

# Software Process Assessment and Certification: The Reference Standard and Weight Determination using QFD-AHP

Shafinah Farvin Packeer Mohamed<sup>1</sup>, Fauziah Baharom<sup>1</sup>, Aziz Deraman<sup>2</sup> and Mazida Ahmad<sup>1</sup>

<sup>1</sup>Universiti Utara Malaysia, Malaysia, {shafinah@uum.edu.my, fauziah@uum.edu.my, mazida@uum.edu.my}

<sup>2</sup>Universiti Malaysia Terengganu, Malaysia, {a.d@umt.edu.my}

## ABSTRACT

Software process certification has become as one of the mechanisms for ensuring the quality of software. To certify that the software process implemented by an organization is good, assessors need to make decision based on various evaluation criteria that need to be weighted, besides numerous software process practices. Thus, it is a complex procedure to be performed. Hence, in this study, Quality Function Deployment (QFD) and Analytic Hierarchy Process (AHP) has been integrated to articulate the reference standard for the software process assessment and certification. QFD is adapted to organize “what” and “how” to assess software process while AHP is utilized to formulate method in determining “weight value” for each evaluation criteria. Combination of both approaches ensure that the assessment criteria are systematically organized and the decision is not made arbitrarily. This paper discusses about the reference standard used for software process assessment and certification. It contributes to the body of knowledge in the Software Certification area, since both QFD and AHP are newly introduced in the area.

**Keywords:** Software engineering, software certification, software process assessment, software quality, ESPAC Model.

## I INTRODUCTION

Software process assessment and certification is used widely as a mechanism to give conformance on the quality of software (Fauziah, Jamaiah, Aziz, & Abdul Razak, 2013; Heck, Klabber, & Eekelen, 2010; Aziz, Jamaiah, Fauziah, Amalina Farhi, & Abdul Razak, 2007). Certification is defined as “the procedure by which a third party gives written assurance that a product, process or service conforms to a specified characteristics” (Rae, Robert, & Hausen, 1995). Certification in the software industry can be implemented by using three approaches which are personnel, product and process Voas (1998). The product based approach (Heck et al., 2010; Jamaiah, 2007; Voas, 1999) is hard to be implemented without implementing the software for a certain period of time. Thus, as an alternative, process approach can be used. This is

based on the Deming’s premise that “the quality of product is influenced by the quality of process used to develop it” (Deming, 1982). This mechanism is important for customers since they need conformance on the quality of software produced to them. Commonly, software practitioners claim that they produce high quality software, however, there exist complains on customers’ dissatisfaction. Thus, by providing certification, customers will feel more confident on the quality and dependability of the software that they invest on. On the other hand, through software process assessment and certification, software practitioners will enforce themselves to implement the best practices of software process towards producing high quality software.

Concerning on software process assessment, the previous studies focus more on producing models and standards for software process improvement (SPI), for instance the ISO/IEC 15504 (O’Regan, 2002) and Capability Maturity Model Integration (CMMI Product Team, 2010). On the other hand, the ISO 9000 (Sedani & Lakhe, 2009) provides a mechanism to certify only on the quality system of an organization. Additionally, the Software Process Assessment and Certification (SPAC) Model which introduced by Fauziah, Jamaiah, Aziz and Abdul Razak (2011) assesses and certifies that a software process has been carried out effectively and efficiently in a project. Unfortunately, the agile and secure software processes are not addressed by this model. Nevertheless, both approaches have become as determinant factors to produce high quality software in today’s business environment (Merkow & Raghavan, 2010; Pressman, 2010). Moreover, the weight values are not considered in the assessment even though it involves multi criteria assessment. On the other hand, weight value allocation is important for multi criteria assessments. In line with previous studies, a research was conducted to construct Extended Software Process Assessment and Certification (ESPAC) Model which addresses the agile and secure software processes, besides includes weight values allocation in the assessment.

ESPAC Model was constructed by using the Evaluation Theory (Scriven, 1991) as the base model. Also, the outcomes from theoretical and exploratory studies (Shafinah Farvin, Fauziah, Aziz, Jamaiah, & Haslina, 2016; Shafinah Farvin,

Fauziah, & Aziz, 2014) were considered as well. This model specifies six components, which are target, evaluation criteria, reference standard, data gathering techniques, synthesis technique and assessment process (Shafinah Farvin, Fauziah, & Aziz, 2015). This paper discusses one of the components, which is the reference standard. The reference standard is developed by integrating the Quality Function Deployment (QFD) and Analytic Hierarchy Process (AHP).

