The Analytical Hierarchy Process as a Tool to Select Open Source Software

RAFA E. AL-QUTAISH and MOHAMMAD I. MUHAIRAT
Department of Software Engineering
Al-Zaytoonah University of Jordan
Amman 11377
JORDAN
rafa@ieee.org, mohmuhaba14@yahoo.com

BASIL M. AL-KASASBEH
Faculty of Information Technology
Applied Science University
Amman 11931
JORDAN
b_kasasbeh@asu.edu.jo

Abstract: - Due to a high number of open source software over the internet, it is a challenge to determine which one – over a verity of alternatives – is the most appropriate for our intended needs. In this paper, the Analytical Hierarchy Process (AHP) technique along with the ISO 9126 will be integrated and applied to the selection process of open source software products. Thus, the intended features (i.e. the ISO 9126 quality characteristics) need to be identified in order to be taken into account to give a set of ranks for the intended open source software, that is, ‘rank one’ will be given to the most appropriate open source software product based on its intended features against the other alternatives of open source software products.


1 Introduction
In practical and due to a high number of open source software over the internet, it is a challenge to determine which one of them – over a verity of alternatives – is the most appropriate for your intended needs. In this paper, the Analytical Hierarchy Process (AHP) technique has been applied to the selection process of the open source software. Thus, the intended features (such as quality characteristics) need to be identified in order to be taken into account to give a set of ranks for the intended open source software, that is, ‘rank one’ will be given to the open source software which is the most appropriate to the intended features against the other alternatives of open source software products.

However, the quality characteristics of the ISO 9126-1 international standard will be used as criteria for the selection process and to choose the most appropriate open source software among the other possible alternatives.

The rest of this paper is structured as follows: Section 2 presents an overview of the decision making techniques. In section 3, we discuss the related concepts to this paper, such as Analytical Hierarchy Process technique and the Open Source Software. Section 4 presents the proposed approach for selecting open source software using AHP technique. In Section 5, a case study has been illustrated. Finally, section 6, discusses and concludes the paper.

2 Decision Making Methods: A General Overview
The process of selecting an open source software product is a decision making process. However, according to Harris [1] decision making is defined as “the study of identifying and choosing alternatives based on the values and preferences of the decision maker”. Making a decision involves a set of alternative choices to be considered, and in such a case we want not only to identify as many of these alternatives as possible but to choose the one that best fits with our needs.

Baker et al. [2] have stated that a general decision making process can be divided into the following steps:
1- Define the problem.
2- Determine requirements.
3- Establish goals.
4- Identify alternatives.
5- Define criteria.
6- Select a decision making tool.
7- Evaluate alternatives against criteria.
8- Validate solutions against problem statement.

The first five steps above are already identified since we are going to select open source software based on its features, such as quality characteristics. Steps seven and eight have been also taken into account. For step 6, there are several tools for solving a decision problem. The selection of an appropriate tool (step 6 above) is not an easy task and depends on the concrete decision problem, as well as on the objectives of the decision makers. Sometimes the simpler method is better, but complex decision problems may require complex methods, as well.

The following are some examples of the decision making methods, tools or techniques:
1- Pros and Cons Analysis method [1, 2],
2- Kepner-Tregoe Decision Analysis method [3],
3- Multi-Attribute Utility Theory method [2],
4- Cost-Benefit Analysis method [1, 2],
5- Custom-Tailored method [1, 2],
6- Maximin and maximax method [1, 2],
7- Conjunctive and Disjunctive method [1, 2],
8- Lexicographic method [4],
9- Simple Multi-Attribute Rating method [5],
10- Generalized Means method [6],
11- ELECTRE method [7],
12- PROMETHEE method [8, 9], and
13- Analytical Hierarchy Process (AHP) method [10].

In this paper, the Analytical Hierarchy Process (AHP) method will be used as a tool to select the appropriate open source software based on a set of selected features, such as quality characteristics. See subsection 3.1 for the details of the AHP method.

3 Related Concepts

3.1 AHP: the Analytical Hierarchy Process
Analytical Hierarchy Process (AHP) is an approach to decision making that involves structuring multiple choice criteria into a hierarchy, assessing the relative importance of these criteria, comparing alternatives for each criterion, and determining an overall ranking of the alternatives [11].

The AHP has been used to support the decision in information systems management [12], to early estimation of the function points [13], for architecture selection of ADSL modem [14], for a multi-criteria logistics-outsourcing decision making [15], to identify design requirements through agent-based simulation for personal air vehicle system [16], for tackling the uncertainty and imprecision of the service evaluation process [17], for irrigation water allocation in a small river basin [18], for the determination of relative significance factor of impact categories [19], as a decision-support system in the petroleum pipeline industry [20], in ecosystem management [21], for risk and opportunity assessment of international construction projects [22], in COTS decision-making [23], for project selection [24], to select a GIS software [25], and to translate common verbal phrases to numerical probabilities [26]. Furthermore, Koscianski and Costa [27] have used the AHP along with the ISO 9126 to build a quality evaluation framework.

