects have four main objectives; produce required functionalities, with acceptable quality, in budget and in schedule. Usually these objectives are implemented by setting requirements for the software projects, and working towards achieving these requirements as well as possible. So how is the intended quality handled in this process of pursuing project goals? The objective of this study is to explore how organizations understand software quality and identify factors which seem to affect the quality outcome of the development process. The study applies two research approaches; a survey with 31 organizations and in-depth interviews with 36 software professional from 12 organizations for identifying concepts that affect quality. The study confirms that the quality in software organization is a complex, interconnected entity, and the definitions of desired and perceived quality fluctuate between different process stakeholders. Overall, in many cases the software organizations have identified the desired quality, but are not communicating it properly.**

Keywords- software quality, quality characteristics, quality goals, mixed method study

I. Introduction

Software quality is a composition of different attributes, with the importance of these attributes varying between different types of software products. For example, the desired or important quality characteristics between a game on a mobile phone and control software of an airplane surely have a big difference. How do organizations actually perceive what the quality they require from their products is and what aspects in the development and testing affect the perceived quality outcome?

The main objectives of software engineering include reduction of costs and improvement of the quality of products [1]. To reach the quality objectives in the product, an organization needs to identify their own quality i.e. those quality characteristics which are important for them. After identifying their preferred quality, the next action would be to find the factors in development and testing, which affect these quality characteristics, and ensure they work as intended.

A model that in this sense attempts to specify the different characteristics of quality is the revised software product quality model, as introduced in the forthcoming ISO/IEC 25010 standard [2]. According to the standard, software quality expresses the degree to which the software product satisfies the stated and implied needs when used under specified conditions. In the model, quality consists of eight characteristics, which are functional suitability, reliability, performance efficiency, operability, security, compatibility, maintainability, and transferability. These characteristics are further divided into 38 subcharacteristics, such as accuracy or fault tolerance, which aim to define the quality in measurable terms. In addition, in software business the quality is related both to the development and testing. In the ISO/IEC 29119 standard [3], software test process is defined to comprise of layers, such as organizational test level and test management level. In our study, these standards describe the research subject, software product quality and software testing in organizations.

Testing has a big influence on quality in software business. Testing is also one of the biggest expenses in software development [4]. In one estimate [5], software producers in United States lose annually 21.2 billion dollars because of inadequate end-product quality. Because of the economical importance of software quality, it is important to understand how organizations understand software quality and how organizations decide on quality requirements. The identification of how organizations perceive quality, i.e. which quality characteristics they consider important, and how the quality requirements are catered, helps them to concentrate on essential parts when improving process outcomes from the viewpoint of quality assurance.

However, this task is not easy, as the development and test processes include many concepts which all have possibility to affect the quality in practice. There are several viewpoints by different process stakeholders, with a different perception on what are the important quality characteristics. In this study we explore these concepts and viewpoints in different types of software organizations to understand *how software development and testing affect the perceived quality of end-product* and *which process activities have a major impact on the perceived software quality outcome*.

The paper is structured as follows. First, we introduce comparable studies and related research in Section 2. Secondly, the research process with the quantitative survey method and the qualitative grounded theory method are described in Section 3. The results of the study are presented in Section 4. Finally, discussion and conclusions are given in Sections 5 and 6.

II. Related research

Software quality is defined in the software product quality standard ISO/IEC 25010 as a combination of different quality characteristics such as security, operability and reliability. However, it is evident that there are also several different approaches on studying quality and quality concepts in the software engineering. So how can something that has so abstract definition as quality be measured or defined for research?

For example, Garvin [6] has discussed the definitions of quality and made extensive definition work for establishing what the quality actually is and how it affects product concepts such as profitability or market situation. Garvin defines five different definitions for quality; transcendent, product-based, user-based, manufacturing-based and value-based definition. Even though they define the same phenomena, product quality, they vary greatly. For example, transcendent quality is "innate excellence", which is an absolute and uncompromising standard for high achievement that cannot be precisely defined, but surely is identified if present. On the other hand, user-based quality is the more common "satisfies user needs" definition, whereas the manufacturing-based definition promotes conformance to the product requirements. Garvin also discusses the different definitions by mentioning that it also explains why different people seem to have different opinion on what is quality; they tend to apply the definition they are most familiar with.

The different aspects and definitions of quality also mean that the measurement of software quality has some considerations. A paper by Jørgensen [7] introduces three assumptions for establishing measurement for software quality: there are no universal quality measurements but meaningful measures for particular environments, secondly, widely accepted quality measurements require maturity in research, and thirdly, quality indicators predict, or indirectly measure quality. In short, Jørgensen establishes that there are no universal measurements, but the approaches using quality indicators – characteristics and attributes – can be used to approximate or predict software quality. Given the perspective of our study, this is in line with our approach of observing and studying the perceived quality and quality-affecting aspects of software process.

Based on the Jørgensen [7] discussion concerning quality indicators and discussion regarding definition of quality by Garvin [6], it seems that applying the classification used in ISO/IEC 25010 would be feasible measurement method. For the survey and qualitative study we also decided to apply literature review to identify different software process activities which would be interesting from the viewpoint of quality. These activities would be called seed categories [8] for the study and form the basis for the survey questionnaire.

For the compilation seed categories [8] in testing, we applied our earlier research results and observations [9] in test processes. Based on our prior research and for example, study by Hansen et al. [10], it is evident that the business orientation affects the testing organization: product oriented organizations should adopt a formal planned testing process and service oriented organizations should adopt a flexible testing process. If the business orientation has an influence on a testing organization, does it have a similar influence on perceived end-product quality? To study this, the construct product/service orientation, was accepted to the seed categories. In addition, Lin et al. [11] also state that quality problems are not only a function of the product or service itself, but also of the development processes. Therefore, constructs describing the development and testing processes and overall process environment were included in this study.

A paper by Boehm and Turner [12] discusses how the applicability of agile [13] or plan-driven methods depends on the nature of the project and the development environment. Boehm and Turner have developed a polar chart that distinguishes between agile methods and plan-driven methods. Abrahamsson et al. [14] writes that agile thinking emerged because software intensive systems were delivered late, over budget, and they did not meet the quality requirements. Therefore the influence of the software development method on perceived quality characteristics was included to the topics of interest.

According to Kit [4], the size and the criticality of the systems among other things emphasize software testing. Also Boehm and Turner [12] select criticality as one of factors affecting the choice of the software development method. Therefore criticality was accepted to our seed categories to see whether it has also an effect on the perceived end-product quality, or preferred quality characteristics.

