Accredited Ranking SINTA 2
Decree of the Director General of Higher Education, Research, and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online on: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 7 No. 1 (2023) 27 - 33 ISSN Media Electronic: 2580-0760

Mobile Application Development to Solve Vehicle Routing Problems in Marketing or Tour Trip Planning

Benni Agung Nugroho[1], Abidatul Izzah[2*], Kunti Eliyen[3]
[1,2,3]Manajemen Informatika, PSDKU Polinema di Kota Kediri
[1]benni.nugroho@gmail.com, [2]abidatul.izzah90@gmail.com[z], [3]kunti.eliyen@polinema.ac.id

Abstract

The vehicle routing problem (VRP) is a combinatorial optimization problem faced by transportation services related to pick up or delivery, such as industrial raw materials distribution, tour and travel, or travel routing problems in general. VRP is an NP-hard problem where the higher the dimensions of the problem will have a higher computational complexity. Without realizing it, VRP problem are often encountered every day. Therefore, it will be very useful if VRP solver is implemented in mobile application media. Previously, the Google Map application also provided a feature for determining route problems, but this did not solve the constraint problems that appeared in VRP. Therefore, the development of existing applications needs to be done. So, the aim of this work is developing a mobile application to get the shortest path and minimal cost in VRP problem. It is integrated by both Mapbox API and Google Maps API to get a real distance for modeling problem. The result show that the developed application can run well in all possibility condition.

Keywords: mobile application, optimization, software development, vehicle routing problem

1. Introduction

The vehicle routing problem (VRP) [1] is a problem in determining the shortest route that can be taken on a trip to several destination points and back to the starting point by minimizing the cost of the trip. In general, the VRP problem is another form of traveling salesman problem (TSP) [2]. The problem is described by a salesman who starts his journey from the starting point then visits several cities and returns to the starting point. So, the total distance traveled is a minimum and each city is visited only once. Some VRP faced in real life include in logistics (how to determine the best route to take in order to be able to deliver ordered goods as quickly and as much as possible), on tour services and travel (how to minimize the time to pick up and deliver travelers to their destination), and at supermarkets (how to determine the fastest route to deliver groceries to customers who shop online during the Covid-19 pandemic) and many others' problem exist.

Furthermore, the VRP is a combinatorial problem which is an NP-Hard problem [3] where the higher the dimensions of the problem, the computational time to solve the problem will increase exponentially. Therefore, the heuristic method [4] is used to obtain the optimal solution of the VRP. The optimal solution is not always the best solution, however, by considering the computational time and the closest solution, this heuristic method can be used to solve the VRP. VRP became one of the most popular combinatorial optimization problems, and it has many variations, for instance, the research about Capacitated Vehicle Routing Problem (CVRP) [5], Vehicle Routing Problem with Pickup and Delivery (VRPPD) [6], Vehicle Routing Problem with Time Windows (VRPTW) [7], Dynamic Vehicle Routing Problem [8] and many other combinations solution of VRP. Chen stated that Vehicle Routing Problem with Simultaneous Pickup and Delivery (VRPSPD) has been implemented in the logistics of various aspects, for instance, VRPSPD for home health care, transportation problem of public libraries, and so on [9].

Based on the explanation, VRP problem are often encountered every day. Therefore, it will be very useful if VRP solver is implemented in mobile application media to help us. If the development of this application is successfully carried out, many benefits can be felt by users. These benefits include being able to save time when doing tours, being able to provide navigation on the go, and saving on fuel costs because the route

Accepted: 30-09-2022 | Received in revised: 17-01-2023 | Published: 01-02-2023

distance will be shorter. So, the aim of this work is developing a mobile application to get the shortest path and minimal cost in VRP problem. Later this application will be used in logistic, tour and travel, supermarket, etc.

