Software component project evaluation based on quality measure

Sara Shahzad[1], Shah Nazir[1], Syed Bakhtawar Shah Abid[1], Islam Zada[1], Rohul Amin[2]

[1]Department of Computer Science, University of Peshawar
[2]Department of Mathematics, University of Peshawar
{sara, shahnzr, raminmath}@upesh.edu.pk, {sbshah27, islamzada28}@yahoo.com

Abstract: Component based software development projects involve the incorporation of different components which are found in different component based project repositories. Quality evaluation of these components is an essential part of component based software development lifecycle, and it consumes significantly massive amount of organization's time and effort. The decision about the evaluation of a good component is very hard and critical. There are a number of quality factors which influence the selection of a particular component. Informal evaluations, not specifically following necessary quality, functional and strategic criteria, result in schedule delays as well as lower entire system quality. In this paper, we have discussed the applicability of Fuzzy Analytic Network Process (FANP) for software component project evaluation, which is based on quality measure defined in ISO/IEC 25010:2011. The methodology carefully derives meaningful evaluation from complex and uncertain decision framework. A case study is presented for the evaluation of component based project and for the validation of the proposed method.

[Sara Shahzad, Shah Nazir, Syed Bakhtawar Shah Abid, Islam Zada, Rohul Amin . **Software component project evaluation based on quality measure.** *Life Sci J* 2014;11(10):98-107]. (ISSN:1097-8135). http://www.lifesciencesite.com. 16

Keywords: ISO/IEC 25010, Software Components, Fuzzy ANP.

1. Introduction

Components are the basic building blocks of a software system, which play a fundamental role in the development of component based systems. Component based software engineering (CBSE) approach facilitates the development of efficient, cost effective and well tested software, and then easy maintenance afterwards[1]. Software systems are becoming large and more complex, which ultimately emphasize the need to reuse commercial off-the-shelf (COTS) components and the available free and open source software (FOSS) components. Considering the economic concerns, there is propensity towards components based system. Component based software development has resulted in a broad-based interest in supporting compositional approaches to software development. This has led to considerable investment in research and development in software reuse, defining industry standards for component interaction, defining domain specific architectures, toolkits and many other related areas. With an increase in the availability of COTS and FOSS software components, developers usually face a problem of making a hard decision about the selection of a component from a pool of available candidates [2].

This paper proposes a method for evaluating components to be integrated in a system. The goal of this method is to evaluate the component projects based on quality measure. The proposed methodology for component evaluation incorporates fuzzy ANP process toward the attributes of ISO/IEC 25010:2011 quality model [3]. The Analytic network process (ANP) has the advantages over other methods as it is known to be best approach for comparing weights of any object about which a decision has to be made. ANP is a powerful tool to deal with complex networks of elements in decision making [4]. As the decisions regarding specific components are mostly made on ad-hoc basis, therefore, fuzzy ANP is well suited for such ad-hoc-based and uncertain decisions.

The paper is organized as follows. Section 2 presents a related work of the available techniques being used for components evaluation. Section 3 discusses the Fuzzy ANP approach which is used as the main methodology in this research. A Case study for the validation of the proposed method is presented in section 4. Derivation of weights from expert opinion is discussed in section 5. The paper concludes in section 6.

2. Related Work

The reuse of software has an encouraging impact on the software cost, quality and productivity. A number of diverse approaches are used for the evaluation, prioritization and adoption of the most suitable component(s). Y. Il-Chul et al. presented a prioritization approach for component compatibility which tests the most significant details and then analyzes the work over different computer network. They assessed the methodology on two large middleware systems [5]. M.L.Tseng et al. argue and discuss the imperative decision criteria which consist of cleaner production accomplishment for the prioritization of an efficient system. The model is top down structure for the cleaner production accomplishment and prioritization [6]. S. Nazir et al. proposed a fuzzy logic model for the selection of software components and designed fuzzy rules from different membership function to select the most suitable and appropriate components [7]. Different fuzzy rules are designed and on the basis of these rules inputs are provided to the model for the selection of a component. W. Zhiqiao proposed an integrated decision model for the selection of reuse component scenario. The purpose of this model is to minimizing cost of developing the system [8]. J. F. Tang et al. proposed an optimization model for software component selection. The model is used to help software developer in selecting the most suitable software component among available components. The model solves the problem of reusability and compatibility [9].

C. K. Kwong et al. presented a model for the optimization of software component selection. The main objective of this model is to maximize functional performance and cohesion, and to minimize the coupling among software modules [10]. L. Mikhailov worked on to proposed pair wise comparisons for deriving priorities and is based on α-cut decomposition. The evaluations of pair wise comparisons show the huge fulfillment and satisfaction of the decision maker. Fuzzy preference programming method is used for deriving most favorable priorities. Updated linear fuzzy preference programming method is also used. The methods are validated by using certain examples [11]. S. Boran et al. worked on fuzzy analytic network process for the prioritization of Six Sigma project. They mentioned that, for the designing of project needs much human inputs, time resources and money which is some time limited. So to avoid such phenomena prioritization becomes more reliable. The method is validated by using a case study [12]. X. Cai. et al. showed the merits and demerits of software components technologies and the features which they inherit. They proposed quality assurance model for component based software development. The method spotlights superiority analysis of components, maturity, customization, design, and amalgamation of components and their maintenance [13]. A. C. Dias-Neto and G. H. Travassos designed a strategy to select model based testing for software projects. It is based on knowledge which telling model based testing strategy and their attributes [14]. S. Nazir et al. used fuzzy ANP for test case prioritization of components [15]. The method incorporates attributes that are controllability, observeability, isolatability, separation of concern, automatibility and heterogeneity.

