

Pencarian Lokasi Tempat Ibadah Terdekat di Kota Bengkulu Menggunakan Algoritma *Ant Colony Optimization* (ACO)

Searching for the Nearest Places of Worship in Bengkulu City Using Ant Colony Optimisation Algorithm

Agung Kharisma Hidayah^{1*}

Nur'aini²

Anisya Sonita³

Saparudin Saroni⁴

^{1,2,3}Teknik Informatika, Universitas Muhammadiyah Bengkulu, Indonesia

⁴Pendidikan Biologi Universitas Muhammadiyah Bengkulu, Indonesia

¹kharisma@umb.ac.id, ²nur889231@gmail.com, ³anisyasonita@umb.ac.id,

⁴saparudin@umb.ac.id

Riwayat Artikel:

Diterima : 1 September 2023
Direview : 14 September 2023
Disetujui : 21 September 2023
Terbit : 19 Desember 2023

Abstrak

Sebagai sebuah negara dengan penduduk yang memiliki ragam agama, antaranya yakni agama Islam, Kristen, Budha dan Hindu menjadikan Indonesia juga memiliki banyak tempat ibadah yang tersebar di seluruh penjuru daerah. Bengkulu merupakan salah satu daerah yang dikunjungi oleh banyak wisatawan, baik yang berasal dari lokal ataupun mancanegara. Oleh sebab itu fasilitas umum yang ada di Bengkulu sangat diperlukan, salah satu fasilitas yang dibutuhkan adalah tempat ibadah. Informasi mengenai tempat ibadah menjadi penting dan dibutuhkan oleh para wisatawan, khususnya informasi tentang rute terdekat yang bisa diambil menuju tempat ibadah tersebut. Penelitian ini dilakukan agar dapat membangun sebuah sistem yang bisa memberikan informasi lokasi terdekat tempat ibadah serta petunjuk arah yang dapat digunakan sebagai panduan menuju lokasi tersebut dengan mengimplementasikan Algoritma Ant Colony Optimization untuk pencarian rute terdekatnya. Aplikasi yang dibuat menggunakan bahasa pemrograman PHP dan database MySQL ini dapat digunakan dengan berbagai macam perangkat karena aplikasi ini berbasis website. Penerapan algoritma Ant Colony Optimization menghasilkan rute yang memiliki jarak terdekat menuju lokasi tempat ibadah.

Kata Kunci: Ant Colony Optimization; MySQL; Pencarian lokasi; PHP; Tempat Ibadah.

Abstract

As a country with a population that has various religions, including Islam, Christianity, Buddhism and Hinduism, Indonesia also has many places of worship spread throughout the region. Bengkulu is an area visited by many tourists, both local and foreign. Therefore, public facilities in Bengkulu are very necessary, one of the facilities needed is a place of worship. Information about places of worship is important and needed by tourists, especially information about the nearest route that can be taken to the place of worship. This research was carried out in order to build a system that can provide information on the nearest places of worship as well as directions that can be used as a guide to that location by implementing the Ant Colony Optimization Algorithm to find the nearest route. This application, which was created using the PHP programming language and MySQL database, can be used with various devices because this application is website-based. The application of the Ant Colony Optimization algorithm produces a route that has the shortest distance to the location of the place of worship.

Keywords: Ant Colony Optimization; MySQL; Location Search; PHP; Worship Place.

1. Pendahuluan

Sistem informasi geografis merupakan media informasi yang sangat dibutuhkan dalam bertransportasi [1]. Dengan menggunakan Sistem Informasi Geografis, seseorang dapat memperoleh peta lokasi dalam bentuk spasial terkait dengan bumi. Sebagai contoh, ketika seseorang mengalami kesulitan dalam menemukan sebuah tempat saat dalam perjalanan, Sistem Informasi Geografis dapat digunakan untuk menemukan serta menampilkan peta yang berisi lokasi yang dicari [2][3].

