LOGIC
Journal of Engineering Design and Technology
Vol. 19 No.3 November 2019; p. 120 - 130

p-ISSN : 1412-114X
e-ISSN : 2580-5649
http://ojs.pnb.ac.id/index.php/LOGIC

MOBILE APPLICATION SEARCHING OF THE SHORTEST ROUTE ON DELIVERY ORDER OF CV. ALFA FRESH WITH BRUTE FORCE ALGORITHM

1,2,3) Politeknik Negeri Sriwijaya, Jl Srijaya Negara Bukit Besar Palembang 30139, Indonesia.

Correponding email [1] :
indri@polsri.ac.id

Indri Ariyanti[1], M. Aris Ganiardi[2], Ulsa Oktari[3]

Abstract. Traveling Salesman Problem is a problem solving used in finding the shortest route to visit all nodes at once and then return to the initial node. Troubleshooting of the Traveling Salesman Problem using the Brute Force algorithm. The object of this research is the courier at CV. Alfa Fresh. The Brute Force algorithm provides a solution for Traveling Salesman Problems to select and determine the shortest routes to deliver orders from the office to the destination. Methode The Brute Force algorithm is methode an algorithm that is used to match patterns with all routes to be traversed to find the shortest route pattern. Result in reseach is the Brute Force algorithm works by enumerating all possible candidates. With this application can facilitate the courier in determining the closest route from the position of the courier. This application can also be a note for the courier to see what order he did. The brute force algorithm can give the shortest route from the courier position well, but the calculation process is a little slower because the algorithm calculates all possible distances.

Keywords : Traveling Salesman Problem, *Brute Force* Algorithm, Shortest Route.

1. INTRODUCTION

CV. Alfa Fresh is an Indonesian company that provides purchasing services for online kitchen needs such as selling fruits, vegetables, meat and other kitchen spices. CV. Alfa Fresh was founded in 2017 in the city of Palembang. This company provides Web and Mobile based applications for marketing. CV. Alfa Fresh itself has many employees with various kinds of parts, one of which is courier. Courier in CV. Alfa Fresh is in charge of delivering orders as well as to other companies, couriers who help buyers to deliver their orders who ordered online.

Couriers visit the place of the customer, where the place or destination of the courier not only one place but several places each day according to the order received. And only need to be delivered once for each customer. If there are 5 places / destinations to be visited by the courier, then there are several routes that may be passed. This causes the courier to be able to arrange the order of his visit in order to get the most effective distance possible. The problem that often occurs is that couriers sometimes forge distances that are too far and too swirling to visit one place to another, this causes couriers to be less efficient in visiting the order in which they want to go.

Resolving the problem of traveling salesman problems can be applied to visiting a number of places. To solve this problem there are various methods that can be used, one of them is by using the Brute Force algorithm. The Brute Force algorithm is an algorithm that is used to match patterns with all routes that will be passed to find

The optimal route pattern. Brute Force is a very common problem solving technique and can be used to solve problems by finding the shortest route. Brute Force works by enumerating all possible candidates, giving the best solution. The shortest route is a problem to find a path between two or more vertices in a graph weighted with the minimum number of graph side weights crossed. In weighted graphs there are optimizations that can be expressed in distances between cities, message delivery times, costs and so on. In this case the weight must be positive, although in other cases it can be negative. The shortest path with the initial vertex s and vertex t destination is defined as the shortest path from s and t with minimum weight and in the form of a simple path [1]. What is done in TSP is forming a travel route. The operator that can be used for Travelling Salesman Problem (TSP) problems is the search for sequences of all locations to select locations that have not been selected one by one so that a complete route of visit is generated from the initial location then visits all other locations exactly once and finally returns to the original location [2].

TSP is divided into 2 types, namely: (1) Asymmetrical TSP, this type of Asimetirs TSP, the cost from city 1 to city 2 is not the same as the cost from city 2 to city 1, with n cities, the size of the search space is a possible path; (2) symmetrical TSP, the type of Symmetrical TSP, the cost from city 1 to city 2 is the same as the cost from city 2 to city 1. If with n cities, the number of possible routes is a possible path.