This paper is started with the description on the QFD and AHP. Then, the reference standard is discussed in detail. The paper is ended with the conclusion.

## II QUALITY FUNCTIONAL DEPLOYMENT

QFD is used as a tool to translates the Voice of the Customer (VoC) into new products systematically. QFD has been adapted in various fields, such as product development, quality management, product design, management, manufacturing, customers' needs analysis, software systems, decision making and services (Lai-Kow & Ming-Lu, 2002; Bouchereau and Rowlands, 2000).

In the area of software process, it has been used for software process improvement by several studies; Richardson and Ryan (2001), Yan (2008) and more recently by Wei and Yonghui (2013). Besides, this approach is also used for evaluations, for instance, supplier evaluation (Tavassoli, Darestani, & Tavassoli, 2018) and evaluation of digital library (Garibay, Gutierrez, & Figueroa, 2010). QFD involves building the matrix which is House of Quality (HOQ), as shown in Figure 1.

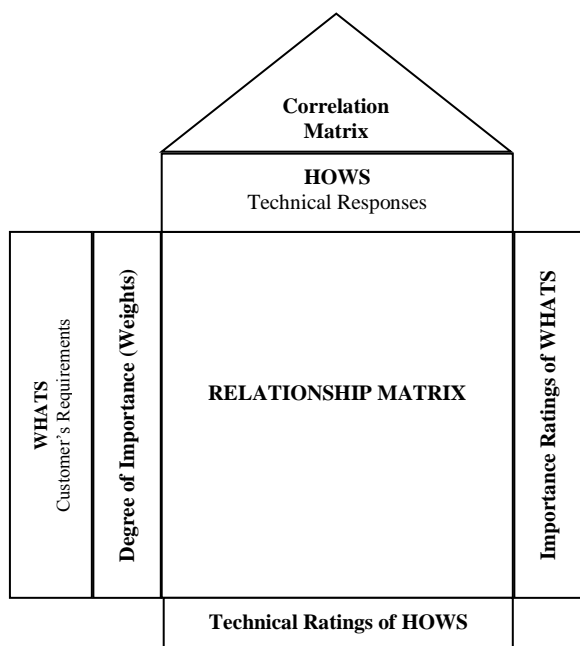


Figure 1. The Basic Structure of HOQ (Cohen, 1995).

Each sections of HOQ has its own meaning; the WHATS are the customers' requirements, the HOWs are the WHATS that are matched with the appropriate technical response along the top, the weight values for each customers' requirements are the importance rates while the rating scales is the score given for each requirements as the relationship matrix. Furthermore, the technical ratings are obtained by using the Weighted Sum Method (WSM) (Chan & Wu, 2005). Finally, the importance ratings for the WHATs are determined, which is placed on the right hand side of the HOQ. The customer requirements which obtained higher relative importance should receive higher attention for future improvements (Chan & Wu, 2005). By using QFD, the evaluation criteria and its respective weight values can be organized systematically. Thus, QFD is used to build the reference standard for the software process assessment and certification.

When using QFD, weight values determination for the WHATS is a crucial step (Garibay et al., 2010). Previously, the weight values are assigned by using absolute values. However, this can cause the decision made not too accurate and degrade the quality of decisions made (Crostack, Hackenbroich, Refflinghaus, & Winter, 2007). The Analytic Hierarchy Process (AHP) is the reliable and extensively applied technique to derive weight values (Ishizaka & Labib, 2011). The AHP and QFD have been used in combination in numerous studies, among them are Yadav and Gangele (2017), Awasthi, Sayyadi, and Khabbazian (2018) and Taghizadeh and Mohamadi (2013). Similar to the abovementioned studies, the AHP and the QFD are adapted in this study. The next section elaborates about AHP.

