

**PENGEMBANGAN ALGORITMA GREY WOLF OPTIMIZER UNTUK
REKOMENDASI ITINERARY INFORMASI PARIWISATA BERBASIS
WEBSITE DI DAERAH KOTA MALANG**

TUGAS AKHIR



**TUBAGUS ALFANSYA KAMAL
NIM: 311610017**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS MA CHUNG
MALANG
2023**

LEMBAR PENGESAHAN
TUGAS AKHIR

**PENGEMBANGAN ALGORITMA GREY WOLF OPTIMIZER UNTUK
REKOMENDASI ITINERARY INFORMASI PARIWISATA BERBASIS
WEBSITE DI DAERAH KOTA MALANG**

Oleh:


TUBAGUS ALFANSYA KAMAL
NIM. 311610017

dari:


PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS dan TEKNOLOGI
UNIVERSITAS MA CHUNG

Telah dinyatakan lulus dalam melaksanakan Tugas Akhir sebagai syarat kelulusan
dan berhak mendapatkan gelar Sarjana Komputer (S.Kom.)


Dosen Pembimbing I,


Hendry Setiawan, ST., M.Kom.
NIP. 20100006

Dosen Pembimbing II,


Windra Swastika, S.Kom., MT., Ph.D.
NIP. 20070039

Dekan Fakultas Sains dan Teknologi,


Dr. Kestiana Mega Prilianti, M.Si.
NIP. 20120035

SURAT PERNYATAAN ORISINALITAS

Dengan ini saya menyatakan bahwa seluruh isi dari tugas akhir saya yang berjudul “Pengembangan Algoritma Grey Wolf Optimizer Untuk Rekomendasi Itinerary Informasi Pariwisata Berbasis Website di Daerah Kota Malang” merupakan hasil karya intelektual mandiri yang diselesaikan tanpa menggunakan konten-konten yang illegal dan bukan merupakan karya orang lain yang saya klaim sebagai karya saya.

Referensi telah dicantumkan di daftar pustaka. Saya bersedia menerima sanksi sesuai peraturan yang berlaku apabila tugas akhir saya terbukti bukan hasil karya saya dan merupakan hasil karya orang lain.

Malang, 18 Juli 2023



Tubagus Alfansya Kamal

311610017

Pengembangan Algoritma Grey Wolf Optimizer Untuk Rekomendasi Itinerary Informasi Pariwisata Berbasis Website di Daerah Kota Malang

Tubagus Alfasnya Kamal
311610017

Abstrak

Saat ini Sektor Pariwisata di Kota Malang sangat jarang diketahui oleh masyarakat luar. Kota Malang hanya dikenal dengan Kota Transit. Padahal ada banyak sekali Pariwisata yang ada di Kota Malang, Pada penelitian ini akan membuat website Pariwisata Kota Malang yang bisa membantu Pemerintah Kota Malang dalam upaya menjadikan Kota Malang dikenal oleh masyarakat sebagai Kota Pariwisata. Tetapi Website akan menjadi tempat untuk dikembangkannya sebuah Itinerary yang hasilnya itu sudah di optimalkan menggunakan Algoritma Optimasi. Pada Itinerary tersebut user akan mendapatkan sejumlah informasi untuk berwisata di kota malang, mulai dari rundown mengunjungi 8 Destinasi Wisata, Estimasi Harga, Waktu Berkunjung di destinasi wisata. Untuk menentukan *Itinerary* ini akan menggunakan algoritma *Grey Wolf Optimizer*, agar hasil dari *itinerary* -nya bisa optimal dalam hal mencari rute pada *Itinerary* tersebut. Itinerary ini bertujuan membantu masyarakat untuk mencari destinasi wisata yang cocok untuk para wisatawan dan karena untuk melakukan Itinerary tersebut dibutuhkan Website sebagai wadah Itinerary serta untuk menyebarkan Details Destinasi Wisata yang ada di Kota Malang. Untuk jarak yang ditemukan menggunakan metode tarik langsung dari titik koordinat ke titik koordinat tempat wisata lainnya. Untuk pengembangan bisa menggunakan google maps API agar bisa menemukan jarak yang tepat dan akurat sesuai dengan keadaan di dunia nyata. Dilakukan juga pengujian Turning Parameter yang berisikan parameter Iterasi dan Populasi, sebagai contoh terdapat 5 Iterasi dan 5 Populasi yaitu 50, 100,150,200,dan 250. Dalam pengujian Tuning Parameter semakin tinggi atau semakin banyak Iterasi dan Populasinya maka akan semakin baik pencarian Itinerary, karena Grey Wolf yang menjadi populasi semakin banyak jadi untuk hasilnya semakin optimal.

Kata Kunci: Itinerary, Pariwisata, Promosi, Website

Development of the Gray Wolf Optimizer Algorithm for Website-Based Tourism Information Itinerary Recommendations in the City of Malang

Tubagus Alfansya Kamal
311610017

Abstract

Currently, the Tourism Sector in Malang City is not well known by outsiders. Malang City is only known as a Transit City. However, there are numerous tourist attractions in Malang City. This research aims to create a Tourism website for Malang City, which can help the Malang City Government in promoting Malang as a Tourism City to the public. The website will serve as a platform to develop an Itinerary that is optimized using the Grey Wolf Optimizer algorithm. The Itinerary will provide users with information on visiting 8 Tourist Destinations, including estimated costs and visiting times for each destination. To determine this Itinerary, the Grey Wolf Optimizer algorithm will be employed to find the optimal route. The main objective of this Itinerary is to assist people in finding suitable tourist destinations for travelers. A website is needed to facilitate the creation of the Itinerary and to disseminate details of the available tourist destinations in Malang City. The distances between destinations will be calculated using the direct pulling method from one coordinate point to another. For development purposes, the Google Maps API can be utilized to accurately find the distances according to real-world conditions. Furthermore, Turning Parameter testing will be conducted, including Iteration and Population parameters. For instance, 5 Iterations and 5 Populations will be tested, such as 50, 100, 150, 200, and 250. The results of this testing will help optimize the Itinerary search. The larger the Iteration and Population values, the better the Itinerary search, as a larger number of Grey Wolves in the population will lead to more optimal results.

Keywords: Itinerary, Promotion, Tourist, Website

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa karena atas rahmat dan hidayah-Nya sehingga penelitian ini dapat berjalan dengan baik. Laporan ini disusun untuk memenuhi salah satu persyaratan dalam menyelesaikan tugas akhir bagi mahasiswa dari Program Studi Teknik Informatika Universitas Ma Chung Malang.

Tugas akhir dilaksanakan guna menjadi sarana pengujian terhadap ilmu yang telah diserap selama menempuh pendidikan di Universitas Ma Chung, Malang. Atas dukungan moral dan materi yang telah diberikan dalam penyusunan laporan ini, penulis menyampaikan ucapan terima kasih kepada:

1. Orang tua dan keluarga yang senantiasa memberi dukungan hingga akhir.
2. Hendry Setiawan, ST., M.Kom., Bapak Windra Swastika, S.Kom., MT. selaku pembimbing dan penguji.
3. Bapak Paulus Lucky Tirma Irawan, S.Kom., MT., Bapak Mochamad Subianto, S.Kom., M.Cs., Bapak Windra Swastika, Ph.D., Ibu Dr. Kestrilia Rega Prilianti, M.Si, Ibu Reyna Marsya Quita, S.Si, M.Sc., Bapak Hendry Setiawan, ST., M.Kom., Bapak Dr.Eng. Romy Budhi, ST., MT., Ir. Oesman Hendra Kelana, M.Cs., dan Bapak Rudy Setiawan., S.Si., M.T. yang telah mendampingi proses studi saya selama ini.
4. Danny Gunawan, Muh. Lucki Worldan, Yusuf Giovanno, dan segenap rekan-rekan yang telah memberi dukungan moral dan membantu dalam pengambilan data.

Susunan laporan tugas akhir ini dibuat dengan sebaik-baiknya, namun tidak menutup kemungkinan adanya kesalahan dan kekurangan. Oleh karena itu penulis menerima kritik dan saran dengan senang hati.

Malang, 15 Maret 2023

Tubagus Alfansya Kamal

DAFTAR ISI

KATA PENGANTAR	I
DAFTAR ISI	II
DAFTAR GAMBAR	IV
DAFTAR RUMUS	VI
DAFTAR TABEL	VII
BAB I	1
Pendahuluan	1
1.1 Latar Belakang Masalah	1
1.2 Identifikasi Masalah	3
1.3 Batasan Masalah	3
1.4 Perumusan Masalah	3
1.5 Tujuan Penelitian	4
1.6 Luaran	4
1.7 Manfaat Penelitian	4
1.8 Sistematika Penulisan	5
BAB II	6
Tinjauan Pustaka	6
2.1 Konsep Dasar Website	6
2.1.1 HTML	6
2.1.2 URI	7
2.1.3 HTTP	7
2.1.4 Unsur -Unsur Website.....	8
2.1.5 Web Server.....	9
2.1.6 Web Browser.....	9
2.2 Pariwisata	10
2.2.1 Pariwisata Kota Malang.....	10
2.3 Algoritma Optimasi Metaheuristik	15
2.4 Grey Wolf Optimizer.....	18
2.5 Itinerary.....	20
2.4 Agile.....	21
BAB III.....	23

Analisis dan Perancangan Sistem	23
3.1 Tahapan Penelitian	23
3.1.1 Tahap Pertama.....	24
3.1.2 Tahap Kedua.....	25
3.1.3 Tahap Ketiga.....	25
3.2 Perancangan Antarmuka Pengguna	26
3.3 Perancangan Basis Data	29
3.4 Perancangan Aplikasi	30
3.5 Penggunaan Algoritma Grey Wolf Optimizer	31
3.6 Pengujian Aplikasi	34
BAB IV	36
Hasil dan Pembahasan	36
4.1 Hasil Tahap Pertama	36
4.2 Hasil Tahap Kedua	40
4.3 Pengujian Grey Wolf Optimizer	50
4.2.1 Pengujian Tuning Parameter.....	50
4.2.2 Pengujian Jarak antar GWO.....	51
BAB V	57
Kesimpulan dan Saran	57
5.1 Kesimpulan	57
5.2 Saran	57

DAFTAR GAMBAR

Gambar 2.1 Balaikota Malang.....	11
Gambar 2.2 Alun - Alun Malang.....	12
Gambar 2.3 Hawai Waterpark.....	12
Gambar 2.4 Kampung Warna Warni Jodipan.....	13
Gambar 2.5 Masjid Agung Jami.....	13
Gambar 2.6 Malang Tempo Doeloe.....	14
Gambar 2.7 Orem - Orem Malang.....	15
Gambar 3.1 Tahapan Penelitian menggunakan metode Agile.....	20
Gambar 3.2 Halaman Awal.....	23
Gambar 3.3 Halaman Destinasi.....	23
Gambar 3.4 Pemetaan.....	24
Gambar 3.5 Halaman Login.....	24
Gambar 3.6 Halaman Tabel Data.....	25
Gambar 3.7 Halaman Tambah atau Sunting Data.....	25
Gambar 3.8 Rancangan Basis Data Aplikasi.....	26
Gambar 3.9 Diagram Alur Penggunaan Aplikasi.....	27
Gambar 3.10 Flowchart Grey Wolf Optimizer.....	29
Gambar 3.11 Desain Penggunaan Algoritma Grey Wolf Optimizer.....	30
Gambar 4.1 Tampilan Home.....	32
Gambar 4.2 Tampilan Destination.....	33
Gambar 4.3 Tampilan Maps.....	33
Gambar 4.4 Tampilan mencari Itinerary.....	34
Gambar 4.5 Tampilan mendapatkan Itinerary.....	34
Gambar 4.6 Database.....	35
Gambar 4.7 Table categories.....	35
Gambar 4.8 Table destination.....	36
Gambar 4.9 Evaluasi Fitness.....	37
Gambar 4.10 Initialize Population.....	38
Gambar 4.11 Posisi Alpha, Beta, dan Delta.....	39
Gambar 4.12 Update Posisi Alpha, Beta, dan Delta latitude.....	40
Gambar 4.13 Update Posisi Alpha, Beta, dan Delta longitude.....	42

Gambar 4.14 Menemukan Best Solution.....	4
--	---

DAFTAR RUMUS

Rumus <u>2.1 Nilai D</u>	16
Rumus <u>2.2 Nilai $X(t+1)$</u>	16
Rumus <u>2.3 vektor A</u>	16
Rumus <u>2.4 vektor C</u>	16
Rumus <u>2.5 menghitung posisi grey wolf</u>	16
Rumus <u>2.6 Nilai X_1, X_2, dan X_3 grey wolf</u>	16
Rumus <u>2.7 posisi terbaik Grey Wolf</u>	1

DAFTAR TABEL

Tabel 2.1 <u>Itinerary</u>	17
Tabel 4.1 <u>Pengujian Tuning Parameter</u>	45
Tabel 4.2 <u>Pengujian Jarak Tuning Parameter 50 iterasi dan 50 populasi</u>	46
Tabel 4.3 <u>Pengujian Jarak Tuning Parameter 100 iterasi dan 100 populasi</u>	47
Tabel 4.4 <u>Pengujian Jarak Tuning Parameter 150 iterasi dan 150 populasi</u>	49
Tabel 4.5 <u>Pengujian Jarak Tuning Parameter 200 iterasi dan 200 populasi</u>	51
Tabel 4.6 <u>Pengujian Jarak Tuning Parameter 250 iterasi dan 250 populasi</u>	53

