

## RANCANG BANGUN SISTEM INFORMASI GEOGRAFIS MENGGUNAKAN KOMBINASI NODE BERBASIS ALGORITMA DIJKSTRA PADA PENENTUAN JALUR TERPENDEK EVAKUASI TSUNAMI DI TELUK PALU

Dwi Shinta Angreni<sup>1</sup>, Wahyu Budiman<sup>2</sup>, Yusuf Anshori<sup>3</sup>, Wirdayanti<sup>4</sup>, Rizqa Raaiqa Bintana<sup>5</sup>

Program Studi Sistem Informasi, Universitas Tadulako<sup>1,2,3,4</sup>, Program Studi Sistem Informasi, Universitas Jambi<sup>5</sup>  
Email: ds.angreni@untad.ac.id<sup>1</sup>

### ABSTRACT

*Indonesia, an archipelagic country situated on the Equator and surrounded by the Ring of Fire, is highly susceptible to natural disasters such as tectonic earthquakes and volcanic eruptions. Palu, one of Indonesia's seismically active regions, is traversed by the Palu-Koro Fault, which has the potential to trigger strong earthquakes and tsunamis, as evidenced by the 7.4-magnitude earthquake in September 2018. To mitigate disaster risks, it is crucial to understand community vulnerability and utilize Geographic Information System (GIS) technology. This study designs a GIS-based system using the modified Dijkstra's Algorithm to determine the shortest tsunami evacuation routes in Palu. Testing results indicate that the system is effective, with a user satisfaction rate of 89.33% and satisfactory route prediction accuracy compared to Google Maps. The system can be relied upon to help communities find the nearest evacuation routes, thereby enhancing safety and preparedness in the face of potential disasters.*

**Keywords:** *Tsunami evacuation rout, Geographic Information System (GIS), Dijkstra Algorithm*

### INTISARI

Indonesia, sebagai negara kepulauan yang terletak di Khatulistiwa dan dikelilingi oleh Cincin Api, sangat rentan terhadap bencana alam seperti gempa bumi tektonik dan letusan gunung berapi. Palu, salah satu wilayah di Indonesia yang aktif secara seismik, dilalui oleh Sesar Palu-Koro, yang memiliki potensi memicu gempa bumi kuat dan tsunami, seperti yang terjadi pada gempa berkekuatan 7,4 SR pada September 2018. Untuk mengurangi risiko bencana, sangat penting memahami kerentanan masyarakat dan memanfaatkan teknologi Sistem Informasi Geografis (SIG). Penelitian ini merancang sistem berbasis SIG menggunakan Algoritma Dijkstra yang dimodifikasi untuk menentukan rute evakuasi tsunami terpendek di Palu. Hasil pengujian menunjukkan bahwa sistem ini efektif, dengan tingkat kepuasan pengguna sebesar 89,33% dan akurasi prediksi rute yang memuaskan dibandingkan dengan Google Maps. Sistem ini dapat diandalkan untuk membantu masyarakat menemukan rute evakuasi terdekat, sehingga meningkatkan keselamatan dan kesiapsiagaan dalam menghadapi potensi bencana.

Kata Kunci : Rute evakuasi tsunami, Sistem Informasi Geografis (SIG), Algoritma Dijkstra

### I. PENDAHULUAN

Indonesia adalah sebuah negara kepulauan yang terletak di Garis Khatulistiwa dan dikelilingi oleh Cincin Api. Cincin Api merupakan garis pertemuan lempeng benua Asia dengan lempeng samudra hindia, yang terletak di sepanjang garis pantai selatan Indonesia. Juga garis pertemuan antara lempeng pasifik dan lempeng asia yang terletak di sepanjang kepulauan Maluku. Akibat situasi geologis ini, di Indonesia juga banyak terdapat sesar dan gunung berapi. Oleh karena itu di Indonesia banyak terdapat bencana alam yang

terkait dengan gempa tektonik dan letusan gunung berapi [1]

Kota Palu merupakan salah satu kawasan seismik aktif di Indonesia karena dilalui segmentasi sesar yang sangat berpotensi membangkitkan gempa bumi kuat yaitu Sesar Palu-Koro yang memanjang dari Palu ke arah Selatan dan Tenggara. Ditinjau dari kedalaman gempa buminya, aktivitas gempa bumi di zona ini tampak didominasi oleh gempa bumi kedalaman dangkal antara 0 hingga 60 kilometer. Dengan demikian, aktivitas tersebut beresiko untuk menimbulkan tsunami. Pada bulan

September 2018, gempa bumi berkekuatan 7,4 skala richter mengguncang wilayah Sulawesi Tengah. Gempa ini memicu terjadinya bencana tsunami di Palu yang menyebabkan kerugian besar. Kerugian besar akibat bencana tsunami dikarenakan tingkat kerentanan. Hal dasar yang dapat dilakukan untuk mengurangi resiko bencana tsunami yaitu dengan mengetahui kemampuan masyarakat dalam menghadapi bencana tsunami dengan cara mengetahui tingkat kerentanan bencana tsunami [2].

