

An Efficient and Objective Generalized Comparison technique for Software Quality Models

Saba Awan

Department of Software Engineering, University of Engineering & Technology, Taxila, Pakistan
Email: sabaawan70@yahoo.com

Faizah Malik

Department of Software Engineering, University of Engineering & Technology, Taxila, Pakistan
Email: faizahmalik47@gmail.com

Ali Javed

Assistant Professor, Department of Software Engineering, University of Engineering & Technology, Taxila, Pakistan
Email: ali.javed@uettaxila.edu.pk

Abstract—To scrutinize the uniqueness of software quality model it is crucial to compare it with existing ones. Quality is generally apprehended in a model that illustrates the features and their interactions. Numerous models for measuring quality of software processes have been recommended to assess particular type of software products. Numerous methodologies and practices have been suggested to perform the specific or general scope based comparisons among eminent models. These comparisons are leak. The Suggested comparison lacks the clear differentiation and in depth analysis. Consequently, a prescribed method of comparison among software quality models has been defined. The suggested technique is applied on an inclusive comparison among renowned software quality models. The consequence of suggested technique demonstrates the power and faintness of quality models.

Index Terms—Characteristics, factors value, model value, software quality, software quality models, weights.

I. INTRODUCTION

Quality includes all appearances and significant characteristics of an artifact or an action which recount to the substantial of given requests. Software is perilous in establishing a economical advantage to many organizations [1], and is gradually fetching the important aspects of production processes, products and facilities. Software engineering [2] including

Software Quality as its essential aspect has an association to the enlargement and advancement of big, multifaceted and vital software-intensive system. Researchers [3] have made enormous efforts to bring about new dimensions in software engineering aspects including quality perspective. Software systems are anticipated to be more flexible, accessible and recyclable.

There are different models which are developed with different features and efficiencies. McCall model was proposed in 1976-7, known as first ever quality model. In this paper, we will discourse by studying the relationship that relate the diverse quality models and find the crucial characteristics between McCall's, FURPS, ISO 9126, Evans and Marciniak, IEEE's and Deutsch & Wills Quality Model. Section 2 present the Literature review of these software quality models, section 3 contain the Proposed Framework and comparison Methodology , Evaluation of proposed methodology has been performed in section 4 whereas section 5 presents the Quantitative analysis of current systems vs. proposed approach and section 6 concludes the paper and Provides the comments about future work.

A. McCall's Model

McCall J.A [4] proposed the first quality model. To plan a complete design of the products quality through its several features was fundamental intention behind this proposal. There are 3 types of quality classifications for a software product i.e. quality related to Product Operation (correctness, reliability, efficiency, integrity and usability) quality related to Product Revision (maintainability, flexibility and testability), and quality related to Product Transition (portability, reusability and interoperability). McCall tries to connect the break between clients and engineers by focusing on various programming quality angles that uncover both the client's perception and the designer's significances.

B. Furps Quality Model

Robert Grady and Hewlett Packard designed Furps model [7] in 1978. Quality attributes are classified by this model into in two types of requirements namely functional and non-functional requirements. Functional requirement are well-defined by response and predictable

results, while nonfunctional requirements are emergent and expected behaviors of system. Weakness of this model is that it miscarries to take into account the interpretation of the portability of software product. Two steps are well-thought-out when this model is used: characterizing quality traits and setting priority. Caswell and Grady considered that establishing needs is critical given the inherent trade-off, i.e. one quality can be picked up at the expense of alternate.

C. ISO 9216

One of the International standards for the progression of software is "ISO 9216". The standard is distributed into two parts which covers the Internal and External Quality Attributes. The properties of the system [8] that can be assessed without accomplishing are covered by the internal quality attributes. While the framework qualities that may be assessed by seeing the framework during its execution comes under the subject of External quality characteristics. ISO 9216 Part-1 is an extension of previous work done by McCall (1977), Boehm (1978), FURPS. The model [9] trails the factor-criteria-metric model and classifies software quality characteristics into six self-governing high-level quality characteristics. Each of these is separated into subordinate quality characteristics.