M. Malawski et al. presented an approach for the scientific applications which are running on grid and cloud environment [16]. It is based on two principles which are component based programming model and the technology for virtualization of the environment. J. Dongarra and V. Eijkhout Planned a Self-Adapting Numerical Software (SANS) system to meet the challenges for the successful management of complex gird infrastructure [17]. Zheng Liu and Qi Xu used principal component analysis for the preprocessing and after that analyzed the assessment methods, which are TOPSIS and ELECTRE methods. When the results are inconsistent then social choice approach is deciding the appropriate [18]. P. Naker directed the problem of releases in software evolution. It describes the selection and ranking of the software components based on some features by applying search based software engineering. The approach is applied on the telecommunication organization by using greedy and simulated annealing algorithm. Then the results are compared with the opinion of the experts and hence results that two different approaches realistically best perform the expert decision approach [19].

Currently fuzzy AHP/ANP are favorable than conventional methods for the evaluation of projects. In the proposed study, fuzzy ANP approach is followed in order to evaluate the components based projects.

3. Fuzzy ANP and Component based project evaluation

Analytic hierarchy process (AHP) is the hierarchal representation of elements relation, while ANP is the network representation in which elements provide interdependencies and feedback [11]. The relation of elements is based on goal, criteria, sub criteria, and alternatives. The calculations of pair wise comparisons of ANP are time consuming, but it leads towards a real situation deemed for feedback and interdependence among criteria. This method gives adoptability for designing a decision model. ANP is simple, intuitive, and easily understandable for the decision maker, and also it is used in crisp decision making problems which deals with the un-balanced judgments [20]. The following diagram visually shows the AHP and ANP method.

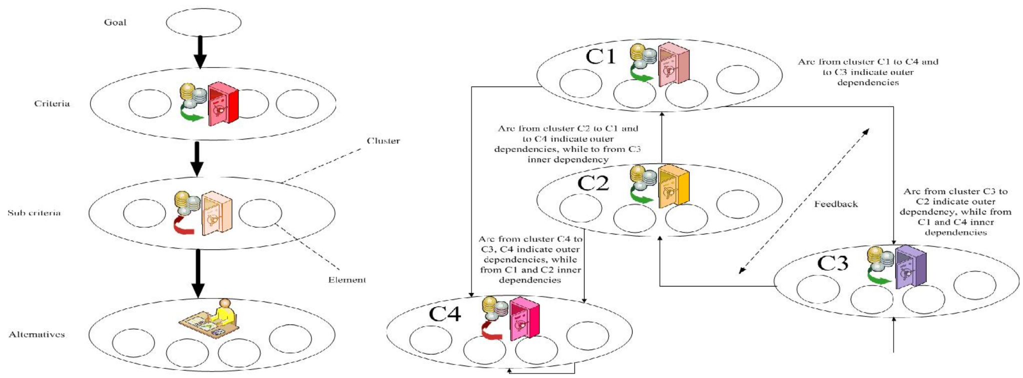

Figure 1. Analytic hierarchy/network process

Fuzzy set concept was introduced by Zadeh in 1965. Fuzzy set is the extensive form of traditional sets. It is useful in situations of uncertainty and vagueness. Fuzzy set is described by membership function and is extremely beneficial where decision making is about uncertainty and vagueness. Here, the decision can be made in linguistic variables (low, strong, very strong and so on) instead of numbers, and these linguistics variables allow precise modeling. It has the degree of membership function in range between interval [0, 1]. In the proposed method the membership functions named as low, medium and upper bound are defined. Also, to plot fuzziness triangular membership functions are defined and used to represent opinions. The triangular membership function has three parameters (l, m, u), which are defined as $l \leq m \leq u$.

The vagueness and uncertainty of judgment on decision maker the crisp pair wise comparison are insufficient in ANP to imprecise the correct judgment of decision makers. Judgment can be made based on the nine points scale (1- 9) given by Saaty [19]. The qualitative nature of the scale for elements is equally, moderately, strongly, and very strongly. This scale is converted into quantitative nature of weights as 1, 3, 5, 7 and 9 respectively. Others 2, 4, 6 and 8 are the middle values representing the range between the lower and higher values. Decision makers face the problem of providing exact values in pair wise comparison in uncertain situation. ANP method fails to resolve the problem of uncertainty and vagueness. Fuzzy numbers are incorporated to resolve such problem. A best way to handle uncertain environment is to comprise the ratios in the form of fuzzy sets. We propose fuzzy analytic network process (FANP) scheme for the evaluation of components based projects.