Diantara teknologi Sistem Informasi Geografis yang sedang mengalami perkembangan saat ini adalah pencarian lokasi menggunakan rute terdekat [4]. Banyak orang memilih rute terdekat karena dapat menghemat waktu perjalanan mereka ke tempat yang dituju. Umumnya, mereka yang mencari rute terdekat adalah orang-orang yang sedang berwisata baik dari lokal ataupun dari mancanegara, yang sedang mencari lokasi wisata atau mungkin hanya ingin singgah sejenak dan mencari tempat yang bisa digunakan untuk beristirahat. Bahkan, ada juga yang ingin mencari lokasi untuk beribadah. Daerah seperti Bengkulu memiliki banyak area penting yang bisa dikunjungi wisatawan. Dikarenakan keberlimpahan kunjungan wisatawan dengan tujuan yang beragam, kota Bengkulu menjadi ramai dikunjungi. Banyak dari mereka yang datang mencari lokasi beribadah dalam kegiatan mereka. Di kota Bengkulu, terdapat berbagai tempat peribadatan bagi umat beragama. Misalnya, terdapat masjid bagi umat muslim, gereja bagi umat kristiani, wihara bagi umat budha, dan pura bagi umat hindu.

Mencari tempat ibadah merupakan hal yang penting bagi wisatawan yang berkunjung ke Kota Bengkulu. Menggunakan sistem pencarian lokasi yang tersedia, wisatawan akan lebih mudah menemukan jalan menuju tempat-tempat ibadah tersebut. Aplikasi ini menyediakan peta lengkap dengan rute yang dapat diikuti untuk mencapai lokasi yang diinginkan.

Banyak Algoritma yang telah dikembangkan hingga sekarang ini untuk membantu para wisatawan dalam mencari lokasi dengan rute terdekat. Algoritma Ant Colony Optimization adalah sebuah metode heuristik yang digunakan untuk memecahkan masalah optimasi dengan memodelkan perilaku koloni semut. [2][5]. Sebagai salah satu metode pencarian jalur terpendek [6], Algoritma Pencarian Jejak Semut ini memiliki keunikan yang membedakannya dari metode pencarian jalur terdekat lainnya. Metode ini terinspirasi dari perilaku semut yang melakukan perjalanan dari sarang menuju sumber makanan untuk menemukan makanan, di mana semut meninggalkan bekas langkah kaki sebagai petunjuk bagi semut lainnya [7].

Pada penelitian yang berjudul "Pencarian Tempat Ibadah Terdekat Menggunakan Algoritma Dijkstra" dilakukan pencarian Tempat Ibadah menggunakan algoritma Dijkstra [8], berbeda dengan penelitian tersebut pada penelitian ini pencarian dilakukan menggunakan algoritma ACO. Pada penelitian lainnya yang berjudul "Implementasi Algoritma Ant Colony Optimization pada Aplikasi Pencarian Lokasi Tempat Ibadah Terdekat di Kota Bandung" juga dilakukan pencarian tempat ibadah menggunakan algoritma ACO [2]. Penelitian ini dilakukan pada tempat penelitian yang berbeda dengan dengan penelitian terdahulu serta beberapa pengembangan pada sisi fitur aplikasi, diantaranya yaitu menambahkan informasi-informasi penting pada keterangan tempat ibadah yang terdapat pada aplikasi. Oleh karenanya, penulis melakukan suatu penelitian dengan judul " Pencarian lokasi tempat ibadah terdekat menggunakan algoritma Ant Colony Optization (ACO)"

2. Metode Penelitian

Pengumpulan Data

Di bawah ini adalah metode yang dipakai untuk mengumpulkan data yang dibutuhkan dalam penelitian:

a. Studi Pustaka.

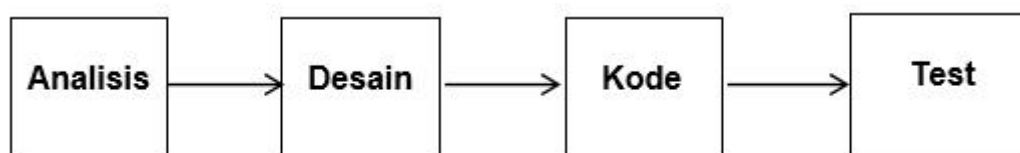
Dalam metode ini berbagai data serta informasi yang berkaitan dengan penelitian dikumpulkan melalui buku, artikel, internet, dan website serta data utama yang diperoleh dengan pengukuran rute lokasi yang dibutuhkan secara langsung menggunakan Global Positioning System (GPS) yang tersedia di gadget.

b. Pengamatan (Observation).

