

## IMPLEMENTASI ALGORITMA *FLOYD-WARSHALL* PADA SISTEM RUTE TERPENDEK : STUDI KASUS PERUSAHAAN LOGISTIK

Rifqi Fathurrohman<sup>1</sup>, Cahyo Prianto<sup>2</sup>.

Sekolah Vokasi, Sarjana Terapan Informatika, Universitas Logistik dan Bisnis Internasional<sup>1,2</sup>

email: <sup>1</sup>1204076@std.ulbi.ac.id, <sup>2</sup>cahyo@ulbi.ac.id

---

### Abstrak

Penelitian ini mengeksplorasi penerapan Algoritma *Floyd-Warshall* pada perusahaan logistik. Algoritma tersebut digunakan untuk menghitung jalur distribusi terpendek dalam mengatasi perubahan rute dan bobot dinamis. Sistem yang dihasilkan dapat meningkatkan keuntungan dengan memberikan solusi logistik yang efisien diintegrasikan ke dalam sistem menggunakan metode penelitian SDLC. Evaluasi dilakukan melalui uji perbandingan rute dengan *Google Maps*, menunjukkan hasil yang baik. Pada implementasi menunjukkan bahwa sistem saat ini hanya mempertimbangkan jarak tanpa memperhitungkan waktu tempuh dan kondisi lalu lintas *real-time*. Meskipun demikian, penelitian ini berhasil membuktikan potensi Algoritma *Floyd-Warshall* dalam meningkatkan distribusi logistik

**Kata Kunci:** Algoritma *Floyd-Warshall*, Sistem Logistik, Perusahaan Logistik, Rute Terpendek

### Abstract

*This research explores the application of the Floyd-Warshall Algorithm to a logistics company. The algorithm is used to calculate the shortest distribution path in overcoming route changes and dynamic weights. The resulting system can increase profits by providing efficient logistics solutions integrated into the system using the SDLC research method. Evaluation was done through a route comparison test with Google Maps, showing good results. The implementation showed that the current system only considers distance without taking into account travel time and real-time traffic conditions. Nonetheless, this research successfully proved the potential of Floyd-Warshall Algorithm in improving logistics distribution*

**Keywords:** *Floyd-Warshall Algorithm, Logistic System, Logistic Company, Shortest Route.*

---

### 1. PENDAHULUAN

Dalam era globalisasi ini, manajemen logistik memegang peran sentral dalam mendukung kinerja efisien dan efektif suatu organisasi atau perusahaan [1]. Keberhasilan pengiriman barang yang tepat waktu dan efisien sangat bergantung pada optimalisasi sistem rute terpendek [2]. Oleh karena itu, penelitian ini memfokuskan pada penerapan Algoritma *Floyd-Warshall*, dengan fokus khusus pada studi kasus Perusahaan logistik [3]. Perusahaan logistik, sebagai lembaga yang bergerak dalam bidang pengiriman dan distribusi, menghadapi tantangan signifikan dalam menjaga efisiensi dan ketepatan waktu dalam setiap pengiriman [4]. Oleh karena itu, pengembangan sistem informasi logistik dengan menerapkan Algoritma *Floyd-Warshall* menjadi alternatif

menarik untuk dijelajahi [5]. Algoritma *Floyd-Warshall* dikenal sebagai algoritma graf yang mampu menghitung jalur terpendek antara setiap pasang simpul pada graf berbobot [6]. Keunggulan algoritma ini terletak pada kemampuannya menangani graf dengan bobot dinamis dan perubahan topologi graf, hal ini menjadi krusial dalam menghadapi perubahan kondisi jalan atau lokasi distribusi barang [7]. Dalam konteks pengembangan sistem informasi logistik Perusahaan logistik, implementasi Algoritma *Floyd-Warshall* menjadi solusi yang menjanjikan untuk menentukan rute terpendek antar lokasi distribusi [8]. Kelebihan algoritma ini, seperti kemampuannya menangani bobot dinamis dan perubahan topologi graf, menjadikannya

pilihan yang sesuai dengan dinamika perubahan yang mungkin terjadi di lapangan [9].

