IMPLICATIONS OF IMPROVED-AHP AND FUZZY COMPREHENSIVE EVALUATION IN RANKING OF FACTORS INFLUENCING REUSABILITY

Deepika
Guru Jambheshwar University of Science and Technology
Hisar, Haryana
India
deepikagodara92@gmail.com

Om Prakash Sangwan
Guru Jambheshwar University of Science and Technology
Hisar, Haryana
India
sangwan0863@gmail.com

Jai Bhagwan
Guru Jambheshwar University of Science and Technology
Hisar, Haryana
India
jaihisar007@gmail.com

ABSTRACT

Software reusability focuses on previously written software specification, code and design. There are several advantages to reusability while developing different software applications. However, in order to effectively reuse software components, there are crucial elements influencing software reusability that must be considered. It is also necessary to consider issues that arise when software is reused. With the objective of identifying significant attributes impacting software reusability, a software reusability model focused on Improved-AHP and Fuzzy Comprehensive Evaluation is suggested. First, a comprehensive literature survey was done to determine various factors that affect software reusability. Second, a survey method conducted with experts and professionals working in the field of software engineering was performed to determine the most important reusability factors. Next, the selected reusability factors were ranked on the basis of improved-AHP. Finally, the Fuzzy Comprehensive Evaluation method was applied to evaluate reusability. The evaluation results indicated that 2% of the experts accept that the effect of these factors on reusability is very low, 1% believe that the effect is low, 11% believe it is medium, 42% believe the effect is high, and 44% believe it is very high. Therefore, reusability of software has an effect at a very high level from the chosen factors corresponding to the results obtained. This also supports our survey results which show that reusability is altered at a high level from the chosen factors. This study will assist software developers to clarify and tackle software reuse problems.

Keywords: reusability; Improved-AHP; Fuzzy Comprehensive Evaluation

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

1. Introduction

Software has made daily life easier and faster in the digital era. It has become an indispensable part of life in areas such as personal entertainment, society, automobiles, telecommunication, shopping, etc. (Ahmaro, bin Mohd Yusoff & Mohd Abualkishik, 2014). Software engineering is significant because for work in industry, or in any business of daily life, explicit software is required in order to accomplish specific tasks.

Currently, in order to combat the software crisis, the software industry is progressing toward extensive software reuse, where software can be easily developed with an existing or plagiaristic code. However, there is inadequate evidence on the adaption of reusability in industries, including whether it can be beneficial for the long term. To date, there is no clear framework which elucidates software reusability (Crnkovic, Chaudron & Larsson, 2006). Development in the field of software reusability has greatly increased in the past decades but is still facing many issues in various reuse paradigms. Due to this, practitioners hesitate to adopt reuse. The main difficulty lies in the identification of suitable evidence. The research on software reuse is scattered instead of concentrated and consecutive (Frakes & Kyo Kang, 2005).

Increasing the percentage of software reuse is the most efficient method to boost or speed up a software product's productivity. Increased size, complexity, and insufficient utilization of technology are the primary causes of rising software development costs. Reuse lowers development time and costs, boosts productivity, and increases reliability. The focus of contemporary software management is concerned with reducing development time and expense. OS interfaces and reusable software libraries can be used to accomplish these goals. Software crises are frequently caused because of a failure to control time, cost, productivity, and dependability. Reusing software involves more than just using the original source code; it also involves a number of other details.

Some examples of software reuse include the following:

- Japanese software industry is built on 85% reuse factor (Standish, 1984)
- Reusability is the main reason for wide acceptance of UNIX (Musa, 1985)
- Majority of business applications include 60% reusable design and code (Lanergan & Grasso, 1984)

When considering the problem of software reuse, a number of factors that impact software reuse must be taken into account in order to reduce manufacturing time and cost. Therefore, choosing the proper variables or characteristics is crucial to the effective reuse of software products. However, in recent years, researchers have randomly chosen different factors without following a specific method, making it impossible to determine whether or not the attributes chosen by the researchers were appropriate for reusability evaluation. Therefore, this study prioritized the identification and ranking of critical elements influencing reusability. This research developed a software reusability model based on Improved-AHP (IAHP) and Fuzzy Comprehensive Evaluation (FCE) that takes advantage of both Analytic Hierarchy Process (AHP) theory and fuzzy logic, i.e., human logic and evaluation criteria which center on relative importance of factors.

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

2. Literature review

This section explains the research that has been done related to reusability, IAHP and FCE. Fuzzy theory is employed to describe data or knowledge that is unclear, probabilistic, and imprecise. Therefore, Fuzzy-AHP (FAHP) is better at dealing with vagueness and imprecision in decision making than regular AHP. FAHP enables decision makers to express their preferences more freely by utilizing fuzzy numbers rather than precise values. In order to determine the application of FAHP to software engineering problems and software design solutions, we collected and analyzed publications from 2011 to 2024.

Many researchers have proposed attributes and models for measuring reusability using various soft computing or machine learning techniques. Software engineering practice can be improved with the use of systematic development to increase reusability of software components. According to Makni et al. (2014) and Karunanithi and Bieman (1993), given the significance of software reuse and metrics of reuse in the field of software engineering processes, as well as for the stakeholders, we should not tackle the same problems again and over again. This study reveals the benefits of software reuse and suggests how software development stakeholders can harness these benefits. Different stakeholders use different metrics to ascertain the information they need.

IAHP is a modified version of the AHP. Traditional AHP uses a nine-point scale whereas IAHP uses a three-point scale. FAHP is considered the best method for testing the fuzziness of decision makers, and is an extension of traditional AHP (Kabir & Hasin, 2013). In fuzzy environments which require multiple criteria decision making, FAHP proves to be quite useful. Wang and Chin (2011) proposed a logarithmic fuzzy preference programming methodology which has potential application in FAHP.

FAHP has numerous applications including the selection of schools, planning of a transportation system, predicting grindability of granite, determining project performance, and determining quality of gemstones (Kusumawardani et al., 2016; Putra et al., 2018; Zhang et al., 2017). It also can be applied in various real-world problems, especially in the engineering of smart vehicles and making decisions in companies. Cebi and Karal (2017) used FAHP for the evaluation of students' projects.

In order to quantify the qualitative human assessment fuzzy theory is the best candidate, the FAHP was used for autonomic computation which is an artificial intelligence-based approach (Dehraj & Sharma, 2020). IAHP and the fuzzy theory approach have been used by Peisheng et al. (2020) for research on the assessment of risk of an information security system. This method does not require a consistency check as the comparison matrix is formed from expert reviews which reduces computing time. The FCE technique also finds numerous applications even without combining it with the AHP. It can be used in real world science projects like air quality assessment (Zhao et al., 2010) and in the evaluation of financial control of any enterprise system and enhancing competitiveness of businesses (Shao, 2009).

Alzahrani and Khan (2024) evaluated a safe software design and decision-making model utilizing an ANN-FAHP hybrid decision making model. The results of the study will help academic researchers and professionals in real-life industry create new methods for projects involving ubiquitous computing.

