

IMPLEMENTASI ALGORITMA FLOYD-WARSHALL UNTUK OPTIMASI RUTE TERPENDEK ANTAR KOTA DI JAWA TIMUR

IMPLEMENTATION OF THE FLOYD-WARSHALL ALGORITHM FOR OPTIMIZING THE SHORTEST ROUTE BETWEEN CITIES IN EAST JAVA

Sumunarsih¹, Muhammad Fachrie²

Program Studi Informatika, Fakultas Sains dan Teknologi, Universitas Teknologi Yogyakarta¹

Program Studi Sains Data, Fakultas Sains dan Teknologi, Universitas Teknologi Yogyakarta²

Sumunarsih16@gmail.com¹

ABSTRACT

The transportation sector plays an important role in improving economic activity and community mobility, especially in areas with good connectivity such as East Java Province. However, the increasing number of vehicles and the complexity of the road network often cause inefficiencies in choosing inter-city travel routes. These problems contribute to higher operational costs, longer travel times, and a decline in the quality of transportation services. To address these issues, this study aims to design and develop an inter-city route optimization system in East Java using the Floyd–Warshall algorithm. The methodology used includes collecting data on distances between cities from the Kaggle platform, which is then verified through Google Maps. The Floyd–Warshall algorithm is applied to find the shortest route between each pair of cities. The system was developed as a web-based application using the Python programming language and the Flask framework to enable direct interaction with users. The results of the experiment show that the independent application of the Floyd–Warshall algorithm achieved an accuracy rate of 87%. In addition, this system also provides a visual representation of the optimized routes through integration with Google Maps, so that users can better understand the recommended routes in their geographical context. In conclusion, this study successfully created a Floyd–Warshall-based route optimization system that is capable of providing the shortest route calculation results with a fairly good level of accuracy. It is hoped that the implementation of this system can contribute to the application of Intelligent Transportation Systems (ITS) in Indonesia.

Keywords: Route Optimization, Floyd-Warshall, Shortest, Transportation, Weighted Graph, East Java Cities

ABSTRAK

Sektor transportasi memiliki peranan krusial dalam mengoptimalkan aktivitas ekonomi dan pergerakan masyarakat, terutama di daerah dengan konektivitas yang baik seperti Provinsi Jawa Timur. Akan tetapi, lonjakan jumlah kendaraan serta kerumitan jaringan jalan sering kali mengakibatkan ketidakefisienan dalam memilih rute perjalanan antarkota. Masalah ini berdampak pada peningkatan biaya operasional, lama perjalanan yang tidak efisien, serta penurunan kualitas layanan transportasi. Untuk mengatasi permasalahan ini, penelitian ini bertujuan untuk merancang dan mengembangkan sistem pengoptimalan rute antarkota di Jawa Timur dengan memanfaatkan algoritma Floyd–Warshall. Metode yang diterapkan dalam penelitian ini mencakup pengumpulan data jarak antar kota menggunakan platform Kaggle yang diverifikasi melalui Google Maps. Selanjutnya, proses dirancang dengan algoritma Floyd–Warshall untuk menentukan jarak terpendek antara setiap pasangan kota. Sistem ini dibangun berbasis web dengan menggunakan bahasa pemrograman Python dan framework Flask untuk memungkinkan interaksi langsung dengan pengguna. Hasil dari pengujian menunjukkan bahwa penerapan algoritma Floyd–Warshall secara independen menghasilkan akurasi sebesar 87%. Selain itu, sistem ini juga menampilkan rute yang dioptimalkan secara visual melalui integrasi dengan Google Maps, sehingga pengguna dapat lebih memahami jalur yang disarankan dalam konteks geografis. Sebagai kesimpulan, penelitian ini berhasil menciptakan sistem pengoptimalan rute berbasis Floyd–Warshall yang mampu memberikan hasil perhitungan rute terpendek dengan tingkat akurasi lumayan baik. Diharapkan, implementasi sistem ini dapat berkontribusi pada penerapan Intelligent Transportation System (ITS) di Indonesia.

Kata Kunci: Optimasi Rute, Floyd-Warshall, Terpendek, Transportasi, Graf Berbobot, Kota Jawa Timur

PENDAHULUAN

Sektor transportasi berhasil tumbuh 21,27% pada triwulan II-2022, jika dibandingkan periode yang sama tahun lalu

(Badan Pusat Statistik 2021). Sebelumnya, pada triwulan I-2022 juga mengalami pertumbuhan positif mencapai 15,79%. “Hasil ini bisa menjadi indikator bahwa

pemulihan di sektor transportasi yang telah kita bersama upayakan sudah terjadi. Momentum ini harus kita jaga agar pada triwulan ketiga dan keempat di tahun 2022 trennya terus meningkat,” ujar Menhub, Senin (8/7) (Kementerian Perhubungan Republik Indonesia 2022). Menhub mengatakan, pada awal tahun 2022 lalu telah menyiapkan sejumlah strategi untuk mendorong percepatan pemulihan sektor transportasi yang terdampak akibat pandemi Covid-19. Transportasi adalah sektor yang sangat penting bagi mobilitas masyarakat, terutama untuk perjalanan jarak jauh antar kota.

Pertumbuhan yang signifikan dalam industri transportasi menandakan adanya pemulihan setelah pandemi Covid-19, khususnya dalam hal mobilitas antar kota. Namun, meskipun ada peningkatan, efisiensi sistem transportasi belum sepenuhnya terwujud, terutama dalam penentuan rute perjalanan antar kota. Masih terdapat banyak perusahaan transportasi dan pelanggan yang bergantung pada metode pemilihan rute tradisional tanpa memanfaatkan teknologi optimasi berbasis algoritma. Dengan bertambahnya konektivitas antar kota di Provinsi Jawa Timur, tantangan baru muncul dalam menemukan rute perjalanan yang paling efisien dari segi jarak dan durasi. Ketidakefisienan ini mempengaruhi peningkatan biaya operasional dan lamanya waktu perjalanan, juga menghalangi efektivitas layanan transportasi. Walaupun ada berbagai algoritma untuk menentukan rute, penerapan praktis dalam sistem transportasi yang didasarkan pada data lokal masih jarang, terutama dengan algoritma Floyd-Warshall yang mampu menghitung jarak terpendek di antara semua pasangan simpul dalam graf berbobot.

Optimasi rute menjadi solusi strategis untuk meminimalkan jarak tempuh, memperpendek durasi perjalanan, dan menekan biaya operasional. Salah satu algoritma yang dapat dimanfaatkan untuk mengatasi persoalan ini adalah algoritma

Floyd-Warshall. Algoritma ini terkenal karena kemampuannya dalam menghitung jarak terpendek antara semua pasangan simpul dalam graf berbobot, sehingga bisa diterapkan dalam sistem optimasi rute transportasi. Pengembangan sistem optimasi rute antar kota dengan memanfaatkan algoritma Floyd-Warshall diharapkan mampu memberikan solusi dalam menentukan jalur tercepat dan termudah. Disamping itu, sistem ini juga dapat berfungsi sebagai referensi bagi perusahaan transportasi lainnya dalam meningkatkan kualitas layanan yang mereka tawarkan.