To analyze and to evaluate a component we must determine how to certify the quality of the component project. Quality components are the foundation to guarantee the quality of the whole component based software system. The proposed method incorporates the attributes of ISO/IEC 25010:2011 to help in the evaluation of component based projects. The details of FANP for proposed method are discussed in following diagram.

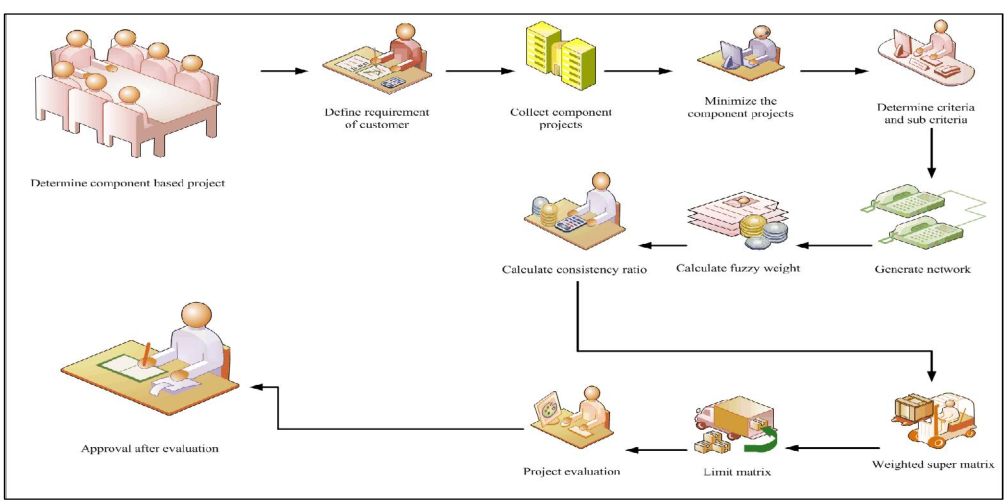

Figure 2. Proposed method based on fuzzy ANP

Figure 2 can be clearly translated into a working Fuzzy ANP algorithm for component projects evaluation, as given below.

1. *Determine* ***component based project*** *proposal*
2. *Define* ***requirement*** *of customer*
3. ***Collect*** *available component based projects*
4. ***Minimize*** *the available projects*
5. *Determine the* ***criteria and sub criteria***
6. ***Generate the network***
7. ***Calculate*** *fuzzy set weights*
8. *Calculate* ***consistency ratio***
9. *If consistency ration is >* ***10 %***
10. ***Normalize*** *the matrix*
11. *Else*
12. ***Calculation done***
13. *Obtain* ***un-weighted super*** *matrix*
14. *Obtained* ***weighted super*** *matrix*
15. ***Limit*** *matrix*
16. ***Evaluation***
17. ***Approval*** *after evaluation*

According to the algorithm given above, the very first phase of the method is to understand the objectives of the proposed component based project. All attribute of the project proposal should be explicitly clear to avoid any misunderstanding on the behalf of the stakeholder of the system. The project proposal must provide an unambiguous definition of user/customer requirements to carry on with a smooth design and development process and timely completion of the project. According to software engineering principles, if changes occur after the design and/or development of the system, it increases the cost of the system and upsurge toward late delivery of the system. When requirements of the customer are well known and clearly understandable, then according to that requirements select the appropriate and suitable component from the available repository. Several components are normally selected from the available repository. After the selection, follow and adopt that project which is most suitable according to the exact requirements of the customer. In this phase non-compatible components projects are filtered out after the selection. Then on the basis of customer requirements criteria and sub criteria are mentioned for the proper designing and development of the system. After that the network of elements is generated and the elements are plotted in such a way that their feedback among elements is possible. The relationship among elements should be clearly shown. After generating the network of elements their pair wise comparisons are calculated based on fuzzy set (0, 1). Membership functions are plotted in range of interval 0 and 1 as required. The following figure shows the linguistic scale of weights.

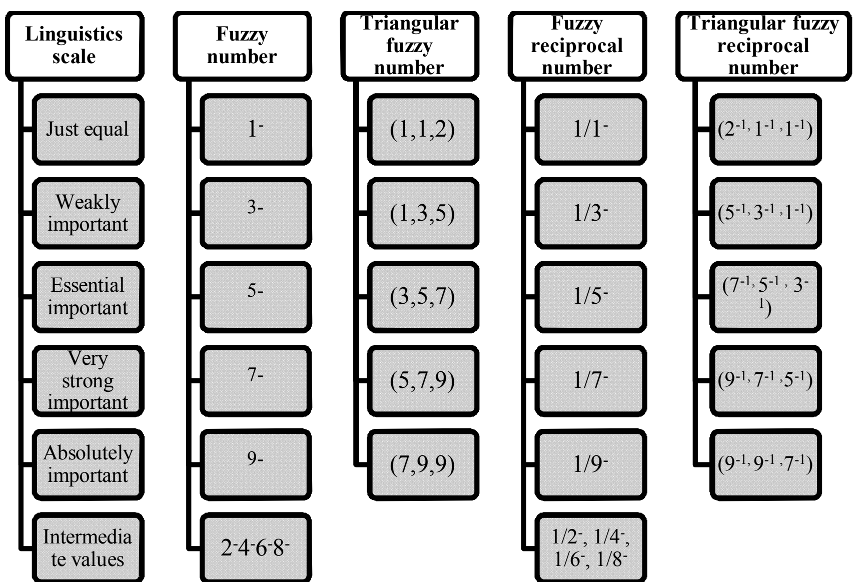

Figure 3. Linguistics scale for weights

Consistency ratio and random consistency ratio should be found out. Random consistency (RI) table, given by Saaty [21] is shown in table 1.