Survei lokasi dilakukan secara langsung dengan melibatkan pengumpulan data di seluruh lokasi tempat ibadah Kota Bengkulu .

Model Pengembangan Sistem

Penelitian ini dilakukan dengan mengembangkan sistem menggunakan pendekatan pemodelan incremental dikarenakan model ini cocok untuk proyek yang memiliki skala kecil. Model incremental dalam rekayasa perangkat lunak akan menghentikan proses pengembangan apabila produk yang dibuat telah mencapai semua fungsi yang diinginkan [9]. Model ini terdiri dari tahapan analisis, analisis desain, kode dan pengujian/testing.



Gambar 1. Model Incremental [9]

Analisa Kebutuhan

Berikut ini adalah cara yang digunakan untuk menganalisa kebutuhan:

1) Analisis Kebutuhan Fungsional

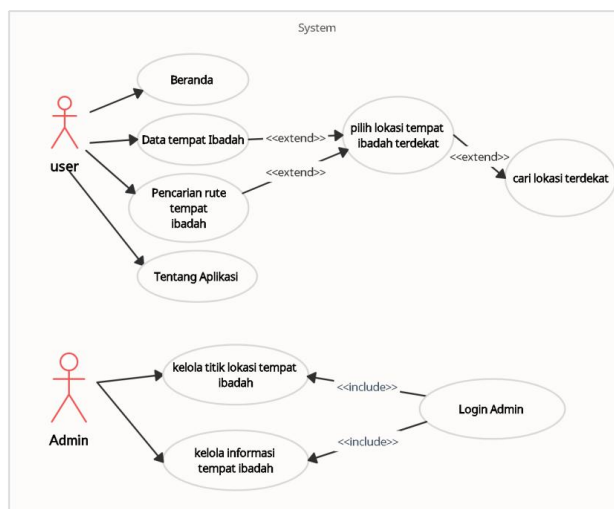
Kebutuhan fungsional memuat segala informasi yang harus ada dan dihasilkan oleh sistem. Aplikasi pencarian lokasi tempat ibadah dengan algoritma ant colony optimization ini diharapkan mampu menunjukkan semua lokasi tempat ibadah yang ada di kota Bengkulu serta dapat memberi rute terdekat kepada pengguna.

2) Analisis Kebutuhan Non Fungsional

Analisa ini merupakan jenis analisis kebutuhan yang mencakup karakteristik perilaku suatu sistem, meliputi kebutuhan perangkat keras (hardware), kebutuhan perangkat lunak (software), dan kebutuhan sumber daya manusia (brainware).

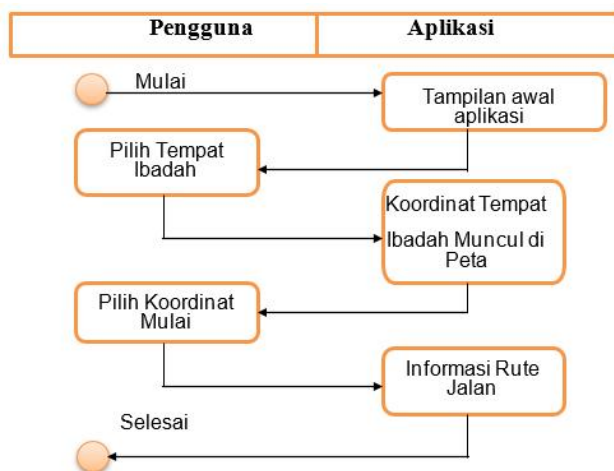
Perancangan Sistem

Diagram use case dapat digunakan untuk mengkompilasi persyaratan sistem, mengkomunikasikan desain dengan pelanggan, dan merancang uji coba untuk semua fungsionalitas sistem. Berikut ini merupakan *use case diagram* yang digunakan dalam penelitian ini:



Gambar 4. Use Case Diagram

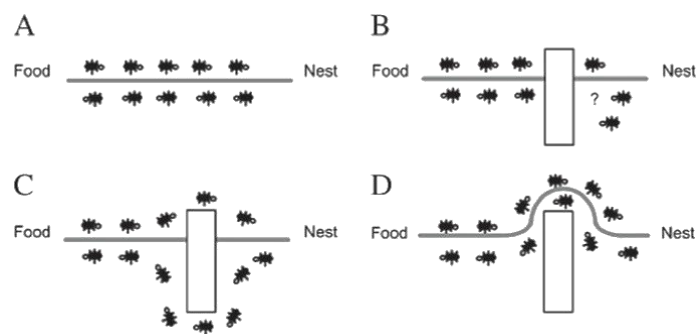
Diagram aktivitas atau activity diagram mengilustrasikan aliran fungsionalitas sistem. Berikut diagram activity yang terjadi pada penelitian ini:



Gambar 5. Activity Diagram

Penerapan Algoritma Ant Colony Optimization

Langkah penerapan metode yang diusulkan untuk penelitian ini adalah algoritma *Ant Colony Optimization*. Koloni semut merupakan suatu metode heuristik yang digunakan untuk mencari solusi optimal dalam masalah optimasi [3]. Ide berdasar algoritma ini berasal dari interaksi semut di koloni mereka dalam mencari dan menemukan sumber makanan. Semut dapat menemukan rute terpendek dari sumber makanan ke sarang mereka tanpa perlu melihatnya secara langsung. Kawan semut memiliki cara komunikasi dan solusi yang inovatif yaitu menggunakan jejak pheromone pada jalur tertentu. Semakin banyak jejak pheromone yang ditinggalkan, semakin banyak semut lain yang akan mengikuti jalur tersebut [10].

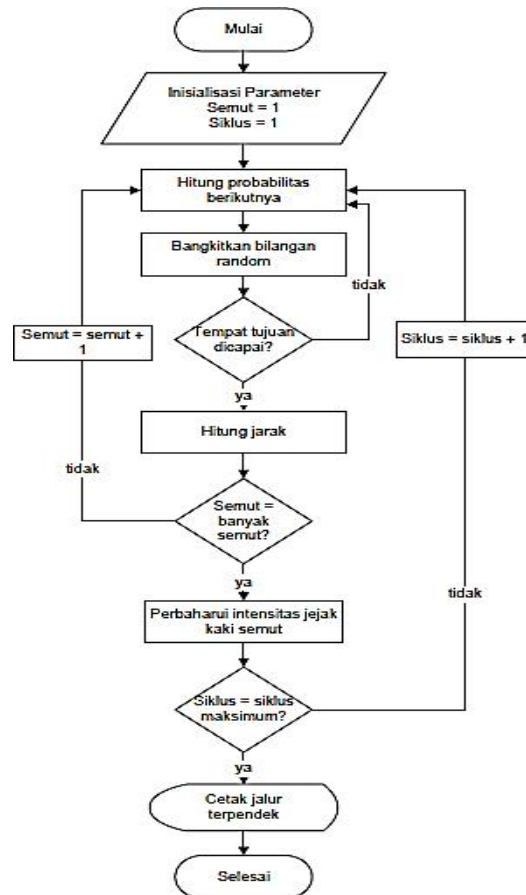


Gambar 2. Model Algoritma Ant Colony Optimization [2]

Pada gambar 2. Seperti terlihat, Gambar A mewakili perjalanan koloni semut mencari jalur tercepat dari sarang menuju sumber makanan. Dua kelompok yang berbeda tersebut melakukan perjalanan. Sekelompok semut bernama L bergerak dari kiri ke kanan, sedangkan sekelompok semut bernama R bergerak dari arah sebaliknya. Kedua kelompok memulai dari titik yang sama dan jatuh ke dalam situasi harus memilih jalan mana yang harus diambil. Grup L dibagi menjadi dua grup tambahan. Ada yang melewati jalan ke atas dan ada pula yang melewati jalan ke bawah.

