Multicriteria Decision Analysis Method for Evaluation of Software Architectures

S. Vijayalakshmi
Department of IT,
ChristCollege of Engg. & Technology,
Puducherry– 605 010, India,

Dr. G. Zayaraz,
Department of CSE,
Pondicherry Engg. College,
Puducherry– 605 014, India,

Dr. V. Vijayalakshmi,
Department of
ECE,
Pondicherry Engg.
College,Puducherry– 605 014, India,

ABSTRACT

Software architectures is a critical aspect in the design and development of software. Architecture of software is a collection of design decisions that are expensive to change. Software architectures are generally designed with particular functional and non-functional requirements. Organizations often need to choose software architecture for future development from several competing candidate architectures. The various stakeholders' quality requirements need to be considered collectively to describe the quality requirements of the envisioned system and therefore build the basis for the comparison and selection criteria. Given the impact that software architecture has on a project's success, the need to choose the right architecture assumes significance. In this paper, a new architecture selection method based on multicriteria decision analysis have been developed and validated using a suitable case study.

Keywords

Quality attributes; Software architecture; Multicriteria Decision Making.

1. INTRODUCTION

Software architectures are abstract design artifacts of the software system to be developed. They are usually constructed from the functional and non-functional requirements of the software system. Choice of which alternative architecture to go with is a crucial part in any software development as this choice affects the quality of final software product. Conventional qualitative architecture evaluation techniques discussed in [1, 2] and quantitative selection technique in [2 - 4] are analyzed to identify their limitations. In order to overcome the limitations and challenges, a quantitative evaluation method based on MCDA methods [5] is proposed. The existing evaluation method provides the rationale for architecture selection process by measuring the conformance to requirements of each candidate architecture.

Architectures, both newly developed and reused have to be rigorously evaluated for its conformance to requirements, as it directly affects the quality of final software product. Choice on architecture alternatives is made on the basis of stakeholders' expectations and preferences. Stakeholders are responsible for making crucial design decisions. Requirements of stakeholders are discrete and preferences may also vary.

The selected architecture may undergo small changes (architectural degeneration) in later phases of the software life cycle, which leads to repetition of the entire evaluation. Thus, the process of architecture evaluation is a complex task. In this research work, an attempt has been made to propose a quantitative evaluation method based on MCDA methods and Multicriteria fuzzy decision making technique. The proposed method also models the variation in preference according to changes in the architecture structure, which avoids the necessity to repeat the entire evaluation process.

Making decision is one of the most fundamental activities of human beings. The subject of decision making is the study of how decisions are actually made and how they can be made better or more successfully. In multicriteria decision analysis (MCDA) problems, a set of architectures are evaluated according to a number of quality attributes. Each quality attribute induces a particular ordering of the architectures and we need a procedure to construct an overall preference ranking. The number of quality attributes in multicriteria decision making is always assumed to be finite and in addition the number of considered architectures is also finite. All methods of multicriteria decision analysis require information regarding the relative or absolute importance of each quality attribute. Most often there exist contradictions between the different decision quality attributes, in the sense that an architecture might be better than a second in terms of one quality attribute and worse in terms of a third one. This kind of decision problem with conflicting quality attributes is very common in business, engineering applications and goods evaluation etc. Usually an MCDA method aims at one of the following four goals.

i. Find the best architecture
ii. Group the architectures into well-defined classes
iii. Rank the architectures in order of total preference
iv. Describe how well each architecture meets all the quality attributes simultaneously.

Different types of analysis of these problems can be found in literature. One way is to use criteria aggregation technique that groups the decision quality attributes into two sets called the 'benefit' quality attributes and the 'cost' quality attributes. Then the architectures are evaluated using the information from these two aggregated indexes. Methods that belong to this category are the weighted sum model [6] and the weighted product model [7, 8]. When quality attributes are divided into 'benefit' and 'cost' groups, there are two approaches for ranking the architectures called the 'benefit to cost' and the 'benefit minus cost' approaches[8-11] and [1]. In [12] it has been established that when these two approaches are used on the same problem, they may lead to different rankings of the architectures. But the weighted product method (WPM) is invariant to these ranking disputes as is the case with multiplicative analytical hierarchy process (MAHP) [13, 14].

The rest of the paper is organized as follows. Section 2 describes the MCDA methods used and. Section 4 deals with a case study wherein all the MCDA methods are applied to the data obtained from a real-time stock monitoring system. The numerical calculations and the methods of ranking the architectures are also presented in this section. A brief conclusion is given in the last section.

