

PENCARIAN JALUR ALTERNATIF SEBAGAI SISTEM EVAKUASI BENCANA TANAH LONGSOR DI KABUPATEN PONOROGO DENGAN ALGORITMA A-STAR BERBASIS PERANGKAT BERGERAK

¹Arif Basofi, ²Rizal Rahmana, ³Arna Fariza

¹²³Politeknik Elektronika Negeri Surabaya

Email: ¹ariv@pens.ac.id, ²rizalrahmana@it.student.pens.ac.id, ³arna@pens.ac.id

Abstrak. Kabupaten Ponorogo merupakan salah satu area di provinsi Jawa Timur yang sering dilanda longsor karena letaknya yang berdekatan dengan perbukitan. Informasi mengenai jalur alternatif sebagai sistem evakuasi bencana tanah longsor sangat diperlukan untuk mengurangi dampak bencana terhadap distribusi ekonomi dan transportasi masyarakat. Aplikasi jalur alternatif dibangun berbasis perangkat bergerak menggunakan algoritma A-star (A*) dan divisualisasikan dengan Google Map. Dengan aplikasi ini, pengguna dapat memperoleh informasi rute alternatif dengan cepat, mudah dan akurat.

Kata Kunci: Jalur alternatif, tanah longsor, sistem evakuasi, algoritma A*

Tanah longsor merupakan salah satu bencana alam yang menimbulkan resiko terhadap kehidupan manusia, kerugian harta benda dan kerusakan lingkungan. Pulau Jawa merupakan salah satu wilayah di Indonesia yang berpotensi rawan longsor. Potensi rawan longsor disebabkan karena kondisi-kondisi geomorfologi, geologi, tanah dan batuan yang menyusun lereng dan bukit, iklim serta hidrologi. Selain itu juga banyak dijumpai lereng dan bukit yang miring dan bergelombang yang berpotensi mengalami gerakan massa tanah dan batuan [1].

Kabupaten Ponorogo merupakan salah satu kabupaten di Provinsi Jawa Timur, Kabupaten ini terletak pada koordinat geografis di antara 111° 17' - 111° 52' BT dan 7° 49' - 8° 20' LS dengan ketinggian antara 92 sampai dengan 2.563 meter di atas permukaan laut. Kabupaten Ponorogo merupakan salah satu daerah di Jawa Timur yang berpotensi mengalami bencana tanah longsor karena bentuk morfologi Kabupaten Ponorogo yang bervariasi seperti dataran tinggi dan perbukitan.

Siklus manajemen bencana yang terdiri dari komponen mitigasi, kesiapsiagaan, respon, pemulihan yang perlu dilakukan secara utuh. Fase pertama, mitigasi adalah untuk memperkecil dampak dari bencana, meningkatkan kesiapsiagaan masyarakat. Fase kedua, kesiapsiagaan yaitu merencanakan bagaimana menanggapi bencana dilakukan di dalam fase ini. Hal

tersebut meliputi merencanakan kesiapsiagaan, penilaian kerentanan, kelembagaan, sistem informasi, basis sumberdaya, membangun sekolah, siaga bencana, sistem peringatan dini, mekanisme tanggap dan pelatihan, kemudian diuji coba kesiapsiagaan terhadap bencana. Fase ketiga, respon adalah upaya memperkecil kerusakan yang disebabkan oleh bencana, pencarian dan penyelamatan korban. Fase keempat, pemulihan yaitu tindakan mengembalikan masyarakat ke kondisi normal [2] [3].

Akses dan penggunaan data spasial yang cepat dan akurat berperan penting untuk pengambilan keputusan pada saat tanggap darurat bencana. Semakin cepat mengambil keputusan yang baik akan mengurangi risiko kerugian korban jiwa, meminimalkan jumlah kerusakan bencana dan menghemat biaya pemulihan bencana. Saat ini, aksesibilitas data spasial masih lemah, hanya 29,63% basisdata digital yang dapat diakses melalui internet [4]. Beberapa penelitian pengelolaan informasi kebencanaan berbasis sistem informasi geografis telah dikembangkan. Penggunaan sistem informasi geografi (SIG) pada Manajemen bencana difokuskan pada relief, penanganan korban, rehabilitasi dan perbaikan. Keefektifan keluaran SIG berdasarkan pada kualitas dan ketersediaan data yang relevan [3]. Perangkat lunak aplikasi pengelolaan dan penanggulangan bencana alam Garut berbasis sistem informasi geografis digunakan sebagai sarana