The Brute force method is a direct approach to solving a problem, which is based on problem statements and defines the concepts involved directly. Troubleshooting using the Brute force Method is very simple, direct and clear. Characteristics of the Brute force method: (1) The number of steps needed is large, (2) Used as a basis for finding a more efficient or creative solution, (3) Almost all problems can be solved by this method, (4) Used as a basis in comparing the quality of an algorithm [3].

Exhaustive search is a brute force solution search technique for problems that involve finding elements with special properties. Usually between combinatory objects such as permutations, combinations, or subsets is part of a set [4]. The use of Brute force on TSP problems can be illustrated as follows: Given many (n) destinations in a city and the distance between each city is known to each other. Find the shortest tour through every other city only once and return to the city from departure. This TSP problem is nothing but finding a Hamilton circuit with a minimum weight. *Exhaustive search* technique steps*:* (1) Make a list (Enumeration) of every possible solution in a systematic way, (2) Evaluate each possible solution one by one, especially the mirror solution, can be issued. Then save the best solution to date (3) After the search ends, announce the best solution, (4) The resources needed in finding solutions using *Exhaustive Search* are very large, although the exhaustive technique theoretically produces a solution.

In order for the mobile application to be utilized by the courier at the company, mobile applications that can replace orders based on the customer's address is needed, then determine the order of the shortest route to be traversed from the data that has been collected. For this reason, research was carried out on Mobile Application Searching of the Shortest Route on Deliver Order of CV. Alfa Fresh with the Brute Force Algorithm. The Mobile Application is expected to help the courier officer on the CV. Alfa Fresh to determine the shortest route to visit customers so that it can shorten the distance traveled and save time.

This study focuses on the analysis, design, and development of EzDelivery which implements the Held-Karp and Fixed-Radius Near Neighbors algorithm (Castillo, Reynaldo E.Agustin, et al : 2019). The fixed-radius near neighbors algorithm is used to find the closest company because of its effectiveness in achieving mobile application functionality.

2. METHODS

2.1 Stage of Research Problems

This stage is the process of formulating problems and limiting the problems to be studied. This is needed in order to be able to better direct the researcher in making the application so that what is done is not out of the predetermined limits.

2.2 Data Collection

In this data collection stage the author refers to opinions [5], where the opinions explain the stage of data collection which is divided into two types, namely:

1. Primary Data

Primary data is data that was first recorded and collected in the study. Primary data where the author conducts a survey directly to the party concerned, namely the party that has the authority to collect the data. The author conducted interviews and observations with employees of CV. Alfa Fresh Palembang. The interview produced an obstacle faced by the CV. Alfa Fresh Palembang where the courier as the delivery order to the customer is still constrained in determining the most effective route to deliver orders to customers, so that the time of shipping orders is still less effective.

2. Secondary Data

Secondary data is data collected from existing sources. Here, here the author collects data indirectly, namely by searching for information through research journals, books, and other document sources, as a reference for the construction of mobile applications that will be made.

2.3 Research Design

In doing this final project, the system design used includes usecase diagrams, activity diagrams, class diagrams, and sequence diagrams.

2.4 Completion of the Brute Force Algorithm With Exhaustive Search Technique On Traveling Salesman Problem

An example of a case study of the application of a brute force algorithm with the Exhaustive Search technique on the Traveling Salesman Problem itself is as follows, with a starting position at CV.Alfa Fresh then the first point to Jalan Putri Rambut Selako no.4 Ilir barat 1, second point Jalan Joko No. 10 Small hill, and third point Jalan Bali Blok C3 Complex Pusri Borang Sako then return to CV. Alfa Fresh.

Table 1 List Order

Position	Latitude	Longitude
Cv. Alfa Fresh (A)	-2.9439079	104.7327026
Jalan Putri Rambut Selako No.4 Ilir barat 1 (B)	-2.9934064	104.7300526
Jalan Joko No 10 Bukit Kecil (C)	-2.9930219	104.7480037
Jalan Bali Blok C3 Komplek Pusri Borang sako (D)	-2.9252864	104.7794028

First, look for the distance between the starting positions to the destination with the euclidean distance calculation formula, namely:

$$d_{ij} = \sqrt{[(x_i - x_j)^2 + (y_i - y_j)^2]} \qquad (1)$$

Information
x_i = Coordinate x for initial point i (latitude)
y_i = Coordinate y for destination point j (longitude)
d_{ij} = Distance between two points i and j

