

Implementasi Metode Dijkstra Untuk Mencari Jarak Terdekat Dalam Melakukan Pengiriman Paket Di PT.PMH

Hermawan Susanto^{*}, Milfa Yetri^{**}, Ita Mariami^{**}

^{*}Program Studi Sistem Informasi, STMIK Triguna Dharma

^{**}Program Studi Sistem Informasi, STMIK Triguna Dharma

Article Info

Article history:

Received Oct 12th, 2020

Revised Oct 20th, 2020

Accepted Oct 30th, 2020

Keyword:

Sistem Pendukung Keputusan,
Dijkstra, Rute

ABSTRACT

Perusahaan yang bergerak dalam pengiriman paket barang sangatlah banyak, yang mengakibatkan persaingan antara perusahaan itu semakin ketat. Sehingga pemilik dari PT. PMH ingin mengembangkan atau meningkatkan pelayanan yang mereka miliki untuk kepuasan konsumen terkait ketepatan waktu pengiriman paket barang. Adapun gagasan ini didapatkan oleh pemilik dikarenakan adanya penurunan customer atau pelanggan yang ingin mengirimkan paket barang. Hal ini terjadi, dikarenakan pernah terjadinya untuk keterlambatan pengiriman paket barang sehingga mendapatkan komplain dari pelanggan.

Berdasarkan permasalahan ada sebuah bidang ilmu pengetahuan sistem pendukung keputusan dengan Algoritma Dijkstra, dimana algoritma ini dapat membantu dalam pemilihan rute terpendek sehingga dapat meminimalisir keterlambatan pengantaran paket barang yang selama ini terjadi pada PT. PMH.

Hasil yang didapatkan adalah rute yang dapat dijadikan alternatif pilihan untuk proses pengiriman paket barang untuk menghindari keterlambatan pengiriman.

Copyright © 2020 STMIK Triguna Dharma.
All rights reserved

First Author

Nama: Hermawan Susanto

Kampus :STMIK Triguna Dharma

Program Studi : Sistem Informasi

E-Mail : Shermawan397@gmail.com

1. PENDAHULUAN

1.1 Latar Belakang

Persatuan Motor Horas atau PMH adalah perusahaan dibidang jasa angkut manusia maupun pengiriman barang. Perusahaan ini didirikan pada tahun 1961 di Parapat Sumatera Utara. Perusahaan PT PMH ini didirikan oleh tiga orang yaitu Bapak Justin Sirait, Bapak Nelson dan Bapak Samosir. Pada awal merintisnya trayek jalur hanya Parapat-Siantar. Setelah beberapa tahun kemudian PMH melebarkan sayapnya dengan mengambil jalur trayek hingga keluar kota. Trayek ini tidak hanya mengangkut manusia, tetapi juga untuk pengiriman paket barang.

Saat ini, perusahaan yang bergerak dalam pengiriman paket barang sangatlah banyak, yang mengakibatkan persaingan antara perusahaan itu semakin ketat. Sehingga pemilik dari PT. PMH ingin mengembangkan atau meningkatkan pelayanan yang mereka miliki untuk kepuasan konsumen terkait ketepatan waktu pengiriman paket barang. Adapun gagasan ini didapatkan oleh pemilik dikarenakan adanya penurunan customer atau pelanggan yang ingin mengirimkan paket barang. Hal ini terjadi, dikarenakan pernah terjadinya untuk keterlambatan pengiriman paket barang sehingga mendapatkan komplain dari pelanggan.

Dalam perkembangan di Era Digitalisasi saat ini, perkembangan teknologi sangat pesat sehingga membuat persaingan diantara perusahaan salah satunya perusahaan dibidang logistic atau perusahaan pengiriman barang sangat ketat. Salah satu layanan yang diharapkan oleh masyarakat atau pengguna jasa adalah adanya ketepatan waktu pengiriman, kenyamanan dan keamanan pengiriman. Oleh karena itu, PT PMH harus dapat melihat kedepan untuk menjaga kestabilan perusahaan terhadap persaingan. Maka dari itu sangat dibutuhkan langkah-langkah yang harus diambil untuk mempercepat proses pengiriman barang untuk pelayanan yang baik dengan cara mencari rute terpendek.

Berdasarkan pada latar belakang dan uraian di atas maka dilakukan penelitian pada PT PMH yang mendasari untuk membuat sebuah sistem pendukung keputusan dengan judul **“IMPLEMENTASI METODE DIJKSTRA UNTUK Mencari Jarak Terdekat Dalam Melakukan Pengiriman Paket di PT.PMH”**.