Penelitian ini bertujuan untuk mengimplementasikan Algoritma *Floyd-Warshall* pada sistem logistik perusahaan logistik dengan meningkatkan efisiensi rute distribusi, termasuk perbaikan dalam hal jarak tempuh, yang akan memberikan dampak pada penghematan biaya, dan optimalisasi penggunaan sumber daya. Untuk mencapai tujuan penelitian ini, peneliti akan menyelaraskan Algoritma *Floyd-Warshall* ke dalam kerangka pengembangan *web* menggunakan Laravel [10]. Sebagai *framework* PHP yang sangat populer, Laravel dapat memberikan fondasi yang stabil dan dapat diperluas untuk implementasi algoritma [11]. Selain itu, penggunaan Laravel mempermudah proses penyelenggaraan dan pemeliharaan sistem [12].

Hasil dari penelitian ini melibatkan pengembangan sistem informasi logistik yang dapat secara otomatis menentukan rute terpendek dengan memanfaatkan Algoritma *Floyd-Warshall* [13]. Sistem ini diantisipasi memberikan solusi logistik yang efisien dan terukur, mengoptimalkan perjalanan barang, dan pada akhirnya, meningkatkan kinerja dan keuntungan Perusahaan logistik. Pendekatan terbaik untuk logistik yang akan diusulkan dalam penelitian ini mencakup pemanfaatan hasil analisis dari implementasi Algoritma *Floyd-Warshall*, untuk mendukung pengambilan keputusan logistik yang lebih cerdas di lingkungan yang unik seperti Perusahaan logistik [14].

## 2. LANDASAN TEORI

### 2.1. Algoritma *Floyd-Warshall*

Algoritma *Floyd-Warshall* adalah metode dalam teori graf yang digunakan untuk mengidentifikasi jalur terpendek antara setiap pasang simpul pada graf dengan bobot [15]. Ditemukan oleh Robert Floyd dan Stephen Warshall, algoritma ini efisien dalam menghitung jarak terpendek di antara semua pasangan simpul dalam graf, bahkan saat terjadi perubahan bobot dan struktur graf [16]. Algoritma ini terkenal karena kemampuannya menangani graf siklik dengan bobot yang dinamis, tidak terbatas pada graf acyclic [17]. Dalam representasi matriksnya, rumus Algoritma *Floyd-Warshall* dapat dijelaskan sebagai berikut.

$$W_{ij} = (W_{ij}, W_{ik} + W_{kj}) \text{ jika } k \geq 1$$

Dengan penjelasan sebagai berikut:

- $W_0$  = matriks keterhubungan pada graf terarah dengan bobot awal.
- $W^*$  = matriks keterhubungan minimal.
- $W_{ij}$  = adalah jalur terpendek dari titik  $V_i$  ke  $V_j$

Rumus ini mencerminkan proses iteratif algoritma *Floyd-Warshall*, di mana jarak terpendek antara setiap pasang simpul diperbarui berdasarkan simpul ketiga yang dianggap sebagai perantara [18]. Selama iterasi, matriks jarak terpendek lengkap terbentuk, menggambarkan solusi optimal untuk permasalahan jalur terpendek dalam graf.

## 3. METODE PENELITIAN

Metode penelitian yang digunakan pada penelitian ini adalah SDLC. Metodologi SDLC (*Software Development Life Cycle*) Pengembangan dimulai dengan fase persyaratan dan kemudian dilanjutkan dengan desain, pengembangan rencana pengujian, implementasi (coding), pengujian, dan penerapan [19]. Diterapkan dalam implementasi algoritma *Floyd-Warshall* untuk sistem rute terpendek dengan menggunakan Laravel, dengan fokus pada studi kasus Perusahaan logistik. Berikut adalah tahapan implementasi dalam metodologi SDLC.