Guimaraes et al. [15] discusses customer participation in software projects. Customer participation seems to improve specifications of the system and thereby it assists project towards satisfactory outcome. Customer participation and trust between customer and supplier were accepted to the categories to explore their influence on perceived quality.

Based on the literature and our previous studies [9,16,17] we understand that there is a multitude of feasible approaches on studying quality and the concepts that could explain the quality in software processes. Therefore identifying the process activities, which have strong impact on quality outcome, would be complicated. Different organizations, even projects within one organization, may weigh quality characteristics differently and the product quality seems to be related to several, if not all, software engineering concepts in some level.

III. RESEARCH METHOD

Based on literature research, the assessment of quality factors and collecting comparable data on perceived quality in varying organizations was known to be difficult. We decided to approach the problem by applying methods to obtain both statistical and observational data from the organizations, from several viewpoints of software development.

Table 1: Description of the OUs participating in the study

OU	Business	Company size[b] / Operation	Participation
Case A	Modeling software developer	Large / International	Survey, Interviews
Case B	MES[a] producer and logistics service systems provider	Medium / International	Survey, Interviews
Case C	ICT consultant	Small / National	Survey, Interviews
Case D	Maritime software system developer	Medium / International	Survey, Interviews
Case E	Internet service developer and consultant	Small / National	Survey, Interviews
Case F	Safety and logistics system developer	Medium / National	Survey, Interviews
Case G	Financial software developer	Large / National	Survey, Interviews
Case H	ICT developer and consultant	Large / International	Survey, Interviews
Case I	Financial software developer	Large / International	Survey, Interviews
Case J	SME[b] business and agriculture ICT service provider	Small / National	Survey, Interviews
Case K	Logistics software developer	Large / National	Survey, Interviews
Case L	MES[a] producer and electronics manufacturer	Small / National	Survey, Interviews
Other 19 Case OUs	Varies: from software service consultants to organizations developing software components for their own hardware products.	Varies	Survey

[a]Manufacturing Execution System [b]as defined in [20]

We decided to apply two different approaches to validate our own data and further enable us to confirm our findings. To achieve this, we designed a theme-based interview and a survey to collect data on quality concepts; the survey collected data on several organizations to gain a perspective on the industry field as a whole, while the interviews collected considerations of the individual organizations. In the survey, we collected data from 31 software organizations, and in the interviews, we interviewed 36 software professionals from 12 different organizations, in topics such as the test process, test methods and quality in testing. The contacted organizations are summarized in Table 1 and the data collection rounds in Table 2. The themes of the interviews and the questionnaire forms are available at http://www2.it.lut.fi /project/MASTO/.

Combining quantitative and qualitative analyses is a form of methodological pluralism. Methodological pluralism means that the applied study approach does not apply one "correct" method of science, but many possible methods that complement each other [18]. The results of the phases were compared to each other to enable additional validation of the soundness of the data and the analysis. In addition, the population of the study was observed in organizational unit (OU) level. The standard ISO/IEC 15504 [19] specifies an organizational unit as a part of an organization that is the subject of an assessment. An organizational unit deploys one or more processes that have a coherent process context and operates within a coherent set of business goals. An organizational unit is typically a part of a larger organization or company, although in small businesses, the organizational unit may include the entire company. This way the comparison between large, multinational company and small, local, operator became feasible for the purposes of this study.

A. Data Collection

For the interviews we had selected 12 OUs, which represented different software domains, company sizes [20] and operating scales. These 12 organizations were collected from our industrial partners, and supplemented with additional organizations by researchers to represent different types of software business. The selection criteria were that the OU produced software products or services, or offered software-production related services as its main source of income, in a professional and commercial manner. We also accepted only one OU per each company to avoid bias of over-weighting large companies or causing bias from certain types of business practices

All the interviewed case organizations also participated in the survey, for which 19 additional organizations were

Table 2: Organization of data collection rounds

Collection phase	1) Semi-structured interview	2) Structured survey with Semi-structured interview	3) Semi-structured interview
Number of participants	12 focus OU interviews	31 OUs, including 12 focus OUs	12 focus OU interviews
Participamt roles	Designer or Programmer	Project- or Testing manager	Tester or Programmer
Description of participants	The interviewee was responsible for or had influence in software design.	The interviewee was responsible for sofware project or testing phase for software product.	The interviewee was dedicated tester or was responsible for testing the software product.
Focus themes	Design- and Production methods, Testing strategy and -methods, Agile methods, Standards, Outsourcing, Perveiced quality	Test processes and tools, Customer participation, Quality and Customer, Software Quality, Testing methods and -resources	Testing methods, Testing strategy and – resources, Agile methods, Standards, Outsourcing, Test automation and – services, Test tools, Perceived quality, Customer in testing

selected to enhance the statistical relevance. The selection of supplemental OUs was based on probability sampling, randomly picking organizations out of our contacts. The final selection was confirmed with a phone call to check that the OU really belonged to the specified population. Out of the contacted 30 additional OUs, 11 were rejected because they did not fit the population criteria despite of the source information.

The data collection sessions for the survey and interviews lasted approximately an hour and they were recorded for further analysis. The interviewees were selected based on the recommendations from the OU, an emphasis being on the responsibilities and job description of the employee. Additionally, we required that the interviewees should be working in the same project team, or contribute to the same software product, in addition of working in the same OU. In two out of the 36 qualitative interviews, the interviewed organization opted to select two persons for interview, as they considered that they did not have a sufficiently experienced or otherwise suitable individual worker at their disposal. The interviewees were also allowed access to the interview questions before the actual interview. We also did not forbid discussion between prior interviewees nor did we encourage it. Additionally, in one occasion in the first phase we allowed the OU to supplement their first round answers as the interviewee had thought that the given answers lacked relevant details. The data collection was done by three researchers during winter of 2008 to summer 2009.

Structurally, the interviews were implemented with a list of semi-structured questions regarding software testing, quality concepts and software process-themed questions. The interviews included such themes as development methods, agile practices, test resources, test automation and perceived quality. The themes were also related to the set of seed categories [8], which contained essential stakeholders and leads from the literature review [21]. Our aim was to further develop these seed categories based on the observations made in organizations to include the practical aspects that effect software quality.

The first round of interviews included software designers. Our intention was to test whether our prior studies and the observation made on software processes (for example [16, 17]) were still valid. Another objective was to see if our seed categories for this study were selected so that they would yield relevant results.

