Available online at www.sciencedirect.com

ScienceDirect

Procedia - Social and Behavioral Sciences 191 (2015) 969 – 974

Procedia
Social and Behavioral Sciences

WCES 2014

Empirical Analysis of Quality Models in Practice in Small IT Companies in SEE Region

Florinda Imeri[a*], Ljupcho Antovski[b] , Mentor Hamiti[c]

[a] *State University of Tetovo, Ilindenska bb, Tetovo, 1200, Macedonia*
[b] *University Ss. Cyril and Methodius, Rugjer Bishkovikj 16, 1000, Skopje, Macedonia*
[c] *South East European University, Ilindenska 335, 1200 Tetova, Macedonia*

Abstract

A significant problem of the global industry is the costs of software problems or errors which affects not only software developers but their customers and the end users of the software. Systems fail because of inadequate performance, security, reliability, usability, or precision, commonly known as quality characteristics. Quality is one of the main assets which enable a firm to enhance its global competitive position. Software metrics and quality models help increase the performance by measuring software quality. Different software quality models are proposed by different researchers. A number of well known quality models are used to build quality software. Using the survey technique this study investigated the use of quality models in practice and the testing practices, conducted among ten software companies in SEE region, Macedonia and Kosovo. This research relies upon collection of empirical data, "qualitative" from interviews with managers regarding the use of quality models, quality assurance techniques, and "quantitative" from questionnaire, filled in by practitioners.

Keywords: software quality, quality models, metrics, testing techniques;

1. Intruduction

A significant problem to IT industry is the cost of software problems or errors which affect not only to producers but to consumers of the software as well. These are costs related to the lack of quality of software. Software quality impacts a number of important factors in our daily lives, such as economy, personal and national security, health,

* Florinda Imeri, Tel.+389 70 834 873
E-mail address: florinda.imeri@unite.edu.mk

doi:10.1016/j.sbspro.2015.04.490

and safety (Naik & Tripathy, 2011). Quality encompasses all phases of a development process-from specification to final delivery, all these performed with highest possible standard. To support quality management of software systems a comprehensive model of software product quality should be available. By combining and relating software quality characteristics and sub-characteristic, organizations, such as ISO and IEEE, have standardized software quality models (Khosravi & Guéhéneuc, 2004). A multitude of quality models have been proposed and applied showing different degrees of success. Different surveys on the field have shown that QM usage in practice seems to differ widely. In region, detailed information regarding the usage of quality models and the problems that software companies face while building software products in different domains regarding quality are missing. Following we will present the findings of a survey on quality models, in general, in practice conducted among ten software companies in SEE region, Macedonia and Kosovo. Our aim was to find information regarding the difficulties that IT companies have while trying to deliver products in time, which QMs are currently in use and which quality assurance techniques are applied. Companies in which we have conducted our survey cover several domains of software development; thus we could not focus on special aspects of software product quality.

During this research, we used the following methods:

- A literature study involving review of the available scientific papers related to the areas of quality, software quality and software quality models
- Face to face interview with project managers
- A survey with a questionnaire filled in by developers and managers, and
- Analysis of the data taken from the survey and interviews

The paper is structured as follows. Section 2 is on similar works, Section 3 and Section 4 are an overview on quality and quality models, Section 5 is a description of the study design , methods of collecting the data and research topics, Section 6 reports and discuses the results of the survey while Section 7 gives validity threats, Section 8 analyses and summarizes the survey findings.

2. Related Research

The literature research has revealed many scientific papers related to Software Quality Assurance Practices. We will present several of them, relevant to our project. The study conducted by Davis et al., describes the results of a postal survey on the practice in software quality and is published in (Davis, Gillies, Smith, & Thompson, 1993). It is not focused particularly on quality models rather it is concentrated on the topics related to quality assurance, QA, practices. Results revealed that 41 % of the respondents do not use any formal QA method whereas; majority (55 %) uses their own method for QA. Jung et al. published a survey of ISO/IEC 9126. They conducted a survey with 75 respondents, with aim to evaluate their satisfaction concerning the quality of a packaged-software product according to the criteria of ISO/IEC 9126's sub-characteristics. Their results show that users' perceptions about product quality vary across user types. Universitat Politèchnica de Catalunya (UPC) has published a numerous papers on the usage of quality models during software package selection. Botella et al. (Botella, Illa, Carvallo, Franch, & Quer, 2002) presented a methodology for building a domain based quality model. Same group of researchers(Botella et al., 2003) have also published a work on quality models for some COTS products such as ERP. These papers show examples of domain-based quality models. A survey on software testing practices in Australia in 2004, published by Ng, S. P. et al.(Ng, Murnane, Reed, Grant, & Chen, 2004) conducted among 65 different organizations have shown that testing methodologies and techniques and automated testing tools are used by the majority of them. Their survey includes information about software testing metrics, standards, and training and education regarding testing. A similar survey was conducted by Wagner et al. (Wagner, Lochmann, Winter, Goeb, & Klaes, 2009) in Germany were 25 managers and users of software quality models were interviewed. Their results show that companies mainly use their own quality models.

