

PENGEMBANGAN APLIKASI PENYUSUNAN RUTE PERJALANAN WISATA JAKARTA BERBASIS WEBSITE DENGAN METODE ALGORITMA *ANT COLONY*

Nenny Anggraini¹, Ire Puspa Wardhani², Susi Widayati³, Wafa Sabrina⁴

^{1,4}Fakultas Sains dan Teknologi, UIN Syarif Hidayatullah Jakarta, Indonesia

^{2,3}Sekolah Tinggi Manajemen Informatika dan Komputer Jakarta STI&K, Indonesia

Email: ¹Nenny.anggraini@uinjkt.ac.id, ²irepuspa@gmail.com, ³widayatisusi@gmail.com,

⁴officialwafa@gmail.com

SEJARAH ARTIKEL

Diterima: 18.06.2025

Direvisi: 11.07.2025

Diterbitkan: 23.07.2025



Hak Cipta © 2025
Penulis: Ini adalah
artikel akses terbuka
yang didistribusikan
berdasarkan ketentuan
Creative Commons
Attribution 4.0
International License.

ABSTRAK

Jakarta merupakan kota yang memiliki trafik padat, masalah rute perjalanan di Jakarta menjadi perhatian tersendiri bagi pengguna jalan. Pencarian jalur terpendek dalam jaringan transportasi di Jakarta seperti mengurangi kemacetan lalu lintas, rekomendasi rute terpendek untuk perjalanan dinas dan pengiriman barang atau mencapai lokasi wisata. Penelitian tentang Aplikasi penyusunan rute perjalanan wisata dengan metode algoritma *ant colony* ini merupakan penelitian yang ingin menyelesaikan permasalahan lamanya waktu menyusun rute perjalanan wisata. Data yang diperoleh dari pengumpulan data melalui wawancara dengan pengusaha tour dan travel yaitu diperlukannya waktu minimal satu hari untuk menyusun rute perjalanan wisata. Sebagai salah satu solusi perlu aplikasi berbasis web yang dapat menyusun rute perjalanan wisata berdasarkan jarak terpendek. Aplikasi ini merupakan aplikasi berbasis website, dengan input data objek wisata. Metode pengembangan aplikasi menggunakan RAD dan algoritma *ant colony* untuk proses penyusunan rute perjalanan wisata terpendek yang terdiri atas 2 (dua) user. Output berupa aplikasi penyusunan rute perjalanan wisatayang menyusun rute perjalanan wisata terpendek.

Kata Kunci: pengembangan aplikasi, website, algoritma *ant colony*, rute, perjalanan wisata.

ABSTRACT

Jakarta is a city with dense traffic, the problem of travel routes in Jakarta is a special concern for road users. Finding the shortest route in the transportation network in Jakarta such as reducing traffic congestion, recommendations for the shortest route for business trips and delivery of goods or reaching tourist locations. Research on the Application of arranging tourist routes using the ant colony algorithm method is a study that wants to solve the problem of the length of time to arrange tourist routes. Data obtained from data collection through interviews with tour and travel entrepreneurs, namely the minimum time required to arrange a tourist route. As one solution, a web-based application is needed that can arrange tourist routes based on the shortest distance. This application is a website-based application, with input of tourist object data. The application development method uses RAD and the ant colony algorithm for the process of arranging the shortest tourist route consisting of 2 (two) users. The output is a tourist route arrangement application that arranges the shortest tourist route.

Keywords: application development, website, ant colony algorithm, route, travel.

1. PENDAHULUAN

Sebagai negara kepulauan terbesar di dunia, Indonesia memiliki banyak objek wisata yang menarik tersebar dari Kota Sabang hingga Merauke. Jakarta sebagai Ibukota Negara Republik Indonesia merupakan salah satu dari tiga tujuan utama wisatawan mancanegara ke Indonesia. Beragam jenis objek wisata terdapat di Jakarta, seperti terlihat pada Tabel 1 berikut ini:

Tabel 1. Sampel Objek wisata Jakarta

(sumber : www.google.com)