In the second round interviews the project and test managers were targeted with both qualitative and quantitative instruments. The twelve OUs participating in the first and third round of the qualitative analysis also participated on the survey, which was supplemented with qualitative themes. During the second round, our objective was to collect data on the organization as a whole, as our interpretation was that the managers were in a better position to estimate organizational concepts such as policy effects, overall process, and quality concerns, in contrast to the desired situation.

The third interview round focused on software testers. During this interview round, the focus was on the software testing phases, testing tools and quality aspects in the testing work, further discussing some of the second round topics. Based on the answers we were able to analyze the practical testing work and the effect of quality aspects to the actual testing work.

B. *Data analysis on survey*

In the quantitative part of the study, the survey method described by Fink and Kosecoff [22] was used as the research method. For the selected approach, methods of data analysis were partially derived from Iivari [23], while the design of the survey instrument was done by the principles derived from Dybå [24]. We used Cronbach alpha [25] for measuring the reliabilities of the constructs consisting of multiple items, and studied the correlations between software quality and other relevant constructs by using Kendall's *tau_b* correlation [26].

Related surveys can be categorized into two types: Kitchenham et al. [27] divide comparable survey studies into exploratory studies, from which only weak conclusions can be drawn, and confirmatory studies, from which strong conclusions can be drawn. This survey belongs to the category of exploratory, observational, and cross-sectional studies.

C. *Data analysis on interviews*

In the qualitative study we decided to apply the grounded theory method [28, 29, 30]. The grounded theory was first conceived by Barney Glaser and Anselm Strauss [29], but later the original method has diversified into two distinct approaches, introduced in later publications by Glaser [31], and by Strauss and Corbin [30]. The Glaserian grounded theory focuses on observing activities within the environment and relies on emergence that cannot be made fully systematic, whereas Strauss-Corbin is more geared towards systematic examination and classification of aspects observed from the environment. The number of participating organizations, the limited ability to non-intrusively observe the developers while working, and the large amount of data generated by the organizations meant that for classifying and analyzing the data, the Strauss-Corbin approach was considered more feasible to implement in this study.

The grounded theory method has three phases for data analysis [30]. These methods are open coding, where the interview observations are codified and categorized. In this phase, the seed categories are extended with new categories which emerge from the data. It is also possible to merge or completely remove the categories that are irrelevant to the observed phenomena. During the open coding, 166 codes in 12 categories were derived from the 36 interview recordings.

The second phase is called axial coding, in which the relations between different categories and codes within categories are explored. In this phase, the focus is on the inter-category relationships, although some necessary adjustments like divisions or merges may be done to the categories.

The last and third phase of grounded analysis is the selective coding. In selective coding, the objective is to define the core category [28, 30], which explains the observed phenomena and relates to all of the other defined categories. However, in some cases the core category can also be a composition of categories in case one category does not sufficiently explain all the observed effects. In addition, the results may yield useful observations, which explain the observed phenomena, even extending to a model to define the observed activities. In this study, the core category can be characterized as such "umbrella category", which we named as *The Effect of Different Software Concepts on Quality*. We further described the category with five observations that explore the different software process activities and their effect on end-product quality in development process. Finally, based on the study results, we summarize the findings as a grounded theory on feasible approach on enhancing end-product quality.

IV. Results

In this chapter, we present the results from both parts of the study. First we begin with the survey results and then discuss the grounded theory analysis.

A. *Results of the quantitative analysis*

The questionnaire was divided based on the major themes of the overall study; general information of the organizational unit, processes and tools, customer participation, and software quality. We were able to calculate several different constructs, which were then tested for feasibility and reliability with Cronbach alpha (results in Table 3) and Kendall's *tau_b* (results later in Table 4) tests. Complete survey instrument is also available at http://www2.it.lut.fi /project/MASTO/. In the following, we present the constructs, which were confirmed to affect the perceived quality outcome.

1) Building quality in software process

The interviewees were asked to give their in-sight of two claims, *quality is built in development* and *quality is build in testing*, to estimate which is the source of the quality in their products. This item also included an assessment of the ISO/IEC 29119 test levels in existing organizational processes. The standard was estimated by the maturity levels – the appropriateness of the process compared to the process needs – of different test process levels comparable with the definitions of the standard. These levels, *organizational test policy*, *organizational test strategy*, *project test management level*, and *test execution*, measured the sophistication of the current test process in the OU. Based on maturity estimates, the construct *Existing* p*rocess conformance with the testing standard model* was calculated to describe the existing level of the structures similar or comparable with the ISO/IEC 29119 standard process [3] levels. The used scale was a 5-point scale [21] where 1 denoted "fully disagree" (this level is very bad in our organization) and 5 denoted "fully agree" "this level is very good in our organization). The results based on answers are presented in Figure 1.

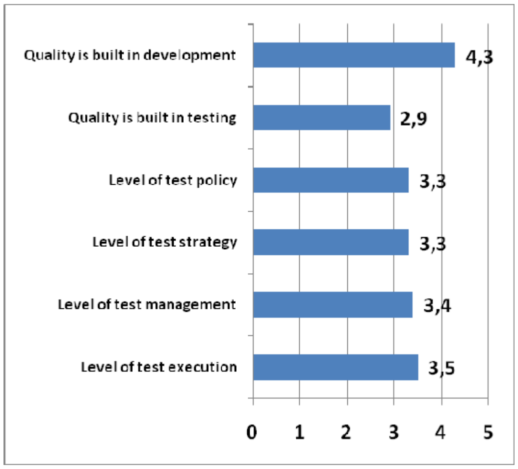

Figure 1. Origin of quality and the realization of the software testing standard ISO/IEC 29119

According to the results, interviewees emphasized that the quality is built in development (4.3) rather than in testing (2.9). Also for the standard, the results are mostly ambiguous in all test process layers, but slightly favor the lower level activities like management and test execution level.

2) Customer participation

The second survey topic was connected to customer participation. This construct, *Customer participation*, described how customers participated in development and testing processes. For customer participation, the constructs were calculated by summing up the answers of the items and dividing the sum by the number of items. From this group, *Customer participation in the general control*, i.e. in the process steering and in decision making in development, reached acceptable Cronbach alpha value only with two items. These items were *our most important customer reviews project management schedules and progress reports made available by us*, and *our most important customer provides domain training to us*. The Cronbach alpha values for these constructs, amongst the other constructs, are listed in Table 3.

