

IMPLEMENTASI ALGORITMA DIJKSTRA UNTUK Mencari RUTE TERPENDEK ANTAR KANTOR DAN ESTIMASI PENGGUNAAN BAHAN BAKAR KENDARAAN (STUDI KASUS PT. TELKOM INDONESIA REGIONAL IV JATENG-DIY)

Fairuz Eka Andiany¹, Wiwien Hadikurniawati²

^{1,2}Teknik Informatika, Fakultas Teknologi Informasi, Universitas Stikubank
e-mail: ¹fairuzeka@hotmail.com, ²wiwien@edu.unisbank.ac.id

ABSTRAK

Pegawai perusahaan umumnya mendapatkan fasilitas yang memadai untuk menunjang pekerjaan. PT Telkom Indonesia menyediakan fasilitas peminjaman mobil dinas untuk mobilitas pegawai dalam menyelesaikan pekerjaan. Untuk karyawan dengan divisi tertentu yang memang pekerjaannya mempunyai mobilitas tinggi, mobil beserta estimasi biaya bahan bakar disediakan untuk mempermudah menuju ke lokasi tujuan. Peminjaman mobil tersebut tentunya dengan prosedur yang telah ditetapkan. Web dipilih sebagai sarana untuk melakukan peminjaman mobil dinas. Web yang dibuat tidak hanya sekedar web peminjaman biasa, namun juga dilengkapi dengan algoritma tertentu untuk menunjukkan rute terpendek dan juga dapat menentukan estimasi biaya bahan bakar yang akan dikeluarkan. Algoritma yang digunakan adalah algoritma Dijkstra. Berdasarkan studi pustaka, algoritma Dijkstra mampu menentukan rute terpendek untuk tujuan yang telah ditentukan.

Kata Kunci: algoritma Dijkstra, rute terpendek, estimasi bahan bakar

1. PENDAHULUAN

Algoritma merupakan suatu metode yang sangat dibutuhkan dalam kehidupan sehari-hari. Mulai dari bangun tidur sampai tidur lagi adalah suatu rutinitas yang secara tidak sadar juga menggunakan algoritma. Secara umumnya, algoritma adalah urutan langkah-langkah untuk menyelesaikan masalah[1]. Namun, ada yang berpendapat bahwa Algoritma adalah urutan penyelesaian masalah yang disusun secara sistematis menggunakan Bahasa yang logis untuk memecahkan suatu permasalahan[2]. Selain digunakan dalam kegiatan sehari-hari, algoritma juga digunakan sebagai pendukung berjalannya suatu sistem. Salah satu contoh algoritma adalah untuk menentukan rute terpendek.

Metode untuk menemukan rute terpendek dalam sebuah sistem, sangatlah dibutuhkan. Hal tersebut membuat user lebih efisien waktu, cepat dan tepat untuk sampai ke lokasi tujuan. Untuk mendapatkan rute terpendek, ada beberapa algoritma yang dapat digunakan. Contohnya adalah algoritma Dijkstra, yang langsung menghitung node terpendek. Algoritma Floyd, yang menghitung terlebih dahulu seluruh jalur yang akan ditempuh [3]. A Star adalah algoritma menggunakan sifat greedy dan fungsi heuristik untuk mendapatkan rutenya [3], algoritma Greedy dapat menghasilkan rute terpendek dengan menghitung nilai lokal optimal setiap mengunjungi kota dan mendapatkan nilai optimasi global pada akhir perjalanan[4] dan masih banyak lagi. Untuk sebuah kasus yang memiliki bobot node non-negative, Algoritma Dijkstra adalah pilihan yang dianjurkan. Selain itu juga menghasilkan hanya satu jalur terpendek.