informasi dalam hal mitigasi bencana alam [5]. Sistem informasi geografis berbasis web telah dikembangkan untuk menyajikan informasi tentang daerah rawan bencana banjir di kota Kendari [6]. Sistem informasi geografis digunakan untuk pemodelan potensi bencana tanah longsor di Kabupaten Semarang dengan menggunakan metode Analytical Hierarchy Process (AHP) [7].

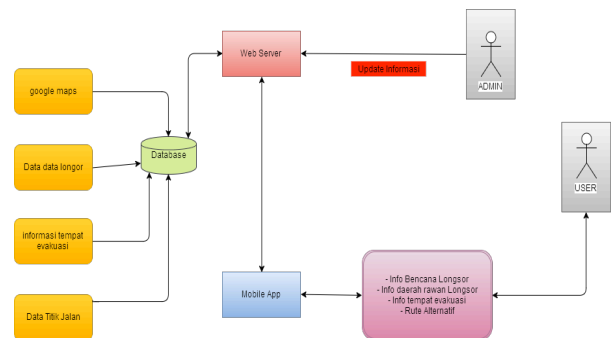
Salah satu informasi yang diperlukan pada sistem evakuasi bencana tanah longsor adalah pencarian jalur alternatif pada jalan yang terdampak bencana tanah longsor. Hal ini diperlukan untuk kelancaran jalur distribusi ekonomi dan transportasi masyarakat yang terdampak bencana tanah longsor. Musabbichin dan Yunitarini (2015) melakukan penelitian untuk penentuan jalur terpendek untuk menghindari daerah rawan banjir di propinsi Jawa Timur dengan pendekatan Breadth First Search (BFS) yang menggunakan fungsi heuristic yang direpresentasikan dalam konsep graph [8]. Algoritma A* memiliki keunggulan yaitu beban komputasi dan waktu simulasi yang paling kecil dibandingkan algoritma *Dijkstra* dan *Floyd-Warshall* tanpa mempengaruhi hasil pencarian jalur terpendek [9]. Hal ini disebabkan algoritma A* melakukan operasi pencarian dengan memanfaatkan nilai heuristic terhadap simpul tujuan, sehingga tidak semua simpul dilakukan pengecekan. Penggunaan algoritma A* untuk pencarian jalur telah digunakan untuk memprediksi jalur terpendek tempat-tempat wisata di wilayah Kota Bandung dengan perangkat android [10]. Algoritma A* juga digunakan untuk mencari jalur terpendek merupakan suatu permasalahan yang sering terjadi pada pengunjung rumah sakit untuk menemukan gedung atau ruangan yang dicari pada Rumah Sakit Umum Bahteramas [11].

Pada penelitian ini dikembangkan sistem pencarian jalur alternatif pada sistem evakuasi bencana tanah longsor berbasis android. Dalam penyampaian informasi harus akurat, cepat, mudah dikomunikasikan dan dapat dipercaya. Algoritma A* digunakan untuk pencarian jalur terpendek karena keunggulan beban komputasi yang lebih kecil tanpa mempengaruhi hasil pencarian jalur terpendek. Hal ini sesuai dengan tujuan penyampaian informasi dan penggunaan perangkat android. Sistem dibangun berbasis

client-server dimana admin memasukkan informasi ruas jalan yang terdampak bencana tanah longsor, sedangkan pemakai mendapatkan informasi melalui perangkat bergerak yang terdiri dari informasi lokasi bencana, jalur alternatif dan tempat evakuasi.

I. Metodologi

Diagram sistem *client-server* yang dibangun dapat dilihat pada Gambar 1. Sistem terdiri dari 2 jenis yaitu berbasis web dan berbasis perangkat bergerak. Sistem berbasis web digunakan oleh admin untuk menambah dan mengubah informasi. Sistem berbasis perangkat bergerak digunakan oleh *user* untuk mencari jalur alternatif.



