State of Quality Engineering Practices: The Pakistan Perspective

Muhammad Naeem Ahmed Khan
Independent Researcher
Islamabad, Pakistan
mnak2010@gmail.com

Aamir Mehmood Mirza
Baluchistan University of Information
Technology, Engineering and
Management Sciences
Quetta, Pakistan
mirza.aamir@buitms.edu.pk

Mahwish Shahid
University of Management and
Technology (UMT)
Lahore, Pakistan
mahwish.shahid@umt.edu.pk

Raja Asif Wagan
Baluchistan University of Information Technology,
Engineering and Management Sciences
Quetta, Pakistan
raja.asif@buitms.edu.pk

Imran Saleem
University of Management and
Technology (UMT)
Lahore, Pakistan
imran.saleem@umt.edu.pk

Abstract−This study analyzes the state of quality engineering practices being exercised in the software industry of Pakistan. Statistics have been collected and analyzed to access important aspects of quality engineering including quality policy, review mechanism, quality assurance activities and practices, quality standards and models, and quality management systems. For this purpose, an elaborated questionnaire was prepared to pertain to various aspects of quality management and more than 30 software houses and software development organizations were surveyed in Islamabad and Rawalpindi. The survey results and a description of the concluding remarks are reported in this paper.

Keywords-quality assurance; software testing; software quality models; software quality engineering; quality management system

I. INTRODUCTION

Software is used in almost every sphere of human activity and its usage is growing constantly. Software quality is an important issue, not only for software designers and developers, but also for customers, users, and communities. Software quality has a major role to play in assessing a software product's success or failure [1]. The quality of software systems is a complex and multi-faceted concept which is difficult to define and measure [2]. It cannot be expressed by a single metric [3], because it includes various aspects such as flexibility, reliability, usability, performance, maintenance, and portability. It pertains to the intrinsic features and characteristics along with customer satisfaction which make hard to precisely define quality. Intrinsic quality refers to the quality of the software itself and customer satisfaction refers to the user perception of quality [4]. Value may be implicitly provided by other methods, or specifically given in the forms of international standards, general guidelines, or detailed checklists. A common approach to deal with this problem is designing and implementing quality models in an organized manner. Implicitly quality can be provided by certain methods

or explicitly can be provided in the forms of universal standards, guidelines, or comprehensive checklists. As a result, quality models have been subjected to extensive study for, a variety of heterogeneous quality models have emerged, e.g. different domain models or test models. The advancement made with these models and standards tends to fit for certain approaches that may be optimum for some projects, but is often ill-suited for others [5]. Since the software industry is experiencing rapid changes due to the advancements in technology, ever-growing user requirements, modernization of business processes, revolutions in network infrastructure and hardware, sophistication in development and execution environment, etc., most of the software quality improvement activities concentrate on the technological aspects of the software development process.

During the last four decades, several quality methods and standards have been developed to steer software developing companies by defining their processes in an effective manner. These quality models are continually evolving, which has eventually become a complicated matter for software industries. In addition, many companies develop indigenous standards and subsequently improve the quality of their systems, when their own software development processes mature. The current state of the affairs of software systems and software development is categorized by the growing sophistication of the end product and the growing demand to produce cost effective software on time [6]. During the last few years, the software industry in Pakistan has seen steady growth. According to the statistics from the Pakistan Software Export Board (PSEB) [7] 144 out of 1500 companies in Pakistan achieved international process or product quality certifications. Out of these 144 businesses, 110 have ISO 9001 certification, 23 have CMMI certification and 11 have ISO 27001 certification. These figures are quite alarming and indicate a

Corresponding author: Mahwish Shahid

strong need to improve the quality of services, products, and IT-infrastructure in the software industry of Pakistan.

