International Journal of Engineering & Technology, 7 (4.1) (2018) 42-46

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paperer

Developing a Conceptual Framework using Exploratory Testing to Improve Software Quality in SME's

Murugan Thangiah*, Shuib Basri, Dhanapal Durai Dominic

Department of Computer and Information Science
Universiti Teknologi PETRONAS, Bandar Sri Iskandar 32610, Seri Iskandar, Perak, Malaysia
**Corresponding author E-mail:tm_gun@hotmail.com*

Abstract

In order to create quality software with standards and agreeing principles, the Small and Medium Software Enterprises (SME's) faces many challenges and issues due to variety of reasons. Starting from the requirement analysis phase, the challenges emerge and continue until the project nears its completion, before being released to the customers or stakeholders. Various issues surfaced during the SDLC phases are identified and analyzed in the study. A Conceptual Framework using Exploratory Testing has been developed based on the study and the quantitative analyses were conducted using survey questionnaire. In this research paper the data analysis of the quantitative survey is conducted using Partial Least Square Structural Equation Modeling. The reliability and validity of the data is evaluated and presented in this paper which is essential to develop the conceptual framework. Further analysis of the survey questionnaire will be carried on and it will be reported in future work.

Keywords: *Agile Development Process, Exploratory Testing, Requirement Analysis, SPI, SMEs*

1. Introduction

Small and Medium Enterprises (SMEs) are becoming a cornerstone and provide a significant contribution in the worldwide industry economy for the past two decades. As far as the software development industry is concerned, SMEs has emerged, grown and strengthened, so that, they represent a major economic activity throughout many countries across the world.
Due to the unprecedented development in the internet technology, designing flexible process, responding to frequent changes in the requirements, realizing a fast time-to-time market but lack of manpower and resources poses great challenges to the organization in the software industry. The development team faces increasing pressure in productivity besides maintaining and/or improving the software quality standards and in thess circumstances, organization are motivated to look out for alternate ways to develop their software [1]. Software Engineers and IT managers may have different opinions regarding many software development principles, however, most of them agree on one common thing above all - Quality [2]. Since software applications are now-a-days become an increasing part of life, most people have had an experience with the software applications that it did not work as expected which resulted in loss of money, loss of time or business reputation.

In this research paper, a conceptual framework is developed based on the literature review and quantitative study was conducted by using questionnaire. This questionnaire was distributed to various small and medium size enterprises in Malaysia and the data analysis was done using SPSS. The analysis is done to verify the reliability and validity in the data. Further analysis will be carried out and reported in future works.

2. Background Study

In order to design and develop any software applications, it requires huge investments, a great deal of time, knowledge and skilled professionals with different domain expertise, and other resources such as tools and infrastructure. Despite the fact that during the past decade, software industry has matured a lot, considerable percentage of software failures had also been increased, which prompted the loss of capital, time, good-will, loss of data and sometimes even resulted in loss of lives due to severe failure in critical applications. In project management, one of the greatest challenges that still remain unsolved is to ascertain what is necessary to do in order to achieve success or failure. According to Montequin V. R., et., [3] in their research study stated as the main reasons for project failure were due to i) inaccurate, incomplete and ambiguous requirement by the customers, ii) frequent changes in the requirement, iii) inaccurate time and cost estimations, and iv) poorly defined specifications documents at the start of the project works. A large number of studies, cases, reports and criticisms for identifying software development projects were found to be neglected for various reasons and either the project has been abandoned or it has not fulfilled the required levels and improvements for the SME's [4].

To ascertain the factors contributing to the success or failure of software system, several research studies had been conducted and some of the common reasons were listed due to the volatile and competitive environment, difficulties in forecasting future scenarios, lack of information about the project scope, allocation of insufficient resources, under-performing project team members, inadequate risk analyses, the absence of corporate culture and involvement from the top management, planning,

execution and so on. These findings strengthen the fact that Information Technology field are continuously facing several troubles and problems.

The focus of this research paper is to find the various factors that have an impact on the organization structure, development process, testing techniques and methods adopted in the SME's as well the knowledge skill and experience of the tester and how it is affect the testing process which resulted in delivering the product to the stakeholder with poor quality. Most of the SME's are practicing agile development process which they believe as the best suitable model with aim to meet the stakeholder expectations due to the involvement of customers and end users towards system development. Arun Kumar, S., et al., stated that traditional approaches may improve the software quality and productivity, however, agile approaches aim to satisfy the real needs of the stakeholders and the project success [5]

The aim of the research study is to identify the shortcomings in the SME's and to study how it severely affect the business value of the projects they are undertaking. This will help the author to propose a new methods and solutions to overcome this shortcomings with the aim to produce a high quality software.