Further, this app can be useful for marketing couriers and tourists. *A*lthough *it* can help both activities, there is a fundamental distinction between search routes for marketing couriers and those for regular or tourist trips. Here are the differences*:* In marketing, mobile apps help one of the most important operational decisions concerns to finding optimal vehicle routes since it offers great potential to reduce time and to improve service quality[10]. Whereas, mobile apps help to plan the itinerary of a city bus catering to bus operators as well as tourists by minimizing operation (mileage) costs [11]. Besides that, by minimizing the routes, marketing couriers can deliver more packages compared before using mobile apps to minimize the routes. With more packages to deliver to the consumers, couriers can get bonus from its company because their work improved than before. A map with a tour route recommendation also extends the time spent by visitors at the area, increases the number of tourist attractions visited and the distance they would travel, while retaining the same level of satisfaction as when they travel without the tour route recommendation [12].

2. Research Methods

The main purpose of this research is to develop mobile application for Android to solve VRP named "Circular Route Optimizer and Finder". In order to accomplish it, this project focuses on determining if the development of the mobile application can achieve the main purpose of this research, determining if this mobile application able to reach the users (i.e. salesman, etc), and testing the functional of application to make sure it goes well. The following sub sections will highlight the task taken in order to achieve the main goal of this project. Figure 1 is a diagram that describes the research method.

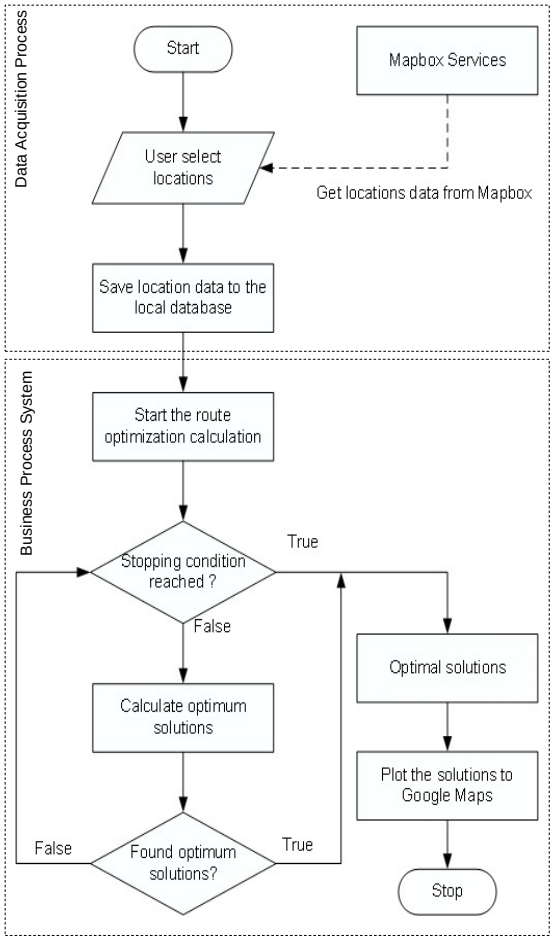


Figure 1. Diagrams for research methods

2.1 Data Acquisition Process

Firstly, we design a system architecture describe the interaction between data, application, user, and maps service provider. Figure 2 shown the system architecture design. First, the app starts with getting the position coordinates using the Cellular Base Transceiver Station (BTS) based GPS. It is used because of its convenience and cheapness[13][14]. Users will search and select the destination locations in the app. The app gets a list of locations that the user

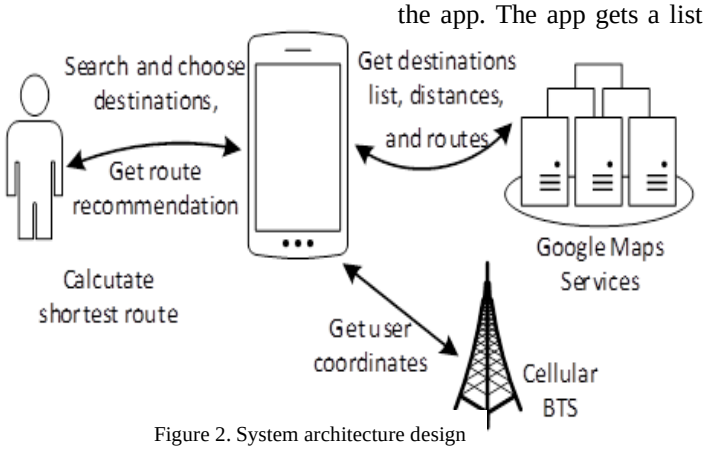


Figure 2. System architecture design

DOI: https://doi.org/10.29207/resti.v7i1.4552

wants to visit and search for the distance, route, time duration, and other information that is needed using Google Maps API Service. It is implemented because this service is proven to be easy and has been widely used [15]-[17]. After the data is obtained, it will be processed and the problem is solved.