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

A Multicriteria Decision Making (MCDM) approach was presented by Thapar and Sarangal (2020) to rank software components according to their reusability. Reusability has also been measured using the hybrid FAHP-metrics technique, which aids in the selection of highly reusable components and verifies the approach's suitability for payment gateways. Singh et al. (2014) proposed a quantitative reusability estimation model using a multi-criterion decision making approach for aspect-oriented software. Validation of the model was done using the AHP and it was cross validated using the FAHP. The AHP is mainly used in multicriteria decision making problems in real situations.

Kumar et al. (2018) suggested using the FAHP for prioritizing attributes related to human trust, dependability, and trustworthiness. This technique aids developers in enhancing software security over an extended length of time. Prioritizing long-lasting security features is essential to guarantee the creation of safe and enduring software. When the weights and ranks of durable security attributes are assessed using the FAHP technique, the findings indicate that trustworthiness is the most crucial factor, followed by dependability and human trust. Setting trustworthiness as the top priority for security features can increase software's lifetime security and boost user satisfaction.

Kumar et al. (2020) used a fuzzy decision-making framework to assess the effectiveness of malware analysis techniques for protecting web applications. The new integrated fuzzy AHP-TOPSIS method can help with the thorough evaluation of options based on multiple parameters. FAHP is a potent analytical tool that can be used to assess complex problems by assigning specific graded goal rates to each one.

The primary objective of the suggested approach in Kumar et al. (2021) was to assess web applications' sustainable security through the use of the FAHP. The process was comprised of many phases, such as issue identification and analysis, hierarchical structure establishment, fuzzy pairwise matrix definition and setup based on expert choices, weight calculation of the fuzzified values, hierarchical connection, defuzzification, and ordering and placement of the defuzzified parameters. Fuzzy values are represented by the Triangular Fuzzy Number (TFN), which has lower, middle, and higher values. With the help of the suggested framework, web developers will be able to maximize sustainable security when creating web apps, which will save time and money while also enhancing the applications' overall sustainability and security.

Furthermore, the literature shows that the FAHP aids decision makers in evaluating the reusability of software components and prioritizing solutions according to their relative value. FAHP is the best choice since it makes it simple to prioritize the elements that impact reusability and facilitates software reuse. Software engineering decision-making models and software design both commonly employ FAHP.

3. Research methodology

Figure 1 illustrates the research methodology used to identify the key characteristics that influence reusability and the implementation procedures, such as IAHP and fuzzy comprehensive assessment that are utilized to validate the chosen reusability attributes through expert review and a literature survey.

We identified the key elements associated with reusability estimation in numerous research investigations. Then, we chose the five most crucial elements influencing reusability based on the dependencies provided. We conducted an expert survey to validate the parameters that were chosen, and used Improved-FAHP to rank the aspects that impact reusability.

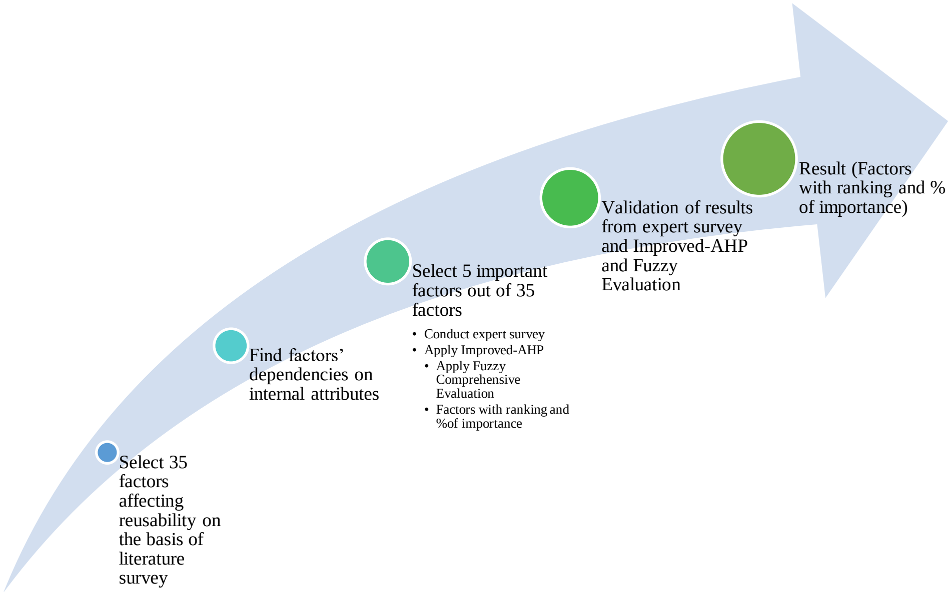


Figure 1 Research methodology

4. Fundamental concept of software reusability

4.1 Reusability concept

Reuse of software, which is the expected resuse potential of software, has been practiced since the beginning of programming. It also has an optimistic influence on the quality of software (Ahmaro, bin Mohd Yusoff & Mohd Abualkishik, 2014). The quality of software plays a crucial role in the construction of novel software and is largely affected by internal characteristics like reusability, maintainability and understandability (Kumar et al., 2018). Reusability is the leading characteristic of software quality.

In order to tackle the problem of the software crisis, software reusability is a crucial component. Reusability is the process in which computer programs are updated and implemented using existing assets (Kaur & Kaushal, 2018). There are various approaches to software reusability that include component-based development, application frameworks, design patterns, aspect-oriented software development, etc. The benefits of software reusability include increased dependability, productivity and effectiveness, accelerated development, reduced operational costs, etc. In the software lifecycle, there are various levels of software reusability which include design reuse, code reuse, application reuse and specification reuse. Table 1 summarizes the definitions of software reusability.

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

Table 1
Definitions of software reusability

Software reusability definition	**References**
An applicable way of resolving software crisis and refining quality and production.	Zhang et al. (2017)
The use of software assets for developing new applications from all phases of software development.	Kim and Stohr (1998)
The degree to which we can adapt a software module beyond one software system or computer platform.	Fazal-e-Amin et al. (2010)
A tool to reduce progress time and cost.	Sharma et al. (2009)
The use of existing software to construct new software.	Jalender et al. (2011)

There are various factors that affect software reusability, which are explained in detail in next section.

4.2 Factors affecting software reusability

It is clear from the literature review that various models have been defined to group internal characteristics of reusability and various researchers have proposed frameworks for measuring reusability. Sometimes, considering a large number of parameters results in outperformance of models (Polat & Nur Alpaslan, 2023), which results in a greater investment of time and decreased performance. Therefore, it becomes important to select important factors affecting reusability. An extensive literature survey was conducted to determine dependencies, which are presented in Table 2 and Figures 2-5. Based on these findings, we performed an expert evaluation to check the importance of the factors selected.

Table 2 gives the summary of factors used by various researchers for assessing reusability. These factors were found in 22 articles from 1992-2024 where a total of 35 attributes were identified which have been used by researchers for reusability estimation.