3. Related Works

In this section the author address the various issues surfaced in the organization in general which is also associated with operation of the organizations. There are so many definitions available for what constitutes an SME. It depends in which country and economic region these organizations are located [6].

3.1. Challenges in Organizational Structure

SME organizations are generally adopting a non-bureaucratic method of conducting business due to their flat organizational structure. Because of this operating methods it encounters many challenges and are highlighted in this section

In SME's starting from the requirement phase until the project is finished, it encounters so many problems due to many reasons such as non-bureaucratic organization structure; lack of expertise in development and testing, lack of resources – software, hardware, testing tools; lack of time due to the changes in the requirement by the stakeholders whereby rework has to be done which is time consuming process. Besides that SME's are operating under restricted budget and strict deadline is another reason which will affect the quality of the product. High employee turnover is another problem often exists in the SME's. Due to the small number of people in the organization, and very few people working on the project, the management processes are carried through in an informal ways and less documentation [7]. O'Sheedy [6] finds that project roles are often assigned to a small number of people and the project manager is in charge of both the management as well as the physical execution of the project.

Mishra, D., & Mishra, A., stated in their research that SME's are afraid of initial expenses [8]. Furthermore, lack of investment in testing environment, hiring an experienced QA expertise, and very limited resources are the another reason for poor quality. Basri. S., et al., [7] emphasized that in order to sustain in the competitive environment, it is important for the SME's to provide knowledge creation and sharing between individuals and groups within the organization. This will help the developer or the tester to transform their knowledge into new techniques which brings a future value to an organization. Dyba et al., in their research found that the levels of experience of the employees, majority of them are beginners – less than a year of experience – who is working in Agile development methods [9].

3.2. Challenges in Requirement Analysis

The success or failure of the project is mainly due to poor requirement. In software engineering, requirement analysis is considered as one of the most important phases in software development lifecycle and this is widely acknowledged in the software industry [10]. In many researches, it has been reported that due to both insufficient and poor requirements and the changes in the requirements frequently due to impractical expectations are some of the reasons for software project failures. [11]. In agile development methods, contrast to the traditional approach, the user requirements keeps on changing due to changes in technology, customer needs and business domains. The customers have become increasingly unable to define their needs up front, at the same time, expecting more features from the software. The requirements emerge during the development process and therefore regarding the functional requirements the documentation becomes meaningless and any documents produced in the early stage quickly becomes irrelevant.

Ramesh, B et al., (2010) emphasized that rather than creating a detailed specification of requirements at the beginning stage, the development team should acquire a high-level understanding of the critical features of the application [12]. Further in the research, several reasons were stated for starting the development activities at the very beginning because of the high volatile nature of the requirements, technical details of the project is unknown and the customer cannot specify exactly what they want through the system and can only define the system when it is completed. Therefore, changes in the requirement very frequently can cause wrong, incomplete and ambiguous requirements. In addition to that, the overall cost of the project will increase and the quality of the system will decrease or fail altogether.

3.3. Challenges in Software Development Process

There are many software process improvement models exist such as CMMI, SPICE, and ISO 9000 norms from the International Standardization Organization etc., These models are aim to prove the quality patterns and recommended to improve the software development process upon implementing [8]. However, these process models are not viable for SME's as it involves lot of costs.

Mirna, M., et al., [13] had stated that the main barrier stoppings SME's to implement software process improvement is due to lack of support and inadequate knowledge in software process improvements. Over the year, though many SPI initiatives have been proposed for the SME's in the software industry, still they are shortcomings due to lack of generalizability, and to achieve the benefits or advantages after implementing the SPI models, it takes lot of time. According to Basri, S at el., [14] most of the time, the software development and maintenance is spent on new product as well as adding new features in the existing system and process of fixing the old bugs is completely ignored and later it became a big problem. Besides that, their limitations of allocating budget and resources also give an impact to SME's process infrastructure like limited budget to response the risk, inadequate training and many other constraints [15].