Table 3. The reliabilities of different constructs (acceptance level of >0.7)

Variable	Cronbach alpha
Existing process conformance with the testing standard model	.894
Customer participation during the specification phase of the development.	.855
Customer participation during the design phase of the development.	.772
Customer participation during the testing phase of the development.	.742
Customer participation in the general control.	.702
Trust between customer and supplier.	.699
Elaboration of the quality attributes.	.818

Additionally, the construct *Trust between customer and supplier* described the confidence that the behaviour of another organization will conform to one's expectations as a benevolent action. For measuring this construct, the questions were derived from Benton and Maloni [32]. When calculating the Cronbach alpha for the construct *Trust*, an

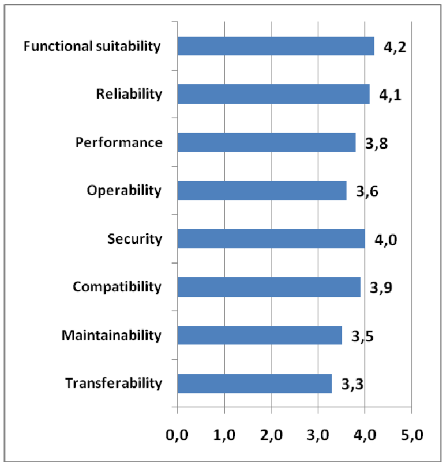

Figure 2. Assessment of fulfilling the different quality characteristics in the end-product

acceptable level was reached with items *our most important customer is concerned about our welfare and best interests* and *our most important customer considers how their decisions and actions affect us*.

3) Quality characteristics and perceived quality

For the third interview topic, interviewees were asked to evaluate the competence level of each ISO/IEC 25010-quality characteristic in their software by a 5-point scale where 1 denoted "this characteristic in our software is taken into account very badly" and 5 denoted "this characteristic in our software is taken into account very well". Interviewees were also advised to leave the attribute unanswered ("this characteristic is irrelevant to our product") if the attribute was not valid for the OU, i.e. if the attribute was irrelevant for the organization. If an organization gave some attribute a high score, it meant that the organization thought that this particular quality characteristic was handled well in the product design and development. The resulting average indicated the perceived level of quality of the organization's product: if organization gave high points to quality characteristics, it was understood that the organization considered their end-product of high quality, if low scores, the organization considered that their product was low quality, or at least not as good as it should be. These results were also used as a construct *perceived overall quality by the organization*. The mean values for all surveyed quality characteristics are included in Figure 2.

Quality characteristics *functional suitability*, *reliability*, *security*, and *compatibility* reached the highest scores, meaning that they were the most well-attended quality characteristics. Even if the results did not vary much (between 3.3 and 4.2), it was indicative that some of the characteristics were generally less attended than others. However, overall all of the attributes were considered at least somewhat important; only in 9 cases (3.6% out of 248 characteristic assessments) the organization considered the assessed characteristic "irrelevant" to their product.

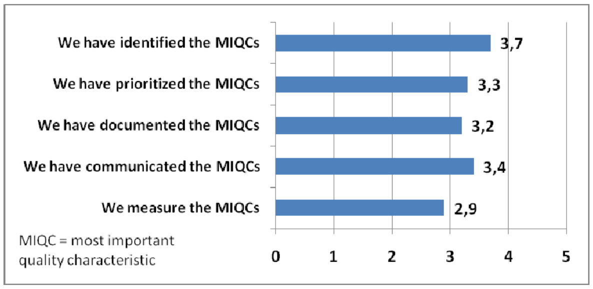

Figure 3. Elaboration of the quality characteristics

In addition of perceiving quality characteristics, the interviewees were asked to evaluate how their organizations elaborated and communicated their quality characteristics. The interviewees were asked to give their in-sight of five claims; *we have (1) identified, (2) prioritized, (3) documented, (4) communicated, (5) we measure the most important quality characteristics.* The construct, *Elaboration of the quality characteristics*, was calculated as the mean of the answers to the claims. Almost all organizations had at least identified their most important quality characteristics (3.7), while measurement and collection of the metrics was not as common (2,9). The results, how organizations elaborate their quality characteristics, are given in Figure 3.

4) Factors for achieving quality characteristics

The effect of different survey constructs was further explored to see how they would correlate with the perceived overall quality of end-product. To achieve this, the Kendall's *tau_b* correlations were calculated between constructs, which were first tested with Cronbach alpha. As based on the Kendall's *tau_b*-analysis, the constructs *Existing process conformance with the testing standard model*, *Elaboration of the quality attributes*, and *Trust between customer and supplier* seemed to positively correlate with the construct *perceived overall quality by the organization* at the 0.01 level. In addition, the influence of some constructs, such as *Customer participation during the design phase of the development*, and *Customer participation in the general control* were almost significant.

Several other constructs were calculated from the data, such as *Software development method* and *Criticality of the OU's end products*, but they did not reach significant correlations and therefore were discarded. The correlations of constructs which had a significant correlation are presented in Table 4, which also includes some insignificant constructs as an example.

Based on these results, the most important factors for achieving better end-product quality and pursuing quality characteristics in software products are closely related to the development process maturity and elaboration of quality goals. Those organizations that had identified and communicated their most important quality characteristics or were confident with their development and test processes were also confident with the levels of their quality characteristics. The organization which thought that the appropriateness and sophistication of their development process was high or said that the identification of the important quality attributes was in a high level, also considered that their products implemented the quality characteristics well. Also aspects such as customer participation in certain parts of the development and trust between stakeholders were also observed to be beneficial.

Table 4. Correlations between different surveyed constructs and the perceived overall quality

Construct	Kendall's tau_b
Software development method	-.195 .158
Criticality of the OU's end products	.171 .226
Existing process conformance with the testing standard model	.505 ** .000
Customer participation during the specification phase of the development.	.120 .377
Customer participation during the design phase of the development.	.231 .092
Customer participation during the testing phase of the development.	.141 .287
Customer participation in the general control.	.261 .057
Trust.	.436 ** .002
Elaboration of the quality characteristics.	.437 ** .001

Kendall's correlation (N=31)

** Correlation is significant at the 0.01 level (2-tailed).

B. *Results from the grounded analysis*

The grounded theory analysis data was collected from 36 interviews held at 12 software producing organizations. We interviewed software designers, project or test managers and testers in three phases. This data was then codified and analyzed, which led to the definition of several categories and factors that were observed to have an effect on the perceived quality and quality output of the product development process, or based on the literature were considered important. In the following sections, we introduce these categories and observations.