In this study, the different methodologies and practices of software quality models currently followed by IT organizations in Pakistan were investigated. The study also focuses on the analysis and evaluation of software quality models being practiced for manufacturing high-quality software products. Furthermore, problems and challenges regarding the implementation of these quality models and standards for possible enhancements are discussed. The main aspects of the current study are:

- A discussion of the importance of quality attributes and techniques with respect to their industrial usage.

- The employment of specific quality models by the practitioners.

- The potential for further improvement of the quality models.

The center of focus of the research is to find answers of the following questions to achieve the above-mentioned objectives:

- RQ1: What is the role of quality objectives in a software organization's quality policy?

- RQ2: What quality assurance procedures and practices do the software organizations adopt to ensure the software product quality?

- RQ3: Do software organizations have a tendency towards quality attributes and their prioritization?

- RQ4: Do process improvement models play a role to achieve the customer's level of confidence in quality?

- RQ5: What is the significance of review process of quality management systems?

A. Pakistan Software Industry: An Overview

Pakistan's tech industry has risen steadily from its initial start in the late 1980s, and is now ready for a take-off based on its initial success. Pakistan provides various competitive advantages over other outsourcing destinations, including the ease of doing business and state-of-the-art IT infrastructure. The Government of Pakistan has been developing the IT sector over the last few years, having established IT parks with low rents, foreign ownership of equity invested in IT and having offered an exemption on income tax from exports of computer software and related services. Table I highlights the key facts of software industry in Pakistan.

TABLE I.   FACTS AND FIGURES OF PAKISTAN SOFTWARE INDUSTRY

Key facts	Figures
Skilled workforce	110000
Software companies	1500
Software Technology Parks (STPs)	9
STP area	700,000 Sq. ft.
Standards and certifications	
ISO 9001	110
CMMI	23
ISO 27001	11

US\$2.8 billion is the estimated share of Pakistan's IT industry in the global market whereas it is offering numerous business opportunities both in public and private sectors [8].

II.   RELATED WORK

Quality in general, is critical for the success of a product. Quality of a product is significantly dependent on the quality of the development process. Authors in [9] published the findings of a survey on software quality practices. Most of the survey questions were related to methodical topics rather than the use of models for product quality. The results of the study showed that 41% of the respondents reported that have no structured quality assurance plan, while the rest used their own technique/process rather than an external one [9]. T. Hall examined the activities related to quality, particularly the attitudes of developers and managers towards quality. The study results showed that 99% of developers used programming standards, 74% employed code reviews or inspections, and 62% collected software metrics data [10]. Authors in [11] conducted a survey on ISO 9126 aiming to investigate whether the categorization of quality attributes was correct and accurate. This resulted in an improved categorization to provide guidance for revising the standard. This new categorization includes separating security from functionality, which is the latest recommendation of the new standard ISO 25010. A survey in Australia was conducted in [12] in order to identify which testing methodologies, techniques, and tools were used in the software industry of the country. The major barriers found for testing were time and cost. The results showed that only 55.3% of the surveyed companies believed that metrics helped them boost the quality of software. In addition, the study found that the use of manual static analysis, i.e. formal and informal evaluations, was more widespread than the automated and tool-based analysis. Authors in [13] interviewed product managers from Swedish companies to determine their quality criteria, establishing that making the criteria well defined and testable was an arduous task [13]. A survey in Canada [14] showed that inadequate testing and training and the lack of standardization for testing increase the likelihood of defects in the production and release versions of software products. The ability to identify patterns that lead to reduced quality and to determine the root causes of decline in product quality suffer from the lack of rigorous testing of the product. Organizations that train their developers on quality assurance and testing procedures, increase the likelihood that process and product defects are detected and resolved accordingly. Cost and lack of expertise are two major obstacles in the adaptation of testing methodology and tools.