3.4. Challenges in Testing in Agile Process

Do not follow the standards - There is a widespread tendency, in industrial and education environments, that the success of SPI is only possible for the large companies which have enough resources to handle and not viable for SME's, because of the organizational structure and the cost involved to implement [16]. In Agile process, Cripin & Gregory [17] stated that testing in agile is different from traditional testing because of the continuous and integrated nature of testing from the very beginning in the project life cycle. Agile projects do not have complete requirements nor architecture due to the characteristics of Agile methodology, how-

ever, both evolve over time [1]. Hence the tester must cope with changing, incomplete requirements, architecture and products. Every functionality in the product aims to deliver a "potentially shippable" software product and hence the developed functionality should be tested, verified and validated in order to assure that the bugs are removed [1].

Though many software process improvement models are out there, SME's are either not aware of them or they thought that these SPI programs implemented by software engineering practices are not focused on the needs and problems facing small organizations [18].

3.5. Challenges faced by Tester

Beer and Ramler [19] in their studies determined that the skilled software program tester can have better level of domain knowledge which allows assisting in the testing activities efficiently. According to their findings, experience plays a crucial role for a software tester which helps to interpret the specification and to make use of customized tools. However, McDaniel et al. [20] had conducted a survey to know the connection between experience and job performance across a wide range of profession and noted that there is correlation between experience and job performance. Further in their study, it is observed that when the length of experience is increased the correlation became weaker. According to Armour [21], suitable software testers have a nose for checking out, possessing "a kind of intuition that tells them what to test and how". Such thoughts discover full expression in the concept of exploratory testing [22], which is an unstructured testing described as "simultaneous learning, test design and test execution".

3.6. Challenges in Testing Methods and Techniques

Poor Q/A testing – Much effort is devoted to the improvement in areas of analysis and requirements, design and code reviews. However, in the spirit of continuous improvement in system quality, there is not much effort to improve its testing techniques to reduce customer found defects. [23]. Another possibility is that unit testing may be performed by developers themselves rather than a distinct testing team.

According to Tanjila Kanij et al, automated Testing methods are widely used us as testing tool in many organizations. By using the tool, it helps to sped up testing processes, improved accuracy, and freed software testers to conduct additional tests rather than conducting repetitive and prosaic tasks. However, it is noticed that every tools will perform only a specific testing methods. *Selenium and* QTP automate the process of test execution and evaluation, and assist with the test and defect management process and do not generate automated test cases. *Loadnrunner* and *Jmeter* are good for performance testing and not suitable for functional testing [24]. One of the reason is testing is done differently in different contexts. For example, an e-commerce software is tested differently from a safety-critical software. In this context, SME's are not capable to have all the testing tools

Hellmann et al., [25], identified a wide variety of tools used for different purposes in their research and found that many of the tools focus on enabling unit and acceptance test in different circumstances. Most of the tools are focus on providing basic testing techniques and relatively few tools provide support for advanced testing techniques in the agile development. *Jumple* is one of testing tool which provides support for mutation testing and it only works with Java.

Evelyn Moritz [23] indicated that the test team has focused their attention primarily on the new features and ignore the various configurations in the existing usage. This resulted in higher cost in correcting if the customer found defect than the cost of fixing a problem before release. If the requirements are not aligned with business needs the resulting development and testing effort is not optimal. This will result in waste of time once the misalignment is noticed and rework occurs [26].

Other factors may be the availability of the QA team. Some organizations have an independent tester within the development team and some do not have such team where the developer has to conduct the testing activities [2]. The advantages of having independent testers are they always unbiased and look for other and different defects, can verify the assumptions made during specification and implementation of the project. However, there is also some disadvantages if there is no proper communication between developers and testers, by having an independent tester, which may result in developer, may lose a sense of responsibility for the quality of the system.

3.7. Conceptual Framework using Exploratory Testing

Exploratory Testing (ET) is an essential part in an agile development process as it is able to help software testers to maintain up with the speedy improvement tempo of agile software initiatives. A set of heuristics may be carried out whilst testing that may guide the tester in a way to perform the testing and to assess the effects which includes boundaries, CRUD (Create, Read, Update, Delete), configuration variations, interruptions (e.g., log off, close down, or reboot) and many others. In agile program, choices manifest a lot quicker and are more subject to unintended consequences of several choices [27]. In different phrases, agile development can be given new and unanticipated functionality so fast and hence, it is not possible to expect the outcomes of each choice beforehand of time. This is where exploratory testing saves the day because it is always ready to be explored as the program always runs. Practioners emphasize that when automated testing is not a viable option, Exploratory Testing can be a great way to uncover some critical defects [28]. The proper application of exploratory testing significantly enhances the effectiveness of the entire testing effort. In the practitioner literature and scientific studies, Exploratory Testing has been widely promoted and also found that it support the effectiveness and efficiency of the ET approach with the studies comparing ET and other testing approaches. However, there are no empirical studies or only limited studies are available regarding the details of the actual exploratory testing practices and activities [29].