2. DESCRIPTION OF THE MCDA METHODS USED

A typical MCDA problem consists of a number of m architectures to be evaluated in terms of n quality attributes. The architectures are denoted by A_i (i=1,2,…,m) and the quality attributes are denoted by C_j (j=1,2,…,n). Each quality attributes is associated with a preference coefficient, denoted by w_j (j=1,2,…,n). The more is the importance of the quality attribute, the higher will be the preference coefficient associated with it. Usually the preference coefficients are normalized so that they add up to one. Further when the architecture Ai is considered in terms of quality attribute C_j, the decision maker is assumed to know the corresponding performance value a_{ij} indicating how well architecture Ai meets quality attribute C_j. Different MCDA methods assume different requirements on these performance values. The above data can be summarized in a decision matrix given below.

Quality attributes

	(w_1	w_2	…	w_n)
				(Preference Coefficients)
	C_1	C_2	…	C_n
Archi.	------	------	------	-----
A_1	a_{11}	a_{12}	…	a_{1n}
A_2	a_{12}	a_{22}	…	a_{2n}
.	.		.	
.	.		.	
.	.			.
A_m	a_{m1}	a_{m2}	…	a_{mn}

In the above matrix the decision quality attributes have not been separated into benefit and cost groups. Next we present the MCDA methods considered here to analyze a decision matrix.

A. The Weighted Product Model (WPM)

This method uses multiplication to rank architectures instead of addition. Each architecture is compared with other in terms of a number of ratios, one for each quality attribute. Each ratio is raised to the power of the relative preference of the corresponding quality attribute. In order to compare two architectures A_k and A_l ($1 \le k, l \le m$) the formula

$$R\left(\frac{A_k}{A_l}\right) = \prod_{j=1}^{n}\left(\frac{a_{kj}}{a_{ij}}\right)^{w_j} \qquad (1)$$

is used. If $R\left(\frac{A_k}{A_l}\right) \ge 1$, then A_k is preferred than A_l.

Then the best architecture is the one that is better than or at least equal to all other architectures.

B. The multiplicative AHP

According to this method, the relative performance values aij and quality attributes preference coefficients wj are computed using formula (1). The details about this method are available in [15, 16].

In all the above methods we assume that in the MCDA problems considered, the quality attributes are divided into 'benefit' and 'cost' groups. As a convention the first k quality attributes C1, C2, ….., Ck are assumed to be the benefit quality attributes, while the remaining Ck+1, Ck+2,…., Cn are assumed to be the cost quality attributes , where 1 < k < n. Further, it is also assumed that quality attribute preference coefficients are normalized as follows:

$$\sum_{j=k+1}^{n} w_j = 1 \text{ and } \sum_{j=k+1}^{n} w_j = 1$$

Under the benefit to cost approach, the architectures are ranked according to their performance values Pi, which are computed according to the formula

$$P_{i,\,Ratio} = \frac{\sum_{j=1}^{k} a_{ij}\, w_j}{\sum_{j=k+1}^{n} a_{ij}\, w_j}, \text{ for i = 1,2,3,…, m.} \qquad (2)$$

Similarly, under the benefit minus cost approach, the performance values are computed by the formula.

$$P_{i,Diff} = \sum_{j=1}^{k} a_{ij}w_j - \sum_{j=k+1}^{n} a_{ij}w_j, \text{ for i=1,2,3,…,m.} \qquad (3)$$

In the case of WPM the expression for $R\left(\frac{A_k}{A_l}\right)$ and $D\left(\frac{A_k}{A_l}\right)$ under the benefit to cost and benefit minus cost approaches are given by

$$R\left(\frac{A_k}{A_l}\right) = \frac{\prod_{i=1}^{k} a_{k_i}^{w_i}}{\prod_{i=k+1}^{n} a_{k_i}^{w_i}} \times \frac{\prod_{j=k+1}^{n} a_{l_j}^{w_j}}{\prod_{j=1}^{k} a_{l_j}^{wj}} \qquad (4)$$

and

$$D\left(\frac{A_k}{A_l}\right) = \prod_{i=1}^{k} a_{k_i}^{w_i} \times \prod_{j=k+1}^{n} a_{l_j}^{w_j} - \prod_{i=k+1}^{n} a_{k_i}^{w_i} \times \prod_{i=1}^{k} a_{lj}^{w_j} \qquad (5)$$

respectively.

