e-ISSN 2757-5195

Çanakkale Onsekiz Mart University

Journal of Advanced Research in Natural and Applied Sciences

Open Access

doi.org/10.28979/jarnas.1204046 2023, Vol. 9, Issue 3, Pages: 637-647 dergipark.org.tr/tr/pub/jarnas

A mobile application-based decision support system for routing and decision making problems

Metehan Bolat[1], Oğuz Bedel[2], Kutay Çetinkurt[3], Fehmi Burcin Ozsoydan[4,*]

[1,2,3,4]Department of Industrial Engineering, Faculty of Engineering, Dokuz Eylül University, İzmir, Türkiye

Article History

Received: 16.11.2022

Accepted: 21.02.2023

Published: 20.09.2023

Research Article

Abstract − Industry 4.0 applications and related topics open up new opportunities in problem solving for companies, production systems and end users. Developments in this field make data gathering and processing easier. Moreover, mobile devices offer more sophisticated approaches, since, they can gather data and process the collected data by various algorithms, which are embedded to devices via applications. Such algorithms that are compatible with mobile devices can be used to increase the mobility and ease of use for the end users. In the present paper, a mobile application-based decision support system is developed for industrial systems including routing and multi-criteria decision-making problems. The developed application is comprised of several modules including sign-in and sign-up modules as well as problem solving modules. Problem solving modules can gather data from both end users and outer sources such as GPS in order to solve both routing problems and multi-criteria decision-making problems. The mentioned application adopts a Simulated Annealing Algorithm to find promising routes for the users, while multi-criteria decision-making module uses both Analytical Hierarchy Process (AHP) and the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) based on the choices of the users. The proposed application is coded on Java and Python programming languages due to their strong integration capabilities with Android Operating System. According to the outcomes of the developed prototype, one can conclude that the proposed application offers promising results and ease of use.

Keywords − *Decision support systems, mobile applications, multi-criteria decision making routing problems, simulated annealing algorithm*

1. Introduction

The rapid development in technology brings about numerous opportunities in various fields such as production systems, information systems, research and development, data science and logistics systems. In addition to such developments, mobile devices offer more sophisticated approaches since they can gather data and process the collected data by various algorithms employed by mobile applications. Thus, by making use of such opportunities offered by both technological developments and mobile applications, numerous daily routine or profession related problems can be solved independently to fixed places or offices. For example, a mobile phone can be used for talking and texting while a device that has access to internet can gather and process data via particular algorithms that are employed by applications, which are deployed to these devices.

The development in programming languages exhibits a similar pattern to the development in technology. In the past decades, big and powerful computers have been used to solve problems encountered in industry. In order to provide easiness in understanding and coding programming languages, a translation software, that is referred to as compiler was developed (Harper and Stockman, 2014). Subsequent to the first introduction of

[1] metebbb04@gmail.com

[2] bedeloguz@gmail.com

[3] kutay215@gmail.com

[4] burcin.ozsoydan@deu.edu.tr

*Corresponding author

Assembly, Fortran (Formula Translation) is developed, since, compilers and programming languages higher performance were required. Subsequently, various programming languages such as BASIC, PASCAL, C, C++, OBJECTIVE-C, HTML, Java, JavaScript, Python, Ruby, PHP, Swift and Python have been developed in accordance with the increment in requirements of hardware and software. Among all these developments, Java and Python are known to be two popular programming languages, since, Java can be used for developing both mobile and desktop applications while Python has a large variety of packages and libraries developed for numerous areas (Berkeley, 2021). Each of the reported programming languages has their own advantages and shortcomings. For example, while Python is not known to be a tailor-made programming language for mobile computing and it uses large amounts of memory. Java, on the other hand, can be used for developing both mobile and desktop applications with high level of detail.

The present study introduces a prototype mobile application-based decision support system (DSS) (Dizman and Özen, 2017) that is particularly devised for industrial systems. The developed application has a modular architecture and is comprised of several modules including sign-in and sign-up modules as well as problem solving modules. Thus, this mobile application uses a database that keeps the records of end-users. Problem solving module can gather data from both users and outer sources such as GPS in order to solve routing problems and multi-criteria decision-making problems. In order to solve routing problems, the developed mobile application adopts a Simulated Annealing (SA) Algorithm (Kirkpatrick et al., 1983). Secondarily, multi-criteria decision-making module adopts Analytical Hierarchy Process (AHP) (Gülenç and Bilgin, 2010) and the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) (James et al., 2021) based on the preferences of the users. The proposed application was coded on Java and Python due to their strong integration capabilities with Android. Thus, the developed application adopts a cooperation and integration of Java and Python programming languages in order to execute given tasks by the end-users. In this regard, while Python is used in the implementation of the developed algorithms, Java is used for mobile application design. Bringing all things together, the developed mobile application-based DSS aims to solve problems which can be encountered even in daily routine or in professional business life independently to any fixed locations like offices. This constitutes the main motivation of the present study.

The rest of this paper is organized as follows: Section 2 is devoted to the related literature review, while Section 3 presents focused problems and the details of the used algorithms. Finally, experimental results and concluding remarks are presented by Section 4 and Section 5, respectively.

2. Literature review

The related literature includes a wide range of reported publications. Due to space limitation, only closely related studies including both mobile applications and some DSS applications adopted in real-life are reported.