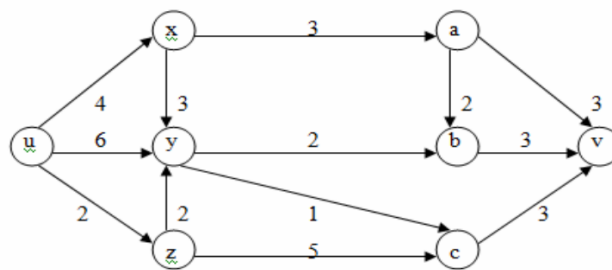
PT. Telkom Indonesia adalah sebuah perusahaan yang menyediakan pelayanan jaringan telekomunikasi di Indonesia. PT. Telkom Indonesia memiliki beberapa web terorganisir sesuai dengan kebutuhannya. Salah satunya adalah web pemesanan mobil dinas. Web ini digunakan untuk para karyawan yang akan pergi dinas ke kantor Telkom di wilayah lain. Dalam kasus mencari rute terpendek, web pemesanan mobil dinas di PT. Telkom Indonesia Regional IV Jateng-DIY masih perlu pembenahan karena dalam web pemesanan mobil tersebut masih belum menggunakan algoritma Dijkstra, sehingga dalam menuju ke kantor Telkom yang lainnya masih belum optimal. Selain itu, untuk perhitungan bahan bakar juga belum diterapkan dalam web tersebut. Penggunaan algoritma Dijkstra yang diimplementasikan dalam web pemesanan mobil dinas di PT. Telkom Indonesia Regional IV Jateng-DIY, maka hanya akan ada satu rute pasti yang terpendek sebagai outputnya. Rute tersebut akan divisualisasikan dengan maps dan akan muncul angka untuk menunjukkan kilometer jarak yang akan ditempuh. Jarak yang ditempuh ini akan mengestimasi rupiah yang dikeluarkan untuk bahan bakar selama perjalanan dinas.

2. TINJAUAN PUSTAKA

Algoritma merupakan susunan langkah-langkah untuk menyelesaikan sebuah permasalahan. Algoritma pencarian rute terpendek maksudnya adalah bagaimana algoritma bekerja untuk mengoptimalkan pencarian rute terpendek dari lokasi awal menuju ke lokasi tujuan dengan memperhitungkan jarak kalkulasi terpendek.

Rute terpendek dapat diselesaikan dengan beberapa algoritma, yaitu Algoritma Dijkstra, Algoritma Astar, Algoritma Floyd, Algoritma Bellman dan masih banyak lagi. Inti dari beberapa algoritma tadi adalah menentukan jarak terpendek dari node-node yang terhubung[5].

Beberapa penelitian sebelumnya, algoritma dijkstra digunakan untuk mencari tempat parkir. Karna menurut peneliti pencarian tempat parkir secara manual masih belum efisien karna membutuhkan waktu lama, terutama ketika sedang berada di pusat perbelajaan. Ketika *driver* menekan tombol untuk mengambil tiket parkir, maka akan tercetak tiket beserta minimap untuk memudahkan *driver* mencari tempat parkir. Software yang digunakan adalah LabView untuk implementasi Dijkstra dan beberapa *hardware* untuk membantu proses pencarian tempat parkir. Diantaranya : Digital converter, GIS, GPS, Infrared, LCD, LED, National Instrument, Printed Circuit Board dan lain lain. Perbedaan penelitian yang dahulu dengan yang sekarang sangatlah signifikan. Mulai dari *software* dan *hardware* yang digunakan, kegunaan dalam pengimplementasian Dijkstra, dan alat alat yang digunakan untuk mendukung penelitian tersebut[6].Algoritma Dijkstra juga digunakandalam proses routing didalam jaringan internet. Routing adalah proses menemukan jalan antara sumber dan tujuan berdasarkan transmisi data. Untuk menemukan rute terpendek untuk transmisi data didalam jaringan, Dijkstra menggunakan koneksi matrik dan berat dari matrik. Matrik terdiri dari beberapa *path* dari node awal ke node tujuan. Software yang digunakan dalam proses routing adalah Matlab[7].



Gambar 1. Graf yang merupakan sebuah *network*

Untuk mendapatkan jalur terpendek dari simpul U ke V, maka langkah-langkah yang dilakukan :

Simpul U adalah sumber (source)

Simpul V adalah muara (sink)

Untuk menentukan jalur terpendek, maka dibuat tabel seperti pada tabel 1.