2. TINJAUAN PUSTAKA

2.1 Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) adalah sistem berbasis *computer* yang interaktif, yang membantu pengambilan keputusan memanfaatkan data dan model untuk menyelesaikan masalah-masalah yang tak terstruktur dan semi terstruktur”. Sebenarnya definisi awalnya, SPK adalah sistem berbasis model yang terdiri dari prosedur-prosedur dalam pemrosesan data dan pertimbangannya untuk membantu manajer dalam mengambil keputusan [5].

Sistem pendukung keputusan sebagai sekumpulan *tools* komputer yang terintegrasi yang mengijinkan seorang *decision maker* untuk berinteraksi langsung dengan komputer untuk menciptakan informasi yang berguna dalam membuat keputusan semi terstruktur dan keputusan tak terstruktur yang tidak terantisipasi [6].

Mendefinisikan sistem pendukung keputusan merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang terstruktur, semiterstruktur dan situasi tidak terstruktur, dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat[7].

2.2 Metode Dijkstra

Algoritma *Dijkstra* adalah salah satu algoritma yang digunakan untuk menyelesaikan masalah jarak terpendek (*shortestpath problem*) pada sebuah graf yang terarah (*directed graph*). Pada algoritma *Dijkstra* juga termasuk dalam algoritma untuk masalah pencarian graf yang mampu menuntaskan masalah mencari lintasan terpendek dengan satu sumber pada sebuah graf yang digunakan untuk menyelesaikan masalah lintasan terpendek dengan satu sumber pada sebuah graf yang tidak memiliki *cost* sisi negatif, dan menghasilkan sebuah alur lintasan terpendek. Misalkan sebuah gedung dan monumen dijadikan titik dan jalanan dijadikan garis, maka algoritma *dijkstra* akan melakukan perhitungan terhadap semua garis dengan bobot terkecil dari setiap titik[9].

Di dalam melakukan perhitungan dengan algoritma *Dijkstra* ada beberapa skema umum yang digunakan pada pencarian jarak terpendek, antara lain:

1. Membuat 3 buah *list*, yaitu *list* jarak (*list* 1), *list* simpul-simpul sebelumnya (*list* 2) dan *list* yang sudah dikunjungi (*list* 3), serta sebuah variabel untuk menampung simpul pada saat ini (*current vertex*).
2. Dalam *list* jarak, diisi dengan nilai tak hingga, kecuali simpul awal yang diisi dengan nilai 0.
3. Pada *list* 2, diisi dengan *false*.
4. Pada *list* 3, diisi dengan *null*.
5. *Current vertex* diisi dengan simpul awal (*start*).
6. Menandai *current vertex* sebagai simpul yang telah dikunjungi.
7. *Update list* 1 dan 2 berdasarkan simpul-simpul yang dapat langsung dicapai dari *current vertex*.
8. *Update current vertex* dengan simpul yang paling dekat dengan simpul awal.
9. Lakukan langkah no 6 sampai semua simpul telah dikunjungi.

Rumusan Algoritma *Dijkstra*:

$V(G) = \{V_1, V_2, V_3, \dots, V_n\}$

$L =$ Himpunan *node* $V(G)$ yang sudah terpilih dalam jalur terpendek.

$D(j) =$ Jumlah bobot jarak terkecil dari V_1 ke V_j

$W(i, j) =$ Jumlah bobot jarak terkecil dari V_1 ke *node* V_j

$W^*(1, j) =$ Jumlah bobot jarak terkecil dari V_1 ke V_j

Secara formal, algoritma *Dijkstra* untuk mencari jarak terpendek adalah:

$L = \{ \}$

$V = \{V_1, V_2, V_3, \dots, V_n\}$

Untuk $I = 2, 3, \dots, n$, lakukan $D(i) = W(1, i)$

Selama $V_n \geq L$ lakukan :

Pilih node V_k V-L dengan $D(k)$ terkecil $L = L \cup \{V_k\}$

Untuk setiap v_j V-L lakukan:

Jika $D(j) > D(k) + Q(k,j)$ maka ganti $D(j)$ dengan $D(k)+Q(k,j)$

Untuk setiap $V_j \leq V$, $w^*(1,j)=D(j)$.

Sebagai contoh perhitungan adalah sebagai berikut : rute pembacaan water meter pada rute 100 yaitu dari *node* 1 ke *node* 17 dimana *node-node* yang kemungkinan dilewati adalah *node* 2 dengan jarak 0.27 km, *node* 3 dengan jarak 0.49 km, *node* 4 dengan jarak 0.52 km dan *node* 18 dengan jarak 0.66 km. Pada aplikasi dihasilkan bahwa rute terpendek dari *node* 1 ke *node* 17 adalah 1-2-3-8-17 dengan total jarak 2.09 km.