Bourouis et al. (2014) propose a mobile application to enable users for eye examination and to protect eye health by providing early diagnosis of eye diseases. According to the reported results, this application provides accurate diagnoses. İlhan (2017) introduces a Travelling Salesman Problem (TSP) interface that can be run on mobile devices. It should be mentioned that with the help of Google Maps API, real-world locations and distances can be obtained instantly. In another study, Ant Colony Optimization Algorithm is used to solve TSP by Liang et al. (2008). In their study, Xu et al. (2018) report theoretical and real-world applications of multiple TSP. They propose a modified Genetic Algorithm (GA) (Thede, 2004) to solve the mentioned problem. Küçükoğlu et al. (2019) present a new solution approach to the Time Windowed Electric TSP (ETSPTW). SA and Tabu Search (TS) algorithms are used to solve this problem. Mu et al. (2017) report and application of AHP. Their reported study includes a model that is established for buying a new car. Rupnik et al. (2018) aim to overcome the shortcomings and difficulties in agricultural systems by using DSSs. Thus, they propose a model denoted by AgroDSS system. This system is designed to be used by farmers and it provides farmers with predictions of scenarios simulated by using artificial intelligence techniques. Darko et al. (2018) report another AHP related study in the field of construction management. Rajak and Shaw (2019) develop a mobile application-based health program mHealth, which adopts AHP and TOPSIS. The usability of the proposed method is demonstrated by implementing a case study. In this study, a significant difference between the weighted and unweighted methods in the ranking is not reported. Vitale (2014) reports a mobile application based DSS for local public transport service. Hyun et al. (2021) report a software development framework to facilitate the development of a decision support system based on a mobile computing platform. The proposed is framework is used for organic fertilizer management in organic farming. Filippopulos et al. (2020) report a computerized clinical DSS and mobile application in order to optimize management of vertigo in primary care. The authors introduce another health care-related publication. An overview of clinical decision support systems is reported by Sutton et al. (2020). Shah et al. (2020) investigate the impacts of different components of mobile dining on customers' perceived value, which leads to actual purchase intentions. The study is conducted via mobile app data. Sarker et al. (2021) present a comprehensive view on mobile data science and intelligent apps to design and develop intelligent mobile applications for the betterment of human life in their diverse day-to-day situation. Another end-user related study is addressed by Andronie et al. (2021). The authors report prior findings indicating that mobile social apps extend throughout consumer attitudes and behaviours by the widespread adoption of smartphones. Thus, the authors show that cutting-edge technological developments associated with customer behaviour and mobile applications result in the rise of data-driven systems. Another mobile application based DSS related study is introduced by Belanche et al. (2020). The authors examine the phenomenon of food delivery services on mobile applications. In one of the recent studies, Mathirajan et al. (2021) develop a cloud-based DSS for public transport to optimize city bus transport operations. In another recent study, Nuanmeesri (2023) reports a mobile application based DSS study for marketing, product distribution and location-based logistics for elderly farmers.

As one can see from the closely related studies reported above, mobile applications deployed by mobile devices have great potential to overcome difficulties in problem solving in daily and professional life, in data gathering and data processing. Moreover, one can conclude that the mobile application-based DSSs have become popular particularly in the last decade. In this regard, the present study aims to contribute to the existing literature by the developed mobile application-based DSS to solve problems which can be encountered even in daily routine or in professional business life.

3. Used Solution Approaches

As addressed before, this study aims to develop a modular mobile application based DSS for routing and multi-criteria decision-making problem. Since TSP is one of the fundamental problems in routing, it is used as the case problem for the routing module of the developed application. It is also known that trajectory-based optimization algorithms are more appropriate for routing problems. Therefore, SA, which is particularly devised for permutation-based optimization problems, is used as the main optimizer of the routing module. The next sub-section is devoted to the used algorithm. Next, following sub-sections present adopted multi-criteria decision-making methods and finally, details of the developed mobile application-based DSS.

3.1. Simulated Annealing Algorithm

In the present work, SA is used as the solution algorithm in the routing module of the developed application. SA is a trajectory-based metaheuristic algorithm particularly devised for combinatorial optimization problems. Depending on a metaphor of annealing processes, it is introduced by Kirkpatrick et al. (1983). While improving solutions are accepted, also non-improving solutions are occasionally accepted based on a probability that is controlled by a parameter, referred to as temperature. Thus, SA has the capability of avoiding local optima. There is a number of applications summarized by Baykasoğlu and Ozsoydan (2016; 2017; 2018) of SA in the related literature.