Sistem informasi geografis (SIG) merupakan sistem yang dirancang untuk bekerja dengan data yang tereferensi secara spasial atau koordinat-koordinat geografi. Sistem informasi geografis adalah bentuk sistem informasi yang menyajikan informasi dalam bentuk grafis dengan menggunakan peta sebagai antar muka. Teknologi Sistem Informasi Geografis dapat digunakan untuk investigasi ilmiah, pengelolaan sumber daya, perencanaan pembangunan, kartografi dan perencanaan rute. Misalnya, SIG bisa membantu perencana untuk cepat menghitung waktu tanggap darurat saat terjadi bencana alam [3].

Berkaitan dengan masalah yang telah dijelaskan maka salah satu solusi yang dapat dilakukan adalah dengan membuat sistem yang dapat digunakan masyarakat atau pemerintah untuk mengarahkan korban bencana alam ke tempat evakuasi. Dengan adanya sistem ini, pihak BPBD Kota Palu dapat memberikan informasi jalur terpendek yang dapat dilalui korban bencana alam menuju tempat evakuasi terdekat. Sebelumnya tahun 2019 telah di bangun sebuah disain aplikasi evakuasi tsunami menggunakan Algoritma Genetika oleh Anshori dkk [4]. Selain itu Putri [5] melakukan perbandingan antara dua metode Dijkstra dan Ant Colony dalam pencarian rute terbaik evakuasi gempa dan tsunami pada tahun 2022.

Berdasarkan latar belakang tersebut diusulkan penelitian dengan Rancang Bangun Sistem Informasi Geografis Menggunakan Kombinasi Node Berbasis Algoritma Dijkstra Pada Penentuan

Jalur Terpendek Evakuasi Tsunami Di Teluk Palu yang diharapkan dapat membantu korban bencana alam menemukan jalur terpendek menuju tempat evakuasi.

## II. LANDASAN TEORI

### A. Algoritma Dijkstra

Algoritma Dijkstra merupakan algoritma yang digunakan untuk menentukan jarak terpendek dari satu vertex ke vertex yang lainnya pada suatu graf berbobot, jarak antar vertex adalah nilai bobot dari setiap edge pada graf. Suatu bobot harus bernilai positif (bobot  $\geq 0$ ). Algoritma Dijkstra ditemukan oleh Edger Wybe Dijkstra. Algoritma Dijkstra dikenal juga sebagai algoritma greedy yaitu algoritma yang penyelesaian masalah dengan mencari nilai maksimum. Cara kerja algoritma Dijkstra dalam pencarian jarak terpendek adalah perhitungan dari vertex asal ke vertex terdekatnya, kemudian ke vertex yang kedua, dan seterusnya [6]

### B. Kombinasi Node Berdasarkan Algoritma Dijkstra

Salah satu algoritma yang terkenal dan paling umum digunakan dalam mencari jalur terpendek adalah algoritma Dijkstra. Namun, penggunaan algoritma Dijkstra tidaklah sederhana dan membutuhkan lebih banyak memori. Kombinasi Node [7] mengimplementasikan algoritma Dijkstra dengan menghapus node yang terdekat dengan node awal. Penggunaan memori lebih efisien dengan menghapus node daripada algoritma Dijkstra asli. Ide dasar dari kombinasi node adalah menggabungkan node dan tetap mempertahankan satu set pelabelan seperti algoritma Dijkstra. Metode kombinasi node menemukan jalur terpendek secara berulang dengan menemukan tetangga terdekat dari node awal, menggabungkan node tersebut dengan awal dan memperbarui bobot tepi yang terhubung ke node terdekat. Berikut ini adalah penjelasan dari pseudocode di atas:

#### 1. Inisialisasi

- Set nilai bobot dari simpul awal  $s$  ke dirinya sendiri ( $W[s,u]$ ) menjadi 0.
- Tetapkan  $v_u$  (satu dari simpul terdekat dari  $s$  yang masih ada di  $V$ ) menjadi  $v_s$ .
- Tetapkan jalur  $P[u]$  dari simpul  $s$  ke simpul  $u$  sebagai  $[s]$ .
- Hapus simpul  $s$  dari himpunan simpul  $V$ .

