

INTEGRASI ALGORITMA DIJKSTRA PADA APLIKASI QGIS UNTUK SIMULASI RUTE TERCEPAT DI MEDAN (STUDI KASUS : RUTE DARI UNIVERSITAS NEGERI MEDAN KE RUMAH SAKIT ROYAL PRIMA)

Iwan Agi Berutu, Sybil Auzi, Salma Ashillah, Putri Harliana

Ilmu Komputer Universitas Negeri Medan
Jalan Wiliam Iskandar Pasar V Medan, Indonesia
iwanberutu962@gmail.com

ABSTRAK

Penelitian ini membahas penerapan algoritma Dijkstra dalam aplikasi QGIS untuk mensimulasikan rute tercepat di Medan, dengan studi kasus perjalanan dari Universitas Negeri Medan (UNIMED) ke Rumah Sakit Royal Prima. Data jaringan jalan diperoleh dari OpenStreetMap, sedangkan pengukuran jarak antar titik dilakukan menggunakan Google Maps untuk menentukan bobot pada setiap segmen jalan. Algoritma Dijkstra diterapkan untuk menghitung jalur terpendek berdasarkan bobot jarak tersebut, dan hasilnya divisualisasikan melalui QGIS untuk memastikan akurasi serta memberikan kemudahan interpretasi bagi pengguna. Penelitian ini menunjukkan bahwa algoritma Dijkstra mampu menentukan rute tercepat secara efektif, dengan tingkat akurasi yang tinggi meskipun diterapkan pada wilayah dengan jaringan jalan yang rumit. Visualisasi di QGIS tidak hanya menampilkan rute dengan jelas dan informatif, tetapi juga memungkinkan analisis lebih lanjut untuk berbagai kebutuhan navigasi. Studi ini diharapkan dapat menjadi referensi dalam pengembangan aplikasi berbasis pemetaan untuk berbagai keperluan, seperti manajemen transportasi, pengelolaan bencana, hingga pengoptimalan logistik, dengan integrasi algoritma pencarian jalur terpendek yang lebih luas dan efisien.

Kata kunci : Algoritma Dijkstra, GIS, QGIS, Jalur Terpendek, Visualisasi rute

1. PENDAHULUAN

Dengan pesatnya perkembangan teknologi informasi dan sistem pemetaan, kebutuhan akan akses cepat dan efisien dalam menentukan rute perjalanan semakin meningkat. Salah satu aplikasi yang sangat relevan dalam konteks ini adalah penggunaan algoritma Dijkstra, yang digunakan untuk menghitung jalur terpendek antar lokasi. Di perkotaan seperti Medan, rute tercepat sangat penting untuk memastikan perjalanan yang efisien, baik itu untuk transportasi pribadi, pengiriman barang, atau layanan darurat. Salah satu studi kasus yang menarik adalah penerapan algoritma Dijkstra untuk menentukan rute tercepat dari Universitas Negeri Medan ke Rumah Sakit Royal Prima. Dengan memanfaatkan aplikasi QGIS, algoritma ini dapat digunakan untuk mengoptimalkan rute perjalanan, meminimalkan waktu dan jarak tempuh, serta memberikan gambaran yang jelas mengenai pemetaan rute terpendek.

Algoritma Dijkstra dirancang untuk menyelesaikan masalah pencarian jalur terpendek pada graf berbobot, yang mana bobot tersebut bisa berupa jarak, waktu, atau biaya yang menggambarkan hubungan antar simpul. Dalam konteks penelitian ini, algoritma Dijkstra digunakan untuk mengoptimalkan rute perjalanan antara Universitas Negeri Medan dan Rumah Sakit Royal Prima, dengan mempertimbangkan waktu tempuh dan jarak sebagai bobot pada graf. Algoritma ini bekerja dengan memancarkan bobot dari simpul sumber, lalu secara bertahap memperbarui jalur menuju simpul tujuan. Menggunakan pendekatan greedy, setiap langkah algoritma memilih jalur

dengan bobot terkecil yang belum diproses, memastikan jalur terpendek ditemukan secara bertahap. Hal ini menjadikan algoritma Dijkstra sangat efektif untuk aplikasi pemetaan rute di perkotaan, termasuk dalam perencanaan perjalanan menggunakan sistem GIS, seperti yang diterapkan dalam simulasi rute tercepat dengan menggunakan QGIS pada studi kasus rute dari Universitas Negeri Medan ke Rumah Sakit Royal Prima.

Berdasarkan temuan-temuan tersebut, penelitian ini mengimplementasikan algoritma Dijkstra untuk mengoptimalkan rute pengiriman paket express di Palangka Raya, dengan fokus pada efisiensi waktu dan pengurangan biaya pengiriman. Sebagai bagian dari perkembangan teknologi informasi dan sistem pemetaan, pemanfaatan algoritma Dijkstra dalam perencanaan rute perjalanan telah menjadi fokus utama dalam meningkatkan efisiensi transportasi. Penggunaan algoritma ini memungkinkan perhitungan jalur terpendek yang dapat mengoptimalkan waktu dan biaya perjalanan, yang sangat penting untuk aplikasi sehari-hari seperti navigasi dan pengiriman barang. Dengan adanya aplikasi Geographic Information System (GIS) seperti QGIS, penerapan algoritma Dijkstra semakin mudah diakses dan diimplementasikan untuk analisis rute dalam konteks perkotaan.

Penelitian ini bertujuan untuk mengintegrasikan algoritma Dijkstra dalam aplikasi QGIS untuk mensimulasikan rute tercepat di Medan, dengan studi kasus rute dari Universitas Negeri Medan ke Rumah Sakit Royal Prima. Fokus utama penelitian ini adalah untuk mengevaluasi efektivitas algoritma Dijkstra dalam menentukan jalur terpendek, serta untuk

menganalisis durasi waktu yang dibutuhkan dalam memvisualisasikan rute menggunakan teknologi GIS. Diharapkan hasil penelitian ini dapat memberikan solusi praktis dalam optimasi rute perjalanan, serta memperkaya penerapan algoritma Dijkstra dalam sistem pemetaan dan transportasi

2. TINJAUAN PUSTAKA

Menurut penelitian sebelumnya, Apriadi dkk. yang membahas penentuan jalur terpendek dan waktu tempuh tercepat bagi pengemudi truk BBM ke SPBU di Pal Lima menggunakan algoritma Dijkstra. Algoritma ini bekerja dengan menentukan simpul awal dan tujuan, kemudian memilih simpul dengan bobot terkecil hingga semua simpul dievaluasi. Hasilnya, jalur terpendek melalui Tol Kapuas 2, Jl. Adi Sucipto, hingga Pal V dengan jarak 12,768 km. Jalur tercepat melalui Tol Kapuas 2, Jl. Mayor Aliyang, hingga Pal V dengan waktu tempuh 27 menit. Penelitian ini menunjukkan efektivitas algoritma Dijkstra untuk optimalisasi logistic[1].

