

Pemanfaatan Algoritma Floyd Warshall dalam Menentukan Jalur Terpendek Bencana Banjir

Yetman Erwadi¹, Sri Handayani^{1*}, Surya Ade Saputera¹, Sandhy Fernandez²

¹Fakultas Teknik, Sistem Informasi, Universitas Muhammadiyah Bengkulu, Bengkulu, Indonesia

²Fakultas Informatika, Sistem Informasi, Institut Teknologi Telkom Purwokerto, Purwokerto, Indonesia

Email: ¹yetman@umb.ac.id, ^{2*}yani@umb.ac.id, ³suryaadesaputera@umb.ac.id, ⁴sandhy@ittelkom-pwt.ac.id

Email Penulis Korespondensi: yani@umb.ac.id

Submitted: 03/07/2024; Accepted: 08/08/2024; Published: 09/08/2024

Abstrak—Provinsi Bengkulu mengalami bencana banjir dan longsor yang menyebabkan korban jiwa dan kerugian harta benda. Kejadian ini terutama terjadi di kawasan Bengkulu Tengah dan Kota Bengkulu, yang disebabkan oleh kerusakan Daerah Aliran Sungai (DAS) Bengkulu. Saat musim hujan, air sungai ini meluap. Dalam hal ini masyarakat juga kebingungan ketika evakuasi terkait dengan bencana banjir, masyarakat bingung dalam menentukan jalur mana yang terdekat untuk jalur evakuasi karena pada lokasi bencana banjir banyak jalur yang bisa dilewati. Dari permasalahan tujuan penelitian ini adalah membuat aplikasi yang dapat menentukan jalur mana yang terdekat agar bisa evakuasi dengan cepat. Dari hasil analisis terhadap aplikasi ini, jelas terlihat bahwa implementasi algoritma Floyd-Warshall dalam aplikasi web ini memiliki tujuan yang sangat penting, yaitu membantu masyarakat dalam menemukan jalur terpendek menuju titik evakuasi saat terjadi banjir. Algoritma Floyd-Warshall dikenal efektif dalam mencari jalur terpendek dalam graf berbobot, seperti yang diimplementasikan dalam konteks aplikasi ini untuk menghitung jarak terpendek antara setiap pasangan simpul atau titik dalam suatu area terdefinisi. Algoritma *Floyd-Warshall* hanya dapat menghitung jalur terpendek di dalam area yang sudah terdefinisi dengan baik dalam hal simpul dan sisi-sisinya, untuk hasil uji dari jarak yang didapatkan pada uji 1 sejauh 0.76 KM, pada uji 2 sejauh 0.40 KM dan pada uji 3 sejauh 0.41 KM.

Kata Kunci: Bencana Banjir; Floyd Warshall; Jalur Terpendek; Website

Abstract—Bengkulu Province experienced floods and landslides that caused loss of life and property. These events mainly occur in the Central Bengkulu region and Bengkulu City, caused by damage to the Bengkulu River Watershed (DAS). During the rainy season, the river overflows. In this case the community is also confused when evacuating related to flood disasters, the community is confused in determining which route is the closest for the evacuation route because at the location of the flood disaster there are many routes that can be passed. From the problem the purpose of this research is to create an application that can determine which path is the closest in order to evacuate quickly. From the results of the analysis of this application, it is clear that the implementation of the Floyd-Warshall algorithm in this web application has a very important purpose, namely to assist the community in finding the shortest path to the evacuation point during a flood. The Floyd-Warshall algorithm is known to be effective in finding shortest paths in weighted graphs, as implemented in the context of this application to calculate the shortest distance between every pair of vertices or points within a defined area. The Floyd-Warshall algorithm can only compute shortest paths within areas that are well-defined in terms of vertices and edges, for the test results of the distance obtained in test 1 as far as 0.76 KM, in test 2 as far as 0.40 KM and in test 3 as far as 0.41 KM.

Keywords: Flood Disaster; Floyd Warshall; Shortest Path; Website

1. PENDAHULUAN

Bencana adalah peristiwa yang menyebabkan kerugian besar bagi masyarakat, bersifat merusak, dan memerlukan waktu lama untuk pemulihannya. Definisi ini dijelaskan lebih lanjut dalam UU Nomor 24 Tahun 2007 tentang Penanggulangan Bencana, di mana bencana didefinisikan sebagai rangkaian kejadian yang menimbulkan ancaman langsung terhadap kehidupan masyarakat, yang disebabkan oleh faktor alam maupun non-alam. Dampak langsung dari bencana meliputi kerusakan lingkungan, kerugian harta benda, dampak psikologis, dan korban jiwa [1].

Pada tanggal 27 April 2019, Provinsi Bengkulu mengalami bencana banjir dan longsor yang menyebabkan korban jiwa dan kerugian harta benda. Kejadian ini terutama terjadi di kawasan Bengkulu Tengah dan Kota Bengkulu, yang disebabkan oleh kerusakan Daerah Aliran Sungai (DAS) Bengkulu. Saat musim hujan, air sungai ini meluap. Hulu DAS ini berada di Kecamatan Taba Penanjung, Bengkulu Tengah, sementara hilirnya terletak di Teluk Segera, Kota Bengkulu. Hulu sungai ini berupa perbukitan bergelombang dengan punggung bukit yang curam hingga sangat curam, sedangkan hilirnya didominasi oleh wilayah datar. Sungai utama di DAS ini adalah Sungai Air Bengkulu.

Penurunan kualitas DAS disebabkan oleh beberapa faktor, termasuk buruknya kinerja pelestarian pengelolaan DAS, yang dapat dilihat dari lemahnya dukungan faktor biogeofisik kawasan, pelestarian lingkungan (penggunaan lahan dan tata air), serta pelestarian lingkungan sosial, ekonomi, dan kelembagaan. Penyebab banjir di wilayah Provinsi Bengkulu berkaitan dengan tingginya intensitas curah hujan. Dalam hal ini masyarakat juga kebingungan ketika evakuasi terkait dengan bencana banjir, masyarakat bingung dalam menentukan jalur mana yang terdekat untuk jalur evakuasi karena pada lokasi bencana banjir banyak jalur yang bisa dilewati. Dari permasalahan tujuan penelitian ini adalah membuat aplikasi yang dapat menentukan jalur mana yang terdekat agar bisa evakuasi dengan cepat.