TINJAUAN PUSTAKA

Penelitian yang dilakukan oleh (Sari et al., 2023), tentang cara menentukan jalan tercepat untuk evakuasi saat keadaan darurat di Boyolali sangat penting karena menggunakan algoritma optimasi graf Floyd-Warshall untuk mempercepat penanganan korban kecelakaan. Temuan dari penelitian menunjukkan bahwa algoritma ini bisa memberikan solusi yang cepat dan efisien untuk menemukan rute evakuasi terdekat berdasarkan jarak dan lokasi dari insiden. Salah satu kritik yang bisa disampaikan adalah bahwa penelitian di Boyolali belum mengintegrasikan data geospasial maupun kondisi jalan yang sebenarnya, yang dapat memengaruhi ketepatan hasil. Disisi lain, penelitian yang dilakukan di Jawa Timur menggabungkan algoritma tersebut dengan Google Maps untuk memvalidasi data dan memvisualisasikan rute, sehingga menghasilkan sistem yang lebih realistis dan dapat diterapkan dalam konteks transportasi modern.

Penelitian yang dilakukan oleh (Rofiq, 2022), membahas penerapan algoritma Floyd-Warshall untuk mengoptimalkan jalur menuju berbagai objek wisata di wilayah Malang Raya. Hasil penelitian menunjukkan bahwa algoritma Floyd-Warshall mampu menghitung jarak terpendek dengan tingkat akurasi yang baik dan memberikan rekomendasi rute optimal

di antara titik-titik tujuan wisata. Namun, kelemahan utama dari penelitian ini adalah sistem yang dikembangkan belum terintegrasi dengan data geospasial aktual maupun platform peta digital seperti Google Maps, sehingga hasil optimasi masih bersifat teoretis dan kurang interaktif bagi pengguna. Selain itu, tidak adanya visualisasi rute dalam bentuk antarmuka web membuat hasilnya kurang praktis untuk diaplikasikan secara langsung. Sebaliknya, penelitian Implementasi Algoritma Floyd–Warshall untuk Optimasi Rute Terpendek Antar Kota di Jawa Timur memperbaiki keterbatasan tersebut dengan mengintegrasikan algoritma menggunakan framework Flask dan Google Maps, memungkinkan pengguna berinteraksi langsung dengan sistem serta melihat hasil rute secara geografis.

Penelitian yang dilakukan oleh (Sitinjak, 2022), membahas analisis antara dua algoritma pencarian jalur terpendek, yaitu Dijkstra dan Floyd–Warshall, dalam konteks pencarian rute menuju fasilitas kesehatan di area urban dengan kepadatan lalu lintas yang tinggi. Penelitian ini memberikan kontribusi yang signifikan karena menunjukkan perbedaan dalam efisiensi kedua algoritma, di mana Dijkstra lebih unggul untuk pencarian dari satu titik ke banyak titik (single source shortest path), sedangkan Floyd–Warshall lebih efektif dalam menghitung jarak terpendek di antara semua pasangan simpul (all pairs shortest path) sekaligus. Namun, studi di Medan masih memiliki kelemahan karena belum memanfaatkan data geospasial yang aktual atau platform peta digital seperti Google Maps, sehingga hasil yang diperoleh bersifat simulatif dan kurang aplikatif di lapangan. Selain itu, penelitian ini juga belum menyediakan antarmuka visual berbasis web yang mampu menampilkan hasil rute secara geografis dan interaktif. Disisi lain, penelitian Implementasi Algoritma Floyd–Warshall untuk Optimasi Rute Terpendek Antar Kota di Jawa Timur mengatasi masalah ini dengan mengintegrasikan algoritma Floyd–

Warshall dalam sistem berbasis web menggunakan framework Flask dan Google Maps, sehingga hasil optimasi dapat divisualisasikan secara langsung sesuai dengan kondisi jalan yang sebenarnya.

Penelitian yang dilakukan oleh (Khairi et al., 2019), merupakan kemajuan lebih lanjut dalam bidang pengoptimalan rute transportasi yang mengintegrasikan dua metode komputasi, yaitu algoritma Floyd–Warshall dan logika Fuzzy. Penelitian ini sangat berharga karena tidak hanya melihat jarak antar kota sebagai faktor utama, tetapi juga mempertimbangkan faktor-faktor dinamis seperti keadaan lalu lintas dan waktu perjalanan aktual, sehingga mampu menciptakan sistem optimasi yang lebih responsif terhadap kondisi riil. Namun, penelitian hybrid Fuzzy–Floyd Warshall memiliki kelemahan, yaitu memerlukan waktu komputasi yang lebih panjang serta infrastruktur data yang lebih rumit, seperti sensor lalu lintas dan integrasi data secara real-time. Disisi lain, penelitian Implementasi Algoritma Floyd–Warshall untuk Optimasi Rute Terpendek Antar Kota di Jawa Timur menghadirkan pendekatan yang lebih sederhana, lebih efisien dalam hal komputasi.

Penelitian yang dilakukan oleh (Novandi, 2013), bertujuan untuk melakukan analisis perbandingan antara dua algoritma graf dalam mencari jalur terpendek antara sepasang titik, dengan harapan untuk memberikan wawasan lebih dalam tentang perbedaan efisiensi, kompleksitas waktu, dan sifat penerapan kedua algoritma tersebut. Dari hasil penelitian, terungkap bahwa algoritma Dijkstra lebih unggul dalam pencarian lintasan tunggal (single pair shortest path) karena memiliki kompleksitas yang lebih ringan, sedangkan Floyd–Warshall lebih efektif dalam pencarian jarak terpendek antara semua pasang titik (all pairs shortest path) karena dapat menghasilkan matriks jarak lengkap dalam satu kali proses komputasi. Namun, penelitian ini memiliki batasan karena belum menghubungkan

hasil analisa dengan penerapan praktis dalam bidang transportasi atau sistem navigasi, sehingga hasilnya lebih bersifat teoritis. Disisi lain, penelitian Implementasi Algoritma Floyd–Warshall untuk Optimasi Rute Terpendek Antar Kota di Jawa Timur memperluas penggunaan algoritma tersebut dengan cara nyata dalam konteks optimasi rute transportasi antar kota, dengan mengintegrasikan sistem berbasis web yang memanfaatkan Flask dan Google Maps untuk menciptakan visualisasi interaktif yang merefleksikan kondisi geografis sebenarnya.