Table 2
Attributes considered by researchers for estimation of reusability

	Reusability attributes																																			
S.No.	Authors	Understandability	Maintainability	Adaptability	Portability	Modularity	Flexibility	Cohesion	Coupling	Complexity	Documentation	Customizability	Reliability	Completeness	Scope Coverage	Variability	Learn-ability	Efficiency	SoC	Availability	Operability	Reuse	Quality	Price	Inheritance	Stability	Usage History	Helpfulness	Attractiveness	Usability Compliance	Security	Response Time	Statelessness	Testability	Generality	Correctness
1	Fazal-a-Amin et al. (2010)	×	×		×		×								×	×																				
2	Papamichail et al. (2018)							×	×	×	×														×											
3	Rodriguez et al. (2011)				×			×		×																										
4	Singh et al. (2018)								×	×																					×	×	×			
5	Singh et al. (2014)		×	×		×	×			×																										
6	Lounis et al. (2004)							×	×	×															×											
7	Kumar et al. (2013)	×	×		×					×		×																								
8	Mehboob et al. (2021)		×	×	×	×		×		×	×	×	×	×				×		×		×	×	×										×	×	×
9	Crnkovic et al. (2006)		×	×						×	×									×		×	×	×												
10	Paschali et al. (2017)		×	×						×	×									×		×	×	×												
11	Kaur et al. (2017)	×	×	×		×																														
12	Sant’anna et al. (2003)	×					×																													

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

13	Singh and Tomar (2016)									×	×	×	×																							
14	Ahmaro et al. (2014)	×	×		×		×				×				×	×										×	×									
15	Kaur and Kaushal [22]							×	×	×									×																	
16	Fazal-e-Amin et al. (2011)	×	×		×		×								×	×																				
17	Chaudhary and Chatterjee (2013)	×	×	×		×																														
18	Singh et al. (2014)	×	×	×		×											×				×															
19	Salamon et al. (1994)	×	×	×	×	×							×	×			×	×										×								
20	ISO (2024)	×															×				×								×	×						
21	Singh (2011)	×		×						×	×																									
22	Kumar et al. (2013)	×			×					×		×																								

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

Table 2 shows that the following ten attributes are the most frequently used and given importance in different research papers: Understandability, Maintainability, Adaptability, Portability, Coupling, Cohesion, Flexibility, Complexity, Customizability (Changeability), and Scope Coverage. Figures 2-5 display the interdependencies between various attributes based on the literature survey:

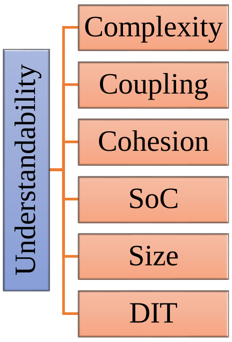


Figure 2 Relationship of understandability with internal attributes (Kaur et al., 2017; Sant'Anna et al., 2003)

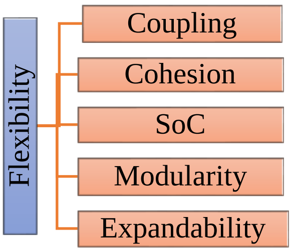


Figure 3 Relationship of flexibility with internal attributes (Salomon & Wallace, 1994)

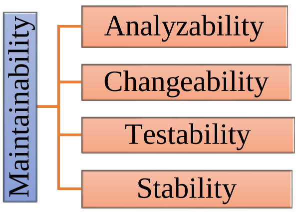


Figure 4 Relationship of maintainability with internal attributes (ISO, 2024)

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

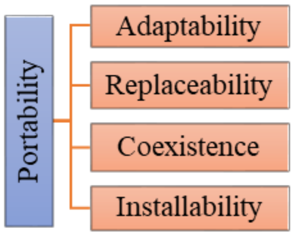


Figure 5 Relationship of portability with internal attributes (ISO, 2024)

Table 2 indicates that nearly all of the critical attributes that are most important and chosen by the majority of research papers for reusability estimation are covered by the interdependencies shown in Figures 2-5. However, since scope coverage is not related to any of these four attributes and is utilized by the majority of researchers, it must be added as a separate attribute of importance for reusability estimation. The 35 reusability factors were reduced to five based on their usage importance as shown in Table 2 and the interdependences displayed in Figures 2-5. Furthermore, the five most important attributes are understandability, maintainability, flexibility, portability, and scope coverage. Table 3 provides an explanation of these properties in relation to reusability.

Table 3
Attributes definition according to reusability

Attributes	**Definition according to reusability**
Understandability	If a developer cannot study something, he cannot gain understanding. Without understanding he cannot use the thing to be studied. It is difficult to maintain a system that in not understood, and it is difficult to change and reuse a system that cannot be understood as a whole.
Maintainability	The capability to change and update a system when required. It is a major issue as updating and removing bugs in a new system is difficult for developers.
Flexibility	The ability to be usable in multiple configurations. If a system is flexible and able to be used in multiple contexts, then it is easily reusable.
Portability	A component is easily portable if it contains all the necessary information and requires little interaction with other modules. An easily portable component is easy to reuse.
Scope Coverage	The greater the number of features a component has, the further it can be extended. A developer will prefer a component that covers most of the

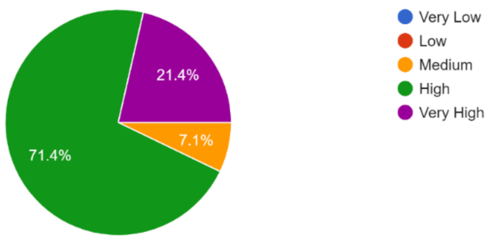
ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

Attributes	**Definition according to reusability**
	functionality of an application which makes reusability of the application easier in further work.

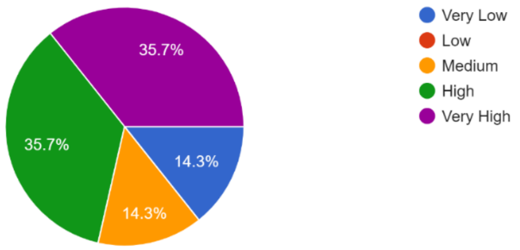
Reusability is affected by five important characteristics that include understandability, maintainability, flexibility, portability and scope coverage. Furthermore, the importance of these selected attributes was justified by conducting an expert survey using Google Forms to determine the significance of these elements and the degree to which they impact reusability. The results of the survey also provide a comparison of these variables.

5. Survey results

Survey research, as used in software engineering, is a methodical inquiry carried out with a number of methods, including online or offline methods. We decided to use Google Forms for our predictive survey study. A panel of 15 software engineering specialists were chosen for the expert evaluation team. Figure 6 shows how many experts believe that a given attribute influences software reusability and to what extent.