Sejalan dengan itu, E. C. Rufus dkk. menunjukkan bahwa algoritma Dijkstra efektif untuk menentukan rute terpendek antar lokasi menggunakan graf berbobot positif dan diterapkan dalam sistem GIS [2]. Sementara itu, penelitian oleh Firwanda dkk. menguji algoritma Dijkstra untuk menentukan rute terpendek dengan menggunakan perhitungan Excel, Python, dan R. Hasilnya menunjukkan rute terpendek dari Politeknik Pos Indonesia ke Badan Pusat Statistik Kota Bandung sejauh 10.24 km, melalui titik-titik seperti N1, N2, dan seterusnya [3].

2.1. Teori Graf

Teori graf merupakan bidang yang sudah ada sejak lama, namun tetap relevan dengan banyak penerapan dalam kehidupan sehari-hari hingga saat ini. Banyak masalah dunia nyata yang sebenarnya dapat direpresentasikan secara visual dalam bentuk graf. Salah satu contoh representasi graf adalah peta. Dari peta tersebut, kita dapat menggali berbagai informasi, seperti menentukan jalur terpendek antara dua lokasi, menggambarkan dua kota yang saling berdekatan dengan warna berbeda, merencanakan jalur transportasi, mengatur jaringan komunikasi atau internet, dan masih banyak lagi. Selain peta, masih banyak contoh lain di dunia nyata yang bisa direpresentasikan sebagai graf. Secara matematis, graf didefinisikan sebagai pasangan himpunan (V, E) , yang ditulis dengan notasi $G = (V, E)$, dimana V adalah himpunan yang berisi simpul-simpul (vertex atau simpul) dan E adalah himpunan sisi (edge) yang menghubungkan pasangan simpul

Untuk menggambarkan ruas-ruas jalan yang ada, salah satu alternatif yang dapat digunakan adalah graf. Graf sangat efektif untuk menggambarkan objek-objek diskrit serta hubungan antar objek tersebut [4]. Penerapan algoritma

Dijkstra dapat diterapkan pada graf berarah maupun graf berbobot. Menurut Rosen, graf dengan bobot pada sisinya berguna untuk menyelesaikan berbagai masalah, seperti menemukan jalur terpendek antara dua kota dalam jaringan transportasi. Selain itu, graf berbobot juga dapat digunakan untuk menentukan jalur terpendek antara satu tempat dengan tempat lainnya. Dalam penelitian ini, ruas-ruas jalan akan digambarkan dalam bentuk graf berbobot, dengan tujuan untuk mempermudah interpretasi dan analisis jalur terpendek yang dapat diambil.

2.2. Algoritma

Istilah *algoritma* (algorithm) berasal dari kata *algorism*, yang diambil dari nama seorang penulis Arab terkenal, yaitu Abu Ja'far Muhammad ibnu Musa Al-Khuwarizmi. Nama Al-Khuwarizmi kemudian diadaptasi oleh orang Barat menjadi *algorism*. Algoritma sendiri adalah serangkaian langkah atau prosedur yang digunakan untuk menyelesaikan suatu masalah. Tanpa disadari, algoritma sering diterapkan dalam kehidupan sehari-hari, mulai dari rutinitas pagi hingga saat kita kembali tidur. Secara umum, algoritma merujuk pada urutan langkah-langkah yang sistematis untuk mencapai solusi [5]. Selain diterapkan dalam rutinitas sehari-hari, algoritma juga berfungsi sebagai pendukung dalam menjalankan berbagai sistem. Salah satu penerapannya adalah algoritma untuk menemukan rute terpendek, seperti Algoritma Dijkstra yang dirancang untuk menentukan jalur terpendek. Namun, penulisan algoritma seringkali sulit dipahami, baik dari segi maksud maupun implementasinya. Untuk mempermudah, dapat digunakan notasi algoritma, yaitu format penulisan yang merepresentasikan langkah-langkah penyelesaian masalah. Ada tiga jenis notasi algoritma yang umum digunakan, yaitu diagram alir (flowchart), deskriptif, dan *pseudo-code*.

2.3. Algoritma Dijkstra

Algoritma Dijkstra pertama kali diperkenalkan oleh Edsger Wybe Dijkstra pada tahun 1959. Algoritma ini dirancang untuk menyelesaikan masalah pencarian jalur terpendek dalam graf, di mana setiap simpul memiliki nilai yang tidak negatif. Dijkstra termasuk dalam kategori algoritma *greedy*, yang umum digunakan untuk menyelesaikan masalah optimasi. Dengan Algoritma Dijkstra, berbagai persoalan pencarian rute terpendek dapat diatasi secara efektif. Algoritma ini mengandalkan pendekatan langkah demi langkah untuk mencapai solusi optimal dalam graf yang memenuhi kriteria tersebut [6]. Algoritma Dijkstra dalam pencarian jalur terpendek berfokus pada menemukan bobot terkecil dalam graf berbobot. Jarak terpendek antara dua atau lebih titik dalam graf dihitung, dan total nilai yang diperoleh adalah yang paling minimal. Misalnya, jika G adalah graf berarah berlabel dengan himpunan titik $V(G) = \{v_1, v_2, \dots, v_n\}$, maka jalur terpendek yang dicari adalah dari v_1 ke v_n .