Penelitian yang dilakukan oleh (Pipit Mulyah, Dyah Aminatun, Sukma Septian Nasution, Tommy Hastomo, Setiana Sri Wahyuni Sitepu, 2020), optimasi rute merupakan cara untuk menemukan jalur terpendek atau paling efisien dalam transportasi, dengan mempertimbangkan sejumlah faktor seperti kapasitas kendaraan dan waktu layanan mengkaji pengoptimalan rute sebagai cara untuk menentukan jalur yang paling efisien dalam sistem transportasi dengan mempertimbangkan berbagai elemen seperti kapasitas kendaraan, waktu pelayanan, dan efisiensi operasional. Meskipun demikian, penelitian ini masih berupa konsep karena belum melibatkan perhitungan algoritmik atau sistem yang berbasis teknologi spasial. Disisi lain, penelitian Implementasi Algoritma Floyd–Warshall untuk Optimasi Rute Terpendek Antar Kota di Jawa Timur membawa konsep tersebut ke dalam praktik dengan menggunakan algoritma Floyd–Warshall dalam sistem yang berbasiskan Flask dan Google Maps, yang mampu secara otomatis menghitung rute terpendek serta menampilkan visualisasi rute dalam bentuk geografis dan interaktif.

Penelitian yang dilakukan oleh (Ningrum et al., 2023), menekankan pentingnya pencarian jalur distribusi yang paling pendek untuk meningkatkan efektivitas transportasi dan logistik dengan menggabungkan metode Floyd–Warshall dan algoritma Greedy sebagai solusi

terhadap permasalahan ini. Meski demikian, penelitian ini masih memiliki kekurangan karena bersifat simulatif dan tidak terhubung dengan data geospasial serta sistem peta digital seperti Google Maps, sehingga hasil optimasinya belum sepenuhnya sesuai dengan keadaan geografis yang sebenarnya. Disisi lain, penelitian Implementasi Algoritma Floyd–Warshall untuk Optimasi Rute Terpendek Antar Kota di Jawa Timur menawarkan metode yang lebih praktis, menyeluruh, dan dapat diterapkan dengan memadukan algoritma Floyd–Warshall dalam sistem berbasis web menggunakan Flask dan Google Maps, sehingga pengguna dapat melihat hasil perhitungan rute secara interaktif dan realistis.

Penelitian yang dilakukan oleh (Vulandari et al., 2021), menjelaskan bahwa metode Floyd–Warshall adalah cara yang efisien untuk menentukan jarak terpendek antara semua pasangan titik dalam graf berbobot. Namun, penelitian tersebut memiliki batasan karena hanya bersifat teori dan belum diterapkan dalam sistem aplikasi yang dapat digunakan langsung oleh pengguna, dan juga belum mengintegrasikan data geospasial serta visualisasi peta digital untuk menampilkan hasil secara interaktif. Disisi lain, studi berjudul Implementasi Algoritma Floyd–Warshall untuk Optimasi Rute Terpendek Antar Kota di Jawa Timur mengembangkan ide tersebut ke dalam bentuk aplikasi praktis berbasis web dengan memadukan algoritma Floyd–Warshall bersama dengan Flask dan Google Maps, sehingga hasil perhitungan rute bisa divisualisasikan secara langsung dan mudah dipahami oleh pengguna.

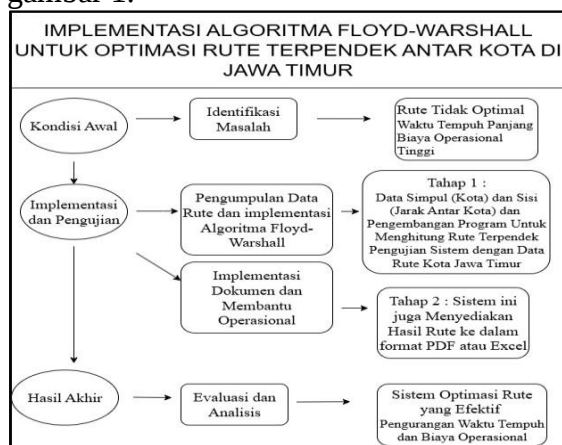
Penelitian yang dilakukan oleh (Inayah et al., 2023), memberikan penjelasan yang terperinci tentang proses kerja algoritma Floyd–Warshall dalam mengidentifikasi rute tercepat. Algoritma ini dimulai dari setiap titik awal dan memperpanjang perjalanan dengan mempertimbangkan setiap simpul secara bergantian sampai tujuan tercapai dengan

bobot total yang paling rendah. Meskipun demikian, riset ini masih tergolong teoritis, karena belum diterapkan dalam sistem digital yang berbasis data geospasial atau peta interaktif, sehingga penerapannya dalam konteks transportasi yang nyata masih terbatas. Sementara itu, penelitian Implementasi Algoritma Floyd–Warshall untuk Optimasi Rute Terpendek Antar Kota di Jawa Timur menghadirkan konsep tersebut dengan pendekatan praktis dan aplikatif melalui penggunaan algoritma dalam sistem yang berbasis Flask dan Google Maps, yang mampu menunjukkan hasil perhitungan rute terpendek dengan cara yang visual dan interaktif.

METODE

A. Kerangka penelitian

Penelitian ini adalah semacam penelitian murni dan terapan. Algoritma Floyd–Warshall digunakan untuk mengubah bobot jalur menjadi matriks kuadratik $\times n$ (jumlah titik Bertex), dan kemudian dianalisis untuk menentukan rute terpendek. Hasil dari algoritma ini adalah rute terpendek kota jawa timur. Oleh karena itu, penelitian ini diharapkan dapat memberikan solusi dalam menentukan saluran distribusi yang efisien dan efektif. Langkah-langkah penelitian yang akan dilakukan dapat dilihat pada ilustrasi di gambar 1.



Gambar 1. Kerangka Penelitian

B. Data Penelitian

Cara Mendapatkan Data

1. Data Sekunder

Informasi untuk penelitian ini diperoleh dari platform Kaggle, yang dikenal sebagai salah satu sumber data publik yang dapat diandalkan dan sering diakses oleh para peneliti serta profesional data. Dataset ini memuat rincian mengenai jarak antar kota di Provinsi Jawa Timur yang digunakan sebagai acuan dalam menghitung rute, untuk memastikan kevalidan dan ketepatan data, dilakukan verifikasi silang dengan hasil perhitungan jarak dan perkiraan waktu perjalanan yang diambil dari Google Maps. Langkah ini bertujuan agar data yang digunakan benar-benar mencerminkan kondisi geografis dan infrastruktur transportasi yang ada.

2. Data Geospasial

Dalam mencari informasi geospasial yang tepat, penelitian ini menggunakan peta digital Google Maps untuk mendapatkan koordinat geografis masing-masing kota yang terlibat. Selain mengukur jarak lurus, Google Maps juga diandalkan untuk menentukan jarak tempuh nyata berdasarkan jalur perjalanan yang ada. Data ini berfungsi sebagai landasan dalam pembuatan graf dan penentuan bobot antar node dalam sistem optimasi. Selain itu, penerapan algoritma Floyd–Warshall dalam studi ini didukung oleh tinjauan literatur dari berbagai sumber akademik dan ilmiah, seperti buku teori graf, artikel penelitian, dan studi sebelumnya yang berhubungan dengan topik optimasi rute serta pemetaan jaringan. Data dapat dilihat pada tabel 1.