Gambar 1. Diagram Umum Sistem

Informasi yang disimpan didalam basis data yaitu meliputi informasi data daerah yang sedang terjadi longsor, data titik jalan, data tempat evakuasi dan data longsor keseluruhan. Informasi mengenai daerah yang terkena bencana tanah longsor, tempat evakuasi, dan data titik jalan diintegrasikan dengan *Google Maps* sehingga data ditampilkan dalam bentuk peta dan ditampilkan di laman dan perangkat bergerak. Admin memasukkan informasi lokasi ruas jalan yang terdampak bencana tanah longsor, pemakai mengakses informasi lokasi bencana longsor, jalur alternatif dan tempat evakuasi.

Algoritma A-Star (A*)

Algoritma A* menggunakan estimasi jarak terdekat untuk mencapai tujuan (goal) dan memiliki nilai heuristic yang digunakan sebagai dasar pertimbangan. Heuristik adalah kriteria, metoda, atau prinsip-prinsip untuk menentukan pilihan sejumlah alternatif untuk mencapai sasaran dengan efektif [9].

Algoritma A* merupakan salah satu jenis algoritma yang digunakan untuk menyelesaikan kasus yang berhubungan dengan *path finding* (pencarian jalan). Dalam hasil pencariannya A* dikatakan komplit dan optimal. Algoritma A* menggunakan teknik pencarian *Breadth First Search*, dimana pencarian dilakukan dengan cara melebar ke setiap node pada level yang sama, dan nantinya akan menemukan rute terbaik dari titik awal sampai tujuan. Algoritma A* juga dilengkapi dengansuatu fungsi *heuristic*. Fungsi *heuristic* yang terdapat pada algoritma A* digunakan sebagai optimasi dalam menentukan node tujuan yang akan dipilih [11].

Algoritma A* merupakan algoritma *breadth-first search* yang menggunakan fungsi heuristik yang kompleks untuk memilih *path*. Algoritma *breadth-first search* selalu memperluas jalan untuk menuju node yang akan dilewati, tetapi tidak memperhitungkan *cost* ke simpul tersebut [11].

Algoritma ini memeriksa node dengan menggabungkan $g(n)$, yaitu *cost* yang dibutuhkan untuk mencapai sebuah node dan $h(n)$ yaitu *cost* yang didapat dari node ke tujuan. Sehingga dapat dirumuskan sebagai:

$$f(n) = g(n) + h(n) \quad (1)$$

dimana

$f(n)$ adalah estimasi total biaya (*cost*) sebuah jalur (*path*) dari node awal ke node tujuan (*goal*) melalui node n.

$g(n)$ adalah biaya (*cost*) yang dibutuhkan oleh sebuah jalur (*path*) untuk mencapai node n dari node awal.

$h(n)$ adalah estimasi biaya (*cost*) sebuah jalur (*path*).

Beberapa terminologi yang terdapat pada algoritma A* adalah *Starting point* merupakan terminology untuk posisi awal sebuah benda [11].

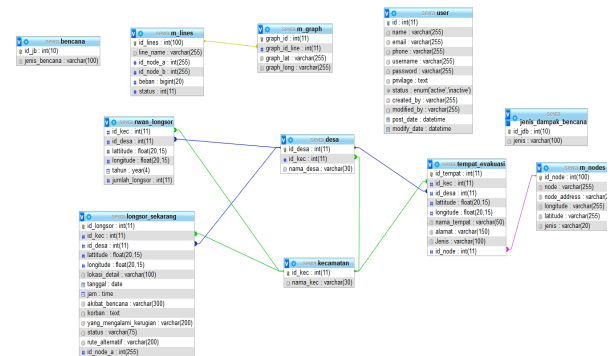
1. *Starting point* merupakan terminology untuk posisi awal sebuah benda.
2. *Simpul (node)* merupakan petak-petak kecil sebagai representasi dari *pathfinding*. Bentuknya dapat berupa persegi, lingkaran maupun segitiga.
3. *A** merupakan simpul yang sedang dijalankan dengan algoritma pencarian jalur terpendek.
4. *Open list* merupakan tempat penyimpanan data simpul yang mungkin

diakses dari *starting point* maupun symbol yang sedang dijalankan.