Penerapan Algoritma Dijkstra dapat digunakan untuk menyelesaikan berbagai jenis pencarian lintasan terpendek. Ini meliputi pencarian lintasan terpendek antara dua simpul tertentu, yang dikenal sebagai *a pair shortest path*; pencarian lintasan terpendek di antara semua pasangan simpul, atau *all pairs shortest path*; pencarian lintasan terpendek dari satu simpul ke semua simpul lainnya, disebut *single-source shortest path*; serta pencarian lintasan terpendek antara dua simpul yang melewati beberapa simpul tertentu, yang dikenal sebagai *intermediate shortest path*. Algoritma Dijkstra didasarkan pada teori graf, yang dibagi menjadi dua jenis utama. Graf tidak berarah adalah graf di mana sisi-sisinya tidak memiliki arah, sehingga pasangan simpul yang terhubung oleh sisi tidak memedulikan urutan, seperti $(u, v) = (v, u)$. Sebaliknya, Graf berarah merupakan graf yang setiap sisinya memiliki orientasi arah, sisi pada graf berarah disebut busur. Graf berarah memiliki sisi yang diberi orientasi arah, sehingga (u, v) dan (v, u) dianggap sebagai sisi yang berbeda [7]. Algoritma Dijkstra sangat populer dalam menyelesaikan masalah pencarian lintasan terpendek. Metode ini bermanfaat ketika kita hendak mencari jarak terpendek antara dua simpul dalam graf yang memiliki nilai bobot positif [8]

2.4. SIG/GIS

Sistem Informasi Geografis (SIG) adalah suatu sistem berbasis komputer yang dirancang untuk mengumpulkan, menyimpan, menggabungkan, mengatur, mentransformasi, memanipulasi, dan menganalisis data yang memiliki referensi geografis. SIG memungkinkan pemrosesan data spasial untuk berbagai tujuan analisis dan pemetaan, sehingga dapat digunakan untuk mendukung keputusan dalam berbagai bidang, seperti perencanaan kota, manajemen sumber daya alam, dan sistem transportasi. SIG menggunakan data spasial yang merujuk pada posisi dan hubungan antar objek di ruang bumi, seperti permukaan bumi, perairan, dan atmosfer. Data spasial ini disimpan menggunakan dua model utama, yaitu model raster dan model vektor. Model data raster menyimpan informasi dalam bentuk matriks atau piksel dengan akurasi bergantung pada resolusi spasialnya, cocok untuk menggambarkan fenomena kontinu. Sementara itu, model data vektor menggunakan titik, garis, dan poligon untuk menggambarkan objek spasial, dengan fleksibilitas tinggi dalam merepresentasikan objek berdimensi satu atau dua dan menyertakan atribut karakteristik objek tersebut [9].

2.5. Google Maps

Google maps adalah aplikasi layanan gratis yang disediakan oleh perusahaan Google yang dikhususkan untuk melayani kebutuhan peta tidak hanya itu banyak sekali fitur-fitur yang terdapat pada peta yang diberikan seperti lokasi saat ini dengan (GPS), lokasi wisata, tempat makan dan juga

perhitungan jarak antar tempat. Google Maps adalah peta digital yang dapat diakses secara online tanpa biaya, disediakan oleh Google melalui laman <http://peta.google.com>. Google Maps digunakan sebagai parameter untuk mengukur keoptimalan karena peta digital ini sangat populer di kalangan masyarakat. Dengan adanya kecerdasan buatan dan algoritma pencarian rute terdekat, Google Maps mempermudah pencarian lokasi dan rute tercepat, sehingga lebih efisien dalam menghemat waktu dan tenaga [10]. Selain itu, fitur interaktif seperti gambar satelit, navigasi waktu nyata, dan opsi mode transportasi menjadikan Google Maps alat yang sangat berguna dalam kehidupan sehari-hari, baik untuk perjalanan pribadi maupun kebutuhan profesional.

3. METODE PENELITIAN

3.1. Desain Penelitian

Pada penelitian ini, kami mengadopsi pendekatan kuantitatif dengan metode eksperimen yang didukung oleh analisis komputasional untuk mengoptimalkan rute menggunakan algoritma Dijkstra dalam konteks pencarian jalur terpendek di aplikasi GIS. Tahap awal penelitian dimulai dengan identifikasi lokasi UNIMED sebagai titik awal dan Rumah Sakit Royal Prima sebagai titik tujuan, diikuti dengan survei jalur-jalan yang menghubungkan kedua titik tersebut. Selanjutnya, kami melakukan studi literatur untuk memahami implementasi algoritma Dijkstra, lalu mengolah data yang terkumpul untuk membangun model graf jaringan dan mengimplementasikan algoritma Dijkstra untuk menghitung rute optimal. Hasil perhitungan jalur terpendek kemudian divisualisasikan menggunakan QGIS untuk menganalisis kemudahan aplikasi dalam menampilkan hasil pencarian jalur.

3.2. Pengumpulan Data

Penelitian ini menggunakan data dari jaringan jalan antara UNIMED dan Rumah Sakit Royal Prima, yang diambil dari Google Maps. Data ini mencakup jarak antara titik-titik yang terhubung dalam jaringan jalan, serta titik-titik persimpangan yang menjadi titik simpul dalam graf. Dengan proses pengumpulan datanya sebagai berikut :

- Mencari Rute: Kami menggunakan Google Maps untuk mencari rute antara UNIMED dan Rumah Sakit Royal Prima. Kami juga mencatat jarak dan Koordinat (longitude dan latitude) untuk setiap titik persimpangan dan segmen jalan.
- Mengidentifikasi Simpul dan Edge: Dalam grafik, titik persimpangan/lokasi di sepanjang rute yang ditemukan disebut sebagai simpul. Hubungan antar simpul digambarkan sebagai edge dengan bobot, yang dapat menunjukkan jarak atau waktu tempuh.

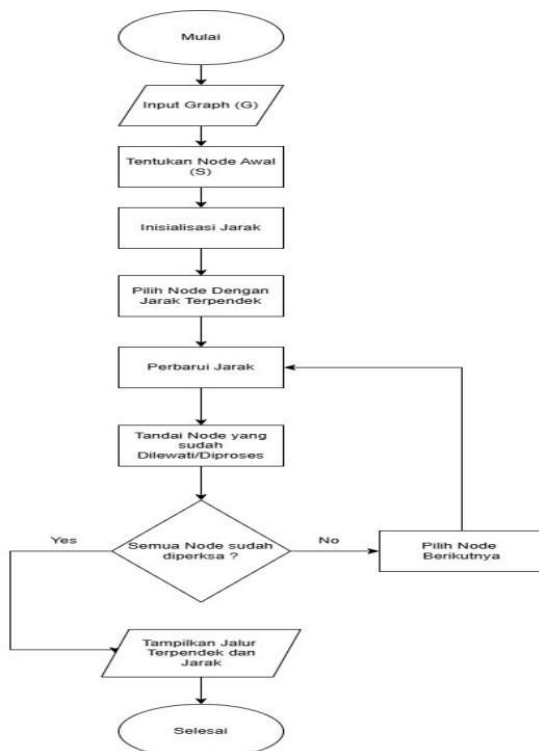
3.3. Penerapan Algoritma Dijkstra

Algoritma Dijkstra adalah sebuah algoritma untuk menyelesaikan permasalahan lintasan terpendek pada graf dengan bobot pada sisi yang bersifat positif. Algoritma Dijkstra bekerja dengan prinsip greedy untuk mencari rute terpendek dari satu simpul ke simpul lainnya dengan dalam graf berbobot.