The application developed uses two map services at once, namely from Mapbox by utilizing the Mapbox API and from Google Maps by utilizing the Google Maps API[18]. The data taken from Mapbox is in the form of location points that form a route that connects the (n) location with the (n+1) location. The location points are in the form of location coordinates in latitude and longitude[19]. While the data taken from Google Maps includes: location name, location address, location coordinates, further information from the location selected with the help of Google Maps which includes photos, location reviews from other users, opening and closing hours of the location and other information. can be provided by Google Maps.

The data that has been successfully acquired from the map service provider will be stored into an SQLite database available on Android mobile devices to be used in calculating the shortest route. SQLite is used because the android application will easily retrieve temporary data entered by the user[20]–[22]. In the SQLite database there is a table named *node_list* to store the selected location data including data of nodes, starting and destination points of the trip, the route between the starting point and the destination, as well as the distance between the starting point and destination. Table 1 explain *node_list* table.

Table 1. Node List Table

Field Name	Data Type	Description
Node	Text	Saving name of two connected points
Dist	Double	Saving distance between two points of location
Points	String	Saving route that connecting two points location
Srcloc	String	Saving latitude-longitude of first point
Dstloc	String	Saving latitude-longitude of last point

2.2 Business Process System

After the data is stored, the optimization process is executed. The optimization process will be repeated until the optimal solution is found. If the optimum one has been found or the stopping condition is reached then the route will be displayed through the plot on the map. The results of this route plot are in the form of a sequence of locations and also traversable road navigation.

3. Results and Discussions

3.1 Interface Implementation

Interface application designed to give excellent user experience for the user. The interface is designed as simple as possible to be easily understood by the user The application interface consists of three interfaces, namely: a search and location selection interface, a location selection refine interface, and a route information interface. Here, in Figure 3 we show the screenshots of the developed app with following explanations.

3.2 Testing Result

Then, system testing is taken to see the effectiveness of the android application that has been developed. The test method used is white box testing method and the black box testing method[23], [24]. The white box testing method is used to verify the internal perspective of system. While the black box is a method used to check whether all functions on the interface can run. For getting valid results, this test is carried out by experts. Next, the test begins by compiling the business process in a flowchart that summarizes all the possibilities of how the system works. The application path can be seen in Figure 4. First, we must specify the complexity value. If E is number of edges, N is number of nodes, and M is cyclomatic complexity value, then M is compute by formula [25]:

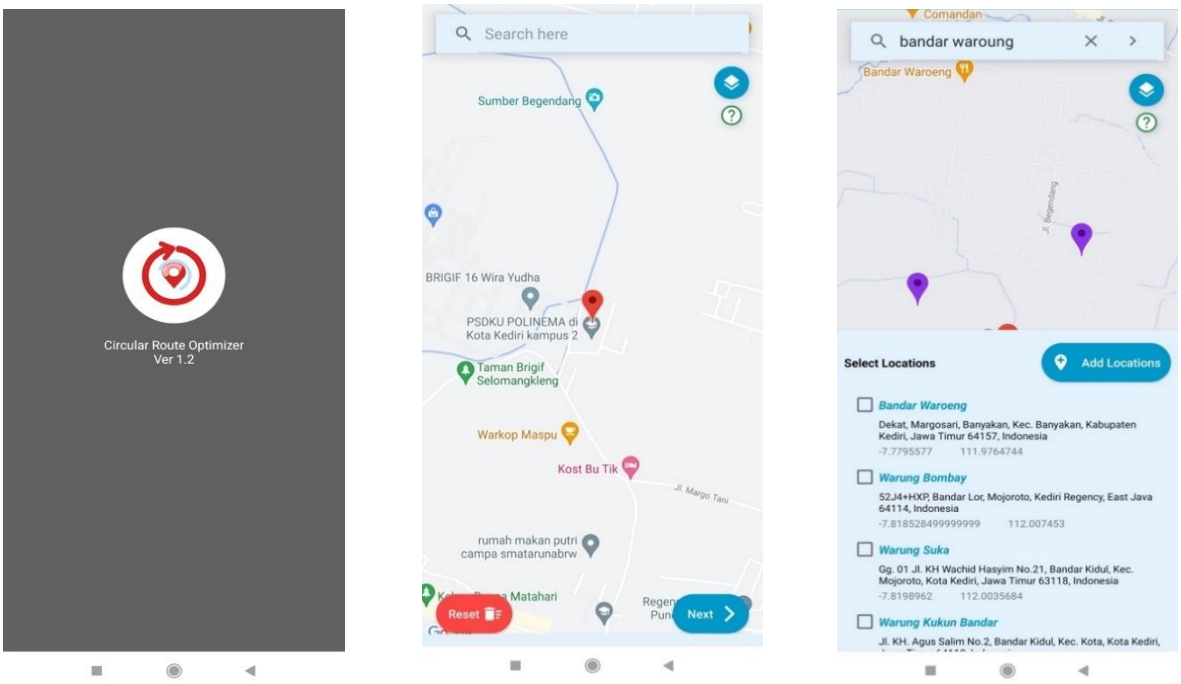
$$M = E - N - 2 \quad (1)$$

So, based on the Fig. 3 above, the cyclomatic complexity value is $M = 18 - 15 + 2 = 5$. The next step is to generate all the paths that are formed when the application is running. There are 5 paths that may be traverse which is shown as follows:

1) Path 1: 1-2-3-4-6-7-8-9-10-11-15
2) Path 2: 1-2-3-4-6-7-8-9-12-14-15
3) Path 3: 1-2-3-5-6-7-8-9-10-11-15
4) Path 4: 1-2-3-5-6-13-14-15
5) Path 5: 1-2-3-5-6-13-14-2-3-5-6-13-14-15

Furthermore, Table 2 shows the test results white box testing accompanied by an explanation of conditions and expected results. The results show that every possible path has produce the desired output. It can be concluded that all test results are valid. Therefore, based on this test, we can confirm that the system is well defined.

DOI: https://doi.org/10.29207/resti.v7i1.4552



(a) (b) (c)

(d) (e) (f)

Figure 3. (a)Splash Page (b) Current Location (c) Pinned Location by Search (d) Result of Searching Location. Purple Color by Pinned, Blue by Search (e) List Location (f) Route generated (as result of computation)