Assuming that i and j are two different solutions for the related problem, $f(i)$ and $f(j)$ represent the objective function values, respectively. Then, for a minimization problem, the acceptance probability of solution j, which is in the neighbourhood of solution i can be evaluated based on the Eq. 3.1, where $T \in \mathbb{R}^+$ is the temperature parameter that can be determined either by users or by problem dependent formulations (Baykasoğlu and Ozsoydan, 2017). As formulated by Eq. 3.2, temperature T is decreased throughout iterations based on another user-supplied parameter $\alpha \in (0,1)$. Thus, acceptance probability of non-improving solutions is decreased throughout iterations. All steps of the used SA are illustrated by Algorithm 1.

$$P_j = \begin{cases} 1 & if \quad f(j) \leq f(i) \\ e^{-\left(\frac{f(j)<f(i)}{T}\right)} & o.w. \end{cases} \tag{3.1}$$

$$T_{k+1} = \alpha \times T_k \tag{3.2}$$

Algorithm 1. A pseudo code for SA.

```
initialize algorithm and parameters
generate a random solution x
gBest = x
        while termination criterion is not met
        while number of neighborhoods is not reached
            generate x'
            if f(x') ≤ f(x) or acceptance criterion is met
                x = x'
                if f(x) ≤ f(gBest)
                    x = x'
                    gBest = x
                end
            end
        end
        decrease T (Eq. 2)
        end
return gBest
```

3.2. Adopted Multi-Criteria Decision Making Methods

The two popular methods, namely, AHP and TOPSIS are adopted in this study. AHP (Saaty, 1970) is known to be a method for organizing and analysing complex decisions. It is comprised of three main components: the problem itself, all possible solutions (alternatives) and the criteria to judge those alternatives. An AHP implementation can be conducted by following the steps given below:

A hierarchical structure with a goal and a number of criteria are generated first. Next, among the determined criteria, a paired comparison matrix is formed up by using the scale of importance comprised of values between 1 and 9. After that, binary comparison matrices are normalized. Subsequently, priority vector is obtained by using the normalized matrix. Thereafter, the consistency rate is calculated and finally, the best alternative is selected. The main steps of AHP are summarized by Algorithm 2.

TOPSIS method is another multi-criteria decision-making method adopted by the developed application. It performs comparisons between the collected data and the available alternatives according to several predetermined criteria.

In this method, a matrix consisting of *M* alternatives and *N* criteria, which is known as the evaluation matrix (decision matrix), is created first. Next, using the data in this matrix, the standard decision matrix is evaluated. After that, weights of the normalized evaluation matrix are evaluated. As in other multi-criteria decision making problems, the used weights should be derived based on the knowledge of an expert for the relevant problem. Next, the best and the worst alternatives are determined for each criteria. Subsequently, one calculates the Euclidean distances between targeted alternatives and the best and the worst alternatives. A pseudo code for TOPSIS is given by Algorithm 3.

Algorithm 2. A pseudo code for the AHP.

```
define hierarchical matrix
define binary comparisons matrix
normalize binary comparisons matrixes
find priority vector
find consistency rate
make the best decision
```

Algorithm 3. A pseudo code for the TOPSIS.

```
define decision matrix
define standard decision matrix
define weighted standard matrix
find positive and negative ideal solution values
find distance values
find proximity coefficients
make the best decision
```

3.3. Developed Mobile Application Based DSS

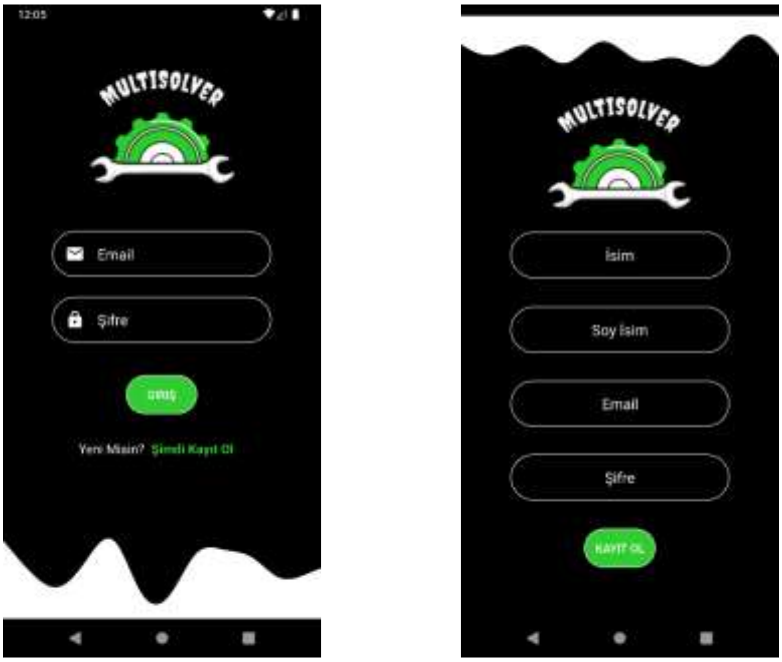
As mentioned above, a mobile application based DSS is proposed as a solution approach. The proposed mobile application is coded by using Android Studio in Java. In addition, the integration of Python and Java, which is necessary to solve the related problems, is provided by Chaquopy.

Since the verification tests of the developed application were conducted successfully, users can deploy this application to their individual devices by first generating a user account (Fig. 1.a). If an already existing account is used, the user can simply login by using e-mail and password (Fig. 1.b). Google Authentication Service is used for login and registration to the application.

Subsequent to creating account by sign-up module or singing-in via an already exiting account, users can access the modules of the application via a scrollable screen using the navigation drawer (Fig. 2.a). Moreover, users can optionally generate a profile, edit their individual user profile (Fig. 2.b) and generate a library (Fig. 2.c) that can record various data.