It has been proved in [17] that under the WPM or the MAHP method the two approaches, namely the benefit to cost and benefit minus cost give results consistent with each other.

3. CASE STUDY

A case study of real-time stock monitoring system [18] is taken and evaluated using the proposed quantitative evaluation method. The main objective of case study is to validate the proposed evaluation method. Input to the evaluation method is set of candidate architectures and its quality characteristics measured. Expected outputs are total satisfaction value for each candidate architecture. Architectures with highest total satisfaction value is selected for further phases of the software development life cycle.

3.1 Introduction

The primary goal of real-time stock monitoring system is to capture, analyze and broadcast stock events information in real-time. It is a soft real-time system where some of the events may miss their deadline without affecting the whole system behavior. The system is a real-time data provider for monitoring stocks of small and medium size stock exchanges for brokers and independent investors. An antenna (feed server), external to the system, provides the data (feed) to the data server. A feed contains the relevant information of a stock exchange transaction. Feeds are supposed to be reliable and available.

The clients, namely the brokers are distributed in different geographical locations and subscribed to the data server. When a change on the feed to which a client has ubscribed occurs, the feed is broadcasted to him/her by the data server, according to a strict time delay. The time delay will depend on the network structure used to send the information to the clients. The type of service offered depends on this delay. Internet facilities through commercial browsers are required for the system. The publisher/subscriber stores the client subscriptions, the actual values in the client subscription DB and the data server respectively.

3.2 Inputs for Evaluation

Inputs to the evaluation method are set of candidate architectures along with their quality characteristics measured. There are three candidate architectures considered namely publisher/subscriber pattern (A), repository pattern (B) and broadcast pattern (C).

3.2.1 Publisher/Subscriber Pattern (A)

In this type of candidate architecture, clients register their interest for stocks with the subscriber. The subscriber records the details of the clients in the database. A change in stock prices causes the publisher to notify these changes to the interested clients. Publisher/Subscriber pattern is shown in Figure 1.

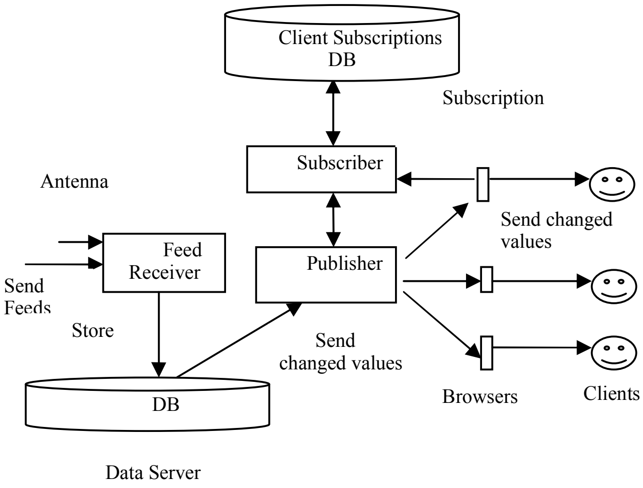

Figure 1. Publisher/Subscriber Pattern (A)

3.2.2 Repository Pattern (B)

In this type of candidate architecture, clients request the server for data about the stocks. Requests by clients may or may not be done periodically. Usage of proper queuing mechanism helps to avoid conflicts among requesting clients. A repository pattern is shown in Figure 2.

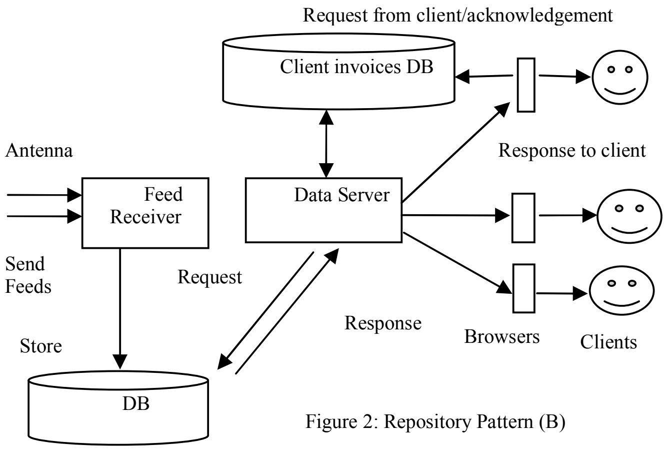

Figure 2: Repository Pattern (B)

3.2.3 Broadcast Pattern (C)

In this type of candidate architecture, a change in stock prices causes the server to broadcast these changes to their clients. Communication between clients and server is unidirectional. Broadcast pattern is shown in Figure 3.