The core category, "*The Effect of Different Software Concepts on Quality*", is defined as a composition of seven other categories. These categories were collected from the topics that interviewees mentioned regularly when discussing about the quality aspects and perceived quality in their software process. For example, standardization and the role of the customer in the process were mentioned regularly. In some occasions a category was included to test the possible lead-ins from the survey and literature reviews. For example, the effect, or more precisely the lack of effect, for concepts such as the product/service-orientation or criticality was studied more closely in the qualitative study. A summary of the categories is shown in Table 5.

The category "*Desired Quality Characteristics in design*" explains the ISO/IEC 25010 quality definitions that were considered to be the most important characteristics from the viewpoint of software designers. These quality aspects were most likely those that were used in software process, especially in specification, design and other early development phases. For comparison, the second category "*Desired Quality Characteristics in testing*" explained the quality characteristics that were considered to be the most important from the viewpoint of testers, and subsequently also those, the testing work focused on.

The category of "*Level and Effect of Criticality*" is a two-fold category. First is the level indicator of criticality of the product the interviewed OU is developing. The criticality level scales from 1-5. In this scale, 5 is the highest level meaning "may cause loss of human life", 4 is "may cause bodily harm or great economical losses", 3 "significant economical losses", 2 "small economical losses" and 1 "no effect or user irritation". The scale is similar to other criticality measurements, discussed for example in [32]. The latter part of the category is assessment on how the test process would change if the criticality level of their product increased.

The category of "*Effect of Customer*" defines the effect the customer has on the end-product quality. This category defines the most influential action or generally the possibilities of the customer to affect the quality, in actions such as "extend deadline" or "allow larger budget". It should be noted that in this category, the most potent effect may also be harmful for the end-product quality, like limiting access to the final working environment or requesting unfeasible changes.

The category of "*Applied standards*" lists the standards the organizations are using in either development or test

Table 5: Categories from qualitative analysis

Category	Description
Desired Quality Characteristics in design	The quality characteristics that the software designers consider important.
Desired Quality Characteristics in testing	The quality characteristics that the software testers consider important.
Level and Effect of Criticality	The level of end-product criticality and the effect of how increased criticality would affect testing work.
Effect of Customer	The effect of customer to the end-product quality, and short description on what constitutes this effect to take place.
Applied Standards	The standards that are officially followed and enforced in either software development or test process of the organization.
Effect of Outsourcing	The effect of outsourcing development process activities has to end-product quality, and short description of what constitutes this.
Product/Service-orientation	The distribution of business in interviewed organization divided between product and service-oriented activities. Assessed by the managers.

process. Even though many organizations applied parts of the standards, or followed the process efficiency unofficially by measures derived from standards, in this category only the fully applied, systematically used, standards and competence certificates are listed.

The category of "*Effect of Outsourcing"* defines the effect outsourcing has on the perceived quality of an end-product, including effects like allowing more focus on core products or critical aspects. This category defines the most influential way the outsourcing affects the process outcome quality.

The category of "*Product/Service-orientation"* represents the ratio between product-oriented activities and service-oriented activities in an OU. This approximation is directly taken from the survey interviews with managers.

1) Observations from the data

The observations were developed based on the categories. These observations either define software development concepts that affected the perceived quality in a software process, affected quality in a product or were considered an important item for composing more complex constructs of software quality. All these observations are based on the findings made during the analysis. The summary of the findings which were the basis for the observations, are available in Table 6.

Observation 1: The importance of quality attributes vary between different process stakeholders in an organization.

The first finding confirmed, as suggested by the literature [6], that the conceptions of quality vary even within one project organization. The software designers and testers were asked to rank the ISO/IEC 25010 quality attributes in the order of importance. Although the testers and designers were working on a same organization, the most important attribute was the same only in four case organizations (A, D, K and L) out of twelve. All the same attributes, although not necessarily in the same order, were mentioned by two organizations, cases L and D.

It seems that the designers were slightly focused towards usability aspects such as operability or functional suitability, while testers were focused towards technical attributes like security or reliability, meaning that each participant had a different view on what quality should be.

Observation 2: The effect of product/service-orientation or criticality on the importance of quality attributes is low.

The product-oriented organizations and service-oriented organizations seem to have similar priorities in quality attributes. For example, Case E, which is a fully service-oriented software producer, promotes the same attributes as Case F, which is mostly product-oriented. Similarly Case G, which has a large emphasis on the service-orientation, has

Table 6: Observations from the case organizations

	DQC[a] in design	DQC[a] in testing	Level and Effect of Criticality	Effect of Customer	Applied standards	Effect of Outsourcing	Product/ Service orientation
Case A	Functional suitability, Reliability	Functional Suitability, Security	2: Security, Reliability get more attention	Enhances the quality by participating closely.	ISO9000-seires, ISTQB-certificates for testers	No meaningful effect.	100% product
Case B	Functional suitability, Performance efficiency	Perfomance efficiency, Operability	3: Performance efficiency, Functional suitability get more attention	Enhances the quality by allowing larger expenses.	CMMi	-	100% product
Case C	Maintainability, Functional suitability, Reliability	Reliability, Operability	4: Central aspects get more attention.	Enhances the quality by providing feedback	ISO9000-series, ISO12207	May weaken the quality by causing instability in the process.	80% product, 20% service
Case D	Functional suitability, Operability	Functional suitability, Reliability, Operability	4: Security gets more attention	Enhances the quality by providing feedback	Officially none	May enhance quality.	55% product, 45% service
Case E	Operability, Security	-	1: Central aspects get more attention	Enhances the quality by allowing larger expenses.	Officially none	May weaken the quality by causing instability in the process.	100% service
Case F	Operability, Functional suitability	Reliability, Compatibility, Security	5: Central aspects get more attention	Enhances the quality by participating closely.	ISO9000-series, domain-based certifications.	May weaken the quality by causing instability in the process.	83% product, 17% service
Case G	Reliability, Performance efficiency, Security	Functional suitability, Operability, Performance efficiency, Reliability	3: Security, Functional suitability get more attention	Enhances the quality by providing feedback	ISO9000-series SPICE, ISTQB-certificates for testers	-	40% product, 60% service
Case H	Security, Reliability	Maintainability, Operability, Performance efficiency, Reliability	2: Functional suitability gets more attention	Enhances the quality by providing feedback	CMMi, ISTQB-certificates for testers	No meaningful effect.	50% product, 50% service
Case I	Functional suitability, Performance efficiency, Reliability	Security, Compatibility, Functional Suitability, Reliability	4: Functional suitability, Reliability get more attention	Enhances the quality by participating closely.	CMMi	Enhances the quality by allowing focus on critical aspects.	63% product, 37% service
Case J	Performance efficiency, reliability, Operability	Functional Suitability, Reliability, Performance Efficiency	2: Reliability, Functional suitability get more attention	Enhances the quality by providing feedback	Officially none	No meaningful effect.	100% product
Case K	Functional suitability, Reliability, Performance efficiency	Functional suitability, Reliability, Security	3: Central aspects get more attention.	Enhances the quality by allowing larger expenses.	Officially none	Enhances the quality by allowing focus on critical aspects.	100% product
Case L	Functional suitability, Operability	Functional suitability	3: Central aspects get more attention.	Weakens the quality by requiring late changes.	Officially none	May weaken the quality by causing instability in the process.	75% product, 25 % service

[a] Desired Quality Characteristics

the same two most important attributes in both design and testing as Case J, which is fully product-oriented. The interviews reflected this consideration to some degree. The type of software may change between projects, but the development and testing process is done in a similar fashion in every project.