5. *Closed list* merupakan tempat penyimpanan data simpul sebelum A* yang juga merupakan bagian dari jalur terpendek yang telah dihasilkan didapatkan.
6. *Harga (cost)* merupakan nilai dari $f(n)$.

Perancangan Basis Data

Skema rancangan basis data dapat dilihat pada Gambar 2.



Gambar 2. Skema Rancangan Basis Data

Tabel *user* digunakan untuk menyimpan data pemakai yang menggunakan aplikasi web dan perangkat bergerak. Tabel *desa*, *kecamatan* dan *rawan_longsor* digunakan untuk menyimpan informasi daerah yang rawan longsor di desa-desa yang ada di Kabupaten Ponorogo. Tabel *m_graph*, *m_node* dan *m_line* digunakan untuk menyimpan titik-titik dan ruas jalan yang terhubung pada titik-titik tersebut untuk menghasilkan jalur alternatif dengan algoritma A*. Tabel *longsor_sekarang* digunakan untuk menyimpan informasi ruas jalan yang terkena dampak tanah longsor dan status dari kondisi jalan saat itu.

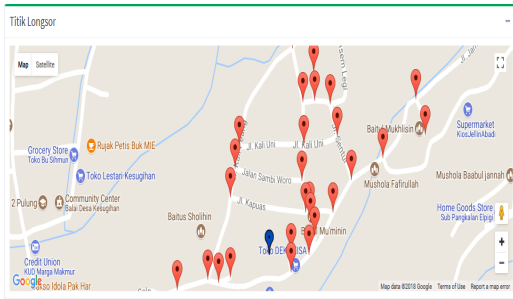
II. Hasil dan Pembahasan

Pengujian dilakukan pada sisi aplikasi web dimana admin memasukkan dan mengubah data titik jalan yang berdampak akibat longsor, dan pengujian pada sisi perangkat bergerak oleh pemakai yang memperoleh informasi rute alternatif menggunakan algoritma A*.

Uji Coba Aplikasi Web

Kejadian bencana tanah longsor terbaru diinputkan oleh admin. Admin

memasukkan posisi jalan yang terdampak longsor untuk dilakukan pencarian jalur oleh sistem. Titik-titik persimpangan jalan disimpan dalam tabel *m_nodes* dan ditampilkan pada *Google Map* seperti Gambar 3. Terdapat beberapa titik persimpangan jalan yang menghubungkan beberapa ruas jalan, yang fungsinya untuk menentukan letak terjadinya longsor di jalan.



Gambar 3. Titik-titik yang Menghubungkan Jalan

Data kejadian tanah longsor dimasukkan oleh admin dengan memasukkan dua titik yang membentuk jalan yang terjadi bencana longsor seperti Gambar 4.

Gambar 4. Form Kejadian Bencana Longsor

Gambar 4 berisi data jalan yang rusak parah akibat terjadi longsor di jalan kantor desa Serag. Input kejadian longsor dan perubahan status pada ruas jalan 1 antara titik 1 ke titik 2 akan menyebabkan kedua titik tersebut memiliki status beban yang sangat besar, dari 1 menjadi 10000 seperti Gambar 5(a) yang artinya jalan tersebut tidak bisa dilewati. Demikian juga jalan antara titik 2 dan 1 yang membentuk ruas jalan 46 akan memiliki status beban 10000 seperti Gambar 5(b). Kedua status tersebut akan menjadi input untuk proses pencarian jalur alternatif.

| id_lines | line_name | id_node_a | id_node_b | beban | status |
|----------|-----------|-----------|-----------|-------|--------|
| 1 | jalan 1 | 1 | 2 | 0 | 10000 |

(a)

| id_lines | line_name | id_node_a | id_node_b | beban | status |
|----------|-----------|-----------|-----------|-------|--------|
| 46 | jalan 46 | 2 | 1 | 0 | 10000 |

(b)

Gambar 5. Perubahan Status Beban oleh Sistem Akibat Input Lokasi Longsor oleh Admin (a) Dari Titik 1 ke Titik 2; (b) Dari Titik 2 ke Titik 1.

