

Implementasi Algoritma Dijkstra Untuk Menentukan Lokasi dan Jarak Tempuh Terpendek Kampus IT di Jakarta

Jeki Sauwani S¹, Vitcky Nanda Putra S², Halim Agung S³

¹Universitas Bunda Mulia
e-mail: mr.jekisauwani@gmail.com

²Universitas Bunda Mulia
e-mail: vitkynptr@gmail.com

³Universitas Bunda Mulia
e-mail: hagung@bundamulia.ac.id

Abstrak

Kemajuan teknologi saat ini tentunya sangat memudahkan masyarakat untuk mencari informasi terkait perguruan tinggi yang diminati, baik itu fasilitas, alamat, dan jalan menuju lokasi, namun pada kenyataannya masih banyak dari masyarakat baik berasal dari dalam maupun luar kota Jakarta yang belum mengetahui lokasi dan *rute* jalan yang harus dilewati untuk menuju ke perguruan tinggi dengan program studi Teknik Informatika di Jakarta. Hal ini dikarenakan kurang jelasnya informasi yang di dapat. Proses pencarian *rute* akan menggunakan *algoritma djikstra* dan hasil akhir yang ditampilkan adalah informasi *rute* jalan yang harus dilalui, juga peta yang menampilkan lokasi perguruan tinggi dengan program studi Teknik Informatika di Jakarta. Hasil pengujian dapat disimpulkan bahwa tingkat ketepatan *rute* yang didapatkan dari *aplikasi* ini adalah sebesar 75 %, dikarenakan masih ada beberapa posisi dan *rute* yang tidak 100 % akurat.

Kata Kunci : Kampus IT, Penentuan jalur terpendek, *Algoritma Dijkstra*

Abstract

Today's technological improvement certainly make it easier for people to find information regarding universities, both addresses, and roads to locations, but in reality there are still many from both inside and outside Jakarta who dont know the location and route to go to college with an Informatics Engineering study program in Jakarta. This is due to the lack of clarity in the information obtained. The route search process will use the djikstra algorithm and the final results displayed are information on the route that must be passed, as well as a map that displays the location of universities with an Informatics Engineering study program in Jakarta. The test results can be concluded that the level of accuracy of the routes obtained from this application is 75%, because there are still several positions and routes that are not 100% accurate.

Keywords: IT Campus, Determination of the shortest path, Dijkstra algorithm

1. Pendahuluan

DKI Jakarta adalah kota dengan jumlah perguruan tinggi terbanyak di Indonesia, dimana masing-masing perguruan tingginya memiliki berbagai macam program studi, salah satunya adalah program studi Teknik Informatika.

Program studi Teknik Informatika merupakan salah satu program studi yang memiliki peningkatan perkembangan yang

sangat pesat saat ini. Kemajuan teknologi di era globalisasi merupakan salah satu faktor yang mempengaruhi perkembangan program studi ini. Hal ini meningkatkan keingintahuan dan minat masyarakat terhadap program studi ini dan menjadikannya sebagai salah satu tujuan untuk melanjutkan pendidikan ke perguruan tinggi.

Tingginya animo masyarakat terhadap program studi Teknik Informatika tentunya tidak hanya berasal dari warga kota Jakarta saja melainkan juga dari berbagai daerah di Indonesia. Banyaknya perguruan tinggi dengan kualitas yang sangat baik di Jakarta menjadi daya tarik tersendiri bagi masyarakat luas untuk melanjutkan pendidikannya ke perguruan tinggi yang ada di Jakarta.

Kemajuan teknologi saat ini tentunya sangat memudahkan masyarakat untuk mencari informasi terkait perguruan tinggi yang diminati, baik itu fasilitas, alamat, dan jalan menuju lokasi, namun pada kenyataannya masih banyak dari masyarakat baik berasal dari dalam maupun luar kota Jakarta yang belum mengetahui lokasi dan *route* jalan yang harus dilewati untuk menuju ke perguruan tinggi dengan program studi Teknik Informatika di Jakarta. Hal ini dikarenakan kurang detailnya informasi yang di dapat.