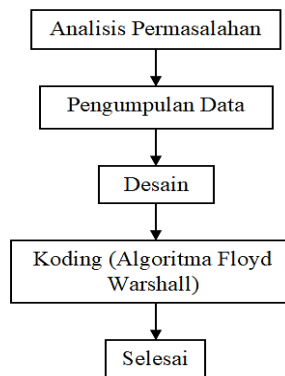
Algoritma Floyd Warshall adalah salah satu bentuk pemrograman dinamis, yaitu sebuah metode yang menyelesaikan masalah dengan menganggap solusi yang diperoleh sebagai serangkaian keputusan yang saling terkait. Solusi-solusi tersebut dibangun dari solusi pada tahap sebelumnya dan mungkin terdapat lebih dari satu solusi [2], [3], [4]. Alasan utama penggunaan Algoritma Floyd Warshall dalam tugas akhir ini adalah karena algoritma tersebut melihat solusi akhir sebagai serangkaian keputusan yang saling terkait. Solusi-solusi tersebut dibangun dari solusi pada tahap sebelumnya. Dengan menggabungkan aplikasi berbasis Android dan Algoritma Floyd-Warshall, diharapkan pengguna dapat lebih mudah mencari lokasi fasilitas umum terdekat dan mendapatkan keputusan yang lebih akurat [5], [6]. Dari permasalahan dan metode yang telah ditentukan maka tujuan dari penelitian ini adalah membuat aplikasi yang dapat menentukan jalur mana yang terdekat agar bisa evakuasi dengan jalur yang sudah terpilih.

Dari penelitian sebelumnya Algoritma Floyd-Warshall sebuah metode yang menyelesaikan masalah dengan memandang solusi sebagai keputusan yang saling terkait. Solusi ini dibangun dari tahap sebelumnya dan mungkin ada lebih dari satu solusi. Algoritma Floyd-Warshall menghitung jarak terdekat antara semua simpul dengan mengkalkulasi jarak dari satu simpul sumber ke simpul tujuan melalui berbagai jalur. Algoritma ini berfungsi untuk menemukan jalur terdekat yang paling minimum dari setiap pasangan simpul, dan menggunakan matriks $n \times n$ sebagai input, di mana n adalah jumlah simpul [7]. Penelitian selanjutnya yaitu dengan membahas pemilihan jalur terpendek pada masjid di Semarang dimana mendapatkan hasilnya adalah algoritma Floyd Warshall dapat memberikan rute jalan terpendek ke masjid, dan lebih mudah diakses sebesar 21,67% dibandingkan dengan navigasi Google Maps [8], selanjutnya ada penelitian yang membahas tentang pemilihan jalur terpendek dengan menggunakan algoritma Floyd Warshall untuk menentukan jarak terpendek ke rumah sakit dimana hasilnya didapatkan bahwa untuk Rumah Sakit Madani jarak paling dekat yaitu 0.88 km yang melewati simpul 2,3, dan 4 [9]. Dari penelitian sebelumnya yang sudah dilakukan pembeda dari penelitian ini dengan penelitian sebelumnya terletak pada titik simpul yang ditentukan serta lokasi koordinat yang berbeda.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Pada tahapan ini akan dilakukan beberapa tahapan penelitian, adapun langkah-langkah yang akan dilakukan kesetiap tahapannya. Berikut tahapan dalam penelitian ini.



Gambar 1. Alur Penelitian

Adapun penjelasan dari gambar 1 sebagai berikut. Analisis ini dilakukan dengan mendengarkan dan mengumpulkan keluhan dari masyarakat yang terkena dampak bencana banjir. Keluhan-keluhan tersebut mencakup berbagai aspek, seperti daerah yang paling sering terendam banjir, durasi banjir, serta dampak yang ditimbulkan terhadap kehidupan sehari-hari. Pada tahap pengumpulan data, data yang dikumpulkan meliputi peta kota Bengkulu, yang digunakan untuk memetakan lokasi-lokasi yang rawan banjir. Pemetaan ini mencakup identifikasi titik-titik kritis dan daerah-daerah yang sering mengalami genangan air. Selain itu, dalam tahap ini juga dilakukan penentuan node dan graph yang akan digunakan dalam pemodelan. Node merepresentasikan titik-titik penting dalam jaringan jalan, sedangkan graph menggambarkan hubungan antar node tersebut. Selanjutnya tahapan desain dilakukan desain aplikasi menggunakan UML (*Unified Modeling Language*). UML digunakan untuk memodelkan struktur dan perilaku sistem yang akan dikembangkan. Selanjutnya pada tahapan koding, pada tahap ini, algoritma Floyd Warshall diterapkan pada aplikasi untuk menentukan jalur terpendek saat terjadi bencana banjir, setelah semua tahap pengembangan selesai, dilakukan pengujian untuk memastikan bahwa aplikasi berjalan sesuai dengan yang diharapkan dan dapat memberikan solusi yang efektif terhadap permasalahan banjir.

