

Perencanaan Perjalanan Wisata Kabupaten Sambas Menggunakan Algoritma Ant Colony Optimization

Tourism Travel Planning in Sambas Regency Using Ant Colony Optimization Algorithm

Maulidawati*¹, Alda Cendekia Siregar², Rachmat Wahid Saleh Insani³

Program Studi Teknik Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Muhammadiyah Pontianak

E-mail : aulisp4.661@gmail.com*¹, alda.siregar@unmuhpnk.ac.id², rachmat.wahid@unmuhpnk.ac.id³

Received 23 February 2026; Revised 3 April 2026; Accepted 9 April 2026

Abstrak - Sebuah perjalanan perlu dilakukan perencanaan agar waktu kunjungan dapat dikelola, salah satunya yaitu kunjungan wisata. Banyaknya wisata yang terdapat di Indonesia masing-masing memiliki daya tarik tersendiri. Salah satunya adalah wisata yang ada di Kabupaten Sambas. Minimnya informasi mengenai lokasi destinasi wisata di daerah tersebut menjadi salah satu kendala yang dialami wisatawan. Tujuan dari penelitian ini yaitu mencari rute terpendek berdasarkan nilai parameter optimal. Dalam permasalahan ini, algoritma yang digunakan yaitu *Ant Colony Optimization* sebagai algoritma optimasi yang dapat diterapkan pada kasus penentuan rute kunjungan wisata. Algoritma tersebut merupakan algoritma yang terinspirasi dari kelompok koloni semut dalam mencari jalur untuk mendapatkan makanan. Pengujian parameter *Ant Colony Optimization* yaitu α , β , dan ρ dengan rentang nilai yang telah ditentukan. Kemudian dibandingkan berdasarkan rata-rata *fitness* terkecil dari hasil 75 kali percobaan *running*. Hasil dari pengujian algoritma *Ant Colony Optimization* didapatkan nilai optimal parameter, yaitu α bernilai 0,02 dengan rata-rata *fitness* 315,56 km, β bernilai 9,0 dengan rata-rata *fitness* 314,31 km, dan ρ bernilai 0,5 dengan rata-rata *fitness* 316,01 km. Setelah didapatkan nilai parameter optimal tersebut, kemudian diujikan kembali dengan dilakukan percobaan *running* sebanyak 5 kali, jadi total keseluruhan *running* sebanyak 80 kali. Didapatkan *fitness* terkecil senilai 312,67 km, nilai *fitness* terbesar senilai 318,32 km, dan rata-rata *fitness* sebesar 315,28 km.

Kata kunci - optimasi, *Ant Colony Optimization*, rute terpendek, perencanaan perjalanan

Abstract – A trip needs to be planned to manage the visit time, one of which is a tourist visit. The many tourist attractions in Indonesia each have their own appeal. One of them is tourism in Sambas Regency. The lack of information regarding the location of tourist destinations in the area is one of the obstacles experienced by tourists. The purpose of this study is to find the shortest route based on optimal parameter values. In this problem, the algorithm used is *Ant Colony Optimization* as an optimization algorithm that can be applied to the case of determining tourist visit routes. This algorithm is an algorithm inspired by groups of ant colonies in finding paths to get food. Testing *Ant Colony Optimization* parameters, namely α , β , and ρ with a predetermined value range. Then compared based on the smallest average *fitness* from the results of 75 *running* trials. The results of the *Ant Colony Optimization* algorithm test obtained optimal parameter values, namely, α is 0.02 with an average *fitness* of 315.56 km, β is 9.0 with an average *fitness* of 314.31 km, and ρ is 0.5 with an average *fitness* of 316.01 km. After obtaining the optimal parameter values, they were tested again by running the experiment five times, for a total of 80 runs. The lowest *fitness* value was 312.67 km, the highest *fitness* value was 318.32 km, and the average *fitness* value was 315.28 km.

Keywords – optimization, *Ant Colony Optimization*, shortest route, trip planner

1. PENDAHULUAN

Perencanaan perjalanan wisata atau yang disebut dengan *Tourist Trip Design Problem* (TTDP) adalah salah satu kendala yang kerap dihadapi oleh pengunjung saat akan melakukan perjalanan wisata ke suatu daerah. Kesulitan yang dihadapi pengunjung yaitu untuk memutuskan lokasi wisata mana saja yang akan dikunjungi, jarak antar lokasi wisata, dan waktu yang akan ditempuh selama perjalanan berlangsung. Oleh sebab itu, setiap pengunjung akan mengambil rute wisata terpendek dan tercepat sehingga dapat dicapai dengan memaksimalkan waktu yang ada. Agar dapat mengatasi permasalahan tersebut maka dibuat sebuah rencana perjalanan atau *trip planner* untuk membantu memilih rute antar lokasi wisata [1].

Perencanaan perjalanan wisata sangat dibutuhkan terlebih lagi bagi turis yang berkunjung ke Indonesia. Indonesia merupakan salah satu negara yang kaya akan destinasi wisata. Masing-masing daerah mempunyai destinasi tersendiri salah satunya yaitu di Kabupaten Sambas. Kabupaten Sambas adalah salah satu kabupaten yang terletak di ujung Kalimantan Barat dimana berbatasan langsung di bagian utara dengan Sarawak, Malaysia. Kabupaten Sambas merupakan salah satu daerah Tingkat II Provinsi Kalimantan Barat yang mana memiliki luas wilayah 6.395,70 km² atau 639.570 ha (4,36 dari luar wilayah Kal-Bar). Sambas yang terletak di bagian utara atau biasa dikenal juga sebagai daerah bagian pantai. Jarak Sambas untuk ke lokasi pantai sekitar 128,5 km dan panjang perbatasan negara sepanjang kurang lebih 97 km [2]. Sebagian besar destinasi di Kabupaten Sambas berupa pantai karena termasuk daerah pesisir. Selain pantai, terdapat bukit, air terjun, keraton, dan danau. Beberapa destinasi kerap dikunjungi baik masyarakat dalam maupun luar daerah.