Dengan keunggulan jaringan internet dan sistem informasi *geografis* di dalamnya, akan dikembangkan sebuah sistem untuk mengatasi persoalan pencarian lokasi dan pencarian *route* terpendek menuju lokasi perguruan tinggi yang ditentukan.

Proses pencarian *route* akan menggunakan *algoritma djikstra* dan hasil akhir yang ditampilkan adalah informasi *route* jalan yang harus dilalui, juga peta yang menampilkan lokasi perguruan tinggi dengan program studi Teknik Informatika di Jakarta.

2. Metode Penelitian

Metode Pengumpulan Data

Dalam penelitian ini menggunakan beberapa metodologi untuk mengumpulkan data yaitu sebagai berikut:

1. Studi pustaka, mengumpulkan data dari membaca beberapa buku, jurnal, dan karya ilmiah yang berhubungan dengan penggunaan *algoritma djikstra* dan penentuan jarak terpendek dalam berbagai kasus.
2. *Dokumentasi*, dalam metode ini peneliti mencari data mengenai hal-hal berupa catatan, *transkrip*, buku, surat kabar, majalah, agenda dan sebagainya.
3. *Observasi*
Peneliti melakukan observasi langsung keadaan *route* jalan kebeberapa perguruan tinggi dengan program studi

Teknik Informatika di Jakarta guna mendapatkan data dan informasi yang akurat terkait lokasi dan *route* yang dilewati menuju masing masing perguruan tinggi.

Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah metode waterfall dengan urutan proses sebagai berikut:

1. *Requirements Analysis and Definition*
Pada tahap ini dikumpulkan secara lengkap kebutuhan-kebutuhan mengenai perangkat lunak, kemudian peneliti menganalisis apa saja kebutuhan-kebutuhan yang dibutuhkan pada perangkat lunak.
2. *System and Software Design*
Pada tahap ini peneliti melakukan desain setelah kebutuhan selesai dikumpulkan secara lengkap.
3. *Implementation and Unit Testing*
Pada proses ini desain program diterjemahkan kedalam kode-kode dengan menggunakan bahasa pemrograman yang sudah ditentukan. Program yang dibangun langsung diuji dengan tujuan agar program dapat diuji dengan baik.
4. *Intergration and System Testing*
Pada proses ini adalah penyatuan unit-unit program. Setelah itu dilakukan pengujian secara keseluruhan mengenai perangkat lunak pencarian perguruan tinggi dengan prodi TI di Jakarta.
5. *Operation and Maintenance*
Pada proses ini dilakukan pengujian cobaan perangkat lunak pencarian restoran, tujuan dari proses ini adalah mengurangi kesalahan yang ada pada perangkat lunak yang dibuat dan sesuai dengan kebutuhan. mengoperasikan program dilingkungannya dan melakukan pemeliharaan, seperti penyesuaian.

3. Hasil dan Pembahasan

Pencarian jarak terpendek pada penelitian ini menggunakan *algoritma djikstra*, penentuan jarak terpendek dimulai dengan menghitung jarak *user* ke masing-masing kampus IT di Jakarta. Perhitungan jarak dilakukan oleh sistem, dengan menentukan posisi awal *user* pada peta dan memilih tujuan kampus IT di Jakarta yang ingin dikunjungi. Perhitungan jarak ini akan menghasilkan rekomendasi *route*

perjalanan yang akan dilewati *user* menuju ke kampus IT tujuan. Dalam mencari solusi, *algoritma djikstra* menggunakan prinsip *greedy*, yaitu mencari solusi *optimum* pada setiap langkah yang dilalui, dengan tujuan untuk mendapatkan solusi *optimum* pada langkah selanjutnya yang akan mengarah pada solusi terbaik. Hal ini membuat kompleksitas waktu *algoritma djikstra* menjadi cukup besar, yaitu seperti pada rumus persamaan berikut :

$$O(V * \text{Log}(v + e))$$

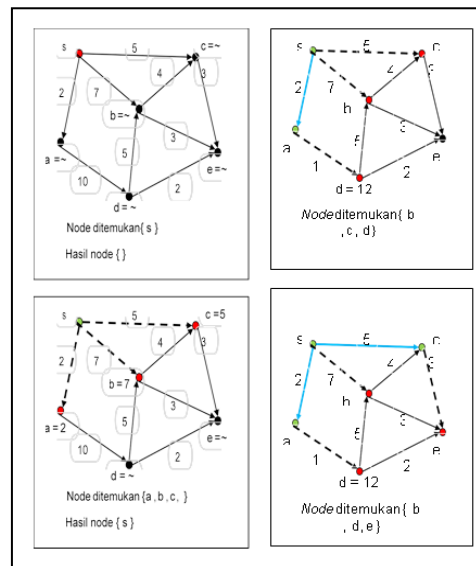
Rumus 1. Persamaan *algoritma djikstra* [1]