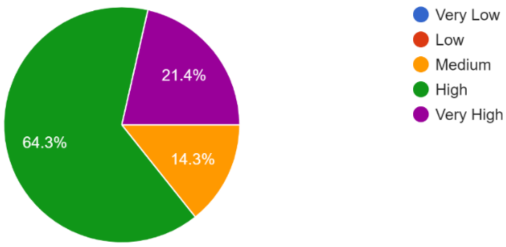


(a) Understanability

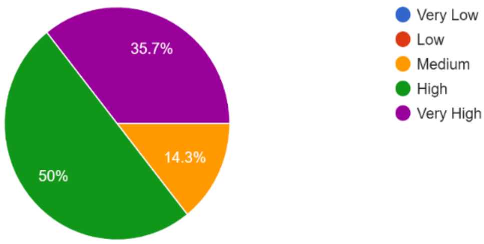


(b) Maintainability

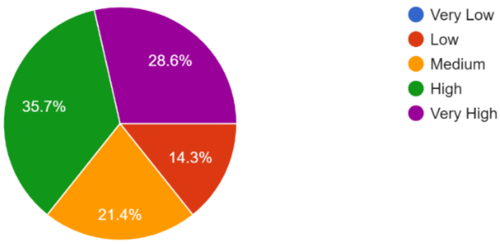
ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256



(c) Flexibility



(d) Portability



(e) Scope Coverage

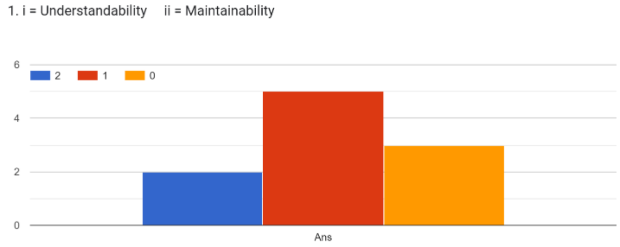
Figure 6 Effect of attributes on reusability as determined by experts (a) understandability (b) maintainability (c) flexibility (d) portability (e) scope coverage

According to Figure 6, 71.4% of experts think that the effect of understandability on reusability is high and 35.7% think that portability effects reusability at very high level. Figure 6 also shows at what level the attributes affect reusability and the percentage of experts that think this.

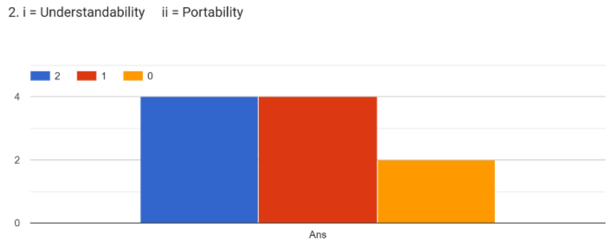
Next, we pairwise compared these attributes to determine which is more important in order to create the comparison matrix for IAHP. Figure 7a-e shows the result of the comparison of factors affecting reusability according to the expert evaluation from Google Forms. Here, 2 indicates that factor (i) is more important than factor (j); 1 indicates that factor (i) is equally important to factor (j); and 0 indicates that factor (i) is less important than factor (j).

According to Figure 7a, 2 represents that understandability is more important than maintainability; 1 represents that understandability is as important as maintainability; and 0 represents that understandability is less important than maintainability.

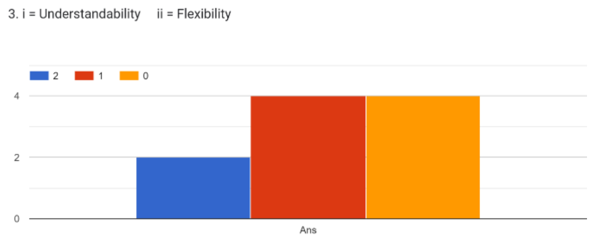
Likewise, all other parts of Figure 7 represent the importance of one factor over another as evaluated by experts.



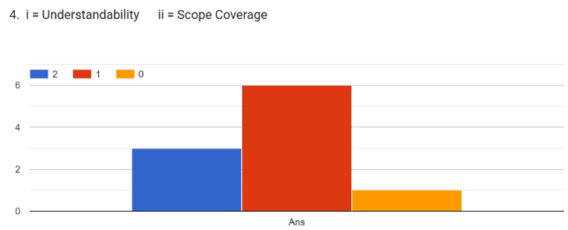
(a) Comparison of understandability and maintainability



(b) Comparison of understandability and portability

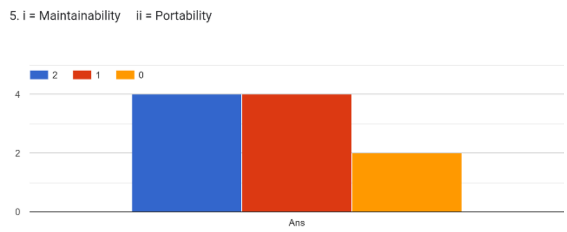


(c) Comparison of understandability and flexibility



(d) Comparison of understandability and scope coverage

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256



(e) Comparison of maintainability and portability

Figure 7a - e Comparison of factors affecting reusability as determined by experts

6. Proposed approaches

6.1 Traditional AHP

The AHP method was introduced by Dr. Thomas L. Saaty in the 1970's. It is a Multicriteria Decision Making (MCDM) method. It includes both objective evaluation and subjective human judgement which has been applied for a decision support system. It can be a qualitative or quantitative decision-making method. The AHP has been developed to include FAHP, IAHP, Gray-AHP and various other implementations according to the scope and situation of the study (Saaty, 1985; 1988). Karpak (2017) addressed the AHP paradigm of decision making and her perspective on Thomas Saaty's book in an article titled "Reflections: Mathematical Principles of Decision Making" where the importance of doing a sensitivity analysis to verify the validity of the priorities established for actual choices is discussed and a summary of each chapter of Saaty's book is given.

Two key features of the AHP technique are also the main sources of the method's weakness, i.e.,inconsistency of the comparison matrix (CM) and the complex nature of CM pairwise methodology. When there are three or four elements in a set, the AHP method can yield different weights for the same set of elements. This is so that different weights can be assigned to the same set of elements in different scenarios of judgment using pairwise comparison, which is made possible by the AHP methodology. However, the sorting and ranking methodology used by the IAHP approach yields consistent weights for the same set of elements. When there are three or four elements, the disparity in weight distribution between the AHP and IAHP is more noticeable. This discrepancy can be attributed to the methods used to create the CMs in the AHP and IAHP (Karaboga & Kaya, 2019).

6.2 Improved AHP

The IAHP is a three-scale method in contrast to the traditional AHP where a nine scale method is used. The AHP requires a consistency check of the comparison matrix, but the IAHP does not because an optimal transfer matrix is used which is based on expert judgements (see Figure 7). The AHP requires repeated adjustments of the comparison matrix but there is no need for adjustments in the IAHP (Cebi & Karal, 2017). The IAHP-FCE method not only analyzes and compares the membership degree of each comment grade of the ultimate evaluation index, but also analyzes and evaluates the expected compliance degree of each evaluation index to arrive at the overall situation of the evaluation object (Jang, 1993).

The stepwise application of the IAHP is as follows:

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

Step 1: Based on the evaluation factors, structure the hierarchy of the criteria. First, state the objective of the problem; then, list factors that affect the objective. A hierarchy of the criteria for the problem is structured and at each level in the hierarchy only seven elements are desirable.

