

Sistem Informasi Navigasi Wisata Kota Jakarta untuk Menentukan Rute Tercepat Menggunakan Algoritma Dijkstra Berbasis Web

Muhammad Rizki Syaumi¹, Achmad Noe'man^{1,*}, Siti Setiawati¹, Prio Kustanto¹

* Korespondensi: e-mail: achmad.noeman@dsn.ubharajaya.ac.id

¹ Informatika; Universitas Bhayangkara Jakarta Raya; JL.Raya Perjuangan No. 81 Marga Mulya, Bekasi Utara Jawa Barat, 17142, (021) 88955882; e-mail: muhhammad.rizki.syaumi18@mhs.ubharajaya.ac.id, achmad.noeman@dsn.ubharajaya.ac.id, siti.setiawati@dsn.ubharajaya.ac.id, prio.kustanto@dsn.ubharajaya.ac.id

Submitted : 12 Maret 2025
Revised : 26 Maret 2025
Accepted : 19 April 2025
Published : 30 Mei 2025

Abstract

This study aims to design a web-based tourism navigation information system using Dijkstra's algorithm to determine the fastest route in Jakarta City. The proposed navigation system assists tourist in planning their trips more efficiently by providing real-time information on the fastest routes, travel distances, and estimated travel times. By implementing Dijkstra's algorithm, the system calculates the optimal route based on the starting location from the user's device and destination data stored in the database. This research employs the waterfall system development method, which includes the stages of analysis, design, implementation, and testing. The testing results demonstrate that the system accurately provides the fastest routes, enhancing convenience and travel efficiency for tourist.

Keywords: Dijkstra's Algorithm, Fastest Route, Jakarta City Tourism, Navigation System, Web-Based System

Abstrak

Penelitian ini bertujuan untuk merancang sistem informasi navigasi wisata berbasis web yang menggunakan algoritma Dijkstra guna menentukan rute tercepat di Kota Jakarta. Sistem navigasi yang diusulkan membantu wisatawan dalam merencanakan perjalanan secara lebih efisien dengan menyediakan informasi rute tercepat, jarak tempuh, dan estimasi waktu perjalanan secara real-time. Dengan mengimplementasikan algoritma Dijkstra, sistem ini menghitung rute optimal berdasarkan data lokasi awal dari perangkat pengguna dan lokasi tujuan yang tersimpan dalam basis data. Penelitian ini menggunakan metode pengembangan sistem waterfall yang melibatkan tahap analisis, perancangan, implementasi, dan pengujian. Hasil pengujian menunjukkan bahwa sistem mampu memberikan rute tercepat secara akurat, sehingga meningkatkan kenyamanan dan efisiensi perjalanan wisatawan.

Kata kunci: Algoritma Dijkstra, Rute Tercepat, Sistem Berbasis Web, Sistem Navigasi, Wisata Kota Jakarta

1. Pendahuluan

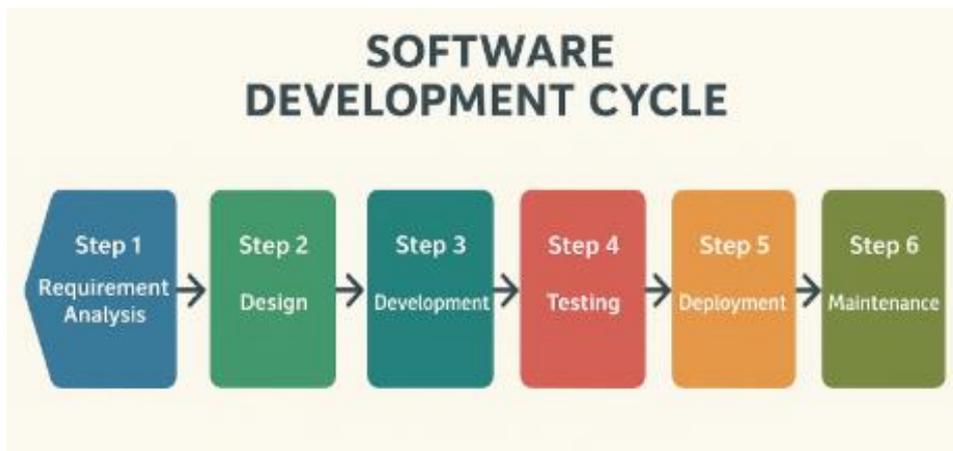
Jakarta merupakan Ibu Kota Negara Indonesia yang memiliki berbagai macam destinasi wisata mulai dari wisata sejarah hingga hiburan (Ramadhan & Anggraini, 2024). Namun, wisatawan sering menghadapi tantangan dalam menentukan rute perjalanan yang efisien, terutama di tengah kompleksitas jaringan jalan di kota ini (Sumantri & Hidayattullah, 2023).