Gambar B dan Gambar C menunjukkan sekelompok semut yang bergerak dengan kecepatan yang sama sambil meninggalkan jejak pheromone di jalurnya. Aroma yang ditinggalkan semut yang berjalan di jalur atas berkurang karena jumlah semut yang melewati jalur atas lebih sedikit dibandingkan jumlah semut yang melewati jalur bawah. Hal ini terjadi karena jarak yang harus ditempuh lebih jauh dibandingkan menggunakan jalan di bawahnya. Sementara itu, pheromone yang terletak di bagian bawah evaporasinya cenderung memiliki masa yang lebih lama. Dikarenakan jumlah semut yang melintasi jalur di bawah melebihi jumlah semut yang melintasi jalur di atas.

Gambar D Dalam akhirnya, semut-semuts lain memilih untuk melewati jalur bawah karena masih terdapat banyak pheromone yang ditinggalkan di sana. Sebaliknya, pheromone di jalur atas telah banyak menguap sehingga semut-semut tersebut tidak memilih jalur tersebut. Ketika bertambah banyak semut yang melintasi jalan, bertambah banyak semut yang mengikuti jalur yang sama. Namun, jika jumlah semut di jalur lebih sedikit, jumlah sisa pheromone akan berkurang dan bahkan mungkin hilang. Mulai saat ini, rute tercepat dari tempat tinggal Anda ke tempat Anda membeli makanan akan dipilih. Untuk lebih jelasnya lihat Gambar 3 dibawah ini:



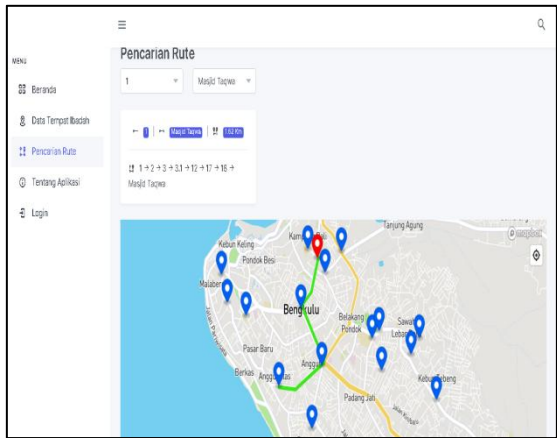
Gambar 3. Flowchart

Metode Pengujian Sistem

Sistem pengujian yang digunakan pada penelitian ini adalah BlackBox Testing. Pengujian BlackBox memungkinkan perekayasa perangkat lunak memperoleh serangkaian kondisi masukan yang sepenuhnya memanfaatkan semua persyaratan fungsional program. Pengujian ini dilakukan dengan cara memeriksa fungsionalitas sistem apakah berfungsi sesuai dengan seharusnya.

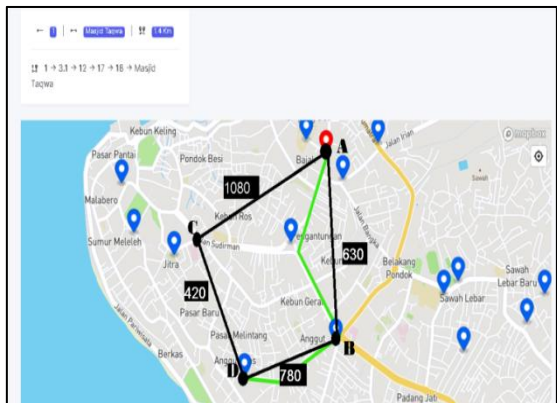
3. Hasil dan Pembahasan

Penelitian mengenai pencarian rute tempat ibadah terdekat ini menghasilkan pencarian rute dengan menggunakan algoritma Ant Colony Optimization dalam pencarian rute terdekatnya. Berikut adalah tampilan aplikasi Pencarian rute tempat ibadah terdekat menggunakan algoritma *Ant Colony Optimization*



Gambar 6. Tampilan Aplikasi

Penerapan algoritma dalam menentukan jalur terpendek pada tempat ibadah di kota Bengkulu dapat dilihat pada gambar 7. Sebagai contoh peta jalur terpendek dari Jalan Bali menuju lokasi masjid At-Taqwa yang berada di Jalan Soekarno Hatta Kota Bengkulu.