Cara kerja algoritma Dijkstra:

- inisialisasi : Tentukan jarak dari sumber s ke semua simpul lainnya sebagai tak hingga (∞) kecuali untuk simpul sumber yang di set 0 ($d(s) = 0$).
- Pilih simpul dengan arak terdekat : Pilih simpul u dengan jarak terkecil yang belum di kunjungi, kemudian tandai simpul tersebut sebagai sudah di kunjungi.
- Perbaiki jarak ke simpul tetangga : Untuk setiap tetangga v dari simpul u , perbarui jarak $d(v)$ menggunakan rumus:
 - $d(v) = \min(d(v), d(u) + w(u,v))$
 - Dimana $w(u,v)$ adalah bobot antara simpul u dan v
- Ulangi : Proses ini berlanjut hingga semua simpul telah di kunjungi atau semua simpul tercapai.

Setelah algoritma ini di jalankan jalur terpendek ke setiap simpul dapat di temukan dan di gunakan untuk menentukan jalur terbaik antara Universita Negeri Medan ke Rumah Sakit Royal Prima. Untuk lebih jelas, berikut adalah flowchartnya.



Gambar 1. Diagram Alur Algoritma Dijkstra

3.4. Simulasi dan Visualisasi

Hasil perhitungan algoritma Dijkstra kemudian divisualisasikan menggunakan aplikasi QGIS. Rute terpendek yang ditemukan akan ditampilkan dalam bentuk peta yang memperlihatkan jalur dari UNIMED menuju Rumah Sakit Royal Prima. Setiap simpul dan edge pada graf digambarkan di dalam peta, dengan jalur terpendek diberi warna atau tanda tertentu agar lebih mudah diidentifikasi. Selain itu, informasi mengenai jarak dan waktu tempuh juga dapat ditambahkan pada peta untuk memberikan gambaran yang lebih rinci.

3.5. Analisis Data

Setelah rute terpendek ditemukan dan divisualisasikan, langkah selanjutnya adalah menganalisis hasilnya. Analisis akan berfokus pada efektivitas algoritma Dijkstra dalam menemukan jalur terpendek dan kemudahan visualisasi hasilnya di QGIS serta efektivitas visualisasi hasil perhitungan di QGIS. Proses ini juga mencakup penilaian terhadap akurasi jalur yang dihitung, dengan membandingkan jalur terpendek yang ditemukan dengan data rute alternatif yang tersedia di lapangan.

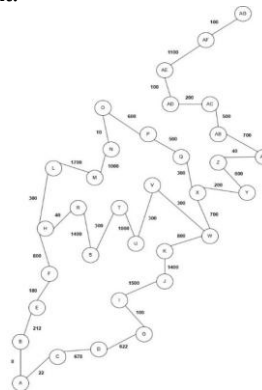
3.6. Evaluasi dan Kesimpulan

Evaluasi dilakukan untuk menilai efektivitas algoritma Dijkstra menghitung jalur terpendek dalam jaringan jalan yang lebih besar, serta untuk mengukur kualitas visualisasi hasil di QGIS. Fokus utama evaluasi ini adalah akurasi hasil pencarian jalur terpendek dan kemampuan QGIS dalam menampilkan hasil tersebut dengan baik. Berdasarkan evaluasi ini, kesimpulan akan diambil mengenai sejauh mana algoritma Dijkstra dapat diterapkan pada jaringan jalan di Medan dan seberapa optimal QGIS dalam memvisualisasikan hasil pencarian jalur.

4. HASIL DAN PEMBAHASAN

4.1. Pemaparan Algoritma

Algoritma Dijkstra diterapkan untuk menghitung rute terbaik melalui graf berbobot yang menggambarkan jaringan jalan antara kedua titik tersebut dengan tujuan untuk menentukan jalur tercepat dan terpendek berdasarkan data yang diperoleh sebagai berikut.



Gambar 2 . Representasi Hubungan Jarak dalam sebuah Graf

Penelitian ini menggunakan data jarak antar node yang disusun dalam format tabel untuk mempermudah analisis graf. Data tersebut mencantumkan tiga komponen utama, yaitu:

- 1) Node asal (Dari): Node tempat perjalanan dimulai.
- 2) Node tujuan (Ke): Node tempat perjalanan berakhir.
- 3) Jarak (Bobot): Nilai jarak atau bobot yang menunjukkan hubungan antara node asal dan tujuan.

Tabel 1 . Data jarak antar titik

Dari	Ke	Jarak (m)
A	B	8
A	C	22
B	E	180
C	D	678
D	G	622
E	F	180
F	H	800
G	I	100
H	L	300
H	R	40
I	J	1500
J	K	1400
K	W	800
L	M	1700
M	N	1000
N	O	10
O	P	600
P	Q	500
Q	X	300
R	S	1400
S	T	300
T	U	1000
U	V	300
V	W	300
W	X	700
X	Y	200
Y	Z	600
Z	AA	40
AA	AB	700
AB	AC	500
AC	AD	200
AD	AE	100
AE	AF	1100
AF	AG	100

Kemudian Langkah-Langkah Algoritma Dijkstra dari Unimed ke RS Royal Prima dengan perhitungan manual sebagai berikut. Kita mulai dari node Unimed (A) dan akan mencari jarak terpendek ke Gerbang Masuk RS. Royal Prima (AG). Berikut adalah langkah pertama perhitungannya:

Inisialisasi:

A: 0, B: ∞ , C: ∞ , D: ∞ , E: ∞ , F: ∞ , G: ∞ , H: ∞ , I: ∞ , J: ∞ , K: ∞ , L: ∞ , M: ∞ , N: ∞ , O: ∞ , P: ∞ , Q: ∞ , R: ∞ , S: ∞ , T: ∞ , U: ∞ , V: ∞ , W: ∞ , X: ∞ , Y: ∞ , Z: ∞ , AA: ∞ , AB: ∞ , AC: ∞ , AD: ∞ , AE: ∞ , AF: ∞ , AG: ∞

Iterasi 1 (Mulai dari A: Unimed):

Dari A, periksa tetangga-tetangga yang terhubung langsung:

- $A \rightarrow B: 8$
- $A \rightarrow C: 22$

Jarak terupdate: A: 0, B: 8, C: 22, D: ∞ , E: ∞ , F: ∞ , G: ∞ , H: ∞ , I: ∞ , J: ∞ , K: ∞ , L: ∞ , M: ∞ , N: ∞ , O: ∞ , P: ∞ , Q: ∞ , R: ∞ , S: ∞ , T: ∞ , U: ∞ , V: ∞ , W: ∞ , X: ∞ , Y: ∞ , Z: ∞ , AA: ∞ , AB: ∞ , AC: ∞ , AD: ∞ , AE: ∞ , AF: ∞ , AG: ∞

Simpul A ditandai sebagai selesai.