BAB I

Pendahuluan

1.1 Latar Belakang Masalah

Saat ini sektor pariwisata menjadi salah satu sektor yang ada di Indonesia sebagai pemasukan ekonomi terbesar. Setiap wilayah di Indonesia pasti memiliki sektor pariwisata masing - masing yang di kelola oleh Dinas Pemuda, Olahraga, dan Pariwisata atau biasa disingkat menjadi Disporapar. Termasuk Kota Malang juga di kelola oleh Disporapar untuk sektor pariwisata. Sektor Pariwisata tergabung jadi satu dengan sektor Ekonomi Kreatif dimana kedua sektor tersebut sering beririsan. Saat ini Disporapar Kota Malang mengelola lebih dari 30 pariwisata yang ada di kota malang, dari Wisata Sejarah, Wisata Museum, Wisata Taman, Wisata Kuliner, dan saat ini yang paling populer di Kota Malang yaitu Kampung Wisata. Untuk Kampung Wisata ini Kota Malang memilih 25 Kampung Wisata yang aktif salah satunya yang terkenal adalah Kampung Wisata Warna Warni Jodipan. Pariwisata di Kota Malang kurang di kenal oleh masyarakat diluar Kota Malang bahkan orang dari Kota Malang pun juga ada yang kurang mengetahui pariwisata di Kota Malang itu apa saja. Mayoritas masyarakat hanya mengetahui pariwisata yang ada di Kota Batu dan Kabupaten Malang saja, Kota Malang hanya di kenal sebagai Kota Transit. (Febrianto Vicki, 2018). Dengan kurang dikenalnya pariwisata Kota Malang melakukan segala cara untuk memperkenalkan pariwisata di Kota Malang. Program dari Pemerintah Kota Malang adalah melakukan promosi di Sosial Media. Pada Sosial Media ini Pemerintah Kota Malang ingin menjangkau juga masyarakat di luar Kota Malang. Karena saat ini platform terbesar untuk melakukan promosi itu adalah Sosial Media. Upaya lainnya yang dilakukan Pemerintah Kota Malang adalah membentuk Kelompok Sadar Wisata (Pokdarwis) setiap daerah. Pokdarwis ini nantinya yang akan membantu masyarakat yang ada di daerahnya terdapat tempat wisata menyalurkan aspirasi ke Pemerintahan Kota Malang langsung, dengan begitu Pemerintah Kota Malang dan masyarakat Kota Malang dapat berkerja sama satu sama lain untuk membangun pariwisata yang ada di Kota Malang. Ada juga upaya yang di buat oleh Pemerintah Kota Malang yaitu membantu mengembangkan UMKM yang ada di setiap daerah, contohnya seperti di Kampung Keripik Tempe Sanan. Di Kampung Keripik Tempe Sanan tersebut warganya membuat Keripik Tempe yang dijadikan

pusat oleh-oleh Kota Malang. Dengan adanya UMKM seperti ini Pariwisata Kota Malang juga akan meningkat dalam hal pengunjung maupun dari ekonomi masyarakatnya. Upaya yang terakhir dari Pemerintah Kota Malang adalah menggandeng anak – anak muda Kota Malang untuk menjadi promotor pariwisata di Kota Malang atau biasa disebut Duta Wisata. Kota Malang memiliki Duta Wisata yang disebut sebagai Kakang Mbakyu Kota Malang. Kakang dan Mbakyu ini diharapkan dapat menjadi garda terdepan promosi pariwisata yang ada di Kota Malang. Tetapi meskipun udah banyak upaya yang dilakukan oleh Pemerintah Kota Malang masih banyak masyarakat yang masih menganggap Kota Malang hanya sebagai Kota Transit saja. Dengan begitu penulis pada penelitian ini ingin membuat website Pariwisata Kota Malang yang bisa membantu Pemerintah Kota Malang dalam upaya menjadikan Kota Malang dikenal oleh masyarakat sebagai Kota Pariwisata.

Website Pariwisata Kota Malang ini akan sangat membantu masyarakat yang ingin liburan di Kota Malang. Pada Website ini nanti akan ada halaman yang berisikan semua informasi mengenai semua wisata yang ada di kota malang, mulai dari wisata kuliner sampai ke kampung wisata. Dengan begitu masyarakat bisa mengetahui tempat mana saja yang ingin dikunjungi sesuai dengan minat wisata para masyarakat. Terdapat juga halaman lain yang akan berisikan tentang acara terbaru yang akan diselenggarakan oleh Kota Malang mengenai Pariwisata di Kota Malang. Pada halaman berikutnya terdapat rekomendasi untuk para pengunjung mengenai wisata apa saja yang harus di kunjungi, yang dimana nantinya akan mengeluarkan *Itinerary*. *Itinerary* ini akan membantu pengunjung untuk menentukan destinasi selanjutnya juga. *Itinerary* ini akan di dapatkan dari informasi yang di dapatkan dari pengunjung, contoh informasinya seperti ketika di Kota Malang menginap di hotel apa atau di daerah mana, dengan begitu website akan merekomendasikan ke tempat wisata yang dekat dengan tempat menginap para pengujung. Untuk menentukan *Itinerary* ini akan menggunakan algoritma optimasi *metaheuristic*, agar hasil dari *Itinerary*-nya bisa optimal dalam hal mencari rute pada *Itinerary* tersebut.

Website Pariwisata ini bertujuan untuk membantu para pengunjung atau masyarakat lebih mengetahui apa saja wisata yang di Kota Malang serta Website ini juga bertujuan untuk membantu upaya Pemerintah Kota Malang dalam hal mempromosikan atau memperkenalkan Pariwisata yang ada di Kota Malang. Dengan begitu harapannya dapat menghapus juga stigma masyarakat mengenai Kota Malang hanya sebagai Kota Transit saja.

1.2 Identifikasi Masalah

Dari beberapa uraian yang dikemukakan pada latar belakang, maka terdapat masalah yang bisa diidentifikasi. Karena Kota Malang yang sering dikenal sebagai Kota Transit jadi tidak sedikit orang yang memang tidak mengetahui Pariwisata yang ada di Kota Malang. Ketika wisatawan luar Kota Malang masuk ke dalam Kota Malang pasti kebingungan mau kemana, karena masih kurangnya informasi mengenai Pariwisata yang ada di Kota Malang, jikalau ada wisatawan yang sudah mengetahui tujuan destinasi wisata mereka ketika berada di Kota Malang tetapi wisatawan tersebut masih tidak mengetahui bagaimana rute untuk sampai ke tujuan destinasi wisata tersebut. Sehingga dapat menumbuhkan masalah baru lagi, karena wisatawan pasti tidak lama berada di Kota Malang untuk liburan.

1.3 Batasan Masalah

Batasan masalah dalam penelitian ini sebagai berikut:

1. Website ini hanya mencakup area wisata di Kota Malang.
2. Wisata yang ada di Kota Malang berupa Wisata Kampung Tematik, Wisata Air, Wisata Religi, Wisata Heritage, Wisata Kuliner, Wisata Taman, dan Wisata Event
3. Itinerary hanya akan menampilkan satu hari berada di Kota Malang.

1.4 Perumusan Masalah

Berdasarkan batasan masalah yang sudah dipilih oleh penulis maka dapat dirumuskan permasalahan sebagai berikut:

Bagaimana cara mengembangkan system Itinerary otomatis untuk Destinasi Wisata yang ada di Kota Malang?

1.5 Tujuan Penelitian

Penelitian ini juga bertujuan sebagai berikut:

Dengan Pengaplikasian sistem algoritma optimasi *Grey Wolf Optimizer* dapat membantu mencari *Itinerary* dengan jarak terdekat untuk mengunjungi Destinasi Wisata yang ada di Kota Malang. Dengan adanya algoritma *Grey Wolf Optimizer*, wisatawan dapat mengetahui tujuan destinasi wisata yang ada di Kota Malang dengan optimal.

1.6 Luaran

Luaran dari penelitian ini yakni berupa aplikasi website wisata Kota Malang yang bisa membuat rekomendasi *Itinerary* untuk wisatawan berkunjung ke wisata Kota Malang.

1.7 Manfaat Penelitian

Manfaat yang dapat diambil dari penelitian ini ialah sebagai berikut:

1. Bagi Universitas Ma Chung dan Program Studi Teknik Informatika ialah dapat mencetak lulusan yang kompeten dan bermanfaat bagi lingkungan sekitar dan sesuai visi, misi Ma Chung, dan 12 nilai Ma Chung.
2. Bagi penulis ialah mendapatkan kesempatan untuk menerapkan ilmu yang telah dipelajari baik dalam lingkup universitas maupun di luar lingkup universitas untuk menambah wawasan dan membuat aplikasi website rekomendasi yang bisa berguna bagi warga pemukiman serta memperkenalkan sistem rekomendasi *Itinerary* secara otomatis yang menggunakan Sistem Optimasi Metaheuristik kepada wisatawan atau masyarakat.
3. Bagi peneliti selanjutnya ialah mendapat referensi untuk melakukan penelitian dan menjadikan penelitian penulis sebagai pembanding untuk membuat hasil penelitian lebih baik.
4. Bagi masyarakat ialah mendapatkan rekomendasi perjalanan wisata atau *itinerary* yang ada di Kota Malang.

1.8 Sistematika Penulisan

Penulisan laporan tugas akhir ini ditulis dengan sistematika penulisan sebagai berikut:

1. Bab I Pendahuluan

Bab 1 akan dibahas tentang latar belakang masalah mengenai Pengembangan Teknik Optimasi *Grey Wolf Optimizer* untuk memberikan rekomendasi kepada masyarakat untuk berkunjung ke tempat wisata yang ada di Kota Malang.

2. Bab II Tinjauan Pustaka

Pada bab ini akan dibahas tentang teori – teori mengenai Wisata Kota Malang, *Grey Wolf Optimizer*, *Itinerary*, dan Agile

3. Bab III Analisis dan Perancangan Sistem

Pada bab ini akan dibahas tentang analisis kebutuhan, analisis masalah, sistem *Grey Wolf Optimizer*, Use Case tentang sistem aplikasi, ERD tentang sistem aplikasi dan aplikasi website Wisata Kota Malang

4. bab IV Hasil dan Pembahasan

Pada bab ini akan dibahas tentang hasil dan juga pembahasan dari pembuatan serta pengujian alat. Pengujian dilakukan untuk mengetahui seberapa baiknya aplikasi website dan alat bekerja.

5. bab V Simpulan dan Saran

Pada bab ini akan dibahas dari penelitian yang telah dilakukan dan saran untuk peneliti selanjutnya yang akan membuat atau mengembangkan hasil penelitian ini ke jenjang berikutnya.

BAB II

Tinjauan Pustaka

2.1 Konsep Dasar Website

Situs adalah kumpulan halaman yang berisi data tertentu dan dapat diakses dengan mudah oleh siapa saja dan dimana saja melalui web. Tim Berners-Lee membuat situs web pertama di dunia dalam proyek World Wide Web (W3) pada akhir tahun 1980-an. Situs juga dibagi menjadi dua arah pertemuan. Situs tersebut dikenal sebagai situs unik dan situs statis. Situs yang kuat adalah situs yang terutama diharapkan akan cukup sukses seperti yang diharapkan. Situs statis adalah situs yang memiliki halaman konten yang tidak berubah. Situs tersebut resmi diluncurkan secara online pada 6 Agustus 1991 dengan URL <http://info.cern.ch> (Ariffudin, 2023). Tim Berners-Lee menguraikan tiga kemajuan situs dasar, khususnya HTML, URI, dan HTTP

2.1.1 HTML

Hyper Text Markup Language (HTML) adalah bahasa yang digunakan untuk membuat halaman situs. HTML adalah bahasa pemrograman yang digunakan untuk membuat halaman situs, yang diakses melalui web. Hypertext menyinggung hyperlink yang mungkin terkandung dalam HTML. Bahasa markup ini mengikuti bagaimana label digunakan, untuk mencirikan desain halaman dan komponen di dalam halaman. Kode dan simbol yang terkandung dalam file atau dokumen adalah dasar dari kompilasi HTML. Sehingga dapat ditampilkan dengan baik di layar komputer. Selanjutnya, dapat dirasakan oleh klien web. Kemampuan HTML untuk membuat halaman situs yang dapat dibaca dan dilihat lebih banyak lagi tanpa masalah. Semua halaman situs di Web dibuat dengan HTML dan tidak ada kasus khusus. HTML disusun pada halaman laporan dengan label atau gambar tertentu. Gambar dan label ini akan membuat teks terlihat mencolok, miring, bergaris bawah, dll. Misalnya, jika Anda membuat teks menjadi teks miring atau miring. Situs yang dibuat pasti memiliki beberapa elemen. Dibuat menggunakan konten java untuk mengontrol perilaku web, eksekusi bahasa pemrograman server PHP, dan merancang web menggunakan CSS. Jika bahasa HTML berfungsi sebagai dasar untuk web, salah satu dari bahasa ini dapat digunakan. Dalam kebanyakan kasus, kami tidak dapat langsung menempatkan tabel, gambar, atau video di halaman blog atau situs web. Sehingga dibutuhkan

bagian-bagian yang diletakkan di web dengan menggunakan bahasa HTML. Kemampuan HTML untuk memeriksa bagian-bagian situs seperti header, footer, primary, route, dll. Selain itu, HTML juga digunakan sebagai bahasa dalam pembuatan struktur terkomputerisasi atau struktur online.

2.1.2 URI

Uniform Asset Identifier (URI) adalah lokasi khusus untuk membuka halaman situs. URI ini secara efektif membedakan aset yang ada di web. URI saat ini sering disinggung sebagai Uniform Asset Finders (URL). URI memberikan metode dasar yang dapat diperluas untuk membedakan aset web. Namun, lebih dari itu, URI juga dapat mengidentifikasi aset lain seperti Arsip, Dokumen, Buku, dan Gambar. URI memiliki dua himpunan bagian atau jenis khususnya URL dan URN. URL tidak hanya menjelaskan cara menjangkau sumber daya, tetapi juga mengidentifikasinya. URL dapat menunjukkan lokasi video, gambar, situs web, atau konten online lainnya. URN adalah pengidentifikasi yang tepat yang terkait dengan suatu area, yang menyiratkan bahwa URN dapat mengenali aset selamanya terlepas dari apakah informasi tersebut dipindahkan ke area lain.

