Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 9, Issue 2, pp: (28-36), Month: May - August 2022, Available at: www.noveltyjournals.com

Multicriteria Decision Analysis Using AHP in Cross-Platform Mobile Applications

Antonios-Dionysios Pavlozas

Department of Computer Engineering
University of West Attica
Egaleo, Greece

DOI: https://doi.org/10.5281/zenodo.6779569

Published Date: 29-June-2022

Abstract: **Nowadays the majority of people are using their mobile phones much more than any other device. Thus, every software company invests in the development of mobile applications. Undoubtedly, some enterprises can concentrate only on one mobile operating system and avoid all the others, yet, it is important to focus on a myriad of mobile devices with various operating systems using a single codebase. So, new technologies have emerged that allow developers to share parts of the codebase between platforms, and which refer to as cross-platform frameworks. However, one of the most challenging situations for application developers is which cross-platform framework to choose. Multicriteria decision analysis (MDCA) appears as an aid tool for this process decision-making. In this paper, the Analytic Hierarchy Process (AHP) between five top cross-platform frameworks will be used with the ultimate goal of finding which of these frameworks presents the best score based on given criteria.**

Keywords: **Cross-platform framework, Mobile application, MCDA, AHP.**

1. INTRODUCTION

In recent decades, smartphones are an integral part of most people's daily lives and can be characterized as probably the most important invention of the twenty-first century. At the same time, there are many different smartphones on the market with different operating systems. Among the most popular operating systems are Android, iOS and Windows. However, this diversity in operating systems is a challenge for businesses to develop applications targeting all or most operating systems from a single codebase.

Hence, the applications for smartphones (mobile apps) are in a stage where they have to exist on multiple platforms in order to reach out as many users as possible. Developing in native languages for multiple platforms requires more resources for software development companies, which will also result in higher expenses. Through cross-platforming, developers can create applications for multiple platforms using the same codebase. Although there are several cross-platform application development frameworks, it is still a challenge for businesses or developers to choose which of them will suffice all their application functionality and user experience requirements.

In this paper, a targeted analysis on five attributes of the five most widely used cross-platform frameworks will be conducted. These cross-platform frameworks are the Flutter, the React Native, the Apache Cordova, the Ionic and the Xamarin. Furthermore, the criteria that will be used in the selection of the best cross-platform framework are the operating system (OS), the code reuse, the price, the playground, and the community.

2. CROSS-PLATFORM FRAMEWORKS

Cross-platform is computer software implemented on multiple computing platforms [1, 2]. In practice, it allows developers, with a specific programming language, to develop software by writing the program only once and running it on all systems with little or without any modification. Currently, there are handfuls of cross-platform options to choose

from, when a developer creates an application. This survey will focus on five cross-platform frameworks, Flutter, React Native, Apache Cordova, Ionic and Xamarin, as they considered the five most used frameworks by software developers worldwide according to a statistical survey conducted the period 2021 in Statista [3].

Flutter is an open source and free cross-platform framework that developed and released by Google in 2017. It uses Dart language to develop applications for Android, iOS, macOS, Windows, Linux and Web from a single codebase [4]. It also includes an online coding playground which called FlutLab. Flutlab is a modern Flutter online IDE and the best place to create, debug, and build cross-platform projects with native performance, as it can reuse code up to 85% [5]. In addition, it should be noted that it has the largest community with 134,000 Github stars.

React Native is Facebook's free open source cross-platform mobile app development framework released in 2015. It uses JavaScript and React.js libraries to build applications for Android, iOS, macOS, Windows and Web [6]. It also includes an online coding playground which called Expo. Expo is a browser-based environment for developing applications and it is a great place to develop applications without needing to install the various SDKs and tools needed for native iOS and Android development. Moreover, pre-developed components in the open source library enable developers to freely access scripts of codes. These scripts are already written and the developers can use them directly, resulting in faster development and reuse of up to 75% of the code [5]. What is more, React Native has one of the largest communities with 100,000 Github stars.