Penelitian terdahulu yang relevan terkait dengan perencanaan perjalanan wisata dan algoritma *Ant Colony Optimization* yaitu terdapat dalam berbagai kasus yang dilakukan peneliti sebelumnya. Penelitian terdahulu membuat sebuah sistem berbasis web yang dapat membantu dalam pencarian obat pada beberapa apotek sesuai obat yang dicari oleh masyarakat dengan menerapkan algoritma *Ant Colony Optimization* (ACO) [3]. Penelitian lainnya menyediakan penentuan rute perjalanan wisata di Kota Tomohon dan memperlihatkan bahwa ACO bisa mengatasi kasus penentuan rute wisata di Kota Tomohon dengan baik dari segi waktu rute maupun biaya [4]. Penelitian berikutnya difokuskan pada implementasi algoritma *Ant Colony Optimization* atau *AntCO* yang berbentuk *website* menggunakan PHP dan digunakan pada kasus menentukan rute terpendek lokasi wisata Kota Palembang [5].

Penelitian terdahulu membangun sebuah sistem yang mampu menyajikan informasi lokasi ATM di Kota Palu sekaligus memberikan petunjuk rute tercepat menggunakan algoritma *Ant Colony Optimization* (ACO) *system* (ACS) [6]. Penelitian tentang wisata dengan beda algoritma yaitu membuat aplikasi untuk memberikan rekomendasi wisata agar pengunjung dapat merencanakan dengan baik saat akan berkunjung ke Pulau Lombok yang mana aplikasi ini menggunakan metode Fuzzy Mamdani dan Algoritma Genetika [7]. Penelitian lainnya melakukan pengujian terhadap parameter - parameter dari *Ant Colony Optimization* sehingga diperoleh parameter yang optimal, yaitu pada iterasi senilai 2000, nilai *alpha*, nilai *beta* 0,1, nilai *rho* 0,3, dan nilai *tau* 0,03. Hasil perbandingan sistem menggunakan algoritma *Ant Colony Optimization*, yaitu dengan rata-rata *fitness* dengan nilai 121,27 dengan keadaan asli sebesar 179,42 ternyata hasil yang diperoleh sistem jauh lebih pendek dari pada kondisi asli sehingga dapat memberikan hasil yang lebih optimal [8].

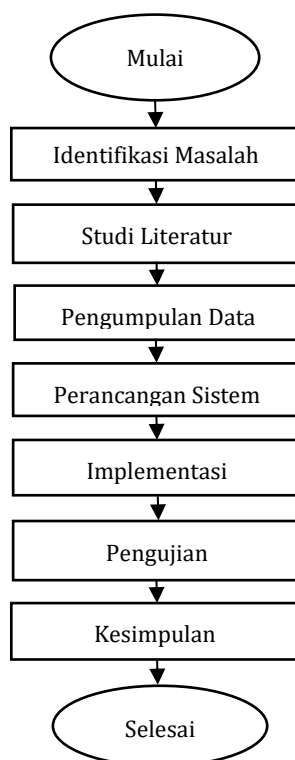
Penelitian tentang pendistribusian air mineral pada PT. Tirta Investama- DC. AQUA yang terdapat di area Minahasa Utara menjadi salah satu distributor AQUA. Didapatkan hasil penelitian dengan perhitungan algoritma *Ant System* menyatakan bahwa jalur distribusi yang diperoleh yaitu 18,14 km yang memiliki jalur distribusi lebih pendek dibandingkan dengan jalur yang sama berdasarkan Google Maps diperoleh jalur distribusi yaitu 19,60 km [9]. Penelitian tentang wisata dengan algoritma Dijkstra yang mana didapatkan bentuk dari graf berdasarkan data awal. Dilakukan perhitungan dengan mengimplementasikan Algoritma Dijkstra didapatkan 11 rute terpendek yang dapat dilalui saat menuju ke destinasi wisata dari titik awalnya adalah terminal

Bojonegoro [10]. Penelitian lainnya tentang rute tercepat menggunakan algoritma Dijkstra menuju fasilitas terdekat di Kabupaten Sanggau. Pada uji UAT diperoleh tingkat preferensi penerimaan sebesar 80,75% terhadap aplikasi yang dirancang, artinya baik dan praktis untuk digunakan dan aplikasi yang dirancang memiliki akurasi kesesuaian 90% dan ketidaksesuaian 10% dengan Google Maps [11].

Berdasarkan penelitian-penelitian terdahulu, sebagian besar studi hanya berfokus pada penerapan algoritma *Ant Colony Optimization* untuk pencarian rute tanpa membahas secara mendalam proses penentuan parameter optimal pada kasus spesifik, serta belum banyak yang mengintegrasikan hasil optimasi ke dalam aplikasi mobile yang dapat digunakan langsung oleh pengguna. Oleh karena itu, kontribusi utama penelitian ini adalah menerapkan algoritma *Ant Colony Optimization* pada kasus perencanaan perjalanan wisata di Kabupaten Sambas, lalu melakukan tuning parameter ACO yaitu *alpha*, *beta*, dan *rho* secara sistematis untuk mendapatkan kombinasi optimal, dan mengimplementasikan algoritma tersebut ke dalam aplikasi *mobile* berbasis android sebagai sistem *trip planner*. Perbedaan penelitian ini dibandingkan penelitian sebelumnya terletak pada fokus optimasi parameter ACO yang dikombinasikan dengan implementasi sistem nyata berbasis *mobile*, sehingga hasil yang diperoleh tidak hanya bersifat teoritis tetapi juga aplikatif.

2. METODE PENELITIAN

Metodologi penelitian digambarkan dalam bentuk diagram alir atau *flowchart*. Perancangan metodologi penelitian bertujuan untuk menghasilkan alur penelitian perencanaan perjalanan wisata Kabupaten Sambas menggunakan algoritma *Ant Colony Optimization*. *Flowchart* metodologi penelitian ditunjukkan pada Gambar 1, yang terdiri dari identifikasi masalah, studi literatur, pengumpulan data, perancangan sistem, implementasi, pengujian, dan kesimpulan.



Gambar 1. *Flowchart* Tahapan Penelitian

2.1. Identifikasi masalah

Tahapan ini membahas mengenai identifikasi masalah pada penelitian perencanaan perjalanan wisata Kabupaten Sambas menggunakan algoritma *Ant Colony Optimization*. Perlu adanya perencanaan perjalanan agar lebih teratur dalam menuju lokasi yang akan dikunjungi khususnya lokasi destinasi wisata. Banyaknya destinasi wisata yang ada di Kabupaten Sambas membuat penulis tertarik untuk menjadikannya studi kasus dalam penelitian perencanaan perjalanan wisata menggunakan algoritma *Ant Colony Optimization*.