DOI: https://doi.org/10.29207/resti.v7i1.4552

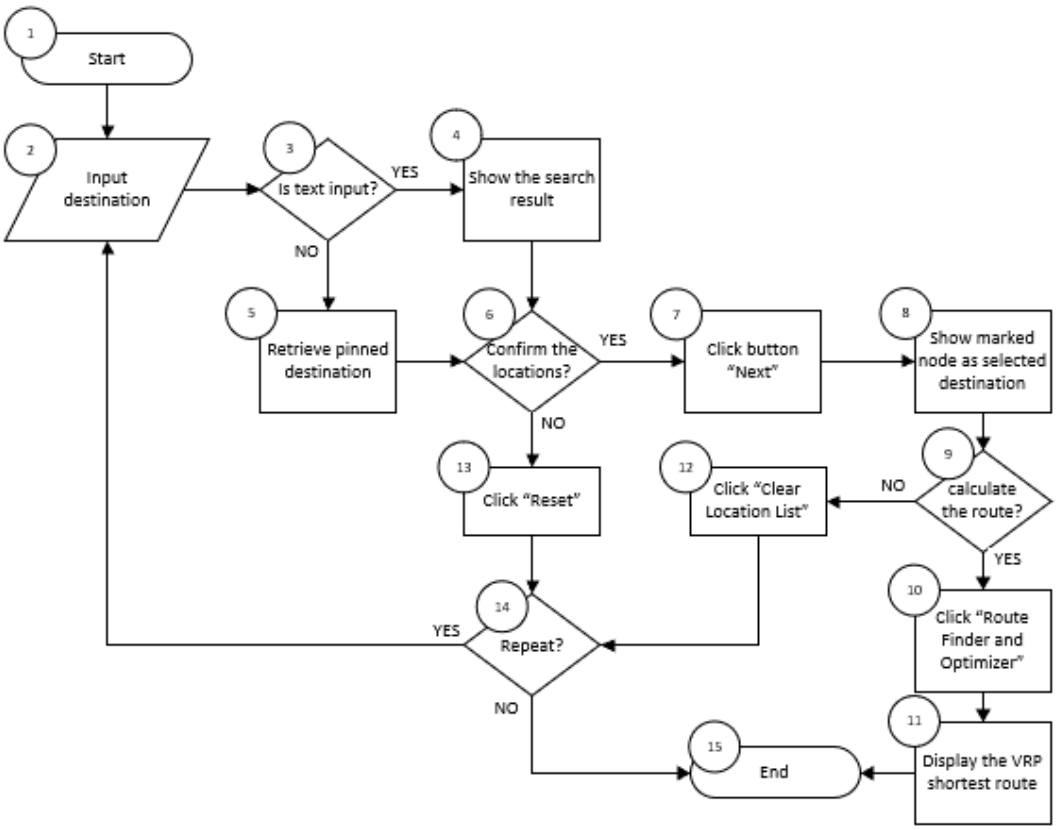


Figure 4. System Flow

Table 2. Node List Table

Path	Condition	Expected Result	Status
1	User inputs location data with text search then route optimizer is executed	Route optimization results will appear	Valid
2	User inputs location data with text search then route optimizer is cancelled.	The data list will appear but then return to the homepage	Valid
3	User inputs data with pinned the location then route optimizer is executed	Route optimization results will appear	Valid
4	User inputs location data but then resets and cancel	The data list will appear but then return to the homepage	Valid
5	User inputs location data but then resets and repeats it from the beginning	After canceling, app can repeat from the beginning	Valid

From the above, it indicates that the system flow is clear and valid. Now we will test the functional capability of the system based on black box testing. This test is carried out without recognizing the ins and outs of the system. This test is sufficient to verify the running of functional requirements.

Table 3. Functional Testing Result

Number	Test Case	Condition	Status
1	Open system	User can open the android apps by selecting the icon then the splash screen and homepage appear	Valid
2	Enter multiple destinations by both searching or picking	Users can enter multiple destination data by searching in database or directly picking them from the map	Valid
3	Verify destination list	User can see a list of input data as well as delete some unnecessary by unchecking	Valid
4	Reset data	User can delete all data and return to the homepage	Valid
5	Calculating the shortest route	Android app run the GA, PSO, or Hybrid PSO-GA method	Valid
6	Show the route map result	User can see a map of the shortest route in sequence which can be started to visit	Valid

DOI: https://doi.org/10.29207/resti.v7i1.4552

Furthermore, application testing by user is needed. User testing was conducted to assess and validate the performance of the application. Therefore, a survey method was carried out with an online questionnaire. Previously, it has been uploaded to the play store under the name "Circular Route Optimizer and Finder" so that correspondents can download it. By questionnaire, the respondents assess the performance of the application based on their respective experiences. The questions we ask are

a. Is the route recommendation displayed by the app clear and easy to understand?
b. Is it easy to add/delete location points?
c. Does the app provide route recommendations quickly?
d. Can the app display location search result exactly?
e. Can the app display your exact position?

And the survey results are presented in Figure 5.