2.1.3 HTTP

HTTP adalah konvensi jaringan lapisan aplikasi yang dibuat untuk membantu siklus pertukaran antar PC. Konvensi ini berguna untuk mengirimkan data seperti catatan, dokumen, tumpul, dan rekaman antar PC. Protokol Hypertext Transfer Protocol (HTTP) mengambil banyak sumber daya dari sebuah tautan, jenis file yang berfungsi sebagai referensi ke file atau direktori lain, seperti namanya. Konvensi HTTP memberikan banyak perintah untuk mengatur korespondensi. PC klien atau sebaliknya, dalam komunikasi antar jaringan. Komputer klien dan server web, atau sebaliknya, bertukar informasi. Dalam korespondensi ini, PC klien membuat ajakan dengan membuka alamat IP atau ruang (URL). Kemudian, pada saat itu, server web menangani ajakan seperti yang ditunjukkan oleh kode yang dimasukkan. Kemampuan HTTP adalah untuk mengatur konfigurasi dan bagaimana informasi dikomunikasikan. HTTP juga berfungsi untuk mengarahkan bagaimana server web dan program memproses berbagai pesanan yang akan datang. Kemampuan lain adalah mendapatkan informasi dari perampok dan pemrogram. Ini dipisahkan oleh pengembangan HTTPS (Hypertext Move Convention Secure). Kemampuannya mirip,

yang mengatur bagaimana informasi ditangani. Sederhananya, HTTPS adalah versi protokol yang lebih aman daripada HTTP.

2.1.4 Unsur - Unsur Website

Terdapat tiga unsur pada website. Tanpa adanya tiga unsur ini, situs website tidak bisa ditemukan maupun diakses oleh pengguna internet. Tiga unsur pada Website adalah Domain, Hosting, dan Bahasa Pemrograman

A. Domain

Domain adalah alamat sebuah website. Tercatat dalam sejarah domain, mulanya untuk mengunjungi suatu situs web client perlu mengetahui alamat IP atau IP Address yang ditandai deretan angka.

B. Hosting

Hosting adalah server tempat dimana semua file website client disimpan serta dapat diakses dan dikelola melalui internet. Web Hosting bisa diibaratkan sebuah rumah dan website adalah seluruh isi rumah tersebut, mulai dari gambar, video, teks, dan lainnya.

C. Bahasa Pemrograman (Coding)

Coding adalah proses penulisan kode untuk membangun website. Beberapa bahasa yang bisa di coding antara lain CSS untuk menampilkan tampilan elemen website, JavaScript agar website lebih dinamis serta interaktif, dan lainnya.

2.1.5 Web Server

Web Server adalah sebuah software (Perangkat Lunak) yang memberikan layang berupa data. Berfungsi untuk menerima permintaan HTTP atau HTTPS dari klien atau kita kenal dengan web browser (Chrome, Firefox). Selanjutnya ia akan mengirimkan respon atas permintaan tersebut kepada client dalam bentuk halaman web. Web server memiliki peran dalam memproses berbagai data yang diminta oleh client (Web Browser). Kemudian ia memberikan hasil atau jawaban berupa dokumen, video, foto, atau beragam bentuk berkas lainnya. Fungsi lain dari Web Server:

A. Membersihkan berbagai cache yang terdapat pada penyimpanan serta semua dokumen yang tidak terpakai lagi.

B. Melakukan pemeriksaan terhadap sistem keamanan yang berasal dari permintaan HTTP berdasarkan permintaan client atau web browser.

C. Menyediakan data berdasarkan permintaan yang masuk agar dapat menjamin keamanan sistem yang berjalan dengan lancar.

Terdapat beberapa jenis - jenis dari web server seperti Web Server Apache, Web Server Nginx, Web Server IIS, dan Web Server Lighttpd.

2.1.6 Web Browser

Web Browser adalah perangkat lunak yang memungkinkan client untuk mencari, mengakses, dan menampilkan halaman website di internet. Setiap halaman website memiliki alamat unik yang disebut dengan URL. Ketika client mengetik alamat web atau menekan hyperlink, client meminta web browser untuk membawamu ke halaman tersebut. Kemudian web browser akan mencari dan mengambil informasi tentang website tersebut dari web server. Lalu, web server akan mentransfer data yang dibutuhkan menggunakan Hypertext Transfer Protocol, yaitu seperangkat aturan tentang bagaimana file teks, gambar, dan video di transmisi di seluruh web. Data yang di transfer lalu diteruskan dalam HTML. HTML berfungsi untuk memberitahu web browser di mana setiap elemen berada di halaman website. Hal ini akan memastikan kesamaan tampilan halaman website. Fungsi Web Browser adalah mencari informasi di internet dengan efektif, menyimpan data di internet, menampilkan data dengan ekstensi tertentu, mendukung pemakaian search engine, dan melindungi perangkat dari website berbahaya. Jenis Web Browser adalah Google Chrome, Safari, Mozilla Firefox, Microsoft Edge, dan Opera.

2.2 Pariwisata

Pariwisata menurut Undang - Undang Nomor 9 tahun 1990 tentang kepariwisataan adalah segala sesuatu yang berhubungan dengan wisata, termasuk pengusahaan objek dan daya tarik wisata serta usaha - usaha yang terkait di bidang tersebut. Pariwisata adalah serangkaian kegiatan perjalanan yang dilakukan oleh perorangan atau kelompok dari tempat tinggal asalnya ke berbagai tempat lain dengan tujuan melakukan kunjungan wisata dan bukan untuk bekerja atau mencari penghasilan di tempat tujuan. Kunjungan yang dimaksud bersifat sementara dan pada waktunya akan kembali ke tempat tinggal semula. Hal tersebut memiliki dua elemen

yang penting, perjalanan itu sendiri dan tinggal sementara di tempat tujuan dengan berbagai aktivitas wisatanya. Pariwisata merupakan konsep yang sangat multidimensional. Tak bisa dihindari bahwa beberapa pengertian pariwisata dipakai oleh para praktisi dengan tujuan yang ingin di capai. Saat ini bidang pariwisata di bawah naungan Kementerian Pariwisata dan Ekonomi Kreatif. Provinsi Jawa Timur bidang Pariwisata di naungi oleh Dinas Kebudayaan dan Pariwisata Provinsi Jawa Timur dan Pariwisata di Kota Malang di naungi oleh Dinas Kepemudaan, Olahraga, dan Pariwisata.

2.2.1 Pariwisata Kota Malang

Kota Malang merupakan wilayah inti bagi perkembangan wilayah Jawa Timur, khususnya wilayah Malang dan Pasuruan yaitu termasuk Satuan Wilayah Pengembangan (SWP-VI). Untuk mencapai hasil pembangunan yang lebih efisien dan efektif, tentunya setiap wilayah memiliki spesifikasi kegiatan yang saling menunjang dan saling melengkapi. Kota Malang berfungsi sebagai pelayanan pariwisata dimana terdapat beberapa obyek wisata yang ditunjang oleh lokasi geografis Kota Malang yang terletak di antara pegunungan dan pantai. Sebagai Kota Pariwisata. Kota Malang sendiri memiliki beberapa obyek destinasi wisata yaitu,

A. Wisata Heritage

Wisata Heritage merupakan perjalanan wisata yang harus dilakukan ke obyek - obyek wisata dengan maksud untuk mengetahui dan mempelajari nilai sejarah yang ada di Kota Malang. Kota Malang sangat banyak sekali bangunan dan obyek-obyek bersejarah, masih banyak gedung-gedung peninggalan penjajahan belanda dan peninggal kerajaan Singosari. beberapa contoh Wisata Heritage yang ada di Kota Malang adalah Balaikota Malang Monument Tugu, dan Patung Pahlawan Trip.



Gambar 2.1 Balaikota Malang

B. Wisata Taman

Wisata taman merupakan ruang terbuka hijau berupa taman kota, median jalan, hutan kota, atau kebun bibit yang unik dan menarik hingga menjadi salah satu tempat kunjungan wisatawan serta dapat menjadi tempat rekreasi keluarga. Taman- taman di Kota Malang selain dibentuk oleh Pemerintah Kota Malang juga seringkali bekerjasama dengan pihak swasta dalam bentuk Corporate Social Responsibility (CSR). Beberapa contoh Wisata Taman yang ada di Kota Malang adalah Alun - Alun Tugu, Alun- Alun Merdeka, dan Merbabu Family Park.



Gambar 2.2 Alun - Alun Kota Malang

C. Wisata Air

Wisata Air merupakan destinasi wisata berbasis air yang umumnya menjadi tempat rekreasi keluarga. Kota Malang sendiri memiliki beberapa opsi tempat Wisata Air yang dapat dikunjungi seperti Hawaii Waterpark, Wisata Aeng Hamid Rusdi Wonokoyo, dan Taman Rekreasi Tlogomas.



Gambar 2.3 Hawaii Waterpark Malang

D. Wisata Kampung Tematik

Salah satu destinasi wisata yang sangat menarik beberapa tahun terakhir ini adalah Wisata Kampung Tematik. Kampung Tematik yang sudah ada dan berkembang serta menjadi jujukan wisatawan di Kota Malang dan membuat Kota Malang menjadi daerah dengan destinasi kampung tematik terbanyak di Jawa Timur. Beberapa contoh Wisata Kampung Tematik yang ada di Kota Malang adalah Kampung Warna Warni, Kampung Wisata Sanan, dan Kampung Keramik Dinoyo.



Gambar 2.4 Kampung Warna Warni Malang

E. Wisata Religi

Wisata Religi merupakan perjalanan wisata ke tempat lain dengan maksud menyaksikan atau melihat kegiatan / upacara maupun tempat - tempat keagamaan. Beberapa wisata religi di Kota Malang juga tergolong ke dalam bangunan - bangunan cagar budaya. Beberapa contoh Wisata Religi yang ada di Kota Malang adalah Gereja Hati Kudus Yesus Kayutangan, Masjid Agung Jami' Malang, dan Kelenteng Eng An Kiong



Gambar 2.5 Masjid Agung Jami

F. Wisata Event

Sebagai kota wisata, Malang tidak hanya menawarkan wisata tematik atau pun wisata kuliner. Ada beberapa event tahunan yang rutin digelar dan menjadi wisata khas Kota Malang, salah satunya adalah Malang Flower Carnival. Malang Flower Carnival merupakan event karnaval yang digelar di beberapa jalanan Kota Malang dan melewati beberapa rute tertentu. Dalam karnaval ini, seluruh peserta yang berpartisipasi, menggunakan kostum bertema bunga yang berwarna-warni. Selain itu, kostum yang digunakan peserta sebagian besar dibuat dari bahan daur ulang. Dinas Kepemudaan, Olahraga dan Pariwisata (Disporapar) Kota Malang berencana menggelar Malang Flower Carnival pada tanggal 16-17 Oktober 2021. Festival bertemakan bunga yang terbesar di Indonesia itu, rencananya akan dilakukan secara hybrid, yakni dengan online dan offline.

Sementara itu sebelumnya, Malang Flower Carnival berhasil meraih penghargaan Silver Champion kategori Tourism & Creative Campaign Award 2021, dalam acara Planet Tourism Indonesia 2021 yang digelar secara virtual, beberapa waktu lalu. Kegiatan tahunan ini juga menjadi event unggulan Kota Malang yang telah masuk dalam Calendar of Event Kementerian Pariwisata dan Ekonomi Kreatif Republik Indonesia (Kemenparekraf RI) sejak tahun 2017.

Wisata Event di Kota Malang minimal dilakukan satu kali dalam setahun. Wisata Event diadakan untuk membantu meningkatkan UMKM yang ada di Kota Malang dan untuk membantu promosi wisata yang ada di Kota Malang, contoh seperti Malang Tempo Doeloe, Malang Tempo Doeloe adalah Wisata Event yang membantu UMKM yang ada di kota Malang karena di event tersebut terdapat beberapa pelaku

UMKM yang memperjual belikan produknya serta dalam event Malang Tempo doeloe dinas pariwisata Kota Malang ingin masyarakat Kota Malang lebih mengetahui sejarah yang ada di Kota Malang dengan kata lain event tersebut membantu mempromosikan Wisata Heritage yang ada di Kota Malang.



Gambar 2.6 Malang Tempo Doeloe

G. Wisata Kuliner

Kota Malang merupakan Kota yang menarik untuk melakukan Wisata Kuliner, baik makanan maupun minuman. Adapun makanan dan minuman khas yang dapat dinikmati seperti Bakso Bakar, Orem -Orem, Gethuk Talon untuk makananya dan untuk minumannya ada Es Talun dan Es Tepan Ketan.