Tabel 1. Data Kota Jawa Timur

Nama Kota	Koordinat longitude	Koordinat latitude
Kab. Bangkalan	-7.037129	112.909425
Kab. Banyuwangi	-8.219182	114.369998
Kab. Blitar	-8.035170	112.150683
Kab. Bojonegoro	-7.281268	111.766610
Kab. Bondowoso	-7.973076	113.898576
Kab. Gresik	-7.127873	112.569136
Kab. Jember	-8.121711	113.693487
Kab. Jombang	-7.544421	112.287347
Kab. Kediri	-7.790040	112.190759
Kab. Lamongan	-7.095168	112.332087
Kab. Lumajang	-8.039563	113.136790
Kab. Madiun	-7.537089	111.626055
Kab. Magetan	-7.630444	111.351517
Kab. Malang	-8.240963	112.721039

Kab. Mojokerto	-7.558437	112.477666
Kab. Nganjuk	-7.567134	111.901325
Kab. Ngawi	-7.434987	111.331909
Kab. Pacitan	-8.100962	111.138638
Kab. Pamekasan	-7.099435	113.521338
Kab. Pasuruan	-7.765100	112.859956
Kab. Ponorogo	-7.858559	111.464424
Kab. Probolinggo	-7.858782	113.475609
Kab. Sampang	-7.026465	113.240058
Kab. Sidorejo	-7.469204	112.668426
Kab. Situbondo	-7.780816	114.189608
Kab. Sumenep	-6.907187	113.912601
Kab. Trenggalek	-8.186198	111.610602
Kab. Tuban	-6.894629	112.040750
Kab. Tulungagung	-8.051380	111.963922
Kota Batu	-7.881434	112.525275
Kota Blitar	-8.094997	112.158923
Kota Kediri	-7.840202	112.019218
Kota Madiun	-7.617421	111.535606
Kota Malang	-7.930532	112.634466
Kota Mojokerto	-7.468816	112.441013
Kota Pasuruan	-7.645612	112.902086
Kota Probolinggo	-7.772450	113.204537
Kota Surabaya	-7.256814	112.751744

Sumber: Google Maps

C. Analisis Perancangan

Tahapan Penelitian dalam pembuatan sistem optimasi rute terpendek diawali dengan proses perancangan yang efisien, selanjutnya melakukan perancangan yang lebih mendetail seperti gambar 2 dibawah ini.



Gambar 2. Analisis Perancangan

Gambar 2 merupakan langkah awal yang dilakukan dalam analisis kebutuhan fungsional dalam mengidentifikasi secara rinci fitur-fitur yang dibutuhkan pada sistem. Langkah ini cukup penting agar sistem yang dikembangkan dapat memenuhi kebutuhan pengguna secara

optimal. Setelah itu, analisis kebutuhan non-fungsional dilakukan guna memastikan bahwa sistem memiliki kinerja, dan aspek non-fungsional lainnya sesuai dengan standar yang ditentukan.

Kemudian dalam tahap perancangan konseptual, dibuat dengan flowchart untuk memvisualisasikan dengan jelas komponen sistem yang akan digunakan. Langkah terakhir dalam tahap perancangan adalah merancang sistem secara menyeluruh, memastikan setiap komponen terintegrasi dan konsisten. Perancangan ini menjadi dasar untuk implementasi sistem optimasi yang sesuai dengan kebutuhan pengguna. Oleh karena itu, penelitian ini akan membahas secara mendalam dasar penting dalam implementasi algoritma Floyd-warshall untuk optimasi rute terpendek antarkota jawa timur.

1. Analisis Kebutuhan Fungsional

Kebutuhan fungsional adalah evaluasi terhadap kemampuan sistem atau aspek-aspek yang akan ditingkatkan dalam sistem yang dirancang serta disusun. Kebutuhan fungsional dapat dijelaskan dalam 3 hal yaitu kebutuhan jenis masukan, proses-proses yang dibutuhkan, dan luaran yang diharapkan

Kebutuhan Masukan

a. Data Rute Jarak

Informasi mengenai rute jarak antarkota di Provinsi Jawa Timur diperoleh dari platform Kaggle, yang merupakan sumber data publik berskala internasional yang sering dipakai dalam penelitian di bidang data sains dan sistem transportasi. Dataset ini memuat rincian tentang jarak antara kota yang menjadi acuan utama dalam penghitungan dan optimasi rute.

b. Data Geospasial

Data geospasial diambil dari Google Maps, yang berfungsi untuk memperoleh koordinat geografis (longitude dan latitude) masing-masing kota sebagai simpul dalam sistem. Google Maps tidak hanya memberikan

lokasi dalam bentuk koordinat, tetapi juga digunakan untuk mengukur jarak sesungguhnya dan perkiraan waktu tempuh berdasarkan keadaan rute yang ada.

Kebutuhan Luaran

a. Jalur Tercepat

Jalur tercepat adalah hasil dari analisis sistem yang menunjukkan rute dengan waktu perjalanan paling cepat antara dua tempat yang telah dipilih. Penentuan rute tercepat didasarkan pada algoritma Floyd–Warshall yang memperhitungkan bobot graf dalam bentuk jarak dan kecepatan rata-rata antara kota. Jalur tercepat tidak selalu berarti jarak terpendek, melainkan rute yang menawarkan efisiensi waktu tertinggi.

b. Durasi Perjalanan

Durasi perjalanan memberikan perkiraan waktu yang diperlukan untuk menyelesaikan rute yang dipilih. Perkiraan ini didasarkan pada data jarak antara kota yang didapat dari Google Maps serta asumsi mengenai kecepatan rata-rata kendaraan di setiap jalur. Dengan adanya informasi tentang durasi perjalanan, pengguna bisa membandingkan beberapa pilihan rute dan memilih yang paling efisien sesuai kebutuhan mereka.

c. Peta Visual

Peta visual adalah elemen penting dalam sistem yang berfungsi untuk menampilkan hasil perhitungan rute dalam bentuk visual yang interaktif. Kolaborasi dengan peta digital seperti Google Maps memungkinkan pengguna untuk melihat lokasi kota, jalur yang ditempuh, serta titik awal dan tujuan dengan jelas. Fitur ini memudahkan pengguna dalam memahami konteks geografis dari hasil optimasi yang diberikan sistem.

d. Informasi Jalur

Informasi jalur memberikan data lengkap mengenai rute perjalanan, termasuk titik transit antar kota, jarak antar lokasi, serta total jarak yang harus

ditempuh. Setiap bagian perjalanan dijelaskan secara terperinci agar pengguna dapat memahami struktur rute yang dihasilkan oleh sistem. Fitur ini tidak hanya menyajikan gambaran umum tentang rute, tetapi juga menunjukkan rincian penting seperti urutan kota yang dilalui dan jarak antar setiap titik.

e. Saran Rute

Saran rute adalah fitur yang memberikan rekomendasi jalur terbaik berdasarkan kriteria tertentu, seperti waktu tempuh tercepat, jarak terpendek, atau kombinasi di antara keduanya. Sistem menganalisis semua kemungkinan rute yang dihitung oleh algoritma Floyd–Warshall, lalu memilih rute yang paling sesuai dengan preferensi pengguna. Pengguna bisa menentukan kriteria sesuai kebutuhan mereka, misalnya prioritas waktu untuk transportasi penumpang atau prioritas jarak untuk efisiensi bahan bakar dalam pengiriman barang.