Admin dapat melakukan perbaikan data apabila jalan sudah diperbaiki melalui *action edit* pada *submenu Listing* data kejadian longsor dan mengubah status kejadian tanah longsor tersebut pada *form* kejadian tanah longsor seperti Gambar 6.

Gambar 6. Mengubah Data Jalan yang Sudah Diperbaiki

Perubahan data perbaikan jalan oleh admin tersebut dari status “belum terselesaikan” menjadi “sudah terselesaikan” oleh admin, secara otomatis akan mengubah status beban dari titik 1 ke titik 2 pada jalan 1 dan jalan 46 dari 10000 menjadi 1 kembali seperti Gambar 7.

| id_lines | line_name | id_node_a | id_node_b | beban | status |
|----------|-----------|-----------|-----------|-------|--------|
| 1 | jalan 1 | 1 | 2 | 0 | 1 |

(a)

| id_lines | line_name | id_node_a | id_node_b | beban | status |
|----------|-----------|-----------|-----------|-------|--------|
| 46 | jalan 46 | 2 | 1 | 0 | 1 |

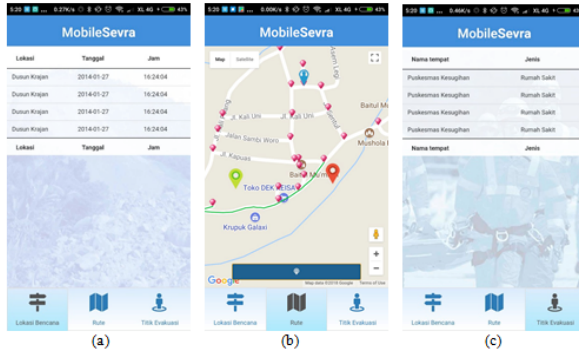
(b)

Gambar 7. Perubahan Status Jalan dari “Belum Terselesaikan” Menjadi “Sudah Terselesaikan” Otomatis Mengubah Status Beban Jalan (a) Dari Titik 1 ke Titik 2; (b) Dari Titik 2 ke Titik 1.

Uji Coba Aplikasi Perangkat Bergerak

Pada aplikasi perangkat bergerak, terdapat 3 menu yang dapat digunakan oleh pemakai, yaitu menu Lokasi Bencana, Rute dan Titik Evakuasi. Menu Lokasi Bencana memberikan informasi lokasi yang sedang terjadi bencana tanah longsor, menu Rute

memberikan informasi jalur kepada pemakai dan menu Titik Evakuasi memberikan informasi titik-titik evakuasi apabila terjadi bencana tanah longsor seperti Gambar 8.



Gambar 8. Menu pada Aplikasi Perangkat Bergerak (a) Lokasi Bencana; (b) Jalur Alternatif; (c) Titik Evakuasi.

Pencarian jalur alternatif jika terjadi longsor yang berdampak pada jalan raya dengan melakukan simulasi peristiwa longsor diantara titik 20 dan 23 yang membentuk ruas jalan 25 dan juga diantara titik sebaliknya 23 dan 20 yang membentuk ruas jalan 70 seperti Gambar 9. Status beban jalan ruas jalan 25 dan 70 secara otomatis akan berubah menjadi 10000.

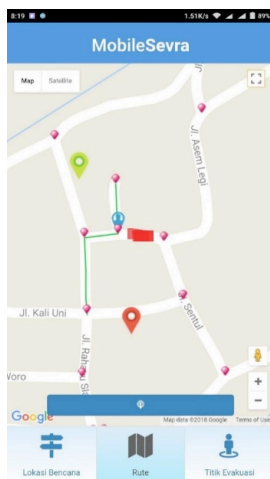
| id_lines | line_name | id_node_a | id_node_b | beban | status |
|----------|-----------|-----------|-----------|-------|--------|
| 25 | jalan 25 | 20 | 23 | 0 | 10000 |

(a)

| id_lines | line_name | id_node_a | id_node_b | beban | status |
|----------|-----------|-----------|-----------|-------|--------|
| 70 | jalan 70 | 23 | 20 | 0 | 10000 |

(b)

Gambar 9. Perubahan Status Beban Jalan Akibat Input Longsor antara Titik 20 dan 23 (a) Ruas Jalan 25; (b) Ruas Jalan 70.