Gambar 2.7 Orem - Orem Malang

2.3 Algoritma Optimasi Metaheuristik

Algoritma mateheuristik (*metaheuristic*) merupakan gabungan kata Meta dan Heuristic. Heuristic artinya mencari jawaban dengan cara trial dan error, sementara meta berarti melampaui atau tingkat yang lebih. Jadi metaheuristik tidak hanya mencari jawaban dengan coba-coba saja, melainkan dengan bantuan atau arahan dari

faktor-faktor lain. Faktor-faktor itulah yang membuat banyaknya variasi dari metaheuristik. Terdapat banyak algoritma-algoritma metaheuristik yang dikenal dan sudah diterapkan. Terdapat banyak contoh untuk algoritma Metaheuristik, yaitu:

a) Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) adalah algoritma metaheuristik populer yang digunakan untuk memecahkan masalah optimasi. Ini terinspirasi dari perilaku kawanan di alam, seperti kawanan burung atau kawanan ikan. Algoritme didasarkan pada gagasan bahwa partikel individu dalam kawanan dapat berkomunikasi satu sama lain dan bekerja sama untuk menemukan solusi optimal. Algoritma PSO pertama kali diperkenalkan oleh Eberhart dan Kennedy pada tahun 1995. Sejak saat itu, algoritme PSO banyak digunakan di berbagai bidang, termasuk teknik, ekonomi, dan ilmu komputer. Ide dasar dari PSO adalah untuk mensimulasikan perilaku segerombolan partikel, dimana setiap partikel merupakan solusi potensial untuk masalah optimasi. Di PSO, partikel bergerak melalui ruang pencarian untuk menemukan solusi terbaik. Pergerakan setiap partikel dipandu oleh pengalamannya sendiri, serta pengalaman partikel lain dalam kawanan itu. Setiap partikel mencatat solusi terbaiknya sendiri dan juga melacak solusi terbaik yang ditemukan oleh gerombolan sejauh ini. Algoritma PSO dimulai dengan secara acak menginisialisasi segerombolan partikel di ruang pencarian. Setiap partikel diberi posisi dan vektor kecepatan. Posisi mewakili solusi potensial untuk masalah tersebut, sedangkan vektor kecepatan menentukan arah dan kecepatan pergerakan partikel. Selama setiap iterasi algoritme, posisi dan kecepatan setiap partikel diperbarui berdasarkan solusi terbaik pribadinya dan solusi terbaik yang ditemukan oleh gerombolan sejauh ini.

b) Differential Evolution (DE)

Differential evolution adalah satu dari beberapa metode metaheuristik yang pemakaiannya cukup luas dalam bidang rekayasa (Santosa dan Jin Ai, 2017). DE termasuk metode pencarian stokastik dan berdasarkan populasi. DE mempunyai kesamaan dengan evolutionary algorithms (EA) yang lain, tetapi berbeda dalam hal informasi jarak dan arah dari populasi yang sekarang digunakan untuk memandu proses pencarian solusi yang lebih baik. Seperti halnya semua keluarga Evolutionary

Algorithm (EA), DE juga didasarkan pada pembangkitan populasi titik- titik untuk mencapai minimum suatu fungsi. DE agak berbeda dengan algoritma EA dalam hal:

Mutasi diterapkan dulu untuk membangkitkan vector percobaan (trial vector), yang kemudian akan digunakan dalam proses crossover untuk menghasilkan satu turunan (offspring) Step size dalam mutasi tidak disampel dari distribusi populasi yang sudah diketahui. Dalam DE, step size dalam mutasi dipengaruhi oleh perbedaan diantara individu dalam populasi sekarang. Titik-titik ini akan disampling secara random sebagai titik awal. Kemudian perlu dilakukan pembatasan nilai-nilai (batas atas maupun batas bawah) dimana kira-kira nilai variable yang dicari akan berada. Pembangkitan populasi awal tentu saja harus memperhatikan batas bawah dan batas atas ini. Setiap vector (dari titik-titik dalam populasi) diberi indeks untuk menandai posisinya. DE membangkitkan titik baru berdasarkan titik yang ada dan selisih antara dua titik dalam populasi. Ini berbeda dengan metoda direct search seperti Nelder Mead yang membangkitkan titik baru dengan cara refleksi atau melakukan sampling dari suatu distribusi tertentu seperti dalam ES.

c) Algoritma Genetika (GA)

Algoritma Genetika dikembangkan pertama kali oleh John Holland dari New York, Amerika Serikat yang dipublikasikan dalam bukunya yang berjudul “Adaption in Natural and Artificial Systems” tahun 1975. Algoritma Genetika merupakan Teknik untuk menemukan solusi optimal dari permasalahan yang mempunyai banyak solusi. Teknik ini akan melakukan pencarian dari beberapa solusi yang diperoleh sampai mendapatkan solusi terbaik sesuai dengan kriteria yang telah ditentukan atau yang disebut sebagai fungsi fitness. Algoritma ini masuk dalam kelompok algoritma evolusioner dengan menggunakan pendekatan evolusi Darwin di bidang Biologi seperti pewarisan sifat, seleksi alam, mutasi gen dan kombinasi (crossover). Karena merupakan Teknik pencarian optimal dalam bidang ilmu komputer, maka algoritma ini juga termasuk dalam kelompok algoritma metaheuristik.

Aplikasi algoritma genetika dapat ditemukan di berbagai bidang terutama bidang-bidang yang memerlukan solusi kombinatorik seperti penjadwalan, peramalan, jarak terpendek dan kombinasi ransum atau bahan. Algoritma genetika sering dipakai untuk melakukan simulasi dengan Komputer untuk mendapatkan solusi terbaik

berdasarkan calon-calon solusi yang visible. Proses pencarian solusi terbaik dimulai dengan merepresentasikan solusi-solusi yang mungkin terjadi berdasarkan domain yang biasanya dalam bentuk string biner (0 dan 1). Dari representasi ini dibentuk populasi individual secara acak yang membentuk suatu generasi. Kemudian dari setiap populasi yang terbentuk dievaluasi dengan menggunakan fungsi fitness untuk dapat memilih populasi terbaik. Kemudian populasi dimodifikasi dengan mutasi dan kombinasi untuk mendapatkan populasi baru. Proses ini diulang sampai mendapatkan individu dari populasi yang mencapai nilai fitness.

Langkah-langkah algoritma genetika dapat diuraikan sebagai berikut:

1. Penentuan representasi individu dari populasi
2. Pembentukan populasi individu secara acak
3. Evaluasi kecocokan setiap individu dalam populasi berdasarkan fungsi fitness
4. Memilih individu dengan nilai kecocokan paling tinggi
5. Kombinasi antar individu terpilih dalam populasi dan mutasi individu dengan tingkat tertentu untuk membentuk populasi baru
6. Proses 3 diulang sampai mendapatkan solusi terbaik.

2.4 Grey Wolf Optimizer

Grey Wolf Optimizer (GWO) adalah metode kecerdasan metaheuristik yang telah disesuaikan secara luas untuk berbagai masalah optimasi, karena karakteristiknya yang mengesankan dibanding metode kecerdasan segerombolan lainnya. *Grey Wolf Optimizer* memiliki parameter yang sedikit, dan tidak diperlukan informasi derivasi dalam pencarian awal. *Grey Wolf Optimizer* juga sederhana, mudah digunakan, fleksibel, dapat diskalakan, dan memiliki kemampuan khusus mencapai keseimbangan yang tepat antara eksplorasi dan eksploitasi selama pencarian yang mengarah pada konvergensi yang menguntungkan. Oleh karena itu, *Grey Wolf Optimizer* baru - baru ini mendapatkan minat penelitian yang sangat besar dengan khalayak yang luar biasa dari beberapa domain dalam waktu yang sangat singkat.

Grey Wolf Optimizer terinspirasi oleh kepemimpinan sosial dan perilaku berburu serigala abu-abu di alam. Algoritma *Grey Wolf Optimizer* mempertimbangkan tiga pemimpin serigala bernama α , β , dan δ sebagai solusi terbaik untuk memimpin sisa serigala bernama serigala ω menuju area yang menjanjikan untuk menemukan solusi

global. Pelacakan, mengejar, mendekati, dan mengacau mangsa sampai berhenti bergerak lalu menyerang mangsa. Adapun secara rinci perilaku serigala abu-abu dalam berburu dibagi kedalam beberapa tahapan yaitu, melingkari mangsa, berburu, menyerang mangsa (eksploitasi dan menyerang mangsa (eksplorasi). Seperti disebutkan diatas, serigala abu - abu mengelilingi mangsanya selama berburu. Adapun model matematisnya adalah sebagai berikut.

$$D = |C \cdot X_p(t) - X(t)| \quad (2-1)$$

$$X(t + 1) = X_p(t) - A \cdot D \quad (2-2)$$

Vektor A dan C dihitung mengikuti

$$A = 2\alpha \cdot r_1 - \alpha \quad (2-3)$$

$$C = 2\alpha \cdot r_2 \quad (2-4)$$

Sementara untuk berburu, model matematisnya adalah sebagai berikut

$$D_\alpha = |C_1 \cdot X_\alpha - X|, D_\beta = |C_2 \cdot X_\beta - X|, D_\delta = |C_3 \cdot X_\delta - X| \quad (2-5)$$

$$X_1 = X_\alpha - A_1 \cdot (D_\alpha), X_2 = X_\beta - A_2 \cdot (D_\beta), X_3 = X_\delta - A_3 \cdot (D_\delta) \quad (2-6)$$

$$X(t - 1) = \frac{X_1 + X_2 + X_3}{3} \quad (2-7)$$

α = Solusi Terbaik Pertama

D = Posisi Grey Wolf

β = Solusi Terbaik Kedua

X = Hasil Terbaik

δ = Solusi Terbaik Ketiga

t = iteration

A = Vektor A

r = Random Number

C = Vektor C

2.4 Itinerary

Itinerary adalah Rencana Perjalanan untuk yang bisa membantu wisatawan untuk melakukan liburan. *Itinerary* berisikan daftar kegiatan beserta estimasi biaya yang dibutuhkan selama perjalanan ke tempat tujuan. Dengan adanya *Itinerary*, semua kebutuhan selama liburan akan lebih diperhitungkan, mulai perjalanan wisata hingga waktu berwisata. Hal yang diperlukan ketika membuat *Itinerary* adalah melakukan riset, membagi wilayah menjadi beberapa zona, membuat estimasi hari, menentukan transportasi dan akomodasi. Berikut contoh *Itinerary* Pariwisata Kota Malang:

Tabel 2.1 *Itinerary*

Waktu	Aktivitas
-------	-----------

05.00-07.00	Perjalanan ke Kota Malang
07.00-07.15	Perjalanan ke Hotel
07.00-07.30	Istirahat dan Mandi
07.30-07.45	Menuju Kampung Warna Warni
07.45-08.45	Kampung Warna Warni Jodipan
08.45-09.00	Menuju Kampung 3D
09.00-10.00	Kampung 3D
10.00-10.15	Menuju Kampung Biru Arema
10.15-11.15	Kampung Biru Arema
11.15-11.30	Menuju Toko Oen
11.30-13.00	Toko Oen
13.00-13.15	Menuju Alun - Alun Merdeka
13.15-14.15	Alun - Alun Merdeka
14.15-14.30	Menuju Kampung Kayutangan Heritage
14.30-20.00	Kampung Kayutangan Heritage
20.00-21.00	Kembali ke Hotel

2.5 Agile

Agile adalah sekumpulan metode pengembangan software yang dilakukan secara bertahap dan berulang (iterasi). Agile development sering disebut sebagai framework karena di dalamnya memang terdapat berbagai metode yang digunakan. Metode Agile sangat cocok untuk proyek jangka pendek. Dikarenakan metode ini akan sangat mudah beradaptasi apabila terjadi perubahan dalam suatu proyek. Dalam Agile Development, nilai terpentingnya terdapat pada tim yang bisa memutuskan suatu hal dengan cepat dan tepat. Agile Development mempunyai 7 tujuan yaitu:

Kerangka Aplikasi yang Bernilai Tinggi dan Berfungsi

Tujuan utamanya adalah menciptakan kerangka aplikasi yang bernilai tinggi dan berfungsi dengan baik. Artinya, menciptakan produk yang memiliki nilai jual tinggi namun menggunakan biaya produksi yang seminimal mungkin. Meskipun begitu, pentingnya kualitas produk tidak boleh diremehkan.

Iteratif, Bertahap, dan Transformatif

Agile adalah teknik pengembangan produk yang dijalankan secara berulang, berkali-kali, dan dapat diubah sebagian jika diperlukan. Dengan kata lain, pendekatan ini fleksibel dan cocok digunakan dalam proyek IT dengan durasi singkat.

Kontrol Biaya dan Pengembangan yang Bernilai

Siklus pengembangan produk dapat disesuaikan dengan kebutuhan pelanggan. Tim pengembangan juga memiliki kontrol terhadap biaya dan waktu yang diperlukan dalam proses pengembangan perangkat lunak berdasarkan persyaratan.

Produksi yang Hebat

Meskipun menggunakan biaya produksi yang minimum, produk berupa perangkat lunak atau program akan tetap menjaga kualitasnya.

Fleksibel dan Pengelolaan Risiko

Tujuan yang fleksibel terkait dengan menjalankan jadwal dengan pelanggan yang dapat ditetapkan kapan saja. Dengan demikian, potensi produk yang dikembangkan tetap terjaga. Metode ini juga digunakan untuk mengurangi kesalahan program sebelum proses penerapan aplikasi akhirnya selesai.

Kolaborasi

Kolaborasi sangat penting bagi tim pengembangan untuk mengevaluasi umpan balik dari pelanggan. Oleh karena itu, diperlukan koordinasi dan komunikasi yang baik dengan semua anggota tim.

Tim yang Mandiri dan Mandiri

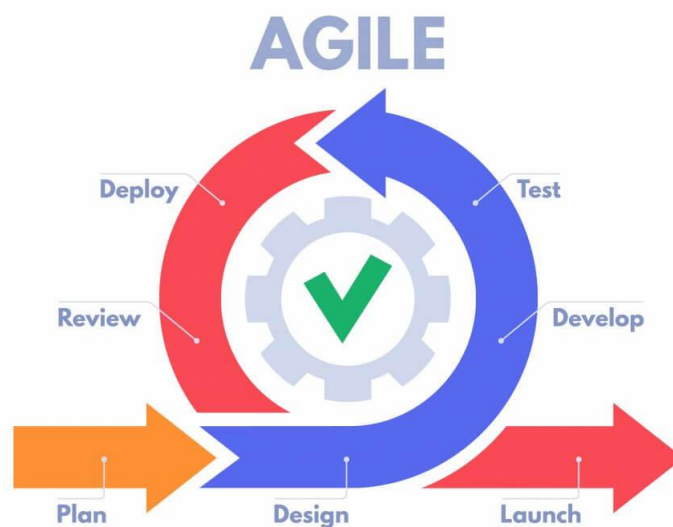
Tujuan utama dari metode Agile adalah memberikan kebebasan bagi para pengembang untuk mengelola proyek pengembangan produk mereka sendiri. Tugas manajer adalah menjadi perantara antara pengembang dan pelanggan. Dengan demikian, kemungkinan terjadinya kesalahpahaman dapat dikurangi.