Algoritma *Floyd Warshall* dalam penelitian ini yaitu pertama data lintasan yang digunakan terdiri dari data primer yang diperoleh langsung melalui observasi dengan Software Google Maps, serta data sekunder yang diperoleh dari penelitian sebelumnya [10], [11], [12]. Tahapan pengolahan data dimulai dengan memodelkan

data yang diperoleh ke dalam bentuk graf penelitian. Selanjutnya, bobot indikator yang digunakan, yaitu jarak, waktu, dan kemacetan, dianalisis dan dirasionalisasi ke dalam rentang yang sama [13], [14], [15]. Ketiga bobot indikator ini kemudian diubah menjadi satu bobot alternatif dengan pendekatan Multiple Attribute Decision Making. Bobot yang diperoleh selanjutnya digunakan sebagai bobot lintasan dalam proses iterasi algoritma Floyd-Warshall. Hasil akhir iterasi algoritma Floyd-Warshall kemudian ditampilkan kembali dalam bentuk graf [16], [17]. Rute terpendek diperoleh dengan menganalisis bobot terkecil antar setiap *edge* yang menghubungkan masing-masing titik dengan ketentuan tidak terjadi loop, di mana setiap lintasan hanya dapat dilalui sekali [18], [19], [20].

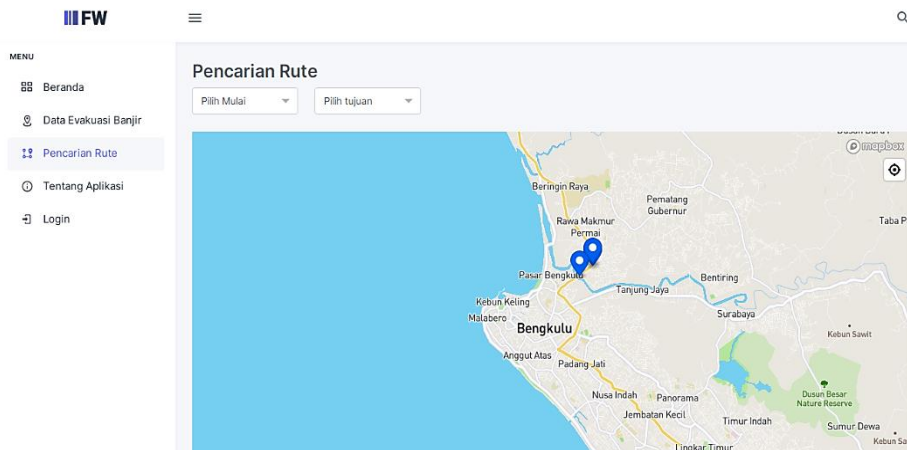
3. HASIL DAN PEMBAHASAN

Hasil uji coba terhadap aplikasi pencarian jalur evakuasi banjir terdekat menggunakan algoritma *Floyd Warshall* yang telah dibuat, uji coba ini bertujuan untuk mengetahui apakah aplikasi yang telah di buat telah sesuai dan berjalan dengan semestinya sesuai. Pada pembahasan ini juga akan di bahas mengenai fitur aplikasi dan *interface* yang telah terdapat di dalam aplikasi pencarian jalur evakuasi banjir terdekat menggunakan algoritma *Floyd Warshall*.

3.1 Uji Coba Aplikasi

Pada saat menggunakan aplikasi ini pastikan koneksi internet di laptop aktif dan mempunyai kualitas jaringan yang baik karena pada bagian map menggunakan API dari mapbox yang akan menampilkan data dari map yang akan ditampilkan, adapun pada gambar 2 tampilan ketika menu awal ditampilkan dengan akses internet menyala.

1. Ketikkan <http://localhost/floydwarshall/> pada URL di browser.



Gambar 2. Pengujian Tampilan pada Menu Awal

3.2 Uji Coba Penggunaan Algoritma *Floyd Warshall* Pada Peta

Penulis melakukan pengujian terhadap perhitungan algoritma *Floyd Warshall* pada peta, apakah Algoritma *Floyd Warshall* dapat menyelesaikan dan menampilkannya dalam peta. Pada pengujian ini melakukan 3 tes dengan lokasi yang berbeda-beda:

	A	B	C	D	E	F	G	H	I	J	K	L	M
A	0	0.07	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
B	0.07	0	0.08	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
C	∞	0.08	0	0.01	∞	∞	∞	∞	∞	∞	∞	∞	∞
D	∞	∞	0.01	0	0.05	∞	∞	∞	∞	∞	∞	∞	∞
E	∞	∞	∞	0.05	0	0.06	∞	∞	∞	∞	0.03	∞	∞
F	∞	∞	∞	∞	0.06	0	0.01	∞	∞	∞	∞	∞	∞
G	∞	∞	∞	∞	∞	0.01	0	0.04	∞	∞	∞	∞	∞
H	∞	∞	∞	∞	∞	∞	0.04	0	0.04	∞	∞	∞	∞
I	∞	∞	∞	∞	∞	∞	∞	0.04	0	0.03	∞	0.26	∞
J	∞	∞	∞	∞	∞	∞	∞	∞	0.03	0	0.11	∞	∞
K	∞	∞	∞	∞	0.03	∞	∞	∞	∞	0.11	0	∞	∞
L	∞	∞	∞	∞	∞	∞	∞	∞	0.26	∞	∞	0	0.14
M	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.14	0

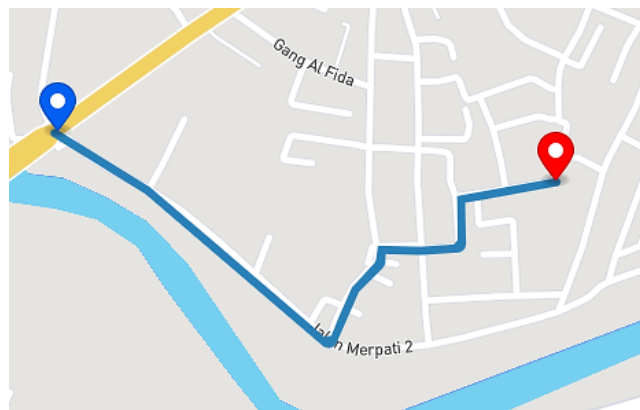
Gambar 3. Matrik Pengukuran 1

Pada gambar 3 menjelaskan bahawa dari semua pengukuran secara manual yaitu A sampai M dibuat sebagai inisial dari titik tujuan, adapun hasil dari angka pada matrik didapatkan dengan hasil jarak antara titik-titik tersebut, setelah diketahui semua nilai dari titik jalur A sampai M dibuatlah matrik untuk mengetahui bahwa tingkat keterhubungan antara jarak yang bisa dilalui untuk mencapai titik evakuasi banjir yang telah ditentukan. Perhitungan setelah dibuat matrik ini maka akan dijumlahkan titik dan jalur sesuai dengan peta yang akan dilalui.