2. Analisis Kebutuhan Non Fungsional

Kebutuhan non-fungsional meliputi sistem yang perlu dikembangkan dengan mempertimbangkan faktor-faktor kualitas seperti kecepatan dalam menghitung rute, ketersediaan akses sistem yang terus-menerus, perlindungan data dan kontrol akses pengguna, serta kemudahan dalam penggunaan dan pemeliharaan. Disamping itu, sistem harus dapat dijalankan di berbagai platform, mampu berkembang untuk mendukung addition data kota, dan memiliki antarmuka yang responsif. Aspek-aspek ini sangat penting untuk memastikan bahwa sistem beroperasi dengan optimal, dapat diandalkan, dan berkelanjutan dalam meningkatkan efisiensi transportasi antar kota di Jawa Timur.

Kebutuhan Perangkat Lunak

a. Visual Studio Code

Visual Studio Code berfungsi sebagai platform pengembangan terintegrasi

- (IDE) untuk menulis, mengedit, dan menjalankan kode program. Software ini mendukung banyak bahasa pemrograman, termasuk Python, dan dilengkapi dengan berbagai ekstensi yang mempermudah proses debugging serta pengujian pada sistem berbasis web.
- b. Python
Bahasa pemrograman Python berperan sebagai bahasa utama dalam pengembangan sistem karena memiliki tata bahasa yang sederhana dan efisien, serta didukung oleh beragam pustaka untuk keperluan komputasi, visualisasi data, dan pemrograman web. Python juga menjadi dasar utama dalam penerapan algoritma Floyd–Warshall di dalam sistem optimasi rute ini.
 - c. Flask
Flask berfungsi sebagai framework web yang menggunakan bahasa Python untuk membuat antarmuka pengguna yang interaktif. Flask memungkinkan pengembang untuk membuat aplikasi web yang ringan dan dapat disesuaikan, sehingga sistem dapat menunjukkan hasil perhitungan dan visualisasi rute secara langsung melalui web browser.
 - d. Google Maps
Google Maps digunakan untuk menampilkan peta digital serta menghitung jarak dan koordinat geografis antara kota-kota di Jawa Timur. Integrasi dengan Google Maps membantu sistem dalam memberikan visualisasi rute yang tepat dan mendukung pengecekan hasil perhitungan algoritma dengan kondisi geografis yang nyata.
 - e. Windows
Sistem operasi Windows berfungsi sebagai platform utama untuk pengembangan dan pengujian sistem. Pemilihan Windows didasari oleh tingkat kompatibilitasnya yang tinggi dengan berbagai alat pemrograman, serta kemudahan dalam pengaturan server lokal dan instalasi pustaka Python.
 - f. Server Localhost
Server localhost dimanfaatkan untuk menjalankan aplikasi secara lokal selama proses pengembangan dan pengujian. Dengan menggunakan localhost, pengembang bisa menguji semua fitur sistem tanpa memerlukan koneksi internet eksternal, sehingga memperlancar proses debugging dan validasi sistem sebelum dipublikasikan ke server publik.
 - g. PythonAnywhere
PythonAnywhere berfungsi sebagai server hosting berbasis cloud untuk menjalankan dan mendistribusikan aplikasi web yang sudah dikembangkan. Platform ini mendukung pelaksanaan aplikasi Flask berbasis Python secara daring, sehingga pengguna dapat mengakses sistem optimasi rute melalui web browser kapan saja dan di mana saja.

D. Perancangan Konseptual

Perancangan konseptual ini bertindak sebagai dasar untuk mengembangkan sistem optimasi jalur, dengan perhatian utama pada penggunaan algoritma Floyd–Warshall guna menentukan jalur terpendek antara kota-kota di Jawa Timur secara komprehensif dan efektif. Menjelaskan cara kerja sistem secara teoritis, mencakup alur pemikiran, algoritme, dan organisasi data yang diterapkan.

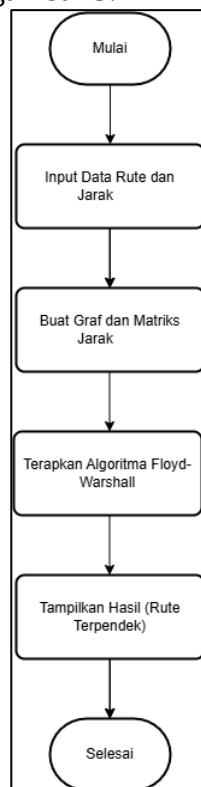
1. Algoritma Floyd–Warshall

Algoritma Floyd-warshall adalah algoritma klasik dalam dunia graf yang digunakan untuk mengidentifikasi jalur terpendek di antara setiap pasangan titik dalam sebuah graf dengan bobot, baik yang berarah maupun tidak. Metode ini beroperasi berdasarkan prinsip pemrograman dinamis, yaitu memperbaharui jarak antar titik secara bertahap sambil mempertimbangkan kemungkinan adanya titik perantara yang dapat menciptakan jalur yang lebih singkat. Prosesnya dilakukan secara berulang hingga semua kombinasi titik

dibandingkan, menghasilkan matriks jarak minimum yang menggambarkan jalur terpendek di seluruh jaringan.

2. Flowchart Sistem

Flowchart sistem digunakan untuk menggambarkan proses kerja logika dalam sistem penentuan rute secara jelas dan teratur. Grafik ini menjelaskan rangkaian langkah dari pengguna yang membuka aplikasi, memasukkan data rute antar kota, hingga sistem melakukan perhitungan menggunakan algoritma Floyd–Warshall untuk mendapatkan rute tercepat. Setelah selesai menghitung, sistem akan memperlihatkan hasil optimasi yang mencakup jalur tercepat, perkiraan waktu tempuh, dan gambaran rute di peta digital. Berikut adalah flowchart yang menggambarkan alur kerja sistem, dapat dilihat pada gambar 3.



Gambar 3. Alur Kerja Sistem

Gambar 3 merupakan Proses pengoptimalan rute dalam sistem ini dimulai saat pengguna menjalankan aplikasi atau mengunjungi halaman utama untuk pengoptimalan rute. Langkah pertama melibatkan memasukkan data rute dan jarak, di mana pengguna atau sistem memberikan informasi berupa pasangan

kota serta jarak yang memisahkan kota-kota tersebut yang akan menjadi dasar perhitungan. Data ini selanjutnya diubah menjadi graf berarah dengan bobot, di mana setiap simpul melambangkan kota dan bobot pada sisi menunjukkan jarak antara simpul-simpul tersebut. Kemudian, sistem menggunakan algoritma Floyd–Warshall sebagai inti dari perhitungan, yaitu dengan memperbarui nilai dalam matriks jarak untuk mencari rute terpendek antara semua pasangan kota ($i-j$) melalui beberapa simpul perantara (k). Setelah semua iterasi selesai, sistem akan menunjukkan hasil perhitungan untuk rute terpendek, yang mencakup urutan kota yang dilalui, total jarak, estimasi waktu perjalanan, tabel hasil, serta visualisasi jalur di peta digital Google Maps. Proses ini dianggap selesai setelah semua hasil pengoptimalan ditampilkan secara komprehensif kepada pengguna.