Bab III

Analisis dan Perancangan Sistem

3.1 Tahapan Penelitian

Penelitian dilakukan menggunakan metode *Software Development Life Cycle Agile*. Metode ini diterapkan karena memiliki tingkat fleksibilitas yang tinggi. Gambar 3.1 berikut menampilkan alur metode *Software Development Life Cycle Agile*.



Gambar 3.1 Tahapan penelitian menggunakan metode agile.

Tahapan awal dimulai dari plan dengan melakukan analisis kebutuhan yang kemudian dilanjutkan dengan design untuk sistem yang akan dibuat. Dalam proses design terdapat proses develop dan test yang akan dilakukan sebelum adanya penyebaran aplikasi. Saat aplikasi telah disebarkan, maka akan dilakukan review guna membuat aplikasi menjadi sebaik mungkin. Semua proses bersifat iteratif atau berulang, sehingga tingkat fleksibilitas menjadi sangat tinggi dan membuat aplikasi dapat lebih cepat dikerjakan dan lebih cepat dilakukan penyesuaian terhadap kebutuhan-kebutuhan baru yang ditemui.

3.1.1 Tahap Pertama

Fase awal dimulai dari tahapan plan yang bisa disebut juga analisis kebutuhan. Dalam pengembangan aplikasi ini, terdapat banyak hal yang dibutuhkan sebagai berikut.

a. Perancangan Antarmuka Pengguna

Antarmuka pengguna dirancang sesuai kebutuhan yang ada. pada dasarnya antarmuka yang dibuat akan menampilkan daftar destinasi yang ada di Kota Malang. Pemetaan grafis destinasi wisata di Kota Malang, dan tampilan untuk rekomendasi destinasi wisata yang sesuai dengan pengguna.

b. Perancangan Database

Database dirancang agar dapat memenuhi segala kebutuhan yang ada dengan baik dan cepat.

c. Pemilihan Bahasa Pemrograman

Bahasa pemrograman sangatlah beragam, setiap Bahasa pemrograman memiliki kelebihan dan kekurangannya masing-masing. Dalam pengembangan aplikasi web, terdapat istilah *front-end* dan *back-end*. Kedua hal ini memiliki peran yang berbeda dan dapat dilakukan menggunakan bahasa pemrograman yang sama ataupun berbeda.

Dalam penelitian ini, bahasa pemrograman untuk sisi *back-end* akan menggunakan bahasa pemrograman PHP dengan kerangka kerja Laravel. Bahasa pemrograman php dipilih karena dengan menggunakan kerangka kerja Laravel aplikasi dapat dibuat dengan lebih terstruktur dan lebih cepat karena terdapat banyak alat-alat yang dapat membantu dalam melakukan pemrograman.

Pada sisi *front-end*, aplikasi ini akan menggunakan Bahasa pemrograman JavaScript dengan library jQuery. Bahasa ini telah banyak digunakan oleh aplikasi-aplikasi raksasa seperti Facebook, Google, Microsoft, dan masih banyak lagi. Bahasa pemrograman JavaScript dikenal cukup rumit dalam penulisan kode perintah nya, hal ini disebabkan oleh perintah-perintah yang digunakan oleh JavaScript cukup panjang. Dengan adanya jQuery, pengkodean menggunakan JavaScript bisa menjadi lebih singkat dan lebih mudah dilakukan.

3.1.2 Tahap Kedua

Tahap kedua terdiri dari perancangan atau pembuatan aplikasi dan pengujian. Pembuatan aplikasi dilakukan secara bertahap sesuai proses alur kerja aplikasi. Setiap menu yang dibuat akan diuji terlebih dahulu fungsi integrasinya terhadap basis data. Pengujian akan meliputi proses CRUD (*Create, Read, Update, Delete*) sesuai kebutuhan tiap menu nya.

Pengujian selanjutnya ialah pengujian rekomendasi destinasi wisata yang terdiri dari komputasi menggunakan algoritma *Grey Wolf Optimizer*. Pengujian ini akan dilakukan dengan cara menjalankan fungsi optimasi yang ada menggunakan titik kordinat dari pengguna aplikasi dan titik kordinat dari destinasi-destinasi wisata yang ada.

3.1.3 Tahap Ketiga

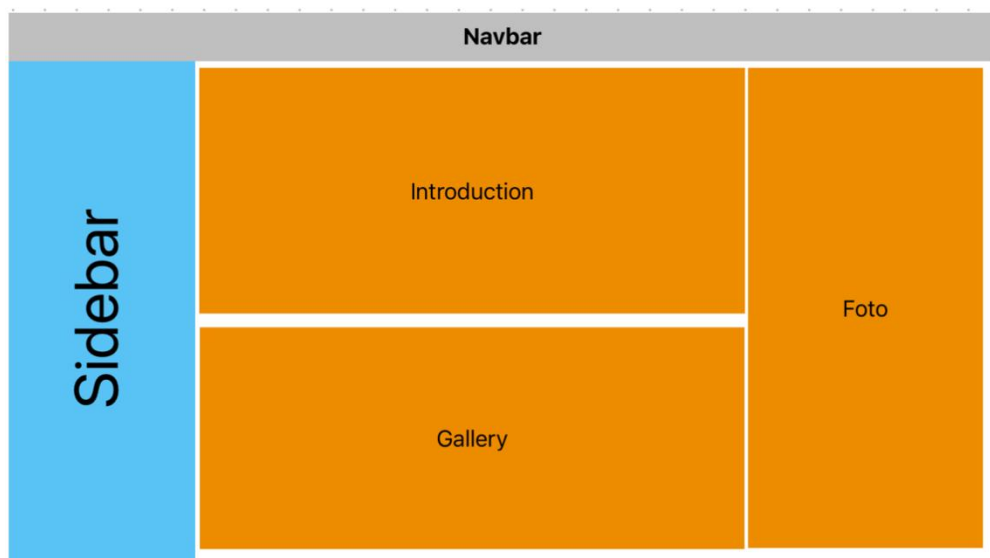
Setelah aplikasi selesai dibuat, tahapan pengujian akan dilakukan dengan melakukan pengujian integrasi sistem yang akan dilakukan oleh penulis dan dilanjutkan dengan revisi mandiri oleh penulis. Saat pengujian mandiri telah selesai direvisi. Maka akan dilanjutkan dengan pengujian pada pihak ketiga yakni responden lain yang akan dilakukan pengujian kualitatif mengenai kepuasan pengguna terhadap aplikasi.

3.2 Perancangan Antarmuka Pengguna

Antarmuka pengguna terdiri dari beberapa bagian seperti:

1. Halaman Awal

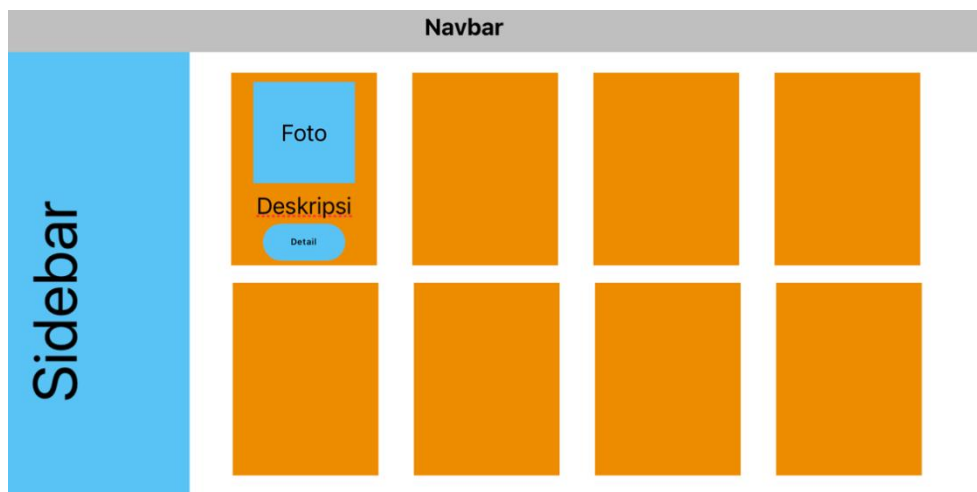
Halaman awal terdiri dari pengenalan mengenai Kota Malang dan beberapa galeri foto wisata di kota malang seperti pada gambar 3.2 berikut.



Gambar 3.2 Halaman Awal

2. Destinasi

Halaman destinasi berisi daftar destinasi dalam bentuk kartu dengan foto, deskripsi dan tombol detail seperti pada gambar 3.3 berikut.



Gambar 3.3 Halaman Destinasi

3. Pemetaan

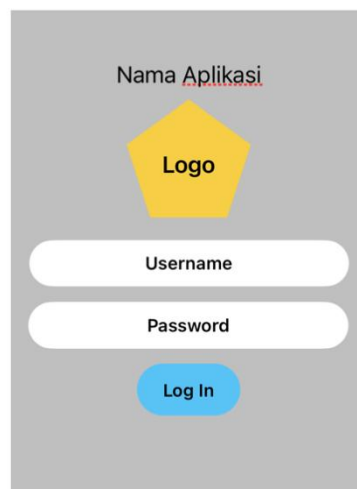
Halaman pemetaan berisi sebuah peta digital kota malang menggunakan leaflet js dengan destinasi destinasi wisata yang ada seperti pada gambar 3.4 berikut.



Gambar 3.4 Pemetaan

4. Halaman Login

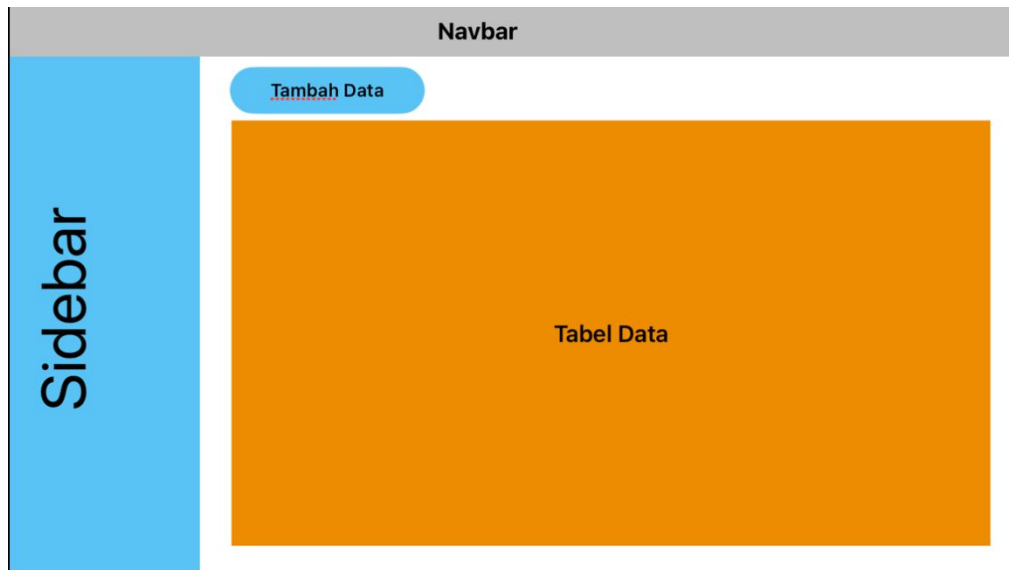
Halaman login berisi masukan pengguna berupa nama pengguna dan password seperti pada gambar 3.5 berikut.



Gambar 3.5 Halaman Login

5. Halaman Tabel Data

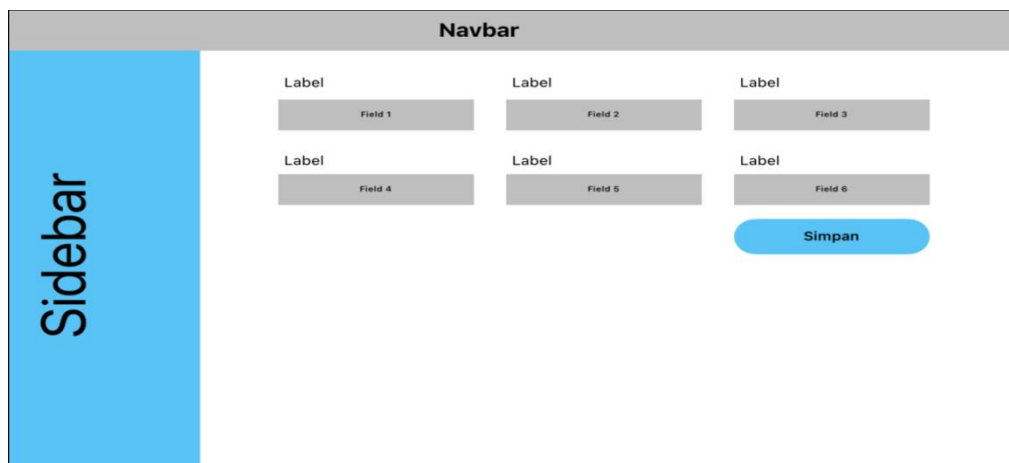
Halaman tabel data berisi sebuah tabel yang menampilkan data kategori dan data destinasi pada masing-masing menu nya seperti pada gambar 3.6 berikut.