Gambar 7. Contoh peta lokasi

Dengan jarak antar lokasi (d_{ij}) secara rinci dapat dilihat pada tabel berikut:

Tabel 1. Jarak Antar Lokasi

	A	B	C	D
A	0	630	1080	-
B	630	0	-	780
C	1080	-	0	420
D	-	780	420	0

Ada beberapa langkah yang dapat diikuti untuk menghitung rute terbaik dari titik A ke titik D dengan menggunakan Algoritma Ant Colony Optimization. Langkah awal yang perlu dilakukan adalah memulai dengan mengatur nilai-nilai parameter algoritma Ant Colony Optimization, termasuk di antaranya yaitu:

$$Q = 0.1$$

$$\tau_{ij} = 0.01$$

$\alpha = 0,1$
 $\beta = 1$
 $\rho = 0.5$
 $m = 2$
 $n = 4$
 $NC_{max} = 1$

Melakukan perhitungan visibilitas point-to-point sebagai sarana menginformasikan kualitas suatu edges, dengan menggunakan jarak titik yang diketahui, visibilitas point-to-point dapat dihitung dengan menggunakan rumus $n = (1 / d^{ij})$

Tabel 2. Visibilitas Antar Titik

	A	B	C	D
A	0	0,001587	0,000925	-
B	0,001587	0	-	0,001282
C	0,000925	-	0	0,002380
D	-	0,001282	0,002380	0

Awalnya, ada dua semut yang memulai perjalanan dari lokasi yang sama. Kedua semut tersebut kemudian mengunjungi titik-titik lain sesuai dengan jalur yang tersedia dengan akhir tujuan mereka adalah titik D. Namun, tidak semua titik akan mereka lewati dalam perjalanan mereka. Bergerak dari satu titik ke titik lainnya menggunakan jalur tertentu yang telah disiapkan oleh koloni semut dengan cara:

- Pilih lokasi yang dikunjungi secara acak
- Simpan lokasi yang dikunjungi ke memori

Tempat pertama yang didatangi oleh semut adalah tempat awal yakni titik A. Titik awal akan dipelihara dalam ingatan setiap semut, kemudian semut akan berkunjung ke titik berikutnya. Proses dilakukan semut pada iterasi pertama adalah :

Visibilitas measure, $\eta (r.)$. Dapat dihitung dengan rumus :

$$\Sigma [\tau(r,u)][\eta (r,s)]\beta \tag{1}$$

Hasil perhitungan visibilitas measure menggunakan rumus (1) adalah 0,00002512

Mengunjungi titik ke-2

Tabel 3. Probabilitas Kunjungan ke Titik 2

Semut	Titik		Probabilitas				J	Rute
	Awal	A	B	C	D			
S1	A	0	0,001587	0,000925	0	B	[A,B]	
S2	A	0	0,001587	0,000925	0	C	[A,C]	

Jika titik tersebut telah tercatat di memori atau tidak terdapat jalur koneksi, kemungkinan bahwa titik tersebut memiliki nilai nol. Dari data yang terdapat dalam Tabel 3, dapat

disimpulkan bahwa sebagian besar semut cenderung memilih titik yang memiliki probabilitas yang paling tinggi. Namun, tidak jarang ada semut yang memilih titik dengan probabilitas yang rendah, yang menunjukkan bahwa semut sebenarnya memilih titik secara acak. Melakukan pengisian daftar titik AB, AC menggunakan persamaan berikut secara lokal:

$$\tau(r,s) \leftarrow (1-\rho) \cdot \tau(r,s) + \rho \cdot \tau_0 \quad (2)$$

Keterangan :

Pengaturan pelepasan pheromone untuk mencegah pheromone menumpuk secara tak terbatas karena jumlahnya akan terus meningkat setiap kali iterasi, dengan ketentuan: $(\eta \cdot L_{nn})^{-1}$ dimana L_{nn} menyatakan jarak antara titik r ke s, η merupakan jumlah node.