Table 1. Random consistency index

n	1	2	3	4	5	6	7	8	9
RI	0	0	0.6	0.9	1.1	1.2	1.3	1.4	1.5

All pair wise comparisons are summarized in a matrix in the form of un-weighted super matrix. Here, the sum of the column total is greater than 1, so the matrix is normalized (based on the above algorithm) until its column total is less than or equal to 1, hence forming weighted super matrix. Limit matrix is obtained by raising the power of weighted super matrix [21]. Raise the power until the row elements of the weighted super matrix stable and same. In limit matrix the decision maker can make decision about the quality of components project. After completing all the necessary steps, evaluate the projects according to the calculations done. The project with highest priority among all available is adopted for the system to be designed. Such evaluation and prioritization of projects leads the developer to a successful system.

4. Case Study

The presented methodology is tested for a specific project of an academic organization which proposed for a component based software system. The case study incorporates attributes of ISO/IE-25010:2011 quality model [3]. The process is visually presented in figure 4.

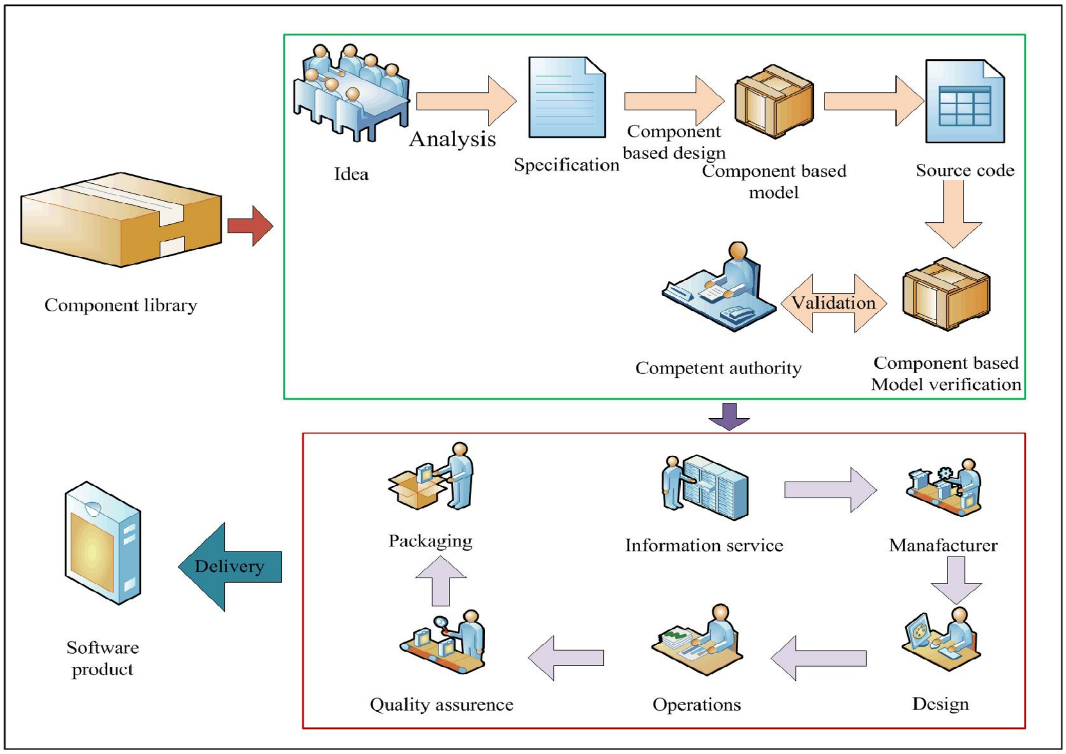

Figure 4. Component based software development process

The proposed evaluation model is depicted in figure 2. It can be further elaborated as follows:

4.1 Collect available projects

Available components based projects are: component based project 1 (CP 1), component based project 2 (CP 2) and component based project 3 (CP 3).

4.2 Criteria and sub criteria

The proposed method applied on ISO-25010 quality model for the evaluation of projects. The attributes of the model are; effectiveness, efficiency, satisfaction, safety and usability. Sub criteria of the attributes are also given, as mentioned in figure 5.

4.3 Generate network

Generate the network of goal, criteria, sub criteria and alternatives of the available components, criteria and sub criteria (attributes of ISO 25010), and the alternatives. The components, criteria and their sub criteria of elements are ordered as given below in figure.

