



Analysis and design of the shortest route search application using the Dijkstra Algorithm to visit the Lurah office in Kupang City

Erna Rosani Nubatonis

Department of Informatics Engineering Study Program, Sekolah Tinggi Manajemen Informatika Komputer (STIKOM) Uyelindo Kupang, Kupang, East Nusa Tenggara, Indonesia

Abstract

Daily community mobility requires efficient travel time, especially when visiting village heads' offices spread across various areas of Kupang City. To minimize travel time, an application is needed that can display the shortest route to the destination village head office. This research aims to analyze and design a shortest route search application using the Dijkstra algorithm in order to visit village heads' offices in Kupang City. Dijkstra's algorithm is an appropriate algorithm for solving the problem of finding the shortest path in a weighted graph. In this research, the distribution of lurah offices is modeled in graph form, with nodes representing the location of lurah offices and edges representing road sections with weights in the form of travel time. The results of analysis and design using the Dijkstra algorithm show that the shortest route to visit all village heads' offices in Kupang City starts from the Oeba Village Head's Office and follows the CBADEFGHIJ route with a total travel time of 55 minutes. Dijkstra's algorithm was proven to be effective in finding the shortest route by considering travel time on each road section. By using this algorithm in the route search application, users can easily find out the fastest route to visit sub-district heads' offices, thereby saving time and increasing mobility efficiency.

Keywords: Dijkstra algorithm, application, shortest route, search, village

Introduction

Travel time really depends on the distance traveled. The closer the distance that must be traveled, the shorter the time required. Finding the shortest route is important in order to minimize travel time. By taking the shortest route, daily mobility will become more efficient. Sometimes someone doesn't know the closest route to reach their destination. In fact, if you know the closest route, travel time can be shortened. In this research, the case example taken was the village head's office in Kupang City. The village head's office is an important place for residents to take care of administrative and governmental needs of the village. The village head offices in Kupang City are spread across various regions. So there are many route options to reach the village head office according to the starting location. The distance from the starting point to the village head's office has different travel times. Sometimes residents ignore the nearest route to the village head's office. Therefore, the author created an application to find the shortest route to the village head office using the Dijkstra algorithm. This application is web based. This algorithm will calculate and determine the shortest route from the initial location to the destination village head office.

Theory

Dijkstra's Algorithm Dijkstra's algorithm is an algorithm used to solve the shortest path problem in weighted graphs. This algorithm was discovered by computer scientist Edsger Dijkstra in 1959. In the context of finding the shortest route to the village head's office, Dijkstra's algorithm can be used to find the path with the shortest distance from an initial location to the destination headman's office. By modeling the village head office as a point and the road as a line, this algorithm performs

calculations on each possible path to determine the route with the smallest weight or distance. The working principle of the Dijkstra algorithm is to search systematically from the starting point to the goal by considering the smallest weight at each step. The algorithm ensures that each point passed has the shortest distance from the start. When reaching the destination, the resulting route is the shortest route from start to finish.

Methodology

The framework for this research can be described as follows

1. Study of literature

At this stage, a search and study of relevant theories is carried out from various references such as books, journals, and other sources related to shortest route search algorithms, Dijkstra's algorithm, graph modeling, and geographic information systems.

2. Data collection

The data collection process was carried out by conducting interviews with related parties, such as the Kupang City Government Service and the Village Head's Office, to obtain information regarding the number, location and distribution of village head offices in Kupang City. Apart from that, direct observations and observation were also carried out to obtain spatial data related to the geographical location and road network in Kupang City.

3. Systems Analysis

At this stage, problems were identified related to finding the shortest route to subdistrict heads' offices in Kupang City, as well as an analysis of the needs of the system to be developed. This analysis includes determining the required features, spatial data

modeling, and implementing the Dijkstra algorithm in the system.

4. System planning

At the design stage, system modeling is carried out using Unified Modeling Language (UML), user interface design, data input design, and system output design. Apart from that, a spatial database was also designed to store data on the location of village heads' offices and road networks.

5. Implementation and Testing

At this stage, the program code is written and the shortest route search system is implemented using the Dijkstra algorithm. The programming language used can be adjusted to suit your needs, such as PHP, Python, or Java. Apart from that, testing is also carried out on the system that has been created, both using black-box testing methods and other tests, to ensure the system runs according to needs and expectations.

Results and discussion

Dijkstra Method Analysis

The Dijkstra algorithm is used to determine the shortest route or shortest path based on certain criteria that are used as boundaries.

For example:

$$V(G) = \{v_1, v_2, \dots, v_n\}$$

$L = \text{Set dot, dot, dot } \in V(G)$ which are already selected in track path shortest.

$D(j) = \text{Amount weight path smallest from } v_1 \text{ to } v_j$. $w(i,j) = \text{Weight of the line from point } v_i \text{ to } v_j$.