## III ANALYTIC HIERARCHY PROCESS

AHP is a technique that is suitable for decision making which involves multiple criteria (Saaty, 1990). It helps the decision makers to choose the best alternative by using systematic steps. To implement AHP, the evaluation criteria are organized in hierarchy, which has at least three levels. The first level is the overall goal of the problem. On the other hand, the second level is the evaluation criteria while the third level contains the alternatives. However, in this study, the hierarchy tree only contains the goal and several levels of evaluation criteria, without the alternatives. This is because AHP is used only for obtaining the weight values. The basic steps involved in the AHP technique are as provided below:

1. Identify the factors that have impact on the quality of software process.
  2. Structure the factors in a hierarchy which comprises the factors, sub factors and evaluation criteria.
  3. Construct pair wise matrixes.
  4. Perform judgements on the evaluation criteria in each pair wise matrixes by using the importance values as depicted in Table 1.
  5. Synthesize pair wise comparisons to get the weight, consistency index (CI), consistency ratio (CR). If the CR value is lower than 0.1 the weight is usable, otherwise the process is repeated.
  6. Obtain the global weight values.
- Both QFD and AHP are integrated to articulate the reference standard. The reference standard is elaborated in detail next.

**Table 1. AHP Preference Scale (Saaty, 1990).**

Intensity of Importance	Definition
1	Equal importance
3	Moderate importance of one over another
5	Essential or strong importance
7	Very strong importance
9	Extreme importance
2,4,6,8	Intermediate values

#### IV ESPAC MODEL REFERENCE STANDARD

To perform the assessment and certification, the reference standard is used. The ESPAC Model does not only define WHAT need to be assessed through the evaluation criteria, but also HOW these evaluation criteria are assessed through the list of Agile and secure software practices. Each evaluation criterion is assigned with appropriate Agile and secure software practices that need to be performed towards achieving the specified evaluation criterion. By having this structure, the assessors are guided on what they should assess during the assessment.

In order to systematically organize the WHATs and HOWs, the QFD approach is adapted. The first phase of QFD is performed by developing the HOQ (Cohen, 1995; Zultner, 1992). The other three phases of the QFD are not necessary for this study as the structures and analyzing methods are the same. There are five main areas in the HOQ adapted in this study as the reference standard: the WHATs, HOWs, relationships between WHATs and HOWs,

weight for each evaluation criterion and evaluation criteria scores, as shown in Figure 2.

The WHATS are represented by each evaluation criterion. The evaluation criteria are defined based on the factors that influence the quality of software process. These factors are elaborated in the next sub section. There are a total of 36 evaluation criteria defined. The HOWS are the Agile and secure software processes that need to be performed. A total of 189 practices defined for Agile software process and 146 practices for secure software process. The relationship matrix among WHATs and HOWs acts as the scoreboard for the assessment. A scale of five values is used; 1= Never, 2= Rarely, 3= Sometimes, 4= Often, and 5=Always, adapted from the Likert Scale (Zikmund, Babin, Carr, & Griffin, 2010). Assessors perform the assessment by assigning the score for each practice. The weight values are determined by using AHP, which enables more accurate decisions. WSM is used to get the total scores for each evaluation criterion.

Agile Requirement Engineering	WEIGHT VALUES FOR EVALUATION CRITERIA											TOTAL SCORE OF EVALUATION CRITERIA	
	1	2	3	4	5	6	7	8	9	10	11		
Completeness													
Consistency													
Accuracy													

**Figure 2. The Structure Of The Reference Standard.**

#### A. AHP Implementation

This section elaborates the AHP implementation, referring to the steps listed in Section III.

*Step 1: The Factors that Influence the Quality of Software Process.* As discussed earlier, a software will be in good quality if the process is in good quality. However, since the software process is implemented by people, there are other influential factors that can indirectly influence the software quality. Consequently, to ensure the correctness of assessment and certification outcomes, the subsequent factors are taken into consideration.

- Process: the quality of process.
- Technology: the technology used during software development.
- People: the quality of people who involved during the development.
- Project constraints: the ability to produce software on-time and within budget.
- Environment: the safety and comfort of working environment where the software is developed.

*Step 2: The Hierarchy Tree.* The factors identified in Step 1 cannot be measured directly, thus they are decomposed to sub factors and criteria. Each of the factors comprises of at least one sub factor and each sub factor has at least one evaluation criterion. Since the ESPAC Model focuses on the agile and secure software process assessment, these factors, sub factors and evaluation criteria are considered from the perspectives of both software processes. Thus, there are two hierarchy trees for this study.