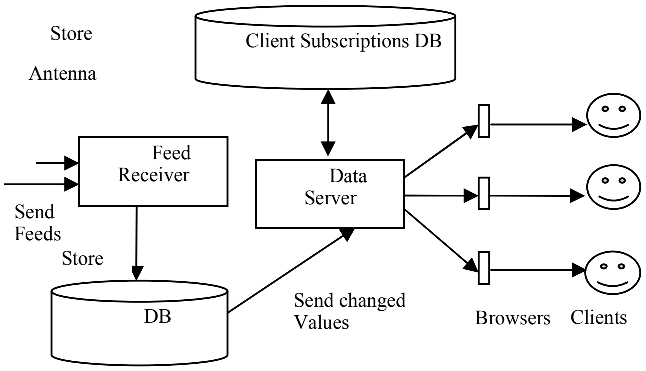

Figure 3: Broadcast Pattern(C)

3. 3 Quality Attributes

The real-time stock monitoring system is identified with eight quality attributes namely response time, learnability, maintainability, recoverability, reusability, cost, development time and team size. Architecture evaluation is carried out for these quality attributes for conformance with stakeholders' requirements. Measured values of candidate architectures are listed in Table 1.

TABLE 1 MEASURED VALUES

Architecture	Response Time (ms)[RT]	Learnability	Reusability (nos) [Reu	Maintainability (sec) [M]	Recoverability (secs) [Rec	Cost(Rs in Lacs) [C]	Team size (nos) [TS]	Development Time(wks) [DT]
A	10	5	1	200	20	8	20	60
B	20	8	5	25	10	4	10	30
C	12	3	1	200	5	6	5	20

3.3.1 Response Time

Response time is defined as the time required for completing a transaction. It is the sum of processing time, queuing time and data transfer time. It is measured in milliseconds (ms). In the case of repository pattern, the response time is high. This is due to the usage of queuing mechanism to handle simultaneous request from the clients. Moreover, each time to access information about stocks from the server, clients make a request to the server and in turn they receive acknowledgement. The requested data is sent by the server to the clients and it turns the server received acknowledgement from the clients. However, in the case of subscriber pattern data is sent to all clients periodically provided the interested clients register themselves once with the server.

3.3.2 Learnability

Learnability is defined as the time required by the user to understand the software and work with it. This includes the training period given for the users. It is measured in hours (hrs).

3.3.3 Maintainability

Maintainability is defined as the time taken to make successful modifications in the architecture. It is measured in hours (hrs). Maintainability mainly depends on the number of components and their interactions required for achieving the functionalities of the system. The publisher/subscriber has the highest number of components.

3.3.4 Recoverability

Recoverability is defined as the time taken to recover from failure state to working state. It is measured in seconds (secs). In repository structure, when the client side fails and recovers it can quickly reestablish the current status by requesting the server. However, in other two structures clients have to wait for the next periodic information.

3.3.5 Reusability

Reusability is defined as the number of components and connectors that can be reused. It is measured in number (nos).

3.3.6 Cost

It is the cost associated with developing the software product. It is measured in rupees (rs). The cost for repository system is low because it is built by using the existing components.

3.3.7 Development Time

This is the measure of time taken to build the software system. Its unit of measurement is weeks (wks). Since the repository pattern reuses existing components, its development time is less.

3.3.8 Team Size

It is defined as the number of technical persons required to build the project. Its unit of measurement is numbers (nos).

3.4 MCDA Methods :

A.WPM

To validate the proposed selection methods, experiments have been conducted using three architectures and eight quality attributes. Preferences of quality attributes were captured from twenty stakeholders. Here $w_1 \ldots w_8$ are the preference coefficients of quality attributes. Response time(RT), Learnability(L), Reusability(Reu) and Maintainability(M) are the benefit quality attributes. Recoverability (Rec), Cost(C), Team size(TS) and Development Time(DT) are the cost quality attributes. Preference coefficient values of benefit and cost quality attributes are added to 1 in MCDA methods.