$w^*(1,j) = \text{Amount weight path smallest from } v_1 \text{ to } v_j$
Formally, *Dijkstra's algorithm* for finding the shortest path is as follows:

1. $L = \{ \}; V = \{v_2, v_3, \dots, v_n\}$.
2. For $i = 2, \dots, n$, do $In) - w(1, i)$
3. During $v_n \notin L$ do:
 - a. Select the point $v_k \in V - L$ with the smallest $D(k)$. $L = L \cup \{v_k\}$
 - b. For each $v_j \in V - L$ do: If $D(j) > D(k) + W(k,j)$ then replace $D(j)$ with $D(k) + W(k,j)$

4. For every $v_j \in V, w^*(1, j) = D(j)$

This algorithm aims to find the shortest path based on the smallest weight from one point to another. The dots will depict each subdistrict office and the lines will depict the time taken. The following is an example of a case of determining the closest route to visit all subdistrict offices in Kupang City.

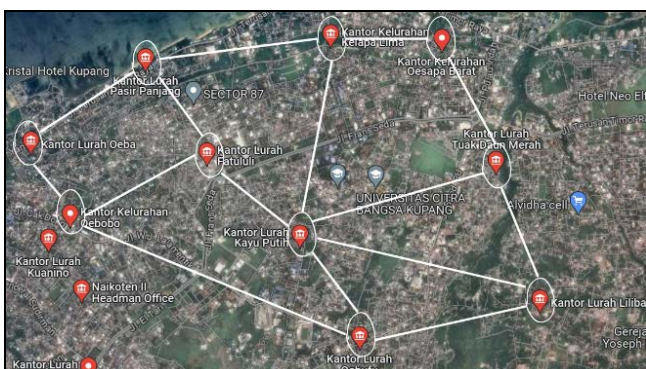


Fig 1: Example route visiting all subdistrict offices in Kupang City

The steps for applying the Dijkstra algorithm in determining the shortest route are as follows:

1. Determine the node point of the existing corner.
2. Determine the point that will be the starting node.
3. Determine the point that will be the end node.
4. Give the weight of the time taken at the first node to the nearest nodes one by one.
5. Dijkstra will develop the search from one point to another and to the next point step by step.

Below is the logical sequence of the Dijkstra algorithm:

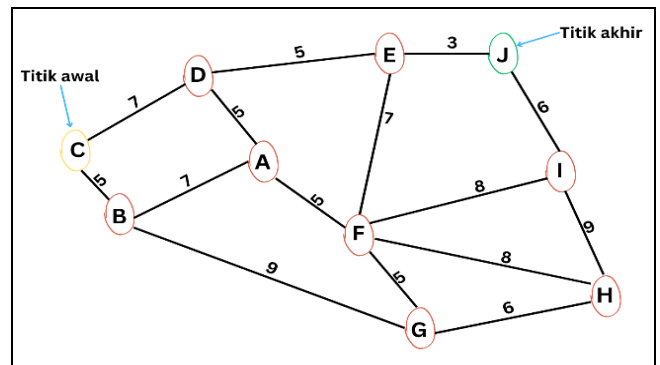


Fig 2: Case example of Dijkstra's algorithm

Information

Each node or point will represent each village head office.

- Node A: Fatululi Village Head Office
- Node B: Oebobo Village Head Office
- Node C: Oeba Village Head Office
- Node D: Pasir Panjang Village Head Office
- Node E: Kelapa Lima Village Head Office
- Node F: Kayu Putih Village Head Office
- Node G: Oebufu Village Head Office
- Node H: Liliba Village Head Office
- Node I: TDM Village Head Office
- Node J: West Oesapa Village Head Office

Each connected edge will have a value which is the travel time taken to pass from one node to another node.

The following is a detailed step-by-step explanation of the shortest path from the start node to the end node. Start node C, end node J. Each connected edge has been given a value

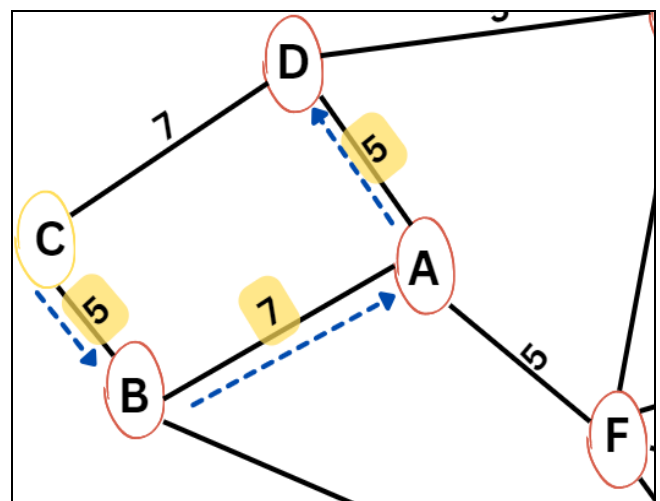


Fig 3: Case example of Dijkstra's algorithm – Step 4
Node D is used as a connecting node, Dijkstra performs calculations by finding connected nodes, namely node C and node E. By determining node E will be the shortest route with a weight value of 22 (17+5).