Gambar 3.6 Halaman Tabel Data

6. Halaman Tambah / Sunting Data

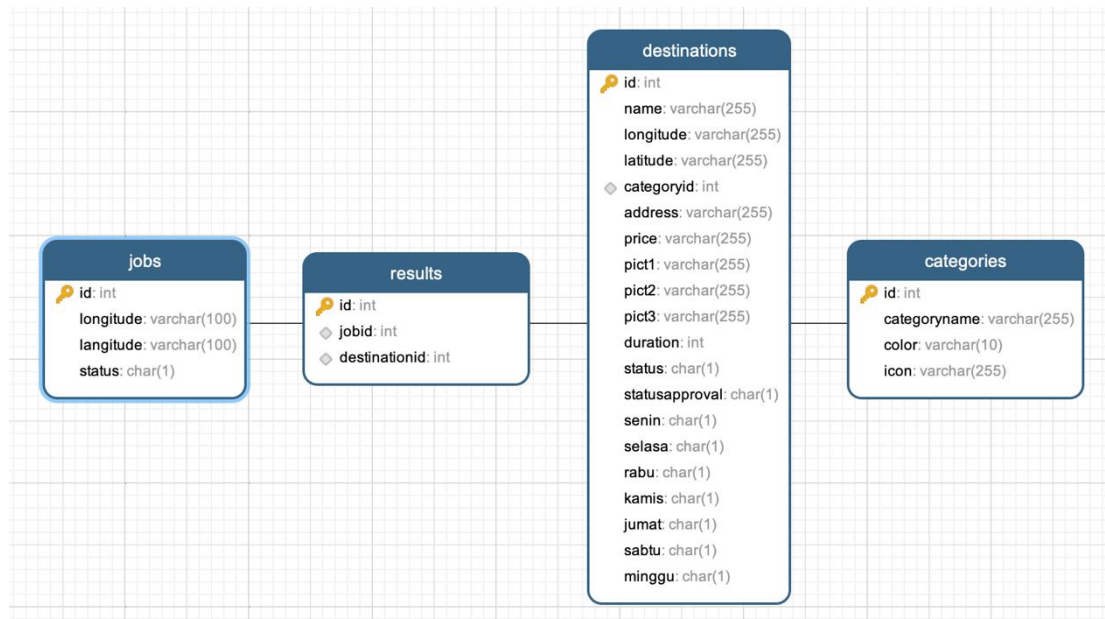
Halaman tambah / sunting data berisi kolom-kolom mengenai data yang ada untuk ditambahkan jika belum ada, dan disunting jika sudah ada seperti gambar 3.7 berikut.



Gambar 3.7 Halaman Tambah atau Sunting Data.

3.3 Perancangan Basis Data

Database memainkan peran penting dalam pengembangan aplikasi, untuk itu adanya rancangan basis data yang baik sangat dibutuhkan. Berikut rancangan basis data untuk aplikasi Rekomendasi Wisata Kota Malang dalam bentuk diagram.



Gambar 3.8 Rancangan Basis Data Aplikasi

Terdapat empat tabel yakni table *jobs*, *results*, *destinations*, dan *categories*. Setiap tabel memiliki peranannya masing-masing yakni.

a. Tabel *destinations*

Tabel *destinations* memuat data-data mengenai destinasi wisata Kota Malang yang terdiri dari nama, garis bujur, garis lintang, id kategori, alamat, harga, gambar 1, gambar 2, gambar 3, durasi untuk rekomendasi lama waktu yang dihabiskan di sana, status masih aktif atau tidak, status persetujuan jika ada data baru, dan hari operasional destinasi tersebut.

b. Tabel *categories*

Tabel ini berisi kategori-kategori dari destinasi wisata yang akan berfungsi untuk melakukan penyaringan berdasarkan kategori seperti kuliner, alam, dan sebagainya.

c. Tabel *jobs*

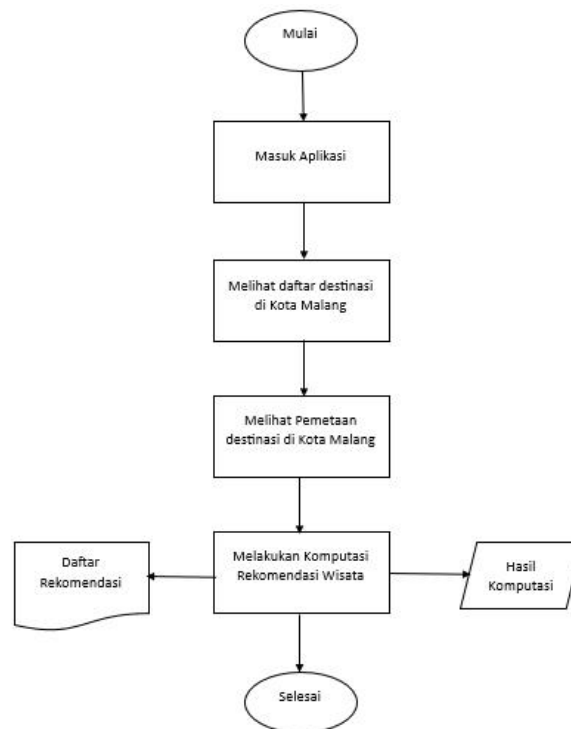
Tabel ini akan digunakan untuk mendaftarkan permintaan rekomendasi yang nantinya akan dijadikan acuan algoritma *Grey Wolf Optimizer* untuk bekerja.

d. Tabel *results*

Pemrosesan yang dilakukan menggunakan *Grey Wolf Optimizer* akan disimpan pada tabel ini dengan acuan id dari tabel *jobs*

3.4 Perancangan Aplikasi

Alur penggunaan aplikasi dirancang dengan rancangan sebagai berikut.



Gambar 3.9 Diagram Alur Penggunaan Aplikasi

Pertama-tama pengguna akan masuk ke halaman awal aplikasi, kemudian aplikasi akan mengantarkan pengguna ke daftar destinasi wisata Kota Malang yang disajikan dalam bentuk kartu-kartu. Setelahnya pengguna akan dibawa untuk melihat tampilan pemetaan interaktif mengenai destinasi-destinasi yang ada di Kota Malang. Dan pada akhirnya pengguna dapat melakukan permintaan rekomendasi untuk perjalanannya. Rekomendasi ini akan didapat dari komputasi *Grey Wolf Optimizer* yang akan menghasilkan data rekam ke database dan daftar rekomendasi yang dapat disimpan oleh pengguna dalam bentuk cetak ataupun dalam bentuk dokumen digital.

3.5 Penggunaan Algoritma *Grey Wolf Optimizer*

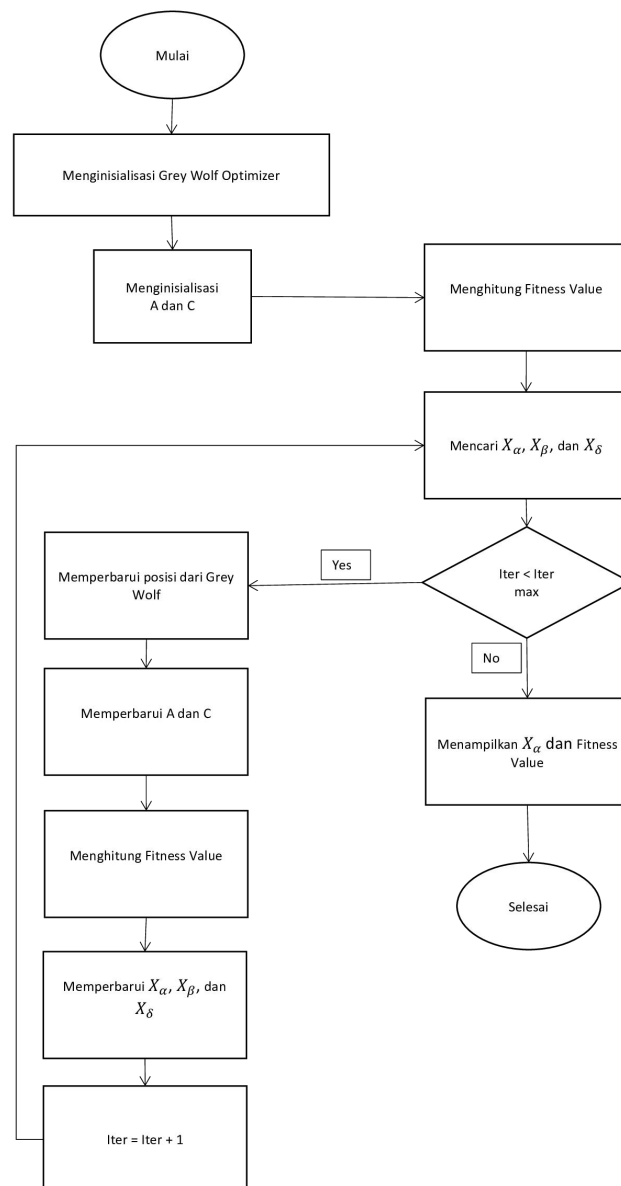
Algoritma *Grey Wolf Optimizer* akan digunakan untuk membandingkan jarak terdekat dari rute *itinerary*. Setelah mempunyai jarak terdekat untuk rute *itinerary* maka dari jarak tersebutlah yang akan diberikan kepada para wisatawan. Algoritma *Grey Wolf Optimizer* membutuhkan data semacam Berapa lama tinggal di Malang berupa Hari, Hotel atau tempat menginap di Kota Malang, data wisata atau index wisata, jarak tempat wisata, rekomendasi tempat wisata dari kementerian parekraf, population size,

dan iteration. Algoritma *Grey Wolf Optimizer* dikatakan sukses ketika menemukan jarak terdekat untuk rute *Itinerary*.

Untuk menentukan jarak terdekat nantinya Algoritma Grey Wolf akan menemukan 3 nilai atau posisi yaitu nilai untuk Alpha, nilai untuk Beta, dan nilai untuk Gamma. Setelah menemukan 3 nilai atau posisi tersebut maka akan di rata - rata dan mendapatkan hasil terdekat.

Itinerary dengan menggunakan *Grey Wolf Optimizer* ini hanya akan dilakukan pencarian *Itinerary* selama 1 hari. Kunjungan ke tempat destinasi wisata diasumsikan selama 1-2 jam, untuk memilah 1 jam atau 2 jam itu tergantung dari tempat wisata tersebut apakah dapat untuk dikunjungi selama 2 jam jika tidak maka akan berkunjung selama 1 jam di tempat wisata tersebut.

Flow Chart *Grey Wolf Optimizer*:



Gambar 3.10 Flowchart *Grey Wolf Optimizer*

Langkah-langkah Flowchart Grey Wolf

1. Menginisialisasi Grey Wolf
2. Menginisialisasi Vektor A dan Vektor C
3. Menghitung Fitness Value
4. Mencari posisi *Alpha*, *Beta*, dan *Delta*
5. Jika Iterasi kurang dari Iterasi maksimal maka memperbarui dari langkah pertama, Jika Iterasi tidak kurang dari Iterasi maksimal maka menampilkan nilai jarak terdekat

6. Menampilkan jarak terdekat dan Fitness Value

Contoh desain penggunaan Algoritma *Grey Wolf Optimizer*:

	A	B	C	D	E	F	G	H	I	J	K
1	Hotel	Wisata1	Wisata2	Wisata3	Wisata4	Wisata5	Distance(Km)			Population Size	100
2	x	1	7	4	5	14	57			Max Iteration	100
3	x	2	6	8	12	3	32	Second Best		Jumlah Wisata yang dikunjungi	1
4	x	6	12	2	7	3	37	Third Best			
5	x	78	45	12	1	45	46			Fungsi Evaluasi/Fitness = Jaraknya dekat	
6	x	5	7	12	25	20	60				
7	x	12	14	4	6	17	40				
8	x	3	2	45	20	41	31	Best			
9	x	17	20	35	78	78	63				
10	x	14	25	45	35	35	59				
11	x	13	12	78	45	5	45				
12											
13											
14	x	3	2	45	20	17	31	First Solution			
15	x	2	6	8	12	3	35	Second Solution			
16	x	10	16	38	27	30	40	Third Solution			

Gambar 3.11 Desain Penggunaan Algoritma *Grey Wolf Optimizer*

3.6 Pengujian Aplikasi

Pengujian Aplikasi dilakukan untuk menilai aplikasi sudah siap digunakan oleh user atau belum siap untuk digunakan:

1. Pengujian Website

Pengujian Website akan dilakukan oleh user dan akan mengisi form yang berisikan penilaian untuk Website. Penilaian tersebut yang akan menjadi tolak ukur aplikasi sudah siap digunakan atau belum siap.

2. Pengujian Itinerary

Pengujian *Itinerary* akan dilakukan untuk menilai *Itinerary* sudah benar dan sudah membantu. Pengujian *Itinerary* juga akan menggunakan form yang disebar untuk tolak ukur penilaian.

3. Pengujian Kecepatan Proses Itinerary

Pengujian kecepatan proses *Itinerary* dilakukan karena terdapat Populasi dan Iterasi, karena jika Populasi dan Iterasi tersebut lebih besar maka proses menemukan *Itinerary* akan semakin berat. Pengujian ini akan dilakukan dengan populasi 50, 100, 150, 200, dan 250. Sama halnya dengan populasi iterasi juga akan diuji dengan iterasi 50, 100, 150, 200, dan 250.