A study conducted in Brazil with the goal of defining the understanding of respondents about the relationship between IT governance models and the adoption of quality instruments. The obtained findings provided a description of the impact of IT-related issues in companies and the degree of expertise, significance, and adoption of different quality instruments [15]. A survey of the quality models practiced in Germany focused on areas of software development, such as standard software being used, custom development methodologies, embedded systems and company size [16]. An exploratory survey of software practices by tech companies in five ASEAN countries

found that software industry fell short in the following four areas: use of automated estimation tools, use of requirements traceability matrices, use of quantitative quality metrics, and the use of change control boards. The best practices of quality orientation, autonomous research teams and Software Engineering Process Group (SEPG), peer reviews, and quality management systems were discussed [17].

Agility supports quality objectives and increases the efficiency of the software development process, thus improving overall product quality. The agile approach should be used when dealing with unpredictable requirements. Agile is a quick process with fast iterations and earlier releases. Different tasks carried out in each increment, reduce the production cost and improve product quality. One of the key benefits of the agile approach is that the customer is involved in the development process and gives feedback. In the agile development, client at site, pair programming, and frequent integration activities enhance the production process in particular, thus enhancing the overall product quality. Extreme Programming (XP) is a development methodology within the agile paradigm and it works on many areas to improve quality [18]. XP not only reduces the time of production but also increases the overall quality of software because it is focused on standards that need to be implemented throughout the software development lifecycle [19]. There is a prevalent misguided assumption that quality means goodness or shininess. The word "quality" is often used to denote the relative value of something which means different things to different people. Therefore, quality must be defined as "compliance to requirements". Consequently, non-compliance found in the software pertains to the absence of quality, and in such situations quality problems become non-compliance problems, and then quality becomes undefined [20]. Software quality is the entirety of features and characteristics of a product or service that underpin its ability to satisfy given needs [21]. Quality has six attributes: functionality, reliability, usability, efficiency, portability, and maintainability. Software quality appears to be particularly problematic when compared to other areas. The main reason for this is that software has no physical existence. Clients do not know of what they need, they periodically need changes, the advancements in hardware and software are rapid, and customers have high expectation levels [22].

III. RESEARCH METHODOLOGY

This study aims to explore the state of affairs of quality models, standards, and optimum practices used in the software industry of Pakistan. The literature review formed the basis for formulating the questionnaire that was addressed to software companies. A mixed approach method was used, i.e. open/close interviews were conducted for qualitative data and the survey was conducted to get an understanding of the problem on a larger scale.

A. Sample Data

Quantitative data were collected through a survey with a total of 100 respondents. The sample size consisted of 50 software houses of various sizes situated in the cities of Islamabad and Rawalpindi.

B. Framework of the Study

A comprehensive analysis was performed to identify the problem areas and challenges faced while implementing the best quality engineering practices. Research questions were addressed using the quantitative research method. The solutions were discussed along with the challenges in the Discussion section. Figure 1 gives an overview of the research process.
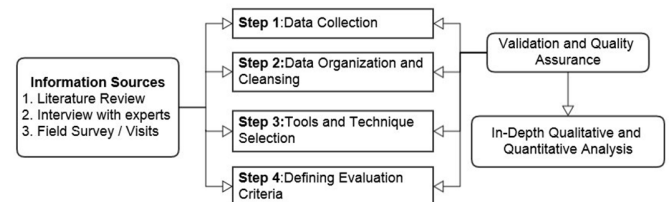

Fig. 1.   Research methodology.

IV. RESULTS

In this section, detailed analysis of the results derived from the surveys is presented. Every Survey Question (SQ) presented in Table II is discussed, and the results are shown in Figure 2 and Table III.