Dimana v dan e adalah simpul dan sisi pada *graf* yang digunakan.

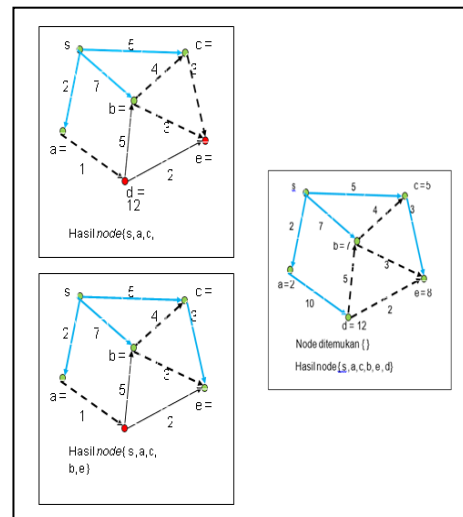
Input dari *algoritma djikstra* berupa sebuah *graf* berbobot $G(e, v)$, sedangkan outputnya berupa *rute* terpendek dari simpul awal (start) ke masing-masing simpul yang ada pada *graf*. Dengan demikian *algoritma djikstra* dapat menemukan solusi terbaik.

Cara kerja *algoritma djikstra* hampir sama dengan cara kerja *algoritma BFS* yaitu dengan menggunakan prinsip antrian (*queue*), akan tetapi antrian yang digunakan *algoritma djikstra* adalah antrian berprioritas (*priority queue*). Jadi hanya simpul yang memiliki prioritas tertinggi yang akan ditelusuri. Dalam menentukan simpul yang berprioritas, *algoritma* ini membandingkan setiap nilai (bobot) dari simpul yang berada pada satu *level*. Selanjutnya nilai (bobot) dari setiap simpul tersebut disimpan untuk dibandingkan dengan nilai yang akan ditemukan dari *rute* yang baru ditemukan kemudian, begitu seterusnya sampai ditemukan simpul yang di cari.

Berikut ini langkah-langkah *algoritma djikstra* dalam mencari *rute* terpendek pada sebuah *graf* .:



Gambar 1. Proses pencarian *rute algoritma djikstra* pada *Graf*. [2]



Gambar 2. Proses lanjutan pencarian *rute algoritma Dijkstra* pada *Graf*. [3]

Setelah langkah-langkah tersebut dijalankan akan dihasilkan solusi terbaik berupa *rute* perjalanan terpendek menuju lokasi tujuan.

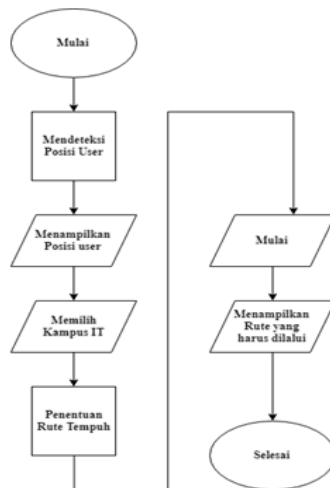
Algoritma ini juga digunakan dalam penelitian yang berjudul Penentuan *Rute* Terpendek Pendistribusian Naskah Ujian Nasional Menggunakan *Algoritma Dijkstra* [7]. Hasil dari penelitian tersebut adalah *algoritma djikstra* mampu menyelesaikan permasalahan pendistribusian naskah ujian nasional di kota Binjai dengan efektif dan cepat, hasil penelitian ini dapat dijadikan acuan untuk pengembangan lanjutan. Selain dari itu juga ada penelitian lain yaitu menggunakan Protokol *Routing OSPF* (*Open Source Path First*) Menggunakan