$\tau(A,B)$ menggunakan rumus (2) menghasilkan 0,006587, sedangkan $\tau(A,C)$ menggunakan rumus (2) menghasilkan 0,0065925. Menghitung visibilitas measure $\eta(r,s)$ menggunakan rumus (1) pada semut 1 menghasilkan 0,0000084445 dan semut 2 menghasilkan 0,0000141015.

Mengunjungi titik ke-3

Tabel 4. Probabilitas Kunjungan ke titik 3

Semut	Titik Awal	Probabilitas				J	Rute
		A	B	C	D		
S1	B	0,001587	0	-	0,001282	D	[A,B,D]
S2	C	0,000925	-	0	0,002380	D	[A,C,D]

Menghitung Pembaruan lokal $\tau(B, D)$ menggunakan rumus (2) menghasilkan 0,006282, $\tau(C, D)$ menggunakan rumus (2) menghasilkan 0,00738. Menghitung visibilitas measure $\eta(r, s)$ menggunakan rumur (1) pada semut 1 menghasilkan 0,0000080535 dan pada semut 2 menghasilkan 0,0000175644.

Sampai pada proses ini, semut S1 dan S2 sudah tiba pada titik yang dituju yaitu titik D, namun semut S1 lebih cepat diakses dikarenakan jarak yang ditemukan paling dekat. Hasil Perhitungan ACO sama dengan yang ditampilkan pada aplikasi seperti yang terlihat pada gambar 7. Garis berwarna hijau pada gambar 7 merupakan rekomendasi jalur terpendek yang bisa dilewati pengguna dari titik awal menuju titik yang dituju. Garis tersebut merupakan jalur terpendek dari titik A menuju D seperti yang ditunjukkan dari hasil perhitungan manual algoritma ACO. Dengan demikian dapat dilihat bahwa hasil perhitungan manual Algoritma ACO sama dengan hasil yang ditunjukkan melalui aplikasi yang dibangun.

Hasil Pengujian

Tabel 5. Uji Coba fungsional user

No	Menu	Pengujian
1	Beranda	Menampilkan Jumlah data tempat ibadah, simpul, dan graph
2	Data Tempat Ibadah	Menampilkan semua informasi tempat Ibadah yang terdapat dari aplikasi

3	pencarian rute	Menampilkan pencarian rute awal dan rute tujuan. Dan menghasilkan rute terdekat dari perhitungan otomatis algoritma ant colony optimization.
4	tentang aplikasi	Menampilkan penjelasan algoritma yang digunakan pada aplikasi pencarian rute terdekat.

Tabel 6. Uji Coba fungsional Admin

No	Menu	Pengujian
1	Beranda	Menampilkan Jumlah data tempat ibadah, simpul, dan graph
2	Data Tempat Ibadah	Menampilkan data tempat ibadah, pada menu ini admin dapat melakukan penambahan, mengubah, dan menghapus data tempat ibadah.
3	Titik semut	Menampilkan titik-titik jalur yang akan dilewati. Pada menu ini admin dapat melakukan penambahan, mengubah dan menghapus data node. Menampilkan jarak jalur yang dilewati. Pada menu
4	graph	graph admin dapat menambah, mengedit, dan menghapus data graph. Menampilkan data admin yang dapat mengakses ke
5	Users	aplikasi. Pada menu users ini admin dapat menambah, mengedit dan menghapus data user admin.

4. Penutup

Kesimpulan

Berdasarkan hasil penelitian dari menerapkan algoritma semut dalam kasus pencarian tempat ibadah yang ada di Bengkulu dapat disimpulkan sebagai berikut:

1. Algoritma Ant Colony Optimization (ACO) dapat diterapkan pada sistem pencarian tempat ibadah.
2. Algoritma Ant Colony Optimizatin (ACO) dapat menentukan rute terpendek.
3. Aplikasi pencarian lokasi tempat ibadah dengan rute terpendek dapat Membantu pengguna aplikasi untuk menemukan jalur terpendek.