Tabel 1. Jarak node dalam *network*

u	x	y	z	a	b	c	v
ux = 4	xy = 3	yb = 2	zy = 2	ab = 2	bv = 3	cv = 3	
uy = 6	xa = 3	yc = 1	zc = 5	av = 3			
uz = 2							

Dimulai dengan simpul u, sebagai simpul awal. Beri harga = 0. Ambil simpul dengan jarak terdekat dari simpul u (pada gambar diatas z = 2), lingkari uz. Hapus semua ruas lain yang berakhir di z. Beri nilai = 2 dibelakang z. simpul yang telah dihitung, ditandai dengan (*).

Tabel 2. Perhitungan jarak

u*(0)	x	y	z*(2)	a	b	c	v
ux = 4	xy = 3	yb = 2	zy = 2(4)	ab = 2	bv = 3	cv = 3	
uy = 6	xa = 3	yc = 1	zc = 5(7)	av = 3			
uz = 2							

Simpul u dan z telah ditandai dengan (*), cari simpul lain yang jarak terdekatnya dihitung dari u. Perhatikan nilai perhitungan atau bobot yang telah tertulis disimpul (0 untuk u dan 2 untuk z). Pada tabel diatas, ux bernilai 4 dan uzy = 2 + 2 = 4 adalah nilai minimum. Diperbolehkan untuk memilih salah satu, contoh uzy. Beri nilai 4 pada y. tandai zy, hapus ruas lain yang menuju y yaitu uy dan xy.

Tabel 3. Penghilangan ruas uy dan xy

$u^*(0)$	x	y	$z^*(2)$	a	b	c	v
$ux = 4$	$xy = 3$	$yb = 2$	$zy = 2(4)$	$ab = 2$	$bv = 3$	$cv = 3$	
$uy = 6$	$xa = 3$	$yc = 1$	$zc = 5(7)$	$av = 3$			
$uz = 2$							

Langkah selanjutnya :

Tabel 4. Pelingkaran $ux = 4$

$u^*(0)$	$x^*(4)$	$y^*(4)$	$z^*(2)$	a	b	c	v
$ux = 4$	$xy = 3$	$yb = 2(6)$	$zy = 2(4)$	$ab = 2$	$bv = 3$	$cv = 3$	
$uy = 6$	$xa = 3(7)$	$yc = 1(5)$	$zc = 5(7)$	$av = 3$			
$uz = 2$							

Tabel 5. Pencoretan $zc = 5(7)$ dan pelingkaran $yc = 1(5)$

$u^*(0)$	$x^*(4)$	$y^*(4)$	$z^*(2)$	a	b	$c^*(5)$	v
$ux = 4$	$xy = 3$	$yb = 2(6)$	$zy = 2(4)$	$ab = 2$	$bv = 3$	$cv = 3(8)$	
$uy = 6$	$xa = 3(7)$	$yc = 1(5)$	$zc = 5(7)$	$av = 3$			
$uz = 2$							

Tabel 6. Pencoretan $ab = 2$ dan pelingkaran $yb = 2(6)$

$u^*(0)$	$x^*(4)$	$y^*(4)$	$z^*(2)$	a	$b^*(6)$	$c^*(5)$	v
$ux = 4$	$xy = 3$	$yb = 2(6)$	$zy = 2(4)$	$ab = 2$	$bv = 3(9)$	$cv = 3(8)$	
$uy = 6$	$xa = 3(7)$	$yc = 1(5)$	$zc = 5(7)$	$av = 3$			
$uz = 2$							

Tabel 7. Pelingkaran $xa = 3(7)$

$u^*(0)$	$x^*(4)$	$y^*(4)$	$z^*(2)$	$a^*(7)$	$b^*(6)$	$c^*(5)$	v
$ux = 4$	$xy = 3$	$yb = 2(6)$	$zy = 2(4)$	$ab = 2$	$bv = 3(9)$	$cv = 3(8)$	
$uy = 6$	$xa = 3(7)$	$yc = 1(5)$	$zc = 5(7)$	$av = 3(10)$			
$uz = 2$							