Iterasi 2 (Dari B: Jl. Selamat Ketaren):

Dari B, periksa tetangga-tetangga yang terhubung:

- $B \rightarrow E: 8 + 212 = 220$

Jarak terupdate: A: 0, B: 8, C: 22, D: ∞ , E: 220, F: ∞ , G: ∞ , H: ∞ , I: ∞ , J: ∞ , K: ∞ , L: ∞ , M: ∞ , N: ∞ , O: ∞ , P: ∞ , Q: ∞ , R: ∞ , S: ∞ , T: ∞ , U: ∞ , V: ∞ , W: ∞ , X: ∞ , Y: ∞ , Z: ∞ , AA: ∞ , AB: ∞ , AC: ∞ , AD: ∞ , AE: ∞ , AF: ∞ , AG: ∞

Simpul B ditandai sebagai selesai.

Iterasi 3 (Dari C: Jl. Selamat Ketaren (arah RS. Haji)):

Dari C, periksa tetangga-tetangga yang terhubung:

- $C \rightarrow D: 22 + 678 = 700$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: ∞ , G: ∞ , H: ∞ , I: ∞ , J: ∞ , K: ∞ , L: ∞ , M: ∞ , N: ∞ , O: ∞ , P: ∞ , Q: ∞ , R: ∞ , S: ∞ , T: ∞ , U: ∞ , V: ∞ , W: ∞ , X: ∞ , Y: ∞ , Z: ∞ , AA: ∞ , AB: ∞ , AC: ∞ , AD: ∞ , AE: ∞ , AF: ∞ , AG: ∞

Simpul C ditandai sebagai selesai.

Iterasi 4 (Dari E: Jl. William Iskandar Ps.V):

Dari E, periksa tetangga-tetangga yang terhubung:

- $E \rightarrow F: 220 + 180 = 400$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: ∞ , H: ∞ , I: ∞ , J: ∞ , K: ∞ , L: ∞ , M: ∞ , N: ∞ , O: ∞ , P: ∞ , Q: ∞ , R: ∞ , S: ∞ , T: ∞ , U: ∞ , V: ∞ , W: ∞ , X: ∞ , Y: ∞ , Z: ∞ , AA: ∞ , AB: ∞ , AC: ∞ , AD: ∞ , AE: ∞ , AF: ∞ , AG: ∞

Simpul E ditandai sebagai selesai.

Iterasi 5 (Dari F: Jl. William Iskandar Ps.V (Putaran U)):

Dari F, periksa tetangga-tetangga yang terhubung:

- $F \rightarrow H: 400 + 800 = 1200$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: ∞ , H: 1200, I: ∞ , J: ∞ , K: ∞ , L: ∞ , M: ∞ , N: ∞ , O: ∞ , P: ∞ , Q: ∞ , R: ∞ , S: ∞ , T: ∞ , U: ∞ , V: ∞ , W: ∞ , X: ∞ , Y: ∞ , Z: ∞ , AA: ∞ , AB: ∞ , AC: ∞ , AD: ∞ , AE: ∞ , AF: ∞ , AG: ∞

Simpul F ditandai sebagai selesai.

Iterasi 6 (Dari D: Jl. Rumah Sakit Haji):

Dari D, periksa tetangga-tetangga yang terhubung:

- $D \rightarrow G: 700 + 622 = 1322$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞ , J: ∞ , K: ∞ , L: ∞ , M: ∞ , N: ∞ , O: ∞ , P: ∞ , Q: ∞ , R: ∞ , S: ∞ , T: ∞ , U: ∞ , V: ∞

W: ∞, X: ∞, Y: ∞, Z: ∞, AA: ∞, AB: ∞, AC: ∞, AD: ∞, AE: ∞, AF: ∞, AG: ∞

Simpul D ditandai sebagai selesai.

Iterasi 7 (Dari E: Jl. William Iskandar Ps.V):

Sudah selesai sebelumnya. Lanjutkan ke simpul dengan jarak terkecil berikutnya, yaitu H (1200).

Iterasi 8 (Dari H: Jl. William Iskandar (Simpang Ps.V)):

Dari H, periksa tetangga-tetangga yang terhubung:

- $H \rightarrow R: 1200 + 40 = 1240$

Jarak terupdate:

A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞, J: ∞, K: ∞, L: 1500, M: ∞, N: ∞, O: ∞, P: ∞, Q: ∞, R: 1240, S: ∞, T: ∞, U: ∞, V: ∞, W: ∞, X: ∞, Y: ∞, Z: ∞, AA: ∞, AB: ∞, AC: ∞, AD: ∞, AE: ∞, AF: ∞, AG: ∞

Simpul H ditandai sebagai selesai.

Iterasi 9 (Dari R: Jl. Keruntung):

Dari R, periksa tetangga-tetangga yang terhubung:

- $R \rightarrow S: 1240 + 1400 = 2640$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞, J: ∞, K: ∞, L: 1500, M: ∞, N: ∞, O: ∞, P: ∞, Q: ∞, R: 1240, S: 2640, T: ∞, U: ∞, V: ∞, W: ∞, X: ∞, Y: ∞, Z: ∞, AA: ∞, AB: ∞, AC: ∞, AD: ∞, AE: ∞, AF: ∞, AG: ∞

Simpul R ditandai sebagai selesai.

Iterasi 10 (Dari S: Jl. Permai):

Dari S, periksa tetangga-tetangga yang terhubung:

- $S \rightarrow T: 2640 + 300 = 2940$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞, J: ∞, K: ∞, L: 1500, M: ∞, N: ∞, O: ∞, P: ∞, Q: ∞, R: 1240, S: 2640, T: 2940, U: ∞, V: ∞, W: ∞, X: ∞, Y: ∞, Z: ∞, AA: ∞, AB: ∞, AC: ∞, AD: ∞, AE: ∞, AF: ∞, AG: ∞

Simpul S ditandai sebagai selesai.