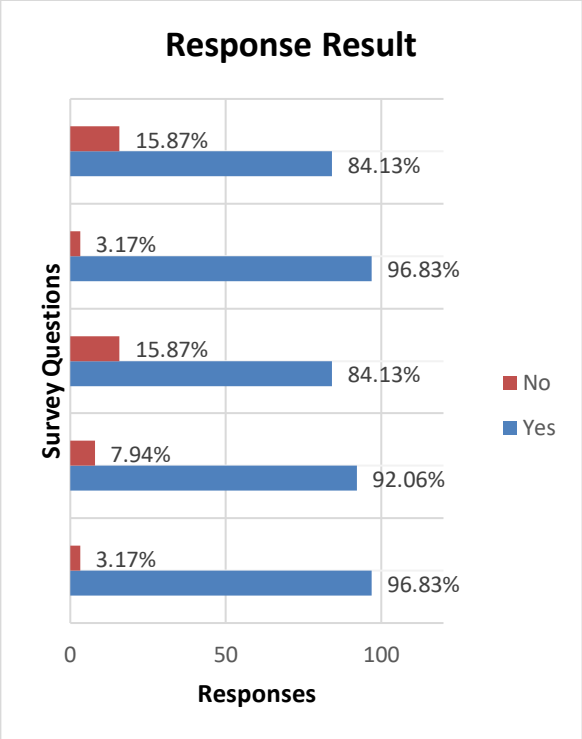


Figure 5. Overall survey result

A number of questions in the questionnaire given to users are actually evaluated to determine the ease of use (a and b), efficiency (c), and effectiveness (d and e) of a software application. Based on the result of 63 respondents who participated, initially, 96.83% of them choose 'Yes' indicating the app is able to display their exact position. Next, 92.06% of the respondents choose 'Yes' indicating the app is able to display location search result by name exactly. So, it shows that users can easily run the features in the apps such as searching for the current location, searching for destination locations, and route recommendation results. Next, 84.13% of the respondents choose 'Yes' indicating the app provide route recommendations quickly. Thus, the application is proven to be able to work in the application without requiring a long time. This is preferred because it is suitable for use in real time while in the vehicle. Next, 96.83% of respondents choose 'Yes' indicating that they were able to add new location points or delete location points easily. Finally, 84.13% of respondents choose 'Yes' indicating that they understood easily about route recommendation information displayed by the application. Then from these results, it shows that the purpose of application development has been successfully carried out, namely to provide the best and most accurate routes so that the problems raised earlier can be resolved.

4. Conclusion

This research is about a software development in Android mobile application which starts from design, implementation, and testing. In this study, application named "Circular Route Optimizer and Finder" was developed to solve the VRP problem by integrating Mapbox API and Google Maps API to provide an optimal route. Furthermore, a series of tests have been carried out by users and experts by black box and white box testing method. The result show that the developed application can run well in all possibility condition. Then, user experience testing with 63 respondents carried out gave valid results that indicates the system development is done well. At the end of this study, we hope that the application can be used by users who need to solve the VRP problem. It also has been uploaded to the android play store under the name "Circular Route Optimizer and Finder". Finally, this similar application has also been developed and available in the Google Playstore with various services. Therefore, for further research, a comparisons study about applications can be made to find the advantages and disadvantages. Thus, it can be obtained which applications are superior and in demand by users.

Acknowledgements

This research was supported by P2M State Polytechnic of Malang, Indonesia in a 2021 research grant. We would like to thank colleagues at the Information Technology Department who have provided insight and expertise that greatly assisted this research.

Reference

[1] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Manage. Sci.*, vol. 6, no. 1, pp. 80–91, 1959.
[2] J. Monnot and S. Toulouse, "The Traveling Salesman Problem and its Variations," in *Paradigms of Combinatorial Optimization: Problems and New Approaches: 2nd Edition*, 2014.
[3] M. Prates, P. H. C. Avelar, H. Lemos, L. C. Lamb, and M. Y. Vardi, "Learning to solve NP-Complete problems: A graph neural network for decision TSP," in *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative*

DOI: https://doi.org/10.29207/resti.v7i1.4552

Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, 2019.

[4] J. J. Cochran, L. A. Cox, P. Keskinocak, J. P. Kharoufeh, J. C. Smith, and M. E. Kurz, "Heuristics for the Traveling Salesman Problem," in *Wiley Encyclopedia of Operations Research and Management Science*, 2011.

[5] M. A. H. Akhand, A. B. M. Junaed, M. F. Hossain, and K. Murase, "Group Search Optimization to solve Traveling Salesman Problem," *Proceeding 15th Int. Conf. Comput. Inf. Technol. ICCIT 2012*, no. December, pp. 72–77, 2012.

[6] P. Chen, H. Huang, and X. Dong, "An ant colony system based heuristic algorithm for the vehicle routing problem with simultaneous delivery and pickup," *ICIEA 2007 2007 Second IEEE Conf. Ind. Electron. Appl.*, no. April, pp. 136–141, 2007.