## 2. Iterasi Utama:

Lakukan iterasi selama bobot dari simpul  $s$  ke simpul  $u$  masih kurang dari tak hingga (artinya jalur masih ditemukan) dan himpunan simpul  $V$  tidak kosong.

## 3. Kombinasi Node:

Hapus simpul  $u$  dari himpunan simpul  $V$ .

## 4. Iterasi untuk Setiap Simpul yang Tersisa:

Lakukan iterasi untuk setiap simpul  $j$  yang masih ada di himpunan simpul  $V$ .

## 5. Perbandingan Bobot:

Jika jumlah bobot dari jalur terpendek saat ini dari  $s$  ke  $u$  ditambah dengan bobot dari  $u$  ke  $j$  lebih kecil dari bobot jalur terpendek dari  $s$  ke  $j$ , maka lakukan langkah-langkah berikut:

## 6. Pembaruan Jalur:

Perbarui jalur terpendek dari  $s$  ke  $j$  dengan menggabungkan jalur dari  $s$  ke  $u$  dan  $u$  ke  $j$  ( $P[j] := P[u] \cup \{j\}$ ).

## 7. Pembaruan Bobot:

Perbarui bobot dari jalur terpendek dari  $s$  ke  $j$  dengan jumlah bobot dari jalur terpendek saat ini dari  $s$  ke  $u$  ditambah dengan bobot dari  $u$  ke  $j$  ( $W[s,j] := W[s,u] + W[u,j]$ ).

## 8. Pencarian Tetangga Terdekat:

Temukan simpul tetangga terdekat dari  $s$  yang masih ada di himpunan  $V$  dan tetapkan sebagai  $v_u$ .

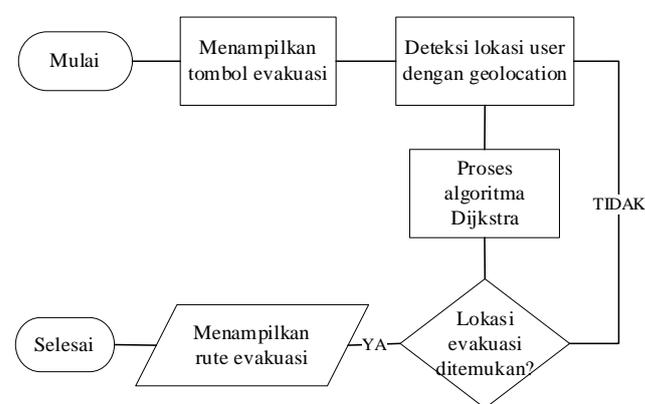
## 9. Hasil Akhir:

Pada akhir algoritma, baris ke- $s$  dalam matriks bobot  $W$  akan berisi jarak terpendek dari simpul  $s$  ke semua simpul lainnya dalam graf.

## III. METODE PENELITIAN

### 2.1. Flowchart

Flowchart sistem dapat dilihat pada gambar 3.1 berikut:



Gambar 1 Flowchart Sistem

Flowchart ini mempresentasikan langkah-langkah yang diambil oleh aplikasi dalam membantu pengguna mencari lokasi evakuasi. Pada awalnya, pengguna memulai dan aplikasi akan menampilkan tombol evakuasi, yang menjadi pemicu untuk memulai proses. Setelah tombol evakuasi ditekan, aplikasi menggunakan geolocation pada perangkat pengguna untuk menentukan posisi pengguna. Proses selanjutnya melibatkan pencarian rute terpendek, di mana algoritma Dijkstra digunakan untuk menemukan jalur terpendek menuju lokasi evakuasi. Jika lokasi evakuasi berhasil ditemukan, aplikasi akan memberikan informasi tentang rute yang harus diikuti oleh pengguna. Namun, jika lokasi evakuasi tidak dapat ditemukan, aplikasi akan menginisiasi kembali proses deteksi lokasi pengguna untuk mencoba menemukan lokasi evakuasi kembali. Dengan demikian, flowchart ini menciptakan suatu siklus yang berkelanjutan hingga aplikasi berhasil menemukan lokasi evakuasi.