D. IEEE's Quality Model

Several standards have been released by IEEE [10]; one of related to quality assurance is IEEE STD 730-1998. Software Quality Assurance Plans is basic agenda of this IEEE's Standard. Basically It is activity based quality model IEEE 1219 std. IEEE Model is concerned with basically providing a standard for maintenance of software by delivering a qualitative model. This standard gives an incremental procedure to association and execution of programming support events. Supplementary standards are also defined i.e. software quality assurance, verification and validation, software configuration management in which associated processes (external processes). This model illustrates and represents numerous dimensions of qualitative factors.

E. Evans and Marciniak Quality Model

Two substitute models have developed after the McCall model; first one is the Evans and Marciniak quality model 1987. These two models [11] convey new inspiration on software quality. Together these models eliminate testability from the McCall model. The Evans and Marciniak model has twelve factors that are grouped into three categories: design, performance and adaptation. Two of the factors proposed by the innovative models are very analogous to the factors of McCall model. Verifiability, security and manageability are the three new factors. Verifiability defines structures design and programming that allows effective certification of design and programming. Safety's [12] main concerns are design to remove peril circumstances for operators of tools as an outcome of faults in method regulator software. These errors can degenerate into inappropriate reaction to

dangerous situations or failure to

F. Deutsch & Willis Quality Model

The Deutsch and Willis model is also an alternative model that has developed in 1988 after the McCall's model. This model too eliminate the testability factor from McCall's model together with Evans and Marciniak quality model which was developed before the Deutsch and Willis model. It has fifteen factors grouped into four classes: operating, performance, change and management.

II. LITERATURE REVIEW

Quality is defined by Krzysztof Sacha [13] as combination of two things i.e. conformance to specifications and meeting customer needs. As models depicts the state of mind/thinking of customers so in this paper [14] different 17 quality models have been discussed and explained to some extent. Their comparison shows that there are multiple attributes which are commonly being shared by each other and some attributes are shared by two or more.

Sony Tripathi [15] defines quality according to their own experiences. This paper presents the explanation of Quality Assurance as an activity related to the processes through which the products are produced. The description and comparison of five models their comparison is shown so that the quality factors which have been commonly shared by all models and also the differences are clearly represented.

Software quality assurance helps to achieve customer satisfaction and quality improvement. Ashwin et al. [16] define SQA as collection of different stages in software development from requirement elicitation up to identifying and fixing defects in final product. Quality is a dependent feature and some of those features are listed in this work.

Su-Hua et al. [17] Investigated Software Quality, Role and Value of Quality Models. Quality factors are the [18] attributes which developer add in the product from internal perspective of product and also they can be the attributes that client requires to be present in developed system. Through the comparison between different quality models important similarities and difference can be found which are then used to develop a model of desired quality attributes according to required system.

Kavita et al. [19] made a comparison of various Software Quality Models in their paper. They described that to calculate and enhance the nature of any product item, software model is essential. Each quality model consists of some particular characteristics that show the quality of developed system. Research methodology of used in this paper tells that, after comparison and analysis between different quality models a model of proposed quality attributes according to organization's need can be built.

A Compared analysis of Software Quality Models performed by Sheikh Fahad et al. [20] states that main objective of a model framework is to check that whether the relevant quality factors exist or not. The proposed

models measure the quality of a software process, and enhancements are applied to improve process quality. The quality attributes contained by the models are afterwards used to show the efficiency of the software product from different perspective which determines quality. This paper drives the analytical comparison between different quality models. The technique used for the comparison is "Questionnaire" and "Interviewing".