Step 2: Construct a pairwise comparison matrix using Figure 7a-e:

Construct a comparison matrix based on attributes significance as governed by the experts using a three-scale method. The comparison matrix C is given:

$$C = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ c_{n1} & c_{n2} & \cdots & c_{mn} \end{pmatrix} \quad (1)$$

Where $c_{ij} = \begin{cases} \text{2-The factor i is of greater importance than the factor j} \\ \text{1-The factor i is of equal importance to the factor j} \\ \text{0-The factor i is of lesser importance than the factor j} \end{cases}$

And we set c_{ij}=1.

According to our work c_1 = understandability, c_2 = maintainability, c_3 = portability, c_4 = flexibility, c_5 = scope coverage.

According to Equation 1, i=5, j=5, So

$$C = \begin{pmatrix} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} \\ c_{21} & c_{22} & c_{23} & c_{24} & c_{25} \\ c_{31} & c_{32} & c_{33} & c_{34} & c_{35} \\ c_{41} & c_{42} & c_{43} & c_{44} & c_{45} \\ c_{51} & c_{52} & c_{53} & c_{54} & c_{55} \end{pmatrix} \quad (2)$$

Step 3: Calculate the ranking index r_i:

$$r_i = \sum_{j=1}^{n} C_{ij} \quad (3)$$

where i = 5, j = 5 $r_i = r_1, r_2, r_3, r_4, r_5$.

Step 4: Construct the judgement matrix B:

$$b_{ij} = \begin{cases} \frac{r_i - r_j}{r_{max} - r_{min}}(k_m - 1) + 1 & (r_i \geq r_j), \\ \left[\frac{r_j - r_i}{r_{max} - r_{min}}(k_m - 1) + 1\right]^{-1} & (r_i < r_j) \end{cases} \quad (4)$$

$$and\ r_{max} = max\{r_i\}, r_{min} = min\{r_i\}, k_m = {r_{max}}/{r_{min}}.$$

Step 5: Solve the above step 4 judgement matrix B to get transfer matrix C, and its element c_{ij} is:

$$c_{ij} = \lg b_{ij}, \quad (5)$$

Where i, j = 1, 2, 3, 4, 5.

Step 6: Solve the above-mentioned transfer matrix C in order to obtain the optimal transfer matrix D, elements d_{ij} (i, j =1,2,3,4,5) of matrix are:

$$d_{ij} = {}^{1}\!/_{n} \sum_{k=1}^{n} (c_{ik} - c_{jk}), \qquad (6)$$

Where n = 5 represents the total row numbers or column numbers in comparison matrix and k = 5 represents the number of columns in the comparison matrix.

Step 7: Construct quasi optimal matrix B' of the judgement matrix B. The elements b'_{ij} of matrix B' are:

$$b'_{ij} = 10^{d_{ij}}, \qquad (7)$$

Step 8: From matrix B' calculate the eigen vector. The product of elements of B' is given as:

$$M_i = \prod_{j=1}^{n} b'_{ij} ,$$

Calculation of the square root:

$$\overline{W_i} = \sqrt[n]{M_i} ,$$

Normalization of the vector $\overline{W} = \overline{(W_1, W_2, \dots, W_n)}^{r}$, n = 5:

$$W_i = \overline{W_i} \setminus \sum_{i=1}^{5} \overline{W_i} , \qquad (8)$$

And the result of hierarchical ranking is provided as follows: W= $(W_1, W_2 \dots, W_5)^{T}$.

6.3 Fuzzy Comprehensive Evaluation

The FCE method is based on fuzzy mathematics. Fuzzy mathematics is used for applications in various real-world problems as it gives practical and reliable results based on human judgement and uncertainty. It deals with both quantitative and qualitative analysis which helps in modeling uncertainty and lack of precision. This method not only captures human thinking but also focuses on the relative importance of evaluation criteria (Wang & Chin, 2011).

FCE is conducted as follows:

Step 1: Determine factor set of evaluation U= $\{u_1, u_2, u_3, ..., u_n\}$. These evaluation factors or indicators represent the nature of features of the first level in the evaluation index method.

Step 2: Evaluate fuzzy comment set V = $\{v_1, v_2, v_3, \dots, v_m\}$. This represents the evaluation results of the experts.

Step 3: Evaluate single factor calculation matrix R from U to V. This matrix is constructed from the group of expert's results. The result of the evaluation of each factor is calculated separately, i.e., $R_1, R_2, R_3, \dots R_n$, where n = 5.

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

Where R_i= (r_{i1}, r_{i2}, r_{i3}, …, r_{im}), (i= 1, 2, …, n), $R_i \in \mu(V)$ and n = 5, m = 5.

The evaluation result meets the normalized condition, i.e., the sum of the weight of each vector is 1, that is, for every i, there is: $r_{i1}+r_{i2}+r_{i3}+\ldots+r_{i5}=1$.

That is:

$$R = \left(r_{ij}\right)_{mn} = \begin{matrix} r_{11} & r_{12} & \cdots & r_{15} \\ r_{21} & r_{22} & \cdots & r_{25} \\ \cdots & \cdots & \cdots & \cdots \\ r_{51} & r_{52} & \cdots & r_{55} \end{matrix} \qquad (9)$$

Step 4: Determine the weight of the index. Weight means the proportion of weight allotted to each factor based on the importance in the evaluation index as obtained by using IAHP in Equation 8. This weight is represented by W which is a fuzzy set containing factors recorded as W= $(W_1, W_2 \ldots, W_n)^T$ which is required to meet normalized conditions.

Where $\sum_{i=1}^{n} W_i = 1, 0 < W_i < 1$, n = 5.

Step 5: Find the result of evaluation B which is product of weight of index and evaluation matrix R represented as:

$$B = W \cdot R = \{b_1, b_2, b_3, \ldots b_m\} \qquad (10)$$

Where m = 5

7. Empirical analysis

The research goal is to identify significant variables influencing reusability. Software reusability is calculated using the assessment result and model described in this research, which also ranks the elements influencing reusability. Five variables make up the criterion level, according to a thorough review of the literature and expert assessment (understandability, maintainability, flexibility, portability, and scope coverage).

7.1 Establishing the index system

The hierarchical model is shown in Table 4. The target layer (A) is reusability of software. The criteria layer (B) is comprised of five parts as follows: understandability, maintainability, flexibility, portability, and scope coverage. The third layer represents the weight obtained using IAHP.