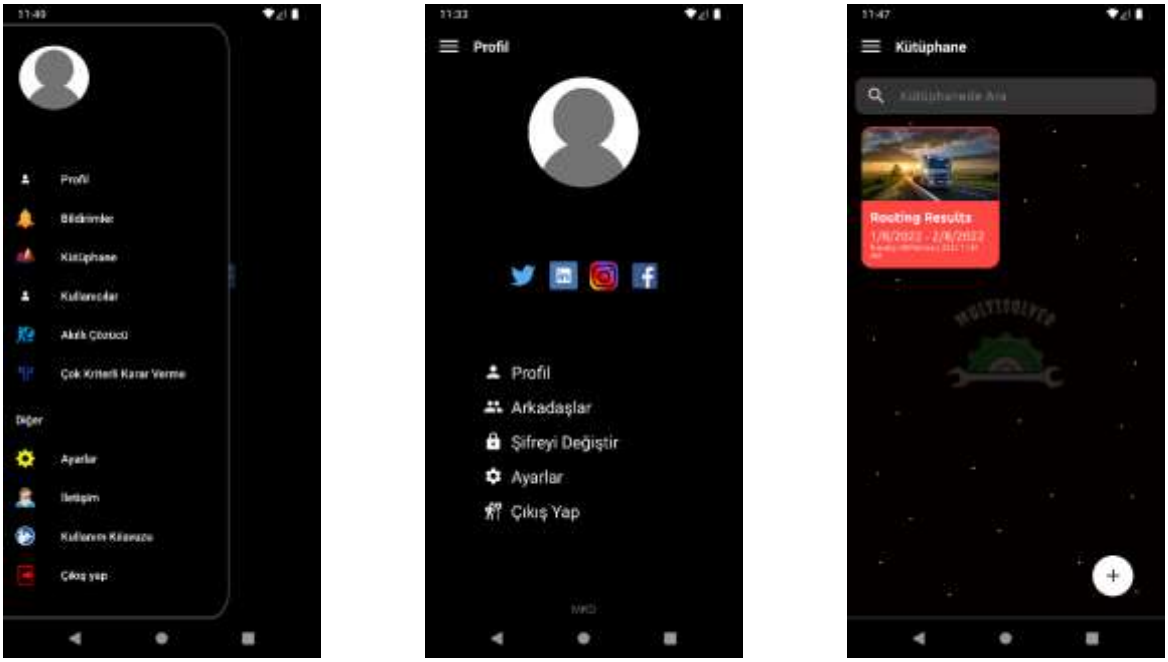
Routing module is illustrated by Fig. 3. As one can see from this figure, in routing module, users can pick some locations from the related map (Fig. 3.a, Fig. 3.b). Then best solution found by SA is printed on the final screen as illustrated by Fig. 3.c. Finally, application modules and interfaces for multi-criteria decision making problems are picturized by Fig. 4. Cleary, AHP and TOPSIS methods can be chosen to be used by the user via the developed mobile application. It should be mentioned that this module asks for the preferences of the users.

In the routing module, there are two options namely, *manual* and *automatic*. In the *manual option*, all required distance information should be provided manually to the matrix of the size that is specified by the user. After the data entry process, the most convenient route and distance for the user is obtained by clicking on the calculate button. In the *automatic option*, the real locations are marked on the map by the user as much as the number of points determined by the user via Google Maps. Finally, the most convenient route and calculated distance is obtained again by the same algorithm. Moreover, the *automatic option* has *airline* and *highway* options. Only Euclidean distance between the selected points is taken into consideration once the *airline option* is selected. Next, the data gathered from Google Maps along with the Android studio (user data) is transferred to Python via Chaquopy. Finally, obtained results are sent back to Android studio via Chaquopy again.