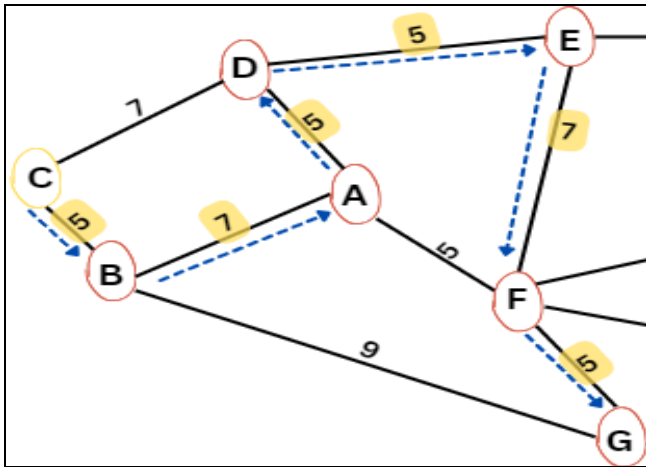


Fig 4: Case example of Dijkstra's algorithm – Step 7

Next, node G is the connecting node and Dijkstra searches for the closest route and finds nodes B and H which are connected to node G. It can be seen that the smallest weight between the two nodes is at node B with a weight value of 40 (34+6).

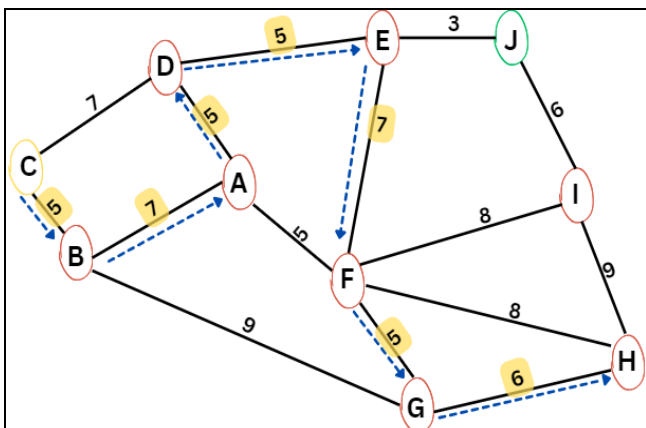


Fig 5: Case example of Dijkstra's algorithm – Step 8

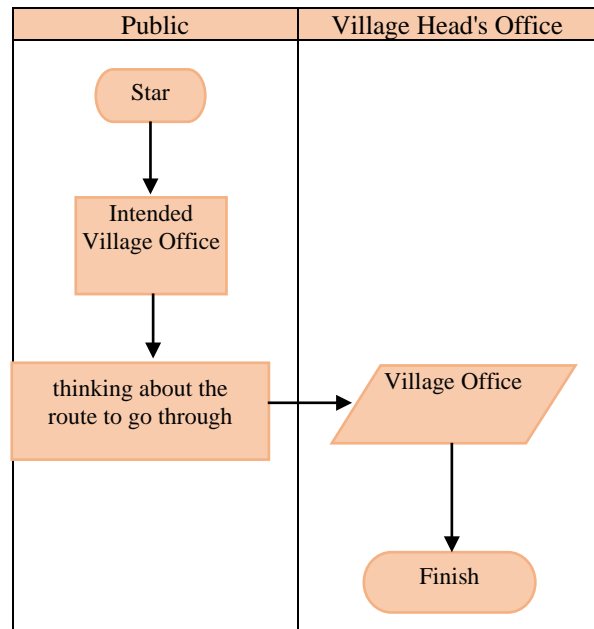
Next, node H is the connected node and the nodes that are directly connected are nodes F, G, and I. Dijkstra carries out calculations and node I is selected with a weight value of 49 (40+9).

Existing System Analysis, Searching for the shortest route plays an important role in daily life, including visiting village heads' offices in Kupang City. Determining the shortest route must be done quickly so that people can immediately know which route is the most efficient to take. By taking the shortest route, people's mobility in visiting the village head office will become more efficient. If someone already knows the nearest route, travel time can be shortened.

The following is an analysis of the existing system:

1. The community starts by deciding which subdistrict office to go to.
2. Next, people will think about and determine the route they will take manually based on their own knowledge.
3. After that, the public will follow a predetermined route to arrive at the destination village head office.