Hasil perhitungan algoritma *Dijkstra* adalah sebagai berikut:

1. $L = \{1\}$
 $d[2] = 0.27$
 $d[3] = \infty$
 $d[4] = \infty$
 $d[18] = \infty$
 $d[17] = \infty$

Rute terpendek yang terhubung dengan *node* 1 adalah *node* 2 karena diantara yang lain memiliki bobot jarak terpendek.

2. Iterasi 1

$$W = \min\{2,3,4,18,17\}=2$$

$$L = \{1,2\}$$

$$D[3] = \min(d[3], d[2]+c[2,3]) = \min(\infty, 0.27+0.49) = 0.76$$

$$D[4] = \min(d[4], d[2]+c[2,4]) = \min(\infty, 0.27+\infty) = \infty$$

$$D[18] = \min(d[18], d[2]+c[2,18]) = \min(\infty, 0.27+\infty) = \infty$$

$$D[17] = \min(d[17], d[2]+c[2,17]) = \min(\infty, 0.27+\infty) = \infty$$

Rute selanjutnya dari *node* 2 adalah *node* 3, karena merupakan *node* yang memiliki bobot jarak terpendek dari *node* yang lain.

3. Iterasi 2

$$W = \min\{3,4,18,17\}=3$$

$$L = \{1,2,3\}$$

$$D[4] = \min(d[4], d[3]+c[3,4]) = \min(\infty, 0.76+0.52) = 1.28$$

$$D[18] = \min(d[18], d[3]+c[3,18]) = \min(\infty, 0.76+0.66) = 1.42$$

$$D[17] = \min(d[17], d[3]+c[3,17]) = \min(\infty, 0.76+\infty) = \infty$$

Rute selanjutnya yang terhubung dengan *node* 3 adalah *node* 4 karena memiliki bobot jarak terpendek.

4. Iterasi 3

$$W = \min\{4,18,17\}=4$$

$$L = \{1,2,3,4\}$$

$$D[18] = \min(d[18], d[4]+c[4,18]) = \min(\infty, 1.28+0.92) = 2.2$$

$$D[17] = \min(d[17], d[4]+c[4,17]) = \min(\infty, 1.28+\infty) = \infty$$

Node yang terhubung dari *node* 4 adalah *node* 18, namun bobot jarak yang dimiliki *node* 18 pada iterasi 3 lebih besar dari bobot jarak dari *node* 3 ke *node* 18 pada iterasi 2, untuk itu maka yang dipilih adalah bobot jarak pada iterasi 2 sehingga hubungan yang terjadi selanjutnya adalah dari *node* 3 ke *node* 18.

5. Iterasi 4

$$W = \min\{18,17\}=18$$

$$L = \{1,2,3,18\}$$

$$D[17] = \min(d[17], d[18]+c[18,17]) = \min(\infty, 2.2+0.67) = 2.87$$

Node selanjutnya yang terhubung dari *node* 18 adalah *node* 17. Karena *node* 17 merupakan *node* tujuan maka iterasi diakhiri

6. Iterasi 5

$$W = \min\{17\}=17$$

$$L = \{1,2,3,18,17\}$$

Hasil akhir didapat bahwa rute terpendek dari *node* 1 ke *node* 17 adalah : 1-2-3-18-17

2.3 Shortest Path Problem (Masalah Jalur Terpendek)

Jalur pendek adalah suatu jaringan pengarah perjalanan dimana seseorang pengarah jalan ingin menentukan jalur terpendek antara dua perusahaan, berdasarkan beberapa jalur alternatif yang tersedia, dimana titik tujuan hanya satu[12].

Kata Terpendek pada persoalan sebuah lintasan memiliki pengertian yaitu proses minimalisasi bobot pada sebuah lintasan graph. Beberapa jenis persoalan lintasan terpendek antara lain [13]:

- Lintasan terpendek antara dua buah simpul.
- Lintasan terpendek antara semua pasangan simpul.
- Lintasan terpendek dari simpul tertentu ke semua simpul yang lain.
- Lintasan terpendek antara dua buah simpul yang melalui beberapa simpul tertentu.

2.4 UML (Unified Modeling Language)

Unified Modeling Language (UML) bukanlah suatu proses melainkan bahasa pemodelan secara grafis untuk menspesifikasikan, memvisualisasikan, membangun dan mendokumentasikan seluruh artifak sistem perangkat lunak. Penggunaan model ini bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem yang dibahas dan bagaimana hubungan antara sistem dengan subsistem maupun sistem lain di luarnya[14].