No	Nama objek wisata	Alamat
1.	Taman Mini Indonesia Indah	Jl. Taman Mini Indonesia Indah, Ceger, Kec. Cipayung, Kota Jakarta Timur, Daerah Khusus Ibukota Jakarta 13820
2.	Wisata Kota Tua Jakarta	Kawasan Kota Tua, Taman Fatahillah No.1 7, RT.7/RW.7, Pinangsia, Kec. Taman Sari, Kota Jakarta Barat, Daerah Khusus Ibukota Jakarta 11110
3.	Taman Wisata Alam Angke	Jl. Mualim Teko Jl. Pantai Indah Kapuk, Kamal Muara, Kec. Penjaringan, Jkt Utara, Daerah Khusus Ibukota Jakarta 14470
4.	Taman Impian Jaya Ancol	Jl. Lodan Timur No.7, RT.14/RW.10, Ancol, Kec. Pademangan, Jkt Utara, Daerah Khusus Ibukota Jakarta 14430
5.	Suropati Park	Jl. Taman Suropati No.5, RT.5/RW.5, Menteng, Kec. Menteng, Kota Jakarta Pusat, Daerah Khusus Ibukota Jakarta 10310
6.	Jakarta AQuarium Safari	Letjen S. Parman St No.106, RT.3/RW.3, South Tanjung Duren, Grogol petamburan, West Jakarta City, Jakarta 11470
7.	Taman Pandang Istana	Jalan Silang Monas Barat Laut, RT.5/RW.2, Gambir, RT.5/RW.2, Gambir, Kecamatan Gambir, Kota Jakarta Pusat, Daerah Khusus Ibukota Jakarta 10110

Berdasarkan hasil wawancara dengan Direktur Utama dari PT Kirana Surya Gemilang sebuah perusahaan Tour dan Travel, diketahui bahwa dalam sebuah perjalanan wisata, penyusunan rute perjalanan wisata bagi wisatawan atau perencanaan rute bagi biro perjalanan merupakan salah satu hal yang penting karena berguna untuk mengetahui susunan acara atau tahap-tahap dari pelaksanaan tour itu sendiri. Permasalahan dalam penyusunan rute perjalanan wisata di PT Kirana Surya Gemilang adalah memakan waktu selama sehari, hal tersebut pun bergantung dengan ketersediaan database. Selain itu untuk membuat rute perjalanan yang baik juga dilakukan survey atau uji coba dari rute perjalanan sebelum akhirnya bisa diberitahukan kepada klien. Proses survey tersebut tentunya memakan waktu dan sangat tidak memungkinkan bagi wisatawan independent untuk melakukan hal serupa seperti yang dilakukan PT Kirana Surya Gemilang.

Untuk mendukung aplikasi penyusunan rute perjalanan wisata yang optimal dibutuhkan suatu algoritma yang mendukung. Algoritma *ant colony* diperkenalkan pada tahun 1991 oleh Marco Dorigo. Sesuai dengan namanya, Algoritma *ant colony* terinspirasi dari perilaku *ant colony* dalam mencari makan dimana semut semut dalam koloni dapat menemukan jarak terpendek dari sarang menuju sumber makanan. Algoritma *ant colony* diusulkan karena dapat menyelesaikan permasalahan optimasi kombinatorial [1], [2]. Dengan menggunakan algoritma *ant colony* maka dimungkinkan untuk mendapatkan susunan rute perjalanan wisata terpendek.