"[The project type is irrelevant] as we make things the same way in any case if we want to keep any quality." – Tester, Case F

"Quality is built in design with test cases [in all projects]." –Tester, Case G

The criticality of the software product seems to have only a small effect on the test process activities. When asked to reflect on how the development priorities of a software project change in the case of a higher criticality, the main features were considered to gain more attention in five organizations. In the other seven organizations, certain quality aspects, such as functional suitability, reliability or security, gained more attention. Overall, the criticality was not considered to cause major process changes in any organizations.

"Security… and reliability, they still are number one; in this business they always are." –Designer, Case G

"Yes, in some cases the security would be concerned" – Tester, Case D

A clear indicator of the effect of criticality was observed when comparing the cases E, F and K. Case K was a completely product-oriented organization with an average criticality, Case E was a completely service-oriented organization with a low criticality and F a high-criticality product-oriented OU. The differences between software products in these organizations can be considered quite large, but yet the effect of criticality was considered similar; the process becomes more rigid but the approach stays the same.

"I think, within our business space, it [testing process] would stay the same" – Designer, Case K

"Activities should always aim to progress work towards objectives [regardless of what we are doing]." –Designer, Case E

"[Security] is something that is always taken into account… but maybe we should focus more on that." – Designer, Case F

Observation 3: The standards in software testing do not affect the quality characteristics, as they are not widely used in practice even though organizations in general are positive towards standardization.

Testing standards and certifications in the case organizations were rarely applied. The most commonly applied standards were CMMi and ISO9000 models, which both focus on general process quality measurements. In five organizations no standards were followed officially, although some method of measuring process efficiency existed in all organizations.

"ISO9000… well officially we do not have any certificates for it, but that is the one we based our own on." – Manager, Case G

"CMMi reviews… as far as I know they, however, have been internal." – Manager, Case H

As for testing-related standards, the application was even more sporadic. Only in three cases, G, H and L, some form of official testing certification was applied.

"We have one tester who has done it [ISTQB]… he trains the other testers. That seems to work for now." – Tester, Case A

"We have this ISTQB. All our testers as far as I know have done it." – Tester, Case H

"We have testers participating in the ISTQB training." – Tester, Case G

Even though many organizations did allow, or were generally positive towards participating on certification training, the number of testers who had actually acquired a formal certification varied. The level of currently applied test-related standards and certificates seems to indicate that organizations could have use for a new testing standard. This was indicated by feedback given by the interviewees when discussing the purposes of the upcoming ISO29119 standard and the standards currently applied:

"It would help us to have some way to organize testing in a smart way. A prepared model would be ideal." – Tester, Case L

Observation 4: The general impact of a customer to the perceived end-product quality is positive, but a customer is required to either provide resources or commit to the project.

The customer in a software project was generally considered to have a positive impact on end-product quality.

"The feedback from the client is important to have." – Designer, Case H

"It is easier to do [good quality] if the customer is involved." – Manager, Case F

"The customer brings their own set of quality requirements… it [quality] becomes everyone's objective." – Manager, Case G

However, to actually have an impact on the quality, the customer was required either to provide a substantial financial contribution to the project, to give relevant feedback or to commit otherwise to the project, offering insight and contributions to the project along its progress.

"If they want high quality [they increase the project budget]" – Designer, Case K

"Giving feedback is the key [to quality]." – Manager, Case J

"Participation to the specification is the first, the second is acceptance testing to see if everything is done as agreed and the third is communication, meaning comments and such…" –Manager, Case A

"The customer has to be active especially in testing and specification phases." – Manager, Case I

On the other hand, one organization also noted that in some occasions the customer may hinder the quality, for example by requiring late or unfeasible changes to the product without providing enough support to allow such operations.

"If a customer wants something really stupid and pays for it, then we have to do it." – Designers, Case L

"In one case, the customer did not allow us to use their systems, so we could not do the final tests." – Designers, Case L

Observation 5: Outsourcing may cause quality issues smaller organizations.

It seems that the OUs from small companies are cautious to apply outsourced resources in their projects. In our study, the small-company originating cases L, C and E all were uncertain or concerned regarding the quality of outsourced resources. They considered outsourced resources and third-party-based software modules hazardous, or at least challenging to implement in their own projects:

"There always seem to be some problems with modules brought from outside." – Designer, Case L

"If you start from scratch when outsourcing, it fails unless a lot of energy is used to assure it." –Manager, Case E

As a contrast, OUs from large companies – cases K, H, I and L – considered outsourcing to be a feasible option. Generally their opinions seemed more positive, even to the extent of considering it to enhance quality by allowing focusing on central aspects of the software.

"In outsourcing, we can require that in our product, we allow only this and that amount of errors… by partnering, we can easily assure quality in those aspects." –Manager, Case K

"They go through the same test process so I don't see any problems with them." –Designer, Case H

"It does not affect. Bought code is reviewed and integrated similarly as our own." – Designer, Case L

It would seem that the OUs from larger companies do gain benefits from belonging to a larger organization, at least in applying outsourced resources. The rationale for this observation may be that large companies have more power or influence; small companies may be unable to pay similar amounts as larger companies, to get exactly what they want, so they experience more problems. Another viable explanation could be that large organizations may have more experience of outsourcing or at least have more resources to organize and administrate the outsourcing activities.

With outsourcing, the effects of open source software modules in professional software products were also discussed by case organizations D, F, H and I. Case F considered open source resources to be useful, as they allowed the development to create secondary features out of existing components. In their organization, the application of open source resources to non-critical parts of the product was considered to improve overall quality.