7.2 Stepwise calculation of weights using IAHP

According to the three-scale method and using Equation 2, the comparison matrix determined by experts is

$$C = \begin{pmatrix} 1 & 1 & 2 & 1 & 1 \\ 1 & 1 & 1 & 1 & 2 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 2 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \end{pmatrix}$$

The ranking index r_i (i = 1,2,3,4,5) according to Equation 3 is given as:

$$(r_1, r_2, r_3, r_4, r_5) = (6,6,3,6,4)$$

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

According to Equation 4, judgement matrix B is constructed; the transfer matrix C is obtained using Equation 5:

$$C = \begin{pmatrix} 0 & 0 & 0.301 & 0 & 0.221 \\ 0 & 0 & 0.301 & 0 & 0.221 \\ 0 & -0.301 & 0 & -0.301 & -0.124 \\ 0 & 0 & 0.301 & 0 & 0.221 \\ -0.221 & -0.221 & 0.124 & -0.221 & 0 \end{pmatrix}$$

The optimal transfer matrix D is obtained using Equation 6:

$$D = \begin{pmatrix} 0 & 0.120 & 0.249 & 0 & 0.192 \\ 0 & 0 & 0.249 & 0 & 0.212 \\ -0.249 & -0.249 & 0 & -0.249 & -0.037 \\ 0 & 0 & 0.249 & 0 & 0.212 \\ -0.212 & -0.212 & 0 & -0.212 & 0 \end{pmatrix}$$

Construction of a quasi-optimal consistent matrix B' using Equation 7 is:

$$B' = \begin{pmatrix} 1 & 1.318 & 1.774 & 1 & 1.555 \\ 1 & 1 & 1.774 & 1 & 1.629 \\ 0.563 & 0.563 & 1 & 0.563 & 0.917 \\ 1 & 1 & 1.774 & 1 & 1.629 \\ 0.613 & 0.613 & 1 & 0.613 & 1 \end{pmatrix}$$

Calculate eigen vector of matrix B':

$M_1 = 3.635$, $M_2 = 2.889$, $M_3 = 0.613$, $M_4 = 2.889$, $M_5 = 0.230$

Normalize the vector and the weight obtained according to Equation 8:

$$W = (W_1, W_2, W_3, W_4, W_5) = (0.290, 0.258, 0.119, 0.258, 0.075)$$

Therefore, using IAHP, the weights of the factors that determine reusability are understandability at 0.290, maintainability at 0.258, flexibility at 0.119, portability at 0.258, and scope coverage at 0.075. Table 4 displays the reusability evaluation index system:

Table 4
Reusability evaluation index system and weights obtained using IAHP

Target Layer	**Criteria Layer**	**Weights**
Reusability	Understandability	0.290
	Maintainability	0.258
	Flexibility	0.119
	Portability	0.258
	Scope Coverage	0.075

In Table 4, reusability is the target layer. The criterion layer takes into account the variables that impact software reusability such as scope coverage, understandability, maintainability, flexibility, and portability. The IAHP was used to produce the weights displayed in the final layer.

7.3 Evaluation of reusability based on Fuzzy Comprehensive Evaluation

According to the steps described in Section 5.3, the evaluation of reusability based on FCE is as follows:

Determine factor set of reusability U = $\{u_1, u_2, u_3, u_4, u_5\}$ = {Understandability, Maintainability, Flexibility, Portability, Scope Coverage} according to step 1 of section 5.3.

Comment set V = $\{v_1, v_2, v_3, v_4, v_5\}$ = {Very Low, Low, Medium, High, Very High} according to step 2.

Single factor evaluation matrix R from U to V according to Equation 3 is given as follows:

$$R = \begin{pmatrix} 0 & 0 & 0.1 & 0.2 & 0.7 \\ 0.1 & 0 & 0.1 & 0.4 & 0.4 \\ 0 & 0 & 0.1 & 0.4 & 0.5 \\ 0 & 0 & 0.1 & 0.7 & 0.2 \\ 0 & 0.1 & 0.1 & 0.4 & 0.3 \end{pmatrix}$$

The determination of the weight of index of reusability is obtained using the IAHP established above. The calculation result is shown below:

Factors	Understandability	Maintainability	Flexibility	Portability	Scope Coverage
Weight	0.290	0.258	0.119	0.258	0.075

Weight meets normalized condition, i.e., $\sum W = 1$

The weight set of every factor can be composed in the form of fuzzy vector:
A = (0.290, 0.258, 0.119, 0.258, 0.075)

The result of evaluation B is a product of the weight of index A and evaluation matrix R:
B = A · R

$$= (0.290, 0.258, 0.119, 0.258, 0.075) \cdot \begin{pmatrix} 0 & 0 & 0.1 & 0.2 & 0.7 \\ 0.1 & 0 & 0.1 & 0.4 & 0.4 \\ 0 & 0 & 0.1 & 0.4 & 0.5 \\ 0 & 0 & 0.1 & 0.7 & 0.2 \\ 0 & 0.1 & 0.1 & 0.4 & 0.3 \end{pmatrix}$$

= (0.02, 0.01, 0.11, 0.42, 0.44)

The outcomes of the FCE indicate the degree to which specified factors can influence reusability, as expressed as a percentage (%) among experts. Therefore, based on the results, the effects these attributes are having on reusability are shown by the following: 0.02 (2%) represents extremely low, 0.01 (1%) describes low, 0.11 (11%) depicts medium, 0.42 (42%) depicts high, and 0.44 (44%) depicts very high.

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

7.4 Results and discussion

As demonstrated in Table 5, 2% of experts think these characteristics have a **very low** influence on reusability, 1% think they have a **low** effect, 11% think they have a **medium** effect, 42% think they have a **high** effect, and 44% think they have a **very high** effect. The conclusion should be at the evaluation level "Very High," which corresponds to the highest value in set B. Therefore, based on the results, the attributes listed have a very significant impact on a software's reusability. This is also consistent with the results of our survey. According to survey data, 71.4% of experts think that understandability has a significant impact on reusability and 50% of the experts believe that portability has a significant impact on reusability. Experts believe that reusability is highly impacted by flexibility in 64.3% of cases. According to 35.7% of experts, maintainability and scope coverage have a significant impact on reusability at a high level. The average of the expert evaluation's results is 51.42%, which has a significant impact on reusability.

Table 5
Estimated values of Fuzzy Comprehensive Evaluation

Reusability Level	**Fuzzy Comprehensive Evaluation Result**
Very Low	2%
Low	1%
Medium	11%
High	42%
Very High	44%

Therefore, it can be inferred that the considered characteristics have a significant impact on software reusability and should be taken into account when estimating software reusability.

8. Conclusion and future work

A reusability valuation model based on IAHP and FCE was implemented in this study. Based on an extensive literature survey and expert evaluation, five of the most important factors that affect reusability were collected and then IAHP was used to rank the reusability factors including understandability, maintainability, flexibility, portability, and scope coverage. The comparison matrix in IAHP was determined by experts by grading different reusability factors. The expert evaluation also provides the basis for further investigations. The weight index of each factor in the reusability evaluation index system was calculated using IAHP. This method does not require any consistency check as it is a three-scale method which replaces the traditional nine-scale method and helps reduce computing time. FCE was used to further calculate the factors' importance. The result of the experimental investigation shows that the suggested methods in this research are time-saving, effective and accurate. IAHP and FCE are used in many decision-making scenarios. The selection criteria for reusability estimation were determined by a thorough review of the literature, expert assessment, IAHP, and FCE because concentrating on these elements will aid in handling the software crisis, and will assist developers in addressing reuse issues. By ensuring and emphasizing the important factors presented in this work, software reusability can overcome software crises and make it easier for users to interact with and comprehend the software. This reduces support costs and improves overall efficacy, thereby reducing problems like project delays and budget overruns. These crucial reusability characteristics can be used in the future to assess any software's

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

reusability using machine learning methods. We may create bespoke datasets or use information from existing repositories to assess software reusability based on these characteristics.