Tabel 1. Tabel Uji Coba 1

PENGUJIAN 1	
Koordinat Awal : 3.780217967781496, 102.27757022371156	
Koordinat Tujuan : Jl. Kalimantan : 3.7795912930256463, 102.27109090646019	
Jalur 1 : 1 ->6 ->4 ->5 ->10 ->11 ->31 ->32 ->14 ->15	0.76 Km
Jalur 2 : 1 ->6 ->4 ->5 ->10 ->12 ->13 ->14 ->15	0.78 Km
Jalur 1 Terpilih : 1 ->6 ->4 ->5 ->10 ->11 ->31 ->32 ->14 ->15 = 0.76 Km	

Pengujian 1 pada tabel 1, pada menu pencarian jalur pada rute menuju titik evakuasi Jl.Kalimantan penulis lihat hanya pada tampilan peta terdapat dua jalur yang mengarah ke lokasi, bila kita lihat di tabel, jalur pertama adalah jalur 1 yang mempunyai panjang 0.76 Km, dan jalur kedua adalah mempunyai panjang 0.78 Km, Algoritma *Floyd Warshall* menyelesaikan pada jalur 1 dengan rute 1 ->6 ->4 ->5 ->10 ->11 ->31 ->32 ->14 ->15 = 0.76 Km. Pada gambar 4 akan ditampilkan secara jalur yang disusun berdasarkan *node* yang telah terpilih.



Gambar 4. Jalur Terpilih Titik Evakuasi Jl.Kalimantan

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	0	0.07	∞	∞	∞	∞	∞	∞	∞	0.02	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
B	0.07	0	0.08	∞	∞	∞	∞	∞	∞	0.02	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
C	∞	0.08	0	0.02	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
D	∞	∞	0.02	0	0.05	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
E	∞	∞	∞	0.05	0	0.06	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
F	∞	∞	∞	∞	0.06	0	0.03	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
G	∞	∞	∞	∞	∞	0.03	0	0.03	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
H	∞	∞	∞	∞	∞	∞	0.03	0	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.03	∞	∞	∞	∞	∞	∞	∞	0.10
I	0.02	∞	∞	∞	∞	∞	∞	∞	0	0.09	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.01	∞
J	∞	0.02	∞	∞	∞	∞	∞	∞	0.09	0	0.02	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
K	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.02	0	0.02	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.07
L	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.02	0	0.02	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
M	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.02	0	0.06	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
N	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.06	0	0.01	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
O	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.01	0	0.01	∞	∞	∞	∞	∞	∞	∞	∞	∞
P	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.01	0	0.03	∞	∞	∞	∞	∞	∞	∞	∞
Q	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.03	0	0.01	∞	∞	∞	∞	∞	∞	∞
R	∞	∞	∞	∞	∞	∞	∞	∞	0.03	∞	∞	∞	∞	∞	∞	∞	∞	0.01	0	0.04	∞	∞	∞	∞	∞	∞
S	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.04	0	0.08	∞	∞	∞	∞	∞
T	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.08	0	0.01	∞	∞	∞	∞
U	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.01	0	0.04	∞	∞	∞
V	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.04	0	0.08	∞	∞
W	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.08	0	0.01	∞
X	∞	∞	∞	∞	∞	∞	∞	∞	0.01	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.01	0	0.05
Y	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.07	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0.05	0
Z	∞	∞	∞	∞	∞	∞	∞	0.10	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0

Gambar 5. Matriks Pengukuran 2

Pada gambar 5 sama seperti gambar 3 akan tetapi perhitungan jarak titik yang dihitung berbeda jalur yang disambungkan. Gambar 10 menjelaskan bahawa dari semua pengukuran secara manual yaitu A sampai z dibuat sebagai inisial dari titik tujuan, adapun hasil dari angka pada matrik didapatkan dengan hasil jarak antara titik-titik tersebut, setelah diketahui semua nilai dari titik jalur A sampai M dibuatlah matrik untuk mengetahui bahwa tingkat keterhubungan antara jarak yang bisa dilalui untuk mencapai titik evakuasi banjir yang telah ditentukan. Perhitungan setelah dibuat matrik ini maka akan dijumlahkan titik dan jalur sesuai dengan peta yang akan dilalui.

Tabel 2. Tabel Uji Coba 2

PENGUJIAN 2	
Koordinat Awal : 3.780217967781496, 102.27757022371156	
Koordinat Tujuan : Merpati 4 : -2.5757513, 101.117507	
Jalur 1 : 1 ->6 ->4 ->18 ->16 ->17 ->24 ->26	0.44 Km
Jalur 2 : 1 ->3 ->8 ->2 ->33 ->34 ->35 ->36 ->28 ->26	0.42 Km
Jalur 3 : 1 ->6 ->7 ->19 ->20 ->21 ->22 ->23 ->25 ->27 ->28 ->26	0.40 Km
Jalur Terpilih : 1 ->6 ->7 ->19 ->20 ->21 ->22 ->23 ->25 ->27 ->28 ->26 = 0.40 Km	

Pengujian 2 pada tabel 2, pada menu pencarian jalur, pada rute menuju titik evakuasi banjir bila penulis lihat hanya pada tampilan peta terdapat tiga jalur yang mengarah ke lokasi, bila kita lihat di tabel, jalur pertama adalah jalur 1 yang mempunyai panjang 0.44 Km, jalur kedua adalah mempunyai panjang 0.42 Km dan jalur yang ketiga mempunyai jarak panjang 0.40 Km. Algoritma *Floyd Warshall* menyelesaikan pada jalur 3 berdasarkan penyusunan matrik dan pengukuran panjang rute dengan hasil rute terpilih 1 ->6 ->7 ->19 ->20 ->21 ->22 ->23 ->25 ->27 ->28 ->26 = 0.40 Km. Pada gambar 6 akan ditampilkan secara jalur yang disusun berdasarkan *node* yang telah terpilih.

