



Penentuan Rute Transportasi Terpendek Pengantaran Barang Jaminan Menggunakan Algoritma *Saving Matrix*

Rizka Utami Azhari Rambe^{1*}, Martiano²

^{1,2}Program Studi Sistem Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara, Medan, Indonesia

Email: ^{1*}rizkautami1216@gmail.com, ²martiano@umsu.ac.id

Abstrak

Proses pengantaran barang jaminan antar outlet pada PT. Indonesia Gadai OKE selama ini dilakukan secara manual tanpa mempertimbangkan efisiensi jarak tempuh, sehingga mengakibatkan rute yang tidak optimal, pemborosan bahan bakar, dan meningkatnya biaya operasional. Penelitian ini bertujuan merancang dan membangun sistem berbasis web yang menerapkan algoritma *Saving Matrix* untuk menentukan rute transportasi terpendek pada proses pengantaran barang jaminan. Metode yang digunakan adalah *engineering research* melalui tahapan pengumpulan data primer berupa koordinat geografis outlet, perancangan sistem menggunakan UML, implementasi algoritma, serta pengujian sistem dengan *black box testing*. Perhitungan jarak antar titik dilakukan menggunakan formula *Haversine*, kemudian nilai *saving* dihitung untuk setiap pasangan outlet dan digabungkan menjadi rute optimal berdasarkan nilai penghematan tertinggi. Hasil penelitian menunjukkan bahwa sistem berhasil dibangun dengan seluruh 19 skenario pengujian dinyatakan valid. Penerapan algoritma *Saving Matrix* menghasilkan rute optimal dengan total jarak 37,69 km, lebih efisien 29,04% dibandingkan rute manual sebelumnya sebesar 53,12 km. Penghematan tersebut setara dengan pengurangan konsumsi bahan bakar sebesar 1,54 liter dan penghematan biaya operasional sebesar Rp15.424 per sesi pengantaran. Dengan demikian, sistem yang dikembangkan terbukti mampu meningkatkan efisiensi distribusi barang jaminan secara signifikan.

Kata Kunci: *Saving Matrix*; Rute Terpendek; Distribusi; Barang Jaminan; *Vehicle Routing Problem*.

PENDAHULUAN

Kegiatan distribusi dan pengantaran barang merupakan komponen kritis dalam operasional perusahaan yang memiliki jaringan outlet tersebar di berbagai wilayah. Efisiensi distribusi secara langsung memengaruhi biaya operasional, kecepatan pelayanan, dan kepuasan pelanggan. Perencanaan rute yang tidak optimal menyebabkan jarak tempuh lebih panjang, pemborosan bahan bakar, serta keterlambatan pengiriman yang berujung pada penurunan kualitas layanan (Dantzig & Ramser, 1959).

Permasalahan optimasi rute kendaraan dalam distribusi barang dikenal dalam literatur ilmiah sebagai *Vehicle Routing Problem* (VRP). VRP pertama kali diperkenalkan oleh Dantzig dan Ramser (1959) sebagai upaya menentukan rute kendaraan yang berangkat dari satu depot menuju sejumlah titik tujuan dengan meminimalkan total jarak atau biaya (Toth & Vigo, 2014). Permasalahan ini tergolong NP-Hard sehingga penyelesaiannya secara eksak menjadi tidak praktis untuk skala besar, mendorong penggunaan algoritma heuristik sebagai pendekatan yang lebih efisien (Golden et al., 2008).

Salah satu algoritma heuristik yang paling banyak diterapkan untuk menyelesaikan VRP adalah algoritma *Saving Matrix* yang dikembangkan oleh Clarke dan Wright (1964). Algoritma ini bekerja dengan menggabungkan dua rute terpisah menjadi satu rute berdasarkan nilai penghematan (*saving value*) terbesar, sehingga total jarak tempuh dapat diminimalkan (Clarke & Wright, 1964). Kelebihan utama algoritma ini terletak pada kesederhanaan implementasi dan waktu komputasi yang cepat, menjadikannya sangat sesuai untuk perusahaan berskala kecil hingga menengah (Laporte, 1992).

Berbagai penelitian telah membuktikan efektivitas algoritma *Saving Matrix* dalam berbagai konteks distribusi. Setiawan dkk. (2022) menerapkan algoritma ini pada distribusi produk dan memperoleh penghematan jarak hingga 24,3%. Susanto dan Rizal (2023) berhasil mereduksi jumlah rute distribusi dari 8 menjadi 4 rute dengan penghematan biaya sebesar 31,7%. Lebih lanjut, Wibowo dkk. (2024) mengintegrasikan *Saving Matrix* dengan sistem informasi geografis (GIS) dan melaporkan peningkatan efisiensi operasional sebesar 35,2% pada perusahaan distribusi skala menengah. Kajian-kajian tersebut menunjukkan konsistensi performa *Saving Matrix* dalam berbagai karakteristik masalah distribusi.