Apache Cordova (formerly PhoneGap) is a free open source framework that was acquired by Adobe Systems in 2011. It utilizes HTML, CSS, and JavaScript to build hybrid web applications for Android, iOS, Windows and OS X [7]. Apache Cordova applications are executed within wrappers targeted to each platform, and rely on standards-compliant API bindings to access each device's capabilities such as sensors, data, network status, etc. Furthermore, the native APIs can be accessed from the Apache Cordova plugins and thus, achieve up to 80% code reuse [8]. Nevertheless, it has a small community with 5,900 Github stars.

Developed and released by Drifty Co. in 2013, Ionic is an open source framework that utilizes web technologies such as HTML, CSS, and JavaScript with integrations for frameworks such as Angular, React and Vue to build hybrid Android, iOS, Windows, and Progressive Web Apps [9]. Basically, Ionic provides the front-end UI framework that handles all the look and feel, and enables UI interactions of a hybrid app. Its commercial use is free but the pro and enterprise use cost 539$/year and 2999$/year respectively. Additionally, Ionic provides up to 98% of reusable code and a relatively good community number of 46,000 Github stars [8].

Released in 2011 and acquired by Microsoft in 2016, Xamarin is an open source cross-platform app development framework that uses C# language, .Net framework and Visual Studio to build applications for Android, iOS, tvOS, watchOS, macOS, and Windows [10]. Its commercial use is free but the business and enterprise use cost 999$/year and 1899$/year respectively. In Xamarin, developers can share as much as 75% of the code across different operating systems [11]. Lastly, the community is not large enough as it reaches 9,300 Github stars.

Table 1 summarizes the attributes of each cross-platform framework that will be used later in multicriteria decision analysis.

Table 1: Cross-platform framework attributes

	Flutter	**React Native**	**Apache Cordova**	**Ionic**	**Xamarin**
OS	Android iOS Windows macOS Linux Web	Android iOS Windows macOS Web	Android iOS Windows OS X	Android iOS Windows Web	Android iOS Windows macOS watchOS tvOS
Code Reuse	85%	75%	80%	98%	75%
Price	Free	Free	Free	Partial Paid	Partial Paid
Playground	☑	☑	☒	☒	☒
Community	134K	100K	5,9K	46K	9,3K

Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 9, Issue 2, pp: (28-36), Month: May - August 2022, Available at: www.noveltyjournals.com

3. MULTICRITERIA DECISION ANALYSIS

A multicriteria decision analysis (MCDA) can be used to identify and compare different policy options by assessing their effects, performance, impacts, and trade-offs. Furthermore, it provides a systematic approach for supporting complex decisions according to pre-determined criteria and objectives [12].

Therefore, MCDA is a useful tool that attempts to capture the informal, subjective and often the incalculable sense of preference of a user or decision maker. It is most applicable to solving problems that are characterized as a choice among alternatives. There are many multicriteria decision analysis methods, but the most widely used is the Analytic Hierarchy Process (AHP) [13, 14]. This method is preferred by decision makers and researchers, because it is simple, easy to understand, and has the ability to measure the relative performance of the alternatives. Before analyzing the AHP method, it is necessary to define two basic conditions. The first is to assume a decision maker (DM), who will set the appropriate conditions where needed, and the second is to change the scale of linguistic attributes [13].

3.1 Decision maker

To make a good decision, a decision maker must know and define the problem, the need and the purpose of the decision, to evaluate the alternatives, the stakeholders and groups affected [12]. A decision maker could be a single person or a group of decision makers. In this paper, the role of the DM is assigned to the author, who determines which criteria are necessary, defines the importance of the various elements and the optimal value of each criterion (maximum or minimum). Moreover, the term criteria indicates the attributes of the cross-platform frameworks of Table 1, and the term alternatives indicates the five different cross-platform frameworks. Table 2 displays the importance of each criterion and matching the cross-platform frameworks with the alternatives, while Table 3 shows the optimal value of each criterion.