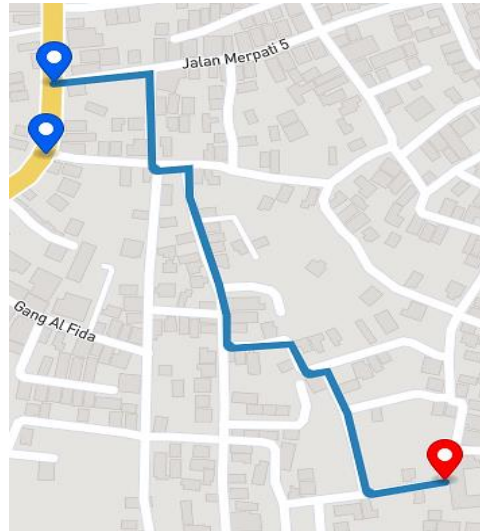


Gambar 6. Jalur Terpilih Titik Evakuasi Merpati 4

Tabel 3. Tabel Uji Coba 3

PENGUJIAN 3	
Koordinat Awal : 3.780217967781496, 102.27757022371156	
Koordinat Tujuan : Merpati 5 : -2.5753894, 101.116873	
Jalur 1 : 1 -> 3 -> 8 -> 2 -> 33 -> 34 -> 35 -> 36 -> 28 -> 26 -> 29	0.43 Km
Jalur 2 : 1 -> 6 -> 4 -> 18 -> 16 -> 17 -> 24 -> 26 -> 29	0.45 Km
Jalur 3 : 1 -> 6 -> 7 -> 19 -> 20 -> 21 -> 22 -> 23 -> 25 -> 27 -> 28 -> 26 -> 29	0.41 Km
Jalur Terpilih : 1 ->6 ->7 ->19 ->20 ->21 ->22 ->23 ->25 ->27 ->28 ->26 ->29 = 0.41 Km	

Pengujian 3 pada tabel 3, pada menu pencarian jalur, pada rute menuju titik evakuasi banjir yaitu merpati 5, bila penulis lihat hanya pada tampilan peta terdapat tiga jalur yang mengarah ke lokasi, bila kita lihat di tabel, jalur pertama adalah jalur 1 yang mempunyai panjang 0.43 K, jalur kedua adalah mempunyai panjang rute yaitu 0.45 Km dan jalur yang ketiga yaitu dengan panjang 0.41 Km, Algoritma *Floyd Warshall* menyelesaikan pada jalur 3 sesuai dengan jalur matrik dan perhitungan melalui rute 1 -> 6 -> 7 -> 19 -> 20 -> 21 -> 22 -> 23 -> 25 -> 27 -> 28 -> 26 -> 29 = 0.41 Km. Pada gambar 7 akan ditampilkan secara jalur yang disusun berdasarkan *node* yang telah terpilih.



Gambar 7. Jalur Terpilih Titik Evakuasi Merpati 5

3.3 Implementasi Sistem

Dalam tahap ini aplikasi yang telah di desain mulai diterapkan dengan membangun komponen-komponen. Aplikasi pencarian lokasi evakuasi banjir terdekat menggunakan algoritma *Floyd Warshall* ini menggunakan database MySQL sebagai basisdata dan menggunakan bahasa PHP untuk pemrograman aplikasinya,

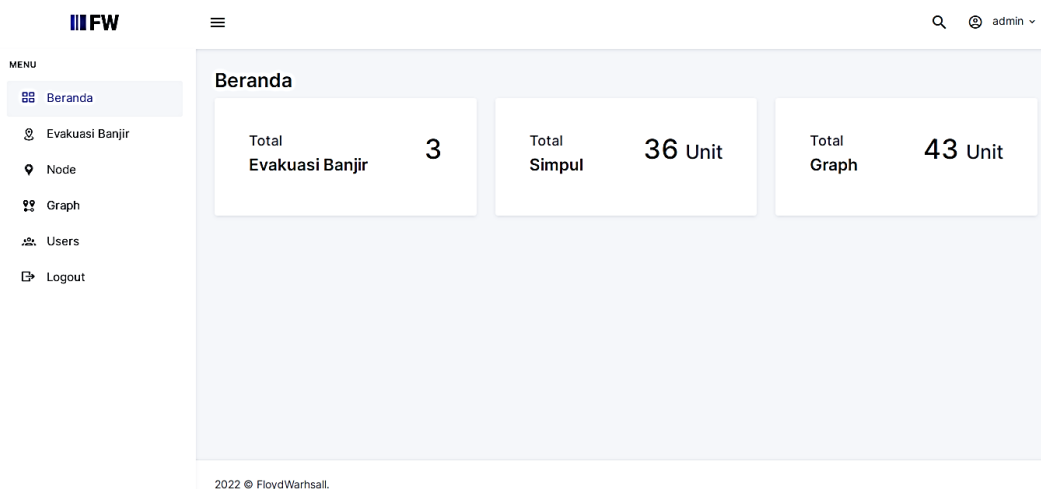
3.4 Implementasi Interface

Dalam aplikasi pencarian lokasi evakuasi banjir terdekat menggunakan algoritma Floyd Warshall ini mengimplementasikan beberapa interface, diantaranya:

1. Halaman aplikasi dengan menampilkan beranda,
2. Halaman aplikasi dengan menampilkan data evakuasi,
3. Halaman aplikasi dengan menampilkan pencarian rute,
4. Halaman aplikasi dengan menampilkan tentang aplikasi,
5. Halaman aplikasi dengan menampilkan rute algoritma Floyd Warshall menuju lokasi evakuasi banjir.