3. METODOLOGI PENELITIAN

3.1 Metode Penelitian

Dalam suatu penelitian, harus ada metode yang digunakan untuk mengumpulkan sumber informasi atau data yang dibutuhkan dalam penelitian berdasarkan permasalahan yang dibahas. Dalam penelitian ini, untuk mendapatkan sumber informasi dan data penelitian dilakukan dengan metode seperti berikut:

1. Pengumpulan Data

Dalam proses pengumpulan data dalam penelitian ini dilakukan dengan 2 teknik untuk mengumpulkan data, yaitu :

a. Observasi

Observasi ini dilakukan dengan cara melakukan riset secara langsung ke lokasi PMH dan bertemu langsung kepada salah satu penanggung jawab di PMH tersebut, untuk keperluan pengumpulan data.

b. Wawancara

Dalam proses observasi (tinjauan langsung ke lapangan) peneliti melakukan wawancara kepada salah satu penanggung jawab di PMH tersebut. Dalam wawancara penelitian, dilakukan dengan mencari informasi atau data yang dibutuhkan saja dalam penelitian.

2. Studi Literatur

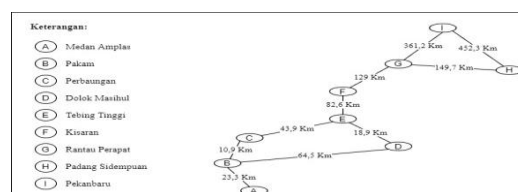
Dalam studi literatur pada penelitian ini dengan cara mengumpulkan data atau referensi untuk mendukung penelitian ini dengan mencari beberapa jurnal dan buku yang berkaitan dengan masalah dalam penelitian ini. Dalam penggunaan referensi untuk penelitian ini yaitu menggunakan 16 jurnal dan 1 Buku.

3. Data Penelitian

Dari hasil observasi, wawancara dan studi literatur yang telah dilakukan. Maka didapatkan data untuk kebutuhan penelitian ini yaitu jalur yang dilalui oleh Bus PMH dan jarak tempuh dari satu titik awal ke titik tujuan dari jalur PMH tersebut. Berikut ini adalah hasil *graph* untuk jalur lalu lintas yang dilalui oleh Bus PMH dalam pengiriman Barang.

Tabel 3.1 Data Customer Pengiriman Barang

No.	Nama Customer	Alamat	Tujuan Pengiriman
1	Jasmirin	Jl. Bandrek Kecamatan Patumbak	Pekanbaru
2	Adam Wijaya Hasibuan	Jl. Benteng Kecamatan Deli Tua	Pekanbaru
3	Arif Rahman Hakim	Jl. S.M Raja Kecamatan Medan Amplas	Pekanbaru



Gambar 3.1 *Graph* Jalur PMH

Untuk keterangan gambar, jarak yang didapatkan berdasarkan usulan dari penanggung jawab PT. PMH untuk masalah pengiriman barang, jarak tersebut didapatkan dari ketentuan jarak dari *Google Map* sedangkan untuk jalur berdasarkan ketentuan dari penanggung jawab PT. PMH.

Tabel 3.2 Inisialisasi *Node Tempat*

No.	<i>Node Tempat</i> Alamat	Inisialisasi Kode <i>Node</i>
1	Medan Amplas	A
2	Pakam	B
3	Perbaungan	C
4	Dolok Masihul	D
5	Tebing Tinggi	E
6	Kisaran	F
7	Rantau Perapat	G
8	Padang Sidempuan	H
9	Pekanbaru	I

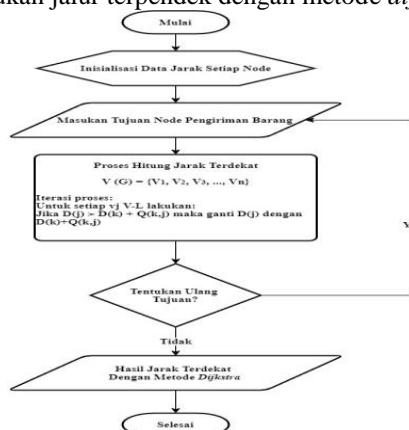
Berdasarkan Gambar *Graph* diatas, maka dapat dijelaskan jarak jalur pada Bus PMH adalah sebagai berikut:

1. Jarak Medan Amplas (A) ke Pakam (B) adalah 23,3 Km
2. Jarak Pakam (B) ke Perbaungan (C) adalah 10,9 Km
3. Jarak Pakam (B) ke Dolok Masihul (D) adalah 64,5 Km
4. Jarak Perbaungan (C) ke Tebing Tinggi (E) adalah 43,9 Km
5. Jarak Dolok Masihul (D) ke Tebing Tinggi (E) adalah 18,9 Km
6. Jarak Tebing Tinggi (E) ke Kisaran (F) adalah 82,6 Km
7. Jarak Kisaran (F) ke Rantau Perapat (G) adalah 129 Km
8. Jarak Rantau Perapat (G) ke Padang Sidempuan (H) adalah 149,7 Km
9. Jarak Rantau Perapat (G) ke Pekanbaru (I) adalah 361,2 Km
10. Jarak Padang Sidempuan (H) ke Pekanbaru (I) adalah 452,3 Km

3.3 Algoritma Sistem

3.3.1 Flowchart Metode Dijkstra

Flowchart metode *dijkstra* merupakan bagan alur penggambaran urutan proses (intruksi) secara detail untuk alur sistem menentukan jalur terpendek dengan metode *dijkstra* yaitu sebagai berikut:

Gambar 3.3 *Flowchart* Metode *Dijkstra*

3.3.2 Analisa Permasalahan

Analasia permasalahan pada penelitian ini adalah bagaimana menerapkan metode *dijkstra* untuk menemukan jalur terpendek dari kasus berikut ini: Bus PMH mendapatkan customer untuk mengirimkan barang ke Pekanbaru. Dari data kasus, untuk melakukan pengiriman ke Pekanbaru maka akan melalui beberapa jalur yang dapat dilalui dari titik awal Bus PMH di Medan Amplas yaitu dengan menggunakan rumus berikut :

Rumusan Algoritma *Dijkstra*:

$$V(G) = \{V1, V2, V3, \dots, Vn\}$$

L = Himpunan *node* V(G) yang sudah terpilih dalam jalur terpendek.

D(j) = Jumlah bobot jarak terkecil dari V1 ke Vj

$W(i,j)$ = Jumlah bobot jarak terkecil dari V_1 ke *node* V_j

$W^*(1,j)$ = Jumlah bobot jarak terkecil dari V_1 ke V_j

Secara formal, algoritma *Dijkstra* untuk mencari jarak terpendek adalah:

$L = \{ \}$

$V = \{V_1, V_2, V_3, \dots, V_n\}$

Untuk $I = 2, 3, \dots, n$, lakukan $D(i) = W(1, i)$

Selama $V_n \geq L$ lakukan :

Pilih *node* $V_k \in V-L$ dengan $D(k)$ terkecil $L = L \cup \{V_k\}$

Untuk setiap $v_j \in V-L$ lakukan:

Jika $D(j) > D(k) + Q(k, j)$ maka ganti $D(j)$ dengan $D(k) + Q(k, j)$

Untuk setiap $V_j \in V, w^*(1, j) = D(j)$.

Dengan rumus diatas, maka akan dilakukan perhitungan untuk mencari jalur terpendek dari Medan Amplas (A) ke Pekanbaru (I) berdasarkan gambar *graph* pada gambar 3.1 maka kemungkinan dilewati adalah *node* B (Pakam) dengan jarak 23,3 Km, *node* C (Perbaungan) dengan jarak 10,9 Km, *node* D (Dolok Masihul) dengan jarak 64,5 Km, *node* E (Tebing Tinggi) dengan jarak 43,9 Km dan 18,9 Km, *node* F (Kisaran) dengan jarak 82,6 Km, *node* G (Rantau Perapat) dengan jarak 129 Km, *node* H (Padang Sidempuan) dengan jarak 149,7 Km dan terakhir *node* I dengan jarak 361,2 Km dan 452,3 Km. Berdasarkan *node* tersebut maka dapat kita ketahui jalur terpendek dengan perhitungan algoritma *dijkstra* sebagai berikut:

1. $L = \{A\}$

$d[B] = 23,3$

$d[C] = \infty$

$d[D] = \infty$

$d[E] = \infty$

$d[F] = \infty$

$d[G] = \infty$

$d[H] = \infty$

$d[I] = \infty$

Rute terpendek yang terhubung dengan *node* A adalah *node* B karena diantara yang lain, hanya *node* A dengan *node* B yang berhubungan langsung dan diketahui bobot jaraknya.