Tabel 8. Pencoretan $bv = 3(9)$

$u^*(0)$	$x^*(4)$	$y^*(4)$	$z^*(2)$	$a^*(7)$	$b^*(6)$	$c^*(5)$	$v^*(8)$
$ux = 4$	$xy = 3$	$yb = 2(6)$	$zy = 2(4)$	$ab = 2$	$bv = 3(9)$	$cv = 3(8)$	
$uy = 6$	$xa = 3(7)$	$yc = 1(5)$	$zc = 5(7)$	$av = 3(10)$			
$uz = 2$							

Setelah semua tabel dikerjakan, maka jalur terpendek dari simpul u dan v adalah 8. Dengan urutan :

$$V \leftarrow C \leftarrow Y \leftarrow Z \leftarrow U$$

3. METODE PENELITIAN

3.1 Perumusan Objek Penelitian

Objek penelitian merupakan hal utama yang harus diperhatikan dalam sebuah penelitian, karna objek penelitian adalah sasaran untuk mendapatkan jawaban dan solusi dari sebuah masalah.

Objek penelitian dalam tugas akhir ini adalah pencarian rute terpendek untuk menuju ke kantor Telkom lainnya dalam web pemesanan mobil dinas. Sedangkan subjek penelitiannya adalah PT. Telkom Indonesia Regional IV Jateng-DIY.

3.2 Metode pengumpulan data

Terdapat dua metode yang dilakukan dalam pengumpulan data untuk penelitian ini, yaitu:

1. Metode wawancara

Wawancara adalah Teknik mengumpulkan data yang dilakukan melalui tatap muka dan tanya jawab langsung antar peneliti dan narasumber. Seiring perkembangan teknologi, kini wawancara dapat dilakukan melalui smartphone. Lebih tepatnya, wawancara yang digunakan adalah wawancara terstruktur dimana peneliti telah mengetahui informasi apa yang hendak digali dari narasumber.

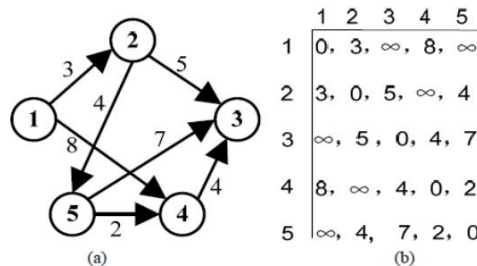
2. Observasi

Observasi adalah metode pengumpulan data yang kompleks karena melibatkan berbagai factor dalam pelaksanaannya. Metode pengumpulan data obeservasi tidak hanya mengukur sikap dari responden, namun juga dapat digunakan untuk merekam berbagai fenomena yang terjadi. Observasi yang digunakan adalah non participant observasion, dimana peneliti tidak ikut secara langsung dalam kegiatan atau proses yang diamati.

4. HASIL DAN PEMBAHASAN

Algoritma Dijkstra melakukan kalkulasi bobot terkecil dari satu node ke node yang lain. Dengan arti lain, Dijkstra menghitung lintasan berdasarkan jarak terpendek yang ditempuh tiap-tiap kota. Salah satu komponen dari algoritma dijkstra adalah graf dan matriks ketetangaan. Graf merupakan pasangan $G = (V,E)$. Secara geometri graf digambarkan sebagai sekumpulan simpul didalam bidang dwimatra(dua dimensi) yang dihubungkan dengan sekumpulan garis(sisi). Dalam matriks ketetangaan terdapat komponen utama sebagai penyusunnya yaitu lintasan. Dua buah simpul dalam sebuah graf dinyatakan bertetangga apabila keduanya terhubung langsung dalam sebuah sisi. Lintasan yang panjangnyaⁿdari simpul awal V_0 ke simpul tujuan V_n didalam suatu graf adalah berselang-seling antar simpul dan sisi-sisi yang dapat dinotasikan sebagai :

$$e_1=(V_0,V_1),e_2=(V_1,V_2), \dots , e_n=(V_{n-1}, V_1) \tag{1}$$



Gambar 2. Graf berbobot dan matriks ketetangaan.