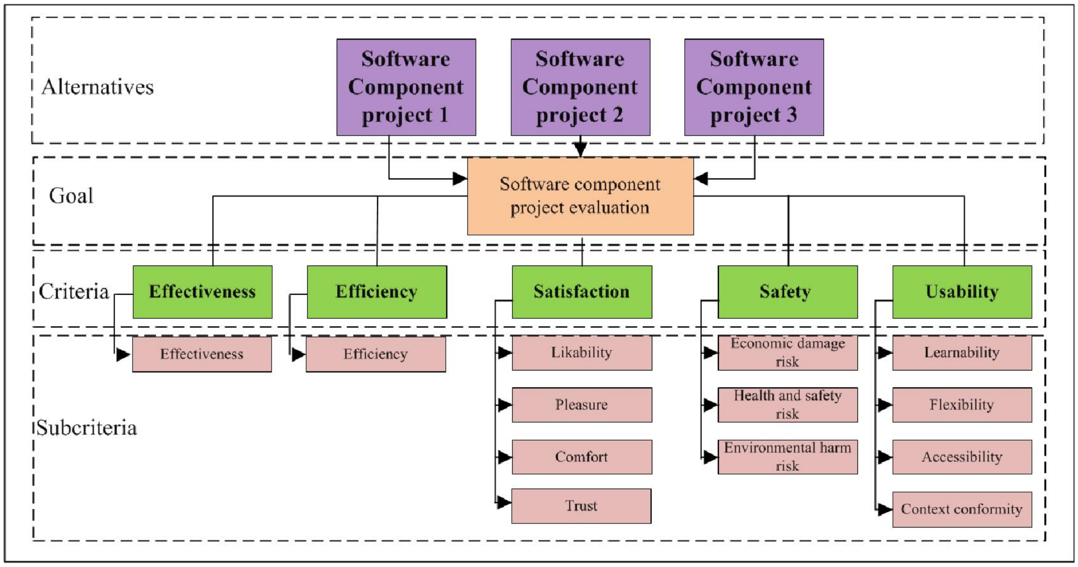

Figure 5. Proposed evaluation model for ISO-25010 Quality model

5. Derivation of weights from expert opinion

Weights for various attributes of projects are result of a conscious critical thinking among multiple domain experts. A set of eights experts were chosen who were given complete overview of the model and attributes that contribute in its formation. These experts after a long debate came up with relative importance of each parameter in pair wise qualitative comparisons. Among experts there were variations in opinion while outlier values were excluded. This gave the relative consensus weight of each parameter. In real life facts are important but opinion also plays an important role. Opinion is basically a quintuple [22] (e_j, f_{jk}, so_{ijkl}, h_i, t_l), where e_j is target entity, f_{jk}, *is* feature of entity e_j, so_{ijkl} is the value of the opinion on feature f_{jk}, of entity e_j at the time t_l and so_{ijkl} value will be between scale of 0 and 9 as given in table 2, h_i is opinion holder and t_l is the time of opinion expression. The process of expert opinion is shown below in figure.

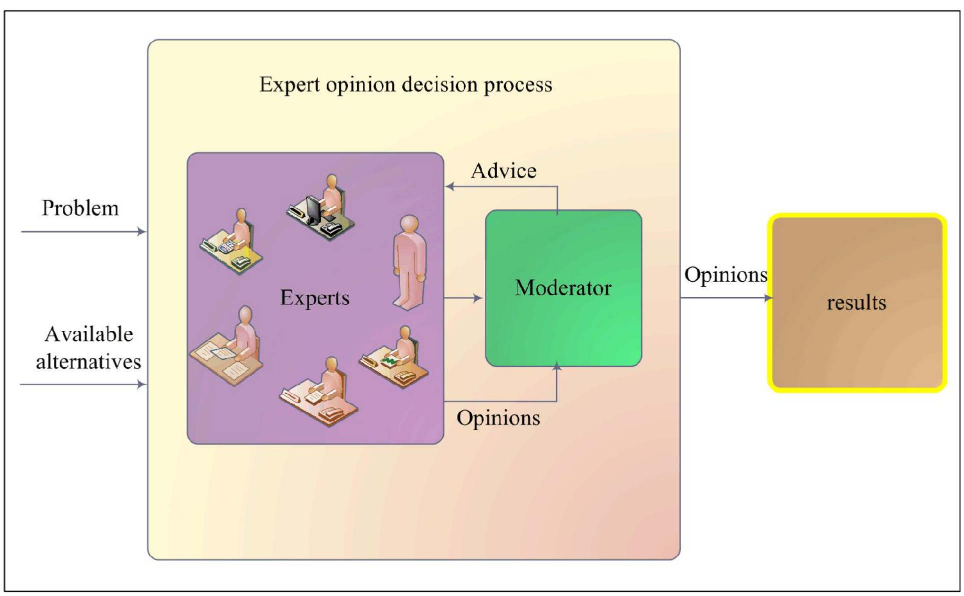

Figure 6. Expert opinion and derivation of weights

The following table adopted from Saaty [23] was given to domain experts and were told to assign weight to attributes.