Iterasi 11 (Dari T: Jl. Pelita 4):

Dari T, periksa tetangga-tetangga yang terhubung:

- $T \rightarrow U: 2940 + 1000 = 3940$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞, J: ∞, K: ∞, L: 1500, M: ∞, N: ∞, O: ∞, P: ∞, Q: ∞, R: 1240, S: 2640, T: 2940, U: 3940, V: ∞, W: ∞, X: ∞, Y: ∞, Z: ∞, AA: ∞, AB: ∞, AC: ∞, AD: ∞, AE: ∞, AF: ∞, AG: ∞

Simpul T ditandai sebagai selesai.

Iterasi 12 (Dari U: Jl. Cahaya):

Dari U, periksa tetangga-tetangga yang terhubung:

- $U \rightarrow V: 3940 + 300 = 4240$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞, J: ∞, K: ∞, L: 1500, M: ∞, N: ∞, O: ∞, P: ∞, Q: ∞, R: 1240, S: 2640, T: 2940, U: 3940, V: 4240, W: ∞, X: ∞, Y: ∞, Z: ∞, AA: ∞, AB: ∞, AC: ∞, AD: ∞, AE: ∞, AF: ∞, AG: ∞

Simpul U ditandai sebagai selesai

Iterasi 13 (Dari V: Jl. Mozasa):

Dari V, periksa tetangga-tetangga yang terhubung:

- $V \rightarrow W: 4240 + 300 = 4540$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞, J: ∞, K: ∞, L: 1500, M: ∞, N: ∞, O: ∞, P: ∞, Q: ∞, R: 1240, S: 2640, T: 2940, U: 3940, V: 4240, W: 4540, X: ∞, Y: ∞, Z: ∞, AA: ∞, AB: ∞, AC: ∞, AD: ∞, AE: ∞, AF: ∞, AG: ∞

Simpul V ditandai sebagai selesai.

Iterasi 14 (Dari W: Jl. KH. Syekh Abdul Wahab Rokan):

Dari W, periksa tetangga-tetangga yang terhubung:

- $W \rightarrow X: 4540 + 700 = 5240$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞, J: ∞, K: ∞, L: 1500, M: ∞, N: ∞, O: ∞, P: ∞, Q: ∞, R: 1240, S: 2640, T: 2940, U: 3940, V: 4240, W: 4540, X: 5240, Y: ∞, Z: ∞, AA: ∞, AB: ∞, AC: ∞, AD: ∞, AE: ∞, AF: ∞, AG: ∞

Simpul W ditandai sebagai selesai.

Iterasi 15 (Dari X: Jl. KL. Yos Sudarso):

Dari X, periksa tetangga-tetangga yang terhubung:

- $X \rightarrow Y: 5240 + 200 = 5440$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞, J: ∞, K: ∞, L: 1500, M: ∞, N: ∞, O: ∞, P: ∞, Q: ∞, R: 1240, S: 2640, T: 2940, U: 3940, V: 4240, W: 4540, X: 5240, Y: 5440, Z: ∞, AA: ∞, AB: ∞, AC: ∞, AD: ∞, AE: ∞, AF: ∞, AG: ∞

Simpul X ditandai sebagai selesai.

Iterasi 16 (Dari Y: Jl. H. Adam Malik):

Dari Y, periksa tetangga-tetangga yang terhubung:

- $Y \rightarrow Z: 5440 + 600 = 6040$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞, J: ∞, K: ∞, L: 1500, M: ∞, N: ∞, O: ∞, P: ∞, Q: ∞, R: 1240, S: 2640, T: 2940, U: 3940, V: 4240, W: 4540, X: 5240, Y: 5440, Z: 6040, AA: ∞, AB: ∞, AC: ∞, AD: ∞, AE: ∞, AF: ∞, AG: ∞

Simpul Y ditandai sebagai selesai.

Iterasi 17 (Dari Z: Adipura Monument):

Dari Z, periksa tetangga-tetangga yang terhubung:

- $Z \rightarrow AA: 6040 + 40 = 6080$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞, J: ∞, K: ∞, L: 1500, M: ∞, N: ∞, O: ∞, P: ∞, Q: ∞, R: 1240, S: 2640, T: 2940, U: 3940, V: 4240, W: 4540, X: 5240, Y: 5440, Z: 6040, AA: 6080, AB: ∞, AC: ∞, AD: ∞, AE: ∞, AF: ∞, AG: ∞

Simpul Z ditandai sebagai selesai.

Iterasi 18 (Dari AA: Adipura Monument (Jl.KH.Adam Malik)):

Dari AA, periksa tetangga-tetangga yang terhubung:

- $AA \rightarrow AB: 6080 + 700 = 6780$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞, J: ∞, K: ∞, L: 1500, M: ∞, N: ∞, O: ∞, P: ∞, Q: ∞, R: 1240, S: 2640, T: 2940,

U: 3940, V: 4240, W: 4540, X: 5240, Y: 5440, Z: 6040, AA: 6080, AB: 6780, AC: ∞ , AD: ∞ , AE: ∞ , AF: ∞ , AG: ∞

Simpul AA ditandai sebagai selesai.

Iterasi 19 (Dari AB: Jl. Sikambang):

Dari AB, periksa tetangga-tetangga yang terhubung:

- AB $\rightarrow$ AC: $6780 + 500 = 7280$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞ , J: ∞ , K: ∞ , L: 1500, M: ∞ , N: ∞ , O: ∞ , P: ∞ , Q: ∞ , R: 1240, S: 2640, T: 2940, U: 3940, V: 4240, W: 4540, X: 5240, Y: 5440, Z: 6040, AA: 6080, AB: 6780, AC: 7280, AD: ∞ , AE: ∞ , AF: ∞ , AG: ∞

Simpul AB ditandai sebagai selesai.

Iterasi 20 (Dari AC: Jl. Damar):

Dari AC, periksa tetangga-tetangga yang terhubung:

- AC $\rightarrow$ AD: $7280 + 200 = 7480$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞ , J: ∞ , K: ∞ , L: 1500, M: ∞ , N: ∞ , O: ∞ , P: ∞ , Q: ∞ , R: 1240, S: 2640, T: 2940, U: 3940, V: 4240, W: 4540, X: 5240, Y: 5440, Z: 6040, AA: 6080, AB: 6780, AC: 7280, AD: 7480, AE: ∞ , AF: ∞ , AG: ∞

Simpul AC ditandai sebagai selesai.