W_1	W_2	W_3	W_4	|	W_5	W_6	W_7	W_8
0.2	0.1	0.4	0.3	|	0.2	0.4	0.25	0.15
RT	L	Reu	M	|	Rec	C	TS	DT

$$\begin{array}{c} A \\ \\ B \\ \\ C \end{array} \left(\begin{array}{cccc:cccc} \frac{10}{42} & \frac{5}{16} & \frac{1}{7} & \frac{200}{425} & \frac{20}{35} & \frac{8}{18} & \frac{20}{35} & \frac{60}{110} \\ \frac{20}{42} & \frac{8}{16} & \frac{5}{7} & \frac{25}{425} & \frac{10}{35} & \frac{4}{18} & \frac{10}{35} & \frac{30}{110} \\ \frac{12}{42} & \frac{3}{16} & \frac{1}{7} & \frac{200}{425} & \frac{5}{35} & \frac{6}{18} & \frac{5}{35} & \frac{20}{110} \end{array} \right)$$

Similar architectures, quality attributes and their preference coefficients were used in this Weighted Product Model for validating the proposed methods based on WPM technique.

Benefit to cost

$$R\left(\frac{A}{B}\right) = \frac{10^{0.2} \times 5^{0.1} \times 1^{0.4} \times 200^{0.3}}{20^{0.2} \times 8^{0.4} \times 20^{0.25} \times 60^{0.15}} \times$$

$$\frac{10^{0.2} \times 4^{0.4} \times 10^{0.25} \times 30^{0.15}}{20^{0.2} \times 8^{0.1} \times 5^{0.4} \times 25^{0.3}}$$

$$= 0.40708 < 1.$$

Similarly, $R\left(\frac{A}{C}\right) = 0.41102 < 1$; $R\left(\frac{B}{C}\right) = 1.00966 > 1$

From the result, B > A, C > A and B > C and hence the ranking is B > C > A.

Benefit minus cost

$$D\left(\frac{A}{B}\right) = \left(10^{0.2} \times 5^{0.1} \times 1^{0.4} \times 200^{0.3}\right) \times \left(10^{0.2} \times 4^{0.4} \times 10^{0.25} \times 30^{0.15}\right)$$

$$- \left(20^{0.2}\times 8^{0.4}\times 20^{0.25}\times 60^{0.15}\right)\times\left(20^{0.2}\times 8^{0.1}\times 5^{0.4}\times 25^{0.3}\right)$$

$$= -108.61382 < 0$$

Similarly, $D\left(\frac{A}{C}\right) = -57.17168 < 0$; $D\left(\frac{B}{C}\right) = 0.71067 > 0$

The ranking is B > A, C > A and B > C .

Therefore, B > C > A.

B. MAHP

In Multiplicative AHP, the above similar architectures, quality attributes and their preference coefficients were used for validating the proposed methods namely WPM and MAHP. The benefit to cost and benefit minus cost are calculated using the benefit and cost quality attributes.

Benefit to Cost

$$R\left(\frac{A}{B}\right) = \frac{\left(\frac{10}{20}\right)^{0.2}\times\left(\frac{5}{8}\right)^{0.1}\times\left(\frac{1}{5}\right)^{0.4}\times 1^{0.3}}{1^{0.2}\times 1^{0.4}\times 1^{0.25}\times 1^{0.15}} \times \frac{\left(\frac{10}{20}\right)^{0.2}\times\left(\frac{4}{8}\right)^{0.4}\times\left(\frac{10}{20}\right)^{0.25}\times\left(\frac{30}{60}\right)^{0.15}}{1^{0.2}\times 1^{0.1}\times 1^{0.4}\times\left(\frac{25}{200}\right)^{0.3}}$$

$$= 0.40707 < 1$$

Similarly, $R\left(\frac{A}{C}\right) = 0.41102 < 1$, $R\left(\frac{A}{C}\right) = 1.00968 > 1$

Thus B > A, C > A and B > C.

Therefore B > C > A.

Benefit minus cost

$$D\left(\frac{A}{B}\right) =$$

$$\left[\left(\left(\frac{10}{20}\right)^{0.2}\times\left(\frac{5}{8}\right)^{0.1}\times\left(\frac{1}{5}\right)^{0.4}\times 1^{0.3}\right)\times\left(\left(\frac{10}{20}\right)^{0.2}\times\left(\frac{4}{8}\right)^{0.4}\times\left(\frac{10}{20}\right)^{0.25}\times\left(\frac{30}{60}\right)^{0.15}\right)\right]$$

$$- \left[\left(1^{0.2}\times 1^{0.4}\times 1^{0.25}\times 1^{0.15}\right)\times\left(1^{0.2}\times 1^{0.1}\times 1^{0.4}\times\left(\frac{25}{200}\right)^{0.3}\right)\right]$$

$$= 0.21814 - 0.53588 = -0.31774 < 0$$

By similar calculations, $D\left(\frac{A}{C}\right) = -0.25325 < 0$;

$D\left(\frac{B}{C}\right) = 0.3174 > 0$

Thus B > A, C > A and B > C and

Hence B > C > A.