Masalah ini sering menyebabkan pemborosan waktu dan biaya, sehingga menurunkan kenyamanan perjalanan. Untuk mengatasi permasalahan tersebut, penelitian ini mengusulkan pengembangan sistem navigasi wisata berbasis web menggunakan algoritma Dijkstra (Amalia & Kusdinar, 2024). Pendekatan ini dipilih karena kemampuannya dalam menemukan solusi optimal untuk menentukan rute tercepat. Algoritma Dijkstra adalah salah satu metode yang paling populer dalam menyelesaikan persoalan pencarian jalur terpendek (*Shortest Path Problem*) (Chatrin et al., 2022). Algoritma ini bekerja secara sistematis dengan mengevaluasi semua kemungkinan jalur dari satu titik ke titik lain pada graf, memberikan solusi yang akurat dan efisien (Chatrin et al., 2022).

Sistem ini dirancang untuk memberikan informasi rute tercepat, jarak tempuh dan estimasi waktu perjalanan secara real-time (Jeffrey & Marcel, 2024; Pradana et al., 2024). Dengan demikian, sistem ini diharapkan dapat membantu wisatawan merencanakan perjalanan dengan lebih baik dan mendukung pengembangan sektor pariwisata di Jakarta (Kekal et al., 2021).

2. Metode Penelitian

Penelitian menggunakan pendekatan *waterfall*, yang mencakup lima fase, di mana perkembangannya dimulai dari fase perencanaan dan berlanjut ke fase pemrosesan (pemeliharaan), yang dilaksanakan secara berurutan. Gambar 1 adalah lima tahap utama dari model *waterfall* (Prayudi, 2024).



Sumber: (Prayudi, 2024)

Gambar 1. Tahapan Penelitian

Pada Gambar 1 merupakan diagram alur horizontal dari *Software Development Cycle* (Siklus Pengembangan Perangkat Lunak) yang menggambarkan enam tahapan utama dalam proses pembuatan perangkat lunak. Diagram ini dimulai dari sebelah kiri dengan Step 1: *Requirement Analysis*, di mana kebutuhan dan tujuan sistem dikumpulkan dan dianalisis secara menyeluruh. Setelah itu, masuk ke Step 2: *Design*, yaitu tahap perancangan struktur sistem, baik dari sisi arsitektur perangkat lunak, antarmuka pengguna, hingga model data. Selanjutnya, Step 3: *Development* dilakukan, yakni tahap implementasi atau pengkodean sesuai dengan desain yang telah dibuat.

Setelah proses pengembangan selesai, langkah berikutnya adalah Step 4: *Testing*, yaitu pengujian perangkat lunak untuk memastikan bahwa sistem berjalan sesuai dengan kebutuhan dan bebas dari bug atau kesalahan. Bila lulus tahap pengujian, sistem akan masuk ke Step 5: *Deployment*, yaitu tahap peluncuran atau implementasi sistem ke lingkungan produksi sehingga bisa digunakan oleh pengguna. Terakhir, diagram menampilkan Step 6: *Maintenance*, yaitu tahap pemeliharaan dan perbaikan sistem secara berkelanjutan untuk memastikan kinerja tetap optimal dan sistem dapat menyesuaikan dengan kebutuhan pengguna di masa depan.