Generally, the evaluation criteria encompasses the characteristics that need to be fulfilled to accomplish the effectiveness and efficiency of software process. The effectiveness is measured based on the completeness, consistency and accuracy of the process in developing software which can fulfill customers' expectations through involvement of good quality people, use of appropriate technology and stability of working environment. On the other hand, the efficiency is measured based on the capability of software process to produce software within estimated time and budget (Fauziah et al., 2013). Each of the factors is assessed based on particular criterion, which is represented by the lowest level of the hierarchy tree. The complete factors and evaluation criteria are provided in Figure 3, which focuses on the agile software process. They are organized in a hierarchical structure, as adapted from the AHP technique.

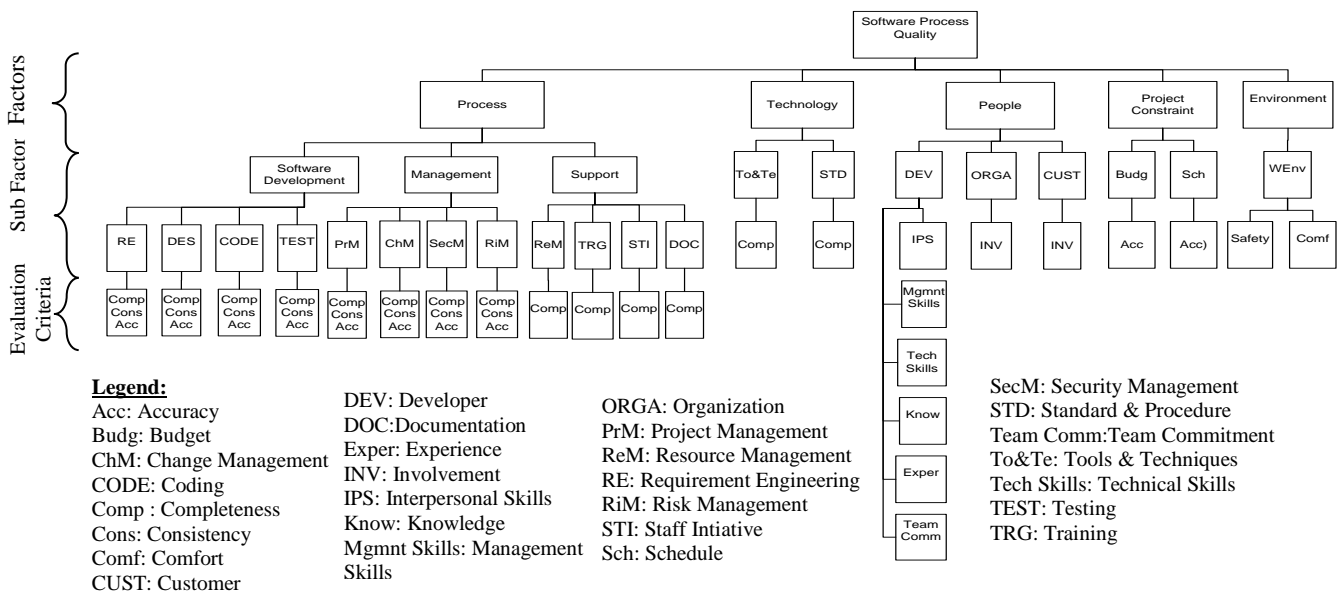


Figure 3. Hierarchy Tree.

*Step 3: Construct pair wise matrixes.* Using the evaluation criteria hierarchy tree, the weight values are obtained. To do this, the sibling criteria at each level of the hierarchy tree are organized in matrix of two dimensions whereby the compared criteria are sorted vertically in the first column and horizontally in the first row of the matrix, as depicted in Table 2. The evaluation criteria are represented by  $(C_1 \dots C_n)$ . To perform the pairwise comparison, the relative importance of each  $C_i$  are compared to the  $C_j$ , which are represented by  $a_{ij}$  by following the rules of  $a_{ij} = 1/a_{ji}$  when  $i \neq j$ , and  $a_{ii} = 1$  when  $i = j$ .

Table 2. The Pair Wise Matrix.