Table 1: Existing System Analysis



System planning, The system design was created for improve the system that is currently running and make it easier to create application programs that will be created. By designing the proposed system you will see the differences between system Which long with system Which new, where the old system was not very effective and efficient. Following is analysis system Which There is:

1. Users start with enter to project menu.
2. Choose village head's office objective And put user icon on the Kupang city map display.
3. Click the search button to get route to the selected village head office.
4. The system will process the route in the data base.
5. Next, the route to the village head's office will be displayed in the program.
6. If want to replace village head's office destination, select the destination subdistrict office again and place the user icon, then click the search button.
7. Otherwise, follow the route that appears in the program.

The results of this implementation are carried out so that system users can understand how this system works. The following is the implementation of the system for determining the shortest route to visit each village head office: Website Main Page, The main interface page of the Shortest Route Search Application website uses the Dijkstra algorithm to visit the Lurah Office in Kupang City.



Fig 6: Main page

Admin Login Page, The admin login page is found on the page that contains the admin login to enter the administrator page so that you can access all the required data management, the following is the admin login page:

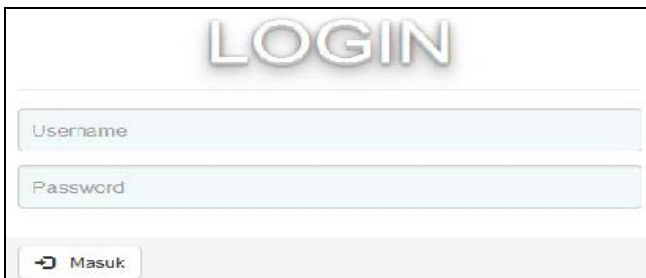


Fig 7: Admin Login Page

Admin Page, Contains admin data processing menu and village head office data processing.



Fig 8: Admin Page

Admin Data Page

The admin data page contains admin data input, adding, editing and deleting admin data. System Testing, Testing is part which is important in cycle development device soft. Testing done for ensure quality and also know weakness from device soft. Objective from testing is ensure that the software built has good quality reliable that is capable interpret basic study of specifications, analysis, design and coding of the software itself criteria testing the intended to find out the correctness of the software and suitability with the specifications declared successful if capable application operate functions based on specification in a way correct. On the contrary testing stated fail if

specifications which cannot be executed in the testing process. The tests that have been carried out are: process which there is in analysis and design of the shortest route search application using the Dijkstra algorithm to visit village head offices in Kupang City. After done testing, then based on results from testing can be taken conclusion that application which has been made to be usable properly.

Conclusion

From the research that has been carried out, there are several main conclusions:

1. The shortest route search application is very important in helping to increase the effectiveness and efficiency of people's mobility in everyday life. By displaying the closest route to a destination location, this application can shorten travel time.
2. Dijkstra's algorithm is proven to be effective for implementation in shortest route search applications. This algorithm works by conducting a systematic search from the initial location to the destination, taking into account the smallest weight or distance at each step.
3. Based on existing data, the results obtained to determine the shortest route to visit each subdistrict office in Kupang City take the shortest time, namely 55 minutes.
4. In designing the application, the distribution of destination locations is modeled in the form of a graph, where each location is represented by a node and the road segments connecting it are represented by edges with a weight in the form of travel time.
5. The results of analysis and design using the Dijkstra algorithm are able to find the shortest
6. route that connects all destination locations by considering the minimum total travel time.

Reference

1. Ginting JY, *et al.* Application for Determining the Route to the Nearest Village Head Office Using the Dijkstra Algorithm. STMIK Kristen Neuman Indonesia Medan, 2018.
2. Lakutu NF. Dijkstra's Algorithm and Greedy Algorithm for Optimizing Goods Delivery Routes at the Gorontalo Post Office. Gorontalo State University, 2023.
3. Gunawam F, *et al.* Shortest Route Search Application for Tourist Attractions in Pekanbaru City Using Floyd Warshall. Pekanbaru, 2020.
4. Gatra CR, *et al.* Determining Optimal Routes for Goods Delivery Services Using Genetic Algorithms. Majapahit Islamic University, 2023.
5. Cantona A, *et al.* Implementation of the Dijkstra Algorithm in Finding the Shortest Route to Museums in Jakarta. National University, 2020.
6. Taufiq M *et al.* Determining the Shortest Route Using a Heuristic Method Based on the A* Algorithm. Semarang State University, 2019.
7. Gautama IPW, Hermato K. Determining the Shortest Route Using Dijkstra's Algorithm on Paths School Bus. Sumbawa University of Technology, 2020.
8. Parapat MD, *et al.* Design and Development of the Shortest Route Application for Mobile-Based Goods Delivery Services using the Dijkstra Algorithm Method. Malang State Polytechnic, 2017.