It has been verified that both the two methods under the two different approaches give consistent ranking of the architectures.

4. CONCLUSION

Thus by overcoming the disadvantages of the existing method [3, 4] an efficient, simple and accurate decision making process for selection of architectural structures have been devised. The validation of the proposed method has also been considered using a suitable case study. The advantage of the proposed method is that it also gives more importance to the stakeholders' preferences and views. This developed method of evaluation provides mathematical evidence on the choice of architecture.

5. ACKNOWLEDGMENTS

Our thanks to Dr. Thillai Govindan, Professor in Mathematics, Annamalai University for helping us to solve WPM and MAHP Method.

6. REFERENCES

[1] Kuwahara .Y and Y. Takeda, “A managerial approach to research and development cost-effectiveness evaluation,” *IEEE Trans. Eng. Manage.*, vol. 37, no. 2, pp. 134–138, May 1990.

[2] Svahnberg . M, C. Wohlin, L. Lundberg and M. Mattsson, “A Method for understanding Quality Attributes in Software Architecture Structures”, *Proceedings of the 14th International Conference on Software Engineering and Knowledge Engineering,* pp. 819-826, July 2002.

[3] Zayaraz . G, Dr. P. Thambidurai, “Quantitative Model for the Evaluation of Software Architectures”, *Journal of Software Quality Professional, American Society for Quality,* Vol.9, no.3, pp. 28-40, June 2007.

[4] Zayaraz .G and Dr. P. Thambidurai, “Software Architecture Selection Framework Based on Quality Attributes”, *Proceedings of the IEEE Conference INDICON*, pp. 67-170, Dec. 2005.

[5] Triantaphyllou .E, The impact of aggregating benefit and cost criteria in four MCDA methods. *IEEE Transactions on Engineering Management,* Vol. 52, No. 2, May 2005.

[6] Fishburn . P. C, “Additive utilities with incomplete product set: Applications to priorities and assignments,” *Oper. Res.*, pp. 12–26, 1967.

[7] Bridgeman . P.W, *Dimensionless Analysis*. NewHaven, CT: Yale Univ.Press, 1922.

[8] Miller . D. W. and M. K. Starr, *Executive Decisions and Operations Research.* Englewood Cliffs, NJ: Prentice-Hall, 1969.

[9] Saaty . T. L and L. G. Vargas, *Decision Making in Economic, Political, Social, and Technological Environments with the AHP*. Pittsburgh, PA: RWS, 1994.

[10] Bozeman .B and J. Melkers, *Evaluating R&D Impacts: Methods andPractice*. Norwell, MA: Kluwer, 1993.

[11] Augood . D. R, "A new approach to R&D evaluation," *IEEE Trans. Eng. Manage.*, vol. EM-22, no. 1, pp. 2–10, Feb. 1978.

[12] Wabalickis . R.N, "Justification of FMS with the analytic hierarchy process," *J. Manuf. Syst.*, vol. 17, pp. 175–182, 1988.

[13] Barzilai .J and F. A. Lootsma, "Power relations and group aggregation in the multiplicative AHP and SMART," in *Proc. 3rd Int. Symp. AHP*, Washington, DC, 1994, pp. 157–168.

[14] Lootsma .F.A, *Multi-Criteria Decision Analysis via Ratio and Difference Judgment*, ser. Applied Optimization Series. Dordrecht, The Netherlands: Kluwer, 1999, vol. 29.

[15] Triantaphyllou .E, *Multi-Criteria Decision Making: A Comparative Study*. Norwell, MA: Kluwer, 2000.

[16] Triantaphyllou .E, "Two new cases of rank reversals when the AHP and some of its additive variants are used that do not occur with the multiplicative AHP," *Multi-Criteria Decision Analysis*, vol. 10, pp. 11–25, May 2001. [Online]. Available: http://www.csc.lsu.edu/trianta.

[17] Saaty. T.L., *The Analytic Hierarchy Process*. NewYork: McGraw-Hill, 1980.

[18] Losavio . F, L. Chirinos, N. Levy and A. Ramdane, "Quality Characteristics of Software Architecture", *Journal of Object Technology,* Vol. 2, no.2, pp. 133-150, March 2003.