2.2. Studi literatur

Pada tahapan ini yaitu mengumpulkan penelitian terdahulu dari berbagai sumber sebagai referensi atau acuan dalam menyelesaikan penelitian ini, baik dari segi perancangan sistem maupun penulisan skripsi. Adapun penelitian terdahulu tersebut berupa artikel jurnal, makalah, skripsi, buku, *e-book*, dan lain sebagainya. Penelitian terdahulu tersebut didapatkan dari internet tepatnya di Google Scholar, yang mana situs ini mudah diakses dalam mencari karya tulis ilmiah. Selain mengunjungi situs di internet, penulis juga mengunjungi perpustakaan untuk mendapatkan informasi lebih mengenai penelitian yang berkaitan dengan masalah yang akan diangkat.

2.3. Pengumpulan data

Selanjutnya yaitu tahap pengumpulan data yang mana data didapatkan dari mengamati titik koordinat lokasi wisata melalui Google Maps tanpa harus datang langsung ke titik lokasi penelitian. Titik lokasi wisata yang dipilih sudah terbaca di Google Maps memudahkan penulis untuk menghitung jarak antar lokasi wisata. Data yang akan digunakan yaitu sebanyak 24 destinasi wisata yang ada di Kabupaten Sambas. Variabel yang digunakan dalam penelitian ini yaitu nama lokasi wisata, alamat wisata, harga masuk, fasilitas wisata, dan titik koordinat. Data akan diolah dalam bentuk *array list* dalam file pemograman Java berbasis android. Tabel 1 menunjukkan data sampel yang digunakan.

Tabel 1. Data *Sample* Lokasi Wisata Kabupaten Sambas

Nama Wisata	Alamat Wisata	HTM	Fasilitas	Titik Koordinat
Batu Nenek	Desa Temajuk, Kec. Paloh	Masuk dan parkir gratis	Tidak tersedia	2.0169258127531218, 109.6019487294292
Teluk Atong	Desa Temajuk, Kec. Paloh	Masuk gratis parkir bayar	Kamar mandi, musholla, penyewaan alat, tempat makan, penginapan area parkir	2.0443856526121738, 109.61733054870913
Rumah Terbalik	Desa Temajuk, Kec. Paloh	Tiket masuk Rp 25.000/orang Parkir Rp 3.000/motor dan Rp 10.000/mobil	Penyewaan permainan, Café Food Court Live music, toilet umum, musholla, penginapan area parkir	1.9898566277234258, 109.59078350582624
Rumah Melayu Sambas	Desa Sungai Rambah, Kec. Sambas	Masuk dan parkir gratis	Tidak bersedia	1.3201361945228491, 109.27569374421923
Pantai Lestari	Desa Tanah Hitam, Kec. Paloh	Tiket masuk Rp 5.000/orang Parkir gratis	Warung, tempat santai, area parkir	1.6335926050315535, 109.22226011621417
Pantai Cemara	Dusun Guntung, Desa Tanah Hitam, Kec. Paloh	Masuk dan parkir gratis	Warung, tempat santai	1.6516981190469233, 109.24404132375452
Pantai Kampak	Dusun Cermai, Desa Sebusus, Kec. Paloh	Masuk gratis, parkir Rp 5.000/motor dan Rp 10.000/mobil	Penginapan, toilet, rumah makan, area parkir	1.907931720058708, 109.33836115301669
Konservasi Penyu	Dusun Cermai, Desa Sebusus, Kec. Paloh	Masuk dan parkir gratis	Toilet, musholla, aula, camping area life jacket, area parkir	1.9459899822552298, 109.3750165440529

2.4. Perancangan sistem

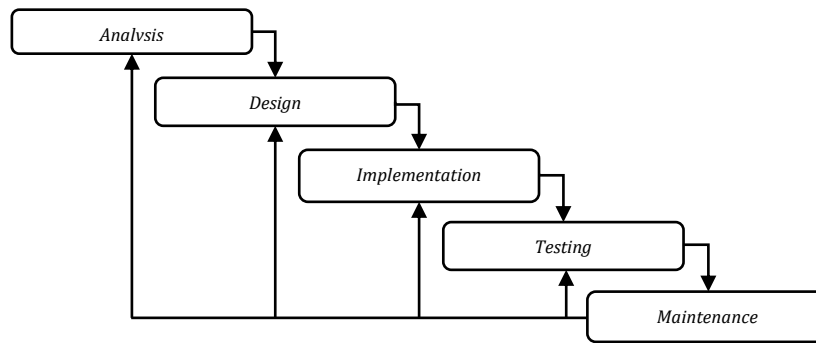
Tahap selanjutnya yaitu perancangan sistem, dimana pada tahapan ini akan dirancang sistem mulai dari desain antarmuka sampai pada tahap diimplementasikannya algoritma *Ant Colony Optimization*. Perancangan sistem ini akan melibatkan *software* dan *hardware*. *Software* yang dimaksud yaitu aplikasi untuk perancangan sistem berbasis web atau android dimana di dalamnya terdapat bahasa pemrograman dan fitur untuk mendesain antarmuka dan menjalankan kode program. *Hardware* sendiri yaitu perangkat keras seperti laptop, *mouse*, dan *smartphone*.

2.5. Implementasi

Setelah dilakukan perancangan sistem, tahapan berikutnya yaitu implementasi metode pengembangan perangkat lunak yaitu metode *waterfall* dan algoritma optimasi yaitu algoritma *Ant Colony Optimization*.

a. Metode *Waterfall*

Metode *waterfall* adalah siklus hidup klasik sebagai pengembangan pada *software* yang tersusun digambarkan dengan pendekatan yang sistematis diawali dengan rincian dari kebutuhan *user* dilanjutkan dengan tahapan-tahapan perencanaan, permodelan, konstruksi, serta penyerahan sistem kepada *user*, dan diakhiri dengan pendukung dari sebuah *software* lengkap yang dibangun [12]. Meskipun model metode ini termasuk yang kuno, namun bagi para pengembang metode ini sangat layak digunakan [13].



Gambar 2. Tahapan Metode *Waterfall*

1. *Analysis*

Tahap analisis yaitu mengumpulkan segala kebutuhan baik itu *software* atau *hardware*, dokumen, dan data yang akan digunakan dalam perancangan aplikasi. Kebutuhan tersebut akan dianalisis sesuai yang diperlukan pengguna dimana nantinya akan berpengaruh pada perancangan *user interface* dan fitur aplikasi.