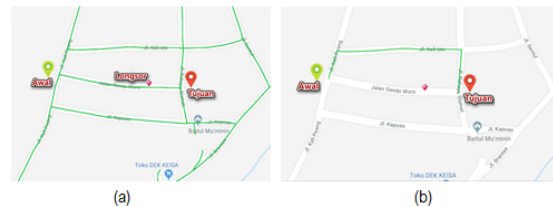


Gambar 10. Jalur Alternatif dengan Algoritma A*.

Apabila *user* akan melewati ruas jalan 25 atau 70, maka akan ditampilkan lokasi kejadian tanah longsor dan jalur alternatif yang dapat dilewati. Sistem melakukan proses perhitungan jarak terdekat menggunakan algoritma A* sesuai input yang dimasukkan oleh admin seperti terlihat pada Gambar 10. Arsiran berwarna merah menunjukkan lokasi bencana longsor, maka *user* tidak bisa melewati jalur tersebut, dan diarahkan ke jalur alternatif.

Analisa Hasil Jalur Alternatif dengan Algoritma A*

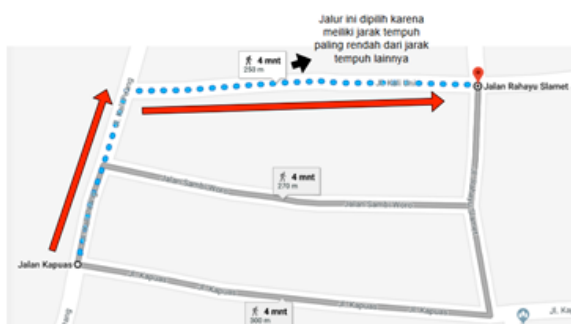
Analisa hasil pencarian jalur alternatif dilakukan dengan membandingkan hasil jalur alternatif dengan algoritma A* dengan hasil pencarian melalui *Google search*. Sebagai input akan dilakukan pencarian jalur dari titik awal ke tujuan seperti pada Gambar 11(a) dengan posisi lokasi longsor ditandai dengan tanda merah yang mengakibatkan jalan tertutup. Secara otomatis sistem akan mengalihkan jalur, dan memilih jalur yang tidak ada hambatan atau jalan tidak tertutup longsor seperti Gambar 11(b).



Gambar 11. Pencarian Jalur Alternatif dari Titik Awal ke Tujuan dengan Lokasi Longsor Ditandai dengan Titik Merah.

Hasil perhitungan dengan *Google search* dari input pada Gambar 11(a) menghasilkan total jarak tempuh sepanjang 243 meter seperti Gambar 12. Sedangkan dengan algoritma A* menghasilkan total jarak tempuh 244,7 meter atau terdapat perbedaan jarak 1,7 meter. Namun hasil yang diperoleh dari search dalam mencari jalur terpendek tidak memperhatikan lebar jalan dan jalan yang tertutup akibat bencana. Keunggulan aplikasi yang dikembangkan ini adalah dapat mencari jalur alternatif dengan algoritma A* berdasarkan informasi status jalan yang tertutup atau tidak dapat dilewati akibat tanah longsor yang dimasukkan oleh admin. Keunggulan dari algoritma A* dapat

memperoleh jalur dengan beban komputasi dan waktu simulasi yang kecil tanpa mempengaruhi hasil pencarian jalur terpendek. Aplikasi perangkat bergerak dan algoritma A* yang digunakan sangat tepat karena dapat menghasilkan jalur alternatif akibat bencana tanah longsor dengan mudah, cepat dan akurat. Sistem *client-server* yang dikembangkan menghasilkan informasi lokasi longsor, jalur alternatif dan tempat evakuasi sebagai sistem evakuasi bencana tanah longsor di Kabupaten Ponorogo. Admin memasukkan dan mengubah data melalui aplikasi web dan pemakai memperoleh informasi melalui perangkat bergerak.