TABLE II.   SURVEY QUESTIONS

SQ #	Survey Question
SQ 1	Do you have any defined quality objectives in your quality policy?
SQ 2	Do you have any framework to review/update quality objectives in your quality policy?
SQ 3	Do you have any defined organizational structure for validation and verification to assure quality?
SQ 4	Do you analyze the customer requirements about the quality of the product?
SQ 4 (a)	Do you formally record the requirements mentioned in SQ 4?
SQ 5	Do you have any documentation to formally record customer's quality requirements?
SQ 6	Do you have any formal procedures to record non-functional requirements?
SQ 7	Do you have any quality control plans for design and development phases?
SQ 8	Do you have a mechanism to ensure that the product meets the requirements (functional and non-functional) of the customer?
SQ 9	Do you follow any standards/models for development processes' improvement?
SQ 10	If yes in SQ 9, then are you satisfied with the performance of the quality models?
SQ 11	Which process improvement model is adopted by your company?
SQ 12	Do you have any review or audit mechanism for processes in your company?
SQ 13	Is there any quality management system in your company?
SQ 14	Do you have a review process for a quality management system?

A. Organizations Quality Policy

SQ 1: Do you have any defined quality objectives in your quality policy?

Software quality is about conforming to functional and non-functional requirements. Achieving structural quality which refers to incorporating non-functional requirements is far more difficult than meeting functional quality. To develop high quality software, companies need to define quality objectives

[23]. For this purpose, companies are required to develop software quality metrics, define defect target levels, and identify quality attributes and factors that affect software quality. It is quite alarming that only 67% of the organizations surveyed in this study define quality objectives.

SQ 2: Do you have any framework to review/update quality objectives in your quality policy?

The field of computing is much dynamic as new tools and techniques are developed every day. On the other hand, the new tools and techniques bring about new challenges due to the constantly emerging computing paradigms. Quality policy and objectives cannot remain stagnant. They must be reviewed and updated frequently. In this study, it has been observed with concern that only 53% of the respondent companies have a framework for reviewing, 30% do not have any framework and the rest (17%) did not answer this question.
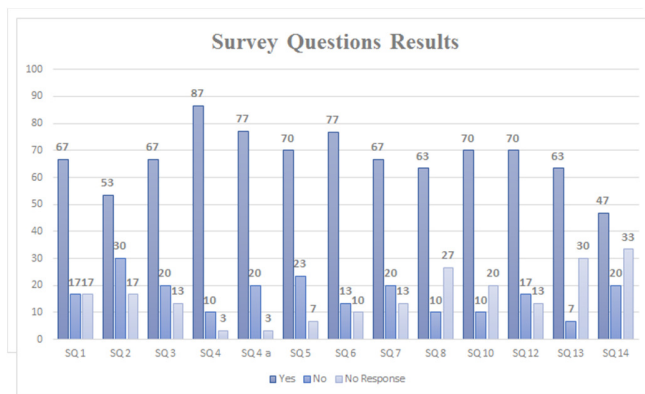

Fig. 2.   Survey results (%).

TABLE III.   SURVEY RESULTS

SQ #	SQ 1	SQ 2	SQ 3	SQ 4	SQ 5	SQ 6
Yes	67	53	67	87	70	77
No	17	30	20	10	23	13
No response	17	17	13	3	7	10
SQ #	SQ 7	SQ 8	SQ 10	SQ 12	SQ 13	SQ 14
Yes	67	63	70	70	63	47
No	20	10	10	17	7	20
No response	13	27	20	13	30	33

B. Software Quality Assurance Activities and Practices

SQ3: Do you have any defined organizational structure for validation and verification to assure quality?

It is of utmost importance to have a well-defined organizational structure and dedicated teams for validation and verification to ensure the high quality of software products. As depicted in Figure 2 and Table III, only 67% of participating organizations have a defined quality organizational structure. Those who do not have their own quality assurance teams are 20% while 13% did not answered this question.

SQ 4: Do you analyze the customer requirements about the quality of the product?

Software quality could simply be defined as meeting the customer requirements. Quality is always required, but it is never stated explicitly and remains a non-functional requirement. Thus, it is crucial to understand the customer's perspective about quality when analyzing quality requirements. This study revealed that 87% of the participants perform quality related analysis.

SQ 4 (a): Do you formally record the requirements mentioned in SQ 4?

It is important to record quality requirements from customer perspective. Only 77% of the participating companies formally record these requirements, while 20% do not record and 3% did not respond.