**Gambar 1** Metode SDLC

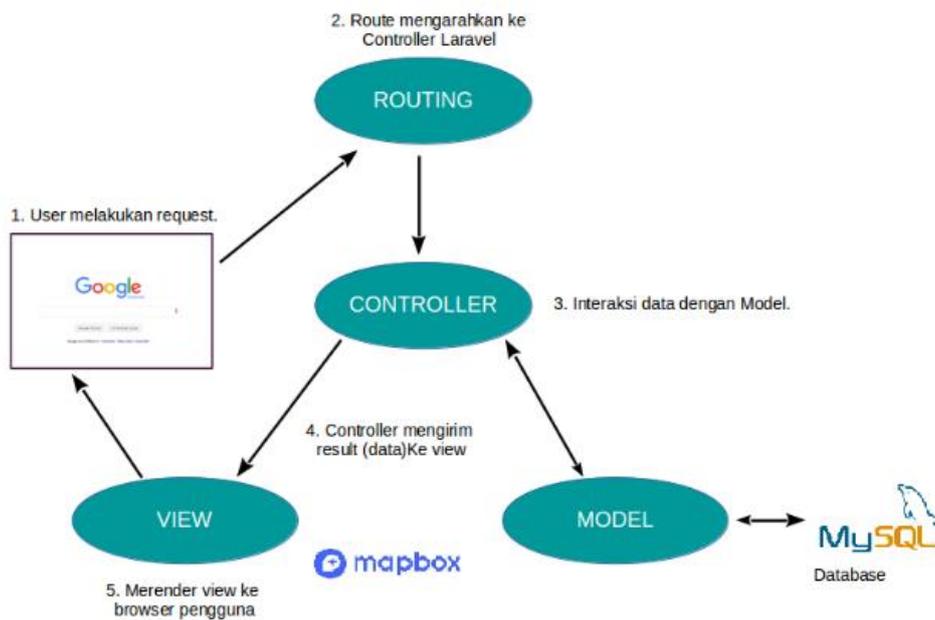
Dalam penerapan sistem rute terpendek dengan menggunakan Laravel, digunakan





Pada gambar 3 *flowchart*, aplikasi ini dimulai dari halaman utama yang menampilkan peta menggunakan *mapbox* serta titik yang sudah ditambahkan oleh *admin* sebelumnya dan menyediakan opsi untuk melihat daftar lokasi serta informasi. Terdapat tiga jenis pengguna, yakni *admin*, *supervisor*, dan kurir. *Admin* dapat *login* untuk mengakses *dashboard* yang menampilkan jumlah data secara keseluruhan, dan dapat menambahkan data pengguna, data lokasi, serta graf. *Supervisor* dapat melakukan proses pendaftaran dan *login*. Setelah *login*, *supervisor* dapat memantau kurir, melihat jumlah data, mengelola profil, menambahkan

data lokasi, dan graf. Kurir juga harus mendaftar terlebih dahulu, tetapi tidak dapat langsung mengakses *dashboard*. Setelah ketiga pengguna berhasil *login*, mereka dapat melakukan perhitungan rute dari set start/titik awal lokasi ke set *end*/titik akhir lokasi. Pengguna juga memiliki opsi untuk menambahkan rute baru dan melihat detail perhitungan algoritma *Warshall* di halaman utama. Setelah selesai, mereka dapat *logout*/keluar dari aplikasi. Untuk perancangan arsitektur sistem terlihat pada gambar 4.