Untuk itu $G=(V,E)$. V merepresentasikan dari vertex/node. E merepresentasikan edge. $V = \{V_1, V_2, V_3, \dots, V_n\}$ termasuk pula n nodes. Matriks ketetangaan untuk graf berarah. $A = (a_{ij}) n \times n$ dijelaskan pada persamaan (1)

$$a_{ij} = \begin{cases} \infty, & (V_i, V_j) \notin E \\ W_{ij}, & (V_i, V_j) \in E \end{cases}, i, j = 1, 2, \dots, n \tag{2}$$

Dimana W_{ij} merupakan bobot dari edge $\langle V_i, V_j \rangle$ sedangkan ∞ menyatakan bahwa tidak ada edge antara V_i dan V_j . L menyatakan gabungan titik akhir yang telah mendapatkan rute terpendek dari titik awal. (V-L) menyatakan gabungan node yang belum dihitung dalam proses pencarian rute terpendek. Gunakan matrix ketetangaan W untuk menyimpan informasi jaringan. W_{ij} dinotasikan sebagai bobot dari garis $\langle V_i, V_j \rangle$. Jika ternyata tidak ada keterkaitan antara V_i dan V_j , selanjutnya diubah menjadi ∞ . d_i menggambarkan sebagai bobot dari titik awal ke node V_i . Inisialisasikan titik awal dengan $d_s=0$ dan $D_i = C_{si}$. $W_{(i,j)}$ dimana mengandung bobot garis dari titik awal hingga titik akhir yang saling berhubungan dalam sebuah graf. Titik awal adalah i dan titik akhir adalah j .

$$V(G) - L = \{V_a, V_b, V_c, \dots, V_n\} - \{i\} = \{V_a, V_b, V_c, \dots, V_n\} \tag{3}$$

Untuk mengetahui node mana yang telah dipilih dalam proses penghitungan gunakan (2). Selama $V_n \notin L$:

a. Pilih titik $V_k \in V-L$ dengan nilai $D(k)$ terkecil

b. Untuk setiap $V_j \in V-L$, lakukan

$$(D(j), (D(k) + W(k,j))) \tag{4}$$

$$\text{Jika } (D(j) > D(k) + W(k,j)) \tag{5}$$

$$\text{Pada iterasi selanjutnya ganti nilai bobot } D(j) \text{ dengan } (D(k) + W(k,j)) \tag{6}$$

Hasil pengimplementasian dalam web adalah sebagai berikut:

Gambar 3. Tampilan Login

Dalam penelitian ini, ada dua level user yang dapat login. Pertama, admin dan yang kedua adalah user. Admin adalah manager yang mempunyai wewenang untuk dapat menyetujui atau menolak permintaan dari user. User adalah karyawan yang akan meminjam mobil. Karyawan mempunyai wewenang untuk mengisi form data diri dan form alamat tujuan.

Gambar 4. Form data diri

Setelah memasukan NIK dan password, jika NIK adalah user karyawan maka akan dilempar ke halaman form data diri. Karyawan wajib mengisi form data diri sebelum memilih lokasi tujuan.

Hasil Hitung Jarak Dan Waktu Tempuh				
Lokasi Asal	Lokasi Tujuan	Jarak Tempuh	Perkiraan Waktu	Estimasi BBM
Telkom Semarang	Telkom Pemalang	132 km	3 h 20 min	150000

Gambar 5. Form Alamat Tujuan

Setelah karyawan mengisi data diri, maka selanjutnya mengisi alamat tujuan. Alamat tujuan ada 4 kantor Telkom sebagaimana yang telah disediakan didatabase. Kantor tersebut adalah Telkom Semarang, Telkom

Demak, Telkom Bawen, dan Telkom Pemalang. Setelah alamat tujuan dipilih, klik run. Maka perhitungan algoritma Dijkstra akan muncul seperti pada di gambar.