2. Iterasi 1

$W = \min\{B, C, D, E, F, G, H, I\} = B$

$L = \{A, B\}$

$D[A, C] = \min(d[C], d[B] + c[B, C]) = \min(\infty, 23,3 + 10,9) = 34,2$

$D[A, D] = \min(d[D], d[B] + c[B, D]) = \min(\infty, 23,3 + 64,5) = 87,8$

$D[A, E] = \min(d[E], d[B] + c[B, E]) = \min(\infty, 23,3 + \infty) = \infty$

$D[A, F] = \min(d[F], d[B] + c[B, F]) = \min(\infty, 23,3 + \infty) = \infty$

$D[A, G] = \min(d[G], d[B] + c[B, G]) = \min(\infty, 23,3 + \infty) = \infty$

$D[A, H] = \min(d[H], d[B] + c[B, H]) = \min(\infty, 23,3 + \infty) = \infty$

$D[A, I] = \min(d[I], d[B] + c[B, I]) = \min(\infty, 23,3 + \infty) = \infty$

Berdasarkan hasil perhitungan iterasi di atas, dapat kita lihat bahwa jarak terpendek adalah dari A, B, C. Maka untuk melanjutkan sampai ke *node* I (Pekanbaru) kita akan mengambil dari jalur A, B dan C. Untuk iterasi selanjutnya maka kita akan memulai dari *node* C (Perbaungan).

3. Iterasi 2

$W = \min\{C, E, F, G, H, I\} = C$

$L = \{A, B, C\}$

$D[A, E] = \min(d[E], d[A, C] + c[[A, C], E]) = \min(\infty, 34,2 + 43,9) = 78,1$

$D[A, F] = \min(d[F], d[A, C] + c[[A, C], F]) = \min(\infty, 34,2 + \infty) = \infty$

$D[A, G] = \min(d[G], d[A, C] + c[[A, C], G]) = \min(\infty, 34,2 + \infty) = \infty$

$D[A, H] = \min(d[H], d[A, C] + c[[A, C], H]) = \min(\infty, 34,2 + \infty) = \infty$

$D[A, I] = \min(d[I], d[A, C] + c[[A, C], I]) = \min(\infty, 34,2 + \infty) = \infty$

Dari hasil iterasi di atas, dapat kita lihat hanya satu jalur yang kita dapat. Maka untuk iterasi berikutnya kita akan memulai dari jalur tersebut.

4. Iterasi 3

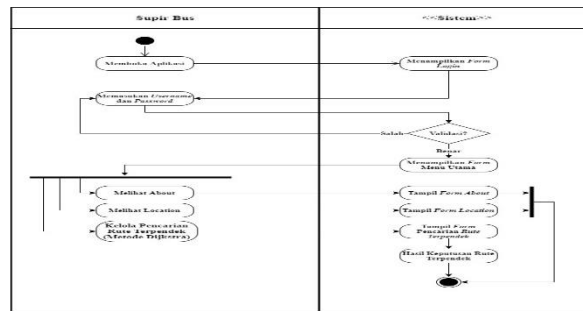
$W = \min\{E, F, G, H, I\} = E$

$L = \{A, B, C, E\}$

$D[A, F] = \min(d[F], d[A, E] + c[[A, E], F]) = \min(\infty, 78,1 + 82,6) = 160,7$

$D[A, G] = \min(d[G], d[A, E] + c[[A, E], G]) = \min(\infty, 78,1 + \infty) = \infty$

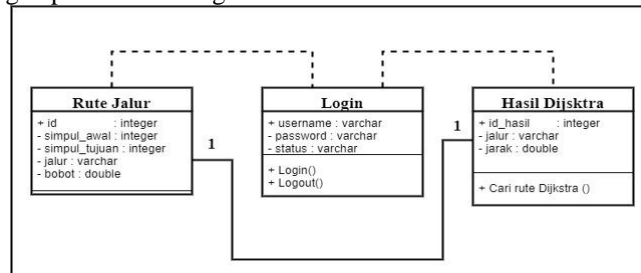
$D[A, H] = \min(d[H], d[A, E] + c[[A, E], H]) = \min(\infty, 78,1 + \infty) = \infty$



Gambar 4.2 Activity Diagram Metode Dijkstra

4.1.3 Class Diagram

Untuk *class diagram* untuk sistem yang akan dibangun yaitu menentukan rute atau jarak terpendek dalam pengiriman barang dapat dilihat sebagai berikut:



Gambar 4.3 Class Diagram Metode Dijkstra

HASIL DAN PEMBAHASAN

5.1 Kebutuhan Sistem

Kebutuhan sistem merupakan hal-hal yang dibutuhkan dalam proses pembuatan sistem. Adapun kebutuhan sistem yang dibutuhkan ada 2 yaitu: Perangkat keras (*Hardware*) dan Perangkat lunak (*Software*).