4. Pengujian Jarak

Pengujian jarak akan dilakukan untuk menilai apakah jarak yang dihasilkan dengan *Grey Wolf Optimizer* sudah yang paling dekat. Pengujian ini akan dibandingkan dengan jarak yang diperoleh *Google Maps*

Bab IV

Hasil dan Pembahasan

4.1 Hasil Tahap Pertama

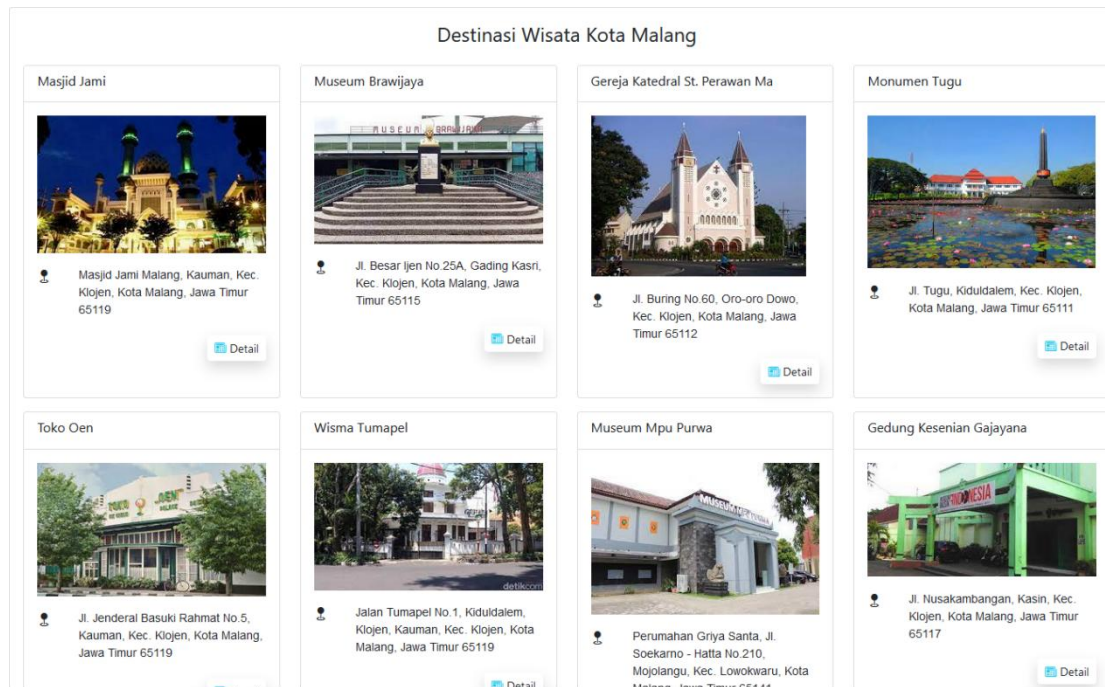
a. Perancangan Antarmuka Pengguna

Antarmuka Pengguna akan menampilkan *Home*, *Destination*, *Maps*. Data *Destination*, dan Data *Categories*. Tampilan untuk Antarmuka seperti gambar di bawah ini:



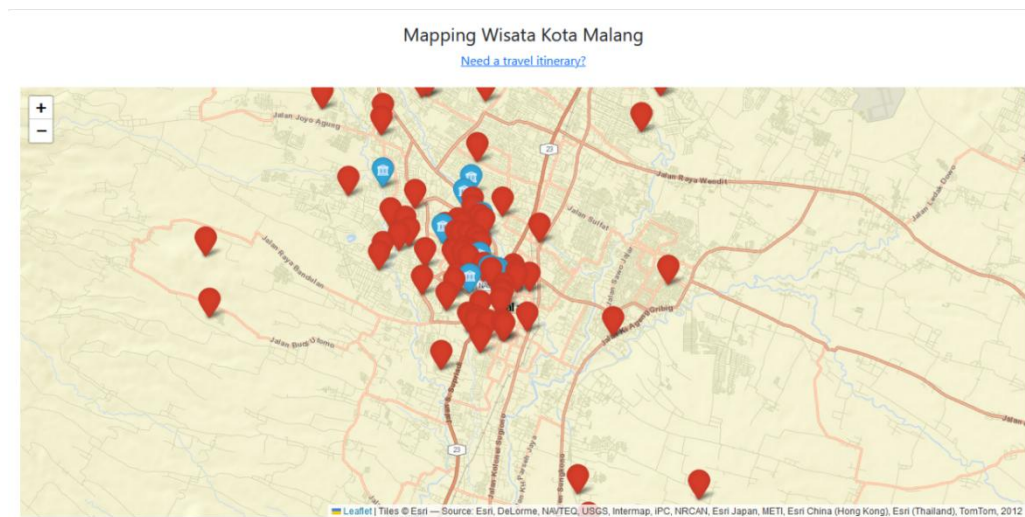
Gambar 4.1 Tampilan Home

Gambar 4.1 menampilkan Tampilan Website pada halaman *Home*, yang berisikan macam-macam wisata yang ada di Kota Malang



Gambar 4.2 Tampilan Destination

Gambar 4.2 Menampilkan halaman website Destination, yang berisikan berbagai detail Destinasi Wisata yang ada di Kota Malang. Dimulai dari Destinasi Kampung Wisata, Wisata Religi, Wisata Taman, Wisata Kuliner, Wisata Heritage, dan Wisata Kuliner.



Gambar 4.3 Tampilan Maps

Gambar 4.3 Menampilkan halaman website Maps, yang berisikan Maps Kota Malang dan berbagai letak Posisi Destinasi Wisata yang ada di Kota Malang.

Itinerary

Where are you staying?

Whiz Prime

Categories
☐ WisataKampung
☒ WisataHeritage
☐ WisataAir
☒ WisataKuliner
☐ WisataTaman
☐ WisataReligi

Show Me Around

Gambar 4.4 Tampilan mencari *Itinerary*

Gambar 4.4 Menampilkan halaman website untuk mencari *Itinerary*, pada halaman website ini akan terdapat pilihan untuk pengguna memilih wisata mana saja yang ingin di kunjungi. Wisata yang ada di categories adalah WisataKampung, Wisata Heritage, WisataAir, WisataKuliner, WisataTaman, dan WisataReligi. Setelah pengguna memilih *Itinerary* yang keluar akan sesuai dengan pilihan pengguna.

Itinerary




Where are you staying?

Whiz Prime

Categories
☐ WisataKampung
☒ WisataHeritage
☐ WisataAir
☒ WisataKuliner
☐ WisataTaman
☐ WisataReligi

Show Me Around

Total Jarak : 21.19 Km

	<u>Whiz Prime</u> ⌚ 8:00 WIB Rp. 358.000 - 463.000 📍 Jl. Jenderal Basuki Rahmat No.85-87, Klojen, Kec. Klojen, Kota Malang, Jawa Timur 65119	1
	<u>Java Dancer Coffee</u> ⌚ 9:00 WIB Rp. 20.000 - 190.000 📍 Java Dancer Coffee - Majapahit, Jl. Majapahit No.2, Kauman, Kec. Klojen, Kota Malang, Jawa Timur 65111	2
	<u>KON.CO Coffe dan Garden</u> ⌚ 11:00 WIB Rp. 17.000 - 24.000 📍 KON.CO COFFEE AND GARDEN, Mojolangu, Kec. Lowokwaru, Kota Malang, Jawa Timur 65142	3

Gambar 4.5 Tampilan mendapatkan *Itinerary*

Gambar 4.5 Merupakan tampilan website *Itinerary* yang sudah didapatkan sesuai dengan keinginan pengguna. Pada contoh diatas pengguna memilih Wisata Kampung dan Wisata Heritage, maka *Itinerary* yang keluar hanya wisata Wisata Kampung dan Wisata Heritage. Itinerary yang muncul terdapat 8 Destinasi Wisata kecuali titik awal yaitu hotel.

b. Hasil Perancangan Database

Dari Perancangan yang di rancang pada bagian BAB 3 hasilnya keluar seperti gambar dibawah ini:



Gambar 4.6 Database

Gambar 4.6 Merupakan Database yang terdiri dari 2 Table, yaitu Table categories dan Table destination. Table categories untuk membedakan jenis-jenis destination wisata yang ada di kota malang.

<div><div><div></div><div></div><div></div></div></div>				id	categoryname	color	icon
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	6	Hotel	blue	bi bi-bank
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	7	WisataKampung	NULL	NULL
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	8	WisataHeritage	NULL	NULL
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	9	WisataAir	NULL	NULL
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	10	WisataKuliner	NULL	NULL
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	11	WisataTaman	NULL	NULL
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	12	WisataReligi	NULL	NULL

Gambar 4.7 Tabel categories

Gambar 4.7 merupakan tabel categories, ada 7 categories Hotel, WisataKampung, WisataHeritage, WisataAir, WisataKuliner, WisataTaman, dan WisataReligi. Categories ini untuk membedakan jenis-jenis destinasi wisata yang ada di Kota Malang.

		id	name	longitude	latitude	categoryid
<input type="checkbox"/>	Edit Copy Delete	1	Masjid Jami	-7.982228046815176	112.6297423413953	12
<input type="checkbox"/>	Edit Copy Delete	4	Museum Brawijaya	-7.971919991225763	112.62123045489021	8
<input type="checkbox"/>	Edit Copy Delete	5	Gereja Katedral St. Perawan Maria Gunung Karmel (...	-7.968481687548863	112.62444630459986	12
<input type="checkbox"/>	Edit Copy Delete	11	Monumen Tugu	-7.9768683	112.6335173	8
<input type="checkbox"/>	Edit Copy Delete	12	Toko Oen	-7.9807376	112.6302707	10
<input type="checkbox"/>	Edit Copy Delete	13	Wisma Tumapel	-7.9857388	112.6228554	8

Gambar 4.8 Table destination

Gambar 4.8 merupakan table untuk menampung destinasi wisata yang ada di Kota Malang. Terdapat 89 destinasi wisata yang sudah di masukan ke dalam database ini. Table destinasi terdapat juga isi detail dari masing-masing destinasi wisata, yaitu longitude, latitude, categoryid, address, price, pict1, pict2, pict3, duration, dan jadwal buka harinya.

4.2 Hasil Tahap Kedua

Hasil Tahap kedua berisikan perancangan dan pembuatan aplikasi *Itinerary* menggunakan *Grey Wolf Optimizer*. Berikut adalah penggalan code pembuatan *Itinerary* menggunakan *Grey Wolf Optimizer*.

```
function evaluateFitness($solution) {
    $totalDistance = 0;
    $numCities = count($solution);

    for ($i = 0; $i < $numCities - 1; $i++) {
        $city1 = $solution[$i];
```

Gambar 4.9. Evaluation Fitness

Code diatas merupakan penggalan code untuk melakukan evaluasi pada nilai fitness

Berikut adalah penjelasan untuk setiap bagian dari kode tersebut:

1. Deklarasi variabel \$totalDistance dengan nilai awal 0. Variabel ini akan digunakan untuk menghitung total jarak antara kota-kota dalam solusi.
2. Menghitung jumlah kota dalam solusi dengan menggunakan fungsi count() pada variabel \$solution dan menyimpan hasilnya dalam variabel \$numCities.
3. Memulai iterasi dari 0 hingga \$numCities - 1 menggunakan perulangan for. Perulangan ini digunakan untuk mengiterasi melalui kota-kota dalam solusi.
4. Di dalam perulangan, mengambil kota saat ini (\$city1) dan kota berikutnya (\$city2) dari \$solution menggunakan indeks \$i dan \$i + 1.
5. Memanggil fungsi calculateDistance() dengan parameter koordinat lintang dan bujur kota (\$city1['lat'], \$city1['lon'], \$city2['lat'], \$city2['lon']). Fungsi ini digunakan untuk menghitung jarak antara dua kota menggunakan koordinat lintang dan bujur mereka.
6. Menambahkan jarak yang dihitung ke \$totalDistance.
7. Setelah perulangan selesai, mengembalikan nilai \$totalDistance sebagai hasil dari fungsi evaluateFitness().

Jadi, fungsi ini mengiterasi melalui kota-kota dalam solusi dan menghitung total jarak antara kota-kota tersebut menggunakan fungsi `calculateDistance()`. Nilai total jarak ini kemudian dikembalikan sebagai hasil dari fungsi.

```
$population = [];  
  
    for ($i = 0; $i < $populationSize; $i++) {  
  
        $population[$i] = $this->  
generateRandomSolution($cities);
```

Gambar 4.10 Initialize Population

Code diatas merupakan penggalan code untuk menginisialisasi populasi yang digunakan untuk algoritma *Grey Wolf Optimizer*

Berikut adalah penjelasan untuk setiap bagian dari kode tersebut:

1. Mendeklarasikan variabel `$population` sebagai array kosong. Variabel ini akan digunakan untuk menyimpan populasi awal yang akan dibentuk.
2. Memulai perulangan `for` dengan variabel iterasi `$i` dimulai dari 0 hingga `$populationSize - 1`. Tujuan dari perulangan ini adalah untuk membuat individu-individu dalam populasi awal.
3. Di dalam perulangan, setiap iterasi, memanggil fungsi `generateRandomSolution()` dengan parameter `$cities`. Fungsi ini digunakan untuk menghasilkan satu solusi acak (individu) yang terdiri dari kota-kota dalam `$cities`.
4. Menyimpan solusi yang dihasilkan ke dalam elemen array `$population` dengan indeks `$i`. Dengan demikian, setiap elemen array `$population` akan berisi satu individu dalam populasi.
5. Setelah perulangan selesai, populasi awal yang terdiri dari `$populationSize` individu telah terbentuk dan disimpan dalam array `$population`.

Dengan demikian, kode ini bertujuan untuk membentuk populasi awal dengan jumlah individu sebanyak `$populationSize`. Setiap individu dibentuk dengan menggunakan fungsi `generateRandomSolution()` yang menghasilkan solusi acak berdasarkan kota-kota yang tersedia dalam array `$cities`.

```
$alphaIndex = array_search(min($fitness), $fitness);  
  
$alpha = $population[$alphaIndex];  
  
$fitness[$alphaIndex] = PHP_INT_MAX;  
  
$betaIndex = array_search(min($fitness), $fitness);  
$beta = $population[$betaIndex];  
  
$fitness[$betaIndex] = PHP_INT_MAX;  
  
$deltaIndex = array_search(min($fitness), $fitness);  
$delta = $population[$deltaIndex];
```

Gambar 4.11 Posisi Alpha, Beta, dan Delta

Code diatas merupakan penggalan code untuk mencari posisi Alpha, Beta, dan Delta

Berikut adalah penjelasan untuk setiap bagian dari kode tersebut:

1. Menggunakan fungsi `array_search()` untuk mencari indeks elemen dengan nilai terkecil dalam array `$fitness` (yang berisi nilai kecocokan dari solusi-solusi dalam populasi).
2. Mengambil solusi dengan indeks tersebut dari array `$population` dan menyimpannya dalam variabel `$alpha`.
3. Mengatur nilai `$fitness` pada indeks `$alphaIndex` menjadi `PHP_INT_MAX` (nilai maksimum yang dapat ditampung oleh tipe data integer di PHP) untuk menandai bahwa solusi `$alpha` tidak akan dipertimbangkan lagi dalam pencarian solusi berikutnya.
4. Menggunakan kembali fungsi `array_search()` untuk mencari indeks elemen dengan nilai terkecil yang belum diubah sebelumnya dalam array `$fitness`.
5. Mengambil solusi dengan indeks tersebut dari array `$population` dan menyimpannya dalam variabel `$beta`.