GNS3 Untuk Mencari Jalur Terpendek Dengan Menerapkan Algoritma *Dijkstra* [8]. Dan kesimpulannya adalah hasil penelitian ini menunjukkan bahwa algoritma *Dijkstra* dapat memberikan solusi routing dengan bobot terkecil ke semua titik yang dituju, selain algoritma *Dijkstra* mampu memaksimalkan kerja protocol OSPF untuk dapat mengubah topologi apabila terdapat kegagalan link, dan perubahan dapat dilakukan dalam hitungan detik. Penelitian lainnya berjudul *dijkstra-based Terrain Generation Using Advanced Weight Functions* [9]. Kesimpulan dari penelitian ini adalah algoritma *dijkstra* dapat dimanfaatkan untuk pemetaan *visualisasi* bentang alam yang juga merupakan bagian dari sistem informasi geografis. Dalam hal ini algoritma *dijkstra* mampu memberikan informasi terkait ketinggian suatu dataran dengan memanfaatkan *node-node* yang terbentuk dari algoritma. Dalam penelitian *Path Optimization Study for Vehicles Evacuation Based on Dijkstra algorithm* [10]. Dengan hasil penelitian algoritma *dijkstra* digunakan untuk menentukan jalur terpendek untuk melakukan percepatan *evakuasi* dalam keadaan darurat. Dalam penentuan jalur terpendek memperhitungkan kepadatan lalu lintas oleh kendaraan sebagai salah faktor yang kemungkinan dapat menghambat *evakuasi*. Dari penelitian ini disimpulkan bahwa kepadatan lalu lintas mampu mempengaruhi *efisiensi evakuasi*. Dalam penelitian *Railway Route Optimization System Using Dijkstra Method* [11]. Disimpulkan dengan memanfaatkan algoritma *Dijkstra*, rute perjalanan kereta api mampu dipetakan sehingga dapat diketahui keberangkatan kereta di stasiun awal dan kemana stasiun tujuannya. Hal ini mampu memecahkan masalah *optimalisasi*, dimana masyarakat dimudahkan untuk mengetahui stasiun kereta mana saja, kereta api yang menuju tujuan mereka tersedia. Dalam penelitian *Geografic Dijkstra-Based Multicast Algorithm For Wireless Sensor Networks* [12], disimpulkan bahwa algoritma *dijkstra* bereperan dalam penentuan *routing* akses dalam jaringan komunikasi *multicast*. Dengan memanfaatkan *node-node* yang terbentuk mampu mengurangi beban *resource* yang diperlukan oleh NSW. Hasil penelitian ini menunjukkan bahwa algoritma *dijkstra* mampu memberikan *routing* yang efektif dan efisien. Dalam penelitian

Implementasi Algoritma Dijkstra Dalam Aplikasi Untuk Menentukan Lintasan Terpendek Jalan Darat Antar Kota Di Sumatera Bagian Selatan [13], kesimpulannya adalah menjelaskan bagaimana algoritma *dijkstra* dimanfaatkan dalam penentuan *rute* terpendek akses dari satu kota ke kota lainnya melalui jalur darat di provinsi Sumatera Selatan. Hasil dari penelitian ini menunjukkan bahwa algoritma *dijkstra* mampu memberikan *rute* terpendek yang *optimal* dengan bobot terkecil. Dalam penelitian *Implementasi Algoritma Dijkstra* Dalam Menemukan Jarak Terdekat Dari Lokasi Pengguna Ke Tanaman Yang Di Tuju Berbasis *Android* (Studi Kasus Di Kebun Raya Purwodadi) [14], ditarik kesimpulan Pada penelitian ini algoritma *Dijkstra* mampu memberikan solusi untuk melakukan pencarian lokasi tanaman yang ingin dituju dengan memeberikan jalur atau *rute* jalan yang *efektif* dengan *presentase* kepuasan *responden* mencapai 70%. Dalam penelitian *Anapplication of Dijkstra's Algorithm to shortest route problem* [16], dengan hasil *Algoritma Dijkstra* digunakan untuk memaksimalkan pendistribusian produk dari suatu perusahaan untuk dikirimkan ke setiap langganannya, dalam penelitian ini menunjukkan algoritma *Dijkstra* mampu memberikan jalur *distribusi* yang *efektif*, sehingga perusahaan menghemat waktu *distribusi* serta *cost* yang dikeluarkan. Dan terakhir dalam penelitian *Pencarian Spbu Terdekat Dan Penentuan Jarak Terpendek Menggunakan Algoritma Dijkstra* (Studi Kasus Di Kabupaten Jember) [17], ditarik kesimpulan penelitian ini menerapkan algoritma *dijkstra* untuk pencarian lokasi spbu terdekat dari tempat *user* selain itu, perhitungan dibagi menjadi dua yang diambil berdasarkan jenis kendaraan motor dan mobil. *Algoritma Dijkstra* mampu memberikan *rute* terbaik untuk menuju SPBU sesuai dengan jenis kendaraan yang digunakan.