Iterasi 21 (Dari AD: Jl. Sekip):

Dari AD, periksa tetangga-tetangga yang terhubung:

- AD $\rightarrow$ AE: $7480 + 100 = 7580$

Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, I: ∞ , J: ∞ , K: ∞ , L: 1500, M: ∞ , N: ∞ , O: ∞ , P: ∞ , Q: ∞ , R: 1240, S: 2640, T: 2940, U: 3940, V: 4240, W: 4540, X: 5240, Y: 5440, Z: 6040, AA: 6080, AB: 6780, AC: 7280, AD: 7480, AE: 7580, AF: ∞ , AG: ∞

Simpul AD ditandai sebagai selesai.

Iterasi 22 (Dari AE: Jl. Pabrik Tenun):

- AE $\rightarrow$ AF: $7580 + 1100 = 8680$
- AE $\rightarrow$ AG: $7580 + 100 = 7680$ (Sampai pada Tujuan)

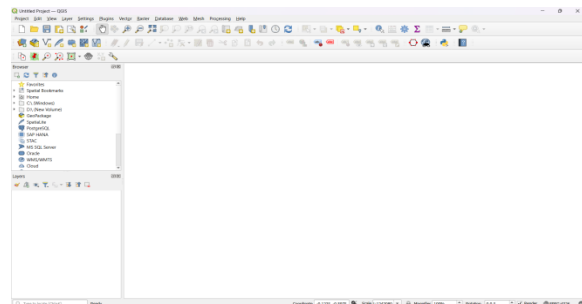
Jarak terupdate: A: 0, B: 8, C: 22, D: 700, E: 220, F: 400, G: 1322, H: 1200, R: 1240, S: 2640, T: 2940, U: 3940, V: 4240, W: 4540, X: 5240, Y: 5440, Z: 6040, AA: 6080, AB: 6780, AC: 7280, AD: 7480, AE: 7580, AF: 8680, AG: 7680 (Gerbang Masuk RS. Royal Prima)

Sehingga Hasil Akhir urutan jalurnya adalah A $\rightarrow$ B $\rightarrow$ E $\rightarrow$ F $\rightarrow$ H $\rightarrow$ R $\rightarrow$ S $\rightarrow$ T $\rightarrow$ U $\rightarrow$ V $\rightarrow$ W $\rightarrow$ Y $\rightarrow$ Z $\rightarrow$ AA $\rightarrow$ AB $\rightarrow$ AC $\rightarrow$ AD $\rightarrow$ AE $\rightarrow$ AG. Dengan Total Jarak: 7680 meter

4.2. Membuka Tampilan Awal QGIS

Setelah menemukan Rute terpendek dari Proses Algoritma Dijkstra tadi, langkah selanjutnya adalah memvisualisasikan rute terpendek tadi menggunakan QGIS. Langkah pertama adalah membuka aplikasi QGIS dan menyiapkan area kerja

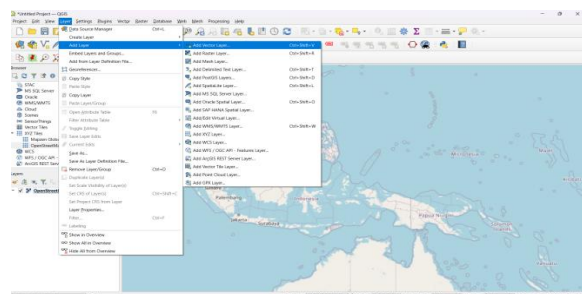
GIS baru. Pada tahap ini, pengguna akan melihat tampilan dasar QGIS yang biasanya kosong atau hanya berisi grid koordinat. Di sini, peta dasar disiapkan untuk memvisualisasikan data geografis.



Gambar 3. Tampilan Awal QGIS

4.3. Mengimpor Layer Menggunakan Tool

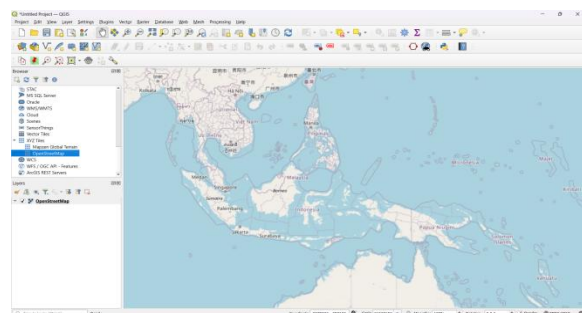
Setelah tampilan awal terbuka, langkah berikutnya adalah mengimpor layer data ke dalam QGIS. Pada Proses ini akan memasukkan jenis file data yang telah kita buat menggunakan python (GeoJSON), yang menjadi komponen utama dalam peta.



Gambar 4. Mengimpor File GeoJSON pada QGIS

4.4. Menampilkan OpenStreetMap di QGIS

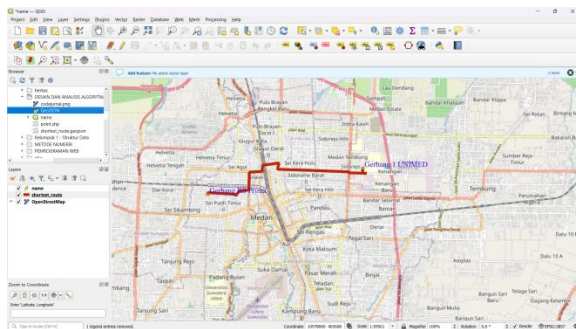
OpenStreetMap (OSM) digunakan sebagai sumber data peta jalan, QGIS menyediakan cara untuk mengakses data OSM secara langsung. Dengan menggunakan plugin seperti "QuickMapServices," data OSM dapat diunduh dan ditampilkan di QGIS. Setelah layer OSM dimuat, jaringan jalan yang digunakan untuk analisis perhitungan jalur terpendek, serta berbagai elemen geografis lainnya yang relevan, dapat dilihat disini.