PT. Indonesia Gadai OKE merupakan perusahaan gadai swasta berizin OJK yang memiliki 12 outlet tersebar di Kota Medan. Kegiatan operasional perusahaan mencakup pengantaran barang jaminan dari kantor pusat (depot) ke setiap outlet secara rutin. Kondisi saat ini menunjukkan bahwa rute pengantaran ditentukan secara manual oleh petugas lapangan berdasarkan kebiasaan, tanpa perhitungan optimasi jarak. Akibatnya, kendaraan operasional hanya mampu melayani 1 hingga 3 outlet per hari, padahal secara optimal dapat menjangkau hingga 6 outlet dalam satu sesi. Ketidakefisienan ini berimplikasi pada peningkatan konsumsi bahan bakar, biaya operasional, serta risiko keterlambatan pelayanan kepada nasabah.

Penelitian ini bertujuan untuk: (1) menganalisis kondisi rute pengantaran yang saat ini digunakan; (2) merancang dan membangun sistem berbasis web dengan menerapkan algoritma *Saving Matrix* untuk menentukan rute transportasi terpendek; serta (3) mengevaluasi tingkat efisiensi yang diperoleh dibandingkan dengan rute manual. Kebaruan penelitian ini terletak pada penerapan algoritma *Saving Matrix* berbasis koordinat GPS dengan formula *Haversine* pada sistem web yang juga dilengkapi fitur pemantauan petugas lapangan secara *real-time* menggunakan GIS, sebuah pendekatan yang belum dieksplorasi pada kasus perusahaan gadai multi-outlet di Indonesia.

METODE

Penelitian ini menggunakan pendekatan *engineering research* dengan metode pengembangan sistem secara terstruktur. Tahapan penelitian meliputi: (1) pengumpulan data, (2) analisis dan perancangan sistem, (3) implementasi algoritma dan sistem, serta (4) pengujian sistem.

Lokasi dan Objek Penelitian

Penelitian dilaksanakan pada PT. Indonesia Gadai OKE, yang berlokasi di Kota Medan, Sumatera Utara, dengan kantor pusat di Jl. Kapten Muslim, Komplek Megacom No. C18–C19 dan D1–D2, Medan Helvetia, sebagai depot (titik awal dan akhir pengantaran). Objek penelitian mencakup 11 outlet aktif perusahaan yang tersebar di berbagai kecamatan di Kota Medan.

Pengumpulan Data

Data primer diperoleh melalui observasi lapangan dan wawancara dengan pihak manajemen PT. Indonesia Gadai OKE, meliputi koordinat geografis (*latitude* dan *longitude*) kantor pusat dan seluruh outlet, data rute pengantaran manual yang selama ini digunakan, serta prosedur operasional pengantaran barang jaminan. Data sekunder diperoleh dari kajian literatur berupa jurnal ilmiah, buku referensi, dan dokumentasi teknis terkait algoritma *Saving Matrix* dan pengembangan sistem informasi.

Perancangan Sistem

Perancangan sistem dilakukan menggunakan *Unified Modeling Language* (UML), meliputi *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram* (Sommerville, 2016; Pressman & Maxim, 2020). Sistem dirancang sebagai aplikasi berbasis web menggunakan PHP 8.1 (*native*), MySQL, HTML5, CSS3 dengan *framework* Tailwind CSS, JavaScript ES6, serta pustaka visualisasi peta *Leaflet.js* dengan *basemap* *OpenStreetMap*. Geocoding dilakukan melalui *Nominatim API* dan visualisasi grafik menggunakan *Chart.js* versi 4.

Implementasi Algoritma Saving Matrix

Algoritma *Saving Matrix* diimplementasikan melalui tahapan sebagai berikut:

Langkah 1 – Perhitungan Matriks Jarak. Jarak antar setiap pasangan titik (depot dan outlet) dihitung menggunakan formula *Haversine* (Sinnott, 1984), yang menghitung jarak lingkaran besar (*great-circle distance*) pada permukaan bola bumi dengan jari-jari $R = 6.371$ km:

$$d = 2R \cdot \arcsin \left(\sqrt{\sin^2 \frac{\Delta\phi}{2} + \cos \phi_1 \cdot \cos \phi_2 \cdot \sin^2 \frac{\Delta\lambda}{2}} \right)$$

Langkah 2 – Perhitungan Nilai Saving. Nilai penghematan untuk setiap pasangan outlet i dan j dihitung dengan rumus Clarke dan Wright (1964):

$$S(i, j) = d(0, i) + d(0, j) - d(i, j) \dots (1)$$

di mana $d(0, i)$ adalah jarak dari depot ke outlet i , $d(0, j)$ adalah jarak dari depot ke outlet j , dan $d(i, j)$ adalah jarak langsung antara outlet i dan j .