3.1. Penjelasan sistem berjalan

Sistem berjalan dengan menggunakan *platform* berbasis website, dimana *user* perlu terhubung dengan jaringan internet untuk mengakses sistem ini. Proses berjalannya sistem digambarkan melalui sebuah *flowchart* sebagai berikut :



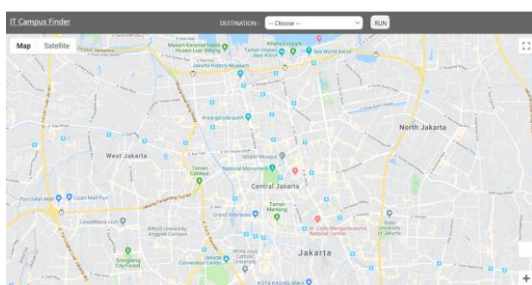
Gambar 3. Flowchart

Dari *flowchart* diatas dapat dijabarkan prosesnya sebagai berikut: Sistem mendeteksi posisi terakhir *user* saat menggunakan aplikasi ini, lalu Posisi ditampilkan diatas sebuah *map*. *User* memilih kampus tujuan pada menu yang telah disediakan. Setelah lokasi tujuan ditentukan sistem akan membuat *route* terpendek menuju lokasi melalui perhitungan algoritma *dijkstra*. Hasil perhitungan akan berupa *route* ditampilkan pada *map* untuk dapat digunakan *user*.

3.2 Implementasi sistem

Form Home

Merupakan tampilan utama pada saat *user* membuka website pencarian jalur terpendek menuju kampus IT di Jakarta.

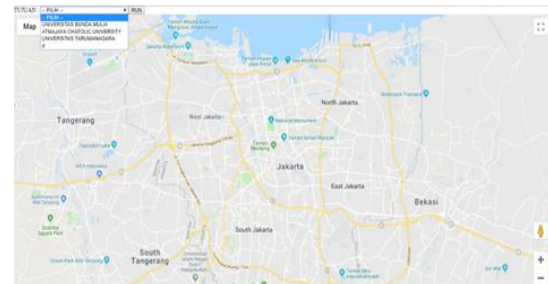
Gambar 4. Form home dengan *map* standar

Pada *home* *user* dapat menentukan lokasi awal dan lokasi kampus mana yang ingin di kunjungi dengan memilih pada menu pilih yang tersedia pada *map*. Lalu menekan *run* untuk

mendapatkan *route* menuju lokasi kampus tujuan.

Form Pilih Kampus IT

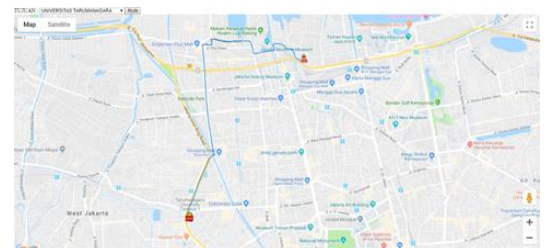
Tampilan data kampus IT yang ada di Jakarta dapat diakses pada menu pilih dengan melakukan klik satu kali pada menu tersebut, maka akan ditampilkan data kampus IT di Jakarta.



Gambar 5. Tampilan pilih kampus IT tujuan.