Table 2. Saaty scale for weights

9		7		5		3		**1**	Importance of elements are **equal** Decision maker is indifferent between elements
9		7		5		**3**		1	First element is **moderately** more important than second one First element is moderately preferred to second one
9		7		**5**		3		1	First element is **strongly** more important than second one First element is strongly preferred to second one
9		**7**		5		3		1	First element is **very strongly** more important than second one First element is very strongly preferred to second one
9		7		5		3		1	First element is **extremely** more important than second one First element is extremely preferred to second one
	8		**6**		**4**		**2**		Intermediate values
1	1/2	1/3	1/4	1/5	1/6	1/7	1/8	1/9	Use reciprocals for inverse comparisons

Based on above table, values of attributes for ISO 25010 would be in the form as follows:

- Sometime effectiveness will be equal (1), moderately (3), strongly (5), very strongly (7), extremely (9) and sometime will be in the range of reciprocal values (1/2, 1/3, ¼, 1/5, 1/6, 1/7, 1/8, 1/9).
- Efficiency will be equal (1), moderately (3), strongly (5), very strongly (7), extremely (9) and sometime will be in the range of reciprocal values (1/2, 1/3, ¼, 1/5, 1/6, 1/7, 1/8, 1/9).
- Satisfaction will be equal (1), moderately (3), strongly (5), very strongly (7), extremely (9) and sometime will be in the range of reciprocal values (1/2, 1/3, ¼, 1/5, 1/6, 1/7, 1/8, 1/9).
- Safety will be equal (1), moderately (3), strongly (5), very strongly (7), extremely (9) and sometime will be in the range of reciprocal values (1/2, 1/3, ¼, 1/5, 1/6, 1/7, 1/8, 1/9).
- Usability will be equal (1), moderately (3), strongly (5), very strongly (7), extremely (9) and sometime will be in the range of reciprocal values (1/2, 1/3, ¼, 1/5, 1/6, 1/7, 1/8, 1/9).

Table 3. Weights range for attributes

Effectiveness	Efficiency	Satisfaction	Safety	Usability
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	4	4	4
5	5	5	5	5
6	6	6	6	6
7	7	7	7	7
8	8	8	8	8
9	9	9	9	9
1/2	½	1/2	1/2	1/2
1/3	1/3	1/3	1/3	1/3
1/4	¼	1/4	1/4	1/4
1/5	1/5	1/5	1/5	1/5
1/6	1/6	1/6	1/6	1/6
1/7	1/7	1/7	1/7	1/7
1/8	1/8	1/8	1/8	1/8
1/9	1/9	1/9	1/9	1/9

The fuzzy pair wise comparisons of available component project with respect to their criteria are given below. Table 4, 5 and 6 show the comparisons of quality attribute with respect to projects. After opinion mining of the experts weights, the pair wise comparisons are given below:

Table 4. Fuzzy pair wise comparison w.r.t. component project 1

	Effectiveness	Efficiency	Satisfaction	Safety	Usability	E.V.
Effectiveness	(1,1, 1)	(1, 3, 5)	(1, 2,3)	(1, 3, 5)	(3, 5, 7)	0.393
Efficiency	(1/5,1/3,1)	(1,1, 1)	(1, 2,3)	(2, 4, 6)	(1, 2,3)	0.240
Satisfaction	(1/3,1/2,1	(1/3,1/2,1	(1,1, 1)	(1, 2,3)	(2, 4, 6)	0.188
Safety	(1/5,1/3,1)	(1/6,1/4,1/2)	(1/3,1/2,1	(1,1, 1)	(1, 3, 5)	0.114
Usability	(1/7,1/5,1/3)	(1/3,1/2,1	(1/6,1/4,1/2)	(1/5,1/3,1)	(1,1, 1)	0.064

Table 5. Fuzzy pair wise comparison w.r.t. component project 2

	Effectiveness	Efficiency	Satisfaction	Safety	Usability	E.V.
Effectiveness	(1,1, 1)	(1, 2,3)	(1, 2,3)	(3, 5, 7)	(1, 3, 5)	0.373
Efficiency	(1/3,1/2,1	(1,1, 1)	(1, 3, 5)	(1, 3, 5)	(3, 5, 7)	0.306
Satisfaction	(1/3,1/2,1	(1/5,1/3,1)	(1,1, 1)	(1, 2,3)	(1, 2,3)	0.151
Safety	(1/7,1/5,1/3)	(1/5,1/3,1)	(1/3,1/2,1	(1,1, 1)	(1, 2,3)	0.095
Usability	(1/5,1/3,1)	(1/7,1/5,1/3)	(1/3,1/2,1	(1/3,1/2,1	(1,1, 1)	0.075

Table 6. Fuzzy pair wise comparison w.r.t. component project 3

	Effectiveness	Efficiency	Satisfaction	Safety	Usability	E.V.
Effectiveness	(1,1, 1)	(1/5,1/3,1)	(1/3,1/2,1	(1/5,1/3,1)	(1/5,1/3,1)	0.080
Efficiency	(1, 3, 5)	(1,1, 1)	(1/3,1/2,1	(1/5,1/3,1)	(1/7,1/5,1/3)	0.114
Satisfaction	(1, 2,3)	(1, 2,3)	(1,1, 1)	(1/3,1/2,1	(1/6,1/4,1/2)	0.140
Safety	(1, 3, 5)	(1, 3, 5)	(1, 2,3)	(1,1, 1)	(1/3,1/2,1	0.245
Usability	(1, 3, 5)	(3, 5, 7)	(2, 4, 6)	(1, 2,3)	(1,1, 1)	0.422

Table 7-11 are used for comparison of projects with respect to their quality attributes.