Objective measurement methods are used by Deepshikha Jamwal [21] to identify the attribute's existence and its deficiency. Software product's overall quality is reflected by the extent to which each of these attributes is present. The research methodology chosen for comparison between quality models in this research paper is "Questionnaires". The results of questionnaires will be used to suggest that which model should be selected for a particular software product.

Ranbireswar et al. [22] used the approach of collecting data from various organizations through primary and secondary research methods to perform the comparison between quality models. In this paper various software quality models are comparatively analyzed. For the comparison of different quality models, data is being collected from various organizations using both primary and secondary research methods. In the Primary method questionnaire have been used and feedback was received. Secondary methods included journals, research papers, Articles, Periodicals etc. Analysis is done depending upon the results of survey by using different analysis tools and presented in tabular form.

BOUKOUCHI et al. [23] provided a comparative analysis to help the software organizations for the selection of appropriate quality model. Model can be general or specific in nature. After Comparison, it concludes that Maintainability, Efficiency, Usability and Portability are common key quality factors in these models whereas Reliability is overall commonly shared quality factor. As quality must not only be essential for delivered product but also for software process therefore by following the principals of such models quality can be achieved in whole software process.

Vilas et al. [24] proposed effective work which is based on software quality and evaluation of quality models. They use a systematic approach towards model comparison. Main agenda of their research is to provide the platform where one can understand and learn how to research appropriate software quality model. Different six quality models are briefly explained defining their quality factors.

By considering the lacks and overlaps in quality models, Parastoo et al. [25] proposed a new Meta Model which is basically a main practice of Model Driven Engineering. Basic agenda of Meta modeling is to develop proper model for MDE and determining different characteristics which affect the activities for maintaining the quality of other quality models. Francisca Losavio [26] used a systematic approach for developing Quality Models to design and develop Software Architecture.

III. PROPOSED FRAMEWORK

In order to clearly differentiate software quality models, Weightages comparison strategy is proposed. This strategy considers the sub factors in addition to the elements to demonstrate acceptable and exact contrasts between quality models. This method includes an overall comparison of some renowned software quality models. The result of the comparison reveals the uniqueness among these models.

A. Architecture

To assign the values for these Factors a simple formula is used. To show whether the attribute is designated as a factor or not, then 50% weightage was given to factor and 25% to the sub-attribute. It was equally distributed between the numbers of the sub factors included in the comparison for the rest of the percentage. The weight of the software attribute is considered equally as the comparison is a common term comparison. Comparison for specific scope can be done by the proper weighting which reveals the model which highlights the attributes of the proposed scope and which is enclosed to the scope.

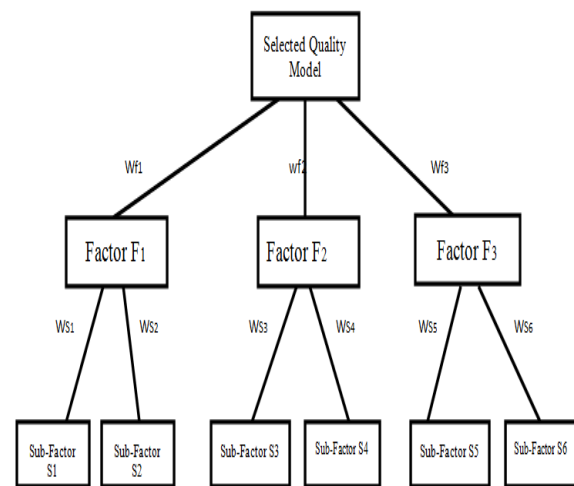


Fig.1. Selected Model Weighting.

B. Methodology

Weightages comparison method comprises of four main steps:

- *Collection of Models:* The methodology of "collection of models" is subject to the extension on the scope planned to be assessed, generally the decently Reputed software quality models are considered.
- *Collection of Factors:* In this task, the factors are clustered, gathered and repeating factors are removed. Different sub-factors from separate models are considered, the child factors are gathered under their parent factors and the repeated sub-factors are removed.