Saran

Aplikasi pencarian lokasi tempat ibadah menggunakan algoritma Ant Colony Optimization ini masih banyak terdapat kekurangannya terutama dari segi tampilan dan data. Selanjutnya agar dapat dilakukan penelitian menggunakan algoritma yang berbeda agar dapat dibandingkan dengan algoritma ACO dan mendapatkan algoritma terbaik yang dapat digunakan untuk pencarian tempat ibadah. Kemudian data yang digunakan untuk pengujian juga masih relatif sedikit sehingga diharapkan adanya penelitian yang menggunakan data yang lebih banyak untuk menguji penggunaan algoritma ACO ini. Lalu tampilan aplikasi ini juga dapat dikembangkan agar lebih menarik.

5. Referensi

- [1] M. S. Lauryn and M. Ibrohim, "Sistem Informasi Geografis Tingkat Kerusakan Ruas Jalan Berbasis Web," *JSiI (Jurnal Sist. Informasi)*, vol. 6, no. 1, p. 20, 2019, doi: 10.30656/jsii.v6i1.1022.
- [2] A. Zarman, M. Irfan, and W. Uriawan, "Implementasi Algoritma Ant Colony Optimization

- pada Aplikasi Pencarian Lokasi Tempat Ibadah Terdekat di Kota Bandung,” *J. Online Inform.*, vol. 1, no. 1, p. 6, 2016, doi: 10.15575/join.v1i1.4.
- [3] M. Dorigo and L. M. Gambardella, “Ant colony system: A cooperative learning approach to the traveling salesman problem,” *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 53–66, 1997, doi: 10.1109/4235.585892.
- [4] A. Gusmão and S. H. Pramono, “Sistem Informasi Geografis Pariwisata Berbasis Web Dan Pencarian Jalur Terpendek Dengan Algoritma Dijkstra,” vol. 7, no. 2, pp. 125–130, 2013.
- [5] A. Hidayat, I. Purnamasari, and D. M. Siringoringo, “Penentuan Jalur Terpendek dengan Metode Heuristik Menggunakan Algoritma Sarang Semut (Ant Colony) (Studi Kasus: Jalan Arteri Sekunder Kota Samarinda) Determination of the Shortest Path with the Heuristic Method Using the Ant Colony Algorithm (Case Study: ,” *J. EKSPONENSIAL*, vol. 11, no. 1, pp. 93–98, 2020.
- [6] S. H. Ok, W. J. Seo, J. H. Ahn, S. Kang, and B. Moon, “An ant colony optimization approach for the preference-based shortest path search,” *J. Chinese Inst. Eng. Trans. Chinese Inst. Eng. A*, vol. 34, no. 2, pp. 181–196, 2011, doi: 10.1080/02533839.2011.565574.
- [7] A. P. Wibowo and A. Ilyas, “Integrasi Algoritma Ant Colony Pada Pencarian Lokasi,” *Ic-Tech*, vol. 13, no. 2, pp. 43–48, 2018, [Online]. Available: <http://ejournal.stmik-wp.ac.id>
- [8] B. lutfi A. Rofiq, H. Februariyanti, J. S. Wibowo, and Z. Budiarmo, “Pencarian Tempat Ibadah Terdekat Menggunakan Algoritma Dijkstra,” *JUSIM (Jurnal Sist. Inf. Musirawas)*, vol. 6, no. 1, pp. 19–28, 2021, doi: 10.32767/jusim.v6i1.1138.
- [9] S. Fernandez, C. Prihantoro, and A. K. Hidayah, “Implementasi Weighted Product Pada Pemilihan Dosen Terbaik di Universitas Muhammadiyah Bengkulu,” *J. Pseudocode*, vol. VIII, no. 2, pp. 126–133, 2021.
- [10] A. Gupta and S. Srivastava, “Comparative Analysis of Ant Colony and Particle Swarm Optimization Algorithms for Distance Optimization,” *Procedia Comput. Sci.*, vol. 173, no. 2019, pp. 245–253, 2020, doi: 10.1016/j.procs.2020.06.029.