Criteria	$C_1$	$C_2$	..	$C_n$
$C_1$	1	$a_{1,2}$	..	$a_{1,n}$
$C_2$	$a_{2,1}$	1	..	$a_{2,n}$
.	.	.	..	.
.	.	.	..	.
$C_n$	$a_{n,1}$	$a_{n,2}$	$a_{n,3}$	$a_{n,n}$

*Step 4: Perform judgments of a pairwise comparison.* The relative importance of each two criteria in the matrix is compared, for example "is  $C_1$  is more/equally/less important than/to  $C_2$  by a factor of 2/3/4/5/6/7/8/9 ( $a_{ij}$ )". The scale of 1 to 9 by Saaty (1990) is used for this purpose (Refer Table 1).

Step 5 (i): Synthesize the pairwise comparison. After completing the pairwise comparisons, the weight values are calculated by using the Normalization of the Geometric Mean (NGM) (Hsiao, 2002). The equation for this method is provided subsequently.

$$w_i = \left( \prod_{j=1}^n a_{ij} \right)^{1/n} / \sum_{i=1}^n \left( \prod_{j=1}^n a_{ij} \right)^{1/n} \quad (1)$$

Where:

- $w_i$  = Weight of evaluation criteria  $i$
- $i$  = 1,2,...,n
- $j$  = 1,2,...,n
- $a_{ij}$  = Pairwise comparison in matrix  $ij$

Step 5 (ii): Perform Consistency Analysis. Next, to eliminate inconsistency of the judgments made, the Consistency Ratio (CR) is calculated. This is the advantage of using AHP, whereby the consistency of decisions can be revealed. The acceptable CR value is less than 0.1 (Saaty, 1990). The CR is calculated by using the subsequent equations.

$$CR = \text{Consistency Index (CI)} / \text{Random Index (RI)} \quad (2)$$

Where CI is calculated using this formula:

$$CI = (\lambda_{\max} - n) / (n-1) \quad (3)$$

Where  $n$  = number of evaluation criteria in the matrix  
 $\lambda_{\max}$  = the average value of consistency vectors

The RI is obtained for the appropriate value of 'n', as provided by Saaty.

Step 6: Obtain the Global Weight Values. The weight values obtained so far is the local weight values. The final weight values are obtained by calculating the global weight values. If the CR value for the pair wise comparison is lower than 0.1, then the global weight values can be calculated, otherwise the pairwise comparison need to be performed again. The global weight values are obtained by multiplying the local weight value of a child by its parents' local weight values (the calculation starts from the lowest level to the first level of hierarchy tree). The equation for the global weight is provided subsequently.

$$GW_i = LW_i * \prod_{j=1}^n P_j \quad (4)$$

Where:

- $GW_i$  = Global weight value for  $i^{\text{th}}$  evaluation criteria
- $LW_i$  = Local weight value for  $i^{\text{th}}$  evaluation criteria
- $P_j$  = Local weight for  $j^{\text{th}}$  parents
- $i$  = 1,2,...,n
- $j$  = 1,2,...,n

## B. Score Calculation for Evaluation Criteria

After completing the global weight value calculation for each evaluation criteria, the scores of the evaluation criteria (the WHATS) are calculated. Each global weight values of the evaluation criteria are multiplied with the total score assigned for each practices (the HOWS). This equation adapts the WSM calculation. Then, the value is divided by the maximum score that can be obtained for a particular evaluation criterion to get the relative score. The maximum score is calculated by multiplying the global weight with 5 (the maximum score for each HOWS), then multiplied with the number of HOWS. Finally, the value is aggregated by multiplying with 100 to get the percentage. The equation for this calculation is:

$$S_i = \left( GW_i * \left( \sum_{i=1}^n R_{ij} \right) \right) / (GW_i * H * 5) * 100 \quad (5)$$

Where:

- $S_i$  = Score of  $i^{\text{th}}$  evaluation criteria
- $GW_i$  = Global weight value for  $i^{\text{th}}$  evaluation criteria
- $R_{ij}$  = The total score rating for each assessed practices in matrix  $ij$
- $H$  = The number of HOWS
- $j$  = 1,2,...,n
- $i$  = 1,2,...,n

By using these scores, the quality level of each evaluation criteria are determined by referring to the Achievement Index of ESPAC Model. Please refer Shafinah Farvin et al. (2015) for further reading.