E. Perancangan Fisik

Perangkat keras meliputi elemen perangkat keras serta perangkat lunak yang diperlukan untuk menerapkan sistem.

1. Hardware

a. Komputer atau Laptop

Komputer atau laptop berfungsi sebagai alat utama dalam tahap pengembangan, penerapan, dan pengujian sistem. Alat ini digunakan untuk menulis kode, menjalankan server lokal, serta melakukan simulasi perhitungan algoritma Floyd–Warshall. Spesifikasi yang dianjurkan mencakup prosesor dengan banyak inti, RAM minimal 8 GB, dan ruang penyimpanan yang cukup untuk mengelola data jarak antar kota serta ketergantungan aplikasi.

b. Server

Server berfungsi sebagai tempat penyimpanan dan pelaksanaan sistem secara daring. Sistem yang telah selesai dibuat akan diunggah ke server supaya dapat diakses oleh pengguna lewat browser. Di samping itu, server juga menyimpan database rute, hasil perhitungan, serta file

konfigurasi sistem, sehingga memastikan ketersediaan, keamanan, dan kecepatan akses bagi pengguna.

c. Koneksi Internet

Koneksi internet digunakan untuk menghubungkan sistem dengan sumber data eksternal seperti Google Maps dan memungkinkan sistem beroperasi secara daring. Koneksi internet yang handal sangat diperlukan agar proses pengambilan data, validasi jarak antar kota, dan tampilan visualisasi peta dapat berjalan dengan cepat dan tepat. Selanjutnya, jaringan internet juga memiliki peran penting dalam sinkronisasi data antara sisi server dan antarmuka pengguna (client-side).

2. Software

a. Bahasa Pemrograman

Python (Backend) adalah bahasa pemrograman Python dipilih sebagai fondasi pengembangan sistem di sisi backend karena sintaksnya yang mudah, serbaguna, dan didukung oleh banyak pustaka untuk pengolahan data, perhitungan algoritma, serta koneksi dengan aplikasi web. Python juga digunakan untuk menerapkan algoritma Floyd–Warshall, yang berfungsi dalam menentukan rute terpendek antara kota-kota di Jawa Timur.

b. Framework

Flask (Frontend) adalah suatu framework berbasis Python yang digunakan untuk menciptakan antarmuka web interaktif (frontend). Flask mendukung arsitektur aplikasi yang sederhana, modular, dan mudah untuk dikembangkan, sehingga memfasilitasi integrasi antara logika backend dengan tampilan web. Dengan menggunakan Flask, hasil perhitungan algoritma dapat ditampilkan langsung kepada pengguna dalam format tabel dan visualisasi peta.

c. Visualisasi

Google Maps (Untuk Visualisasi Peta) digunakan untuk Google Maps

dimanfaatkan untuk menampilkan visualisasi rute secara geografis berdasar pada hasil perhitungan sistem. Integrasi ini memberikan kesempatan bagi pengguna untuk melihat jalur terpendek secara langsung di peta digital, beserta titik awal, titik akhir, dan perkiraan jarak yang harus ditempuh. Disamping meningkatkan pemahaman hasil, visualisasi ini juga menjadikan sistem lebih mudah dipahami dan digunakan.

d. Tools Pengembangan

Visual Studio Code (VS Code) dipakai sebagai lingkungan pengembangan terintegrasi (IDE) untuk menulis, mengedit, dan menjalankan kode. VSCode menawarkan berbagai ekstensi yang mempermudah proses debugging, pengelolaan proyek, serta integrasi dengan sistem kontrol versi seperti Git. Penggunaan VSCode membantu pengembang bekerja dengan lebih efisien dalam membangun dan menguji sistem optimasi rute berbasis web.

HASIL DAN PEMBAHASAN

Perancangan Antarmuka

1. Halaman Login

Dihalaman login ini admin harus memasukkan username dan password untuk mengakses ke halaman utama, jika sebelumnya admin sudah login tetapi lupa atau tidak logout, maka saat run kembali maka yang tampil adalah halaman utama, karena sistem membaca kita belum logout, jika ingin menampilkan halaman login maka kita harus mengetik logout pada server diatas, dapat dilihat pada gambar 4.

Gambar 4. Halaman Login

2. Halaman Utama

Pada halaman ini adalah halaman utama sekaligus jalan menuju pilihan yang ada, ada 3 pilihan yaitu daftar kota (berisi 38 daftar kota yang ada sesuai dengan dataset), matriks jarak (berisi jarak antar dua kota), dan yang terakhir berisi rute terpendek yang ingin kita akses, dapat dilihat pada gambar 5.

Gambar 5. Halaman Utama

3. Halaman Daftar Kota

Dihalaman ini adalah dari halaman utama kemudian kita mengklik daftar kota maka yang keluar adalah daftar kota lengkap dengan jarak dan durasi yang dilalui antar 2 kota, dapat dilihat pada gambar 6.

Daftar Jarak Antar Kota di Jawa Timur			
Kota Asal	Kota Tujuan	Jarak	Durasi
Kota Mojokerto	Kota Bitar	101.0 km	1 jam 41 menit
Kota Bitar	Kota Mojokerto	101.0 km	1 jam 41 menit
Kota Mojokerto	Kota Madiun	121.0 km	2 jam 0 menit
Kota Madiun	Kota Mojokerto	121.0 km	2 jam 0 menit
Kota Mojokerto	Kota Pasuruan	95.4 km	1 jam 35 menit
Kota Pasuruan	Kota Mojokerto	95.4 km	1 jam 35 menit
Kota Mojokerto	Kota Batu	60.7 km	1 jam 0 menit
Kota Batu	Kota Mojokerto	60.7 km	1 jam 0 menit
Kota Mojokerto	Kota Probolinggo	133.0 km	2 jam 13 menit
Kota Probolinggo	Kota Mojokerto	133.0 km	2 jam 13 menit

Gambar 6. Halaman Daftar Kota

4. Halaman Jarak Kota

Dihalaman ini berisi matriks jarak dari 38 kota dengan kota awal dan kota tujuan terlengkap dari 38 kota. Ini bisa kita

lihat saat akses halaman utama dan mengklik matriks jarak, maka yang keluar adalah halaman seperti diatas, dapat dilihat pada gambar 7.