Langkah 3 – Pengurutan dan Penggabungan Rute. Seluruh nilai *saving* diurutkan dari tertinggi ke terendah. Rute dibentuk dengan menggabungkan pasangan outlet yang memiliki nilai *saving* tertinggi, dengan syarat: (a) kedua outlet belum berada dalam satu rute yang sama, dan (b) setidaknya satu outlet berada di ujung rute yang sedang dibentuk. Proses berlanjut hingga seluruh outlet tergabung dalam satu atau lebih rute optimal.

Langkah 4 – Evaluasi Efisiensi. Tingkat efisiensi dihitung menggunakan rumus:

$$\text{Efisiensi (\%)} = ((\text{Jarak Manual} - \text{Jarak Optimal}) / \text{Jarak Manual}) \times 100\% \dots (2)$$

Pengujian Sistem

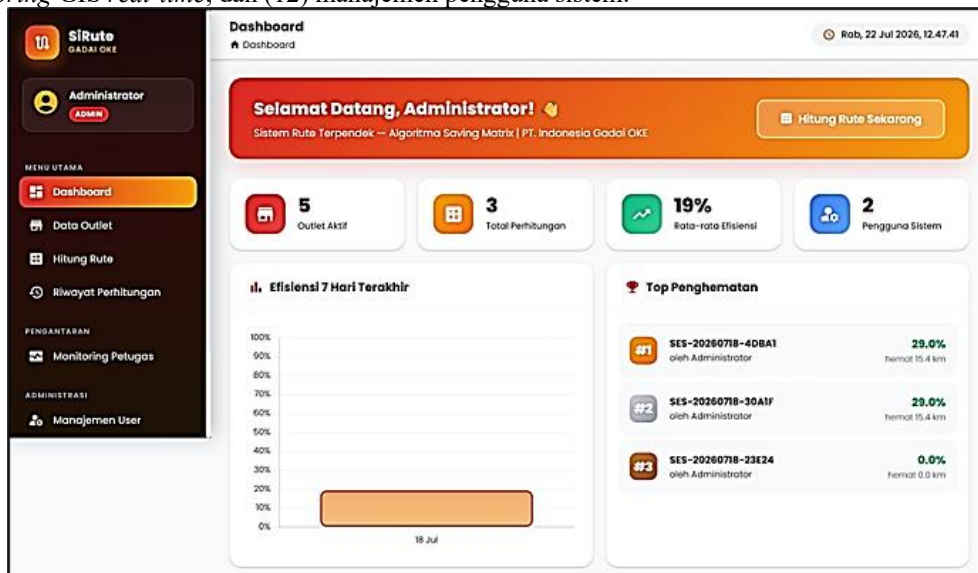
Pengujian dilakukan menggunakan metode *Black Box Testing*, yaitu pengujian berbasis fungsi yang memeriksa kesesuaian antara *input* yang diberikan dengan *output* yang dihasilkan sistem, tanpa memeriksa struktur kode internal (Black, 2009). Pengujian mencakup 19 skenario yang meliputi seluruh modul sistem.

HASIL DAN PEMBAHASAN

Implementasi Sistem

Sistem penentuan rute transportasi terpendek berhasil dibangun sebagai aplikasi berbasis web dengan lingkungan implementasi: PHP 8.1 (*native*), MySQL dengan ekstensi MySQLi, Tailwind CSS (CDN), JavaScript ES6, Leaflet.js (OpenStreetMap), Nominatim API untuk geocoding, Chart.js v4 untuk visualisasi grafik, dan SweetAlert2 untuk notifikasi antarmuka. Sistem diuji pada *web server* Apache (XAMPP) dengan perangkat berbasis Windows dan browser Google Chrome.

Sistem terdiri dari dua jenis pengguna dengan hak akses berbeda, yaitu Admin dan Petugas Lapangan, serta memiliki 12 modul fungsional utama: (1) autentikasi pengguna, (2) manajemen data outlet, (3) pengelolaan titik pusat/depot, (4) matriks jarak otomatis, (5) perhitungan rute *Saving Matrix*, (6) visualisasi hasil rute pada peta interaktif, (7) perbandingan rute manual dan optimal, (8) riwayat perhitungan dengan ekspor PDF, (9) manajemen tugas pengantaran, (10) pemantauan GPS petugas lapangan, (11) *monitoring GIS real-time*, dan (12) manajemen pengguna sistem.