Gambar 12 Hasil Pencarian Jalur Terpendek dengan Google Search

III. Simpulan

Aplikasi sistem evakuasi dikembangkan berbasis *client-server* dimana admin memasukkan dan mengubah data melalui aplikasi web dan pemakai memperoleh informasi melalui perangkat bergerak. Admin memberikan informasi status ruas jalan yang terdampak akibat tanah longsor, sedangkan pemakai mendapatkan informasi lokasi ruas jalan yang berdampak tanah longsor, jalur alternatif dan tempat evakuasi. Jalur alternatif dihasilkan dengan menggunakan algoritma A* yang mempunyai kelebihan beban komputasi dan waktu simulasi yang kecil menghasilkan dapat diakses dengan mudah, cepat dan akurat melalui perangkat bergerak.

IV. Daftar Pustaka

- [1] Ulhaq, M. J. D. Aplikasi Sistem Informasi Geografi Untuk Pemetaan Zona Ancaman Bahaya Longsoran (Landslide) Kecamatan Selogiri Kabupaten

Wonogiri. Dissertation. Universitas Muhammadiyah Surakarta. 2013.

- [2] Koenti, I. J. Diskresi dalam Penanggulangan Bencana di DIY dengan Paradigma Kontinjensi. *Jurnal Hukum IUS QUIA IUSTUM*. 2016; 23(3): 461-485.
- [3] Rais, R., & Arsy, R. F. Analisis Sistem Manajemen Resiko Bencana Dengan Menggunakan Sistem Informasi Geografi (Studi Kasus: Kec. Polobangkeng Utara Kab. Takalar). *Jurnal Ilmiah Matematika dan Terapan*. 2012; 7(1):43-53.
- [4] Purwanto, T. H., & Hadi, M. P. Pengembangan Prototipe Infrastruktur Data Spasial dan Sistem Informasi Geografis Untuk Manajemen Bencana Alam (Studi Kasus Tanggap Darurat Letusan Gunungapi Merapi di Daerah Istimewa Yogyakarta). Dissertation. Universitas Gadjah Mada. 2015.
- [5] Setiawan, R., Kurniadi, D., & Bunyamin, H. Perancangan Sistem Pengelolaan Penanggulangan Bencana Alam Garut Berbasis Sistem Informasi Geografis. *Jurnal Algoritma*. 2017; 14(2): 215-221.
- [6] Asgari, H. D., Pramono, B., Ransi, N., & Isnawaty, I. (2018, January). *Sistem Informasi Geografis Daerah Rawan Banjir Berbasis Web di Kota Kendari, Sulawesi Tenggara*. Proceedings of Seminar Nasional Riset Kuantitatif Terapan. Kendari. 2017; 1(1):106-112.
- [7] Ramadhan, T. E., Suprayogi, A., & Nugraha, A. L. (2017). Pemodelan Potensi Bencana Tanah Longsor Menggunakan Analisis Sig Di Kabupaten Semarang. *Jurnal Geodesi Undip*. 2017; 6(1): 118-127.
- [8] Musabbichin, L., & Yunitarini, R. (2015). Sistem Informasi Geografis (SIG) Penentuan Jalur Terpendek untuk Menghindari Daerah Rawan Banjir (Studi Kasus Propinsi Jawa Timur). *Jurnal Simantec*. 2015. 4(3):187-194.
- [9] Djojo, M. A., & Karyono, K. (2013). Pengukuran Beban Komputasi Algoritma Dijkstra, A*, dan Floyd-warshall pada perangkat android.

- ULTIMA Computing*. 2013; 5(1):13-17.
- [10] Wulandari, S. R., Purwanto, Y., & Irawan, B. *Evaluasi Algoritma Pencarian Jalur Pada Aplikasi e-iTRIP Guna Menentukan Rute Pariwisata Kota Bandung Berbasis Perangkat Mobile Android*. Proceedings of the Seminar Nasional Aplikasi Teknologi Informasi (SNATI). Yogyakarta. 2012.
- [11] Yamin, M., & Talai, M. B. *Aplikasi Pencarian Jalur Terpendek pada Rumah Sakit Umum Bahteramas Menggunakan Algoritma A*(A-Star)*. *Jurnal Informatika*. 2015; 9(2):1065-1078.

Halaman ini sengaja dikosongkan.