3. Background

Product quality and software quality have been defined by many authors. To provide a clear picture about quality related terms, following we present a number of quality and software quality attributes definitions. The term quality is used in all kinds of situations, making difficult to give exact and accurate definition. Software quality is an abstract concept; its presence can be difficult to define, but its absence is seen instantly. There are many variations on the definition of software quality. Quality has different meanings for customers, users, management, marketing, developers, testers, quality engineers, maintainers, and support personnel. Different organizations define quality and its characteristics in different ways. Dictionary definitions of quality mainly are focused on excellence which presents the professional's focus. ISO 9126 ("ISO/IEC 9126," n.d.)defines quality as a set of attributes of a software product by which its quality is described and evaluated". The IEEE Standard Glossary ("IEEE standard glossary of SE terminology," n.d.) defines software quality as a planned and systematic set of activities to ensure that quality is built into the software.

- **Quality characteristics/factors-** Are used to describe the product and form the basis for the evaluation (Firesmith, 2004).
- **Quality sub-characteristics-** To make it easier and more precise quality characteristics may be refined into multiple levels of sub-characteristics (Firesmith, 2004), the objective of a which is to characterize a part of a quality characteristic, an aspect of a quality of a work product or process and help in defining the term of "quality" for an endeavour.
- **Quality Criterion**- A criteria can be seen a base for comparison or a reference point against which other things can be evaluated. A software quality criterion is an attribute of a quality characteristic that is related to software development process (Naik & Tripathy, 2011).
- **Quality Metrics-** According to("IEEE standard glossary of SE terminology," n.d.) quality metric (1) is a quantitative measure of the degree to which an item possesses a given quality attribute and (2) a function with software data as input and a numerical value as output interpreted as the degree to which the software has a certain quality attribute.

4. Quality Models

The set of characteristics and the relationships between them form quality model, which offer the basis for specifying and evaluating quality requirements(Franch & Carvallo, 2003). It is important that all relevant quality characteristic are specified and evaluated, using validated or accepted metrics. Software quality models may be used during the development of a new product or when selection of COTS is needed.

According to Dromey (Dromey, 1995) quality models should make clear high-level quality attributes. A good quality model must:

- Offer a guide which systematically introduce and build quality software
- Provide means which will systematically identify quality defects
- Introduce a structure which will be adaptable and understandable at all levels

From the research on software quality models it can be seen that different models have been published. Some describe the direct relation between metrics and factors like whereas others are more advanced and use a particular methodology for defining the relation between quality attributes, characteristics and metrics. The most well known models are : McCall's quality factors proposed in 1976, Barry Boehm's quality model presented in 1978, FURPS in 1987, 9126 in 1991, and Dromey model in 1996.

5. Study Design

Evaluating software quality means performing systematic investigation of the software capability to implement specified quality requirements. Dromey says that no quality process can exist if it is not based on a product quality model. We expect that a quality model is chosen or defined based on the requirements of the software that is being built. We have conducted a survey in small IT companies. Almost all companies in region have attributes of small companies since they all have less than 25 people involved in the process of software development. Our aim was to study whether software companies in region are familiar and use quality models. To the best of our knowledge, we believe that this survey, on software quality assurance practices and in particular the usage of quality models, is the first one carried out in Macedonian and Kosovo IT industry. We analyzed the usage of QM from the viewpoint of managers and practitioners in the context of software developing industry. Our research is based on an explorative survey in which managers and practitioners are asked for applied approaches. The interviewees come from organizations that develop software in different domains. The number of interviews was limited since not all project managers responded to our request. We have conducted a face-to-face interview. During the implementation phase we have prepared a questionnaire comprised of both closed and open type of questions. We have carried out our observations in small IT companies and relied on interviews of people working there. Their working domain is different, such as standardized software product development, custom software development and consulting and/or integrated hardware/software development. A threat may come from the asked questions, which in a way are tended to direct the answers in a certain direction. Interviewees were project managers and people responsible for software development, selection of which was not done randomly, thus the answers may be biased. Another threat may come from the sample which was not as big as we planned since not all project managers were available to be interviewed. Our goal was to interview people that have to do only with quality assurance processes, thus the results may be biased.