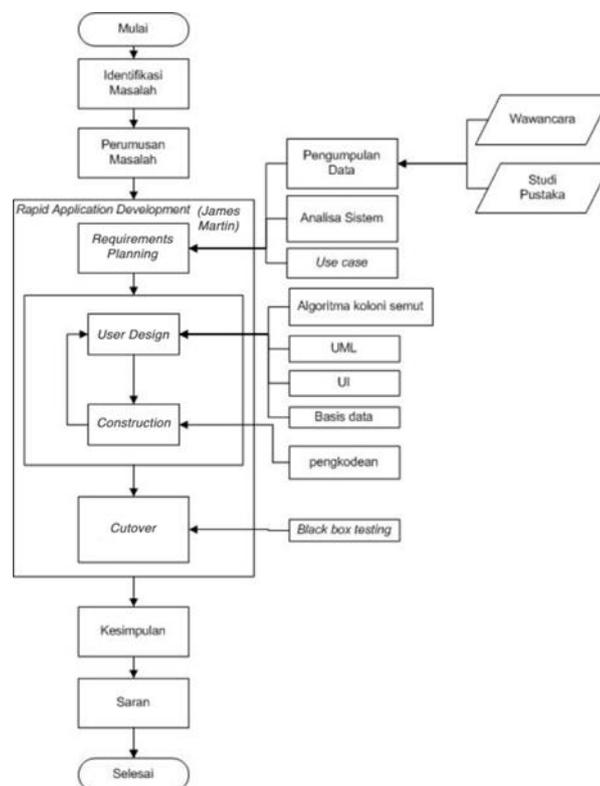
Penelitian [1] Hasil penelitian ini menunjukkan bahwa penerapan Algoritma *Ant colony* System (ACS) berhasil menentukan rute distribusi produk air mineral di Kota Palu dengan lebih baik, menghasilkan rute terbaik dengan bobot jarak 10,24 KM. Algoritma ini terbukti dapat menghasilkan rute dengan jarak terbaik meskipun waktu eksekusi dipengaruhi oleh ukuran kasus. Namun, kekurangan penelitian ini antara lain hasil yang diperoleh bervariasi setiap kali pencarian rute dilakukan karena bersifat probabilitas, dan waktu eksekusi cenderung lama terutama untuk kasus dengan banyak titik. Selain itu, hasil algoritma sangat bergantung pada penentuan nilai parameter ACS seperti β , q_0 , q , α , ρ , jumlah semut, dan jumlah siklus, yang memerlukan analisis khusus untuk setiap kasus TSP. Dan [3] Penelitian ini berhasil menerapkan Algoritma *Ant colony Optimization* (ACO) dalam bentuk aplikasi web menggunakan PHP untuk pencarian rute wisata terpendek di Kota Palembang. Hasil pengujian menunjukkan bahwa algoritma dapat menghasilkan jalur terpendek, seperti rute dimulai dari Pundi Kayu dengan nilai total sebesar 205.12025621393. Kelemahan penelitian ini adalah kurangnya analisis mendalam terhadap pengaruh berbagai parameter ACO (seperti jumlah semut, iterasi, alpha, beta) terhadap hasil spesifik dalam konteks studi kasus ini. Selain itu, penelitian ini hanya mempertimbangkan jarak sebagai faktor utama tanpa membandingkan kinerja ACO dengan algoritma lain, sehingga keunggulan relatifnya belum teruji. Sedangkan pada penelitian [2] menunjukkan bahwa implementasi ACO untuk menyelesaikan Traveling Tourism Problem (TTP) di Pulau Timor dengan 20 lokasi wisata berhasil menghasilkan jalur terpendek berdasarkan jarak. Pengujian menemukan bahwa peningkatan jumlah semut dan jumlah siklus cenderung menghasilkan rute yang lebih pendek, meskipun memerlukan waktu simulasi lebih lama, serta parameter α , β , dan ρ mempengaruhi hasil dan waktu simulasi. Kekurangan utama penelitian ini adalah hanya mempertimbangkan faktor jarak dalam perhitungan bobot rute, implementasi dilakukan menggunakan software (Matlab) untuk simulasi bukan aplikasi siap pakai, dan contoh implementasi menggunakan jumlah semut hanya 1, yang mungkin tidak sepenuhnya menunjukkan kekuatan algoritma koloni semut secara kolektif. Kemudian penelitian [4] berhasil mengembangkan aplikasi pemandu wisata berbasis Android untuk Provinsi Sumatera Utara yang memanfaatkan Algoritma ACO dalam menentukan rute terpendek ke tempat wisata. Hasilnya adalah tersedianya rute terpendek untuk mencari tempat pariwisata di wilayah tersebut. Namun, kekurangan penelitian ini adalah kurangnya penjelasan detail mengenai proses implementasi ACO

dalam aplikasi secara spesifik, seperti pengaturan parameter atau hasil rute spesifik yang dihasilkan. Selain itu, penelitian ini lebih fokus pada pengembangan aplikasi daripada analisis mendalam terhadap kinerja atau optimasi algoritma ACO itu sendiri, serta tidak menjelaskan secara rinci bagaimana ACO diterapkan dalam konteks spesifik aplikasi ini.

Sedangkan tujuan penelitian ini adalah untuk mengembangkan sebuah aplikasi berbasis website yang mampu menyusun rute perjalanan wisata di Jakarta secara optimal dengan menerapkan metode Algoritma *Ant colony* (ACO). Aplikasi ini bertujuan untuk membantu wisatawan dalam merencanakan perjalanan mereka dengan lebih efisien, baik dari segi waktu maupun jarak tempuh, dengan memberikan rekomendasi rute terbaik yang menghubungkan destinasi-destinasi wisata pilihan pengguna di wilayah Jakarta. Melalui pemanfaatan algoritma yang terinspirasi dari perilaku koloni semut dalam menemukan jalur terpendek, diharapkan aplikasi ini dapat menghasilkan solusi perutean yang lebih baik dibandingkan perencanaan manual, sehingga meningkatkan pengalaman dan efisiensi wisatawan saat menjelajahi kota Jakarta.