Matriks Jarak Antar Kota di Jawa Timur												
Kota Asal \ Kota Tujuan	Kabupaten Bangkalan	Kabupaten Banyuwangi	Kabupaten Bitar	Kabupaten Blitar	Kabupaten Bojonegara	Kabupaten Brondong	Kabupaten Jember	Kabupaten Jombang	Kabupaten Kediri	Kabupaten Lamongan	Kabupaten Lumajang	Kabupaten Madiun
Kabupaten Bangkalan	0.0 km (0 jam 00 menit)	147.0 km (3 jam 27 menit)	70.0 km (1 jam 34 menit)	77.0 km (1 jam 37 menit)	201.0 km (4 jam 22 menit)	70.0 km (1 jam 34 menit)	77.0 km (1 jam 37 menit)	191.0 km (4 jam 17 menit)	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	180.0 km (3 jam 50 menit)	200.0 km (4 jam 20 menit)
Kabupaten Banyuwangi	147.0 km (3 jam 27 menit)	0.0 km (0 jam 00 menit)	122.0 km (2 jam 42 menit)	43.0 km (1 jam 10 menit)	220.0 km (4 jam 40 menit)	122.0 km (2 jam 42 menit)	191.0 km (4 jam 17 menit)	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	180.0 km (3 jam 50 menit)	200.0 km (4 jam 20 menit)	200.0 km (4 jam 20 menit)
Kabupaten Bitar	70.0 km (1 jam 34 menit)	122.0 km (2 jam 42 menit)	0.0 km (0 jam 00 menit)	137.0 km (3 jam 17 menit)	201.0 km (4 jam 22 menit)	137.0 km (3 jam 17 menit)	244.0 km (5 jam 11 menit)	191.0 km (4 jam 17 menit)	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)
Kabupaten Blitar	77.0 km (1 jam 37 menit)	43.0 km (1 jam 10 menit)	137.0 km (3 jam 17 menit)	0.0 km (0 jam 00 menit)	201.0 km (4 jam 22 menit)	201.0 km (4 jam 22 menit)	244.0 km (5 jam 11 menit)	191.0 km (4 jam 17 menit)	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)
Kabupaten Bojonegara	201.0 km (4 jam 22 menit)	220.0 km (4 jam 40 menit)	201.0 km (4 jam 22 menit)	201.0 km (4 jam 22 menit)	0.0 km (0 jam 00 menit)	201.0 km (4 jam 22 menit)	244.0 km (5 jam 11 menit)	191.0 km (4 jam 17 menit)	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)
Kabupaten Brondong	70.0 km (1 jam 34 menit)	122.0 km (2 jam 42 menit)	137.0 km (3 jam 17 menit)	201.0 km (4 jam 22 menit)	201.0 km (4 jam 22 menit)	0.0 km (0 jam 00 menit)	244.0 km (5 jam 11 menit)	191.0 km (4 jam 17 menit)	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)
Kabupaten Jember	77.0 km (1 jam 37 menit)	43.0 km (1 jam 10 menit)	137.0 km (3 jam 17 menit)	201.0 km (4 jam 22 menit)	201.0 km (4 jam 22 menit)	244.0 km (5 jam 11 menit)	0.0 km (0 jam 00 menit)	191.0 km (4 jam 17 menit)	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)
Kabupaten Jombang	191.0 km (4 jam 17 menit)	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	191.0 km (4 jam 17 menit)	191.0 km (4 jam 17 menit)	0.0 km (0 jam 00 menit)	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)
Kabupaten Kediri	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	155.0 km (3 jam 15 menit)	155.0 km (3 jam 15 menit)	150.0 km (3 jam 10 menit)	155.0 km (3 jam 15 menit)	155.0 km (3 jam 15 menit)	155.0 km (3 jam 15 menit)	0.0 km (0 jam 00 menit)	150.0 km (3 jam 10 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)
Kabupaten Lamongan	150.0 km (3 jam 10 menit)	180.0 km (3 jam 50 menit)	150.0 km (3 jam 10 menit)	150.0 km (3 jam 10 menit)	150.0 km (3 jam 10 menit)	150.0 km (3 jam 10 menit)	150.0 km (3 jam 10 menit)	150.0 km (3 jam 10 menit)	150.0 km (3 jam 10 menit)	0.0 km (0 jam 00 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)
Kabupaten Lumajang	180.0 km (3 jam 50 menit)	200.0 km (4 jam 20 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	0.0 km (0 jam 00 menit)	180.0 km (3 jam 50 menit)
Kabupaten Madiun	200.0 km (4 jam 20 menit)	200.0 km (4 jam 20 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	180.0 km (3 jam 50 menit)	0.0 km (0 jam 00 menit)

Gambar 7. Halaman Jarak Kota

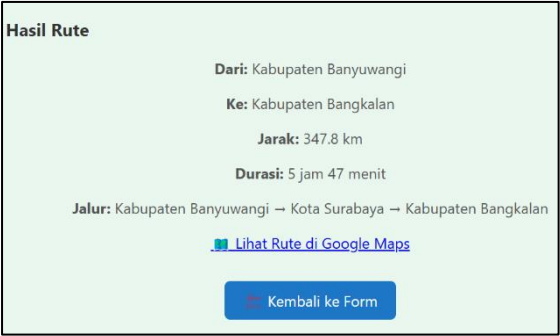
5. Halaman Pencarian Rute

Dihalaman ini berisi pilihan kota awal dan kota tujuan, jumlah kota yang ada di dataset berisi 38 kota, maka yang tampil pada pilihan kota awal dan tujuan adalah pilihan 38 kota, jika di klik cari maka yang tampil adalah lintasan terpendeknya, untuk lebih lengkapnya bisa dilihat pada tampilan Maps dibawah. Halaman lintasan terpendek ini muncul setelah kita mengklik cari rute terpendek pada halaman utama, dapat dilihat pada gambar 8.

Gambar 8. Halaman Pencarian Rute

6. Halaman Maps

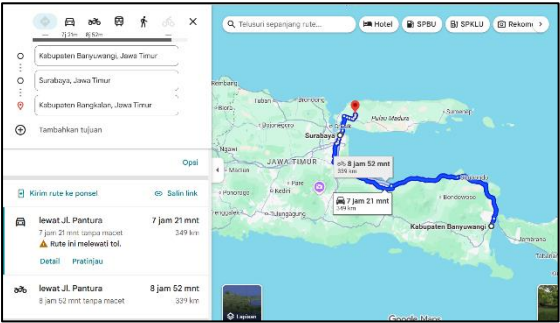
Dihalaman ini menampilkan hasil rute berupa dari kota awal, ke kota tujuan, jarak kedua kota, durasi yang ditempuh, dan jalur yang dilalui berdasarkan perhitungan Floyd-warshall yang mana menentukan rute terpendek dari semua kemungkinan jalur yang ada. Kemudian ada visualisasi Maps jika kita mengklik lihat rute di Google Maps, dapat dilihat pada gambar 9.



Gambar 9. Halaman Maps

7. Halaman Maps Web

Halaman ini berisi rute Maps yang sama saat kita mengklik cari, jarak, durasi dan jalur yang dilalui sama seperti yang ada di dataset, dan merupakan rute tercepat berdasarkan perhitungan floyd-warshall, dapat dilihat pada gambar 10.



Gambar 10. Halaman Maps Web

Hasil Pengujian

Hasil pengujian pada Implementasi Algoritma Floyd-Warshall untuk Optimasi Rute Terpendek Antar Kota Jawa Timur dapat dilihat pada tabel 2.