Gambar 1. Tampilan Halaman Dashboard Sistem

Sirute
GADAI OKE

Administrator
ADMIN

MENU UTAMA

- Dashboard
- Data Outlet
- Hitung Rute
- Riwayat Perhitungan

PENGANTARAN

- Monitoring Petugas

ADMINISTRASI

- Manajemen User

Hitung Rute

Pilih outlet tujuan dan tentukan urutan manual (sebelum optimasi). Sistem akan menghitung rute optimal menggunakan Algoritma Saving Matrix (Clarke-Wright).

1 Pilih Outlet → 2 Urutan Manual → 3 Proses Optimasi → 4 Hasil

Pilih Outlet Tujuan

Kantor Pusat PT. Indonesia Gadai OKE (0)
Depot / Titik Keberangkatan – 3.60763900, 98.64680500

ID	Nama Outlet	Alamat	Koordinat
00-01	Gadai Oke 1 (Pasar 3)	Jl. Jamin Ginting No.884/632 Pasar 3, Padang Bulan, Kec...	
00-02	Gadai Oke 2	Jl. Marston Raya No. 8 No.895, Tanah Niram Ratus, Kec...	
00-03	GADAI OKE 3	Jl. Jamin Ginting No.899, Beringin, Kec. Medan Selayang...	
00-04	Gadai Oke 4	Jl. Arief Rahman Hakim No.276/276, Pk. Merah Tim, Kec...	
00-05	Gadai Oke 5	Jl. Matarhari Raya No.8, Helvetia Tengah, Kec. Medan Hel...	

0 outlet dipilih

Lanjut ke Urutan Manual

Gambar 2. Halaman Proses Perhitungan Rute

Data Uji Coba

Pengujian algoritma dilakukan menggunakan data primer koordinat GPS satu titik pusat (depot) dan lima outlet PT. Indonesia Gadai OKE yang telah dimasukkan ke dalam sistem. Pemilihan lima outlet representatif bertujuan memperlihatkan seluruh tahapan algoritma secara terperinci.

Tabel 1. Data Titik Pusat dan Outlet Uji Coba

No	Kode	Nama Titik	Latitude	Longitude
1	KP-001	Kantor Pusat (Gadai OKE 8)	3,601539	98,646805
2	GO-01	Gadai OKE 1 (Pasar 3)	3,551423	98,657206
3	GO-02	Gadai OKE 2 (Marelan)	3,692457	98,653385
4	GO-03	Gadai OKE 3 (Pasar 7)	3,542828	98,653763
5	GO-04	Gadai OKE 4 (AR Hakim)	3,570284	98,700481
6	GO-05	Gadai OKE 5 (Matahari Raya)	3,610112	98,641094

Sumber: Data Primer PT. Indonesia Gadai OKE, 2026

Perhitungan Matriks Jarak (Haversine)

Berdasarkan koordinat pada Tabel 1, sistem menghitung jarak antar seluruh titik secara otomatis menggunakan formula *Haversine*. Hasil matriks jarak disajikan pada Tabel 2.

Tabel 2. Matriks Jarak Antar Titik (km, Haversine)

Titik	Pusat	GO-01	GO-02	GO-03	GO-04	GO-05
Pusat	0,0000	5,6909	10,1360	6,5739	6,8965	1,1447
GO-01	5,6909	0,0000	15,6880	1,0293	5,2406	6,7664
GO-02	10,1360	15,6880	0,0000	16,6380	14,5557	9,2574
GO-03	6,5739	1,0293	16,6380	0,0000	6,0169	7,6126
GO-04	6,8965	5,2406	14,5557	6,0169	0,0000	7,9403
GO-05	1,1447	6,7664	9,2574	7,6126	7,9403	0,0000

Sumber: Hasil Perhitungan Sistem, 2026

Perhitungan Nilai Saving

Setelah matriks jarak diperoleh, nilai penghematan dihitung untuk setiap pasangan outlet menggunakan persamaan (1). Hasil seluruh 10 kombinasi pasangan outlet diurutkan dari nilai tertinggi ke terendah pada Tabel 3.