## V CONCLUSION

This paper discusses about the reference standard used for software process assessment and certification, which is the ESPAC Model. To systematically organize the evaluation criteria and the software process practices in the reference standard, the Quality Function Deployment (QFD) was utilized. Thus, through adapting the QFD, the ESPAC Model has clearly define not only WHAT are the evaluation criteria, but also HOW these evaluation criteria are assessed through the list of Agile and secure software practices. Additionally, the weight values for each evaluation criteria are obtained by using the Analytic Hierarchy Process (AHP). Combination of both approaches ensures that the assessment criteria are systematically organized and the decision is not made arbitrarily. Consequently, it provides more accurate and reliable certification results. Both QFD and AHP are newly adapted in the area of software process assessment and certification,

thus it contributes to the body of knowledge in this area. The reference standard was validated by seven software practitioners. They used the reference standard to assess their own completed projects. Their feedbacks indicated that they are satisfied with the proposed reference standard and suggested that it is practical to be implemented in the real world environment.

## REFERENCES

- Awasthi, A., Sayyadi, R., & Khabbazian, A. (2018). A combined approach integrating gap analysis, QFD and AHP for improving logistics service quality. *International Journal of Logistics Systems and Management*, 29(2), 190-214.
- Aziz Deraman, Jamaiah Yahya, Fauziah Baharom, Amalina Farhi Ahmad Fadzlah, & Abdul Razak Hamdan. (2007). Continuous quality improvement in software certification environment. *Proceedings of the International Conference on Electrical Engineering and Informatics*, 11-17.
- Chan, L. K., & Wu, M. L. (2005). A systematic approach to Quality Function Deployment with a full illustrative example. *Omega*, 33(2), 119-139. doi: 10.1016/j.omega.2004.03.010
- CMMI Product Team. (2010). CMMI for Development V1.3. (Technical Report No. CMU/SEI-2010-TR-033). Retrieved from <http://www.sei.cmu.edu/reports/10tr033.pdf>
- Cohen, L. (1995). *Quality Function Deployment: how to make QFD work for you*. Reading, MA: Addison-Wesley.
- Crostack, H. A., Hackenbroich, I., Refflinghaus, R., & Winter, D. (2007). Investigations into more exact weightings of customer demands in QFD. *Asian Journal on Quality*, 8(3), 71-80.
- Deming, W. (1982). *Out of the crisis*. Cambridge, MA: MIT Center for Advanced Engineering Study.
- Fauziah, B., Jamaiah, Y., Aziz, D., & Abdul Razak, H. (2011). SPQF: Software Process Quality Factor for software process assessment and certification. *International Conference on Electrical Engineering and Informatics*, 1-7. doi: 10.1109/ICEEI.2011.6021526
- Fauziah, B., Jamaiah, Y., Aziz, D., & Abdul Razak, H. (2013). Software Process Certification: A Practical Model for Maintaining Software Quality. *International Journal of Information Processing and Management*, 4(3), 51-61. DOI: 10.4156/ijipm.vol4.issue3.5
- Garibay, C., Gutierrez, H., & Figueroa, A. (2010). Evaluation of a digital library by means of quality function deployment (QFD) and the Kano model. *The Journal of Academic Librarianship*, 36(2), 125-132. doi: 10.1016/j.acalib.2010.01.002
- Heck, P., Klabbers, M., & Eekelen, M. (2010). A software product certification model. *Software Quality Journal*, 18(1)37-55. doi: 10.1007/s11219-009-9080-0
- Hsiao, S. W. (2002). Concurrent design method for developing a new product. *International Journal of Industrial Ergonomics*, 29(1), 41-55. doi: 10.1016/S0169-8141(01)00048-8
- Ishizaka, A., & Labib, A. (2011). Review of the main developments in the Analytic Hierarchy Process. *Expert Systems with Applications*, 38(11), 14336-14345. doi: 10.1016/j.eswa.2011.04.143
- Jamaiah Yahya. (2007). The development of software certification model based on product quality approach. (Unpublished doctoral dissertation). Universiti Kebangsaan Malaysia, Selangor, Malaysia.