Tabel 2. Hasil Pengujian Sistem.

Skenario Pengujian	Input	Hasil yang Diharapkan	Kondisi
Pencarian rute dari Kota A ke kota B	Mojokerto ke Malang	Menampilkan rute terpendek + jarak + estimasi durasi	Normal
Kota tidak tersedia dalam data	Banyuwangi ke Jakarta	Tampilkan error kota tidak ditemukan	Normal (handling)
Menampilkan tabel matriks jarak	-	Tabel lengkap jarak + durasi antar semua kota	Normal
Akses fitur tanpa login	Langsung ke /api/shortest-path	Redirect ke halaman login	Normal

Klik tombol lihat di Google Maps setelah hasil	Surabaya ke Batu	Arahkan ke Google Maps dengan lintasan kota	Normal
--	------------------	---	--------

Tabel 2 memperlihatkan hasil dari serangkaian pengujian sistem yang dilakukan untuk memastikan aplikasi optimasi rute terpendek yang menggunakan algoritma Floyd-Warshall berfungsi dengan benar sesuai dengan skenario yang telah disusun. Pada percobaan pertama, saat pengguna mencari rute dari Kota Mojokerto menuju Kota Malang, sistem berhasil menampilkan rute terpendek lengkap dengan jarak tempuh dan perkiraan waktu perjalanan, yang mengindikasikan bahwa fungsi utama dari algoritma beroperasi secara baik. Kemudian, pada percobaan kedua, ketika pengguna memasukkan kota yang tidak terdaftar dalam data, seperti Banyuwangi ke Jakarta, sistem menunjukkan pesan “error kota tidak ditemukan”, yang menandakan bahwa sistem error handling berfungsi dengan efektif untuk mencegah kesalahan masukan. Pada pengujian selanjutnya, sistem diuji untuk menampilkan matriks jarak antara kota-kota, di mana hasilnya menunjukkan tabel yang lengkap berisi jarak serta estimasi waktu perjalanan di antara semua kota yang ada dalam basis data, dan keseluruhan proses berjalan tanpa masalah. Untuk pengujian keamanan akses, ketika pengguna mencoba menggunakan fitur tanpa melakukan login melalui endpoint /api/shortest-path, sistem secara otomatis melakukan redirect ke halaman login, memastikan bahwa proses autentikasi pengguna telah dilaksanakan dengan baik. Akhirnya, pada skenario klik tombol Lihat di Google Maps setelah hasil perhitungan ditampilkan, sistem mampu mengalihkan pengguna langsung ke tampilan jalur di Google Maps, sesuai dengan rute yang telah dihitung oleh algoritma. Secara keseluruhan, hasil pengujian menunjukkan bahwa semua fitur utama sistem beroperasi stabil, sesuai dengan ekspektasi, dan siap digunakan secara optimal untuk mendukung optimasi rute antar kota di Jawa Timur.

SIMPULAN

Berdasarkan penelitian yang telah dilakukan dalam Implementasi Algoritma Floyd-Warshall untuk Optimasi Rute Terpendek Antar Kota Jawa Timur dapat diambil beberapa kesimpulan sebagai berikut. Berdasarkan pengembangan dan penerapan sistem pengoptimalan rute menggunakan algoritma Floyd-Warshall, bisa disimpulkan bahwa sistem ini telah efektif dalam menentukan jalur terpendek antar kota. Sistem ini dapat menerima masukan kota asal dan tujuan, menghitung jarak terdekat, serta menampilkan rute terbaik beserta estimasi waktu perjalanan. Uji coba awal menunjukkan bahwa sistem ini mampu memberikan hasil perhitungan yang rasional dan sesuai dengan teorema graf berbobot, serta mampu menangani masukan tidak valid dengan baik. Desain antarmuka pengguna yang responsif juga mempermudah interaksi pengguna dengan sistem, sementara integrasi dengan Google Maps memberikan nilai tambah dalam

Namun, sistem ini masih memiliki beberapa keterbatasan, seperti penyimpanan data statis yang berbasis CSV, belum adanya pengelolaan pengguna yang dinamis, serta integrasi yang belum sepenuhnya dilakukan dengan data operasional rute terpendek. Oleh karena itu, pengembangan di masa mendatang akan difokuskan pada peningkatan fitur, integrasi database, keamanan login, dan pengujian validasi langsung dengan data nyata. Oleh sebab itu, sistem ini memiliki potensi yang signifikan untuk diterapkan secara nyata dalam meningkatkan efisiensi rute perjalanan dan mendukung pengambilan keputusan operasional di lapangan.

DAFTAR PUSTAKA

Inayah, A. M., Cintya Resti, N., & Kediri, I. (2023). Analisa Perbandingan Algoritma Floyd-Warshall Dan Algoritma Dijkstra untuk Penentuan Rute Terdekat. *Jurnal Ilmiah Matematika Realistik (JI-MR)*, 4(2),

146–155.

- Khairi, I., Yudaningsy, E., & Dachlan, H. S. (2019). Optimasi Pencarian Jalur Lalu Lintas Antar Kota di Jawa Timur dengan Algoritma Hybrid Fuzzy-Floyd Warshall. *Eeccis*, 7(2), 165–170.
- Ningrum, E. R., Sanwidi, A., Akbarita, R., & Qomaruddin, M. N. H. (2023). Optimasi Rute Pendistribusian Gas Elpiji Menggunakan Algoritma Floyd Warshall Dan Algoritma Greedy. *JURNAL ILMIAH MATEMATIKA DAN TERAPAN*, 20(1), 1–14. <https://doi.org/10.22487/2540766x.2023.v20.i1.15568>
- 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.
- Pipit Muliyah, Dyah Aminatun, Sukma Septian Nasution, Tommy Hastomo, Setiana Sri Wahyuni Sitepu, T. (2020). 濟無No Title No Title No Title. *Journal GEEJ*, 7(2), 9–31.
- Rofiq, M. A. (2022). Penentuan Rute Terpendek Objek Wisata Malang Raya Menggunakan Algoritma Floyd-Warshall.
- Sari, F. N., Juhari, J., & Herawati, E. (2023). Penerapan Algoritma Floyd-Warshall pada Jalur Evakuasi Korban Kecelakaan di Boyolali. *Jurnal Riset Mahasiswa Matematika*, 2(5), 208–213. <https://doi.org/10.18860/jrmm.v2i5.2006>
- Sitinjak, F. (2022). Perbandingan Algoritma Dijkstra dan Floydwarshall untuk Mencari Jalur Terpendek dengan Contoh Kasus Mencari Rumah Sakit Terdekat di Kota Medan. *Login: Jurnal Teknologi Komputer*, 16(1), 9–22.
- Vulandari, R. T., Hasbi, M., & Tristanto, B. (2021). Penerapan Algoritma Floyd Warshall pada Sistem Informasi Puskesmas Kabupaten Karanganyar.

*Square : Journal of Mathematics and
Mathematics Education*, 3(1), 20–29.
[https://doi.org/10.21580/square.2021
.3.1.7716](https://doi.org/10.21580/square.2021.3.1.7716)