Tabel 3. Hasil Perhitungan Nilai Saving (Diurutkan Menurun)

No	Pasangan	Perhitungan	Nilai Saving
1	GO-01 – GO-03	$5,6909 + 6,5739 - 1,0293$	11,2355
2	GO-03 – GO-04	$6,5739 + 6,8965 - 6,0169$	7,4535
3	GO-01 – GO-04	$5,6909 + 6,8965 - 5,2406$	7,3468
4	GO-02 – GO-04	$10,1360 + 6,8965 - 14,5557$	2,4768
5	GO-02 – GO-05	$10,1360 + 1,1447 - 9,2574$	2,0233
6	GO-01 – GO-02	$5,6909 + 10,1360 - 15,6880$	0,1389
7	GO-03 – GO-05	$6,5739 + 1,1447 - 7,6126$	0,1060
8	GO-04 – GO-05	$6,8965 + 1,1447 - 7,9403$	0,1009
9	GO-02 – GO-03	$10,1360 + 6,5739 - 16,6380$	0,0718
10	GO-01 – GO-05	$5,6909 + 1,1447 - 6,7664$	0,0692

Sumber: Hasil Perhitungan Sistem, 2026

Pembentukan Rute Optimal

Berdasarkan nilai *saving* pada Tabel 3, rute dibentuk secara iteratif sebagaimana disajikan pada Tabel 4 di bawah ini.

Tabel 4. Proses Penggabungan Rute (Algoritma Saving Matrix)

Iterasi	Pasangan (Nilai Saving)	Status	Rute Sementara
1	GO-01 – GO-03 (11,2355)	Digabungkan	[GO-01, GO-03]
2	GO-03 – GO-04 (7,4535)	GO-03 di ujung, GO-04 digabungkan	[GO-01, GO-03, GO-04]
3	GO-01 – GO-04 (7,3468)	Sudah satu rute, dilewati	[GO-01, GO-03, GO-04]
4	GO-02 – GO-04 (2,4768)	GO-04 di ujung, GO-02 digabungkan	[GO-01, GO-03, GO-04, GO-02]
5	GO-02 – GO-05 (2,0233)	GO-02 di ujung, GO-05 digabungkan	[GO-01, GO-03, GO-04, GO-02, GO-05]
6–10	Pasangan sisa	Semua titik sudah dalam satu rute	[GO-01, GO-03, GO-04, GO-02, GO-05]

Sumber: Hasil Perhitungan Sistem, 2026

Hasil akhir adalah satu rute tunggal: **Pusat** → **GO-01** → **GO-03** → **GO-04** → **GO-02** → **GO-05** → **Pusat**, yang melayani seluruh outlet dalam satu perjalanan sesuai dengan batasan satu kendaraan tanpa *time window*.

Perbandingan Rute Manual dan Rute Optimal

Rute manual merepresentasikan urutan kunjungan outlet berdasarkan urutan pendaftaran yang selama ini digunakan petugas lapangan. Perbandingan rincian jarak kedua rute disajikan pada Tabel 5 dan Tabel 6.

Tabel 5. Rincian Rute Manual (Kondisi Saat Ini)

Urutan	Dari	Ke	Jarak (km)
1	Pusat	GO-01	5,6909
2	GO-01	GO-02	15,6880
3	GO-02	GO-03	16,6380
4	GO-03	GO-04	6,0169
5	GO-04	GO-05	7,9403
6	GO-05	Pusat	1,1447
Total			53,1189 km

Sumber: Data Primer Operasional PT. Indonesia Gadai OKE, 2026

Tabel 6. Rincian Rute Optimal (Hasil Algoritma Saving Matrix)

Urutan	Dari	Ke	Jarak (km)
1	Pusat	GO-01	5,6909
2	GO-01	GO-03	1,0293
3	GO-03	GO-04	6,0169
4	GO-04	GO-02	14,5557
5	GO-02	GO-05	9,2574
6	GO-05	Pusat	1,1447
Total			37,6948 km

Sumber: Hasil Perhitungan Sistem, 2026

Analisis Efisiensi Rute

Perbandingan komprehensif antara rute manual dan rute optimal disajikan pada Tabel 7. Estimasi konsumsi bahan bakar menggunakan asumsi 1 liter per 10 km dengan harga BBM Rp10.000/liter.

Tabel 7. Perbandingan Rute Manual dan Rute Optimal

Parameter	Rute Manual	Rute Optimal	Penghematan
Total Jarak Tempuh	53,1189 km	37,6948 km	15,4241 km
Konsumsi BBM (1 L/10 km)	5,3119 L	3,7695 L	1,5424 L
Estimasi Biaya BBM	Rp53.119	Rp37.695	Rp15.424
Efisiensi Jarak	—	—	29,04%