Within the scope of our research we address three research questions:

RQ1.What is the rate of the delayed projects and which are the reasons?

RQ2. What QMs are used?

RQ3. Does your company have a quality assurance team and how do you measure quality?

The results reported in this paper are based on the completed questionnaires and interviews. Despite the small sample that took part in our survey, but based on the data we obtained, we believe, heightened our confidence to report our observations in this article.

6. Presentation of the Results

The data obtained from the interviews and the questionnaire has given us a broad spectrum of insights. However, we will present only the most important ones which were related to our research questions. We asked practitioners and managers about the rate of delayed projects, RQ1. Most of the developers answered they don’t know and some of them said that the rate is very low. Interviews with managers helped us to get more precise results. Almost all projects are late, or are released with bugs which are then fixed after reported by the customers. Main reason for this is the process of requirements specification which is very time consuming. Managers complain that clients are not always ready to take active part during software development.

To address RQ2 it was asked which QMs are used and results show that all of the respondents claimed not to have any formal QA method at all, usually they use their own method for QA, and are not very much aware of the models that are in use. Interviews, with managers show that companies apply their own standards depending on the product they build “it depends on the project, the customer and the time available”. They use internally checklists, adjusted to a particular project.

Results on the question which attributes of the quality are more important to your company, are presented in the Fig.1.

Testing plays an important role in achieving and assessing the quality of a software product(Friedman & Voas., 1995), it improves the quality of the product by repeating a test–find defects–fix cycle during development helps assessing how good the system is when performing a system-level tests before releasing a product. Friedman and Voas (Friedman & Voas., 1995) have described software testing as a verification process for software quality assessment and improvement. While implementing a program unit, the programmer tests whether or not the unit works in normal circumstances. The same applies to an entire system—once a system has been integrated; the developers test whether or not the system performs the required functions.

To shed light on RQ3, we asked about the process of quality assurance, quality analysis techniques and metrics. For the first one, the options were (1) testing (unit testing or system testing), (2) review and inspections (during the development), and (3) customer feedback (during testing and after release). Results are presented in the Fig. 2.

Software metrics are the only mechanized tools for assessing the value of internal attributes and are defined as “standard of measurement, used to judge the attributes of something being measured, such as quality or complexity, in an objective manner”(Grady & Caswell., 1987), but subjective measurement of quality comes from human estimation.

To get answers regarding quality metrics, question Q7-Q9 From our survey we have concluded that human evaluation is the only way for measuring software quality. All companies declared not to use any quality metric. Based on the interviews we had with managers we found that companies do not have a quality assurance team, instead this is done by the developers of the project. From the questionnaire we can conclude that most of the developers are not educated or trained on software testing techniques.