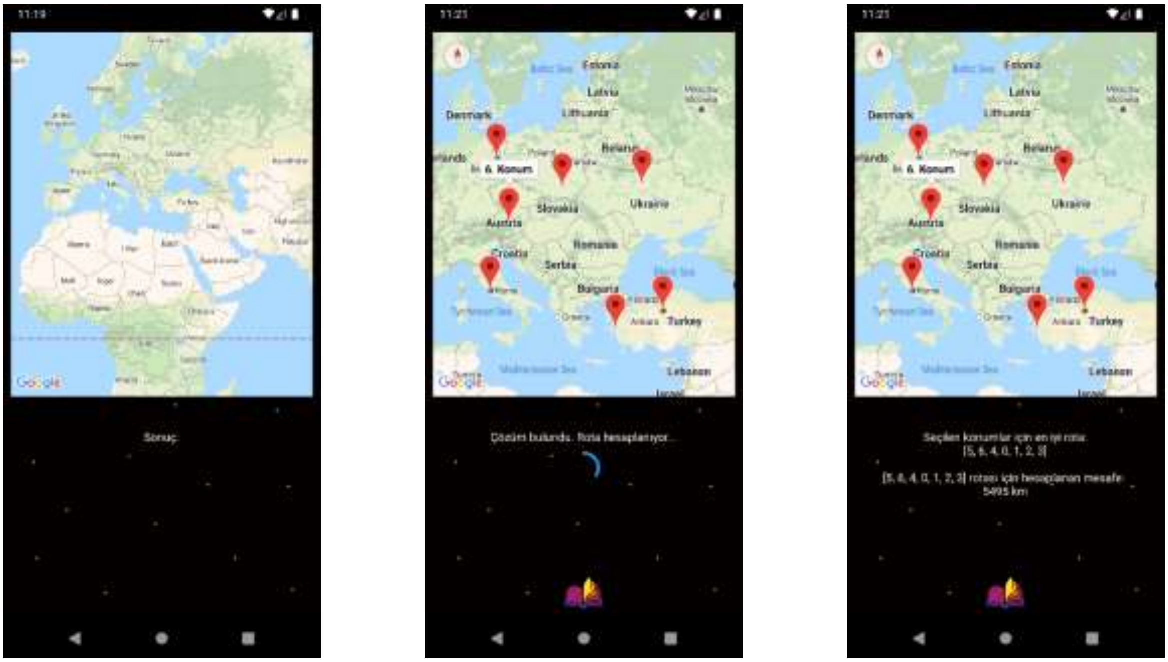
a **b**

Figure 1**.** An illustration for a) sign-in module and b) sign-up module



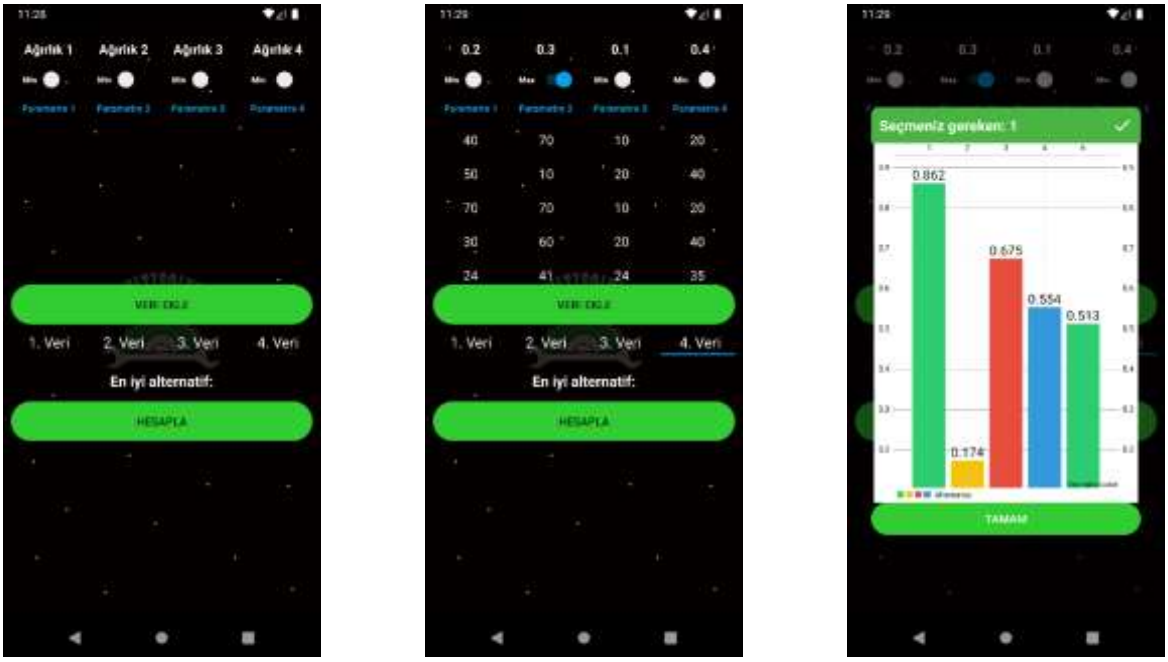
a **b** **c**

Figure 2. a) Navigation Drawer, b) Profile Module, c) Library Module

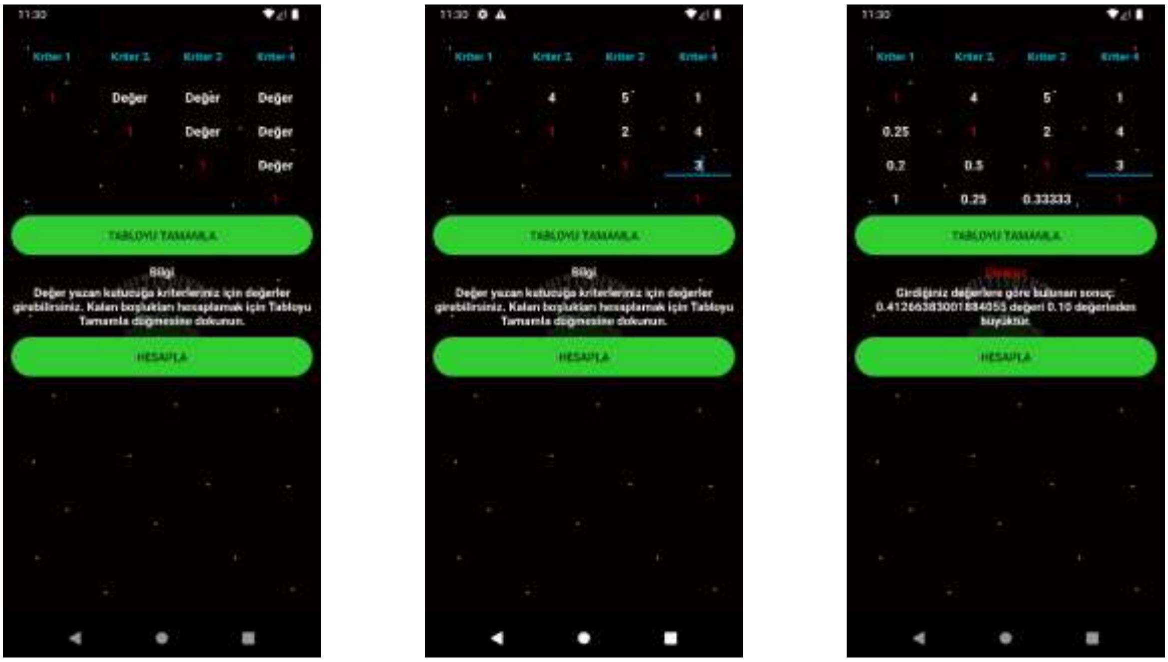


a **b** **c**

Figure 3. a) Navigation Drawer, b) Profile Module, c) Library Module



a b c



d e f

Figure 4. a) TOPSIS Screen, b) TOPSIS Data c) TOPSIS Result d) AHP Screen e) AHP Data f) AHP Result

As shown by Fig. 4, after selecting the number of criteria in the TOPSIS option, the names, weights and goal (max/min) of the criteria should be provided. The data of the alternatives that are planned to be compared should also be provided together with this data. After the required data entries are completed, analytical comparison of the alternatives is carried out by clicking on the calculate button. Finally, the result of these analytical calculations is shown to the user in a tabular form, where the most convenient alternative is specified. Secondarily, in the AHP option, after selecting the number of criteria, the upper part of the table, whose diagonal is the unit matrix, is filled according to the importance values. Then the lower part is filled automatically by clicking the complete table button. It is determined whether the criteria can be used for the problem according to the importance values entered by clicking the calculate button. Solutions of all solved problems are recorded in the library module.

4. Experimental Study

The scope of this study is not to present the ever-best solutions that are solved by mobile applications. On the contrary, the aim here is to present an alternative approach to bring various problems together within a mobile application that can be used as a DSS for end-users. In this context, a sample problem data is generated to test the routing module. The related data is provided by Table 1.

All tests are conducted on 3 smartphones using Android Operating System and on a PC. The first smartphone has a 2.3 GHz Samsung Exynos 9 Octa 8895 CPU (2017) and 6 GB RAM. The second one has a 2.8 GHz Qualcomm Snapdragon 845 CPU (2017) and 6 GB RAM and the third one has a 1.6 GHz Samsung Exynos 7 Octa 7870 (2016) and 3 GB RAM. Finally, the PC used in this experiment has I7-6700HQ CPU (2015) and 24 GB of RAM. All obtained results are given by Table 2.