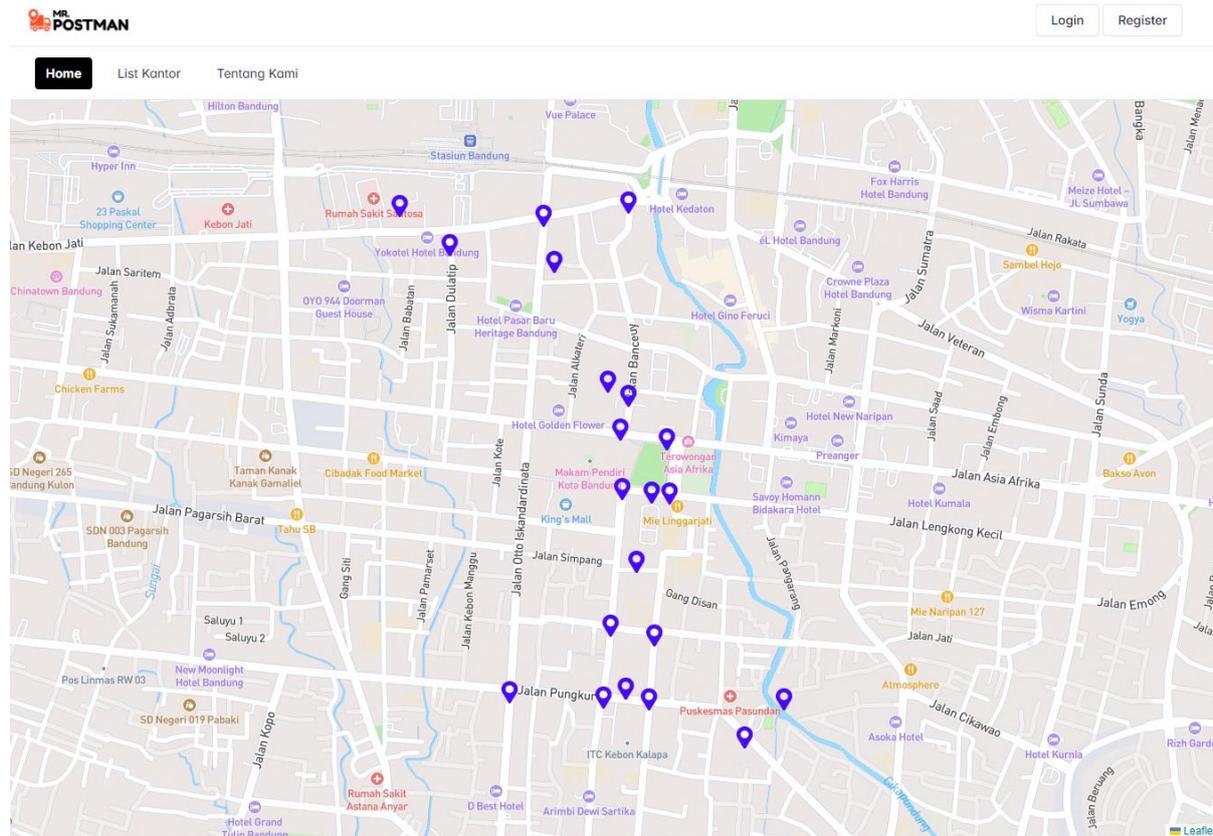


Gambar 4 Arsitektur Sistem

Dalam gambar 4 arsitektur sistem Laravel, saat pengguna berinteraksi dengan aplikasi, *browser* mengirimkan permintaan ke server, kemudian sistem routing Laravel mengarahkannya ke *class* dan *method* yang berisikan algoritma *Floyd warshall*. *Controller* bertindak sebagai perantara antara permintaan dan model algoritma, terutama untuk mengakses data dari database [21].

### 4.3. Desain

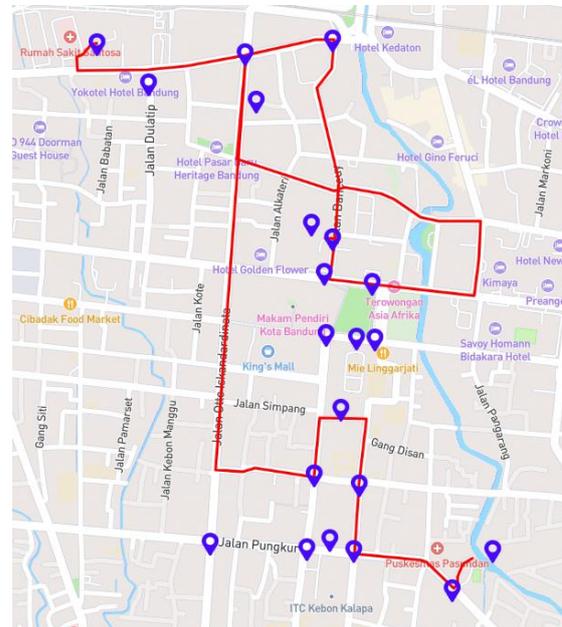
Pada tahap ini, dilakukan analisis desain mendalam terhadap data yang telah terpilih dan perancangan sistem berdasarkan hasil analisis tersebut. Dalam konteks penelitian ini, perancangan mencakup penerapan algoritma *Floyd-Warshall* dengan menggunakan *framework* Laravel. Berikut adalah desain *user interface* untuk *dashboard*.



Gambar 5 Dashboard

#### 4.4. Implementasi dan Uji Model

Pada pengembangan untuk implemetasi Algoritma *Floyd-Warshall* dalam memberikan rekomendasi rute terpendek untuk lokasi pengiriman barang pada sistem logistik Perusahaan logistik dilakukan melalui perbandingan antara perhitungan otomatis algoritma dengan perhitungan manual. Dalam konteks ini, jarak antar lokasi distribusi barang di Bandung dihitung, dan hasil perhitungan menggunakan implementasi Algoritma *Floyd-Warshall* pada sistem logistik Perusahaan logistik.