#### IV. HASIL PENELITIAN DAN PEMBAHASAN

##### 1. Data lokasi evakuasi

Pengumpulan data dilakukan melalui metode wawancara dan studi lapangan guna memperoleh informasi yang komprehensif terkait lokasi evakuasi dan koordinat persimpangan jalan. Lokasi evakuasi yang didapatkan sebanyak 17 posko aman yang tersebar di seluruh kota Palu.

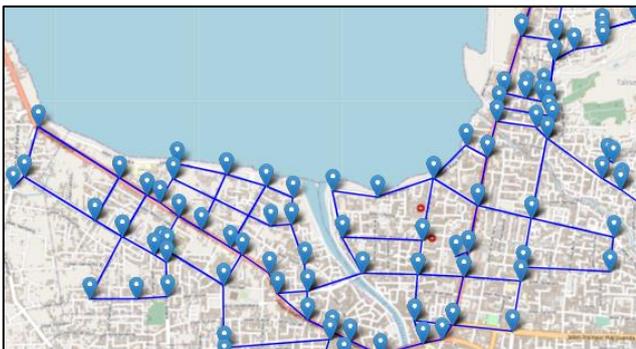
##### 2. Data Simpul

Pada penelitian ini, data simpul merujuk pada informasi koordinat persimpangan jalan yang diperoleh selama studi lapangan. Simpul ini merupakan titik-titik yang membentuk struktur jaringan jalan di wilayah sekitar teluk Palu. Proses pengumpulan data simpul dilakukan dengan menggunakan perangkat Android atau GPS, yang memungkinkan penentuan titik koordinat geografis yang akurat untuk setiap persimpangan.

melihat perbandingan gerakan yang dilakukan oleh lengan robot. Pada tabel 4 merupakan pengujian kendali lengan robot dengan menggunakan potensiometer.

##### 3. Pembentukan Graf

Setelah memperoleh data lokasi evakuasi dan data simpul, langkah berikutnya adalah membentuk graf atau peta jaringan untuk memberikan representasi visual di wilayah penelitian. Berikut hasil pembentukan graf di teluk Palu.



Gambar 2. Graf pada wilayah sekitar teluk palu

##### 4. Implementasi pada aplikasi dengan menggunakan Google Maps

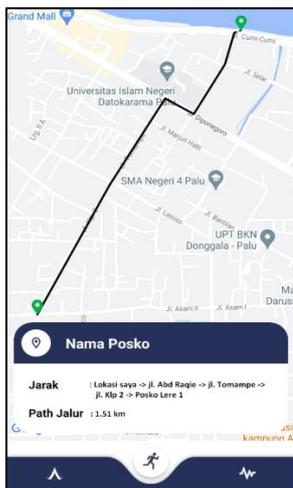
Setelah semua data terkumpul, langkah selanjutnya adalah mengimplementasikan informasi lokasi evakuasi dan data simpul ke dalam aplikasi yang dikembangkan menggunakan Android Studio. Aplikasi ini dirancang dengan memanfaatkan API Google Maps guna menyajikan peta yang interaktif dan menyediakan layanan pencarian lokasi evakuasi.



Gambar 3. Halaman button evakuasi



Gambar 4. Halaman informasi lokasi



Gambar 5. Halaman jalur evakuasi tsunami

Fungsionalitas pencarian ini diintegrasikan dengan data simpul, sehingga pengguna dapat mengakses informasi detail mengenai rute evakuasi dan jarak tempuh antar lokasi. Selain itu, peta interaktif juga memungkinkan pengguna untuk mendapatkan pandangan visual terhadap topologi jaringan jalan di sekitar wilayah teluk Palu. Dengan implementasi ini, aplikasi Android yang menggunakan Google Maps API menjadi alat yang efektif dalam mendukung upaya mitigasi bencana. Hasil implementasi sistem dapat dilihat pada Gambar 3 sampai Gambar 5.