Looking for the distance between the initial start, namely CV. Alfa Fresh to the destination Jalan Putri Rambut Selako No. 4 Ilir barat 1 (A-B) with the following formula :

$$\begin{aligned}
d_{ij} &= \sqrt{[(x_i - x_j)^2 + (y_i - y_j)^2]} \\
&= \sqrt{((-2.9439079) - (-2.9934064)^2 + ((104.7327026) - (104.7300526)^2)} \\
&= \sqrt{(0.0494985)^2 + (0.00265)^2} \\
&= \sqrt{(0.002450101502 + 0.0000070225} \\
&= \sqrt{0.002457124002} \\
&= 0.049569385733 \text{ X } \frac{111.322}{1000} \\
&= 0.0005518163159
\end{aligned}$$

Looking for the distance between the initial start, namely CV. Alfa Fresh to the destination of Jala Joko No. 10 Small hill (A-C) with the following formula:

$$\begin{aligned}
d_{ij} &= \sqrt{[(x_i - x_j)^2 + (y_i - y_j)^2]} \\
&= \sqrt{((-2.9439079) - (-2.9930319)^2 + ((104.7327026) - (104.7480037)^2)} \\
&= \sqrt{(0.049124)^2 + (0.0153011)^2}
\end{aligned}$$

$$= \sqrt{(0.002413167376 + 0.0002341236612}$$
$$= \sqrt{0.0026472910372}$$
$$= 0.0514518322045 \text{ X } \frac{111.322}{1000}$$
$$= 0.005727720865$$

Looking for the distance between the initial start, namely CV. Alfa Fresh to the destination of Jalan Bali Blok C3 Komplek Pusri Borang sako (A-D) with the following formula:

$$d_{ij} = \sqrt{[(x_i - x_j)^2 + (y_i - y_j)^2]}$$
$$= \sqrt{((-2.9439079) - (-2.9252864)^2 + ((104.7327026) - (104.7794028)^2)}$$
$$= \sqrt{(-0.0186215)^2 + (-0.0467002)^2}$$
$$= \sqrt{(0.0003467026225 + 0.00218090868004}$$
$$= \sqrt{0.00252766894229}$$
$$= 0.05027592805996 \text{ X } \frac{111.322}{1000}$$
$$= 0.00559681686349$$

Looking for the distance between the initial start, namely Jalan Putri Rambut Selako No. 4 Ilir barat 1 to the destination of Jalan Joko No.10 Bukit Kecil (B-C) with the following formula:

$$d_{ij} = \sqrt{[(x_i - x_j)^2 + (y_i - y_j)^2]}$$
$$= \sqrt{((-2.9934064) - (-2.99330319)^2 + ((104.7300526) - (104.7480037)^2)}$$
$$= \sqrt{(-0.0003745)^2 + (-0.0179511)^2}$$
$$= \sqrt{(0.00000014025025 + 0.00032224199121}$$
$$= \sqrt{0.00032238224146}$$
$$= 0.01795500602785 \text{ X } \frac{111.322}{1000}$$
$$= 0.00199878718103$$

Looking for the distance between the initial start, namely Jalan Putri Rambut Selako No. 4 Ilir barat 1 to the destination of Jalan Bali Blok C3 Komplek Pusri Borang sako (B-D) with the following formula:

$$d_{ij} = \sqrt{[(x_i - x_j)^2 + (y_i - y_j)^2]}$$
$$= \sqrt{((-2.9934064) - (-2.9252864)^2 + ((104.7300526) - (104.7794028)^2)}$$
$$= \sqrt{(-0.06812)^2 + (-0.00493502)^2}$$
$$= \sqrt{(0.0046403344 + 0.00243544224004}$$
$$= \sqrt{0.00707577664004}$$
$$= 0.08411763572545 \text{ X } \frac{111.322}{1000}$$
$$= 0.00936414344423$$