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

REFERENCES

Ahmaro, I. Y. Y., bin Mohd Yusoff, M. Z. & Mohd Abualkishik, A. (2014). The current practices of software reusability approaches in Malaysia. *Malaysian Software Engineering Conference (MySEC)*, Langkawi, Malaysia, 172-176. https://doi.org/10.1109/MySec.2014.6986009

Alzahrani, A. & Khan, R.A. (2024). Secure software design evaluation and decision-making model for ubiquitous computing: a two stage ANN- Fuzzy AHP approach. *Computers in Human Behaviour, 153*(2024), 108109. https://doi.org/10.1016/j.chb.2023.108109.

Cebi, A. & Karal, H. (2017). An application of fuzzy analytic hierarchy process (FAHP) for evaluating students project. *Educational Research and Reviews*, *2017*, 120-132. https://doi.org/10.5897/err2016.3065

Chaudhary, R. & Chatterjee, R. (2013). Predilection of reusability over maintainability in aspect-oriented systems. *International Journal of Computers and Technology*, *6*(3), 423-435. https://doi.org/10.24297/ijct.v6i3.4482.

Crnkovic, I., Chaudron, M. & Larsson, S. (2006). Component-based development process and component lifecycle. *2006 International Conference on Software Engineering Advances (ICSEA'06)*, Tahiti, French Polynesia, 44-44. https://doi.org/10.1109/ICSEA.2006.261300

Dehraj, P. & Sharma, A. (2020). An empirical assessment of autonomicity for autonomic query optimizers using fuzzy-AHP technique. *Applied Soft Computing Journal*, *90*, 106137. https://doi.org/10.1016/j.asoc.2020.106137

Frakes, W.B. & Kang, K. (2005). Software reuse research: status and future. *IEEE Transactions on Software Engineering*, *31*(7), 529-536. https://doi.org/10.1109/TSE.2005.85

Fazal-e-Amin, Mahmood, A.K. & Oxley, A. (2010). Proposal for evaluation of software reusability assessment approach employing a mixed method. *ACM SIGSOFT Software Engineering Notes*, *35*(5), 1-4, https://doi.org/10.1145/1838687.1838703

Fazal-e-Amin, Mahmood, A.K., & Oxley, A. (2011). A mixed method study to identify factors affecting software reusability in reuse intensive development. *2011 National Postgraduate Conference*, Perak, Malaysia, 1-6. https://doi.org/10.1109/natpc.2011.6136324

International Organization for Standardization (ISO). (2024). Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Quality model overview and usage. ISO/IEC 25002: 2024 https://www.iso.org/obp/ui/en/#iso:std:78175:en

Jalender, B.Govardhan, A. & Premchand, P. (2011). Breaking the boundaries for software component reuse technology. *International Journal of Computer Applications*, *13*(6), 37-41. https://doi.org/10.5120/1782-2458

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

Jang, J.S.R. (1993). ANFIS: Adaptive network based fuzzy inference system. *IEEE Transactions on Systems and Man, and Cybernetics, 23*, 665-685. https://doi.org/10.1109/21.256541

Kabir, G. & Hasin, M. A. (2013). Comparative analysis of Artificial Neural Networks and neuro-fuzzy models for multicriteria demand forecasting. *International Journal of Fuzzy System Applications, 3*(1), 1-24. http://doi.org/10.4018/ijfsa.2013010101

Karaboga, D. & Kaya, E. (2019). Adaptive network based fuzzy inference system (ANFIS) training approaches: A comprehensive survey. *Artificial Intelligence Review, 52,* 2263-2293. https://doi.org/10.1007/s10462-017-9610-2

Karpak, B. (2017). Reflections: Mathematical principles of decision making. *International Journal of the Analytic Hierarchy Process, 9*(3), 341-348. https://doi.org/10.13033/ijahp.v9i3.521

Karunanithi, S. & Bieman, J.M. (1993). Candidate reuse metrics for object oriented and Ada software. *Proceedings of the First International Software Metrics Symposium*, Baltimore, MD, 120-128., https://doi.org/10.1109/METRIC.1993.263794

Kaur, P.J. & Kaushal, S. (2018). A fuzzy approach for estimating quality aspect-oriented systems. *International Journal of Parallel Programming*, *48*, 850-869. https://doi.org/10.1007/s10766-018-0618-2

Kaur, P.J., Kaushal, S., Sangaiah, A.K., & Piccialli, F. (2017). A framework for assessing reusability using package cohesion measure in aspect-oriented systems. *International Journal of Parallel Programming*, *46*, 543-564. https://doi.org/10.1007/s10766-017-0501-6.

Kim, Y. & Stohr, E.A. (1998). Software reuse: Survey and research directions. *Journal of Management Information Systems*, *14*(4), 113–147. https://doi.org/10.1080/07421222.1998.11518188

Kumar, V., Kumar, R., & Sharma, A. (2013). Applying neuro-fuzzy approach to build the reusability assessment framework across software component releases - An empirical evaluation. *International Journal of Computer Applications*, *70*(15), 41–47. https://doi.org/10.5120/12041-8047.

Kumar, R., Khan, S.A., Agarwal, A. & Khan, R.A. (2018). Measuring the security attributes through Fuzzy Analytic Hierarchy Process: Durability perspective. *ICIC Express Letters*, *12*(6), 615-620. http://dx.doi.org/10.24507/icicel.12.06.615

Kumar, R., Alenezi, M., Ansari, T.J., Gupta, B.K., Agarwal, A. & Khan, R.A. (2020). Evaluating the impact of malware analysis techniques for securing web applications through a decision-making framework under fuzzy environment. *International Journal of Intelligent Engineering and Systems*, *13*(6), 94-109. http://dx.doi.org/10.22266/ijies2020.1231.09

Kumar, R., Baz, A., Alhakami, H. Alhakami, W., Agrawal, A., & Khan, R. (2021). A hybrid fuzzy rule-based multi-criteria framework for sustainable-security assessment of web applications. *Ain Shams Engineering Journal*, *12*(2), 2227-2240. https://doi.org/10.1016/j.asej.2021.01.003

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256

Kusumawardani, C.A., Rosyidi, C.N. & Jauhari, W.A. (2016). The evaluation of criteria and subcriteria of research project selection using fuzzy analytical hierarchy process method. *2nd International Conference of Industrial, Mechanical, Electrical, and Chemical Engineering (ICIMECE)*, Yogyakarta, Indonesia, 112-117. https://doi.org/10.1109/ICIMECE.2016.7910429

Lanergan, R.G. & Grasso, C.A. (1984). Software engineering with reusable designs and code. *IEEE Transactions on Software Engineering*, *SE-10*(5), 498-501. https://doi.org/10.1109/TSE.1984.5010273.