Pada penelitian ini dilakukan pengujian Beta Testing untuk memberikan penilaian terhadap aplikasi yang dibangun. Berdasarkan 6 pertanyaan yang diisi oleh 20 responden menghasilkan nilai rata-rata 89,33%. Pada pengujian akurasi penulis peneliti menggunakan dua metrik evaluasi utama, yaitu Mean Absolute Error (MAE) dan Root Mean Square Error (RMSE), untuk mengukur akurasi algoritma dalam mencari jalur evakuasi terpendek dibandingkan dengan perhitungan jarak yang diberikan oleh Google Maps. Berdasarkan perhitungan yang telah dilakukan maka diperoleh nilai MAE sekitar 0.08 dan RMSE sekitar 0.09, hasil ini memberikan gambaran seberapa besar kesalahan antara panjang rute yang dihasilkan oleh algoritma Dijkstra dan Google Maps. Semakin rendah nilai MAE dan RMSE, semakin baik kinerja algoritma Dijkstra dalam menghasilkan

rute yang mendekati rute Google Maps pada dataset evakuasi tsunami yang diberikan.

## V. KESIMPULAN

1. Pengujian pada lingkungan pengembang aplikasi menunjukkan bahwa antarmuka pengguna merespons dengan baik terhadap berbagai input yang diuji. Sistem mampu mengolah data secara efektif dan memberikan output sesuai dengan spesifikasi yang diterapkan. Selain itu, pengujian di lingkungan pengguna melalui penyebaran kuesioner menghasilkan tingkat kepuasan sebesar 89,33%, yang mencerminkan kepuasan tinggi terhadap fungsionalitas dan pengalaman pengguna.
2. Proses evaluasi juga menunjukkan akurasi yang memuaskan dalam memprediksi panjang rute dibandingkan dengan Google Maps. Berdasarkan hasil tersebut, sistem ini dapat dimanfaatkan untuk pencarian jalur evakuasi tsunami dengan akurasi yang tinggi, sehingga Algoritma Kombinasi Node terbukti andal dalam memilih rute terpendek. Hal ini meningkatkan keselamatan dan kesiapsiagaan masyarakat dalam menghadapi bencana.

## DAFTAR PUSTAKA

- [1] H. Suprayitno and R. A. A. Soemitro, "Pemikiran awal tentang prinsip tindakan mitigasi preventif resiko bencana alam bagi manajemen aset infrastruktur & fasilitas," *Jurnal Manajemen Aset Infrastruktur & Fasilitas*, vol. 3, 2019, Accessed: Oct. 10, 2024. [Online]. Available: <http://iptek.its.ac.id/index.php/jmaif/article/view/5187>
- [2] H. T. Sarapang, O. H. Rogi, and P. Hanny, "Analisis kerentanan bencana tsunami di Kota Palu," *Spasial*, vol. 6, no. 2, pp. 432–439, 2019.
- [3] A. G. Rumondor, S. R. Sentinuwo, and A. M. Sambul, "Perancangan jalur terpendek

evakuasi bencana di kawasan boulevard manado menggunakan algoritma dijkstra,” *Jurnal Teknik Informatika*, vol. 14, no. 2, pp. 261–268, 2019.

- [4] Y. Anshori, D. S. Angreni, and S. R. Arifin, “Tsunami Evacuation Routes Optimization using Genetic Algorithms: A Case Study in Palu,” in *MATEC Web of Conferences*, EDP Sciences, 2020, p. 01008. Accessed: Oct. 10, 2024. [Online]. Available: [https://www.matec-conferences.org/articles/mateconf/abs/2020/27/mateconf\\_icudr2019\\_01008/mateconf\\_icudr2019\\_01008.html](https://www.matec-conferences.org/articles/mateconf/abs/2020/27/mateconf_icudr2019_01008/mateconf_icudr2019_01008.html)
- [5] R. F. Putri, “PERBANDINGAN ALGORITMA DIJKSTRA DAN ANT COLONY UNTUK JALUR EVAKUASI TERDEKAT BENCANA ALAM GEMPA BUMI DAN TSUNAMI,” PhD Thesis, Universitas Malikussaleh, 2022. Accessed: Oct. 10, 2024. [Online]. Available: <https://rama.unimal.ac.id/id/eprint/191/>
- [6] M. K. Harahap and N. Khairina, “Pencarian Jalur Terpendek dengan Algoritma Dijkstra,” *Sinkron: jurnal dan penelitian teknik informatika*, vol. 2, no. 1, pp. 18–23, 2017.
- [7] B. Amaliah, C. Fatichah, and O. Riptianingdyah, “Finding the Shortest Paths Among Cities in Java Island Using Node Combination Based on Dijkstra Algorithm,” *International Journal on Smart Sensing and Intelligent Systems*, vol. 9, no. 4, pp. 2219–2236, Jan. 2016, doi: 10.21307/ijssis-2017-961