KBMOnline Logout

Selamat datang, Admin
Silahkan cek request dibawah ini

List Approval

#	NIK	Jenis Keperluan	Jenis Pemesanan	No. Telp Kantor	No. Hp	Keterangan	#
1	160098	Sosial	Mobil	8449914	08123444	Approved	Lihat Details
2	160098	Dinas	Mobil dan Sopir	8449914	089765	Approved	Lihat Details
3	160098	Event	Mobil	8449914	0856777775	Approved	Lihat Details

Details View

#	NIK	Lokasi Asal	Lokasi Tujuan	Jarak	Waktu	Bensin
1	1600098	Semarang	Pemalang	132 km	3 h 20 min	200000
2	1600098	Semarang	Bawen	42.7 km	59 min	130000
3	1600098	Semarang	Demak	33.1 km	1 h 1 min	100000

Gambar 6. Halaman admin

Admin, hanya dapat mengakses satu halaman saja, yaitu halaman admin atau halaman approval. Halaman ini berisi permintaan dari karyawan yang telah mengisi beberapa form diatas. Untuk menyetujui atau menolak permintaan adalah kondisional. Disamping tombol approved, ada tombol untuk melihat detail. Ketika lihat detail diklik, maka akan muncul penjabaran seperti yang ada di gambar 8 bagian details view.

5. KESIMPULAN

Dari hasil penelitian yang dilakukan, maka ditarik beberapa kesimpulan yaitu:

- Sistem ini dibuat untuk mempermudah pekerjaan karyawan perusahaan dalam proses peminjaman mobil.
- Sistem ini mengurangi penggunaan kertas dalam pengajuan peminjaman mobil.
- Sistem ini berbasis localhost, yang hanya bisa diakses ketika hanya berada didalam kantor saja.
- Algoritma Dijkstra memilih bobot node terpendek untuk menentukan jalur terpendek.
- Web yang menggunakan algoritma Dijkstra akan sangat terbantu dalam pencarian rute terdekat.
- Dengan algoritma Dijkstra maka user dapat mengetahui jarak tempuh, estimasi waktu (asumsi kondisi jalanan lancar), dan perkiraan bahan bakar kendaraan yang akan dipakai.

DAFTAR PUSTAKA

- [1] Munir, R., 2016, *Algoritma Dan Pemrograman Dalam Bahasa Pascal, C, Dan C++ Edisi Keenam*. Bandung : Penerbit Informatika.
- [2] Barakbah, Ali R, Tita K., and Ahmad, S. A., 2013, *Logika dan Algoritma*. Surabaya : Politeknik Negeri Surabaya.
- [3] Cahyadi, M. Azan and P, M. Arif Bambang. and Widhiarso, Wijang and Yohannes, Yohannes, 2017, *Perbandingan Algoritma A*, Dijkstra Dan Floyd Warshall Untuk Menentukan Jalur Terpendek Pada Permainan "Bacteria Defense" Palembang: .STMIK GI MDP*
- [4] Herli A.M, Indra, K. R., Purbandini, 2015, *Sistem Pencarian Hotel Berdasarkan Rute Perjalanan Terpendek Dengan Mempertimbangkan Daya Tarik Wisata Menggunakan Algoritma Greedy*. Surabaya: Universitas Airlangga.
- [5] Yusaputra, Rizky. (2013). *Aplikasi Mobile Pencarian Rute Terpendek Lokasi Fasilitas Umum Berbasis Android Menggunakan Algoritma Floyd-Wharsall*. Riau :Universitas Islam Negeri Sultan Syarif Kasim Riau.
- [6] Jafaar, H., M.H Zabidi, Soh A.C., Hoong, T.P., Shafie, S., Ahmad, S.A., 2014, *Intelligence Guidance Parking Sistem Using Modified Dijkstra's Algorithm*. Malaysia: Universiti Putra Malaysia.
- [7] Gupta, N., Kapil, M. Anand, K. J., Md. U., 2016, *Applying Dijkstra's Algorithm in Routing Process*.