- *Factors Weightage:* Depending upon the scope that needed to be assessed, Factors (wf1, wf2.....wfn) n sub-factors (ws1, ws2.....wsn) are weighted accordingly.
- *Values of Factors:* Above assigned weight is used as a basis to calculate the value of a factor in the model .Firstly, using (Formula 1) value of factors which are common in particular models has been calculated. Secondly, using (Formula 2) the total value of selected model is counted depend on the counted values of their factors which is termed as the total value for quality model denoted by QV.
- *The resemblance:* the counted value of factors is compared with other respective models. It represents the inclusive differentiation among software quality models having those factors.

$$FV = Wf + \sum_1^n WS_n \quad (1)$$

$$QV = \sum_1^n FV_n / n^f \quad (2)$$

IV. EVALUATION

The quality characteristics from different models were joined and recurring was removed in accordance with the description of them. The sub-factors of specific factor from particular models were joined and similarly the recurring ones are removed. The values were assigned equally which gave 50% to attribute designated as the quality factor and 25% has been given to the sub-factor.

This correlation does not consider any form of software infrastructure or such working domain, as it is a general one so the value of the factors selected from the model are same. Among the sub factors that were collected for the factor from those models, the value 50 was equally divided. Finally the value (QV) for each model was calculated by using formula 2. Table 1 presents the total value for each model.

Table 1. Total Value (QV) of Software Quality Model

Factor of Selected Model	McCall's	Furps	Deutsch & Wills	IEEE's	ISO 9216	Evans and Marciniak
Correctness	100	0	75	25	0	100
Human Engineering	0	25	0	0	0	0
Efficiency	75	25	100	100	100	100
Functionality	0	100	0	100	100	0
Usability	100	100	100	100	100	100
Testability	75	25	0	25	25	0
Flexibility	75	0	75	0	0	75
Performance	0	100	100	0	0	0
Portability	50	0	50	100	100	0
Interoperability	100	100	75	25	0	100
Maintainability	50	25	75	0	100	75
Integrity	100	0	100	0	0	100
Reusability	50	0	50	25	0	50
Supportability	0	100	0	100	0	0
Reliability	100	100	100	100	100	100
QV(Total value for quality Model)	58.33 %	40%	60%	46.6%	41.6%	53.3%

FURPS model assess the product efficiency from customer perspective, developer perspective is ignored in assessing the quality factors such as reusability and maintainability .this model lacks software correctness reliability and portability. Also, the quality factors

assessment measured the operation of system accordingly. Additionally, this model fails to take account of the software reusability, portability and integrity. The total quality value calculated for this model is less than the weighted average value for selected quality models.

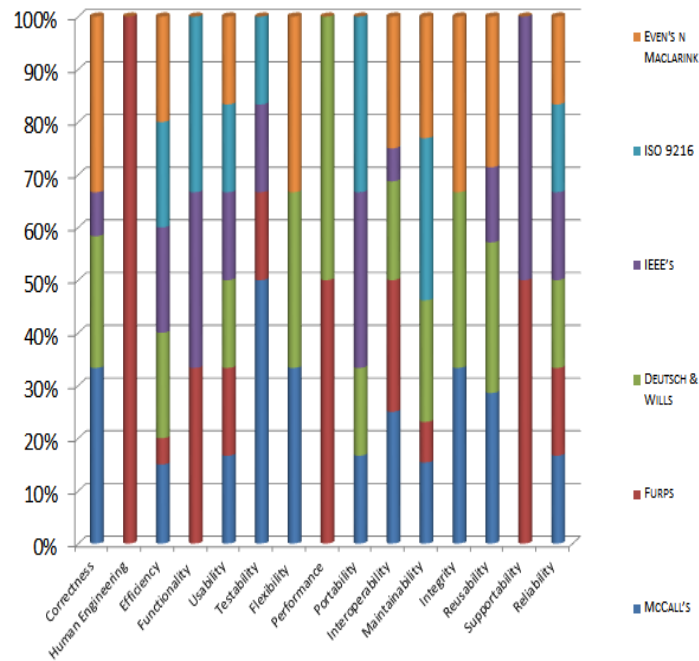


Fig.2. Stacked Cylindrical Presentations for Quality Models Comparison.