Finding the distance between the initial start, namely Jalan Joko No. 10 Small hill to the destination of Jalan Bali Blok C3 Komplek Pusri Borang sako (C-D) with the following formula :

$$d_{ij} = \sqrt{[(x_i - x_j)^2 + (y_i - y_j)^2]}$$
$$= \sqrt{((-2.9930319) - (-2.9252864)^2 + ((104.7480037) - (104.7794028)^2)}$$
$$= \sqrt{(-0.064745)^2 + (-0.0313991)^2}$$
$$= \sqrt{(0.004322405025 + 0.000985903480}$$
$$= \sqrt{0.005308308505}$$
$$= 0.07285813959332 \text{ X } \frac{111.322}{1000}$$
$$= 0.00811071381581$$

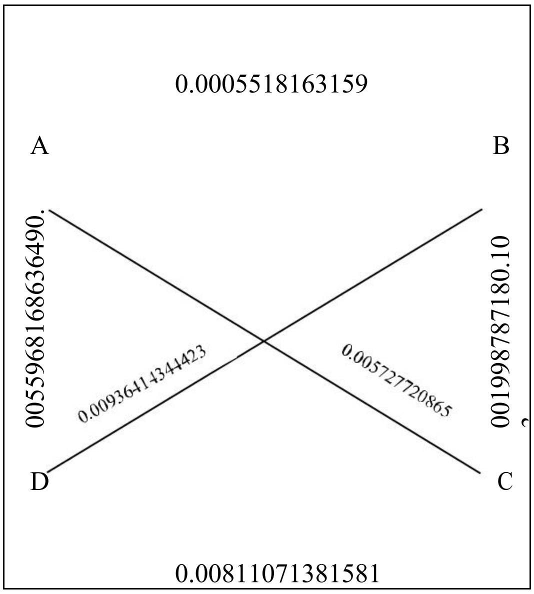


Figure 1 Illustration results of calculation of order distance

After searching for distances, we can get all the weights of all distances and we must calculate each weight, and choose the Hamilton circuit with the smallest weight.

R1 = (A,B,C,D,A) = 0.0005518163159 + 0.001998787 18103 + 0.00811071381581 + 0.00559681686349
= 0.01625813417623

R2 = (A,B,D,C,A) = 0.0005518163159 + 0.009364143 44423 + 0.00811071381581 + 0.005727720865
= 0.02375439444094

R3 = (A,C,B,D,A) = 0.005727720865 + 0.0019987871 8103 + 0.00936414344423 + 0.00559681686349
= 0.02268746835375

R4 = (A,C,D,B,A) = 0.005727720865 + 0.0081107138 1581 + 0.00936414344423 + 0.0005518163159
= 0.02375439444094

R5 = (A,D,B,C,A) = 0.00559681686349 + 0.00936414 344423 + 0.00199878718103 + 0.005727720865
= 0.02268746835375

R6 = (A,D,C,B,A) = 0.00559681686349 + 0.00811071 381581 + 0.00199878718103 + 0.0005518163159
= 0.01625813417623

So the calculation of the *exhaustive search* technique above can be done to repair the number of route searches, namely by knowing that half of the travel routes are the result of reflection of the other half route, that is by changing the direction of the route, the route is:
R1 dan R6 = 0.01625813417623
R2 dan R4 = 0.02375439444094
R3 dan R5 = 0.02268746835375
Then the route chosen based on the total distance of the smallest or closest is R1 and R6

2.5 System Development Method

If in making the application the author uses the *Brute Force* algorithm, then in developing the system the author uses the Waterfall system development method. The *waterfall* method is done in stages one by one starting from the top stage to the bottom stage. The stages involved include the analysis, design, coding, testing, and *support* phases. The *waterfall* development method was chosen because the application of steps in the *waterfall* was in accordance with the method of development carried out by the author.

a. Analysis of software requirements

The stage for analyzing the needs of external devices is the process of gathering system requirements such as Android Studio for editors coding applications; Sublime Text is useful to encode the web, MySql to store databases, PHP to encode php programs.

b. Design
The design stage is a needs analysis of system design software in use case diagrams, activity diagrams, class diagrams, and sequence diagrams.

c. Program code creation
The stage of making the program code is the process of doing coding system (Encoding Software) on Android Studio and Sumbile Text by the Programmer in accordance with the analysis and design stages that have been made before.

d. Testing
The testing phase of the author uses the Black Box method, testing is done to find out whether the system is running well and not. In conducting testing the author makes a test plan as a reference in conducting testing.

e. Support or maintenance
This maintenance stage checks the application system if there is a mistake, the application will be corrected.