3. Hasil dan Pembahasan

3.1. Penerapan Algoritma Dijkstra

Algoritma Dijkstra digunakan untuk menentukan rute tercepat dari lokasi awal ke lokasi tujuan berdasarkan data lokasi pengguna dan destinasi wisata yang tersimpan dalam basis data. Algoritma ini menghitung jarak terpendek antara *node* (lokasi) dengan mempertimbangkan bobot (jarak) antar *node* (Pradana et al., 2024).

Tabel 1 adalah sampel data lokasi wisata yang digunakan untuk penerapan algoritma Dijkstra.

Tabel 1. Data Lokasi

No	Nama Lokasi Wisata	Latitude	Longitude
1	Taman Mini Indonesia Indah	-6.30188930	106.89036446
2	Monumen Nasional	-6.17526439	106.82765705
3	Kawasan Wisata Kota Tua Jakarta	-6.13488735	106.81428031
4	Taman Impian Jaya Ancol	-6.12480254	106.83436024
5	Pasar Pecinan Glodok	-6.14257735	106.81482807
6	Museum Nasional Indonesia	-6.16548230	106.82186040
7	Taman Margasatwa Ragunan	-6.30496351	106.82025692
8	Masjid Istiqlal	-6.16996733	106.83137927
9	KidZania Jakarta	-6.22487085	106.80997189
10	Stadion Utama Gelora Bung Karno	-6.21850267	106.80143122

Sumber: Hasil Penelitian (2025)

Latitude dan *longitude* adalah sistem koordinat geografis yang digunakan untuk menunjukkan posisi suatu lokasi di permukaan bumi. *Latitude* (lintang) mengukur jarak ke utara atau selatan dari garis khatulistiwa, sedangkan *longitude* (bujur) mengukur jarak ke timur atau barat dari garis meridian utama (*Greenwich*). Untuk mendapatkan nilai *latitude* dan *longitude* dari

suatu tempat, salah satu cara yang paling umum adalah dengan menggunakan *Google Maps*. Pengguna cukup mengetik nama lokasi pada kolom pencarian, kemudian mengklik kanan pada titik tersebut dan memilih opsi "*What's here?*" sehingga akan muncul koordinatnya secara langsung. Selain itu, koordinat juga dapat dilihat dari *URL Google Maps*, di mana nilai *latitude* dan *longitude* biasanya tercantum setelah simbol "@" dalam format *@latitude,longitude*. Selain metode manual, ada pula situs penyedia layanan koordinat seperti *latlong.net* atau *gps-coordinates.net* yang mempermudah pencarian hanya dengan memasukkan nama tempat. Untuk skala yang lebih besar dan otomatis, koordinat dapat diambil melalui pemrograman menggunakan *Google Maps Geocoding API* atau *Nominatim* dari *OpenStreetMap*, yang akan mengembalikan data *latitude* dan *longitude* dalam format JSON berdasarkan input teks nama lokasi.

Setelah mendapatkan data koordinat, algoritma Dijkstra dapat digunakan untuk menentukan jalur terpendek antar lokasi. Algoritma ini sangat efisien dalam memproses graf berbobot, di mana setiap lokasi wisata dianggap sebagai simpul (*node*) dan jarak antar lokasi sebagai bobot (*weight*) (Noviriandini & Safitri, 2017). Jarak antar lokasi dihitung terlebih dahulu menggunakan rumus Haversine, yang memperhitungkan kelengkungan bumi dalam mengestimasi jarak dari dua titik berdasarkan koordinat lintang dan bujunya. Setelah seluruh jarak dihitung dan dibentuk dalam struktur graf, algoritma Dijkstra akan mencari jarak terpendek dari satu titik asal ke semua titik lainnya dengan cara mengevaluasi jarak terpendek yang diketahui dan terus memperbarui jarak tersebut jika ditemukan jalur yang lebih pendek (Novandi, 2013). Proses ini dilakukan hingga semua simpul telah dievaluasi dan jalur optimal ditemukan. Dengan demikian, kombinasi data koordinat geografis dan algoritma Dijkstra menjadi pendekatan efektif untuk membangun sistem navigasi atau penjadwalan perjalanan wisata yang efisien dan berbasis data spasial. Langkah Langkah Penyelesaian Algoritma Dijkstra:

a. Inisialisasi *Node* dan Bobot

Data lokasi wisata diambil dari basis data, termasuk koordinat *latitude* dan *longitude*. Setiap lokasi dihubungkan dengan *node* lainnya menggunakan bobot berupa jarak antar *node*.