The Deutsch and Willis model is a substitute model that has developed after the McCall model. Its basic categorization of factors is as operating, performance, change and management. This is also a general model like ISO and there is absence of factors like human

engineering supportability, and software correctness. The total quality value calculated for this model is more than the weighted average value for selected quality models as McCall's model do has.

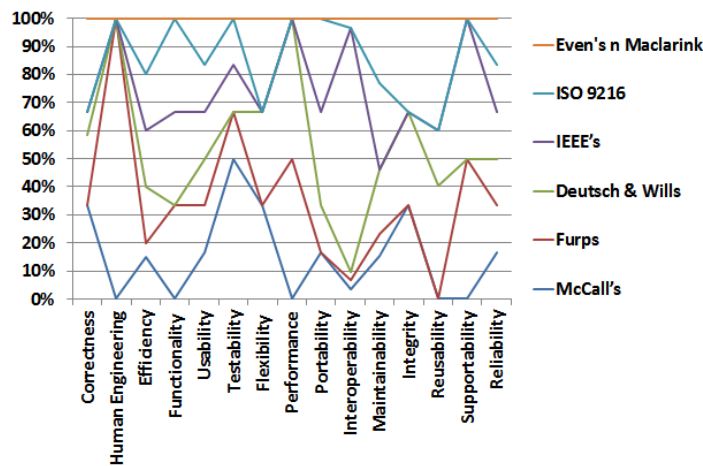


Fig.3. Stacked Line Presentations for Quality Models Comparison.

McCall model establish layered approach having number of layers 2 as Furps model do has. It reflects software product attributes, [27] excepting the functionality of the product and the human engineering factor, as it is central to identify the usability factor for the software artifact. Whereas understandability factor is enclosed indirectly through the sub-attributes .it has the absence of the connections between the factors, which basis the intersecting in its relations. The total quality value calculated for this model is more than the weighted average value for selected quality models.

ISO model tracks the factor-criteria-metric model and classifies software quality characteristics into self-governing advanced quality features. Each of these is fragmented down into subordinate quality traits, which anticipated to overawing of the confusion prevailing in most of the software quality models. As [27-28] it is a general model which has many factors which are missing such as supportability, human engineering and software correctness. The total quality value calculated for this model is less than the weighted average value for selected quality models as Furps model do has.

IEEE Model is fundamentally about the maintenance of software product as it is a standard to provide a

qualitative model. It takes iterative approach [30] for execution and management of software maintenance events. This models fails to cover the human engineering perspective as McCall’s model do. But it has maximum

concern with efficiency and functionality of software product. The total quality value calculated for this model is less than the weighted average value for selected quality models as Furps and ISO do have.

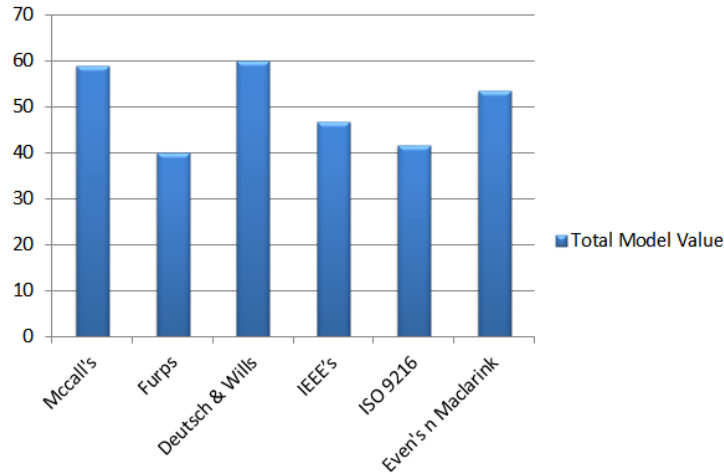


Fig.4. Total Quality Value Comparisons for Quality.