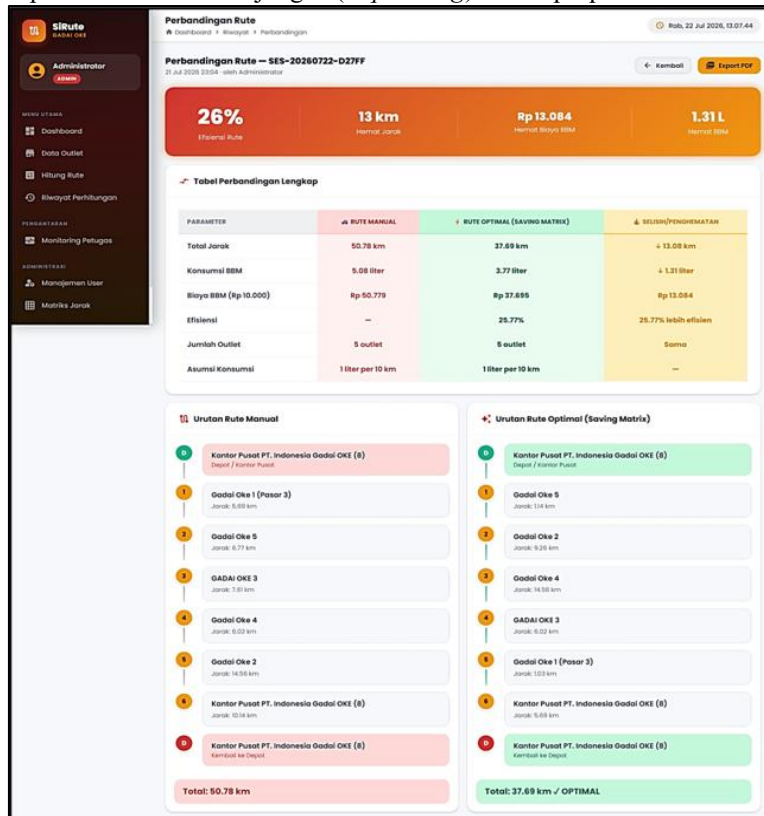
Sumber: Hasil Perhitungan Sistem, 2026

Menggunakan persamaan (2), tingkat efisiensi diperoleh:

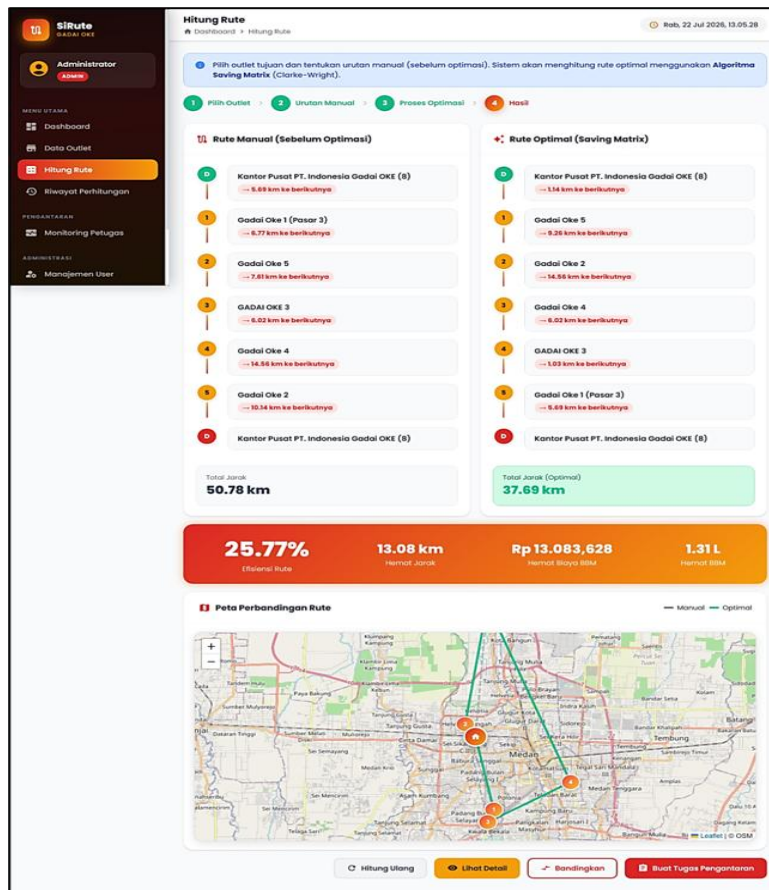
$$\text{Efisiensi} = ((53,1189 - 37,6948) / 53,1189) \times 100\% = 29,04\%$$

Hasil ini menunjukkan bahwa algoritma *Saving Matrix* mampu menghemat jarak tempuh sebesar 15,42 km (29,04%) untuk satu sesi pengantaran ke lima outlet. Perlu dicatat bahwa nilai jarak yang digunakan

merupakan jarak *great-circle* (Haversine), bukan jarak jalan riil, sehingga nilai efisiensi tersebut merepresentasikan perbaikan urutan kunjungan (*sequencing*) secara proporsional.