5.1.1 Perangkat Keras (*Hardware*)

Perangkat keras yang dibutuhkan dalam penelitian ini adalah sebagai berikut:

1. Laptop dengan spesifikasi (*core i3-5005U*)
2. Kapasitas Hardisk (500 GB)
3. Ram 2 GB DDR3

5.1.2 Perangkat Lunak (*Software*)

Adapun perangkat lunak yang digunakan untuk membangun sistem dalam penelitian ini yaitu sebagai berikut:

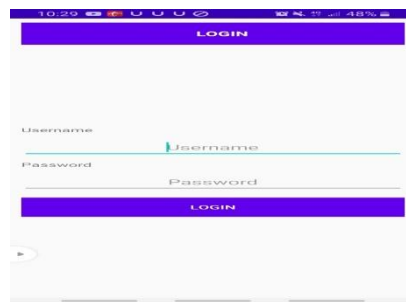
1. Sistem Operasi *Windows 10*
2. *Software* Aplikasi yang digunakan yaitu:
 - a. *Android Studio 4.1.3*
 - b. *Xampp 3.2.4*
 - c. *Emulator*

5.2 Hasil Tampilan Antarmuka

Tampilan antar muka adalah gambar hasil tampilan halaman dari aplikasi desktop sistem yang telah dibuat yang disesuaikan dengan perancangan *interface* sebelumnya. Adapun gambaran tampilannya adalah sebagai berikut:

1. Halaman Tampilan *Login*

Pada halaman *login* ada beberapa fungsional yang terletak di halaman *login*, yaitu : penginputan *username*, *password* dan tombol *login*.

Gambar 5.1 Halaman Tampilan *Login*

2. Halaman Tampilan Menu Utama

Halaman menu utama merupakan halaman untuk menampilkan halaman utama yang sebagai penghubung untuk menampilkan halaman lain dari sistem:



Gambar 5.2 Halaman Tampilan Menu Utama

3. Halaman Proses Metode Dijkstra

Halaman ini adalah halaman yang digunakan untuk proses pencarian rute.



Gambar 5.3 Halaman Tampilan Metode Dijkstra

4. Halaman Tampilan *My Location*

Halaman ini menampilkan titik lokasi dari *user*.

Gambar 5.4 Halaman Tampilan *My Location*

5. Halaman Tampilan About

Halaman ini untuk menampilkan keterangan aplikasi.



.Gambar 5.5 Halaman Tampilan About

5.3 Pengujian

Adapun kesimpulan dari penelitian ini berdasarkan dari rumusan masalah adalah sebagai berikut:

1. Dalam proses penentuan *rute* terpendek dalam pengiriman barang kepada konsumen PT.PMH yaitu berdasarkan titik *node* yang telah ditentukan untuk jalur pengiriman barang PT.PMH.
2. Dalam membangun aplikasi untuk pendukung keputusan dalam penentuan *rute* terpendek pengiriman barang pada PT.PMH yaitu dengan menerapkan struktur sistem pendukung keputusan kedalam aplikasi sistem yang akan dibangun.

6.2 Saran


Dalam penelitian ini diharapkan mendapatkan saran sebagai pertimbangan kepada pihak-pihak yang berkepentingan guna untuk mengembangkan penelitian yang telah dibuat agar dapat menyempurnakan hasil dari penelitian ini, sehingga penelitian ini bisa lebih baik lagi kedepannya. Adapun saran yang dapat diberikan adalah sebagai berikut:

1. Agar kedepannya dapat melakukan penerapan metode lain yang dapat menentukan *rute* jalur terpende agar dapat dilakukan perbandingan terhadap metode yang digunakan saat ini dan dapat dilakukan penilaian metode mana yang lebih baik untuk digunakan.
2. Aplikasi sistem masih memerlukan perbaikan dari beberapa hal seperti keamanan, penambahan *user* baru dan sistem *backup*.
3. Diharapkan agar kedepannya dapat menyempurnakan sistem ini, dikarenakan sistem ini hanya dapat memproses untuk jalur sampai kota pekanbaru saja.