"The best open source modules follow standards more closely than some commercial products" –Tester, Case F

Cases D and H expressed similar considerations; Case D had implemented some secondary features with open source resources, while Case H was yet to apply open source but was willing to try should something applicable be found. Case I applied some open source along with other outsourced modules, but was unsure if especially the "opensourceness" had any effect on quality.

V. Discussion

One common theme seems to be the confidence in testing process. Both the survey results and qualitative analysis established that there are some indicators which affect the perceived software quality, such as appropriateness of the testing process in relation to the product, to communication of most important quality characteristics or to customer participation to the development process. Along with the appropriateness of test process, the overall level of standardization seems to have positive effect on quality. However, especially in testing, the existing test processes rarely seem to apply standards to a large degree. Even if the overall attitudes towards test standardization and test certification programs are positive, the application of standards in several studied organizations was still at a too low level to actually have a visible influence on the process.

As for other studied process concepts, the software development method, the product/service-orientation of the organization nor the criticality affected the perceived quality to a large degree; it seems that the product quality can be sufficient with any development method, and that the main criteria for quality characteristics comes from the product domain, not from the criticality of the product. Surely the highly critical software goes through more rigorous testing than that with low criticality, but the importance of quality characteristics is not related to criticality. For example, in the case organizations in finance domain, the most important quality characteristics were reliability, security and functional suitability, regardless whether the application itself was used to service individual users or a large network. The criticality level varied between levels of 2 (small economical losses) and 4 (great economical losses), but the quality goals and importance of quality characteristics, stayed the same. Similarly, a software development method, whether it applied agile practices or traditional design-based approach, nor the product/service-orientation, affected the importance of quality characteristics.

One interesting observation was that designers and testers rarely had similar considerations on the "most important quality characteristics". This phenomenon surely has an effect on pursuing quality in the software products, as the elaboration of desired quality did correlate with the improvement of quality. Overall, it seems that the desired quality characteristics are usually not identified nor communicated strongly enough throughout the organizations, as the identified quality characteristics were usually based on personal preferences, similarly as discussed by Garvin [6].

In our study we have established that there are factors which affect the perceived end-product quality. The participation of the customer, defining and communicating the quality objectives, and creating a feasible software process, which has addresses the needs of desired quality, were established to have positive correlation with the perceived quality. Summarizing these findings to one grounded theory, it would seem that *creation of appropriate, systematic test and development processes, promoting active participation from customers, and identifying and communicating the desired quality characteristics through the organization offer a good starting point for pursuing better quality in end-products*.

Applying two different approaches allowed this study to observe the quality from different viewpoints, and overall do comparisons between different sources of data. In this sense, the threats to validity for the results of this study are low, but there are some concerns for the study validity.

First of all, in survey the sample size of 31 organizations may seem somewhat limited. However, similarly as in [23], the sample size is small but sufficient if analyzed correctly. In our study, the threat of overfitting the data - over-representing certain sub-groups of participants - was addressed by selecting the organizations to represent different software domains and types of organizations, and triangulating the data with different approaches. Also in terms of the number of organizations, a paper by Sackett [34] discusses the conceptualization of signal-to-noise-ratio in statistical research. Their approach to define confidence as based in practicality of observations: *confidence = (signal / noise) * square root of sample size*. In practice, this indicates that the confidence for the result being non-random weakens if the amount of noise increases while signal decreases. In the Sackett model, the attributes are abstracted, meaning that the noise can be considered to be uncertainty on any source of data. The concept is that the confidence in the survey data increases the validity of the study. Our study addressed this problem by organizing face-to-face interviews with clients and applied researchers as the interviewers to ensure that the interviewees understood the questions and terminology correctly. Therefore in Sackett terms it can be argued that our signal was very good and noise low, so the overall confidence should be good.

As for the validity of the qualitative parts of this study, there are some threats that should be addressed [35]. For example, Golafshani [36] discusses the validity and reliability of qualitative research, and makes some notions on the topic. First of all, the reliability and validity in a qualitative study are not the same, traditionally mathematically proved concepts, as in a quantitative study, but rather a conceptualization of trustworthiness, rigor and quality of the study. To increase the validity in qualitative study, the research must eliminate bias and remain truthful to the observed phenomena. Similarly, a grounded theory is not a theory in mathematical sense, establishing a universal truth or causality, but rather a generalization of observations, offering guidelines and considerations for best practices when taken outside of the study scope [30].

The concept of research validity has been taken even further by Onwuegbuzie and Leech [37], who create model for threats of validity in qualitative studies. They summarize that in qualitative research the threats to internal validity and external credibility are context sensitive; in quantitative studies the objective is to minimize the amount and effect of invalid data, but in qualitative studies, the threats have to be individually assessed based on truth value, applicability, generalizability and such. As these measurements are interpretative, the validity should be addressed by providing enough documentation on the research process, analysis methods and reasoning for the presented results.

As mentioned in the study by Jørgensen [7], in measuring and comparing quality, there are no universal measurements. There is only a possibility to produce relevant results within the context. Obviously our study has the same limitations, but for our research objectives in observing perceived quality in software development, our intention was to observe and identify software engineering aspects which should be used to define general guidelines on how the quality should be addressed or improved. In this objective, we managed to identify the effect of several components, such as the role of the customers, product criticality, process appropriateness or development method.

VI. Conclusions

In this paper we have presented our multi-method study on observing perceived quality in software organizations. Our results indicate that there are several concepts which affect the perceived software quality, such as customer, outsourcing or communication between stakeholders. On the other hand, it also seems that several process concepts such as criticality, product/service-orientation, development method or open source approach do not have any major effect on the perceived end-product quality. It is obvious that high-criticality products do have fewer faults than those on the low end of the scale, but the desired quality characteristics do not change significantly between the criticality levels. Another important finding was that even within one organization the importance of quality attributes seems to have variation between different stakeholders and viewpoints of the software process.

In the majority of the organizations, the testers and designers considered quite differently of what are the “most important” or “most desired” quality attributes of the product. It seems that the desired objectives, and desired quality, must be communicated clearly to reach every stakeholder in the organization as the desired quality and quality requirements are not obvious, “common sense” aspect. Overall, it seems that generally feasible approach in pursuing better end-product quality would be to create systematic test and development processes, promote active participation of customers and identify and communicate the desired quality characteristics through the organization.

As for future work, it is evident that concepts which in this study were observed to have correlation with perceived quality are also closely related to software process improvement. It would be beneficial to study how these observations could be integrated into a process improvement project, and empirically validate the study-established factors, which had observable effect on the perceived end-product quality.