3.4.1 Halaman Beranda

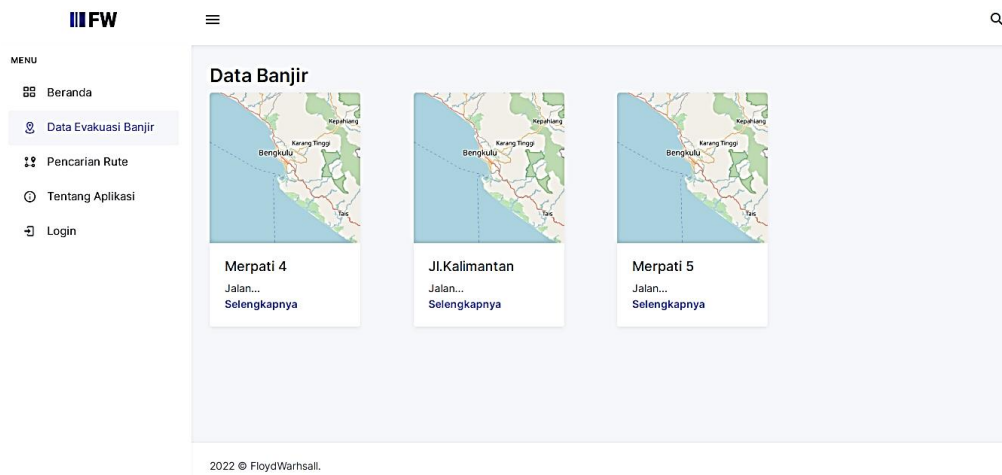
Gambar 8 merupakan halaman yang ditampilkan saat pertama kali aplikasi di buka. Pada halaman ini berisi data tampilan evakuasi banjir, simpul dan graph. Adapun dari informasi tampilan tersebut kita mengetahui jumlah dari data tersebut.



Gambar 8. Halaman Beranda

3.4.2 Halaman Menu Data Evakuasi Banjir

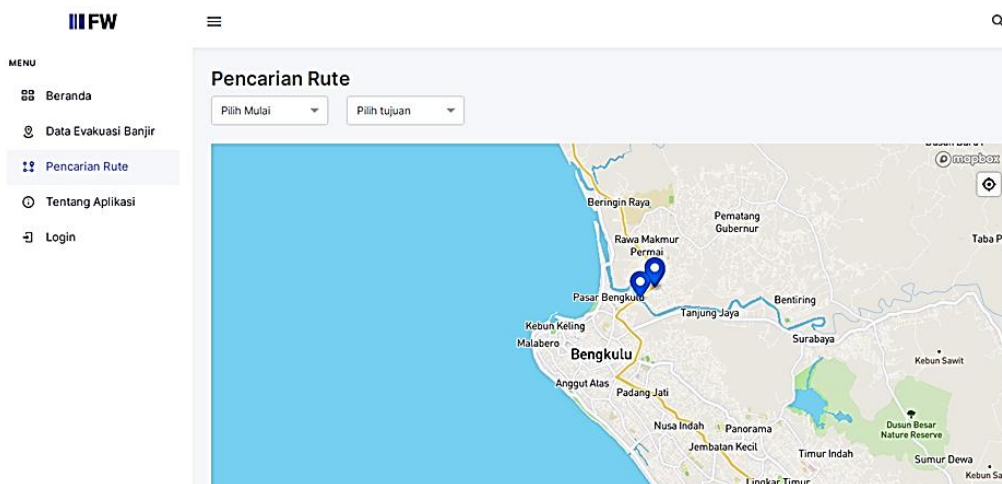
Gambar 9 merupakan halaman yang menampilkan menu data evakuasi banjir dari aplikasi, pada tampilan ini dapat menginformasika jumlah tempat evakuasi banjir sebagai bahan untuk penelitian.



Gambar 9. Halaman Menu Data Evakuasi Banjir

3.4.3 Halaman Pencarian Rute

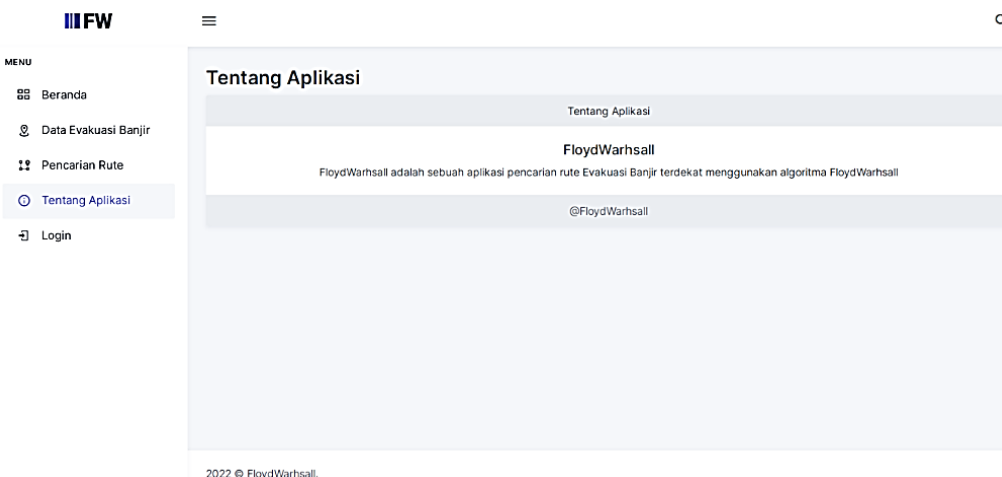
Gambar 10 merupakan halaman yang menampilkan pencarian rute awal dan rute tujuan, adapun dengan internet menyala agar dapat menampilkan map.