Gambar 6 Rute Sample A → D

Pada Gambar 11 tergambar rute berwarna merah yang menghubungkan Titik A → D. Informasi terkait mencakup jarak, titik awal (*start*), rute, dan titik akhir (*end*) yang ditampilkan di bagian bawah. perhitungan manual untuk mendapatkan rute terpendek dari lokasi distribusi barang di Bandung dengan

menggunakan Algoritma *Floyd-Warshall* dapat ditemukan pada Matriks hubung  $K = 1$  sampai dengan  $K = 4$  merupakan index process/matriks hubung  $1 \rightarrow 4$ . Hasil pengujian menunjukkan bahwa implementasi Algoritma *Floyd-Warshall* pada sistem logistik Perusahaan logistik telah berhasil. Uji coba rute terpendek dalam penelitian ini melibatkan situasi di mana pengantar paket akan melakukan pengiriman ke Titik A, B, C, dan D. Dengan menggunakan metode algoritma *Floyd Warshall* untuk menemukan rute terpendek dari node 10 ke node 11, 12, dan 13, hasilnya menunjukkan pada Tabel dibawah ini menunjukkan data jarak lokasi titik A, B, C, dan D.

**Tabel 1** Lokasi Sampel

Pos.	Node	Latitude	Longitude
A	10	-6.91979	107.606
B	11	-6.91559	107.601
C	12	-6.92742	107.6103
D	13	-6.92413	107.6067

**Tabel 2** Jarak Antar Lokasi

No	Lokasi Awal	Lokasi Tujuan	Jarak (Km)
1.	10	11	2.1 km
2.	10	12	2.9 km
3.	10	13	2 km
4.	11	13	2.3 km
5.	11	12	2.1 km
6.	12	11	1.5 km
7.	12	13	0.9 km
8.	13	12	0.8 km

Tahapan perhitungan algoritma *Floyd Warshall* untuk menetapkan rute terpendek diperlihatkan melalui representasi matriks. Pada matriks ini, setiap baris mencerminkan lokasi awal, sementara setiap kolom mencerminkan lokasi tujuan. Kemudian matriks tersebut dihitung menggunakan matriks hubung dengan nilai  $K$  dari 1 sampai dengan 4.

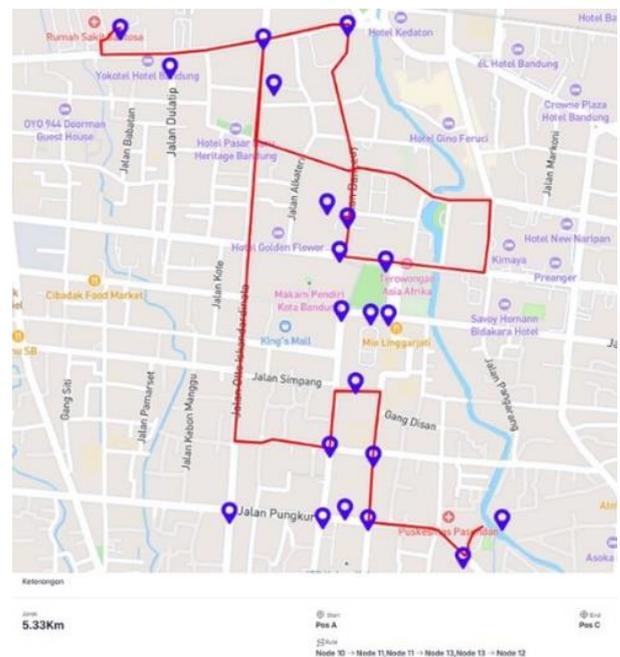
**Tabel 3** Matriks Akhir Rute Terpendek