SQ 5: Do you have any documentation to formally record customer's quality requirements?

Non-functional requirements are not explicitly stated by users, but are described informally and are often conflicting. This makes them hard to model, measure and evaluate. To make them measurable, it is important to discuss and if required, negotiate with the customers. It is observed that 70% of the participating companies have developed procedures to deal with non-functional requirements and 23% organizations do not have any formal procedures.

SQ 6: Do you have any formal procedures to record non-functional requirements?

Studies have shown that defects identified at an early stage can be fixed at a lower cost. Therefore, quality control activities should be initiated at the very early phases of software development lifecycle and product quality should be maintained throughout the SDLC. This requires meticulous planning and mechanism to ensure the execution of the developed plan. Every phase of SDLC should have a detailed quality control plan, which includes all the required methods, processes, and procedures. Among the participating companies 77% have quality control plans for design and development phases, 13% of them do not, and 10% did not answer this question.

SQ 7: Do you have any quality control plans for design and development phases?

Validation and verification activities finish with the delivery of products to customers. Validation and verification teams act as client's representative and ensure that the end product meets all functional and non-functional requirements. Validation and Verification (V&V) teams develop methods to control and prevent the delivery of products that do not conform to the requirements. Only 67% of the participating companies ensure that no such product is released, 20% do not have any mechanism for controlling the delivery of non-conforming products while the 13% of participating organizations did not answer this question.

SQ 8: Do you have a mechanism to ensure that the product meets the requirements (functional and non-functional) of the customer?

Software quality metrics are used to assess the development process, project, and product. Companies use these metrics to assess the performance and quality, and if necessary, take corrective action to fix those areas that cause defects or violate the targeted defect level/threshold. A 63% of the participating companies ensure software quality metrics, whereas 10% answered that they have no such mechanism.

C. Quality Standards and Models

Quality cannot be improved if we cannot measure it. For this purpose, attributes should be identified and quantified. Prioritizing these attributes is also important in order to ensure that neither the most desired attributes are compromised nor the least priority attributes get more than desired consideration. Table IV and Figure 3 depict quality attributes and their importance for different responding organizations. The study results indicate that functionality, usability, and reliability are high priority attributes while portability is the least important attribute.

TABLE IV.   QUALITY ATTRIBUTES AND THEIR PRIORITY

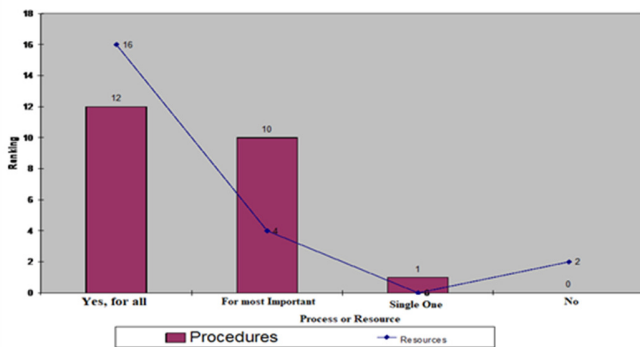| Attribute	Priority						
	1	2	3	4	5	6	7
Functionality	16	04	0	01	00	02	00
Reliability	09	05	02	04	01	01	01
Usability	03	09	02	03	02	02	01
Efficiency	02	07	07	03	01	00	02
Maintainability	03	03	02	05	04	01	03
Portability	04	03	02	01	03	03	04
Did not reply	02						

Fig. 3.   Quality standards.

SQ 9: Do you follow any standards/models for development processes' improvement?

Quality can only be accessed if minimum threshold and baseline are defined and proper quality standards have been developed or followed. Quality standards could be procedure-specific or resource-specific. It is observed that most of the organizations have defined quality standards either for all resources or all the procedures (Figure 3). Developing high quality software is a challenging task, hence the main focus of the entire software industry remains on developing sophisticated processes to produce better quality products. The development processes need continuous assessment and improvement due to the nature of the product which keeps changing. In this survey, 66% companies have implemented models, standards, and optimum practices for quality products.