Form Tampilan Jalur Pilihan

Tampilan *route* yang harus dilalui oleh *user*

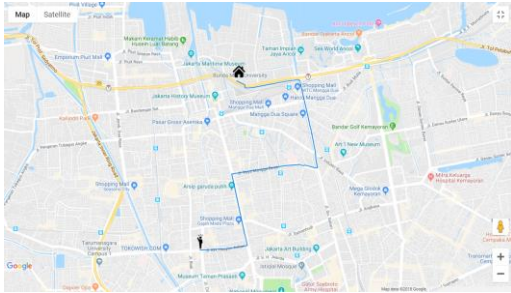
Gambar 6. Tampilan *route* yang dilalui dengan *map* standar

Route jalan yang akan didapatkan *user* ditandai dengan jalur berwarna biru pada *map*. Selain itu akan ditampilkan dua marker lainnya sebagai penanda lokasi *user* dan lokasi tujuan.

3.3. Hasil pengujian

Hasil Pengujian yang dilakukan untuk membuktikan bahwa *algoritma dijkstra* mampu memberikan solusi *optimum* jalur terpendek dalam penelitian ini, berikut penjabaran hasil pengujian.

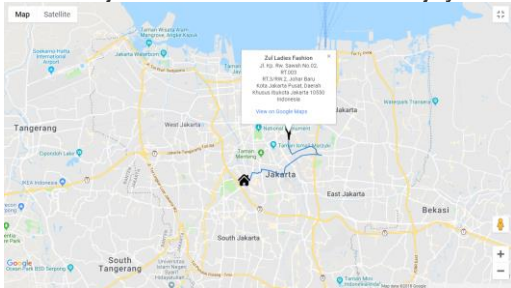
1. Bank Sinar Mas Petojo menuju Univeritas Bunda Mulia



Gambar 7. *Rute* dari Bank Sinar Mas menuju Universitas Bunda Mulia

Menunjukkan *rute* terpendek yang dapat dilalui dari Bank Sinar Mas Petojo menuju Univeritas Bunda Mulia.

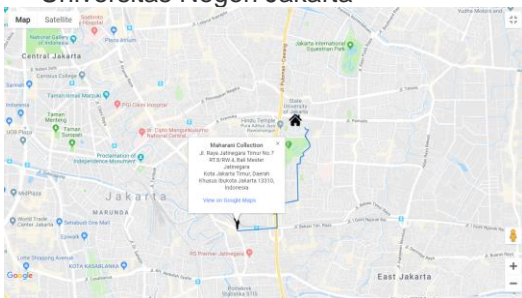
2. Zul Ladies Fashion Jl. Kp. Rw. Sawah menuju Universitas Katolik Atmajaya



Gambar 8. *Rute* dari Zul Ladies Fashion menuju Universitas Katolik Atmajaya

Menunjukkan *rute* terpendek yang dapat dilalui dari Zul Ladies Fashion Jl. Kp. Rw. Sawah menuju Univeritas Katolik Atmajaya.

3. Maharani Collection, Jatinegara menuju Universitas Negeri Jakarta



Gambar 9. *Rute* dari Maharani Collection menuju Universitas Negeri Jakarta

Menunjukkan *rute* terpendek yang dapat dilalui dari Maharani Collection, Jatinegara menuju Univeritas Negeri Jakarta.

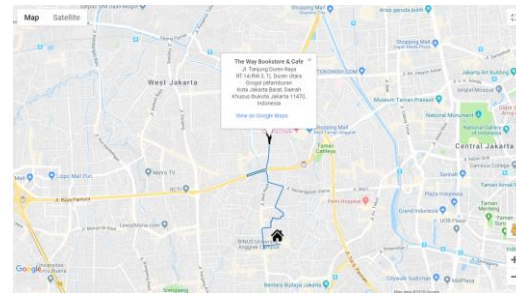
4. Bakso Sengon Kramat Jaya-Alidin Toyo menuju UNTAR



Gambar 10. *Rute* terpendek dari Alidin Toyo menuju UNTAR

Menunjukkan *rute* terpendek yang dapat dilalui dari Bakso Sengon Kramat Jaya-Alidin Toyo UNTAR.

5. The Way Bookstore & Café menuju BINUS



Gambar 11. *Rute* terpendek dari The Way Bookstore & Café menuju BINUS

Menunjukkan *rute* terpendek yang dapat dilalui dari The Way Bookstore & Café menuju BINUS.