b. Pemilihan Lokasi Awal dan Tujuan

Lokasi Awal ditentukan berdasarkan data geolokasi perangkat pengguna. Lokasi tujuan dipilih dari daftar lokasi wisata yang tersedia dalam sistem.

c. Penerapan Algoritma Dijkstra

Dalam penerapan algoritma dilakukan langkah berikut: (1) *Node* awal diberi jarak 0, sedangkan *node* lainnya diberi jarak tak terhingga; (2) Periksa *node* dengan jarak terkecil yang belum dikunjungi; (3) Perbarui jarak ke *node* tetangga jika jalur melalui *node* saat ini lebih pendek; (4) Tandai *node* saat ini sebagai telah dikunjungi; (5) Ulangi langkah ini hingga semua *node* telah dikunjungi atau *node* tujuan ditemukan.

3.2. Hasil Pengujian

Sumber Data *Latitude* dan *Longitude* untuk setiap lokasi wisata diperoleh melalui *Google Maps*. Proses ini melibatkan pencarian lokasi berdasarkan nama tempat dan mengembalikan koordinat geografisnya.

Tabel 2. Hasil Pengujian

Lokasi Awal	Lokasi Tujuan	Jarak Tempuh	Waktu Tempuh	Rute Tercepat
Jalan Taman Merbabu 12, Puri Cendana, Sasak Tiga, Mangunjaya, Kab Bekasi, Jawa Barat, Jawa, 17519, Indonesia	Taman Mini Indonesia Indah	28.40 KM	28 menit	Jalan Pekayon Raya, Jalan Tol Lingkar Luar Jakarta
Jalan Taman Merbabu 12, Puri Cendana, Sasak Tiga, Mangunjaya, Kab Bekasi, Jawa Barat, Jawa, 17519, Indonesia	Monumen Nasional	38.08 KM	36 menit	Jalan Tol Jakarta–Cikampek, Jalan Tol Jakarta–Cikampek
Jalan Taman Merbabu 12, Puri Cendana, Sasak Tiga, Mangunjaya, Kab Bekasi, Jawa Barat, Jawa, 17519, Indonesia	Kawasan Wisata Kota Tua Jakarta	41.51 KM	39 menit	Jalan Sultan Agung, Jalan Bekasi Raya
Jalan Taman Merbabu 12, Puri Cendana, Sasak Tiga, Mangunjaya, Kab Bekasi, Jawa Barat, Jawa, 17519, Indonesia	Taman Impian Jaya Ancol	42.02 KM	39 menit	Jalan R. E. Martadinata, Pantai Indah
Jalan Taman Merbabu 12, Puri Cendana, Sasak Tiga, Mangunjaya, Kab Bekasi, Jawa Barat, Jawa, 17519, Indonesia	Pasar Pecinan Glodok	39.80 KM	37 menit	Jalan Sultan Agung, Jalan Bekasi Raya