SQ 10: If yes in SQ 9, then are you satisfied with the performance of the quality models?

Among those who have developed quality standards and models, only 70% are satisfied with the performance of the quality models, 10% think it is under-performing and 20% did not answered this question.

SQ 11: Which process improvement model is adopted by your company?

The participants were given a list with possible answers. The results are shown in Figure 4. We can see that the majority used either an ISO model or a company specific solution.
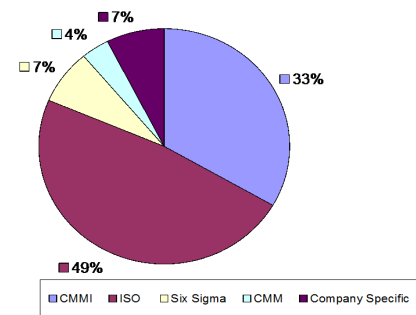

Fig. 4.   Process improvement model statistics (%).

SQ 12: Do you have any review or audit mechanism for processes in your company?

Reviews and audit activities ensure that the defined processes are actively being practiced for all the undertaken projects. Any anomaly is identified as minor or major non-conformance and project managers take appropriate corrective actions. It is observed that 70% of the participating companies have developed review and audit mechanisms, 17% organizations do not and 13% did not answer the question.

D. Quality Management System

SQ 13: Is there any quality management system in your company?

In complex and large software development, defect identification is a difficult task. Therefore, it is very important to identify defects as early as possible. Failure to do so results in significant increase in risk of time or cost overrun. To prevent this, software companies deploy quality management systems which can prevent defects from occurring. It is observed that 63% of the participating companies have implemented a quality management system.

SQ 14: Do you have a review process for a quality management system?

Reviewing the effectiveness of the quality management system at planned intervals enables companies to take preventive and corrective actions, but only 47% of the

participating companies do so, while 20% do not whereas 33% did not answer this question.

V.   DISCUSSION

The nature of software quality is dynamic, with changing face of technology, it should be revisited and upgraded according to new challenges. To produce high quality software, it is important to develop a quality policy in accordance with the organizational priorities and measures of success. Additionally, a mechanism to timely review and update this quality policy should also be established. Quality policy and objectives cannot remain stagnant, they should be reviewed and updated frequently. This study identified that only 67% of the surveyed companies met the defined quality objectives and only 53% have a framework for reviewing and updating quality objectives and policy. Quality policy cannot be implemented without a well-defined organizational quality structure. It includes dedicated and independent V&V teams which ensure that the developed product meets the goals and objectives defined in the quality policy. The study data shows that only 66% of the participating organizations have a defined quality organizational structure. Their V&V teams develop methods to control and prevent the delivery of the products that do not conform to the requirements. Apparently, only 60% of the organizations ensure that no product that does not adhere to the required quality standards is released.

There are several quality models like ISO 9126 or ISO/IEC 25010:2011 which can be adopted for quality attainment. As the most of the revenue of Pakistan's software industry comes from outsourced projects, an internationally recognized quality model can improve the confidence level of the customers of IT companies. Nevertheless, companies may also develop their own quality models to achieve a certain level of confidence about the quality of the developed products. Defect identification and resolution is a costly task. Quality management systems can prevent defects from occurring, which allows overcoming the risk of schedule and cost overrun [24, 25]. It has been observed that 63% of the participant companies have adopted and implemented a quality management system. Reviewing the effectiveness of a quality management system at planned intervals enables companies to take timely preventive and corrective measures. Data should be collected to assess and improve the effectiveness of quality management systems, but the survey results show that only 47% of the surveyed companies have a review process.