Table 7. Fuzzy pair wise comparison w.r.t. Effectiveness

	C P 1	C P 2	C P 3	E.V
C P 1	(1,1,1)	(1,3,5)	(2,4,6)	0.549
C P 2	(1/5,1/3,1)	(1,1,1)	(1,2,3)	0.275
C P 3	(1/6,1/4,1/2)	(1/3,1/2,1)	(1,1,1)	0.137

Table 8. Fuzzy pair wise comparison w.r.t. Efficiency

	C P 1	C P 2	C P 3	E.V
C P 1	(1,1,1)	(1,2,3)	(3,5,7)	0.553
C P 2	(1/3,1/2,1)	(1,1,1)	(1,2,3)	0.277
C P 2	(1/7,1/5,1/3)	(1/3,1/2,1)	(1,1,1)	0.138

Table 9. Fuzzy pair wise comparison w.r.t. Satisfaction

	C P 1	C P 2	C P 3	E.V
C P 1	(1,1,1)	(1,3,5)	(3,5,7)	0.519
C P 2	(1/5,1/3,1)	(1,1,1)	(1,3,5)	0.260
C P 3	(1/7,1/5,1/3)	(1/5,1/3,1)	(1,1,1)	0.130

Table 10. Fuzzy pair wise comparison w.r.t. Safety

	C P 1	C P 2	C P 3	E.V
C P 1	(1,1,1)	(1,2,3)	(3,5,7)	0.553
C P 2	(1/3,1/2,1)	(1,1,1)	(1,2,3)	0.277
C P 3	(1/7,1/5,1/3)	(1/3,1/2,1)	(1,1,1)	0.138

Table 11. Fuzzy pair wise comparison w.r.t. Usability

	C P 1	C P 2	C P 3	E.V
C P 1	(1,1,1)	(1,2,3)	(5,7,9)	0.527
C P 2	(1/3,1/2,1)	(1,1,1)	(1,2,3)	0.263
C P 3	(1/9,1/7,1/5)	(1/3,1/2,1)	(1,1,1)	0.132

5.1 Weighted super matrix

Details regarding weighted super and limit matrix are given in methodology section.

Table 12. Weighted super matrix

		Quality attributes					Available Projects		
		Effectiveness	Efficiency	Satisfaction	Safety	Usability	C P1	C P 2	C P 3
Quality attributes	Effectiveness	0.00	0.00	0.00	0.00	0.00	0.39	0.37	0.08
	Efficiency	0.00	0.00	0.00	0.00	0.00	0.24	0.31	0.11
	Satisfaction	0.00	0.00	0.00	0.00	0.00	0.19	0.15	0.14
	Safety	0.00	0.00	0.00	0.00	0.00	0.11	0.10	0.24
	Usability	0.00	0.00	0.00	0.00	0.00	0.06	0.08	0.42
Available Projects	C P 1	0.55	0.55	0.52	0.55	0.53	0.00	0.00	0.00
	C P 2	0.28	0.28	0.26	0.28	0.26	0.00	0.00	0.00
	C P 3	0.14	0.14	0.13	0.14	0.13	0.00	0.00	0.00

5.2 Limit matrix

Table 13. Limit matrix

		Quality attributes					Available Projects		
		Effectiveness	Efficiency	Satisfaction	Safety	Usability	C P1	C P 2	C P 3
Quality attributes	Effectiveness	0.013	0.013	0.013	0.013	0.013	0.000	0.000	0.000
	Efficiency	0.009	0.009	0.009	0.009	0.009	0.000	0.000	0.000
	Satisfaction	0.006	0.006	0.006	0.006	0.006	0.000	0.000	0.000
	Safety	0.005	0.005	0.005	0.005	0.005	0.000	0.000	0.000
	Usability	0.004	0.004	0.004	0.004	0.004	0.000	0.000	0.000
Available Projects	C P 1	0.000	0.000	0.000	0.000	0.000	0.021	0.021	0.021
	C P 2	0.000	0.000	0.000	0.000	0.000	0.011	0.011	0.010
	C P 3	0.000	0.000	0.000	0.000	0.000	0.005	0.005	0.005

5.3 Component based project (best)

The following figure clearly shows that CP3 is the top most component based project followed by CP1 and then CP2.

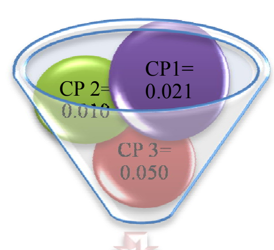

Figure 7. Component based project evaluation results

Conclusion

FANP is an excellent tool to handle uncertainty and vagueness by combining fuzzy sets and analytic network process. In this study, software component based projects are evaluated based on ISO-25010 quality model. The proposed method follows FANP approach to evaluate component projects which are based on quality measure. In the case study, several component projects were available. Initially, from among them, the top most three component projects were selected for evaluation. The ANP approach was not sufficient for modeling the component projects evaluation, due to the reason that usually the weights derived from the experts' opinions give rise to an ambiguous and uncertain situation. To overcome this problem of uncertainty and to resolve the situation of ambiguity fuzzy ANP approach is used. FANP presents a more precise picture of the decision making process. The proposed methodology helps the developer to develop more precise and well functional component based software system.