Based on the literature, a conceptual framework on Exploratory Testing is developed which has six independent variables, one dependent variable and one moderation variable. Independent variables are key factors which should affect the product quality in the SME's. The dependent variable is the quality factors which is the essential factor for SME's to produce a quality project for the stakeholders and the moderated variable will be the key factors to achieve the perceived quality. The following diagram Figure 1 shows the tentative model of the conceptual framework on Exploratory testing.

4. Research Methods

4.1 Population and Sample Size

Small and Medium size Enterprises (Software development Companies only) are considered as the target population for this study. This population only includes SME's which are undertaking software development activities. Other SME's which provides support services, consultant, selling a particular product, hardware, networking and installation are ignored for this study. The study was conducted at a random sample of 310 companies representing, Selangor, Kuala Lumpur and Penang in Malaysia. First a qualitative study was conducted based on interview and observation with the SME's. This is to ensure that those companies are actually involved in software development activities and also to ensure that

the number of employees and the annual turnover are within the range of SME's definition specified in the SME Corporation, Malaysia.

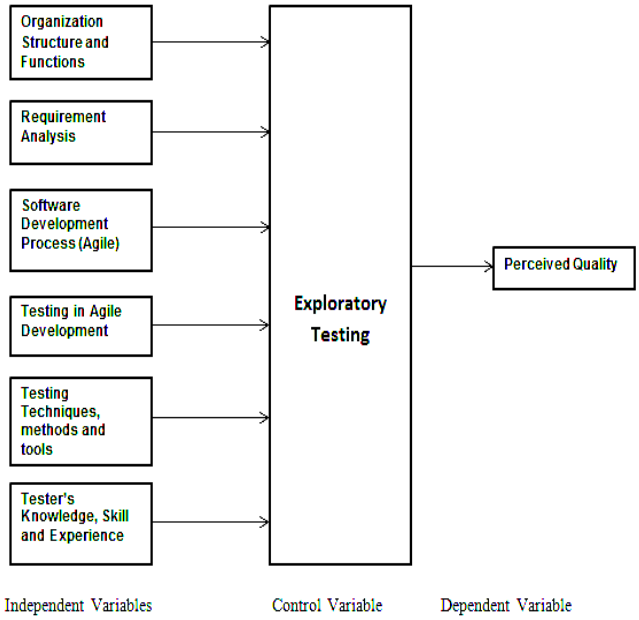

Fig. 1: Exploratory Testing Conceptual Framework

A survey questionnaire was created based on the conceptual framework and distributed to the various small and medium size software industries. Out of 310 companies given only 200 samples are selected and the remaining survey was rejected for various reasons. The data collected from the questionnaire was entered in the SPSS statistical tools and reliability test was conducted. The following are the result of reliability test of the independent variables. Based on the definitions of the factors identified in the conceptual model (see Figure 1), one dependent, six independent, and one moderating variables were operationalized and used to collect the data in this study.

4.2. Variables and Measures

Independent variables. To measure the impact of each of the six independent variables involved in the research model, a multi-item, five-point bipolar Likert scales were used which ranged from "Strongly Disagree" (1) to "Strongly agree" (5) for all indicators. For each independent variable, the item ratings were summarized to form a summated rating scale. All the items were written specifically for this research study since this is the first study of its kind within Exploratory Testing

Dependent variable. Each person responsible or involved in testing activities was asked to answer the questionnaire to measure the quality of the product using the 5-point, bipolar, Likert scales. This was operationalized and measured based on multi-item measures and all the items were written specifically for this research study.

Moderating variable. To analyze the most influential variation in software organizations to achieve the software quality, the control variable, Exploratory Testing, was operationalized and included in the study. Table 1 depicts the summary of the questionnaire survey data for the independent and the control variable along with the corresponding reliability measures (Cronbach's α).