Table 2: DM's preferences

Ranking Based on Importance	Criteria	Alternatives	Cross-Platform Frameworks
1	OS	Alt. 1	Flutter
2	Code Reuse	Alt. 2	React Native
3	Price	Alt. 3	Apache Cordova
4	Playground	Alt. 4	Ionic
5	Community	Alt. 5	Xamarin

Table 3: DM's optimal value

	Alt. 1	Alt. 2	Alt. 3	Alt. 4	Alt. 5
OS (Best value: most OS)	Android iOS Windows macOS Linux Web	Android iOS Windows macOS Web	Android iOS Windows OS X	Android iOS Windows Web	Android iOS Windows macOS watchOS tvOS
Code Reuse (Best value: max %)	85%	75%	80%	98%	75%
Price (Best value: free)	Free	Free	Free	Partial Paid	Partial Paid
Playground (Best value: checked)	☑	☑	☒	☒	☒
Community (Best value: max stars)	134K	100K	5,9K	46K	9,3K

3.2 Scale of linguistic attributes

It is observed that not all the attributes have a numerical unit, and there will be a scale conversion in these attributes keeping in mind which value is optimal. So, the features that will be converted are OS, Price and Playground.

Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 9, Issue 2, pp: (28-36), Month: May - August 2022, Available at: www.noveltyjournals.com

The conversion of the values of the first criterion (OS) will be given based on how many different platforms they support. Two values will be used for the second criterion (price), in which the number 0 indicates the cross-platform framework that is completely free, and the number 1 the cross-platform framework that is partially paid. Two values will also be given to the third criterion (playground). Particularly, the number 1 will indicate the cross-platform framework that has an online playground and the number 0 will indicate the cross-platform framework that does not have it. The other two criteria (code reuse and community) do not need to be converted, only the percentage will be subtracted and the exact number will be assigned. After the conversions, the Table 4 will only contain numeric values with the converted optimal value of each criterion.

Table 4: Optimal value

	Alt. 1	Alt. 2	Alt. 3	Alt. 4	Alt. 5
OS (Best value: max)	6	5	4	4	6
Code Reuse (Best value: max)	85	75	80	98	75
Price (Best value: min)	0	0	0	1	1
Playground (Best value: min)	1	1	0	0	0
Community (Best value: max)	134000	100000	5900	46000	9300

4. ANALYTIC HIERARCHY PROCESS METHOD

The Analytic Hierarchy Process, normally called AHP, is a powerful, yet, simple method for making decisions. It aims to quantify the relative priority of the given set according to the appropriate value scale. The decision is usually based on the perception of the individual (DM), who is supposed to make the final decision and to assess priorities, emphasing the importance of consistency and correlation of the alternative, which has been compared in the whole decision-making process [15, 16]. AHP involves eight steps.

4.1 Define the problem

The AHP process begins by defining the problem and the alternatives to be evaluated. The problem to be addressed in this paper is to choose the best cross-platform framework among five alternatives.

4.2 Develop a hierarchy model

The objective is located at the top level of the hierarchy. The second level represents the criteria and the third level displays the alternatives. Each criterion is associated with all five alternatives as shown in Figure 1.

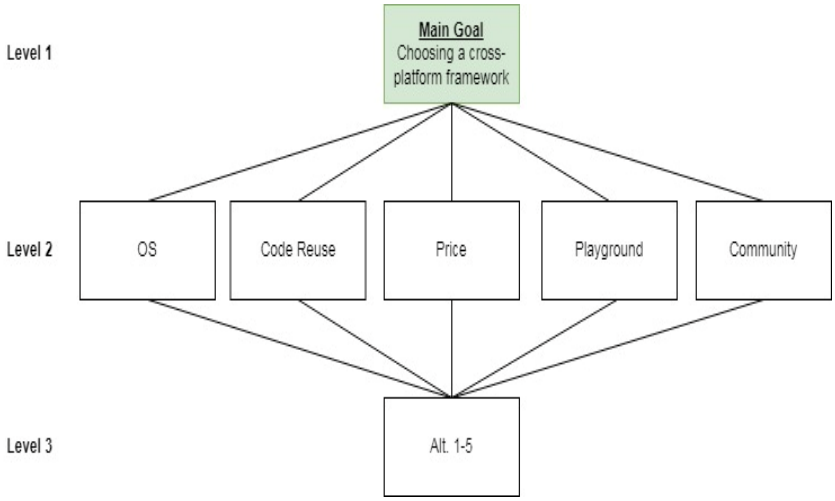

Figure 1. AHP hierarchical structure tree

Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 9, Issue 2, pp: (28-36), Month: May - August 2022, Available at: www.noveltyjournals.com

4.3 Construct a pairwise comparison matrix

A pairwise comparison matrix gives the relative importance of two elements with respect to the goal and is fundamental to the AHP methodology. Pairwise comparisons are made for each level of the hierarchy. Consequently, a pairwise comparison matrix (size 5×5) is, firstly, constructed for the criteria, and then, for each alternative in relation to each criterion.