AHP helps in capturing the subjective and objective evaluation measures, providing a useful mechanism for checking the consistency of the evaluation measures and alternatives suggested by the team, thus, reducing bias in decision making. Furthermore, it allows organizations to minimize common pitfalls of decision making process, such as lack of focus, planning, participation or ownership, which ultimately are costly distractions that can prevent teams from making the right choice.

AHP is very useful method when the decision-making process is complex. Certainly, when the decision cycle involves taking into account a variety of multiple criteria which rating is based on a multiple-value choice, AHP splits the overall problem into as many evaluations of lesser importance, while keeping at the same time their part in the global decision. The AHP consists of the following four steps:

1- Decomposing.
2- Weighing.
3- Evaluating.
4- Selecting.

3.2 OSS: the Open Source Software
The Open Source Software has been defined by Perens [28] as the software which its license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. According to Golden [29], the open source is software which has the following features:
1- Source code availability to its users.
2- It is distributed at no cost (free).

Therefore, to make any software as an open source software, the above two features should be implemented to that software. Implementing the first feature of the open source software (source code availability) may leads to the reuse of the available source code within other software products; thus, it is really important to focus on the quality of the components of the open source software. Distributing the open source software at no cost will increase the number of users of that software; therefore, the developers of such software products should focus on its quality to be a competitive one and to largely increase its number of users.
In addition, Coar [30] stated that the open-source software must comply with the following criteria:
1- The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources.
2- The program must include source code.
3- The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.
4- The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time.
5- The license must not discriminate against any person or group of persons.
6- The license must not restrict anyone from making use of the program in a specific field of endeavor.
7- The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.
8- The rights attached to the program must not depend on the program's being part of a particular software distribution.
9- The license must not place restrictions on other software that is distributed along with the licensed software.
10- No provision of the license may be predicated on any individual technology or style of interface.

4 Identifying the Selection Criteria
The ISO 9126-1 [31] international standard has been built to contain six quality characteristics for the internal and external software products. In addition, it contains a set of four characteristics to be applied to the software product quality in-use. In this paper, the external software product quality characteristics will be tackled as criteria for selection which open source software to be used against the other alternatives. The criteria (external software product quality characteristics) which could be used are:
1- Functionality (F).
2- Reliability (R).
3- Usability (U).
4- Efficiency (E).
5- Maintainability (M).
6- Portability (P).

In addition, with ISO 9126, there are another three standards; that is:
1- ISO 9126-2 on external quality metrics [32],
2- ISO 9126-3 on internal quality metrics [33], and
3- ISO 9126-4 on quality in use metrics [34].

However, each of them contains a set of software quality metrics related to each of the above six quality characteristics. Therefore, to measure any of the six quality characteristics, a set of related quality measures should be calculated then converted to a percentage value to represent the corresponding quality characteristics [35].

5 Using AHP to Select Open Source Software
The following are the steps in which they should be followed in order to get the ranking of the alternative OSS products:
1- Select which criteria (quality characteristics of ISO 9126-1) you want to apply.
2- Determine the relative importance of the selected criteria.
3- Build a matrix of the relative importance.
4- Square the matrix.
5- Compute the eigenvector (to four decimal places), as the following:
   a. Sum the rows.
   b. Sum the row totals.
   c. Divide the row sum by the row totals.
   d. The result is the eigenvector.
6- If the eigenvector does not change from the previous iteration go to next step, otherwise go to Step 4.
7- Identify a set of alternative OSS products which accomplish your needs.
8- Collect the related software metrics based on ISO 9126-2 (external metrics) for each of the selected characteristic (from Step 1).
9- Compute the corresponding software metrics for each criterion.
10- For each criterion, convert the result to a percentage value.
11- In terms of criteria (Quality Characteristic) determine the reference of each alternative OSS over another based on the percentage value of each criteria.
12- Compute the eigenvector to determine the relative ranking of alternative OSS under the selected criteria.
13- Build a matrix of the eigenvectors of the relative ranking of each alternative OSS under each criterion.
14- Multiply the matrix of the alternative OSS eigenvectors by the criteria eigenvector, the result is the ranking of the alternative OSS.
6 Applying the AHP Approach: An Example

Suppose that we have five alternative OSS products to choose from, and we have selected the following three quality characteristics to be the selection criteria for our alternative OSS products:

1- Reliability (R).
2- Usability (U).
3- Maintainability (M).