Lounis, H. &Ait-Mehedine, L. (2004). Machine learning techniques for software product quality assessment. *QSIC 2004 Proceedings*, 102-109. https://doi.org/0-7695-2207-6/04

Makni, L., Zaaboub, N. & Ben-Abdallah, H. (2014). Reuse of semantic business process patterns. *Proceedings of the 9th International Conference on Software Engineering and Applications (ICSOFT 2014), SciTePress*, 36-47. https://doi.org/10.5220/0005003500360047

Mehboob, B., Chong, C.Y., Lee, S.P., Lim, J.M.Y. (2021). Reusability affecting factors and software metrics for reusability: A systematic literature review. *Software: Practice and Experience*, *51*(6), 1416–1458. https://doi.org/10.1002/spe.2961

Musa, J.D. (1985). John D. Musa on software: Productivity, quality, and human factors. *IEEE Spectrum*, *22*(1), 37-37. https://doi.org/10.1109/MSPEC.1985.6370520

Papamichail, Michail, Diamantopoulos, T. Chrysovergis, I. Samlidis, P. & Symeonidis, A. (2018). User-perceived reusability estimation based on analysis of software repositories. *IEEE Workshop on Machine Learning Techniques for Software Quality Evaluation (MaLTeSQuE)*, Campobasso, Italy, 49-54.https://doi.org/10.1109/maltesque.2018.8368459

Paschali, M.E., Ampatzoglou, A., Bibi, A., Chatzigeorgiou, A., & Stamelos, L. (2017). Reusability of open-source software across domains: A case study. *The Journal of Systems and Software*, *134*, 211-227. https://doi.org/10.1016/j.jss.2017.09.009.

Peisheng, L., Yunping, H., Xiaole, A., Shunshun, W. &Zhenglin, L. (2020). Research on information system risk assessment based on improved-AHP fuzzy theory. *Journal of Physics*, 1693, 1-7, https://doi.org/10.1088/1742-6596/1693/1/012046

Polat, A.G, & Alpaslan, F.N. (2023). The reusability prior: Comparing deep learning models without training. *Machine Learning: Science and Technology*, *4*(2), 1-18. https://doi.org/10.1088/2632-2153/acc713

Putra, M.S.D., Andryana, S., Fauziah, & Gunaryati, A. (2018). Fuzzy analytical hierarchy process method to determine the quality of gemstone. *Advances in Fuzzy Systems*, *2018*, 1-6, https://doi.org/10.1155/2018/9094380.

Saaty, T.L. (1985). Decision making for leaders. *IEEE Transactions on Systems, Man, and Cybernetic*s, *SMC-15*(3), 450-452. https://doi.org/10.1109/tsmc.1985.6313384.

Saaty, T.L. (1988). How to make a decision: The Analytic Hierarchy Process. *Proceedings of the International Symposium on the Analytic Hierarchy Process*. https://doi.org/10.13033/isahp.y1988.042.

Salomon, W.J., Wallace, D.R. (1994*). Quality characteristics and metrics for reusable software (Preliminary Report).* Gaithersburg, MD: National Institute of Standards and Technology https://doi.org/10.6028/nist.ir.5459.

Sant'Anna, C., Garcia, A., Chavez, C., Lucena, C., von Staa, A. (2003). On the reuse and maintenance of aspect-oriented software: An assessment framework. *Anais Do XVII Simpósio Brasileiro de Engenharia de Software (SBES 2003)*. https://doi.org/10.5753/sbes.2003.23850.

Sanz-Rodriguez, J., Dodero, J.M. &Alonso, S.S. (2011). Metrics-based evaluation of learning object reusability. *Software Quality Journal*, *19*(1), 121-140. https://dl.acm.org/doi/abs/10.1007/s11219-010-9108-5

Sharma, A., Grover, P.S., & Kumar, R. (2009). Reusability assessment for software components. *ACM SIGSOFT Software Engineering Notes*, *34*(2), 1–6. https://doi.org/10.1145/1507195.1507215

Shao, C. (2009). The implication of fuzzy comprehensive evaluation method in evaluating internal financial control of enterprise. *International Business Research*, *2*(1), 210-214. https://doi.org/10.5539/ibr.v2n1p210.

Singh, Y., Bhatia, P.K., & Sangwan, O. (2011). Software reusability assessment using soft computing techniques. *ACM SIGSOFT Software Engineering Notes*, *36*(1), 1-7. https://doi.org/10.1145/1921532.1921548

Singh, P.K, Sangwan, O.P., Pratap, A., & Singh, A.P. (2014). An analysis on software reusability in context of object oriented and aspect oriented software development. *International Journal of Information Security and Cybercrime*, *3*(2), 19–28, https://doi.org/10.19107/ijisc.2014.02.02

Singh, A.P., & Tomar, P. (2016). Web service component reusability evaluation: A fuzzy multi-criteria approach. *International Journal of Information Technology and Computer Science*, *8*(1), 40–47. https://doi.org/10.5815/ijitcs.2016.01.05

Singh, A., Tomar, P., & Pratap, A. (2016). Component reusability metrics to measure reusability of web services using fuzzy multi-criteria approach. *International Journal of Information Technology and Computer Science*, *8*(1), 1-16, https://doi.org/10.5815/ijitcs.2016.01.05

Singh, C., Pratap, A., Singhal, A. (2014). Estimation of software reusability for component based system using soft computing techniques. *5th International Conference - Confluence the Next Generation Information Technology Summit (Confluence)*, Noida, India, 788-794. https://doi.org/10.1109/confluence.2014.6949307

Standish, T.A. (1984). An essay on software reuse. *IEEE Transactions on Software Engineering*, *SE-10*(5), 494-497. https://doi.org/10.1109/TSE.1984.5010272

Thapar, S.S. & Sarangal, H. (2020). Quantifying reusability of software components using hybrid fuzzy analytical hierarchy process (FAHP) - Metrics approach. *Applied Soft Computing Journal*, *88*(2020), 105997. https://doi.org/10.1016/j.asoc.2019.105997.

Wang, Y.M. &Chin, K.S. (2011). Fuzzy analytic hierarchy process: A logarithmic fuzzy preference programming methodology. *International Journal of Approximate Reasoning*, *52*(4), 541-553, https://doi.org/10.1016/j.ijar.2010.12.004 .

Zhang, Z., Cao, J.W.H., Lu, Q. &Ding, M. (2017). Application of a fuzzy analytical hierarchy process for predicting the grindability of granite. *World Journal of Engineering and Technology*, *2017*, 117-125. https://doi.org/10.4236/wjet.2017.54b013

Zhao, X.,Qi, Q.,& Li, R. (2010). The establishment and application of fuzzy comprehensive model with weight based on entropy technology for air quality assessment. *2010 Symposium on Security Detection and Information Processing*, 217-222. https://doi.org/10.1016/j.proeng.2010.11.034.

ISSN 1936-6744
https://doi.org/10.13033/ijahp.v17i1.1256