Table 1: Results of Reliability Analysis using SmartPLS 3.0

Reliability Analysis					
Independent Variables	Number of Constructs	Items Deleted	Cronbach's Alpha	Composite Reliability (CR)	Average Variance Extracted (AVE)
Requirement Analysis	3	none	0.813	0.889	0.729
Software Development Process	4	none	0.794	0.865	0.620
Testing Technique and Methods	5	1	0.752	0.847	0.587
Organizational Structure	7	2	0.829	0.860	0.559
Testing in Agile Development	6	none	0.856	0.891	0.580
Tester's Knowledge, Skill, Experience	5	none	0.814	0.869	0.574
Exploratory Testing (Control Variable)	8	2	0.835	0.883	0.573

4.3. Assessment of Reliability and Validity

The consistency and the stability of a score from a measurement scale is referred as Reliability. Internal consistency analysis is used to evaluate the reliability of the multiple item measurement scales and researchers were traditionally using coefficient alpha [30] and detailed item analysis based on Nunnally's method [31]. The Cronbach's alpha assumes that all indicators are equally reliable and the individual indicator's reliability are the utmost priority. This raises the issues in which Cronbach's Alpha might not be proper as a measure of reliability and also it is sensitive to the number of items in the scale. Hence the different measure of internal consistency reliability called as Composite Reliability (CR) is assessed. Composite reliability values between 0.6 and 0.7 are acceptable any values between 0.7 and 0.9 is satisfactory. Values above 0.95 are not considerable because it shows that all the indicator variables are showing the same phenomenon.

To obtain satisfactory values for the coefficient alpha while retaining the domain coverage, the analysis revealed that some measurement items required to be eliminated to achieve the minimum threshold values. Table 1 reports the key factors associated with the original sets of measurement items and the reliability coefficients for the resulting scales. Furthermore, Table 1 shows that the reliability coefficients ranged from 0.847 to 0.891. All the measurement items and constructs developed for this study were judged to be reliable as it satisfies the minimum threshold value 0.7 or higher. Similarly, the threshold value for composite reliability (CR) must be higher than 0.7 and the threshold value for AVE is 0.5 and above. An instrument that is composed entirely of new scale is particularly good, since the scales do not contain large number of measurement items. This assessment validation is an important criteria in the data analysis process and from the above results, the research instruments shows the reliability in the data and thus model validation could be conducted by using structural assessment which will be next step to be followed.

5. Conclusion

The reliability and validity of the collected sample data is needed to be assessed for the data analysis process. The data analysis was conducted using SmartPLS 3.0. For the research instrument, the data must be verified and validated before being assessed for further analysis of this research study. Based on the quantitative survey, the analysis is partially done and further analyses are to be carried out with the available data. The quality of a product is depends on the six key factors identified and described in the conceptual framework. Furthermore, detail analysis is to be carried

out to validate the conceptual model using Partial Least Square Structural Equation Modeling.

References

[1] Van der Broek, R., Bonsangue, M.M., Chaudron, M., Van Merode, H., "Integrating testing into Agile software development processes",2nd International Conference on Model-Driven Engineering and Software Development (MODELSWARD),pp.561 - 574, 2014.

[2] Iacob, I.M., & Constantinescu, R., "Testing: First Step towards Software Quality", Journal of Applied Quantitative methods, Vol. 3, No. 3, 2008.

[3] Montequin, V.R., Cousillas, S., Ortega, F., & Villanueva, J., Analysis of the success factors and failure causes in Information and Communication Technology (ICT) projects in Spain. International Conference on Project Management, pp. 992-999. 2014.

[4] Meier, S. R., Technology Portfolio Management for Project Managers. 2017. Available at [https://www. pmiwdc.org/sites/default files/presentations/201703/PMIW_LocalCommunity_Tysons_presentation_2017-02.pdf].

[5] Arunkumar, S., & Arun Kumar, T., "Study the Impact of Requirements Management Characteristics in Global Software Development Projects: An Ontology based Approach", International Journal of Software Engineering and Applications, vol. 2, No. 4, pp. 107-125, 2011.

[6] O'Sheedy, D., "Agile Project Management for IT Projects in SMEs: A Framework and Success Factors", The International Technology Management Review, Vol. 3, No. 3, pp.187-195, 2013.

[7] Basri .S, & O Connor, R., "Evaluation on Knowledge Management Process in Very Small Software Companies: A survey", 2012..

[8] Mishra D. and Mishra A. "Software Process Improvement in SMEs: A Comparative View". Computer Science and Information Systems, Vol. 6, No. 1, pp.111-140. 2009.