ACKNOWLEDGMENT

This study was supported by the ESPA-project (http://www.soberit.hut.fi/espa), funded by the Finnish Funding Agency for Technology and Innovation, and by the companies mentioned in the project pages.

REFERENCES

[1] Osterweil L. J. (1997). "Software processes are software too, revisited: an invited talk on the most influential paper of ICSE 9," in International Conference on Software Engineering, Proceedings of the 19th international conference on software engineering, Boston, pp. 540-548.

[2] ISO/IEC (2008). ISO/IEC 25010, Softaware Engineering – Software product Quality Requirements and Evaluation (SQuaRE) Quality Model.

[3] ISO/IEC (2009). ISO/IEC 29119, Software Testing Standard.

[4] Kit E. (1995). Software Testing in the Real World: Improving the Process. Reading, MA: Addison-Wesley.

[5] Tassey G. (2002). The Economic Impacts of Inadequate Infrastructure for Software Testing. U.S. National Institute of Standards and Technology report, RTI Project Number 7007.011.

[6] Garvin D.A. (1984). "What Does "Product Quality" Really Mean?", Sloan Management Review, Issue 4, pages 25-43.

[7] Jørgensen M. (1999). "Software quality measurement", Advances in Engineering Software, Vol 30(2), pages 907-912.

[8] Miles M.B. and Huberman A.M. (1994). Qualitative Data Analysis. Thousand Oaks, CA: SAGE Publications.

[9] Taipale O. and Smolander K. (2006). "Improving Software Testing by Observing Practice", Proceedings of the 5th ACM-IEEE International Symposium on Empirical Software Engineering (ISESE), 21-22 September 2006, Rio de Janeiro, Brazil, IEEE, pp. 262-271.

[10] Hansen M. T., Nohria N. and Tierney T. (1999). "What's Your Strategy for Managing Knowledge?," Harvard Business Review, vol. 77, pp. 106-116.

[11] Lin J., Brombacher A.C., Wong Y.S. and Chai K.H. (2004) "Analyzing Quality Information Flows in Cross-company Distributed Product Development Processes," in Engineering Management Conference, Singapore, pp. 973-977.

[12] Boehm B. and Turner, R. (2003), "Using Risk to Balance Agile and Plan-Driven Methods," Computer, vol. June, pp. 57-66.

[13] Fowler M. & Highsmith, J. (2001), The Agile Manifesto.

[14] Abrahamsson P., Salo O., Ronkainen J. & Warsta J. (2002), Agile Software Development Methods: Review and Analysis. VTT Publications 478.

[15] Guimaraes T., McKeen J. D. & Wetherbe J. C. (1994) The Relationship Between User Participation and User Satisfaction: An Investigation of Four Contingency Factors. MIS Quarterly/December 1994, 26.

[16] Karhu K., Repo T., Taipale O. and Smolander K. (2009). "Empirical observations on software testing automation", IEEE Int. Conf. on Software Testing Verification and Validation, Denver, USA.

[17] Kasurinen, J., Taipale, O. and Smolander, K. (2009). "Analysis of Problems in Testing Practices", Proc. 16th Asia-Pacific Conference on Software Engineering, 1.-3.12., Penang, Malaysia.

[18] Hirschheim R. A. (1985), 'Information systems epistemology: an historical perspective', in R. H. E Mumford, G Fitzgerald, T Wood-Harper (ed.), Research Methods in Information Systems, North-Holland, Amsterdam.

[19] ISO/IEC (2002). ISO/IEC 15504-1, Information Technology - Process Assessment - Part 1: Concepts and Vocabulary, 2002.

[20] EU (2003), "SME Definition," European Commission.

[21] Seaman C.B. (1999). "Qualitative methods in empirical studies of software engineering", IEEE Transactions on Software Engineering, vol. 25, pp. 557-572.

[22] Fink A. and Kosecoff, J. (1985). How to conduct surveys A Step-by-Step Guide. Newbury Park, CA: SAGE Publications, Inc..

[23] Iivari J. (1996). "Why Are CASE Tools Not Used," Communications of the ACM, vol. 39, pp. 94-103.

[24] Dybå T. (2000). "An Instrument for Measuring the Key Factors of Success in Software Process Improvement," Empirical Software Engineering, vol. 5, pp. 357-390.

[25] Cronbach L. J. (1951), "Coefficient Alpha and the Internal Structure of Tests," Psychometrika, vol. 16, pp. 279-334.

[26] SPSS (2010). SPSS 17.0. Chicago: SPSS Inc. http://www.spss.com. Referenced 29.3.2010.

[27] Kitchenham B. A., S. L. Pfleeger, L. M. Pickard, P. W. Jones, D. C. Hoaglin, K. E. Emam, and J. Rosenberg (2002). "Preliminary Guidelines for Empirical Research in Software Engineering," IEEE Transactions on Software Engineering, vol. 28, pp. 721-733.

[28] Paré G. and Elam J.J. (1997). Using Case Study Research to Build Theories of IT Implementation. The IFIP TC8 WG International Conference on Information Systems and Qualitative Research, Philadelphia, USA. Chapman & Hall.

[29] Glaser B. and Strauss A.L. (1967). The Discovery of Grounded Theory: Strategies for Qualitative Research. Chicago: Aldine.

[30] Strauss A. and Corbin J. (1990). Basics of Qualitative Research: Grounded Theory Procedures and Techniques. SAGE Publications, Newbury Park, CA, USA.

[31] Glaser B.G. (2002), "Constuctivist Grounded Theory?", Forum: Qualitative Social Research (FQS), Vol 3(3).

[32] Benton W. C. & Maloni M. (2004) The influence of power driven buyer/seller relationships on supply chain satisfaction. Journal of Operations Management, 22.

[33] Bhansali, P.V. (2005). "Software safety: current status and future direction", ACM SIGSOFT Software Engineering Notes, Vol 30(1).

[34] Sackett D.L. (2001). "Why randomized controlled trials fail but needn't: 2. Failure to employ physiological statistics, or the only formula a clinician-trialist is ever likely to need (or understand!)". CMAJ 165 (9): 1226–37, October.

[35] Robson C. (2002). Real World Research, Second Edition. Blackwell Publishing,

[36] Golafshani N. (2003). "Understanding Reliability and Validity in Qualitative Research", The Qualitative Report, Vol 8(4), pages 596-607.

[37] Onwuegbuzie A.J. and Leech N.L. (2007). "Validity and Qualitative Research: An Oxymoron?", Quality and Quantity, Vol 41(2), pages 233-249. DOI: 10.1007/s11135-006-9000-3.