#	10	11	12	13
10	0	2.2	5.3	4.5
11	~	0	3.1	2.3
12	~	1.6	0	0.9
13	~	2.4	0.8	0

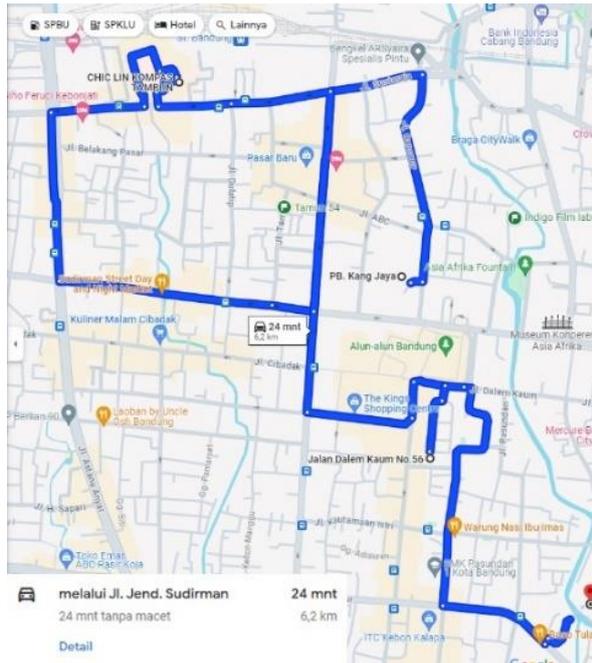
Dari matriks akhir rute terpendek menggunakan algoritma *floyd-warshall* didapat perhitungan :

1.  $A(10) \rightarrow B(11) + B(11) \rightarrow C(12) + C(12) \rightarrow D(13) = 2.2 + 3.1 + 0.9 = 6.2$
2.  $A(10) \rightarrow B(11) + B(11) \rightarrow D(13) + D(13) \rightarrow C(12) = 2.2 + 2.3 + 0.8 = 5.3$
3.  $A(10) \rightarrow C(12) + C(12) \rightarrow B(11) + B(11) \rightarrow D(13) = 5.3 + 1.6 + 2.3 = 9.2$
4.  $A(10) \rightarrow C(12) + C(12) \rightarrow D(13) + D(13) \rightarrow B(11) = 5.3 + 0.9 + 2.4 = 8.6$
5.  $A(10) \rightarrow D(13) + D(13) \rightarrow B(11) + B(11) \rightarrow C(12) = 4.5 + 2.4 + 3.1 = 10$
6.  $A(10) \rightarrow D(13) + D(13) \rightarrow C(12) + C(12) \rightarrow B(11) = 4.5 + 0.8 + 1.6 = 6.9$

Dari perhitungan di atas, ditemukan bahwa rute terpendek adalah node  $10 \rightarrow 11 + 11 \rightarrow 13 + 13 \rightarrow 12$  atau  $A \rightarrow B + B \rightarrow D + D \rightarrow C$ , dengan jarak berjumlah sejauh 5.3 km. Artinya, rute terpendek yang dipilih oleh pengirim paket saat ini adalah posisi awal – Titik B – Titik D – Titik C (akhir) yaitu sejauh 5.3 km. Hasil perhitungan manual ini menghasilkan rute yang identik dengan rute terpendek yang diperoleh melalui website sudah sesuai, Untuk melihat uji pembandingan rute yang sedang berjalan dengan rute *Google Maps* dapat dilihat pada gambar 7 dan 8.