4.4 Perform judgment for pairwise comparison

In this step DM has to compare each element by using Saaty's scale of relative importance [12]. The scale ranges from one to nine, where one implies that the two elements are the same or are equally important. On the other hand, number nine implies that one element is extremely more important than the other one in a pairwise matrix (Table 5).

Table 5: Saaty's scale

Importance Scale	Definition of Importance Scale
1	Equally Important
2	Equally to Moderately Important
3	Moderately Important
4	Moderately to Strongly Important
5	Strongly Important
6	Strongly to Very Strongly Important
7	Very Strongly Important
8	Very Strongly to Extremely Important
9	Extremely Important

Table 6 shows the comparisons based on DM, who asks each time how important one criterion is in relation to another.

Table 6: Pairwise comparison matrix

	OS	Code Reuse	Price	Playground	Community
OS	1	2	5	6	7
Code Reuse	1/2	1	5	6	7
Price	1/5	1/5	1	3	4
Playground	1/6	1/6	1/3	1	3
Community	1/7	1/7	1/4	1/3	1

4.5 Synthesizing the pairwise comparison

To calculate the vectors of priorities, the average of normalized column (ANC) method is used. In ANC, the elements of each column are divided by the sum of the column and, then, the elements in each resulting row are added and this sum is divided by the number of elements in the row (n). This is a process of averaging over the normalized columns [16]. In mathematical form, the vector of priorities can be calculated as shown in (1):

$$W_i = \frac{1}{n} \sum_{j=1}^{n} \frac{aij}{\sum_i^n aij}, \quad i, j = 1,2,\ldots\ldots n \qquad (1)$$

Gradually applying the formula, in Table 7 the sum of each value is calculated at the end of each column ($\sum_i^n \boldsymbol{aij}$).

Table 7: First priority vector

	OS	Code Reuse	Price	Playground	Community
OS	1	2	5	6	7
Code Reuse	1/2	1	5	6	7
Price	1/5	1/5	1	3	4
Playground	1/6	1/6	1/3	1	3
Community	1/7	1/7	1/4	1/3	1
Sum	**2,01**	**3,51**	**11,58**	**16,33**	**22**

Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 9, Issue 2, pp: (28-36), Month: May - August 2022, Available at: www.noveltyjournals.com

In Table 8 all the elements of each column is divided by the sum of the column **($aij / \sum_i^n aij$)**.

Table 8: Second priority vector

	OS	Code Reuse	Price	Playground	Community
OS	0,50	0,57	0,43	0,37	0,32
Code Reuse	0,25	0,28	0,43	0,37	0,32
Price	0,10	0,06	0,09	0,18	0,18
Playground	0,17	0,05	0,03	0,06	0,14
Community	0,07	0,04	0,02	0,02	0,05

Table 9 displays the weight (Wi) of each criterion, which is calculated from the average of each row.

Table 9: Third priority vector

	OS	Code Reuse	Price	Playground	Community
OS	0,50	0,57	0,43	0,37	0,32
Code Reuse	0,25	0,28	0,43	0,37	0,32
Price	0,10	0,06	0,09	0,18	0,18
Playground	0,17	0,05	0,03	0,06	0,14
Community	0,07	0,04	0,02	0,02	0,05
Weight (Wi)	**0,44**	**0,33**	**0,12**	**0,09**	**0,04**

4.6 Calculate Consistency

Consistency is to check whether the calculated values are correct or not, since the comparisons are carried out through personal or subjective judgments, which in some degree of inconsistency may occur. The consistency is determined by the consistency ratio (CR). Consistency ratio (CR) is the ratio of consistency index (CI) divided by the random index (RI); where RI is represented by average CI values gathered from a random simulation of Saaty's pairwise comparison matrices CIs [17]. The Saaty's average random index based on matrix size is shown in Table 10.