The Evans and Marciniak model describes twelve factors that are clustered into three categories: design, performance and adaptation. Usability and interoperability of software product is main concern of this model. Functionality perspective of software product lacks in it like in McCall’s model and also it does not consider human engineering perspective. The total quality value calculated for this model is more than the weighted average value for selected quality models as McCall’s and Deutsch and Willis model do has.

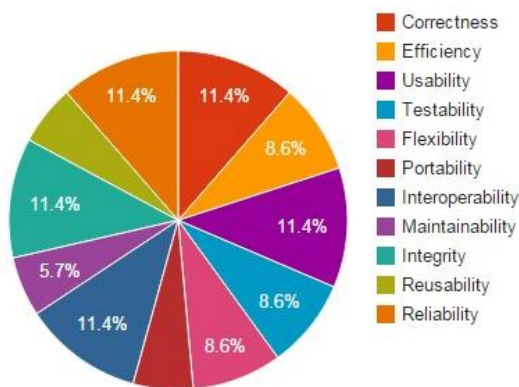


Fig.5. Pie Chart Representation of Factors Value per quality Model.

V. QUANTITATIVE ANALYSIS

The Current methodologies used for comparison between software quality models have used different criteria and approaches towards quality model’s comparison. Simple Attributes Chart Comparison System is one of currently used approaches. In this method, attributes present selected models are listed out and direct model search out is done to show the absence or presence of an attribute in quality models. Another approach towards the quality model’s comparison is Questionnaires. In this method the data is collected from various organizations, through which the questionnaires are developed. The questionnaire contains various software quality characteristics .Questionnaire are circulated to different organizations and feedback received. In Interviews system, the professionals from various software development organizations are interviewed personally to collect the data about the required quality attributes for them from each selected model then accordingly comparison is carried out from the collected data. In survey analysis system the study is carried out about different quality model and then online or manual surveys are conducted for comparison purpose. The suggested methodology in this paper is Weightages Comparison Method (Proposed System). In quantitative analysis, different parameters are chosen to check the performance of above mentioned comparison systems against these parameters/attributes.

Table 2. Comparison of Current Approaches

Current Approaches for comparison of Quality Models	Attribute Prevalence	Less Time Duration	Factor's significance analysis	Quality Models in Depth Analysis	Weighted Factors Count
Questionnaires	✓	✗	✗	✓	✗
Factors Frequency Comparison Method Comparison System	✓	✓	✗	✗	✗
Survey analysis System	✓	✗	✗	✗	✗
Interviews with organizations	✓	✗	✓	✓	✗
Simple Attributes Chart Comparison System	✓	✓	✗	✗	✗
Weightages Comparison Method(Proposed System)	✗	✓	✓	✓	✓

VI. CONCLUSION AND FUTURE WORK

The growing impact of software measurement has led to a growth in extent of study on emerging the new software procedures. This is an inclusive study to compute multiple features of different software qualitative models and evaluate their relative significance. In this paper, we have tried to present the clear differentiation between quality models through our proposed system to measure the significance of each characteristic within the quality model. We evaluated the 6 important quality models using our proposed system. This paper provides some assistance for researchers and practitioners for improved understanding and mixture of weighted factors for their purposes. As weightages comparison method has taken a general approach towards weightages assignment without considering any specialized domain. Future work can include using this approach to compare the quality model for specific domain where experts can be invited to assign weights to attributes based on the context of the selected domain.

ACKNOWLEDGMENT

We would like to express gratitude to Assistant Professor Ali Javed, Department of Software Engineering, University of Engineering and Technology Taxila, for his support and productive guidance to complete this research.