**Gambar 7** Sistem Yang Sedang Berjalan



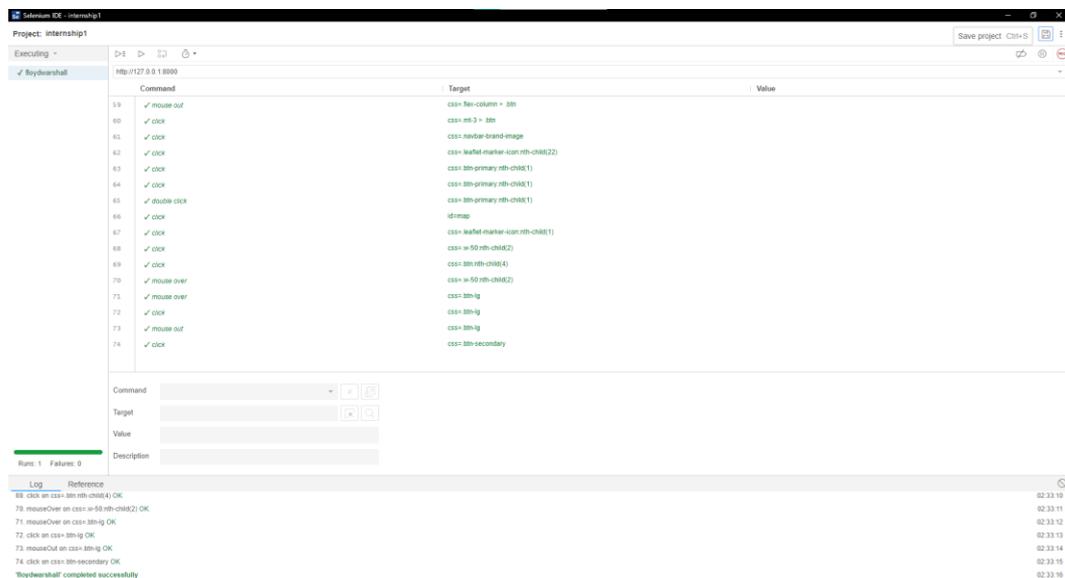
**Gambar 8** Google Maps

Pada tahap ini, dilakukan eksperimen perbandingan antara rute yang dibuat oleh sistem pada (gambar 7. bagian kiri) dengan jalur yang terlihat di *Google Maps* (gambar 11. bagian kanan). Dalam percobaan ini, ketika

menggunakan titik awal dan titik akhir yang sama, jalur yang ditampilkan oleh *Google Maps* memiliki panjang sejauh 6,2 km. Sebaliknya, melalui perhitungan sistem yang berjalan, jalur yang dihasilkan menunjukkan jarak yang lebih pendek, yakni sejauh 5,3 km. Temuan ini menunjukkan bahwa sistem memiliki kemampuan untuk menghasilkan jalur yang lebih efisien dibandingkan dengan layanan pemetaan seperti *Google Maps*. Adapun kekurangan yang ditemukan, pada tahap ini sistem yang sudah berjalan hanya diukur dari indikator jarak saja, tidak dengan waktu tempuh dan keadaan *traffic*/lalu lintas yang ada secara *real-time*.

#### 4.5. Pengujian Sistem

Sistem diuji menggunakan *Selenium* IDE untuk memastikan konsistensi dan keandalan fungsionalitasnya. *Selenium* IDE berfungsi sebagai alat otomatisasi pengujian yang menjalankan berbagai skenario pengujian untuk memverifikasi kinerja aplikasi sesuai dengan spesifikasi yang telah ditetapkan sebelumnya. Hasil pengujian sistem dapat dilihat pada gambar



**Gambar 9** Pengujian Sistem

## 5. KESIMPULAN DAN SARAN

Dalam penutup, penelitian ini berhasil menerapkan Algoritma *Floyd-Warshall* pada sistem logistik perusahaan, meningkatkan rute distribusi dengan baik. Sistem mampu menghasilkan rute terpendek, jarak antar rute, dan mendukung keputusan logistik cerdas.

Hasil menunjukkan bahwa sistem yang dibangun memiliki kemampuan untuk menghasilkan rute yang lebih efisien dan lebih singkat dibandingkan dengan layanan pemetaan seperti *Google Maps* memiliki panjang sejauh 6,2 km. Sebaliknya, melalui perhitungan sistem yang berjalan, jalur yang

dihasilkan menunjukkan jarak yang lebih pendek, yakni sejauh 5,3 km pada rute yang sama. Adapun sistem saat ini hanya memperhitungkan jarak tanpa mempertimbangkan waktu tempuh dan lalu lintas *real-time*.