Gambar 5. Menampilkan Peta Jalan Menggunakan Open Street Map

4.5. Menampilkan Rute Terpendek pada Peta QGIS

Kemudian Setelah menggunakan OpenStreetMap (OSM) untuk sumber data jaringan jalan, langkah selanjutnya adalah visualisasi hasil rute terpendek di QGIS. Proses ini dilakukan dengan mengimpor hasil perhitungan dalam bentuk file (.geoJSON) ke dalam QGIS. Setelah file dimuat, QGIS akan menggambarkannya rute pada peta, dan sehingga terlihat visualisasi rute terpendek pada Peta. Lalu untuk Membuat tampilan lebih terlihat informatif adalah dengan menambahkan Label Unimed Sebagai Titik Awal Pada Rute Keberangkatan dan Rumah Sakit Royal Prima sebagai Titik Akhir Tujuan.



Gambar 6. Visualisasi Rute Terpendek antara Universitas Negeri Medan dengan Rumah Sakit Royal Prima

Hasil dari visualisasi ini memberikan representasi yang jelas tentang jalur terpendek yang dapat dilalui dengan kualitas yang dapat dipertanggungjawabkan berkat penggunaan data spasial Dari OpenStreetMap (OSM), yang terkenal akurat untuk analisis jaringan jalan. Selain itu perbandingan dengan data jalur alternatif lapangan menunjukkan bahwa rute yang dihitung hampir sama dengan kondisi nyata yang ada. Kemudian penambahan label pada titik Unimed sebagai titik awal dan RS Royal Prima sebagai tujuan akhir semakin memudahkan pemahaman tentang jalur yang dihitung.

5. KESIMPULAN DAN SARAN

Penelitian ini menunjukkan bahwa algoritma Dijkstra efektif dalam menentukan rute terpendek dari Universitas Negeri Medan ke Rumah Sakit Royal Prima dengan tingkat akurasi yang baik yang dihitung menggunakan Google Maps sebagai referensi jarak. Algoritma ini berhasil menghasilkan jalur terpendek yang optimal, mempertimbangkan setiap titik secara global. Visualisasi rute di QGIS memperlihatkan peta yang jelas dan informatif, membantu dalam memahami jalur yang diambil. Meskipun penelitian ini menggunakan data rute yang sederhana, hasil yang diperoleh dapat diandalkan. Untuk penelitian selanjutnya, disarankan untuk memperluas model dengan mempertimbangkan faktor-faktor lain seperti

kondisi jalan, kepadatan lalu lintas, atau penggunaan data geografis yang lebih detail untuk meningkatkan akurasi dan relevansi hasil perhitungan. Selain itu, integrasi algoritma lain seperti A* dapat dipertimbangkan untuk meningkatkan performa dalam kasus yang lebih kompleks.

DAFTAR PUSTAKA

- [1] I. Ahadi, M. N. Habibah, P. P. D. Deria, and M. Fauzi, "Penerapan Algoritma Dijkstra untuk Mencari Rute Terpendek pada Pengiriman Produk Wafer di PT. XYZ," *Jurnal Manajemen Teknologi dan Teknik Industri Universitas Kadiri*, vol. 4, no. 1, pp. 1–13, Jan. 2022, doi: 10.30737/jurmativ.v4i1.1838.g1917.
- [2] N. Awallodien, W. Gata, and N. Qomariyah, "Algoritma Dijkstra dalam Penentuan Rute Terpendek pada Jalan Raya Antar Kota Jakarta - Tangerang," *Just IT: Jurnal Sistem Informasi, Teknologi Informasi dan Komputer*, vol. 13, no. 1, pp. 8–13, Sep. 2022. Available: <https://jurnal.umj.ac.id/index.php/just-it/index>.
- [3] M. Muharrom, "Implementasi Algoritma Dijkstra dalam Penentuan Jalur Terpendek Studi Kasus Jarak Tempat Kuliah Terdekat," *Indonesian Journal of Business Intelligence*, vol. 3, no. 1, Jun. 2020.
- [4] E. C. Rufus, R. R. Riyadi, and D. N. Hasibuan, "Penerapan Algoritma Dijkstra dalam Menentukan Rute Terpendek untuk Jasa Pengiriman Barang di Palangka Raya," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 3, Jun. 2024.
- [5] G. N. R. Ramadhan, R. K. A. Bachrun, and A. Syaifulloh, "Penerapan algoritma Dijkstra untuk menentukan rute terpendek tempat tinggal ke Kampus 2 UIN Sunan Ampel Surabaya," *Indonesian Journal of Business Intelligence*, vol. 7, no. 1, pp. 1-12, Jun. 2024
- [6] A. Y. FIRWANDA, C. PRIANTO, AND W. I. RAHAYU, "PENENTUAN RUTE TERPENDEK LOKASI BADAN PUSAT STATISTIK KOTA BANDUNG DENGAN ALGORITMA DIJKSTRA," *JURNAL TEKNIK INFORMATIKA*, vol. 9, no. 1, 2021. AVAILABLE: [HTTP://JURNAL.STMIK-DCL.AC.ID/INDEX.PHP/JUTEKIN](http://jurnal.stmik-dcl.ac.id/index.php/jutekin).
- [7] F. D. SIANIPAR, M. H. ARIFIN, W. AULIA, AND P. HARLIANA, "ESTIMASI RUTE TERDEKAT DARI UNIVERSITAS NEGERI MEDAN KE SPBU TERDEKAT MENGGUNAKAN ALGORITMA GREEDY," *JURNAL TEKNIK INFORMATIKA*, vol. 9, no. 1, pp. 76–85, 2021.
- [8] R. APRIADI, B. PRIHANDONO, AND Y. YUNDARI, "PENERAPAN ALGORITMA DIJKSTRA PADA PENDISTRIBUSIAN BAHAN BAKAR MINYAK DI PONTIANAK," *EDUMATSAINS*, vol. 7, no. 2, pp. 270–280, JAN. 2023. AVAILABLE: [HTTP://EJOURNAL.UKI.AC.ID/INDEX.PHP/EDUMATSAINS](http://ejournal.uki.ac.id/index.php/edumatsains).

- [9] M. FAUZI, A. ZAMSURI, AND Y. YUNEFRI, "SEBAGAI MEDIA PENCARIAN PONDOK PESANTREN DI KOTA PEKANBARU DENGAN METODE DIJKSTRA," *JURNAL TEKNIK INFORMATIKA*, VOL. 6, NO. 1, PP. 42–50, MAR. 2020.
- [10] S. Panggabean, W. Gata, A. R. Syarif, S. Rahmadani, and T. Widiyanto, "Implementasi Algoritma Dijkstra untuk Menentukan Jalur Terpendek Wilayah Pasar Minggu dan STMIK Nusamandiri Jakarta," *Jurnal Swabumi*, vol. 9, no. 1, pp. 78–85, Mar. 2021