References

1. Mohagheghi P and Conradi R. Quality, productivity and economic benefits of software reuse: a review of industrial studies. *Empirical Softw. Engg.*, 2007 vol. 12, pp. 471-516.
2. Meyer B, The grand challenge of Trusted Components, in *Proceedings of the 25th International Conference on Software Engineering (ICSE '03). IEEE Computer Society,* Washington, DC, USA, 2003, pp. 660-667.
3. Wagner S, Quality Models, in *Software Product Quality Control*: Springer-Verlag Berlin Heidelberg, 2013.
4. Palanisamy P, Zubar A, and Kapoor S, A Model for Supplier Selection using Analytic Network Process, in *Tenth International Conference on Operations and Quantitative Management, ICOQM-10* Nashik, India, 2011, pp. 808-814.
5. Yoon I-C, Sussman A, Memon A, and Porter A, Prioritizing component compatibility tests via user preferences, in *IEEE International Conference on Software Maintenance (ICSM)*, 2009, pp. 29-38.
6. Tseng M L, Lin Y H, Chiu A S F, and Liao J C H. Using FANP approach on selection of competitive priorities based on cleaner production implementation: a case study in PCB manufacturer, Taiwan. *Clean Technologies and Environmental Policy,* 2008 vol. 10, pp. 17-29.
7. Nazir S, Khan M A, Anwar S, Khan H, and Nazir M, "A Novel Fuzzy Logic Based Software Component Selection Modeling," in *International Conference on Information Science and Applications (ICISA)*, 2012, pp. 1-6.
8. Zhiqiao W, Kwong C K, Tang J, and Chan J W K. Integrated model for software component selection with simultaneous consideration of implementation and verification. *Computers & Operations Research,* 2012 vol. 39, pp. 3376-3393.
9. Tang J F, Mu L F, Kwong C K, and Luo X G. An optimization model for software component selection under multiple applications development. *European Journal of Operational Research,* 7/16/ 2011 vol. 212, pp. 301-311.
10. Kwong C K, Mu L F, Tang J F, and Luo X G. Optimization of software components selection for component-based software system development. *Computers & Industrial Engineering,* 2010 vol. 58, pp. 618-624.
11. Mikhailov L. Deriving priorities from fuzzy pairwise comparison judgements. *Fuzzy Sets and Systems,* 2003 vol. 134, pp. 365-385.
12. Boran S, Yazgan H R, and Goztepe K. A fuzzy ANP-based approach for prioritising projects: a Six Sigma case study. *Int. J. Six Sigma and Competitive Advantage,* 2011 vol. 6, pp. 133-155.
13. Cai X, Lyu M R, and Wong K-F, Component-Based Software Engineering: Technologies, Development Frameworks, and Quality Assurance Schemes, in *Proceedings of the Seventh Asia-Pacific Software Engineering Conference*, Washington, DC, USA, 2000, pp. 372-379.
14. C D-N A and H T G. Model-based testing approaches selection for software projects. *Information and Software Technology,* 2009 vol. 51, pp. 1487-1504.
15. Nazir S, Shahzad S, Mukhtar N, Khan H, Zada I, Nazir M, and Amin R. Test case prioritization for components using FANP. *Life Sci J,* 2014 vol. 11, pp. 504-511.
16. Malawski M, Gubała T, and Bubak M. Component-based approach for programming and running scientific applications on grids and clouds. *International Journal of High Performance Computing Applications,* 2012 pp. 275-295.
17. Dongarra J and Eijkhout V. Self-Adapting Numerical Software for Next Generation Applications. *International Journal of High Performance Computing Applications,* 2003 vol. 17, pp. 125-131.
18. Liu Z and Xu Q, Comparison and Prioritization among Evaluation Models with Principal Component Analysis in Suppliers Selecting, in

International Conference on Management and Service Science (MASS), 2011, pp. 1-4.
19. Baker P, Harman M, Steinh¨ofel K, and Skaliotis A, Search Based Approaches to Component Selection and Prioritization for the Next Release Problem, in *22nd IEEE International Conference on Software Maintenance, 2006. ICSM '06* 2006, pp. 176-185.
20. Saaty T L, Fundamental of the analytic network process, in *ISAHP* Kobe, Japan, 1999, pp. 1-14.
21. Saaty R W and Saaty T L, *SUPER DECISIONS, Software for Decision Making with Dependence and Feedback*. Pittsburgh, 2003.
22. Liu B, *Sentiment Analysis and Opinion Mining*: Morgan & Claypool, 2012.
23. Saaty T L. Relative measurement and its generalization in decision making why pairwise comparisons are central in mathematics for the measurement of intangible factors the analytic hierarchy/network process. *Revista de la Real Academia de Ciencias Exactas Físicas y Naturales A Matemáticas (RACSAM),* 2008 vol. 102, pp. 251-318.

5/29/2014