Table 10: Saaty's RI

Size of Matrix (n)	Random Consistency Index (RI)
1	0
2	0
3	0,52
4	0,89
5	1,11
6	1,25
7	1,35
8	1,4
9	1,45
10	1,49

If the CR>0,1 the judgment are untrustworthy as they are too close to randomness and the AHP method is valueless or must be repeated. The mathematical forms of CI and CR can be calculated as shown in (2) and (3):

$$CI = (\lambda max - n) / (n-1)\text{ , n= size of matrix} \quad (2)$$

$$CR = CI / RI, \quad (3)$$

According to formula (2), the Eigenvalue (λmax) must be calculated first. The λmax is calculated by taking the average of the ratio as shown in (4):

$$\lambda max = \frac{\sum_{i=1}^{n} Ratio_{(i)}}{n}\text{, n= size of matrix} \quad (4)$$

The ratio is found if the weighted sum of each row is divided by the weight of each criterion. For this purpose, the Table 6 which is not normalized is multiplied by the weight of each criterion that was found in Table 9.

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 9, Issue 2, pp: (28-36), Month: May - August 2022, Available at: www.noveltyjournals.com

Table 11: Ratio matrix

	OS	Code Reuse	Price	Playground	Community
OS	0,44	0,66	0,60	0,54	0,28
Code Reuse	0,22	0,33	0,60	0,54	0,28
Price	0,09	0,07	0,12	0,27	0,16
Playground	0,07	0,06	0,04	0,09	0,12
Community	0,06	0,05	0,03	0,03	0,04
Weighted Sum	**2,520**	**1,970**	**0,704**	**0,378**	**0,210**
Weight	**0,44**	**0,33**	**0,12**	**0,09**	**0,04**
Ratio	**5,73**	**5,97**	**5,87**	**4,2**	**5,25**

So, applying formulas (2), (3) and (4), it is found that $\lambda max = 5,4$, CI = 0,1 and CR = 0,09. As the value of CR is less than 0,1, the judgments of DM are acceptable and the analysis can be continued.

4.7 Perform steps 3–6 for alternatives

Continuing the analysis with the AHP method, the comparison matrix of alternatives in relation to each criterion will be implemented performing steps 3-6. Due to the large extensive counts and the large number of matrices, only the weights of each alternative and the values of formulas that determine the consistency will be presented in Table 12.

Table 12: Consistency matrix of alternatives

	OS	Code Reuse	Price	Playground	Community
Alt. 1 (W_1)	0,36	0,20	0,31	2,10	2,39
Alt. 2 (W_2)	0,15	0,07	0,31	2,10	1,38
Alt. 3 (W_3)	0,06	0,13	0,31	0,26	0,25
Alt. 4 (W_4)	0,06	0,53	0,04	0,26	0,62
Alt. 5 (W_5)	0,36	0,07	0,04	0,26	0,35
λmax	5,04	5,11	5,04	5,01	5,30
CI	0,009	0,029	0,009	0,003	0,075
RI	1,11	1,11	1,11	1,11	1,11
CR	0,008	0,026	0,008	0,003	0,068
Consistent?	YES	YES	YES	YES	YES

4.8 Develop overall ranking

The last step to complete the AHP method and classify cross-platform frameworks is to multiply the weight of each alternative shown in Table 12 with the weights of the criteria presented in Table 9. Once this is done, the weighted sum is found, where the score achieved by each cross-platform framework is presented in Table 13.

Table 13: Weighted sum of alternatives with respect to criteria

	Alt. 1	Alt 2.	Alt. 3	Alt 4.	Alt 5.
OS (W=0,44)	0,158	0,066	0,026	0,026	0,158
Code Reuse (W=0,33)	0,066	0,023	0,043	0,175	0,023
Price (W=0,12)	0,037	0,037	0,037	0,005	0,005
Playground (W=0,09)	0,189	0,189	0,023	0,023	0,023
Community (W=0,04)	0,096	0,055	0,010	0,025	0,014
Weighted Sum	**0,55**	**0,37**	**0,14**	**0,25**	**0,22**

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 9, Issue 2, pp: (28-36), Month: May - August 2022, Available at: www.noveltyjournals.com

5. RESULTS

The results from the analysis show that out of the five competing cross-platform frameworks, which satisfy five criteria, Flutter has the best score with importance difference from React Native as shown in Table 14.