[7] W. Ran, L. Liu, and G. Yang, "A hybrid ant colony algorithm for vehicle routing problem with time windows," *Inf. Technol. J.*, vol. 12, no. 20, pp. 5701–5706, 2013.

[8] V. Pillac, M. Gendreau, C. Guéret, and A. L. Medaglia, "A review of dynamic vehicle routing problems," *Eur. J. Oper. Res.*, vol. 225, no. 1, pp. 1–11, 2013.

[9] R.-M. Chen and P.-J. Fang, "Solving Vehicle Routing Problem with Simultaneous Pickups and Deliveries Based on A Two-Layer Particle Swarm optimization," in *2019 20th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, 2019, pp. 212–216.

[10] F. P. Goksal, I. Karaoglan, and F. Altiparmak, "A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery," *Comput. Ind. Eng.*, vol. 65, no. 1, pp. 39–53, 2013.

[11] S. A. Bagloee, M. Tavana, D. Di Caprio, M. Asadi, and M. Heshmati, "A multi-user decision support system for online city bus tour planning," *J. Mod. Transp.*, vol. 25, pp. 59–73, 2017.

[12] M.-C. Zheng, "How a Map with a Tour Route Recommendation Promotes Circuitous Tourism," *J. Asian Archit. Build. Eng.*, vol. 14, no. 2, pp. 363–370, 2015.

[13] M. Singhal and A. Shukla, "Implementation of Location based Services in Android using GPS and Web Services," *Int. J. Comput. Sci. Issues*, vol. 9, no. 1, pp. 237–242, 2012.

[14] M. E. Martawati, F. Rohman, H. F. D. Kurniawan, and I. N. Abidin, "The effect of distance Base Transceiver Station (BTS) on speed of vehicle safety response based on internet of things," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 732, no. 1, 2020.

[15] H. Zhang, M. Li, Z. Chen, Z. Bao, Q. Huang, and D. Cai, "Land use information release system based on Google Maps API and XML," in *2010 18th International Conference on Geoinformatics*, 2010, pp. 1–4.

[16] S. Hu and T. Dai, "Online Map Application Development Using Google Maps API , SQL Database , and ASP . NET," *Int. J. Inf. Commun. Technol. Res. ISSN*, vol. 3, no. 3, pp. 102–110, 2013.

[17] A. Pejic, S. Pletl, and B. Pejic, "An expert system for tourists using Google Maps API," in *2009 7th International Symposium on Intelligent Systems and Informatics*, 2009, pp. 317–322.

[18] K. Saputra, M. Furqan, T. F. Abidin, and D. H. Yunadi, "Google Maps and Mapbox Api Performance Analysis on Android-Based Lecture Attendance Application," *J. Nat.*, vol. 19, no. 3, pp. 64–68, 2019.

[19] G. Sajeevan, "Latitude and longitude - A misunderstanding," *Curr. Sci.*, vol. 94, no. 5, pp. 568–569, 2008.

[20] A. T. Mat, "Related Papers," *Over Rim*, pp. 191–199, 2017.

[21] S. Mutti, E. Bacis, and S. Paraboschi, "SeSQLite: Security enhanced sqlite: Mandatory access control for android databases," *ACM Int. Conf. Proceeding Ser.*, vol. 7-11-Decem, pp. 411–420, 2015.

[22] I. Musleh, S. Zain, M. Nawahdah, and N. Salleh, "Automatic generation of android SQLite database components," *Front. Artif. Intell. Appl.*, vol. 303, no. September, pp. 3–16, 2018.

[23] S. Nidhra, "Black Box and White Box Testing Techniques - A Literature Review," *Int. J. Embed. Syst. Appl.*, vol. 2, no. 2, pp. 29–50, 2012.

[24] M. E. Khan, "Different approaches to white box testing technique for finding errors," *Int. J. Softw. Eng. its Appl.*, vol. 5, no. 3, pp. 1–14, 2011.

[25] A. Abran and M. Lopez, "An Analysis of the McCabe Cyclomatic Complexity Number," *14th Int. Work. Softw. Meas. IWSMMetrikon*, pp. 391–405, 2004.

DOI: https://doi.org/10.29207/resti.v7i1.4552