2. METODE PENELITIAN

Penelitian ini menggunakan mekanisme penelitian yang terlihat pada gambar 1 berikut ini :



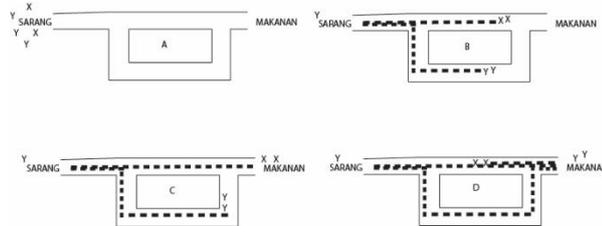
Gambar 1. Mekanisme penelitian

Pada gambar 1 terlihat mekanisme penelitian yang diawali dengan pengumpulan data, identifikasi masalah, perumusan masalah. Kemudian dilanjutkan dengan tahap pengembangan aplikasi dengan *Rapid Application Development* (RAD). Langkah pertama *Requirement Planning* (pengumpulan data dengan wawancara dan studi literatur, kemudian analisa sistem dan use case), *User Design*, *Construction* (menggunakan metode Algoritma Koloni Semut, membangun aplikasi dengan UML, *User Interface* dan Basis Data), kemudian tahap *construction* (dengan *source code*), *cutover* (tahap pengujian dengan *black box testing*). Terakhir adalah tahap kesimpulan hasil penelitian.

2.1. Algoritma *Ant colony*

Pada 1992, Marco Dorigo memperkenalkan algoritma *Ant colony Optimization* (ACO) yang terinspirasi dari penelitian ini. ACO mengganti semut asli dengan semut buatan, menggunakan graf untuk merepresentasikan jalur, dan menerapkan prinsip feromon serta aturan stokastik untuk memilih rute [2]. Algoritma *ant colony* diaplikasikan untuk menyelesaikan masalah *Travelling Salesman Problem*. Telah diketahui sebelumnya bahwa ketika berjalan semut akan meninggalkan suatu zat bernama pheromone, pheromone inilah yang digunakan sesama semut untuk menemukan sumber makanan terbaik dan jarak terdekat.

Pada awalnya semut akan bergerak secara acak, selama pergerakannya, semut akan meninggalkan pheromone. Kemudian apabila semut telah menemukan sumber makanan maka semut akan mengambil makanan tersebut sesuai yang dapat diangkutnya untuk dibawa kembali ke sarang. Apabila semut lain menemukan jejak pheromone maka ia akan mengikuti jejak tersebut yang akan menggiringnya menuju sumber makanan [5]. Semakin dekat jarak sumber makanan dengan sarang maka semakin cepat waktu tempuh yang diperlukan oleh semut, itu berakibat dengan semakin banyak jejak pheromone yang bertumpuk di jalur tersebut sebelum pheromone itu menguap. Kemudian semut lain akan mengikut rute yang memiliki jejak pheromone terkuat, simulasi pergerakan ant colony dapat dilihat pada gambar 2 berikut:

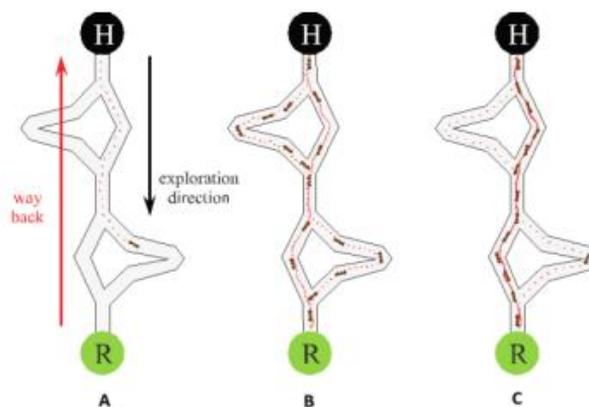


Gambar 2. Simulasi Pergerakan *ant colony* (urutan dari a ke d)

Dalam pengaplikasiannya terdapat perbedaan antara *ant colony* di dunia nyata dan *ant colony* buatan untuk algoritmya sebagai berikut:

- Semut buatan akan memiliki sedikit ingatan
- Semut buatan tidak buta sepenuhnya
- Semut buatan akan berada di lingkungan dimana waktu bersifat diskrit

Gambar 3 menjelaskan algoritma ACO secara sederhana, dimana H adalah Hive / Sarang Semut dan R adalah Resource atau Sumber makanan. Pada gambar 3(a) semut akan berjalan dari sarang menuju sumber makanan dan terdapat banyak rute yang dapat dipilih. Semut berjalan secara acak menuju ke makanannya dan pada setiap perjalanan semut meninggalkan feromon ke tanah. Feromon yang tertinggal diwakilkan dengan titik merah di dalam rute pada gambar 3. Gerombolan semut kemudian akan membagi perjalanan yang dilewati, beberapa semut melewati sisi kanan dan beberapa semut akan melewati sisi kiri, ditunjukkan pada gambar 3(b). Pada gambar 3(c) Gerombolan semut mampu menemukan rute terpendek dari sarang menuju sumber makanan dengan memanfaatkan jejak feromon yang mereka tinggalkan. Semakin banyak feromon yang tertinggal di suatu rute, semakin besar kemungkinan semut lain akan memilih rute tersebut [2].



Gambar 3. *Ant colony Optimization*

2.2. Wisata

Menurut Undang undang No 10 Tahun 2009 tentang Kepariwisataaan, wisata didefinisikan sebagai kegiatan perjalanan yang dilakukan oleh seseorang atau sekelompok orang dengan mengunjungi tempat tertentu untuk tujuan rekreasi, pengembangan pribadi, atau mempelajari keunikan daya tarik wisata yang dikunjungi dalam jangka waktu sementara.

Menurut United Nations World Tourism Organization (UNWTO) pariwisata terdiri dari kegiatan orang-orang yang bepergian ke dan tinggal di tempat di luar lingkungan biasa selama tidak lebih dari satu tahun berturut-turut untuk liburan, bisnis, dan tujuan lainnya [6], [7].

2.3. Google Maps API

Google Maps merupakan layanan aplikasi peta online yang disediakan oleh Google secara gratis. Layanan peta Google Maps secara resmi dapat diakses melalui situs <http://maps.Google.com>. Pada situs tersebut dapat dilihat informasi geografis pada hampir semua permukaan di bumi kecuali daerah kutub utara dan selatan. Layanan ini di buat sangat interaktif, karena di dalamnya peta dapat digeser sesuai keinginan pengguna, mengubah level zoom, serta mengubah tampilan jenis peta [8].

Google Maps mempunyai banyak fasilitas yang dapat di pergunakan misalnya pencarian lokasi dengan memasukkan kata kunci, kata kunci yang dimaksud seperti nama tempat, kota atau jalan, fasilitas lainnya yaitu perhitungan rute perjalanan dari satu tempat, ke tempat lain. Pada Google Maps API terdapat 4 jenis pilihan model peta yang disediakan oleh Google, diantaranya adalah:

- a. Roadmap, untuk menampilkan peta biasa 2 dimensi
- b. Satellite, untuk menampilkan foto satelit
- c. Terrain, untuk menunjukkan relief fisik permukaan bumi dan menunjukkan seberapa tingginya suatu lokasi, contohnya akan menunjukkan gunung dan sungai
- d. Hybrid, akan menunjukkan foto satelit yang diatasnya tergambar apa yang tampil pada roadmap (jalan dan nama kota) Google Map dibuat dengan menggunakan kombinasi dari gambar peta, database, serta obyek-obyek interaktif yang dibuat dengan bahasa pemrograman HTML, JavaScript dan AJAX, serta beberapa bahasa pemrograman lainnya.

2.4. Rapid Application Development (RAD)

Rapid Application Development merupakan metode pengembangan perangkat lunak dibuat oleh James Martin. Fase yang terdapat dalam metode RAD versi James Martin adalah requirements planning atau perencanaan persyaratan, user design atau desain pengguna, atau konstruksi, dan yang terakhir adalah fase *cutover* atau implementasi. Karakteristik dari metode RAD ini, pada fase perancangan dan konstruksi akan berjalan secara paralel, maksudnya adalah fase perancangan akan terus mengakomodir seluruh persyaratan dan desain dari aplikasi yang dibutuhkan pada fase konstruksi. Jadi ketika pada fase konstruksi semua desain dan persyaratan aplikasi sudah dibuat, maka selanjutnya akan melakukan tes uji coba aplikasi ke *user* atau pengguna [9], [10].