It should be noted that all devices in all runs are able to find the same result. Moreover, in the present experimental study conducted with three different processors, it is observed that smart phone processors have close performance to the processors used by the tested PC. It is clear that this circumstance is due to the rapid development in smartphone processors and in the related technology. Most smartphone processors today perform similarly to those of the used by computers in terms of speed. Thus, it is shown that, mobile devices are also appropriate to solve small-scaled or medium-scaled routing problems without too specific attributes.

Therefore, it might be reported that the use of such devices and such approaches have great potential in problem solving.

Table 1. A sample data to test the routing module.

		1	2	3	4	5	6	7	8	9	10	11	12
	Distances (km)	Izmir	Istanbul	Moscow	Helsinki	Berlin	Kopenhagen	Warsaw	London	Budapest	Rome	Barcelona	Amsterdam
1	Izmir	0	327	2077	2425	1899	2211	1613	2573	1205	1306	2151	2314
2	Istanbul	327	0	1755	2152	1747	2032	1396	2508	1073	1381	2240	2218
3	Moscow	2077	1755	0	895	1614	1569	1161	2507	1569	2376	3013	2151
4	Helsinki	2425	2152	895	0	1108	889	918	1826	1461	2203	2608	1505
5	Berlin	1899	1747	1614	1108	0	355	512	930	694	1185	1501	573
6	Kopenhagen	2211	2032	1569	889	355	0	670	954	1018	1534	1761	618
7	Warsaw	1613	1396	1161	918	512	670	0	1442	543	1311	1858	1085
8	London	2573	2508	2507	1826	930	954	1442	0	1453	1434	1140	359
9	Budapest	1205	1073	1569	1461	694	1018	543	1453	0	810	1501	1147
10	Rome	1306	1381	2376	2203	1185	1534	1311	1434	810	0	859	1295
11	Barcelona	2151	2240	3013	2608	1501	1761	1858	1140	1501	859	0	1239
12	Amsterdam	2314	2218	2151	1505	573	618	1085	359	1147	1295	1239	0

Table 2. Obtained results for the sample problem

device	*CPU*	*CPU time (s)*	*Best Value (km)*	*Route for Best Value*
Smartphone	Exynos 9 Octa 8895	7.33	9480	5-6-4-3-7-9-2-1-10-11-8-12
	Snapdragon 845	3.71	9480	6-4-3-7-9-2-1-10-11-8-12-5
	Exynos 7 Octa 7870	16.05	9480	3-7-9-2-1-10-11-8-12-5-6-4
Computer	I7-6700HQ	2.65	9480	5-6-4-3-7-9-2-1-10-11-8-12