2. *Design*

Tahap desain yaitu merancang segala yang dibutuhkan sebelum pada tahap implementasi dimana dalam tahap desain terdapat proses desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, dan representasi antar muka.

3. *Implementation*

Tahap implementasi yaitu dimana penulis melakukan pemrograman dan meimplementasikan *user interface* yang telah dirancang sebelumnya menggunakan kode program. Pada tahapan ini penulis menggunakan Android Studio sebagai *framework* untuk melakukan pemrograman dengan bahasa Java dalam pengembangan aplikasi berbasis mobile.

4. *Testing*

Tahap pengujian yaitu mengujikan aplikasi yang telah dikembangkan apakah sudah sesuai dengan yang direncanakan atau masih ada fungsi, fitur, maupun antarmuka yang belum sesuai. Pengujian dilakukan secara keseluruhan pada aplikasi untuk mencari

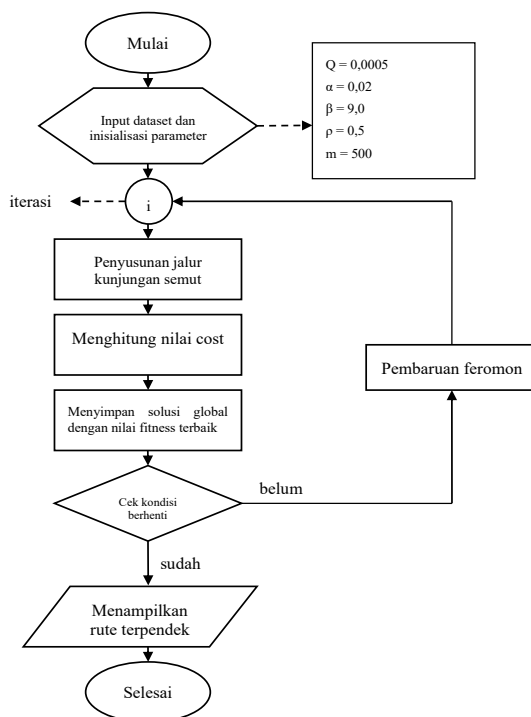
kemungkinan adanya kesalahan dalam perancangan dengan menggunakan *blackbox testing*.

5. *Maintenance*

Tahap pemeliharaan merupakan tahapan akhir dari metode pengembangan perangkat lunak *waterfall*. Pada tahapan ini *software* akan dilakukan pemeliharaan guna mempertahankan fungsional yang ada pada aplikasi tersebut dan akan digunakan untuk perencanaan rute perjalanan wisata di Kabupaten Sambas sebagaimana mestinya.

b. Algoritma *Ant Colony Optimization*

Ant Colony Optimization (ACO) merupakan pengembangan dari *Ant Colony* yang bekerja mulai dari, sejumlah m semut ditempatkan pada sejumlah n titik berdasarkan beberapa aturan inialisasi (misalnya, secara acak). Masing-masing semut membuat sebuah kunjungan yang mana itu sebuah solusi jalur evakuasi yang mungkin dengan meimplementasikan sebuah aturan transisi status secara berulang-ulang. Saat membuat kunjungannya, tiap semut juga merubah jumlah *pheromone* pada *edge-edge* yang dikunjunginya dengan meimplementasikan aturan pembaruan *pheromone local* yang telah disebutkan. Setelah semua semut selesai dengan perjalanan mereka, jumlah *pheromone* pada *edge-edge* diubah kembali dengan meimplementasikan aturan pembaruan *pheromone global*. Dalam membuat kunjungan, semut ‘dipandu’ oleh informasi *heuristic* atau mereka lebih memilih *edge-edge* yang pendek dan oleh informasi *pheromone*. Sebuah *edge* yang punya jumlah *pheromone* tinggi ialah pilihan yang sangat diinginkan. Kedua aturan pembaruan *pheromone* itu dibuat supaya semut cenderung untuk memberi lebih banyak *pheromone* pada *edge-edge* yang harus mereka lewati [14].



Gambar 3. *Flowchart Ant Colony Optimization*

Berdasarkan *flowchart* algoritma *Ant Colony Optimization*, proses dimulai dengan inialisasi parameter dan *pheromone* awal. Selanjutnya, setiap semut membangun solusi dengan memilih jalur berdasarkan probabilitas yang dipengaruhi oleh *pheromone* dan *heuristic*. Setelah semua semut menyelesaikan rute, dilakukan perhitungan nilai *cost* dan penyimpanan solusi terbaik secara global. Kemudian dilakukan pembaruan *pheromone* melalui proses evaporasi dan

deposit *pheromone*. Proses ini diulang hingga memenuhi kondisi berhenti, dan solusi terbaik yang diperoleh merupakan rute terpendek yang dihasilkan oleh algoritma. Urutan formulasi matematis algoritma *Ant Colony Optimization* sebagai berikut:

1. Representasi masalah

$$G = (V, E) \quad (1)$$

dengan:

V = himpunan *node* (lokasi wisata)

E = himpunan *edge* (jalur antar lokasi)

d_{ij} = jarak antara *node* i dan j

2. Inisialisasi parameter

$Alpha (\alpha)$ = pengaruh *pheromone*

$Beta (\beta)$ = pengaruh *heuristic*

$Rho (\rho)$ = evaporasi *pheromone*

Q = konstanta *pheromone*

m = jumlah semut

Pheromone awal:

$$\tau_{ij} = \tau_0 \quad (2)$$

3. Fungsi *heuristic*

Berdasarkan jarak:

$$\eta_{ij} = \frac{1}{d_{ij}} \quad (3)$$

4. Probabilitas perpindahan (penyusunan jalur)

$$P_{ij}^k = \frac{(\tau_{ij})^\alpha (\eta_{ij})^\beta}{\sum_{l \in N_k} (\tau_{il})^\alpha (\eta_{il})^\beta} \quad (4)$$

dengan:

P_{ij}^k = probabilitas semut ke- k dari i ke j

N_k = *node* yang belum dikunjungi

5. Perhitungan *cost* (*fitness*)

Menghitung nilai *cost*:

$$L_k = \sum_{(i,j) \in \text{rute}_k} d_{ij} \quad (5)$$

L_k = total jarak rute semut ke- k

6. Penyimpanan solusi terbaik

Menyimpan solusi global:

$$L_{best} = \min(L_k) \quad (6)$$

7. Pembaruan *pheromone*

Evaporasi:

$$\tau_{ij} = (1 - \rho) \tau_{ij} \quad (7)$$

Deposit *pheromone*:

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, & \text{jika edge } (i, j) \text{ dilalui semut ke } - k \\ 0, & \text{lainnya} \end{cases} \quad (8)$$

Update global:

$$\tau_{ij} = (1 - \rho) \tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k \quad (9)$$

8. Kondisi berhenti

Iterasi berhenti jika:

Iterasi = $iterasi_{maks}$ (10)
 atau solusi sudah konvergen.