Gambar 10. Halaman Menu Pencarian Rute

3.4.4 Halaman Tentang Aplikasi

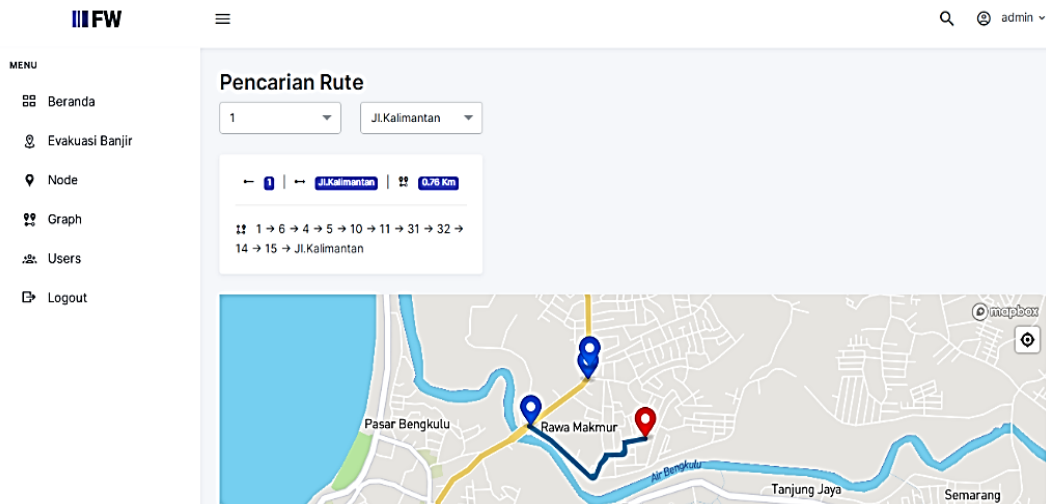
Gambar 11 merupakan halaman yang menampilkan penjelasan dari aplikasi itu sendiri, dan bisa ditambahkan informasi yang dibutuhkan.



Gambar 11. Halaman Tentang Aplikasi

3.4.5 Halaman Peta Menampilkan Rute Algoritma Floyd Warshall

Gambar 12 merupakan halaman yang menampilkan lokasi evakuasi banjir yang sudah dipilih dan lokasi titik berangkat pengguna dengan sudah menampilkan rute Floyd Warshall yang di tandai dengan warna merah.



Gambar 12. Halaman Peta dengan Rute Algoritma Floyd Warshal

4. KESIMPULAN

Dari hasil analisis terhadap aplikasi ini, jelas terlihat bahwa implementasi algoritma Floyd-Warshall dalam aplikasi web ini memiliki tujuan yang sangat penting, yaitu membantu masyarakat dalam menemukan jalur terpendek menuju titik evakuasi saat terjadi banjir. Algoritma Floyd-Warshall dikenal efektif dalam mencari jalur terpendek dalam graf berbobot, seperti yang diimplementasikan dalam konteks aplikasi ini untuk menghitung jarak terpendek antara setiap pasangan simpul atau titik dalam suatu area terdefinisi. Adapun hasil uji dari jarak yang didapatkan pada uji 1 sejauh 0.76 KM, pada uji 2 sejauh 0.40 KM dan pada uji 3 sejauh 0.41 KM. Ada beberapa kelemahan yang perlu diperhatikan. Salah satunya adalah ketergantungan algoritma ini pada definisi awal area atau graf yang digunakan. Algoritma Floyd-Warshall hanya dapat menghitung jalur terpendek di dalam area yang sudah terdefinisi dengan baik dalam hal simpul dan sisi-sisinya. Jika terdapat area baru yang belum terdefinisi atau perubahan pada struktur graf, seperti penambahan simpul atau sisi baru, algoritma ini tidak dapat beroperasi dengan efektif. Hal ini dapat mengakibatkan ketidakakuratan atau bahkan kegagalan dalam menemukan jalur terpendek atau titik evakuasi yang optimal saat situasi darurat banjir terjadi. Pengembang aplikasi perlu mempertimbangkan penggunaan strategi pembaruan dinamis pada algoritma Floyd-Warshall. Dengan memperbarui graf secara real-time atau saat ada perubahan struktur, aplikasi dapat tetap berfungsi secara optimal dan memberikan solusi yang akurat bagi pengguna dalam mencari jalur evakuasi. Selain itu, penting juga untuk mempertimbangkan integrasi dengan sistem informasi geografis (GIS) yang dapat mendukung pemetaan dan pembaruan data secara lebih efisien.

REFERENCES

- [1] M. Z. F. H. Fauzi Yulian, "PEMODELAN POTENSI BENCANA BANJIR DI DAERAH PERKOTAAN MENGGUNAKAN SIG," *Geometika*, vol. 28, no. 1, 2022.
- [2] R. T. Vuldari, M. Hasbi, and B. Tristanto, "Penerapan Algoritma Floyd Warshall pada Sistem Informasi Puskesmas Kabupaten Karanganyar," *Square : Journal of Mathematics and Mathematics Education*, vol. 3, no. 1, pp. 20–29, Apr. 2021, doi: 10.21580/square.2021.3.1.7716.
- [3] M. A. Rahmatullah and N. Nisar, "Algoritma Floyd Warshall untuk Menentukan Jarak Terpendek dalam Pencarian Oleh-Oleh di Bandar Lampung Berbasis Android," *Teknik Elektro dan Informatika*, vol. 2, no. 3, pp. 3031–500, 2024, doi: 10.61132/jupiter.v2i3.346.
- [4] H. P. L. Kurnia Putri Laily, "Jurnal Kajian dan Terapan Matematika Algoritma Floyd-Warshall dan Penerapannya pada Penentuan Rute Terpendek Objek Wisata di Kabupaten Gunungkidul," *Jurnal Kajian dan Terapan Matematika*, vol. 9, no. 1, 2023, [Online]. Available: <http://journal.student.uny.ac.id/ojs/index.php/jktm>.
- [5] A. Bramato Wicaksono Putra, A. Aulia Rachman, and A. Santoso, "Perbandingan Hasil Rute Terdekat Antar Rumah Sakit di Samarinda Menggunakan Algoritma A*(star) dan Floyd-Warshall," *Sistem Informasi dan Komputer*, vol. 9, no. 1, pp. 59–68, 2020, doi: 10.32736/sisfokom.v9i1.685.
- [6] A. M. Inayah, N. Cintya Resti, and I. Kediri, "ANALISA PERBANDINGAN ALGORITMA FLOYD-WARSHALL DAN ALGORITMA DIJKSTRAUNTUK PENENTUAN RUTE TERDEKAT," *Jurnal Ilmiah Matematika Realistik (JI-MR)*, vol. 4, no. 2, pp. 146–155, 2023.
- [7] N. Nugraha and D. Wuldari, "IMPLEMENTASI ALGORITMA FLOYD-WARSHALL PADA SISTEM INFORMASI GEOGRAFIS KAMPUS UNIKU BERBASIS ANDROID STUDI KASUS: (UNIVERSITAS