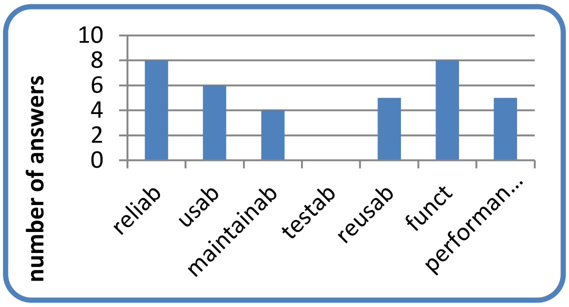

Figure 1. Quantitative results on quality characteristics

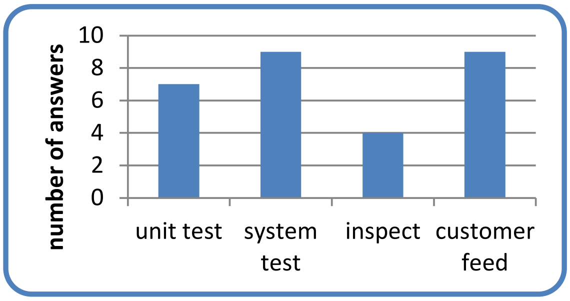

Figure 2. Quantitative results on techniques of quality assurance

7. Discussion of the Results

The results from the survey provided very valuable insights into the attitudes and working practices of the companies. Literature suggest different reasons which stop projects to be delivered in time, such as requirements specification, number of bugs, etc. Regarding the late projects (RQ1), from our survey we can conclude that almost all projects are late, and the main reason, as project managers and developers declared, behind it is the process of requirements specification which also directly influences to the quality of the end product. There are a large number of various QMs in use. However, company-specific models seem to be popular complemented with standards and laws. The process for quality assurance (RQ2) differs widely in different companies and projects. They are used as a source for requirements and as a basis for quality analysis. From the fig. 2 we can see that the mostly used quality assurance techniques are dynamic testing and customer feedback. Based on the information we got from the questionnaire, most of the developers are not educated or trained on software testing techniques. Companies, which were part of our survey are all small IT companies and cannot afford a special team for quality assurance. Regarding metrics for measuring quality (RQ3), the results of the qualitative analysis are confirmed also by the quantitative one, companies do not use any metric, either are aware of them. Instead the measurement is done based on the human evaluation.

8. Conclusions and Future Work

Quality models have been subject to research for many years. The objective of quality models is to describe, assess and/or predict quality. QMs consist of a number of quality characteristics (or factors, as called in some models). These characteristics could be used to reflect the quality of the software product from the view of that characteristic. A real challenge is selecting the most appropriate quality model. The aim of this study was to evaluate the current use of QMs in practice in the small software companies, with various business domains, in SEE region. Even though there are a numerous QMs which are used in industry, in practice in region we could identify a serious challenges which we aim to address in future. The research findings indicate that in many companies, schedules for software products are largely unpredictable, and state of quality-management practices is very weak. Developers attempt to build quality into the product by avoiding defects during the development process. The main problem is the process of operationalization of QMs, i.e. how to make them work in a real environment and produce quantified results. From the all the characteristics presented, one characteristic is common to all quality models, that is „reliability". Our survey shows that reliability and functionality, except others, are two characteristics that are considered from all companies, during the quality assurance process. Studying and analyzing quality models in use will help us to identify the principal elements for a new Software Quality Model. As a long term goal, we aim to contribute to the development of a new quality model relying on the results of our survey.

Acknowledgements

We would like to thank all IT companies and professionals for providing data, which enabled the paper, get framed.

.

References

Botella, P., Burgues, X., Carvallo, J. P., Franch, X., Pastor, J. A., & Quer, C. (2003). Towards a Quality Model for the Selection of ERP Systems. *Component-Based Software Quality. Springer Berlin Heidelberg*, 225–245.

Botella, P., Illa, X., Carvallo, J., Franch, X., & Quer, C. (2002). Using Quality Models for Assessing COTS Selection. *WER*, *3*.

Davis, C., Gillies, A., Smith, P., & Thompson, J. (1993). Current practice in software quality and the impact of certification schemes. *Software Quality Journal*, *2*(3), 145–161.

Dromey, R. (1995). A model for software product quality. *Software Engineering, IEEE Transactions on*.

Firesmith, D. (2004). A hard look at quality management software. *OPEN Process Framework*.

Franch, X., & Carvallo, J. (2003). Using quality models in software package selection. *Software, IEEE*.

Friedman, M. A., & Voas., J. M. (1995). *Software assessment: reliability, safety, testability*. John Wiley & Sons, Inc.

Grady, R. B., & Caswell., D. L. (1987). *Software metrics: establishing a company-wide program*. Prentice Hall.

IEEE standard glossary of SE terminology. (n.d.).

ISO/IEC 9126. (n.d.), 1–18.

Khosravi, K., & Guéhéneuc, Y. (2004). On issues with software quality models. *The Proceedings of the 11th Working Conference on Reverse Engineering.*

Naik, S., & Tripathy, P. (2011). *Software Testing and Quality assurance -Theory and Practice*. John Wiley & Sons, Inc., Hoboken, New Jersey.

Ng, S. P., Murnane, T., Reed, K., Grant, D., & Chen, T. Y. (2004). A preliminary survey on software testing practices in Australia. *2004 Australian Software Engineering Conference Proceedings.*, 116–125.

Wagner, S., Lochmann, K., Winter, S., Goeb, A., & Klaes, M. (2009). Quality models in practice: A preliminary analysis. *Empirical Software Engineering and Measurement, ESEM 2009. 3rd International Symposium On. IEEE*.