5. Concluding Remarks

As reported by the present paper, mobile devices offer more sophisticated approaches since they can gather data and process the collected data by various algorithms, which can be embedded to devices via applications. Thus, numerous problems can be solved independently to fixed places or offices. In this regard, the present study introduces a prototype of a mobile application-based decision support system (DSS) developed for industrial systems including routing and multi-criteria decision-making problems. The developed application is comprised of several modules including sign-in and sign-up modules as well as problem solving modules. Problem solving modules can gather data from both users and outer sources such as GPS in order to solve both routing problems and multi-criteria decision-making problems. The mentioned application adopts a Simulated Annealing Algorithm to find promising routes for the users, while multi-criteria decision-making module uses both Analytical Hierarchy Process (AHP) and the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) based on the choice of the users. The proposed application is coded on Java and Python due to their strong integration capabilities with Android.

As shown by the results of this study, the proposed methods have great potential to solve various problems while they also have some shortcomings. It is probable that the developed application might work slow particularly in large-scaled problems due to not having satisfying hardware features. Moreover, incompatibility problems due to device and operating systems versions, problems caused by the integration of libraries and services used by the application might emerge. These constitute some limitations of the proposed method. Nevertheless, as shown the experimental study, computational performances of the mobile devices are found as similar to that of the PC’s for small-scaled problems. This circumstance is due to the rapid development in smartphone processors and in the related technology. Thus, it is shown that, mobile devices are also appropriate to solve small-scaled or medium-scaled routing problems without too specific attributes. Therefore, making use of such devices have great potential. To sum up, the contributions of the present study can be summarized as follows:

- The proposed method discards the necessity for fixed locations and offices.
- The developed application takes end-user preferences into account in problem solving.
- Both daily routine problems and industrial problems can be handled by the proposed method.
- The developed application provides access to previous transactions by the record module.
- Modular design of the developed app makes it easy for integration and extension

Extending the present work for other various problems is scheduled as the future work.

Author Contributions

Metehan Bolat: Conceptualization, Methodology, Software, Validation, Testing, Writing - Original Draft.

Oğuz Bedel: Conceptualization, Methodology, Software, Validation, Testing, Writing - Original Draft.

Kutay Çetinkurt: Conceptualization, Methodology, Software, Validation, Testing, Writing - Original Draft.

Fehmi Burcin Ozsoydan: Conceptualization, Methodology, Investigation, Writing - Review & Editing, Supervision.

Conflicts of Interest

The authors declare no conflict of interest.

References

Andronie, M., Lăzăroiu, G., Ștefănescu, R., Ionescu, L., & Cocoșatu, M. (2021). Neuromanagement decision-making and cognitive algorithmic processes in the technological adoption of mobile commerce apps. Oeconomia Copernicana, 12(4), 1033-1062. https://doi.org/10.24136/oc.2021.034

Baykasoğlu, A., & Özsoydan, F. B. (2016). An improved approach for determination of index positions on CNC magazines with cutting tool duplications by integrating shortest path algorithm. International Journal of Production Research, 742-760. https://doi.org/10.1080/00207543.2015.1055351

Baykasoğlu, A., & Özsoydan, F. B. (2017). Minimizing tool switching and indexing times with tool duplications in automatic machines. The International Journal of Advanced Manufacturing Technology, 1775-1789. https://doi.org/10.1007/s00170-016-9194-z

Baykasoğlu, A., & Özsoydan, F. B. (2018). Minimisation of non-machining times in operating automatic tool changers of machine tools under dynamic operating conditions. International Journal of Production Research, 1548-1564. https://doi.org/10.1080/00207543.2017.1357861

Belanche, D., Flavián, M., & Pérez-Rueda, A. (2020). Mobile apps use and WOM in the food delivery sector: the role of planned behavior, perceived security and customer lifestyle compatibility. Sustainability, 12(10), 4275. https://doi.org/10.3390/su12104275

Berkeley. (2021). Berkeley Extension. Retrieved from: https://bootcamp.berkeley.edu/blog/most-in-demand-programming-languages/

Bourouis, A., Feham, M., Hossain, M. A., & Zhang, L. (2014). An intelligent mobile based decision support system for retinal disease diagnosis. Decision Support Systems, 341-350. https://doi.org/10.1016/j.dss.2014.01.005

Darko, A., Chan, A. P., Effah, E. A., Owusu, E. K., Pärn, E. A., & Edwards, D. J. (2018). Review of application of analytic hierarchy process (AHP) in construction. International Journal of Construction Management, 436-452. https://doi.org/10.1080/15623599.2018.1452098

Dizman, H., & Özen, E. (2017). Küçük İşletmelerde Karar Destek Sistemlerinin Farkındalığı Üzerine Bir Araştırma: Yerel Bir Yaklaşım (Ege Bölgesi). Finans Ekonomi ve Sosyal Araştırmalar Dergisi, 137-152. https://doi.org/10.29106/fesa.344962

Enrique, M. & Pereyra-Rojas, M. (2017). Group Decision-Making in AHP. In Practical Decision Making, by Enrique Mu, 81-90. Pittsburgh. https://doi.org/10.1007/978-3-319-33861-3