3. RESULTS AND DISCUSSION

3.1 System Requirements Analysis

a. Functional Requirements

Rosa and Shalahudin [6], Explain functional requirements is a need related to product functions, for example, information systems must be able to print reports, information systems must be able to display graphics, etc.

The functional requirements of the system to be built are as follows:

1. The website and application system can authorize and authenticate users such as admin and courier.
2. The website system created can input data, edit data, and delete data in the form of admin, customer, order, order, product, user, and courier data.
3. The application system can search for the nearest route.
4. The application system can see the order history that has been completed.

b. Non Functional Requirements

Non-functional needs are additional needs that do not have inputs, processes, and outputs. However, this non-functional requirement should be fulfilled because it will determine whether this system will be used by the user or not.

The related non-functional needs are two, namely:

1. Admin websites that will be built include:
   a. The website must be accessible with a browser.
   b. The website must be accessible to the user formulated.
   c. Has a friendly user interface
   d. The website system has a password, so only registered users can access the website.

2. Courier applications to be built include:
   a. The ease of use of software can be accessed anywhere and anytime through mobile-based applications.
   b. Safety factors related to the use of the API Key (Application Programming Interface).

3.2 System Planning

a. Usecase Diagram

According to Rosa and Shalahuddin [6], use case diagram is a model for information system behavior to be created. The use case describes an interaction between one or more actors with the information system that will be created. Roughly speaking, usecase is used to find out what functions are in an information system and who has the right to use those functions.

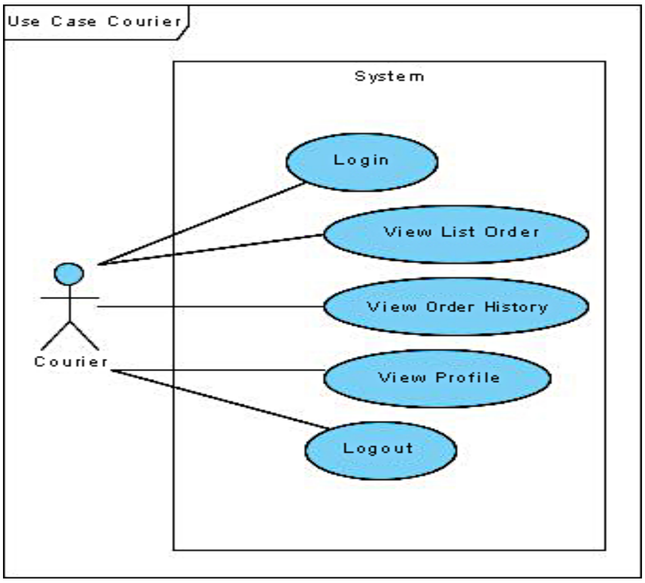


Figure 2 *Usecase* Courier Diagram

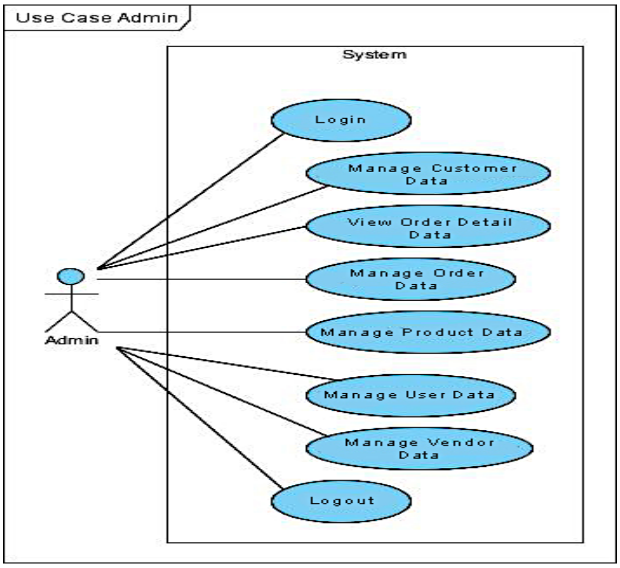


Figure 3 *Usecase* Admin Diagram

b. Class Diagram

Rosa and Shalahudin Class diagrams illustrate the system structure in terms of defining the classes that will be made to build the system [6]. Classes have what are called attributes and methods or operations. Attributes are variables owned by a class. Whereas, operations or methods are functions that are owned by a class.