DAFTAR PUSTAKA

- [1] P. Algoritma *et al.*, "No Title," pp. 69–78, 2016.
- [2] R. Taufiq, A. A. Permana, T. Cahyanto, and R. Adha, "Sistem Pendukung Keputusan Penerimaan Karyawan Menggunakan Simple Additive Weighting Studi Kasus PT. Trafoindo Prima Perkasa," vol. 4, no. 4, pp. 186–194, 2018.
- [3] M. A. F. Nugroho, Y. W. Syaifudin, and D. Puspitasari, "Penentuan Jarak Terpendek Menggunakan Metode Dijkstra Pada Data Spasial Openstreetmap (Studi Kasus : Pada Perusahaan Pengantaran Barang Wahana Logistik Kota Malang)," *Smatika J.*, vol. 9, no. 01, pp. 45–50, 2019, doi: 10.32664/smatika.v9i01.265.
- [4] M. N. Parapat, D. Kusbianto, and C. Rahmad, "JASA KIRIMAN BARANG BERBASIS MOBILE DENGAN METODE," pp. 15–19.
- [5] T. Limbong, *Sistem Pendukung Keputusan: Metode & Implementasi*, Cetakan I. Medan: Yayasan Kita Menulis, 2020.
- [6] O. Nurdiawan, "Seleksi Penerima Bantuan Sosial Berdasarkan Sistem Pendukung Keputusan Dalam Upaya Mengurangi Siswa Rawan Putus Sekolah," *J. Teknol. Inf.*, vol. XIII, no. 2, pp. 32–40, 2018.
- [7] A. Nurdianto, S. J. Hartati, and Y. M. Maulana, "Rancang Bangun Sistem Pendukung KeNurdianto, A., Hartati, S. J., & Maulana, Y. M. (2016). Rancang Bangun Sistem Pendukung Keputusan Penempatan Tenaga Pendidik Menggunakan, 5(2), 1–8.putusan Penempatan Tenaga Pendidik Menggunakan," vol. 5, no. 2, pp. 1–8, 2016.
- [8] P. W. Luh Made Yulyantari, *MANAJEMEN MODEL PADA SISTEM PENDUKUNG KEPUTUSAN*. Yogyakarta: Penerbit ANDI, 2019.
- [9] A. Cantona, F. Fauziah, and W. Winarsih, "Implementasi Algoritma Dijkstra Pada Pencarian Rute Terpendek ke Museum di Jakarta," *J. Teknol. dan Manaj. Inform.*, vol. 6, no. 1, pp. 27–34, 2020, doi: 10.26905/jtmi.v6i1.3837.
- [10] A. Pitri, "Penerapan Metode Dijkstra Pencarian Rute Terpendek Sekolah Luar Biasa (SLB) di Kota Medan," *J. Ris. Komput.*, vol. 5, no. 6, pp. 638–643, 2018.
- [11] M. S. Yusuf, H. M. Az-zahra, and D. H. Apriyanti, "Implementasi Algoritma Dijkstra Dalam Menemukan Jarak Terdekat Dari Lokasi Pengguna Ke Tanaman Yang Di Tuju Berbasis Android (Studi Kasus di Kebun Raya Purwodadi)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 1, no. 12, pp. 1779–1781, 2017.
- [12] Ahyar Rivai dan Hasibuan, "Penerapan algoritma floyd warshall untuk menentukan jalur terpendek dalam pengiriman barang," *J. Ris. Komput.*, vol. 3, no. 6, pp. 20–24, 2016.
- [13] M. K. Harahap, "Pencarian Jalur Terpendek dengan Algoritma Dijkstra," vol. 2, pp. 18–23, 2019.
- [14] T. A. Kurniawan, "Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 1, p. 77, 2018, doi: 10.25126/jtiik.201851610.

BIOGRAFI PENULIS

	<p>Nama : Hermawan Susanto</p> <p>NIRM : 2017020024</p> <p>Status : Mahasiswa</p> <p>Program Studi : Sistem Informasi (SI)</p> <p>E-Mail : Shermawan397@gmail.com</p> <p>Bidang Keilmuan : 1. Programming 2. Desain Grafis</p>
	<p>Nama : Milfa Yetri, S.Kom., M.Kom</p> <p>NIDN : 0109038802</p> <p>Jabatan : Dosen Tetap STMIK Triguna Dharma</p> <p>Program Studi : Sistem Informasi (SI)</p> <p>E-Mail : milfa.anfa03@gmail.com</p> <p>Bidang Keilmuan : 1. Teknik Presentasi 2. Arsitektur Komputer 3. Animasi 4. Pengantar Teknologi Informasi 5. Perakitan dan Perawatan Komputer 6. Data Mining</p>
	<p>Nama : Itam Mariami, S.E., M.Si</p> <p>NIDN : 0103046601</p> <p>Jabatan : Dosen Tetap STMIK Triguna Dharma</p> <p>Program Studi : Sistem Informasi (SI)</p> <p>E-Mail : Itamariami66@gmail.com</p> <p>Bidang Keilmuan : 1. Manajemen</p>