Gambar 3. Grafik Perbandingan Jarak, Konsumsi BBM, dan Biaya antara Route Manual dan Route Optimal



Gambar 4. Halaman Hasil Route Dengan Urutan Kunjungan Outlet dan Peta Interaktif Leaflet.js

Pengujian Sistem (Black Box Testing)

Pengujian dilakukan terhadap 19 skenario yang mencakup seluruh modul fungsional sistem. Hasil pengujian disajikan pada Tabel 8.

Tabel 8. Hasil Pengujian Sistem (Black Box Testing)

No	Modul/Fitur	Skenario Pengujian	Hasil Diharapkan	Kesimpulan
1	Login	Kredensial benar	Masuk ke dashboard sesuai role	Valid
2	Login	Kredensial salah	Pesan kesalahan, akses ditolak	Valid
3	Data Outlet	Tambah outlet via geocoding	Koordinat terisi otomatis, data tersimpan	Valid
4	Data Outlet	Tambah outlet via tautan Google Maps	Koordinat diurai dari tautan	Valid
5	Data Outlet	Ubah dan hapus data outlet	Data ter-update/terhapus	Valid
6	Titik Pusat	Tetapkan satu titik pusat/depot	Hanya satu titik pusat aktif diizinkan	Valid
7	Matriks Jarak	Tampilkan matriks jarak	Jarak dihitung otomatis (Haversine)	Valid
8	Hitung Rute	Pilih outlet, jalankan perhitungan	Proses bertahap, hasil rute optimal tampil	Valid
9	Hitung Rute	Jalankan tanpa memilih outlet	Peringatan outlet belum dipilih	Valid
10	Hasil Rute	Tampilkan pada peta interaktif	Rute manual & optimal warna berbeda	Valid
11	Perbandingan	Grafik perbandingan jarak, BBM, biaya	Grafik dan tabel tampil sesuai data	Valid
12	Riwayat	Filter berdasarkan tanggal/pencarian	Data riwayat tersaring sesuai kriteria	Valid
13	Riwayat	Ekspor laporan ke PDF	Berkas PDF dapat diunduh/dicetak	Valid
14	Tugas	Admin tugaskan rute ke petugas	Tugas muncul pada halaman petugas	Valid
15	Tugas	Petugas ubah status siap/proses/selesai	Status ter-update real-time	Valid
16	GPS Tracking	Rekam lokasi petugas tiap 15 detik	Titik lokasi ter-update pada peta	Valid
17	Monitoring GIS	Pantau posisi seluruh petugas aktif	Posisi tampil real-time pada peta	Valid
18	Manajemen User	Tambah, ubah, nonaktifkan, reset password	Perubahan tersimpan dan berlaku	Valid
19	Logout	Pengguna keluar dari sistem	Sesi berakhir, diarahkan ke login	Valid

Sumber: Hasil Pengujian, 2026

Seluruh 19 skenario pengujian menunjukkan hasil valid, yang berarti sistem berfungsi sesuai spesifikasi kebutuhan dan siap dioperasikan pada lingkungan produksi.

Pembahasan

Hasil penelitian ini membuktikan bahwa algoritma *Saving Matrix* efektif diterapkan pada kasus distribusi barang jaminan PT. Indonesia Gadaai OKE. Efisiensi sebesar 29,04% yang diperoleh sejalan dengan temuan-temuan sebelumnya dalam literatur. Setiawan dkk. (2022) melaporkan efisiensi 24,3% pada distribusi produk komersial; Susanto dan Rizal (2023) memperoleh reduksi biaya 31,7% pada distribusi makanan; sedangkan Wibowo dkk. (2024) mencapai 35,2% pada distribusi logistik dengan integrasi GIS. Perbedaan nilai efisiensi antara penelitian ini dan penelitian terdahulu wajar terjadi karena dipengaruhi oleh karakteristik sebaran geografis outlet, jumlah titik distribusi, dan pola jarak antar titik yang berbeda-beda pada setiap objek penelitian (Golden et al., 2008).

Keunggulan sistem yang dikembangkan dalam penelitian ini dibandingkan implementasi konvensional terletak pada dua aspek: pertama, penggunaan koordinat GPS *real-time* dengan formula *Haversine* memberikan estimasi jarak yang lebih akurat dibandingkan perhitungan manual berbasis peta statis; kedua, integrasi fitur pemantauan GIS memungkinkan supervisor memantau realisasi rute secara langsung, yang tidak ditemukan pada sebagian besar sistem serupa dalam literatur (Wibowo et al., 2024). Keterbatasan penelitian ini adalah penggunaan jarak *Haversine* (garis lurus) sebagai pendekatan, bukan jarak jalan riil, serta belum dipertimbangkannya faktor *time window* dan variasi kapasitas kendaraan, yang menjadi peluang pengembangan lebih lanjut.