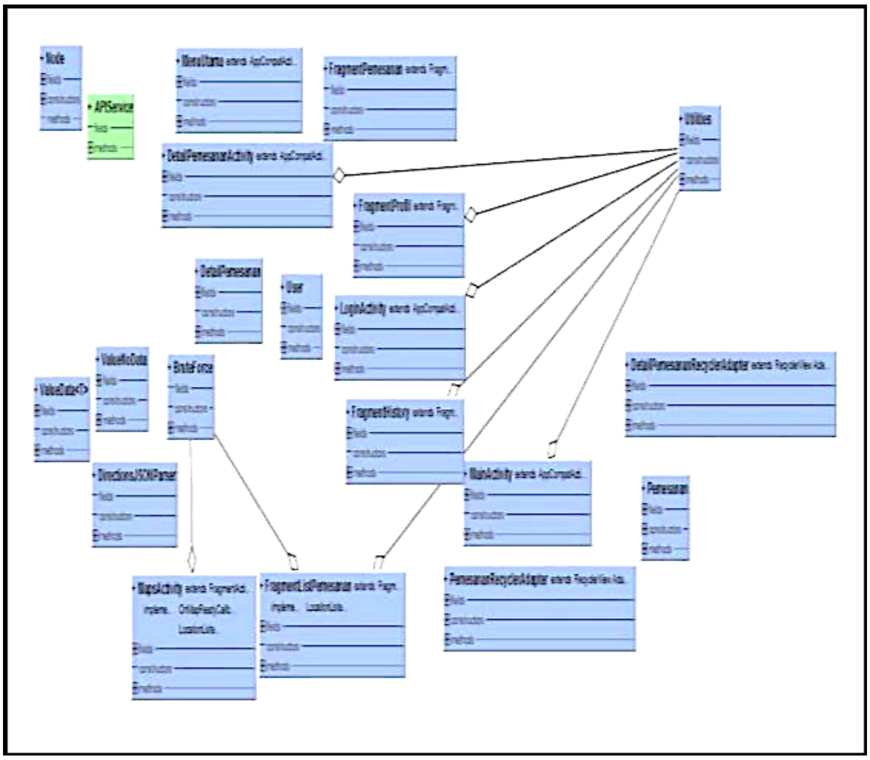

Node
APIService
MenuUtama extends AppCompatAct..
FragmentPemesanan extends Fragm..
Utilities
DetailPemesananActivity extends AppCompatAct..
FragmentProfil extends Fragm..
DetailPemesanan
User
LoginActivity extends AppCompatAct..
ValueNoData
ValueData<T>
BruteForce
DetailPemesananRecyclerAdapter extends RecyclerView.Ada..
FragmentHistory extends Fragm..
DirectionsJSONParser
MainActivity extends AppCompatAct..
Pemesanan
MapsActivity extends FragmentAct..
impleme.. OnMapReadyCallb.. LocationLis..
FragmentListPemesanan extends Fragm..
impleme.. LocationLis..
PemesananRecyclerAdapter extends RecyclerView.Ada..
fields
constructors
methods

Figure 4 *Class* Application Diagram

3.3 System Implementation

The following is the interface of the system application:

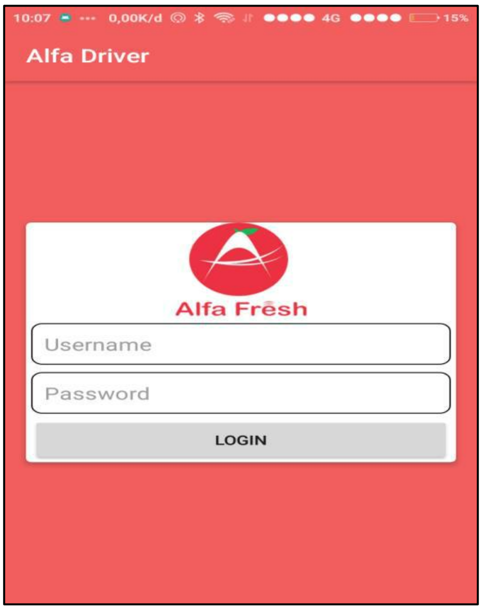
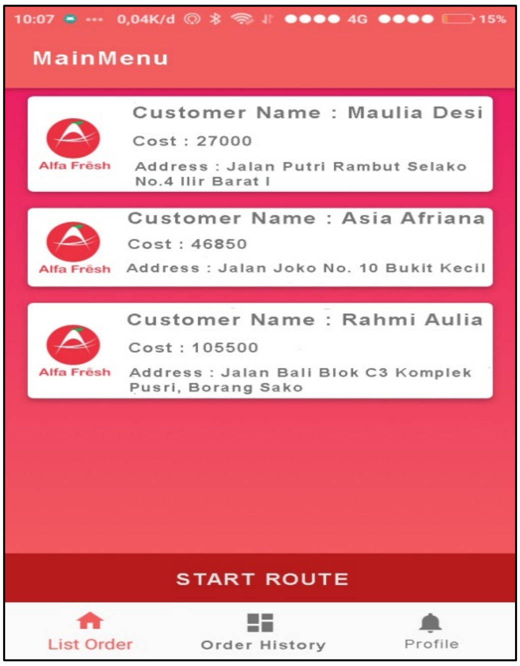


Figure 5 Application Login Page

On this page the courier logs in first to access this application by filling in the username and password then press the login button. If successful the courier login can access on the next page.



10:07 0,04K/d 4G 15%

MainMenu

Alfa Frêsh
Customer Name : Maulia Desi
Cost : 27000
Address : Jalan Putri Rambut Selako No.4 Ilir Barat I

Alfa Frêsh
Customer Name : Asia Afriana
Cost : 46850
Address : Jalan Joko No. 10 Bukit Kecil

Alfa Frêsh
Customer Name : Rahmi Aulia
Cost : 105500
Address : Jalan Bali Blok C3 Komplek Pusri, Borang Sako

Figure 6 Application List Order page

This main page functions to view the list of orders and by pressing the Access button Start the Route then the courier can find out the route of the order to be followed. In addition, this page contains 3 access panels in the form of Order Lists, order history, and profiles.

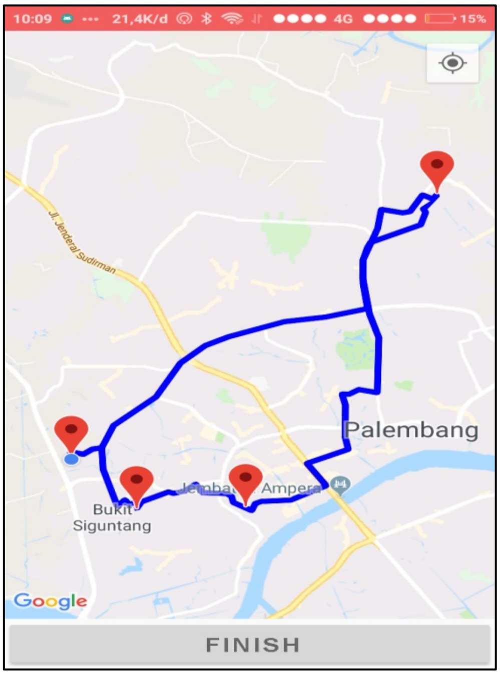
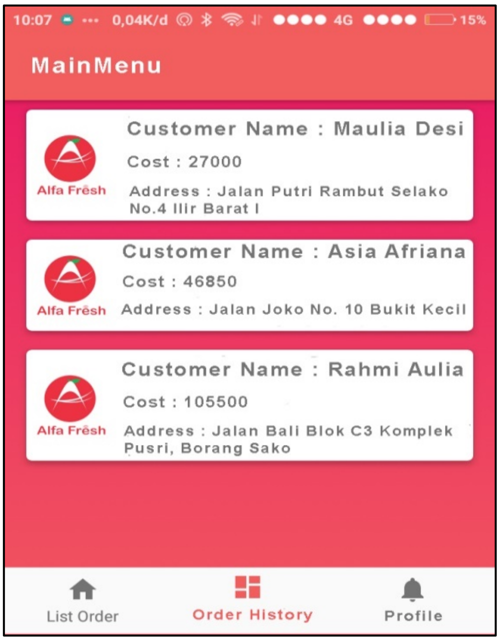


Figure 7 Start Page of the Application

It is a route map page for orders that are in the application. This page will help the courier get the shortest distance route to deliver orders and be accompanied by directions to directions such as those available in Google Maps.