9. *Output*

Hasil akhir:

Rute optimal = rute dengan L_{best} (11)

2.6. *Pengujian*

Tahapan setelah dilakukannya implementasi algoritma *Ant Colony Optimization* pada sistem yang telah dirancang, langkah selanjutnya yaitu melakukan pengujian terhadap perangkat lunak atau aplikasi yang telah dirancang dengan menggunakan *blackbox testing*. Selain itu juga, akan dilakukan pengujian terhadap parameter-parameter algoritma ACO dalam beberapa kali untuk mengetahui nilai parameter yang optimal. Parameter-parameter tersebut akan dilakukan perbandingan melalui grafik sehingga dapat diketahui rata-rata *fitness* terkecil dari total pengujian yang dilakukan.

2.7. *Kesimpulan*

Setelah dilakukan pengujian parameter algoritma *Ant Colony Optimization* maka dapat ditarik kesimpulan untuk menentukan nilai optimal dari perhitungan rute terpendek perjalanan menuju titik-titik lokasi wisata yang telah ditentukan setelah didapatkan hasil rata-rata *fitness* dari perbandingan parameter - parameter algoritma ACO.

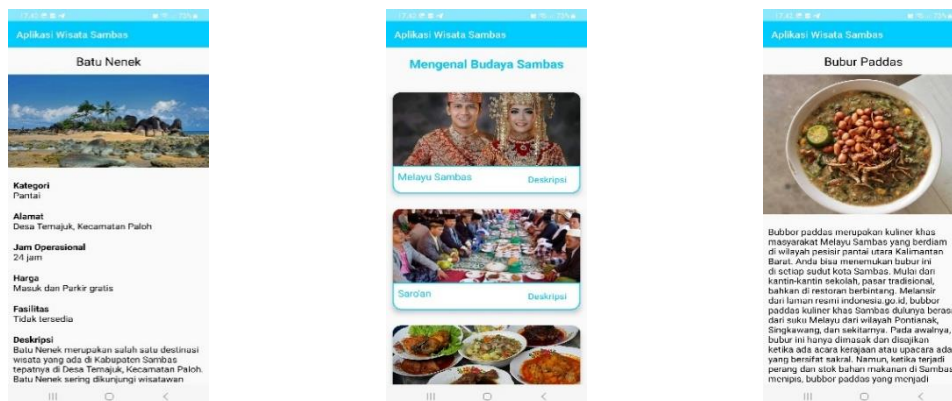
3. HASIL DAN PEMBAHASAN

3.1. *Hasil Tampilan Aplikasi Mobile*

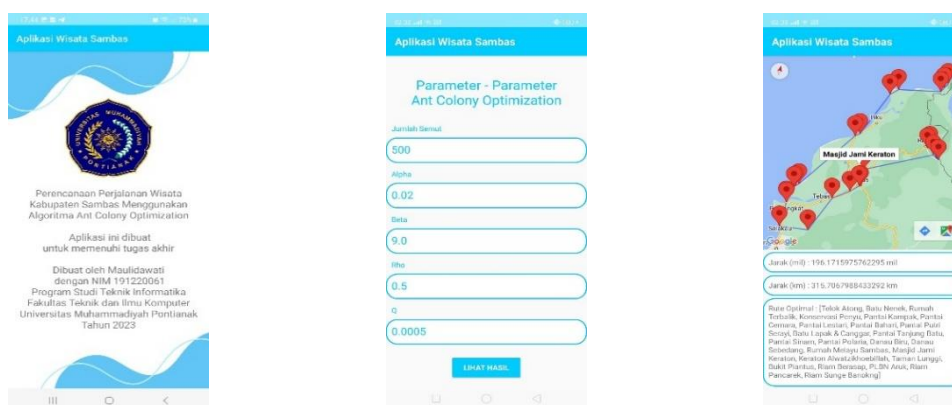
Hasil *Screenshot* dari sistem bisa dilihat pada Gambar 4 *Splash Screen*, *Home Page*, dan *List Wisata Page*, Gambar 5 *Deskripsi Wisata Page*, *List Budaya Page*, dan *Deskripsi Budaya Page*, Gambar 6 *Tentang Page*, *Parameter Page*, dan *Maps* untuk menampilkan hasil rute dari perhitungan *Ant Colony Optimization*.



Gambar 4. *Splash Screen*, *Home Page*, dan *List Wisata Page*



Gambar 5. Deskripsi Wisata Page, List Budaya Page, Deskripsi Budaya Page



Gambar 6. Tentang Page, Parameter Page, Maps

3.2. Pengujian Aplikasi dengan Blackbox

Adapun pengujian terhadap aplikasinya menggunakan pengujian *Blackbox testing* merupakan salah satu metode untuk melakukan pengujian sistem tanpa mengetahui susunan kode dari perangkat lunak tersebut. Pengujian ini dilakukan untuk memvalidasi dan mengamati perilaku *input* dan *output* yang dilakukan oleh sistem [15]. Berikut di bawah ini merupakan hasil dari pengujian *Blackbox*.