Lokasi Awal	Lokasi Tujuan	Jarak Tempuh	Waktu Tempuh	Rute Tercepat
Jalan Taman Merbabu 12, Puri Cendana, Sasak Tiga, Mangunjaya, Kab Bekasi, Jawa Barat, Jawa, 17519, Indonesia	Museum Nasional Indonesia	38.41 KM	36 menit	Jalan Sultan Agung, Jalan Bekasi Raya
Jalan Taman Merbabu 12, Puri Cendana, Sasak Tiga, Mangunjaya, Kab Bekasi, Jawa Barat, Jawa, 17519, Indonesia	Taman Margasatwa Ragunan	35.94 KM	34 menit	Jalan Pekayon Raya, Jalan Tol Lingkar Luar Jakarta
Jalan Taman Merbabu 12, Puri Cendana, Sasak Tiga, Mangunjaya, Kab Bekasi, Jawa Barat, Jawa, 17519, Indonesia	Masjid Istiqlal	35.70 KM	34 menit	Jalan Bekasi Raya, Jalan Letnan Jenderal Suprpto
Jalan Taman Merbabu 12, Puri Cendana, Sasak Tiga, Mangunjaya, Kab Bekasi, Jawa Barat, Jawa, 17519, Indonesia	KidZania Jakarta	34.96 KM	30 menit	Jalan Tol Jakarta–Cikampek, Jalan Tol Cawang–Pluit
Jalan Taman Merbabu 12, Puri Cendana, Sasak Tiga, Mangunjaya, Kab Bekasi, Jawa Barat, Jawa, 17519, Indonesia	Stadion Utama Gelora Bung Karno	37.06 KM	33 menit	Jalan Tol Jakarta–Cikampek, Jalan Tol Cawang–Pluit

Sumber: Hasil Penelitian (2025)

Perhitungan Algoritma Dijkstra dengan langkah langkah penerapan dalam sistem adalah:

a. Representasi Graf

Lokasi wisata diwakili sebagai *node* dalam graf. Bobot antar *node* dihitung berdasarkan jarak antara dua lokasi menggunakan formula *Haversine*, yaitu pada formula 1.

Sistem Informasi Navigasi Wisata Kota Jakarta untuk Menentukan Rute Tercepat Menggunakan Algoritma Dijkstra Berbasis Web

$$d = 2r \cdot \arcsin(\sqrt{\sin^2(2\Delta\phi) + \cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2(2\Delta\lambda)}) \quad (1)$$

Di mana d adalah jarak antara dua lokasi (*node*); r adalah jari-jari bumi (6371 km); $\Delta\phi$ adalah selisih *latitude* dalam radian; $\Delta\lambda$ adalah selisih *longitude* dalam radian; ϕ_1, ϕ_2 adalah *latitude* lokasi pertama dan kedua dalam radian.

b. Inisialiasasi *Node* dan Bobot

Node awal diberi nilai jarak 0, sementara *node* lainnya diinisialisasi dengan nilai jarak tak terhingga. Simpan semua *node* dalam struktur data seperti *priority queue* untuk memprioritaskan *node* dengan jarak terkecil.

c. Iterasi Perhitungan

Pilih *node* dengan jarak terkecil yang belum dikunjungi. Perbarui jarak ke semua *node* tetangga jika jarak melalui *node* saat ini lebih kecil daripada jarak yang sudah tercatat. Tandai *node* saat ini sebagai telah dikunjungi dan keluarkan dari *queue*.

d. Pencarian Jalur Terpendek

Proses berlanjut hingga *node* tujuan ditemukan atau semua *node* telah diperiksa. Rekonstruksi jalur terpendek dengan melacak kembali *node* dari tujuan ke asal menggunakan tabel predecessor.

e. Hasil Akhir

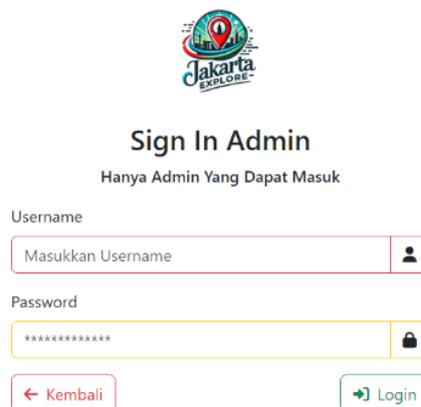
Sistem menampilkan rute terpendek beserta jarak total dan estimasi waktu perjalanan berdasarkan kecepatan rata-rata kendaraan.

3.4. Desain Sistem

Web ini menawarkan antarmuka yang ramah pengguna yang mencakup fungsi *Login*, data lokasi, dan data pengguna.

a. Halaman *Login* website

Halaman ini menampilkan sebuah form input username dan password untuk dapat masuk ke dalam dashboard admin. Berikut ini adalah tampilan halaman Login.