Berikut adalah tabel hasil pengujian sistem yang sudah penulis buat berdasarkan data asli dari *map* :

Tabel 1 Tabel hasil pengujian

No	Titik awal	Kampus IT tujuan	Hasil
1	Bank Sinar MAS, Petojo	UBM	<i>Rute</i> terpendek
2	Bubur Ayam Pinangsia	UBM	<i>Rute</i> terpendek
3	RX Game Jl. Lodan Raya	UBM	<i>Rute</i> terpendek
4	RM Karya Minang Jl. Lodan Raya	UBM	<i>Rute</i> terpendek

5	PT Nipsea Paint And Chemicals Co Ltd	UBM Mulia	Rute terpendek
6	Seafood 212 BINTANG LAUT ANCOL	UBM	Rute terpendek
7	Zul Ladies Fashion Jl. Kp. Rw. Sawah	Atmajaya	Rute terpendek
8	Gado2 & Nasi Soto Ayam Karet	Atmajaya	Rute terpendek terpendek
9	Warung Sate Madura, Kebayoran	Atmajaya	Rute terpendek
10	An Hotel Jakarta, Kuningan	Atmajaya	Rute terpendek
11	Maharani Collection, Jatinegara	UNJ	Rute terpendek
12	Apotek Parantha Farma Jl. Cipinang Baru	UNJ	Rute terpendek
13	STMA Trisakti	UNJ	Rute terpendek
14	Sinar Garut Mekarsari 2	UNJ	Rute terpendek
15	Mendoan Cempaka Warna	UNJ	Rute terpendek
16	Bakso Sengon Kramat Jaya-Alidin Toyo	UNTAR	Rute terpendek
17	Evo Distro Jl. Gajah Mada	UNTAR	Rute terpendek
18	Masjid Al-Huda grogol	UNTAR	Rute terpendek
19	Youth Center GRII Daan Mogot	UNTAR	Rute terpendek
20	Kost Tropis Wanita Jl. Tanjung Duren	UNTAR	Rute terpendek
21	The Way Bookstore &	BINUS	Rute terpendek

	Cafe		
22	MIE AYAM BABE AHDA, Duri	BINUS	Rute terpendek
23	PT. SINAR PALASARI INDONESIA	BINUS	Rute terpendek
24	Taman Komplek DPR	BINUS	Rute terpendek
25	Bakso Rusuk Total Solo Kemanggisan	BINUS	Rute terpendek
26	Pisang Ijo Si Nona	Esa Unggul	Rute terpendek
27	WARTUR Intercone	Mercu Buana	Menunjukkan rute terpendek

Setelah melakukan pengujian penulis dapat menyimpulkan bahwa tingkat ketepatan *rute* yang didapatkan dari *aplikasi* ini adalah sebesar 75 %, dikarenakan masih ada beberapa posisi dan *rute* yang tidak 100 % akurat.

4. Kesimpulan

Berdasarkan permasalahan yang telah dijabarkan dan serangkaian penelitian yang telah penulis lakukan. Maka dapat disimpulkan beberapa hal sebagai berikut :

1. Cara menemukan *rute* tercepat dan *rute* terpendek dari suatu lokasi ke lokasi yang lain, dalam hal ini berupa suatu persimpangan ke persimpangan yang lain adalah dengan menggunakan *algoritma Dijkstra* dengan *input* berupa persimpangan asal dan tujuan serta sebuah *graf* yang merepresentasikan *node* sebagai persimpangan dan simpul atau path sebagai jalur yang menghubungkannya.
2. Faktor-faktor yang dapat diperhitungkan dalam mencari jalur tercepat yaitu waktu tempuh dari suatu persimpangan ke persimpangan lainnya.

Referensi

- Munir, Rinaldi. (2005), *Buku Teks Ilmu Komputer Matematika Diskrit Edisi Ketiga*, Informatika, Bandung, ISBN/ISSN : 979-96-446-3-1.
- Munir, Rinaldi. (2005), *Algoritma dan Pemrograman dalam Bahasa Pascal dan C Buku 1*, Informatika, Bandung, ISBN : 979-95779-2-6.