KESIMPULAN

Penelitian ini berhasil merancang, mengimplementasikan, dan menguji sistem berbasis web untuk menentukan rute transportasi terpendek dalam pengantaran barang jaminan PT. Indonesia Gadaai OKE menggunakan algoritma *Saving Matrix*. Kesimpulan yang diperoleh adalah sebagai berikut: Pertama, sistem berbasis web telah berhasil dibangun dengan 12 modul fungsional, dan seluruh 19 skenario pengujian *black box* dinyatakan valid, membuktikan sistem berfungsi sesuai kebutuhan operasional. Kedua, algoritma *Saving Matrix* dengan perhitungan jarak *Haversine* berhasil membentuk rute optimal melalui tahapan yang terstruktur dan terkomputasi secara otomatis. Ketiga, penerapan algoritma *Saving Matrix* terbukti meningkatkan efisiensi rute pengantaran sebesar 29,04%, dengan penghematan jarak 15,42 km, penurunan konsumsi bahan bakar 1,54 liter, dan penghematan biaya Rp15.424 per sesi pengantaran dibandingkan rute manual.

Penelitian selanjutnya disarankan untuk mengintegrasikan data jarak jalan riil melalui layanan *routing engine* seperti OSRM (*Open Source Routing Machine*) atau Google Maps Distance Matrix API, mempertimbangkan variabel *time window* dan kapasitas kendaraan yang bervariasi, serta membandingkan performa algoritma *Saving Matrix* dengan metode metaheuristik seperti *Genetic Algorithm* atau *Ant Colony Optimization* untuk memperoleh solusi yang lebih optimal.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada PT. Indonesia Gadaai OKE atas izin dan kemudahan akses data selama penelitian berlangsung, serta kepada Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara atas dukungan akademik dalam penyelesaian penelitian ini.

DAFTAR PUSTAKA

- Black, R. (2009). *Managing the Testing Process: Practical Tools and Techniques for Managing Hardware and Software Testing* (3rd ed.). Wiley Publishing.
- Clarke, G., & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12(4), 568–581. <https://doi.org/10.1287/opre.12.4.568>
- Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, 6(1), 80–91. <https://doi.org/10.1287/mnsc.6.1.80>
- Golden, B. L., Raghavan, S., & Wasil, E. A. (2008). *The Vehicle Routing Problem: Latest Advances and New Challenges*. Springer. <https://doi.org/10.1007/978-0-387-77778-8>
- Laporte, G. (1992). The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(3), 345–358. [https://doi.org/10.1016/0377-2217\(92\)90192-C](https://doi.org/10.1016/0377-2217(92)90192-C)
- Leaflet. (2023). *Leaflet: An open-source JavaScript library for mobile-friendly interactive maps*. <https://leafletjs.com>
- OpenStreetMap Foundation. (2024). *OpenStreetMap*. <https://www.openstreetmap.org>
- Pressman, R. S., & Maxim, B. R. (2020). *Software Engineering: A Practitioner's Approach* (9th ed.). McGraw-Hill Education.

- Setiawan, A., Nugroho, B., & Hartati, S. (2022). Optimasi rute distribusi produk menggunakan algoritma saving matrix pada perusahaan distribusi di Jawa Tengah. *Jurnal Teknik Industri*, 24(1), 55–66. <https://doi.org/10.22219/JTIUMM.Vol24.No1.55-66>
- Sinnott, R. W. (1984). Virtues of the Haversine. *Sky and Telescope*, 68(2), 159.
- Sommerville, I. (2016). *Software Engineering* (10th ed.). Pearson Education.
- Susanto, H., & Rizal, M. (2023). Penerapan algoritma saving matrix untuk optimasi rute distribusi makanan pada perusahaan katering di Surabaya. *Jurnal Sistem dan Teknologi Informasi*, 11(3), 201–212. <https://doi.org/10.26418/justin.v11i3.56721>
- Toth, P., & Vigo, D. (2014). *Vehicle Routing: Problems, Methods, and Applications* (2nd ed.). Society for Industrial and Applied Mathematics (SIAM). <https://doi.org/10.1137/1.9781611973594>
- Wibowo, S., Kurniawan, R., & Santoso, D. (2024). Integrasi algoritma saving matrix dengan sistem informasi geografis untuk optimasi rute distribusi logistik. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 11(2), 315–326. <https://doi.org/10.25126/jtiik.2024112345>