Tabel 2. Hasil Pengujian *Blackbox*

No.	Fungsi yang Diuji	Skenario Pengujian	Hasil	Kesimpulan
1.	Lihat Halaman Wisata	Mengklik menu wisata yang ada di halaman utama	Proses berhasil, pengguna dapat masuk ke halaman wisata	Valid
2.	Mencari Nama Wisata	Mengetikkan nama wisata yang ada di menu pencarian pada halaman wisata	Proses berhasil, pengguna mendapatkan nama wisata sesuai yang dicari	Valid
3.	Lihat Deskripsi Wisata	Mengklik gambar atau nama wisata yang ada di halaman wisata	Proses berhasil, halaman deskripsi wisata ditampilkan	Valid
4.	Lihat Halaman Budaya	Mengklik menu budaya yang ada di halaman utama	Proses berhasil, pengguna dapat masuk ke halaman budaya	Valid
5.	Lihat Deskripsi Budaya	Mengklik gambar atau nama budaya yang ada di halaman budaya	Proses berhasil, halaman deskripsi budaya ditampilkan	Valid
6.	Lihat Halaman Tentang	Mengklik menu tentang yang ada di halaman tentang	Proses berhasil, halaman tentang ditampilkan	Valid

7.	Lihat Halaman Parameter	Mengklik menu maps yang ada di halaman utama	Proses berhasil, halaman parameter ditampilkan	Valid
8.	Lihat Halaman Maps	Mengklik tombol lihat hasil pada halaman parameter	Proses berhasil, pengguna dapat melihat rute pada halaman maps	Valid

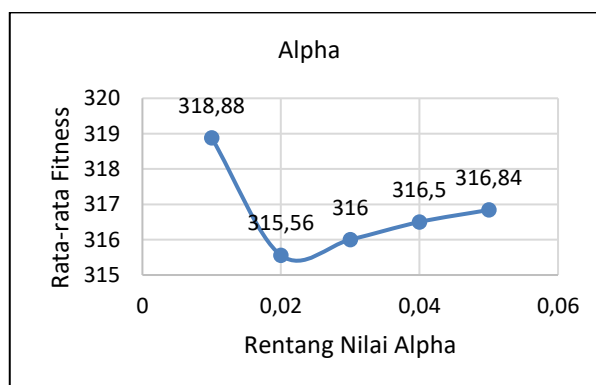
3.3. Pembahasan

Pada pengujian parameter *Ant Colony Optimization* melibatkan beberapa parameter yang biasa digunakan. Pada pengujian ini penulis menggunakan parameter *alpha*, *beta*, dan *rho*. Masing-masing dari parameter akan diujikan dengan beberapa kali dijalankan (*running*) kemudian didapatkan rata-rata *fitness* dari hasil perhitungan.

Tabel 3. Pengujian Parameter *Alpha*

Nilai <i>alpha</i> yang diujikan	Total jarak terbaik/ <i>fitness</i> pada setiap 5 kali di- <i>running</i>	Rata-rata <i>fitness</i>
0,01	1. 325,38 km 2. 325,38 km 3. 314,13 km 4. 314,45 km 5. 315,04 km	318,88 km
0,02	1. 313,62 km 2. 314,96 km 3. 319,27 km 4. 313,10 km 5. 316,87 km	315,56 km
0,03	1. 319,27 km 2. 314,84 km 3. 316,39 km 4. 317,03 km 5. 312,48 km	316,00 km
0,04	1. 313,49 km 2. 316,77 km 3. 314,55 km 4. 314,02 km 5. 323,66 km	316,50 km
0,05	1. 320,10 km 2. 322,32 km 3. 313,95 km 4. 314,13 km 5. 313,71 km	316,84 km

Berdasarkan Tabel 3, hasil pengujian parameter *alpha* menunjukkan bahwa rata-rata *fitness* terbaik diperoleh pada *alpha* = 0,02 dengan nilai sebesar 315,56 km. Nilai tersebut merupakan yang paling kecil dibandingkan dengan nilai *alpha* lainnya dalam rentang 0,01 hingga 0,05.



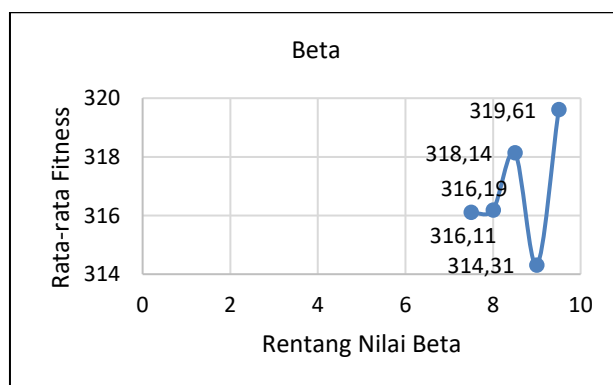
Gambar 7. Grafik Nilai *Alpha*

Berdasarkan Gambar 7, terlihat bahwa grafik nilai α menunjukkan tren penurunan dari $\alpha = 0,01$ ke $\alpha = 0,02$, kemudian kembali meningkat pada nilai α berikutnya. Titik minimum pada grafik berada pada $\alpha = 0,02$, yang mengindikasikan bahwa nilai tersebut merupakan titik optimal. Hasil yang diperoleh menunjukkan bahwa nilai $\alpha = 0,02$ memberikan keseimbangan yang optimal antara eksplorasi dan eksploitasi. Hal ini terlihat dari grafik yang menunjukkan titik minimum pada nilai tersebut, sehingga dapat disimpulkan bahwa pengaruh *pheromone* dalam proporsi kecil lebih efektif dalam menghasilkan rute optimal pada kasus ini.

Tabel 4. Pengujian Parameter β

Nilai β yang diujikan	Total jarak terbaik/ <i>fitness</i> pada setiap 5 kali di- <i>running</i>	Rata-rata <i>fitness</i>
7,5	1. 314,84 km 2. 326,94 km 3. 313,83 km 4. 312,25 km 5. 312,67 km	316,11 km
8,0	1. 321,09 km 2. 314,93 km 3. 318,87 km 4. 312,58 km 5. 313,49 km	316,19 km
8,5	1. 316,86 km 2. 321,21 km 3. 320,32 km 4. 317,16 km 5. 315,17 km	318,14 km
9,0	1. 314,96 km 2. 312,98 km 3. 314,45 km 4. 313,83 km 5. 315,32 km	314,31 km
9,5	1. 325,33 km 2. 316,72 km 3. 323,97 km 4. 314,84 km 5. 317,18 km	319,61 km

Berdasarkan Tabel 4, hasil pengujian parameter β menunjukkan bahwa rata-rata *fitness* terbaik diperoleh pada $\beta = 9,0$ dengan nilai sebesar 314,31 km. Nilai ini merupakan yang paling kecil dibandingkan dengan nilai β lainnya dalam rentang 7,5 hingga 9,5.