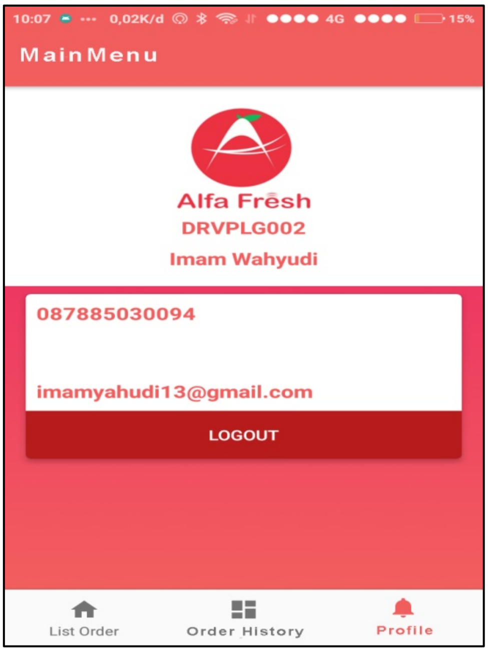
Customer Name : Maulia Desi
Cost : 27000
Address : Jalan Putri Rambut Selako No.4 Ilir Barat I

Customer Name : Asia Afriana
Cost : 46850
Address : Jalan Joko No. 10 Bukit Kecil

Customer Name : Rahmi Aulia
Cost : 105500
Address : Jalan Bali Blok C3 Komplek Pusri, Borang Sako

Figure 8 Display the order history page of the application

This page is a page that will display order history if the courier has finished delivering customer orders.



DRVPLG002
Imam Wahyudi

087885030094

imamyahudi13@gmail.com

Figure 9 Application Profile Display Page

This page is a page that will display the profile of the messenger and there is a logout button for the courier to exit the application.

4. CONCLUSION AND SUGGESTIONS

4.1 Conclusion

Based on the discussion described in the previous chapters, there are several conclusions is as follows:

a. With this application it can make easier for couriers to determine the closest route to the position of the courier.

b. This application can also be a note for the courier to see what order he did.
c. The brute force algorithm can give the shortest route from the courier position well, but the calculation process is a little slower because the algorithm calculates all possible distances.

4.2 Suggestions

As for suggestions that can be given by writer based on observations that have been done include:
a. It would be more useful if this application was added about kilometers that he had traveled to become a reminder to service the vehicle.
b. This mobile application will be better if it is equipped with feature receiver messages from the office.

5. REFERENCES

[1] Andayani Sri, Wulan Endah Perwitasari, (2014), Penentuan Rute Terpendek Pengambilan Sampah di Kota Merauke Menggunakan Algoritma Dijkstra, Seminar Nasional Teknologi Informasi dan Komunikasi Terapan 2014, ISBN: 979-26-0276-3, hal 165.

[2] Tri Dian Wiyanti, (2013), Algoritma Optimasi Untuk Menyelesaikan Travelling Salesman Problem, Jurnal Transformatika, vol 11, hal 2.

[3] Wilson, Yuliant, Izzatul, (2015), Analisis Penyelesaian Traveling Salesman Problem Dengan Metode *Brute Force* Menggunakan *Graphic Processing Unit*, e-Proceeding of Engineering, vol 2, hal 1875-1876.

[4] Munir, Rinaldi. Diklat Kuliah IF2251. *Strategi Algoritmik*

[5] Suryabrat Sumadi, (2015), Metodelogi Penelitihan, Jakarta, Rajawali Pers.

[6] A.S Rosa , dan M. Shalahuddin. (2014). *Rekayasa Perangkat Lunak Struktur dan Berorientasi Objek.* Bandung: Information.