## 6. DAFTAR PUSTAKA

- [1] S. Min, Z. G. Zacharia, and C. D. Smith, "Defining Supply Chain Management: In the Past, Present, and Future," in *Journal of Business Logistics*, Wiley-Blackwell, Mar. 2019, pp. 44–55. doi: 10.1111/jbl.12201.
- [2] Y. Liu, "An optimization-driven dynamic vehicle routing algorithm for on-demand meal delivery using drones," *Comput Oper Res*, vol. 111, pp. 1–20, Nov. 2019, doi: 10.1016/j.cor.2019.05.024.
- [3] I. Tiffani and H. Wandebori, "Proposed Restructuring Strategy for Sustainable Future of PT Pos Indonesia," *European Journal of Business and Management Research*, vol. 5, no. 4, Jul. 2020, doi: 10.24018/ejbm.2020.5.4.384.
- [4] P. Ricardianto *et al.*, "Service quality and timeliness: Empirical evidence on the parcel delivery service in Indonesia," *Uncertain Supply Chain Management*, vol. 11, no. 4, pp. 1645–1656, Sep. 2023, doi: 10.5267/j.uscm.2023.7.004.
- [5] E. D. Madyatmadja, H. Nindito, R. A. Bhaskoro, A. Verasius, D. Sano, and C. P. M. Sianipar, "Algorithm To Find Tourism Place Shortest Route: A Systematic Literature Review," *J Theor Appl Inf Technol*, vol. 28, no. 4, 2021, [Online]. Available: [www.jatit.org](http://www.jatit.org)
- [6] Faridawaty Marpaung, Arnita, Tommy, and Putri Harliana, "Floyd-Warshall Algorithm for Web-Based Route Optimization for a Road Network Study in Medan City," *Jurnal Multidisiplin Madani*, vol. 3, no. 11, pp. 2192–2208, Nov. 2023, doi: 10.55927/mudima.v3i11.6674.
- [7] B. Logeshwary Natarajan and M. Balaji, "Application of Graph Theory in Online Network Services to Determine the Shortest Journey," *Int. J. Advanced Networking and Applications*, pp. 4030–4034, 2019.
- [8] A. Khamami and R. Saputra, "The shortest path search application based on the city transport route in Semarang using the Floyd-warshall algorithm," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jun. 2019. doi: 10.1088/1742-6596/1217/1/012116.
- [9] J. Skarding, B. Gabrys, and K. Musial, "Foundations and Modeling of Dynamic Networks Using Dynamic Graph Neural Networks: A Survey," *IEEE Access*, vol. 9, pp. 79143–79168, 2021, doi: 10.1109/ACCESS.2021.3082932.
- [10] A. Akbarkhodjaev and H. E. Bal, "Development of an indoor wayfinding system with a focus on crowd avoidance," 2022.
- [11] Z. Lei, H. Zhou, W. Hu, and G. P. Liu, "Unified and Flexible Online Experimental Framework for Control Engineering Education," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 1, pp. 835–844, Jan. 2022, doi: 10.1109/TIE.2021.3053903.
- [12] M. Laaziri, K. Benmoussa, S. Khouliji, K. Mohamed Larbi, and A. El Yamami, "A comparative study of laravel and symfony PHP frameworks," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 1, p. 704, Feb. 2019, doi: 10.11591/ijece.v9i1.pp704-712.
- [13] F. H. B. de Souza *et al.*, "Parallelization Of Shortest Path Class Algorithms: A Comparative Analysis," *Pesquisa Operacional*, vol. 43, 2023, doi: 10.1590/0101-7438.2023.043.00272130.
- [14] Y. Benmessaoud, L. Cherrat, and M. Ezziyyani, "Real-Time Self-Adaptive Traffic Management System for Optimal Vehicular Navigation in Modern Cities," *Computers*, vol. 12, no. 4, Apr. 2023, doi: 10.3390/computers12040080.
- [15] I. H. Toroslu, "Improving The Floyd-Warshall All Pairs Shortest Paths Algorithm," 2021.
- [16] E. D. Ekmen, "A Study on performance evaluation of optimization algorithms in the shortest path problem," 2020.
- [17] B. K. Kumar and E. S. Reddy, "Modified Floyd Warshall Algorithm for cache management in information centric network," *International Journal of Intelligent Engineering and Systems*, vol. 13, no. 1, pp. 146–155, Feb. 2020, doi: 10.22266/ijies2020.0229.14.

- [18] R. L. Guðmundarson and G. W. Peters, "Assessing Portfolio Diversification via Two-Sample Graph Kernel Inference," 2023. Accessed: Jan. 02, 2024. [Online]. Available: <https://ssrn.com/abstract=4348306>
- [19] B. Sugiantoro, M. Anshari, and D. Sudrajat, "Developing Framework for Web Based e-Commerce: Secure-SDLC," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jul. 2020. doi: 10.1088/1742-6596/1566/1/012020.
- [20] H. J. Christanto and Y. A. Singgalen, "Analysis and Design of Student Guidance Information System through Software Development Life Cycle (SDLC) dan Waterfall Model," *Journal of Information Systems and Informatics*, vol. 5, no. 1, pp. 259–270, Mar. 2023, doi: 10.51519/journalisi.v5i1.443.
- [21] M. Stauffer, *Laravel: Up & Running*. O'Reilly Media, Inc, 2023. Accessed: Jan. 11, 2024. [Online]. Available: <https://books.google.co.id/books?id=LZrWEAAAQBAJ&printsec=frontcover&hl=id#v=onepage&q&f=false>