VI.   CONCLUSION

The main objective of this study was to analyze the state of quality engineering practices in Pakistan with focus on four aspects: organization's quality policy, software quality assurance activities and practices, quality standards and models, and quality management system. This research contributes towards the understanding of the software quality assurance best practices in Pakistan and analyzes product quality models and differentiates them with respect to their usage as well as possible improvement potentials from the viewpoint of quality managers and quality model users. This study can help software development companies to identify whether these quality models and standards assert considerable impact to achieve, enhance, and sustain the quality of software systems. The findings of this study can assist software houses to increase the level of quality of their products. They can also enable companies to realize the importance of standards and models in their working environment and how these contribute to enhance the quality of software products.

REFERENCES

[1] C. Yue, "A projection-based approach to software quality evaluation from the users' perspectives," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 9, pp. 2341–2353, Sep. 2019, doi: 10.1007/s13042-018-0873-y.

[2] S. Yadav and B. Kishan, "Analysis and Assessment of Existing Software Quality Models to Predict the Reliability of Component-Based Software," *International Journal of Emerging Trends in Engineering Research*, vol. 8, no. 6, pp. 2824–2840, Jun. 2020, doi: 10.30534/ijeter96862020.

[3] N. Fenton and J. Bieman, *Software Metrics: A Rigorous and Practical Approach*, 3rd ed. Boca Raton: CRC Press, 2014.

[4] S. H. *Metrics and Models in Software Quality Engineering*, 2nd ed. Boston: Addison-Wesley, 2002.

[5] A. M. Davis, "Achieving Quality In Software Requirements," *Software Quality Professional*, vol. 1, no. 3, pp. 37–44, Jun. 1999.

[6] B. Kitchenham and S. L. Pfleeger, "Software quality: the elusive target [special issues section]," *IEEE Software*, vol. 13, no. 1, pp. 12–21, Jan. 1996, doi: 10.1109/52.476281.

[7] "Pakistan Software Export Board," *Pakistan Software Export Board*. https://www.pseb.org.pk/ (accessed Sep. 19, 2020).

[8] "Pakistan Software Houses Association for IT & ITeS," *P@SHA*. https://www.pasha.org.pk/ (accessed Sep. 19, 2020).

[9] C. J. Davis, A. Gillies, P. Smith, and J. B. Thompson, "Current practice in software quality and the impact of certification schemes | Request PDF," *Software Quality Control*, vol. 2, no. 3, pp. 145–161, Sep. 1993, doi: 10.1007/BF00402266.

[10] T. Hall, "What do developers really think about software quality?," in *Software Quality Management III Vol 1 Quality Management*, vol. 13, C. A. Brebbia and G. Staples, Eds. WIT Press, 1995, pp. 359–368.

[11] H. W. Jung, S. G. Kim, and C. S. Chung, "Measuring software product quality: a survey of ISO/IEC 9126," *IEEE Software*, vol. 21, no. 5, pp. 88–92, Sep. 2004, doi: 10.1109/MS.2004.1331309.

[12] S. P. Ng, T. Murnane, K. Reed, D. Grant, and T. Y. Chen, "A preliminary survey on software testing practices in Australia," in *2004 Australian Software Engineering Conference. Proceedings.*, Apr. 2004, pp. 116–125, doi: 10.1109/ASWEC.2004.1290464.

[13] R. Berntsson Svensson, T. Gorschek, and B. Regnell, "Quality Requirements in Practice: An Interview Study in Requirements Engineering for Embedded Systems," in *Requirements Engineering: Foundation for Software Quality*, 2009, pp. 218–232, doi: 10.1007/978-3-642-02050-6_19.

[14] V. Garousi and T. Varma, "A replicated survey of software testing practices in the Canadian province of Alberta: What has changed from 2004 to 2009?," *Journal of Systems and Software*, vol. 83, no. 11, pp. 2251–2262, Nov. 2010, doi: 10.1016/j.jss.2010.07.012.