At the beginning, we need to determine the importance of the selected criteria, see formula (1). For example, the Usability is 2 times as important as Reliability, whereas, the maintainability is 3 times as important as Usability. Formula (2) shows the same contents of formula (1) but after converting them to four decimal values.

\[
\begin{array}{ccc}
R & U & M \\
R & 1 & 2/1 & 1/2 \\
A= & U & 1/2 & 1 & 1/3 \\
M & 2/1 & 3/1 & 1 \\
\end{array}
\]  \hspace{1cm} (1)

\[
\begin{array}{ccc}
R & U & M \\
R & 1.0000 & 2.0000 & 0.5000 \\
A= & U & 0.5000 & 1.0000 & 0.3333 \\
M & 2.0000 & 3.0000 & 1.0000 \\
\end{array}
\]  \hspace{1cm} (2)

Now, we have to square \((A*A)\) the matrix in formula (2), the results are shown in formula (3) below.

\[
\begin{array}{ccc}
R & U & M \\
R & 3.0000 & 5.5000 & 1.6666 \\
B= & U & 1.6666 & 2.9999 & 0.9166 \\
M & 5.5000 & 10.0000 & 2.9999 \\
\end{array}
\]  \hspace{1cm} (3)

Now, let's compute the first eigenvector, that is, sum the rows (for example, the summation of the first row is 10.1666), sum the row totals (34.2496), and finally divide the row sum by the row totals, as in formula (4).

\[
\begin{array}{c}
10.1666 \\
C= & 5.5831 \\
18.4999 \\
34.2496 \\
& 0.2968 \\
& 0.1630 \\
& 0.5401 \\
\end{array}
\]  \hspace{1cm} (4)

The above process (computing the eigenvector) must be iterated until the eigenvector solution does not change from the previous iteration. However, the result of the above formula is the eigenvectors, as shown in Table 1.

**Table 1:** The First Eigenvector.

<table>
<thead>
<tr>
<th>R</th>
<th>U</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2968</td>
<td>0.1630</td>
<td>0.5401</td>
</tr>
</tbody>
</table>

The second eigenvector have been computed by squaring the formula (3). The second eigenvector is shown in Table 2. Anyway, we will stop the process of computing the eigenvector since the two eigenvectors are close to each other.

**Table 2:** The Second Eigenvector.

<table>
<thead>
<tr>
<th>R</th>
<th>U</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2969</td>
<td>0.1634</td>
<td>0.5396</td>
</tr>
</tbody>
</table>

From the above table, we can note that the Maintainability is the most important criterion, the Reliability is the second most important criterion, and the Usability is the least important criterion.

Now, suppose that we have collected and computed the related external metrics of the selected quality characteristics (Reliability, Usability, and Maintainability) for each of the five open source software products, and then we have converted them to percentage values, as in Table 3 below.

**Table 3:** The Percentage values of the Selected Criteria (Quality Characteristics) for each OSS product.

<table>
<thead>
<tr>
<th>R</th>
<th>U</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSS 1</td>
<td>0.4378</td>
<td>0.6789</td>
</tr>
<tr>
<td>OSS 2</td>
<td>0.7598</td>
<td>0.5819</td>
</tr>
<tr>
<td>OSS 3</td>
<td>0.5436</td>
<td>0.2383</td>
</tr>
<tr>
<td>OSS 4</td>
<td>0.6762</td>
<td>0.9135</td>
</tr>
<tr>
<td>OSS 5</td>
<td>0.8976</td>
<td>0.3465</td>
</tr>
</tbody>
</table>

Finally, we need to multiply the matrix of Table 3 by the eigenvector of Table 2; this multiplication will give us the ranking of the 5 alternative Open Source Software (OSS) products. The results are shown in Table 4 below.
Table 4: The Multiplication of the Table 3 by Table 2.

| OSS 1 | 0.3096 |
| OSS 2 | 0.7435 |
| OSS 3 | 0.6721 |
| OSS 4 | 0.8482 |
| OSS 5 | 0.7225 |

From Table 4, we can note that the OSS 4 is the most appropriate one based on the 3 quality characteristics; the OSS 2 is the second most appropriate one, and so on. In more details, Table 5 shows the ranking of the five OSS products.

Table 5: The Ranking of the Five OSS Products.

<table>
<thead>
<tr>
<th>Open Source Software</th>
<th>Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSS 1</td>
<td>5</td>
</tr>
<tr>
<td>OSS 2</td>
<td>2</td>
</tr>
<tr>
<td>OSS 3</td>
<td>4</td>
</tr>
<tr>
<td>OSS 4</td>
<td>1</td>
</tr>
<tr>
<td>OSS 5</td>
<td>3</td>
</tr>
</tbody>
</table>

7 Conclusion

Due to a high number of open source software over the internet, it is a challenge to determine which one – over a verity of alternatives – is the most appropriate for our intended needs.

In this paper, the Analytical Hierarchy Process (AHP) technique and the ISO 9126 quality model have been integrated and applied to the selection process of the open source software products. Thus, the intended features (i.e. the ISO 9126 quality characteristics) need to be identified in order to be taken into account to give a set of ranks for the intended open source software, that is, ‘rank one’ will be given to the open source software which is the most appropriate to the intended features (quality characteristics) against the other alternatives of open source software products.

Furthermore, in order to apply the proposed approach in this paper, the external metrics for each of the selected ISO 9126-1 quality characteristics should be calculate. Thus, this approach should be used together with the ISO 9126-1 on quality model and ISO 9126-2 on external metrics.

Finally, from the given example in Section 6, we can note that this new approach is flexible in giving a ranking for each of the alternative open source software products in which anyone need to choose from.

Acknowledgment

The publication of this paper was financially supported by Al-Zaytoonah University of Jordan, Airport Street, Amman, Jordan.

References