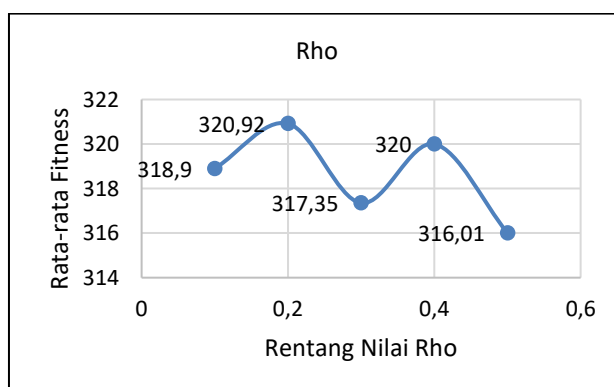
Gambar 8. Grafik Nilai β

Berdasarkan Gambar 8, grafik menunjukkan pola fluktuatif dengan penurunan *fitness* hingga mencapai titik minimum pada $\beta = 9,0$, kemudian meningkat kembali pada nilai β yang lebih besar. Hal ini menunjukkan bahwa peningkatan pengaruh *heuristic* mampu memperbaiki kualitas solusi hingga batas tertentu. Namun, jika nilai β terlalu besar, algoritma menjadi terlalu deterministik dan kurang mengeksplorasi kemungkinan solusi lain. Dengan demikian, nilai $\beta = 9,0$ merupakan nilai yang paling optimal karena mampu memaksimalkan pemanfaatan informasi jarak tanpa mengurangi kemampuan eksplorasi algoritma secara signifikan.

Tabel 5. Pengujian Parameter *Rho*

Nilai <i>rho</i> yang diujikan	Total jarak terbaik/nilai <i>fitness</i> pada setiap 5 kali <i>di-running</i>	Rata-rata <i>fitness</i>
0,1	1. 325,86 km 2. 313,49 km 3. 314,35 km 4. 326,97 km 5. 313,83 km	318,90 km
0,2	1. 323,77 km 2. 323,97 km 3. 316,32 km 4. 320,37 km 5. 320,17 km	320,92 km
0,3	1. 319,27 km 2. 314,93 km 3. 319,67 km 4. 315,69 km 5. 317,21 km	317,35 km
0,4	1. 314,55 km 2. 327,13 km 3. 317,71 km 4. 315,04 km 5. 325,58 km	320,00 km
0,5	1. 315,90 km 2. 315,91 km 3. 313,42 km 4. 320,07 km 5. 314,76 km	316,01 km

Berdasarkan Tabel 5, hasil pengujian parameter *rho* menunjukkan bahwa rata-rata *fitness* terbaik diperoleh pada $\rho = 0,5$ dengan nilai sebesar 316,01 km. Nilai ini merupakan yang paling kecil dibandingkan dengan nilai *rho* lainnya dalam rentang 0,1 hingga 0,5.



Gambar 9. Grafik Nilai *Rho*

Berdasarkan Gambar 9, grafik menunjukkan pola yang tidak stabil dengan beberapa fluktuasi *fitness* pada setiap perubahan nilai *rho*. Namun demikian, terlihat bahwa *fitness* cenderung lebih rendah pada nilai *rho* yang lebih besar. Hal ini menunjukkan bahwa tingkat evaporasi yang tinggi membantu algoritma menghindari konvergensi prematur dan meningkatkan peluang menemukan solusi yang lebih optimal. Oleh karena itu, nilai $\rho = 0,5$ dianggap paling optimal karena mampu menjaga keseimbangan antara eksplorasi dan eksploitasi dalam proses pencarian rute.

Pengujian selanjutnya yaitu dilakukan terhadap parameter-parameter yang telah didapatkan rata-rata *fitness* terkecil berdasarkan pengujian nilai acak dari rentang nilai yang ditentukan. Total *running* yang sudah dilakukan sebanyak 75 kali untuk parameter *alpha* 25 kali, *beta* 25 kali, dan *rho* 25 kali. Terakhir gabungan rata-rata *fitness* terkecil dilakukan *running* kembali sebanyak 5 kali, jadi total *running* yang telah dilakukan sebanyak 80 kali *running*.

Tabel 6. Pengujian *Alpha*, *Beta*, dan *Rho*

Di- <i>running</i> ke-	Total jarak terbaik/ <i>fitness</i>
1	318,32 km
2	318,17 km
3	312,67 km
4	313,42 km
5	313,83 km
Rata-rata <i>fitness</i>	315,28 km

Berdasarkan Tabel 6, pengujian dilakukan menggunakan kombinasi parameter optimal yang telah diperoleh sebelumnya, yaitu $\alpha = 0,02$, $\beta = 9,0$, dan $\rho = 0,5$. Hasil pengujian menunjukkan bahwa *fitness* terkecil adalah 312,67 km, sedangkan *fitness* terbesar adalah 318,32 km, dengan rata-rata *fitness* 315,28 km. Variasi *fitness* yang relatif kecil antar *running* menunjukkan bahwa algoritma memiliki tingkat kestabilan yang baik dalam menghasilkan solusi. Hal ini mengindikasikan bahwa kombinasi parameter yang digunakan sudah cukup optimal dan konsisten dalam menemukan rute terbaik.

Secara keseluruhan, kombinasi parameter tersebut mampu menghasilkan keseimbangan yang efektif antara eksplorasi dan eksploitasi. Nilai *alpha* yang kecil mengurangi dominasi *pheromone*, nilai *beta* yang besar memperkuat pengaruh jarak sebagai faktor utama, dan nilai *rho* yang tinggi membantu menghindari stagnasi solusi. Dengan demikian, hasil penelitian ini menunjukkan bahwa proses tuning parameter pada algoritma *Ant Colony Optimization* memiliki peran yang sangat penting dalam meningkatkan kualitas solusi, khususnya dalam kasus perencanaan perjalanan wisata di Kabupaten Sambas.

4. KESIMPULAN

Berdasarkan analisis dan pengujian yang telah dilakukan dapat disimpulkan bahwa algoritma *Ant Colony Optimization* mampu memberikan rute optimal untuk kasus perencanaan perjalanan wisata Kabupaten Sambas. Dimulai dengan menginput data dan menginisialisasi parameter, penyusunan jalur kunjungan semut, menghitung nilai *cost*, dan menyimpan solusi global dengan nilai *fitness* terbaik. Kemudian dicek kondisi berhenti, jika belum maka *pheromone* diperbarui dan dilakukan iterasi lagi, jika sudah akan ditampilkan rute terpendek. Data yang digunakan yaitu *latitude* dan *longitude* yang diinput ke dalam *array list* pada sistem yang dibangun. Sistem dibangun menggunakan bahasa pemrograman Java dan bahasa *markup XML* pada Android Studio.