Filippopulos, F. M., Huppert, D., Brandt, T., Hermann, M., Franz, M., Fleischer, S., & Grill, E. (2020). Computerized clinical decision system and mobile application with expert support to optimize management of vertigo in primary care: study protocol for a pragmatic cluster-randomized controlled trial. Journal of neurology, 267, 45-50. https://doi.org/10.1007/s00415-020-10078-0

Gülenç, İ. F., & Bilgin, G. (2010). Yatırım Kararları İçin Bir Model Önerisi: Ahp Yöntemi - A Model Proposal For Investment Decisions: Ahp Method. Öneri Dergisi, 97-107. https://doi.org/10.14783/od.v9i34.1012000233

Harper, D., & Stockman, L. M. (2014). obliquity. obliquity. Retrieved from: https://www.obliquity.com/computer/fortran/history.html

Hyun, S., Yang, S. M., Kim, J., Kim, K. S., Shin, J. H., Lee, S. M., ... & Fleisher, D. H. (2021). Development of a mobile computing framework to aid decision-making on organic fertilizer management using a crop growth model. Computers and Electronics in Agriculture, 181, 105936. https://doi.org/10.1016/j.compag.2020.105936

İlhan, İ. (2017). An Application on Mobile Devices with Android. Applied Artificial Intelligence, 332-345. https://doi.org/10.1080/08839514.2017.1339983

James, A. T., Vaidya, D., Sodawala, M., & Verma, S. (2021). Selection of bus chassis for large fleet operators in India: An AHP-TOPSIS approach. Expert Systems with Applications, 186, 115760. https://doi.org/10.1016/j.eswa.2021.115760

Kirkpatrick, Scott, C. D. Jr. Gelatt, and M. P. & Jr. Vecchi. (1983). Optimization by Simulated Annealing. Science 671-680. https://doi.org/10.1126/science.220.4598.671

Küçükoğlu, İ., Dewil, R., & Cattrysse, D. (2019). Hybrid simulated annealing and tabu search method for the electric travelling salesman problem with time windows and mixed charging rates. Expert Systems with Applications, 279-303. https://doi.org/10.1016/j.eswa.2019.05.037

Liang, Y., Marchese, M., Shi, X., & Yang, J. (2008). An ant colony optimization method for generalized TSP problem. Progress in Natural Science, 1417-1422. https://doi.org/10.1016/j.pnsc.2008.03.028

Mathirajan, M., Devadas, R., & Ramanathan, R. (2021). Transport analytics in action: A cloud-based decision support system for efficient city bus transportation. Journal of Information and Optimization Sciences, 42(2), 371-416. https://doi.org/10.1080/02522667.2019.1688948

Nuanmeesri, S. (2023). Mobile application for the purpose of marketing, product distribution and location-based logistics for elderly farmers. Applied Computing and Informatics, 19(1/2), 2-21. https://doi.org/10.1016/j.aci.2019.11.001

Rajak, M., & Shaw, K. (2019). Evaluation and selection of mobile health (mHealth) applications using AHP and fuzzy TOPSIS. Technology in Society, 59, 101186. https://doi.org/10.1016/j.techsoc.2019.101186

Rupnik, R., Kukar, M., Vračar, P., Košir, D., Pevec, D., & Bosnic, Z. (2018). AgroDSS: A decision support system for agriculture and farming. Computers and Electronics in Agriculture, 260-271. https://doi.org/10.1016/j.compag.2018.04.001

Saaty, T. L. (1990). How to make a decision: The analytic hierarchy process. European Journal of Operational Research 9-26. https://doi.org/10.1016/0377-2217(90)90057-I

Sarker, I. H., Hoque, M. M., Uddin, M. K., & Alsanoosy, T. (2021). Mobile data science and intelligent apps: concepts, AI-based modeling and research directions. Mobile Networks and Applications, 26, 285-303. https://doi.org/10.1007/s11036-020-01650-z

Shah, A. M., Yan, X., Shah, S. A. A., & Ali, M. (2020). Customers' perceived value and dining choice through mobile apps in Indonesia. Asia Pacific Journal of Marketing and Logistics, 1-28. https://doi.org/10.1108/APJML-03-2019-0167

Sutton, R. T., Pincock, D., Baumgart, D. C., Sadowski, D. C., Fedorak, R. N., & Kroeker, K. I. (2020). An overview of clinical decision support systems: benefits, risks, and strategies for success. NPJ digital medicine, 3(1), 17. https://doi.org/10.1038/s41746-020-0221-y

Thede, Scott. (2004). An introduction to genetic algorithms. Journal of Computing Sciences in Colleges. 20. htts://doi.org/10.5555/1040231.1040247

Vitale, A., Festa, D. C., Guido, G., & Rogano, D. (2014). A Decision Support System based on smartphone probes as a tool to promote public transport. Procedia - Social and Behavioral Sciences, 224-231. https://doi.org/10.1016/j.sbspro.2014.01.055

Xu, X., Yuan, H., Liptrott, M., & Trovati, M. (2018). Two phase heuristic algorithm for the multiple-travelling salesman problem. Methodologies and Application, 6567-6581. https://doi.org/10.1007/s00500-017-2705-5