- Peranginangin, Kasiman. 2006. *Aplikasi Web dengan PHP dan MySQL*. Andi. Yogyakarta. ISBN: 979-763-526-0.
- Kristanto, Andri. (2003), *Struktur Data dengan C++*, Graha Ilmu, Yogyakarta, ISBN: 978-979-756-450-6. 1.
- Chamero, Juan. 2006. *Dijkstra's Algorithm As a Dynamic Programming strategy*, http://www.intag.org/downloads/ds_006.pdf, Diakses 10 Oktober 2018.
- Sjukani, Moh. (2007), *Algoritma (Algoritma dan Struktur Data 1) dengan C, C++, dan Java*, Mitra Wacana Media, Jakarta, ISBN/ISSN : 978-979-1092-29-6.
- Syahputra, Siswan. 2017, Penentuan Rute Terpendek Pendistribusian Naskah Ujian Nasional Menggunakan Algoritma Dijkstra, Jurnal Teknik Informatika Kaputama (JTIK), Vol 1 No.1, Januari 2017, ISSN :2548-9704.
- Udariansyah, Devi. 2018, Menggunakan Protokol Routing OSPF (Open Source Path First) Menggunakan GNS3 Untuk Mencari Jalur Terpendek Dengan Menerapkan Algoritma Dijkstra, Jurnal Informanika, Volume 4 No.1, ISSN :2407-1730.
- Kirill Golubev, Aleksander Zagarskikh, dan Andrey Karsakov. 2016. *Dijkstra-based Terrain Generation Using Advanced Weight Functions*, Procedia Computer Science 101, 2016, Pages 151 – 159, doi: 10.1016/j.procs.2016.11.019.
- Yi-zhou Chen, Shi-fei Shen, Tao Chen, dan Rui Yang. 2014. Path Optimization Study for Vehicles Evacuation Based on Dijkstra algorithm, Procedia Engineering 71 (2014) 159 – 165, doi: 10.1016/j.proeng.2014.04.023.
- Pramod Pandey dan Sunanda Dixit. 2014. Railway Route Optimization System Using Dijkstra Method, International Journal on Recent and Innovation Trends in Computing and Communication, ISSN: 2321-8169.
- Bartosz Musznicki, Mikołaj Tomczak, dan Piotr Zwierzykowski. 2013. *Geografic Dijkstra-Based Multicast Algorithm For Wireless Sensor Networks*, Image Processing & Communication, vol. 17, no. 1-2, pp. 33-46, doi : 10.2478/v10248-012-0013-3.
- Fitria, Apri Triansyah. 2013. Implementasi Algoritma Dijkstra Dalam Aplikasi Untuk Menentukan Lintasan Terpendek Jalan Darat Antar Kota Di Sumatera Bagian Selatan, Jurnal Sistem Informasi (JSI), VOL. 5, NO. 2, Oktober 2013, Halaman 611-621, ISSN Online : 2355-4614.
- Muhammad Syamsuddin Yusuf, Hanifah Muslimah Az-Zahra, Diah Harnoni Apriyanti. 2017. Implementasi Algoritma Dijkstra Dalam Menemukan Jarak Terdekat Dari Lokasi Pengguna Ke Tanaman Yang Di Tuju Berbasis Android (Studi Kasus di Kebun Raya Purwodadi, Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, Vol. 1, No. 12, Desember 2017, hlm. 1779-1787, e-ISSN: 2548-964X
- Edmonds, Jeff. 2008, *How to Think About Algorithm*, Cambridge University Press, New York. ISBN-13: 978-0521614108
- Ojekudo, Nathaniel Akpofure dan Akpan, Nsikan Paul. 2017. Anapplication of Dijkstra's Algorithm to shortest route problem, Volume 13, Issue 3 Ver. 1 (May. - June. 2017), PP 20-32, e-ISSN: 2278-5728.
- Windi Eka Yulia R., Dwiretno Istiadi, Abdul Roqib. 2015. Pencarian Spbu Terdekat Dan Penentuan Jarak Terpendek Menggunakan Algoritma Dijkstra (Studi Kasus Di Kabupaten Jember), Vol: 4, No. 1, Maret 2015, ISSN: 2302.