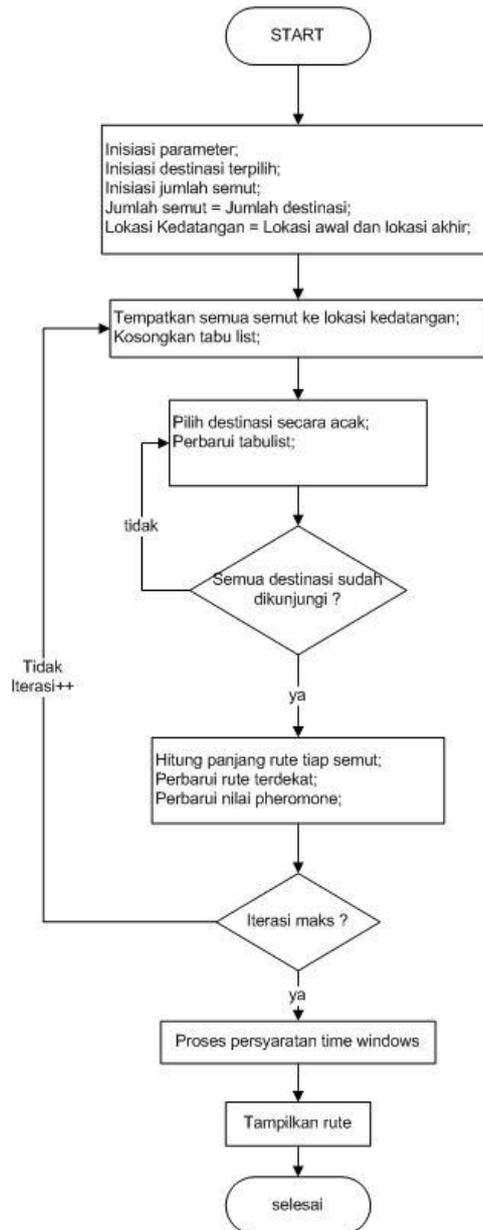
3. HASIL DAN PEMBAHASAN

Pada bagian ini dapat diuraikan mengenai hasil dari penelitian beserta pengujian yang telah dilakukan. Selain itu, disampaikan juga mengenai pembahasan dari penelitian maupun pengujian yang telah dilakukan.

Hasil dan pembahasan seharusnya merupakan bab yang paling banyak isinya pada sebuah paper. Isi Hasil dan Pembahasan dapat mencapai 40-60% dari keseluruhan paper.

3.1. Rancangan Algoritma *ant colony*

Berikut ini merupakan tahapan atau diagram alur algoritma *ant colony* terlihat pada gambar 4 :



Gambar 4. Flowchart Algoritma *ant colony*

a. Pemilihan Rute Objek Wisata

Tahap selanjutnya semut akan memilih objek wisata yang akan dikunjungi terlebih dahulu.

Persamaan penghitungan visibilitas, memungkinkan obyek wisata yang dekat dapat dikunjungi terlebih dahulu :

$$\mu_{ij} = \frac{1}{d_{ij}} \dots\dots\dots (1)$$

Keterangan :

μ_{ij} = visibilitas

d_{ij} = jarak antara kota i ke j

Persamaan untuk peluang dipilihnya suatu kota adalah :

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\mu_{ij}]^\beta}{\sum_{j \in allowed_k} [\tau_{ij}(t)]^\alpha \cdot [\mu_{ij}]^\beta} & \text{jika } j \in allowed_k \\ 0 & \text{untuk lainnya} \end{cases} \dots\dots\dots (2)$$

Keterangan :

$\tau_{ij}(t)$ = intensitas pheromone pada jalur (i,j)

- μ_{ij} = visibilitas
- α = parameter yang mengontrol tingkat intensitas pheromone
- β = parameter yang mengontrol tingkat visibilitas
- allowed_k = kota yang belum dikunjungi oleh semut

b. Pembaruan *Pheromone*

Setelah tiap tiap semut melakukan satu iterasi. Maka akan terpilih jalur terpendek. Setelah itu dilakukan pembaruan *pheromone* di tiap tiap jalur. Semua jalur akan mendapatkan penguapan *pheromone* sesuai dengan persamaan berikut :

$$\Delta\tau_{ij}(t + n) = \rho\tau_{ij}(t) + \Delta\tau_{ij} \dots\dots\dots (3)$$