REFERENCES

- [1] John J Marciniak, "Encyclopedia of software engineering", 2vol, 2nd ed., John Wiley & Sons, Inc. New York, USA, 2002.
- [2] Barbara Kitchenham, and Shari Lawrence Pfleeger, "Software quality: the elusive target special issues section", Journal IEEE Software, IEEE Computer Society Press Los Alamitos, CA, USA, Volume 13, No. 1, pp. 12-21, 1996.
- [3] Walt Scacchi, "Process Models in Software Engineering", October 2001 Final Version to appear in, J.J. Marciniak (ed.), Encyclopedia of Software Engineering, 2nd Edition, John Wiley and Sons, Inc, New York, December 2001.
- [4] R. E. Al-Qutaish, "Quality Models In Software Engineering Literature: An Analytical And Comparative Study", Marsland Press, Michigan, USA, vol. 6, no. 3, (2010), pp. 166-175.
- [5] McCall, J. A., Richards, P. K., and Walters, G. F., "Factors in Software Quality", Nat'l Tech. Information Service, no. Vol. 1, 2 and 3, 1977.
- [6] Suman, Manoj Wadhwa, M.D.U. Rohtak "A Comparative Study of Software Quality Models "International Journal of Computer Science and Information Technologies, Vol. 5 (4), 2014.
- [7] Sony Tripathi "A Survey on Quality Perspective and Software Quality Models" IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p- ISSN: 2278 8727Volume 16, Issue 2, Ver. VI (Mar-Apr. 2014),
- [8] ISO/IEC 9126-1: Software Engineering - Product Quality-Part 1: Quality Model, International Organization for Standardization, Switzerland, 2001.
- [9] ISO/IEC 9126-3: Software Engineering – Product Quality- Part 3: Internal Metrics, International Organization for Standardization, Switzerland, 2003.
- [10] R. G. Dromey, "A model for software product quality". IEEE Transactions on Software Engineering, 21(2):146–162, 1995.
- [11] Kitchenham, B. and Pfleeger, S. L., "Software quality: the elusive target [special issues section]", IEEE Software, no. 1, pp. 12-21, 1996.
- [12] Namita Malhotra, Shefali Pruthi, 2012. "An Efficient Software Quality Models for Safety and Resilience", International Journal of Recent Technology and Engineering (IJRTE), ISSN: 2277-3878, Volume-1, Issue-3, and August 2012.
- [13] Krzysztof Sacha, "Evaluation of Software Quality" Warsaw University of Technology, Nowowiejska 15/19 00-665 Warszawa, Poland k.sacha@ia.pw.edu.pl.
- [14] Suman, Manoj Wadhwa "A Comparative Study of Software Quality Models" International Journal of Computer Science and Information Technologies, Vol. 5 (4), 2014, 5634-5638.
- [15] Sony Tripathi, "A Survey on Quality Perspective and Software Quality Models IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 2, Ver. VI (Mar-Apr. 2014), PP 63-72 .
- [16] Ashwin Tomar and VM Thakare, "Identification and Listing Of Factors Affecting Software Quality Assurance" Dept Computer Science, Siddhant Institute of Computer Application, Pune, MH, India .
- [17] Su-Hua Wang, Durgesh Samadhiya, 3Dengjie Chen "Software Quality: Role and Value of Quality Models" International Journal of Advancements in Computing Technology Volume 3, Number 6, and July 2011
- [18] Alexandre Alvaro, Eduardo Santana de Almeida, Silvio Romero de Lemos Meira "Quality Attributes for a Component Quality Model" Federal University of

Pernambuco and C.E.S.A.R – Recife Center for Advanced Studies and Systems, Brazil.