Adapun parameter *Ant Colony Optimization* yang diujikan yaitu *alpha* (α), *beta* (β), dan *rho* (ρ) untuk mendapatkan nilai optimal. Masing-masing parameter diberikan rentang nilai. Rentang nilai *alpha* dari 0,01 – 0,05, rentang nilai *beta* dari 7,5 – 9,5, dan rentang nilai *rho* dari 0,1 – 0,5. Dilakukan percobaan *running* sebanyak 75 kali, 25 kali untuk *alpha*, 25 kali untuk *beta*, dan 25 kali untuk *rho*. Percobaan awal digunakan nilai *default* dari setiap parameter yaitu, *alpha*

dengan nilai 0,01, β dengan nilai 9,5, dan ρ dengan nilai 0,2. Nilai parameter optimal yang didapatkan yaitu nilai α 0,02 dengan rata-rata $fitness$ terkecil 315,56 km, nilai β 9,0 dengan rata-rata $fitness$ terkecil 314,31 km, dan nilai ρ 0,5 dengan rata-rata $fitness$ terkecil 316,01 km. Setelah didapatkan nilai parameter optimal tersebut, kemudian diujikan kembali dengan percobaan *running* sebanyak 5 kali, jadi total percobaan *running* sebanyak 80 kali. Didapatkan $fitness$ terkecil senilai 312,67 km, $fitness$ terbesar senilai 318,32 km, dan rata-rata $fitness$ sebesar 315,28 km.

Berdasarkan penelitian yang telah dilakukan, diharapkan kepada penelitian selanjutnya dapat mengembangkan algoritma *Ant Colony Optimization* ini dengan menambahkan titik pengguna ke titik awal rute perjalanan wisata. Pada sistem juga dapat ditambahkan perkiraan kedatangan pengunjung berdasarkan jam operasional dan perkiraan biaya transportasi yang harus dipersiapkan selama di perjalanan.

DAFTAR PUSTAKA

- [1] M. K. Ramadhansyah, "IMPLEMENTASI ALGORITMA ANT COLONY OPTIMIZATION (ACO) UNTUK PERENCANAAN PERJALANAN WISATA," 2018.
- [2] A. Aslan, N. T. P. Sihaloho, I. H. Nugraha, B. Karyanto, and Z. Zakaria, "Paradigma Baru Tradisi 'Antar Jung' Pada Masyarakat Paloh, Kabupaten Sambas," *IBDA' : Jurnal Kajian Islam dan Budaya*, vol. 18, no. 1, pp. 87–103, Apr. 2020, doi: 10.24090/ibda.v18i1.3354.
- [3] J. Khatib Sulaiman and J. Hardiansyah, "Implementasi Metode Ant Colony pada Pemetaan Lokasi Apotek Beserta Informasi Pencarian Obat Berbasis Website," *Indonesian Journal of Computer Science*.
- [4] T. Kaunang and K. D. Hartomo, "Pencarian Rute Optimal Wisata Alam Kota Tomohon Menggunakan Ant Colony Optimization (ACO)."
- [5] D. Udjulawa and S. Oktarina, "PENERAPAN ALGORITMA ANT COLONY OPTIMIZATION UNTUK PENCARIAN RUTE TERPENDEK LOKASI WISATA (STUDI KASUS WISATA DI KOTA PALEMBANG)," vol. 3, no. 1, 2022.
- [6] D. W. Nugraha, Amriana, and R. Setiawati, "Implementasi Algoritma Ant Colony Optimization (ACO) Pada Pencarian Jalur Terpendek Automatic Teller Machine (ATM) Di Kota Palu," 2020, doi: 10.30743/infotekjar.v4i2.1437.
- [7] Z. Harir, I. Bagus, K. Widiartha, R. Afwani, and P. Korespondensi, "APLIKASI PERTIMBANGAN WISATA DI PULAU LOMBOK DENGAN METODE FUZZY MAMDANI & ALGORITMA GENETIKA," vol. 7, no. 6, pp. 1261–1270, 2020, doi: 10.25126/jtiik.202071197.
- [8] A. A. Hasna, B. Rahayudi, and A. W. Widodo, "Optimasi Rute Distribusi Produk Minuman dan Makanan pada Distributor Nestle (CV Forward Kediri) dengan Algoritma Ant Colony Optimization," 2021. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [9] D. E. A. Manuputty, C. E. J. C. Montolalu, T. Manurung, and K. Kunci, "Penentuan Jalur Terpendek Distribusi Air Mineral Menggunakan Ant Colony Optimization," 2021. [Online]. Available: <https://ejournal.unsrat.ac.id/index.php/decartesian>
- [10] M. Abdul Aziz, "Implementasi Algoritma Dijkstra dalam Menentukan Rute Terpendek Terhadap Destinasi Wisata Kabupaten Bojonegoro," vol. 7, 2023.
- [11] Reyhan, Rachmat Wahid Saleh Insani, and Barry Ceasar Octariadi, "Geographical information system for garbage collection in sanggau city and shortest path using dijkstra's algorithm," *TEKNOSAINS : Jurnal Sains, Teknologi dan Informatika*, vol. 11, no. 1, pp. 49–61, Jan. 2024, doi: 10.37373/tekno.v11i1.485.
- [12] M. Yulianti, "SISTEM INFORMASI PENDAFTARAN PESERTA DIDIK BARU (PPDB) SMK IPTEK TANGSEL BERBASIS WEB DENGAN METODE WATERFALL," 2023. [Online]. Available: <https://journal.mediapublikasi.id/index.php/logic>

- [13] E. Listiyan and E. R. Subhiyakto, “Rancang Bangun Sistem Inventory Gudang Menggunakan Metode Waterfall (Studi Kasus Di CV. Aqualux Duspha Abadi Kudus Jawa Tengah),” 2021.
- [14] A. E. Prasetya, “Pencarian Rute Tercepat Mobil Ambulance Menggunakan Algoritma Ant Colony Optimization,” 2019. [Online]. Available: <http://ejurnal.stmik-budidarma.ac.id/index.php/jurikom|Page|381>
- [15] D. Yuliawati, A. Andriyadi, and Nursiyanto, “Pengujian Sistem Informasi E-Monitoring Pengelolaan Pembangunan Desa Dengan Menggunakan Metode Blackbox Testing,” *IJCCS*, vol. x, No.x, no. 93, 2022.