- $\Delta\tau_{ij}(t)$ = kuantitas nilai *pheromone* jalur i,j pada iterasi t
- ρ = parameter penguapan
- τ_{ij} = intensitas *pheromone* jalur (i,j) pada iterasi sebelumnya

Kemudian jalur yang termasuk dalam tabu list dari setiap iterasi akan mendapatkan penambahan *pheromone* sesuai dengan persamaan berikut :

$$\Delta\tau_{ij} = \begin{cases} \frac{Q}{L_k} & \text{jika semut ke - k menggunakan busur (i,j)} \\ 0 & \text{untuk lainnya} \end{cases} \dots\dots\dots (4)$$

Keterangan :

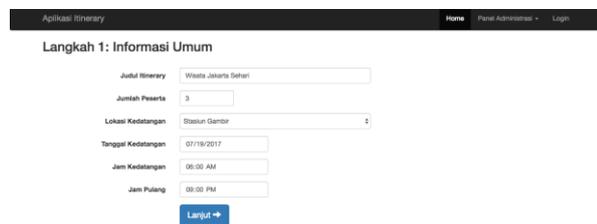
- $\Delta\tau_{ij}$ = kuantitas nilai *pheromone*
- Q = koefisien deposit *pheromone*
- L_k = panjang rute yang dihasilkan oleh semut ke k

Terdapat perbedaan pada pembentukan hasil akhir dalam implementasi algoritma *ant colony* ini pada aplikasi penyusunan rute perjalanan wisata dengan implementasi pada TSP. Pada penelitian ini ditambahkan sebuah proses pengecekan *time windows*, yaitu kecocokan jam buka dan jam tutup objek wisata serta durasi selama di objek wisata dengan durasi perjalanan wisata yang dimasukkan oleh pengunjung.

Time windows bersifat keras, jadi apabila terdapat objek wisata yang tidak memenuhi syarat diatas. Maka objek wisata tersebut akan dihapuskan dari solusi akhir.

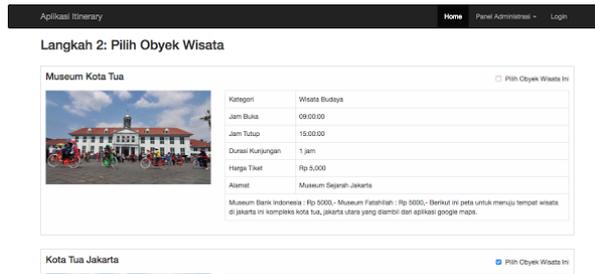
3.2. Perancangan Sistem

Penelitian ini mengembangkan aplikasi berbasis website dimana pada halaman utama *user* melakukan pengisian data informasi umum mengenai perjalanan yang akan direncanakan. setelah selesai, maka *user* dapat menekan tombol Lanjut. Gambar 5 merupakan halaman utama *user*



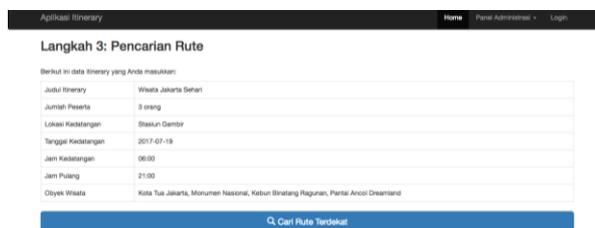
Gambar 5. Halaman utama *user*

Selanjutnya pada gambar 6 *user* memilih objek wisata yang diinginkan pada halaman Pilih Obyek Wisata untuk menentukan destinasi yang akan dicapai, setelah input data tersebut, maka *user* dapat menekan tombol Lanjut untuk proses selanjutnya.



Gambar 6. Pilih Obyek Wisata

Selanjutnya pada gambar 7 merupakan proses pencarian rute terdekat dengan menampilkan informasi umum yang telah user masukkan sebelumnya dan jika sudah sesuai maka useri dapat menekan tombol Cari Rute Terdekat yang kemudian aplikasi akan memproses dengan menggunakan algoritma *ant colony* tersebut untuk menyusun rute perjalanan wisata.