- KUNINGAN),” *Jurnal Teknologi dan Manajemen Informatika*, vol. 6, no. 2, 2021, [Online]. Available: <https://journal.uniku.ac.id/index.php/jejaring>
- [8] M. Syaifur Rohman, G. Wilujeng Saraswati, N. Anisa Sri Winarsih, and D. Nuswantoro, “IMPLEMENTASI ALGORITMA FLOYD WARSHALL PADA APLIKASI DEWAN MASJID INDONESIA (DMI) KOTA SEMARANG UNTUK MENENTUKAN MASJID TERDEKAT,” *Jurnal Informatika: Jurnal pengembangan IT (JPIT)*, vol. 8, no. 3, 2023.
- [9] T. M. Diansyah, D. Handoko, and corespondent author, “Penerapan Algoritma Floyd Warshall dengan Menggunakan Euclidean Distance dalam Menentukan Rute Terbaik,” *Jurnal Ilmu Komputer dan Sistem Informasi (JIRSI)*, vol. 2, no. 2, 2023, [Online]. Available: <https://jurnal.unity-academy.sch.id/index.php/jirsi/index>
- [10] B. Setiawan and L. Hermawan, “JIP (Jurnal Informatika Polinema) EDUKASI PROTOKOL KESEHATAN BERBASIS GAME MENGGUNAKAN ALGORITMA FLOYD WARSHALL,” *Jurnal Informatika Polinema*, vol. 8, no. 1, 2021.
- [11] M. Syaifur Rohman, G. Wilujeng Saraswati, N. Anisa Sri Winarsih, and D. Nuswantoro, “IMPLEMENTASI ALGORITMA FLOYD WARSHALL PADA APLIKASI DEWAN MASJID INDONESIA (DMI) KOTA SEMARANG UNTUK MENENTUKAN MASJID TERDEKAT,” *Pendrikan Kidul, Kec. Semarang Tengah*, vol. 8, no. 3, 2023.
- [12] F. Gunawan, Y. Fatma, H. Mukhtar, F. I. Komputer, and U. M. Riau, “APLIKASI Pencarian Rute Terpendek Tempat Wisata di Kota Pekanbaru Menggunakan Floyd Warshall,” *Fasilkom*, vol. 10, no. 1, 2020, [Online]. Available: www.statista.com
- [13] I. Hndayani, “Sistem Informasi Geografis Daerah Rawan Banjir Dengan Menerapkan Metode Floyd Warshall (Studi Kasus: BPBD Binjai),” *Informasi dan Teknologi Ilmiah*, vol. 9, no. 2, 2022.
- [14] A. M. Inayah, N. Cintya Resti, and I. Kediri, “ANALISA PERBANDINGAN ALGORITMA FLOYD-WARSHALL DAN ALGORITMA DIJKSTRAUNTUK PENENTUAN RUTE TERDEKAT,” *Jurnal Ilmiah Matematika Realistik*, vol. 4, no. 2, pp. 146–155, 2023.
- [15] S. Farhan, S. Andryana, N. Hayati,) Fakultas, T. Komunikasi, and D. Informatika, “IMPLEMENTASI BELLMAN-FORD DAN FLOYD-WARSHALL DALAM MENENTUKAN JALUR TERPENDEK MENUJU UNIVERSITAS NASIONAL BERBASIS ANDROID,” *Jurnal Ilmiah Penelitian dan Pembelajaran Informatika*, vol. 5, no. 2, 2020.
- [16] R. Bangun *et al.*, “Rancang Bangun Aplikasi Pencarian Toko Oleh-Oleh Terdekat Berbasis Android Menggunakan Algoritma Floyd Warshall,” *Jurnal Ilmiah Ilmu Komputer Banthayo Lo Komputer*, vol. 3, no. 1, 2024.
- [17] S. P. Handayani Meri, “PENERAPAN METODE FLOYD WARSHALL TEMPAT BERSEJARAH DI SUMATERA UTARA BERBASIS WEBSITE,” *Jurnal Teknologi Informasi dan Indutri*, vol. 3, no. 1, 2023.
- [18] E. Feriani and A. Tarigan, “Implementasi Algoritma Floyd-Warshall Dalam Perancangan Aplikasi Jareno (Jasa Renovasi),” *JURNAL ILMIAH CORE IT*, vol. 10, no. 4, pp. 1978–1520, 2020.
- [19] H. Wijayanto, D. Nugroho, and B. A. Santoso, “SISTEM INFORMASI GEOGRAFIS PENENTUAN RUTE TERPENDEK LOKASI VILLA MENGGUNAKAN ALGORITMA FLOYD WARSHALL,” *Jurnal Teknologi Informasi dan Komunikasi (TIKomSiN)*, vol. 8, no. 1, Apr. 2020, doi: 10.30646/tikomsin.v8i1.474.
- [20] F. Ridwan and R. D. Agustin, “Penggunaan Algoritma Floyd-Warshall untuk Menentukan Rute Terpendek Menuju Air Terjun Waimarang,” *Laplace: Jurnal Pendidikan Matematika*, vol. 3, no. 2, pp. 87–94, Oct. 2020, doi: 10.31537/laplace.v3i2.375.