Table 14: Overall ranking

Alternatives	Cross-Platform Framework	Score	Ranking
Alt. 1	Flutter	0,55	1
Alt. 2	React Native	0,37	2
Alt. 3	Apache Cordova	0,14	5
Alt. 4	Ionic	0,25	3
Alt. 5	Xamarin	0,22	4

Furthermore, it should be noted that the three best cross-platform frameworks are the ones most used by the developers in 2021 [3]. The difference according to the analysis lies in the fact that Apache Cordova is used more than Xamarin, while Xamarin has better score, and so better attributes.

6. CONCLUSION

This paper presents the methodology for evaluating and selecting the most appropriate candidate for making a cross-platform mobile application by implementing the AHP method. AHP provides a rational framework for a needed decision by quantifying its criteria and alternative options, and for relating those elements to the overall goal. The results from the analysis show that the Flutter is by far the best candidate for building a cross-platform mobile application. Through the evaluation of this method the mobile developers will have the opportunity to look for which platform is best; and also provides the best platform that can be used by companies, to save money and time.

REFERENCES

[1] M. Latif, Y. Lakhrissi, E.H. Nfaoui and N. Es-Sbai, “Cross platform approach for mobile application development: A survey,” in *International Conference on Information Technology for Organizations Development (IT4OD)*, 2016.

[2] T.A. Majchrzak, A. Biørn-Hansen and T.M. Grønli, “Comprehensive Analysis of Innovative Cross-Platform App Development Frameworks,” in *Proceedings of the 50th Hawaii International Conference on System Sciences (HICSS)*, 2017.

[3] Statista, “Cross-platform mobile frameworks used by software developers worldwide from 2019 to 2021,” 2021. [Online]. Available: https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/

[4] Flutter, “Flutter documentation,” 2022. [Online]. Available: https://docs.flutter.dev/

[5] I. Swarna, J. Purnama and R. Anthony, “Cross-Platform Analysis and Development of Online Catering Platform (Kunyahku),” *Journal of Applied Information, Communication and Technology*, vol. 7, no. 2, pp. 79–89, 2021.

[6] React Native, “Running On Device,” 2022. [Online]. Available: https://reactnative.dev/docs/running-on-device

[7] Apache Cordova, “Documentation,” 2022. [Online]. Available: https://cordova.apache.org/docs/en/10.x/

[8] A.D. Pavlozas, “Internet technologies for the development of compatible applications on different computing platforms (Master Dissertation),” *University of West Attica Institutional Repository*, 2021.

[9] Ionic, “Introduction to Ionic,” 2022. [Online]. Available: https://ionicframework.com/docs/

[10] Xamarin, “Cross-platform with Xamarin,” 2022. [Online]. Available: https://dotnet.microsoft.com/en-us/apps/xamarin/cross-platform

[11] H. Nair and V. Yadav, “Study on Cross-Platform Mobile App Development With Xamarin,” *International Journal of Trend in Scientific Research and Development (IJTSRD)*, vol. 2, no. 4, pp. 2554–2557, 2018.

[12] R.F.S.M. Russo and R. Camanho, "Criteria in AHP: A Systematic Review of Literature," *Procedia Computer Science*, vol. 55, pp. 1123–1132, 2015.

[13] T.D.C. Frazão, D.G.G. Camilo, E.L.S. Cabral and R.P. Souza, "Multicriteria decision analysis (MCDA) in health care: a systematic review of the main characteristics and methodological steps," *BMC Medical Informatics and Decision Making*, vol. 18, no. 90, pp. 1–16, 2018.

[14] E. Broniewicz and K. Ogrodnik, "A Comparative Evaluation of Multi-Criteria Analysis Methods for Sustainable Transport," Energies, vol. 14, no. 5100, pp. 1-23, 2021.

[15] T. Atanasova-Pacemska, M. Lapevski and R. Timovski, "Analytical Hierarchical Process (AHP) Method Application in the Process of Selection and Evaluation," in *2014 International Scientific Conference (UNITECH)*, vol. 14, pp. II 373-380.

[16] R. Velmurugan, S. Selvamuthukumar and R. Manavalan, "Multi criteria decision making to select the suitable method for the preparation of nanoparticles using an analytical hierarchy process," *Die Pharmazie*, vol. 66, no. 11, pp. 836-42, 2011.

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