- Lai-Kow, C., & Ming-Lu, W. (2002). Quality Function Deployment: a literature review. *European Journal of Operational Research*, 143(3), 463-497. doi: 10.1016/S0377-2217(02)00178-9
- Merkow, S. M. & Raghavan, L. (2010). *Secure and resilient software development*. Boca Raton: Auerbach Publications.
- O'Regan, G. (2014). Software Process Improvement. In O'Regan, G. *Introduction to software quality* (pp.199-209). Switzerland: Springer International Publishing.
- Pressman, R. S. (2010). *Software Engineering a practitioner's approach 7th Ed*. New York: McGraw-Hill Higher Education.
- Rae, A., Robert, P., & Hausen, H. L. (1995). *Software evaluation for certification principles, practice and legal liability*. England: McGraw-Hill.
- Richardson, I., & Ryan, K. (2001). Software Process Improvements in a very small company. *Software Quality Professional*, 3(2), 23-35. Retrieved from <http://www.itu.dk/~katten/speciale/Software%20Process%20Improve%20in%20a%20Very%20Small%20Company.pdf>
- Saaty, T. L. (1990). How to make a decision: the Analytic Hierarchy Process. *European Journal of Operation Research*, 48 (1), 9-26. doi: 10.1016/0377-2217(90)90057-1
- Scriven, M. (1991). *Evaluation thesaurus: fourth edition*. Newbury Park: Sage Publications.
- Sedani, C. M., Lakhe, R. R. (2009). Critical factors to attain iso 9000 certification: a survey of Indian SMEs. *International Conference on Emerging Trends in Engineering and Technology (ICETET)*, 765-764. doi: 10.1109/ICETET.2009.65
- Taghizadeh, H., & Mohamadi, P. (2013). Identifying educational services quality using Quality Function Deployment model (QFD) and Analytic Hierarchy Process (AHP). *African Journal of Business Management*, 7(15), 1250-1257. doi: 10.5897/AJBM10.1613
- Tavassoli, M. A., Darestani, S. A., & Tavassoli, S. A. (2018). Supplier selection and evaluation using QFD and ELECTRE in quality management system environment (case study: Faravari & Sakht Company). *International Journal of Productivity and Quality Management*, 24(1), 84-100.
- Shafinah Farvin Packeer Mohamed, Fauziah Baharom, Aziz Deraman, Jamaiah Yahya, & Haslina Mohd. (2016). An Exploratory Study on secure software practices among software practitioners in Malaysia. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, 8(8), 39-45.
- Shafinah Farvin Packeer Mohamed, Fauziah Baharom & Aziz Deraman. (2015). ESPAC Model: Extended Software Process Assessment and Certification Model for Agile and secure software processes. *ARPN Journal of Engineering and Applied Sciences*, 10 (3), 1364-1373.
- Shafinah Farvin Packeer Mohamed, Fauziah Baharom & Aziz Deraman. (2014). An exploratory study of Agile based software development practices. *Journal of Software Engineering and its Application*, 8 (5), 85-114. doi: <http://dx.doi.org/10.14257/ijseia.2014.8.5.09>
- Voas, J. (1998). The software quality certification triangle: crosstalk. *The Journal of Defense software engineering*, 12-14. Retrieved from <http://www.crosstalkonline.org/storage/issue-archives/1998/199811/199811-Voas.pdf>
- Voas, J. (1999). User participation-based software certification. In Vermesan, A. & Coenen, F. *Validation and Verification of Knowledge Based Systems* (pp. 267-276). US: Springer-Verlag.
- Wei, X., & Yonghui, C. (2013). Comments on Software Process Improvement methodologies using QFD. *Applied Mathematics & Information Sciences*, 7(3), 1137-1143. Retrieved from <http://t.naturalspublishing.com/files/published/68vp82lf77ig9.pdf>
- Yadav, J. S., & Gangele, A. (2017). Optimization of Selection Parameters through QFD-AHP-TOPSIS Methodology. *International Journal of Theoretical and Applied Mechanics*, 12(4), 833-844.
- Yan, S. (2008). Business oriented software process improvement based on CMM and CMMI using QFD. (Doctoral dissertation). Retrieved from [https://mospace.umsystem.edu/xmlui/bitstream/handle/10355/26432/Sun\\_2008.pdf?sequence=1](https://mospace.umsystem.edu/xmlui/bitstream/handle/10355/26432/Sun_2008.pdf?sequence=1)
- Zikmund, W. G., Babin, B. J., Carr, J. C., & Griffin, M. (2010). *Business research methods*, 8th edition. South-Western: Cengage Learning.
- Zultner, R. E. (1992). *Quality Function Deployment (QFD) for software*. American Programmer.