[9] Dybå, T., & Dingsøyr, T. Empirical studies of agile software development: A systematic review. *Information and software technology*, *50*(9-10), 833-859. 2008.

[10] Asghar, S., & Umar, M., "Requirement engineering challenges in development of software applications and selection of customeroff-the-shelf (COTS) components". International Journal of Software Engineering, 1(1), 32-50. 2010.

[11] Hussain, A., Mkpojiogu, & E., Kamal, F., The Role of Requirements in the Success or Failure of Software Projects. EJ Econjournals. 6. 6-7, 2016.

[12] Ramesh, B & Cao, L & Baskerville, R., Agile requirements engineering practices and challenges: An empirical study. Inf. Syst. J.. 20. 449-480. 2010.

[13] Mirna, M., Jezreel, M., Calvo-Manzano, A.J., Gonzalo, C., & Tomas,S.F., "Expected requirements in support tools for software process improvement in SME's", 9th Electronics, Robotics, and Automative Mechanics Conference, 2012.

[14] Basri, S. & O'Connor, R., "Understanding the Perception of Very Small Software Companies towards the Adoption of Process Standards , in Riel et al (Eds), Systems, Software and Services Process Improvement, CCIS Vol. 99, Springer-Verlag, pp. 153-164, 2010.

[15] Kalpana, A.M., & Ebenezer E.J., "Software Process Improvization Framework nased on Fuzzy Logic approach for optimizing Indian Small Scale Software Organizations", International Journal of Multimedia and Ubiquitous Engineering, Vol. 6, No. 1, 2011.

[16] Pino, F.J., Garcia,F., & Piattini,M., "Software Process Improvement in small and medium software enterprises: a systematic review", Software Quality Journal, pp.237-261, 2008.

[17] Cripin, L., & Gregory, J., "Agile Testing – A practical guide for testers and agile team", Addison and Wesley, 2009.

[18] Van Zyl, J., "Software Testing in a Small company: A Case Study", Technical Report, Univesitiy of Pretoria, 2010.

[19] Beer, A & Ramler, R., "The role of experience in software testing practice," in SEAA '08: Proceedings of the 2008 34th Euro micro Conference Software Engineering and Advanced Applications Washington, DC, USA: IEEE Computer Society, pp. 258–265, 2008.

[20] McDaniel, M.A., Schmidt, F.L., & Hunter, J.E., "Experience correlates of job performance," *Journal of Applied Psychology*, vol. 73, no. 2, pp. 327–330, 1988.

[21] Armour, P.G., "The unconscious art of software testing," Communications of the ACM, vol. 48, no. 1, pp. 15–18, 2005.

[22] Bach, J., *Exploratory testing Explained*, E. v. Veenendaal, Ed. The Testing Practitioner. UTN Publishers, 2002, www.satisfie.com.

[23] Moritz, E., " Case Study: how Analysis of Customer Found Defects can be Used by System Test to Improve Quality", ICSE, pp.123-129, 2009.

[24] Kanij, T., Merkel,R., &Grundy, J., "A Preliminary Survey of Factors Affecting Software Testers", 23rd Australian Software Engineering Conference, 2014.

[25] Hellmann, T. D., Sharma, A., Ferreira, J., & Maurer, F., Agile Testing: Past, Present, and Future--Charting a Systematic Map of Testing in Agile Software Development. In *Agile Conference (AGILE), 2012* (pp. 55-63). IEEE.

[26] Higgins, T., "Poor Requirements – What is the cause?", http://www.blueprint.com/poor_requirements_what_is_the_cause/; 2012. [Accessed on 7th October, 2017].

[27] Hendrickson, E., Exploratory Testing in an Agile context, Quality Tree Software Inc, 2011.

[28] Ghahrai, A., (2015); Why Exploratory Testing is Important in Agile Projects. Available at[https://www.testingexcellence.com/ exploratory-testing-important-agile-projects/], Accessed on 25th April, 2018.

[29] Itkonen J, M. V. Mäntylä and C. Lassenius. "How Do Testers Do It? An Exploratory Study on Manual Testing Practices", International Symposium on Empirical Software Engineering and Measurement. 2009.

[30] Cronbach, L.J., "Coefficient Alpha and the Internal Consistency of Tests," Psychometrica, vol. 16, pp. 297-334, 1951.

[31] Nunnally, J.C., & Bernste, I.A., Psychometric Theory, third ed. New York: McGraw Hill, 1994.