Gambar 7. Pencarian Rute

Aplikasi memproses data yang dimasukkan *user* dengan menggunakan algoritma *ant colony* untuk menyusun rute perjalanan wisata terdekat, dan hasilnya ditampilkan dalam bentuk pop up pada gambar 8 berikut :



Gambar 8. Hasil Pencarian Rute Terdekat

4. KESIMPULAN

Penelitian ini berhasil menerapkan metode algoritma *ant colony*. Implementasi aplikasi penyusunan rute perjalanan wisata memberikan hasil berupa informasi rute perjalanan wisata terdekat dan hasil survey menunjukkan bahwa sebanyak 93.3% *end user* menyatakan bahwa aplikasi ini sangat membantu, dan 60% koresponden menyatakan bahwa aplikasi ini sangat mudah untuk digunakan. Penelitian ini masih terdapat banyak kekurangan dari aplikasi ini dan memungkinkan untuk dilakukan pengembangan selanjutnya berupa pengembangan aplikasi versi mobile untuk memudahkan pengguna mengaksesnya melalui smartphone dan menambahkan pilihan kendaraan, seperti angkutan umum atau sepeda motor.

DAFTAR PUSTAKA

- [1] D. W. Nugraha, A. Y. Erwin Dodu, and S. Septiana, "Sistem Penentuan Rute Pendistribusian Produk Air Mineral Menggunakan Algoritma Ant Colony System," *Ilk. J. Ilm.*, vol. 11, no. 2, pp. 86–94, 2019, doi: 10.33096/ilkom.v11i2.418.86-94.
- [2] Y. R. Kaesmetan and M. V. Overbeek, "Ant Colony Optimization for Traveling Tourism Problem on Timor Island East Nusa Tenggara," *Indones. J. Artif. Intell. Data Min.*, vol. 3, no. 1, p. 28, 2020, doi: 10.24014/ijaidm.v3i1.9274.
- [3] D. Udjulawa and S. Oktarina, "Penerapan Algoritma Ant Colony Optimization Untuk Pencarian Rute

- Terpendek Lokasi Wisata,” *Klik - J. Ilmu Komput.*, vol. 3, no. 1, pp. 26–33, 2022, doi: 10.56869/klik.v3i1.326.
- [4] Z. Anshory, “Penerapan Algoritma Ant Colony Optimization Pada Aplikasi Pemandu Wisata Provinsi Sumatera Utara Berbasis Android,” *J. Comput. Syst. Informatics*, vol. 1, no. 2, pp. 61–67, 2020, [Online]. Available: <https://ejournal.seminar-id.com/index.php/josyc/article/view/106>
- [5] W. Fuadi, D. Fariadi, and D. A. Aqsa, “Ant Colony Optimization Pada E-Tourism Untuk Pemilihan Lokasi Tempat Wisata,” *J. Teknol. Terap. Sains 4.0*, vol. 4, no. 3, p. 92, 2023, doi: 10.29103/tts.v4i3.14616.
- [6] W. T. Atmojo, “Implementasi Algoritma Dijkstra Untuk Mencari Rute Tujuan Wisata Terpendek Dalam Aplikasi Berbasis Web,” in *Prosiding SeNTIK*, 2019. [Online]. Available: <https://ejournal.jak-stik.ac.id/index.php/sentik/article/download/205/65>
- [7] D. Widyaningrum and C. Danuputri, “Rancangan Sistem E-itinerary Wisata Di Kawasan Jakarta Utara Dengan Penerapan Algoritma Dijkstra,” *JT J. Tek.*, vol. 12, no. 2, p. 6909090, 2023, [Online]. Available: <http://jurnal.umt.ac.id/index.php/jt/index>
- [8] R. Ilham, A. Soetedjo, and A. Faisol, “Pengembangan Aplikasi Pencarian Rute Terpendek Dengan Metode Algoritma A * Berbasis Web,” *J. Elektro ELTEK*, vol. 2, no. 2, pp. 171–175, 2011.
- [9] S. Anisah, “Implementasi Metode Rapid Application Development,” *STRING (Satuan Tulisan Ris. dan Inov. Teknol.*, vol. 7, no. 1, 2022.
- [10] E. Febriani, B. Imran, and R. Muslim, “SISTEM INFORMASI E-COMMERCE PENJUALAN KERAJINAN ROTAN BERBASIS WEBSITE PADA DESA LOANG MAKA KECAMATAN JANAPRIA,” *J. Comput. Technol.*, vol. 1, no. 1, pp. 32–40, 2023, [Online]. Available: http://repository.pnb.ac.id/id/eprint/10155%0Ahttp://repository.pnb.ac.id/10155/2/RAMA_57401_2015323103.pdf