[15] R. S. de Espindola, E. M. Luciano, and J. L. N. Audy, "An Overview of the Adoption of IT Governance Models and Software Process Quality Instruments at Brazil - Preliminary Results of a Survey," in *2009 42nd Hawaii International Conference on System Sciences*, Big Island, HI, USA, Jan. 2009, pp. 1–9, doi: 10.1109/HICSS.2009.70.

[16] F. Deissenboeck, E. Juergens, K. Lochmann, and S. Wagner, "Software quality models: Purposes, usage scenarios and requirements," in *2009 ICSE Workshop on Software Quality*, Vancouver, BC, USA, May 2009, pp. 9–14, doi: 10.1109/WOSQ.2009.5071551.

[17] A. Imam, S. A. Khoja, and I. Shariff, *Improving Software Quality: a Benchmarking Approach*. CSREA Press, 2007.

[18] M. Huo, J. Verner, L. Zhu, and M. A. Babar, "Software quality and agile methods," in *Proceedings of the 28th Annual International Computer*

Software and Applications Conference, 2004. COMPSAC 2004., Hong Kong, China, Sep. 2004, pp. 520–525 vol.1, doi: 10.1109/CMPSAC.2004.1342889.

[19] Y. Dubinsky and O. Hazzan, "Improvement of software quality: Introducing extreme programming into a project-based course," presented at the 14th International Conference of the Israel Society for Quality, 2002.

[20] P. B. Crosby, *Quality Is Free: The Art of Making Quality Certain: How to Manage Quality - So That It Becomes A Source of Profit for Your Business*, 1st ed. New York, NY: McGraw-Hill, 1979.

[21] D. Ince, *Iso 9001 and Software Quality Assurance*. New York, NY: McGraw-Hill, 1994.

[22] A. Gillies, *Software Quality: Theory and Management*. London, UK: Chapman & Hall, 1992.

[23] P. Jain, A. Sharma, and L. Ahuja, "A Customized Quality Model for Software Quality Assurance in Agile Environment," *International Journal of Information Technology and Web Engineering (IJITWE)*, vol. 14, no. 3, pp. 64–77, Jul. 2019, doi: 10.4018/IJITWE.2019070104.

[24] M. N. A. Khan, A. M. Mirza, and I. Saleem, "Software Risk Analysis with the use of Classification Techniques: A Review," *Engineering, Technology & Applied Science Research*, vol. 10, no. 3, pp. 5678–5682, Jun. 2020.

[25] A. Chenarani and E. A. Druzhinin, "A Quantitative Measure For Evaluating Project Uncertainty Under Variation And Risk Effects," *Engineering, Technology & Applied Science Research*, vol. 7, no. 5, pp. 2083–2088, Oct. 2017.

AUTHORS PROFILE

Muhammad Naeem Ahmed Khan is an independent researcher from Islamabad, Pakistan

Aamir Mehmood Mirza is an assistant professor at the Department of Computer Science, Baluchistan University of Information Technology, Engineering and Management Sciences, Pakistan. He received his M.S. degree in Embedded Systems in 2008 from Halmstad University, Sweden. His areas of interest are software quality engineering practices, embedded systems and measures, and metrics.

Mahwish Shahid is a research associate at the Office of Research Innovation and Commercialization (ORIC), University of Management and Technology, Lahore, Pakistan. She received her M.S. degree from UET, Lahore in 2016. Her areas of interest include blockchain technology, gamification, software engineering, machine learning, and data mining.

Raja Asif Wagan is an assistant professor of Information Technology, Balochistan University of Information Technology, Engineering and Management Sciences Quetta, Pakistan. His areas of interest are wireless sensors, networks communication, algorithms, and routing protocols.

Imran Saleem is an Assistant Professor at the School of Professional Advancement, University of Management and Technology, Lahore, Pakistan. He received his M.S. degree in Software Engineering Management from Southampton Solent University, UK in 2010. His areas of interest are blockchain technology, software engineering measurement and metrics, and quality engineering.