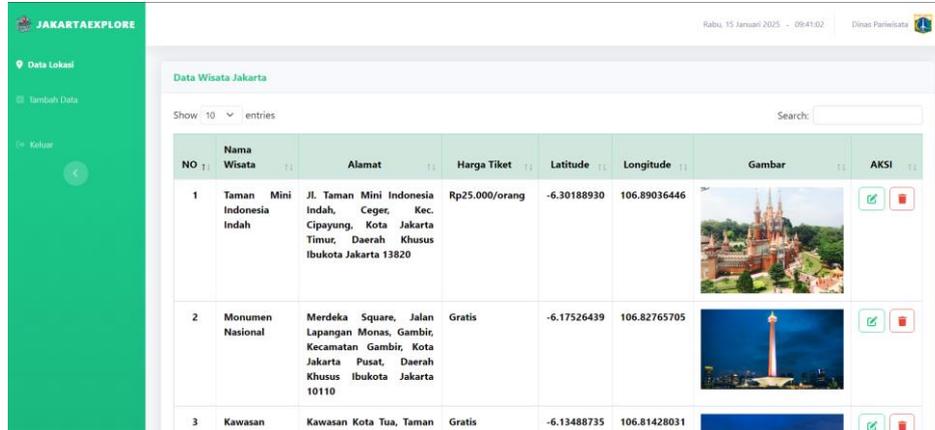


Sumber: Hasil Penelitian (2025)

Gambar 2. Halaman Login

b. Halaman Data Lokasi

Setelah admin melakukan login, sistem akan mengalihkan ke halaman data lokasi, data lokasi ini akan menampilkan tabel data lokasi wisata yang sudah dimasukkan ke dalam database, Gambar 3 adalah tampilan halaman data lokasi.



Sumber: Hasil Penelitian (2025)

Gambar 3. Halaman Data Lokasi

c. Halaman Data Pengguna

Halaman ini menampilkan data pengguna yang menampilkan data data terkait akun yang sudah dibuat sebelumnya seperti super admin dan admin.



Sumber: Hasil Penelitian (2025)

Gambar 4. Halaman Data Pengguna

3.5. Pengujian

Hasil evaluasi *black-box* menunjukkan bahwa sistem berfungsi sesuai dengan kebutuhan pengguna. Pengujian dilakukan dengan metode *black-box* untuk memvalidasi fungsionalitas utama.

Tabel 3. Pengujian *Black-Box*

No	DeskripsiPengujian	Skenario Pengujian	Hasil Yang Diharapkan	Hasil Pengujian
1	Login admin	Memasukan username dan password lalu klik login	Masuk ke halaman data lokasi	Berhasil
2	Logout admin	Klik tombol logout pada sidebar	Kembali ke halaman login	Berhasil

Sistem Informasi Navigasi Wisata Kota Jakarta untuk Menentukan Rute Tercepat Menggunakan Algoritma Dijkstra Berbasis Web

No	Deskripsi Pengujian	Skenario Pengujian	Hasil Yang Diharapkan	Hasil Pengujian
3	Menu Data Lokasi	Pilih menu data lokasi	Masuk ke halaman data lokasi	Berhasil
4	Menu Tambah Data	Pilih menu tambah data	Masuk ke halaman tambah data	Berhasil
5	Menghapus data pengguna	Klik tombol hapus	Data akan terhapus dari tampilan tabel data Pengguna	Berhasil
6	Merubah data pengguna dengan mengisi semua data	Klik tombol edit	Akan muncul notifikasi "Data berhasil diubah" dan dialihkan ke halaman data pengguna	Berhasil

Sumber: Hasil Penelitian (2025)