- [19] Kavita Sharma, Kumud Sharma. "Comparison of Various Software Quality Models" Proc. of the Intl. Conf. on Recent Trends in Computing and Communication Engineering RTCCE 2013.
- [20] Sheikh Fahad Ahmad, Mohd. Rizwan Beg, Mohd. Haleem "A Comparative Study of Software Quality Models" International Journal of Science, Engineering and Technology Research (IJSETR) Volume 2, Issue 1, January 2013.
- [21] Dr. Deepshikha Jamwal," Analysis of Software Quality Models for Organizations" International Journal of Latest Trends in Computing (E-ISSN: 2045-5364) 19 Volume 1, Issue 2, December 2010.
- [22] Ranbireswar S. Jamwal, Deepshikha Jamwal & Devanand Padha "Comparative Analysis of Different Software Quality Models" Proceedings of the 3rd National Conference; INDIACOM-2009.
- [23] Boukouchi Youness, Marzak Abdelaziz, Benlahmer Habib and Moutachouik Hicham "Comparative Study of Software Quality Models" IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 6, No 1, November 2013.
- [24] Dr.Vilas. M. Thakare and Ashwin B. Tomar "A SYSTEMATIC STUDY OF SOFTWARE QUALITY MODELS" International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.4, October 2011.
- [25] Parastoo Mohagheghi, Vegard Dehlen SINTEF "A Metamodel for Specifying Quality Models in Model-Driven Engineering ", P.O.Box 124 Blindern N-0314 Oslo, Norway.
- [26] Francisca Losavio "Quality Models to Design Software Architecture" JOURNAL OF OBJECT TECHNOLOGY Published by ETH Zurich, Chair of Software Engineering ©JOT, 2002.
- [27] R. E. Al-Qutaish, "Quality Models In Software Engineering Literature: An Analytical And Comparative Study", of American Science, Marsland Press, Michigan, USA, vol. 6, no. 3, (2010), pp. 166-175.
- [28] B. Behkamal, M. Kahani and M. K. Akbari, "Customizing ISO 9126 Quality Model for Evaluation of B2B Applications" Journal Information and Software Technology, vol. 51, Issue 3, (2009).
- [29] P. Botella, X. Burgués, J.P. Carvallo, X. Franch, G. Grau, J. Marco, C. Quer "ISO/IEC 9126 in practice: what do we need to know?"
- [30] Victor R. Basli and Albert J.Turner, "Iterative Enhancement: A practical Technique for software development" IEEE Transaction on software Engineering, Vol. SE- 1, No.4.



Engr. Faizah Malik accomplished her BS Software Engineering degree in July 2013 from University of Engineering and Technology Taxila, Pakistan. Her areas of concern are Software Quality Assurance, Software Requirement Engineering, Mobile Application Development, Web Development, Data mining, and Operating System. Currently she is MS Scholar as well as Research Associate in the Department of Software Engineering at University of Engineering and Technology, Taxila, Pakistan.



Engr. Ali Javed is serving as an Assistant Professor in the Department of Software Engineering at UET Taxila, Pakistan. He is also a PhD Scholar in Computer Engineering Department at UET Taxila, Pakistan. Currently he is serving as a visiting PhD research scholar at University of Michigan, USA. He has received his MS degree in Computer Engineering from UET Taxila, Pakistan in February, 2010. He received Chancellor's Gold Medal in MS Computer Engineering degree. He has received B.Sc. degree with honors in Software Engineering from UET Taxila, Pakistan, in September, 2007. He got 3rd position in Software Batch-2003F in BS Software Engineering degree. His areas of interest are Digital Image Processing, Computer vision, Video Summarization, Machine Learning, Mobile Application Development, Software Requirements Engineering, Software Quality Assurance and Software testing.

Authors' Profiles



Engr. Saba Awan is a MS Intellectual in the Department of Software Engineering at University of Engineering and Technology Taxila, Pakistan. She has graduated from University of Engineering and Technology Taxila in Software Engineering in July 2013. Her areas of interest are Agile Application

Development, Software Quality Management, Database Application Development, Object Oriented programming, Software Architecture and Design.