4. Kesimpulan

Hasil penelitian menunjukkan bahwa sistem informasi navigasi wisata yang dirancang sesuai dengan tujuan. Sistem ini berhasil menentukan rute tercepat menggunakan algoritma Dijkstra dengan menampilkan informasi rute, jarak tempuh, dan estimasi waktu secara *real-time* melalui peta interaktif berbasis *Leaflet JS* dan *API OSRM*. Dengan demikian, sistem ini mampu memberikan kemudahan bagi wisatawan dalam merencanakan perjalanan wisata di Jakarta secara efisien. Prospek pengembangan lebih lanjut meliputi integrasi fitur estimasi biaya perjalanan berdasarkan jenis kendaraan dan kondisi lalu lintas terkini, serta pengembangan aplikasi berbasis *mobile* untuk meningkatkan aksesibilitas pengguna. Selain itu, penerapan penelitian ini dapat diperluas ke kota-kota lain di Indonesia, sehingga mendukung pengembangan sektor pariwisata nasional secara lebih luas.

Daftar Pustaka

- Amalia, F. D., & Kusdinar, A. B. (2024.). *Penerapan Algoritma Dijkstra untuk Menemukan Tempat Ibadah Terdekat di Kota Sukabumi Menggunakan Metode Agile*.
- Chatrin, B., Hanna, P., & Yosefina, R. (2022). Penerapan Algoritma Dijkstra Untuk Menentukan Rute Terpendek Dari Pusat Kota Surabaya Ke Tempat Bersejarah. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 4(1), 213–223.
- Jeffry, J., & Marcel, M. (2024). Rancang Bangun Aplikasi Tes Minat dan Bakat Berbasis Web dengan Pendekatan Scrum. *Jurnal Teknik Informatika Dan Sistem Informasi*, 10(2), 317–333. <https://doi.org/10.28932/jutisi.v10i2.8896>
- Kekal, H. P., Gata, W., Nurdiani, S., Rini, A. J. S., & Wita, D. S. (2021). Analisa Pencarian Rute Tercepat Menuju Tempat Wisata Pulau Kumala Kota Tenggara Menggunakan Algoritma Greedy. *Jurnal Ilmiah Ilmu Komputer*, 7(1), 9–15. <https://doi.org/10.35329/jiik.v7i1.179>
- Novandi, R. A. D. (2013). Perbandingan Algoritma Dijkstra dan Algoritma Floyd-Warshall dalam

- Penentuan Lintasan Terpendek (Single Pair Shortest Path). *IF2251 Strategi Algoritmik*, 1, 1–5.
- Noviriandini, A., & Safitri, M. (2017). Implementasi Algoritma Dijkstra Untuk Menentukan Jalur Terpendek Wilayah Pisangan Dan Kampus Nusa Mandiri Tangerang. *Jurnal Pilar Nusa Mandiri*, 13(2), 181–186.
- Pradana, Y. A., Setyawati, Y., Dewi, L. P., Shobri, M. Q., Adhantoro, M. S., Kurniaji, G. T. B., & Romadloni, N. T. (2024). Penentuan Rute Optimal Wisata di Kota dan Kabupaten Madiun Menggunakan Algoritma Genetika. *Jurnal Keilmuan Dan Keislaman*, 49–56. <https://doi.org/10.23917/jkk.v3i1.223>
- Prayudi, A. (2024). Pengembangan Aplikasi E-Tanao Berbasis Website. *Jurnal Rekayasa Sistem Informasi Dan Teknologi*, 1(3), 70–75. <https://doi.org/10.59407/jrsit.v1i3.443>
- Ramadhan, I., & Anggraini, I. (2024). Pengaruh Kualitas Pelayanan Pemandu Wisata Terhadap Kepuasan Peserta Kegiatan Pengenalan Destinasi Wisata Disparekraf Dki Jakarta. *Media Bina Ilmiah*, 18(6), 1555–1564. <https://doi.org/10.33758/mbi.v18i6.716>
- Sumantri, E., & Hidayattullah, S. (2023). Penerapan Algoritma A*Star Untuk Mencari Rute Terpendek Dari Kemayoran Ke Destinasi Monumen Nasional (MONAS). *Jurnal Sains Dan Teknologi*, 5(2), 673–680. <https://doi.org/10.55338/saintek.v5i2.1432>