6. Mengatur nilai \$fitness pada indeks \$betaIndex menjadi PHP_INT_MAX untuk menandai bahwa solusi \$beta tidak akan dipertimbangkan lagi dalam pencarian solusi berikutnya.

7. Sekali lagi, menggunakan fungsi array_search() untuk mencari indeks elemen dengan nilai terkecil yang belum diubah sebelumnya dalam array \$fitness.

8. Mengambil solusi dengan indeks tersebut dari array \$population dan menyimpannya dalam variabel \$delta.

9. Mengatur nilai \$fitness pada indeks \$deltaIndex menjadi PHP_INT_MAX untuk menandai bahwa solusi \$delta tidak akan dipertimbangkan lagi dalam pencarian solusi berikutnya.

Dengan demikian, kode tersebut bertujuan untuk menemukan tiga solusi terbaik (alpha, beta, dan delta) dalam populasi berdasarkan nilai kecocokan solusi-solusi

```
for ($i = 0; $i < $populationSize; $i++) {  
    for ($j = 0; $j < $numCities; $j++) {  
        $random1 = mt_rand() / mt_getrandmax();  
        $random2 = mt_rand() / mt_getrandmax();  
        $A1 = 2 * $alpha[$j]['lat'] * $random1 - $alpha[$j]['lat'];  
        $C1 = 2 * $random2;  
        $D_alpha = abs($C1 * $alpha[$j]['lat'] - $population[$i][$j]['lat']);  
        $random1 = mt_rand() / mt_getrandmax();  
        $random2 = mt_rand() / mt_getrandmax();  
        $A2 = 2 * $beta[$j]['lat'] * $random1 - $beta[$j]['lat'];  
        $C2 = 2 * $random2;  
        $D_beta = abs($C2 * $beta[$j]['lat'] - $population[$i][$j]['lat']);  
        $random1 = mt_rand() / mt_getrandmax();  
        $random2 = mt_rand() / mt_getrandmax();  
        $A3 = 2 * $delta[$j]['lat'] * $random1 - $delta[$j]['lat'];  
        $C3 = 2 * $random2;  
        $D_delta = abs($C3 * $delta[$j]['lat'] - $population[$i][$j]['lat']);  
        $newLat = ($alpha[$j]['lat'] + $beta[$j]['lat'] + $delta[$j]['lat']) / 3 +  
$A1 + $A2 + $A3;  
        $newLat -= min($newLat, $cities[$j]['lat']);  
        $newLat = min($newLat, $cities[$j]['lat']);  
    }  
}
```

Gambar 4.12 Update Posisi Alpha, Beta, dan Delta Latitude

Code diatas merupakan code untuk update posisi Alpha, Beta, dan Delta jika Iterasi kurang dari Iterasi Max.

Berikut adalah penjelasan untuk setiap bagian dari kode tersebut:

1. Perulangan pertama menggunakan variabel \$i untuk mengiterasi melalui setiap solusi dalam populasi dengan indeks \$i.
2. Di dalam perulangan pertama, perulangan kedua menggunakan variabel \$j untuk mengiterasi melalui setiap kota dengan indeks \$j dalam solusi.
3. Di setiap iterasi perulangan kedua, kode ini menghasilkan dua angka acak (\$random1 dan \$random2) dengan menggunakan fungsi mt_rand() untuk digunakan dalam perhitungan selanjutnya.
4. Menghitung nilai \$A1 dan \$C1 berdasarkan formula yang diberikan dalam kode, yang melibatkan komponen dari solusi alpha dan nilai acak yang dihasilkan sebelumnya.
5. Menghitung \$D_alpha (jarak absolut antara $C1 * \alpha[j][lat]$ dan $population[i][j][lat]$).
6. Mengulangi langkah-langkah 4-5 untuk solusi beta dan solusi delta, dan menghitung masing-masing \$D_beta dan \$D_delta.
7. Menghitung nilai baru untuk \$newLat berdasarkan formula yang diberikan dalam kode, yang melibatkan komponen dari solusi alpha, beta, dan delta, serta variabel \$A1, \$A2, dan \$A3 yang dihasilkan sebelumnya.
8. Mengurangi \$newLat dengan nilai terkecil antara \$newLat dan \$cities[\$j][lat].
9. Memastikan bahwa \$newLat tidak melebihi nilai \$cities[\$j][lat] dengan membatasinya menjadi nilai terkecil antara \$newLat dan \$cities[\$j][lat].

Dengan demikian, kode ini bertujuan untuk menghasilkan koordinat lintang baru (\$newLat) untuk setiap kota dalam setiap solusi dalam populasi, berdasarkan komponen dari solusi alpha, beta, dan delta, serta perhitungan yang melibatkan angka acak.

```
$random1 = mt_rand() / mt_getrandmax();

    $random2 = mt_rand() / mt_getrandmax();

    $A1 = 2 * $alpha[$j]['lon'] * $random1 - $alpha[$j]['lon'];

    $C1 = 2 * $random2;

    $D_alpha = abs($C1 * $alpha[$j]['lon'] - $population[$i][$j]['lon']);

    $random1 = mt_rand() / mt_getrandmax();

    $random2 = mt_rand() / mt_getrandmax();

    $A2 = 2 * $beta[$j]['lon'] * $random1 - $beta[$j]['lon'];

    $C2 = 2 * $random2;

    $D_beta = abs($C2 * $beta[$j]['lon'] - $population[$i][$j]['lon']);

    $random1 = mt_rand() / mt_getrandmax();

    $random2 = mt_rand() / mt_getrandmax();

    $A3 = 2 * $delta[$j]['lon'] * $random1 - $delta[$j]['lon'];

    $C3 = 2 * $random2;

    $D_delta = abs($C3 * $delta[$j]['lon'] - $population[$i][$j]['lon']);

    $newLon = ($alpha[$j]['lon'] + $beta[$j]['lon'] + $delta[$j]['lon']) / 3 +

$A1 + $A2 + $A3;

    $newLon -= min($newLon, $cities[$j]['lon']);

    $newLon = min($newLon, $cities[$j]['lon']);

    $population[$i][$j]['lat'] = $newLat;

    $population[$i][$j]['lon'] = $newLon;
```

Gambar 4.13 Update Posisi Alpha, Beta, dan Delta Longitude

Code diatas merupakan terusan code untuk update posisi Alpha, Beta, dan Delta jika Iterasi kurang dari Iterasi Max.

Berikut adalah penjelasan untuk setiap bagian dari kode tersebut:

1. Menghitung \$newLat dengan menjumlahkan nilai koordinat lintang dari solusi alpha, beta, dan delta, kemudian dibagi dengan 3. Hasilnya ditambahkan dengan variabel \$A1, \$A2, dan \$A3 yang dihasilkan sebelumnya. Ini menghasilkan nilai sementara untuk \$newLat.
2. Mengurangi nilai \$newLat dengan nilai terkecil antara \$newLat dan \$cities[\$j]['lat'].
3. Memastikan bahwa \$newLat tidak melebihi nilai \$cities[\$j]['lat'] dengan membatasinya menjadi nilai terkecil antara \$newLat dan \$cities[\$j]['lat'].
4. Menghasilkan dua angka acak (\$random1 dan \$random2) dengan menggunakan fungsi mt_rand() untuk digunakan dalam perhitungan selanjutnya.
5. Menghitung nilai \$A1 dan \$C1 berdasarkan formula yang diberikan dalam kode, yang melibatkan komponen dari solusi alpha dan nilai acak yang dihasilkan sebelumnya.
6. Menghitung \$D_alpha (jarak absolut antara \$C1 * \$alpha[\$j]['lon'] dan \$population[\$i][\$j]['lon']).
7. Mengulangi langkah-langkah 5-6 untuk solusi beta dan solusi delta, dan menghitung masing-masing \$D_beta dan \$D_delta.
8. Menghitung nilai baru untuk \$newLon berdasarkan formula yang diberikan dalam kode, yang melibatkan komponen dari solusi alpha, beta, dan delta, serta variabel \$A1, \$A2, dan \$A3 yang dihasilkan sebelumnya.
9. Mengurangi \$newLon dengan nilai terkecil antara \$newLon dan \$cities[\$j]['lon'].
10. Memastikan bahwa \$newLon tidak melebihi nilai \$cities[\$j]['lon'] dengan membatasinya menjadi nilai terkecil antara \$newLon dan \$cities[\$j]['lon'].

Dengan demikian, kode ini bertujuan untuk menghasilkan koordinat bujur baru (\$newLon) untuk setiap kota dalam setiap solusi dalam populasi, berdasarkan

```

k $fitness = [];
a
    for ($i = 0; $i < $populationSize; $i++) {
        $fitness[$i] = $this->evaluateFitness($population[$i]);
    }

    $bestIndex = array_search(min($fitness), $fitness);
    $bestSolution = $population[$bestIndex];

    return $bestSolution;

```

Gambar 4.14 Menemukan Best Solution

Code diatas merupakan code untuk menemukan best solution dari perhitungan *Grey Wolf Optimizer*, dan hasil yang di keluarkan menjadi *Itinerary* untuk pengguna.

Berikut adalah penjelasan untuk setiap bagian dari kode tersebut:

1. Mendeklarasikan array \$fitness sebagai array kosong yang akan digunakan untuk menyimpan nilai kecocokan dari setiap solusi dalam populasi.
2. Memulai perulangan for dengan inisialisasi variabel \$i dengan nilai awal 0. Perulangan ini akan berjalan selama nilai \$i kurang dari \$populationSize.
3. Di setiap iterasi, menghitung nilai kecocokan solusi pada \$population[\$i] menggunakan fungsi evaluateFitness() dan menyimpannya dalam array \$fitness pada indeks yang sesuai (\$i).
4. Setelah perulangan selesai, mencari indeks elemen dengan nilai terkecil dalam array \$fitness menggunakan fungsi array_search() dan menyimpannya dalam variabel \$bestIndex.
5. Mengambil solusi terbaik dari populasi dengan menggunakan indeks \$bestIndex dan menyimpannya dalam variabel \$bestSolution.
6. Mengembalikan solusi terbaik (\$bestSolution) sebagai hasil dari fungsi.

Dengan demikian, kode ini bertujuan untuk menghitung nilai kecocokan dari setiap solusi dalam populasi, menemukan solusi terbaik berdasarkan nilai kecocokan tersebut, dan mengembalikan solusi terbaik tersebut.

4.3 Pengujian GWO

Pengujian algoritma GWO dilakukan dengan dua metode yakni pengujian tuning parameter dan pengujian jarak antar GWO. Pengujian tuning parameter

dilakukan guna mendapatkan tuning parameter dengan waktu pemrosesan yang tidak terlalu lama. Pengujian jarak antar GWO akan dilakukan perulangan sebanyak 5 kali dan akan dilihat dari 5 kali GWO apakah hasilnya semakin bagus atau semakin tidak optimal.

4.3.1 Pengujian Tuning Parameter

Parameter yang diuji ialah jumlah iterasi dan jumlah populasi, sejatinya semakin banyak jumlah populasi dan jumlah iterasi akan semakin baik, namun semakin banyak jumlah populasi dan iterasi akan mempengaruhi di waktu dan sumber daya computer dalam pemrosesannya.

Tabel 4.1. Pengujian Tuning Parameter

No	Jumlah Iterasi	Jumlah Populasi	Jumlah Waktu
1	250	250	17.89s
2	200	200	11.83s
3	150	150	9.15s
4	100	100	2.88s
5	50	50	2.38s

Kesimpulan Pengujian:

Dari pengujian Tuning Parameter bisa di ambil kesimpulan jika jumlah populasi dan jumlah iterasi ditambah maka akan semakin lama menemukan rute tercepat dari destinasi yang ada di kota malang.

4.3.2 Pengujian Jarak antar GWO

Pengujian jarak dilakukan dengan melakukan serangkaian destinasi yang berisikan dari ke 6 *Categories* dengan Hotel yang sama, total jarak yang ditempuh oleh GWO dan dibandingkan dengan jarak GWO lainnya dari 5 rangkaian tuning parameter dari sub bab 4.3.1.

Tabel 4.2. Pengujian Jarak antar GWO dengan Tuning Parameter 50 iterasi dan 50 populasi

No	Rangkaian Destinasi	Total Jarak GWO
1	Whiz Prime → Monumen Adipura → Patung Singo Edan Arema → Wisata Aeng Hamid Rusdi → Kampung Wisata Payung → Kampung Heritage Kajoetangan → Patung Pahlawan Trip → Kampung Warna Warni Jodipan → Taman Indie River View Resto	23.05 KM
2	Whiz Prime → Seoulscent → Taman Kunang-Kunang → Patung Pahlawan Trip → Kampung Wisata Keramik Dinoyo → Angsle & Ronde Titoni → Mie T-Rex → Monumen TGP → Patung Hamid Rusdi	22.74 KM
3	Whiz Prime → Habbit Eatery → Kampung Wisata Payung → Monumen Tugu → Vihara Vajra Eart Kertanegara → Patung Arema Singo Edan → Ijen Boulevard → Taman Singha Merjosari → Monumen Adipura	22.56 KM
4	Whiz Prime → Taman Slamet → Hutan Kota Tunggul Wulung → Hawaii Waterpark → Kajoetangan Street → Gereja Ijen Katedral → Monumen Pesawat MIG 12 Fighter → Museum Pendidikan Indonesia → Taman Cerme	20.04 KM
5	Whiz Prime → Alun - Alun Merdeka → Mie T-Rex → Es Talun → Museum Mpu Purwa → Kampung Putih → Kampung Budaya Polowijen → Gereja Katedral Ijen → Kampung Sejarah	17.61 KM

Hasil yang didapatkan pada Tabel 4.2 adalah dengan dilakukannya pada iterasi 50 dan populasi 50 dan dilakukan percobaan berkali-kali maka akan menghasilkan Hasil yang random atau acak untuk hasil jarak terdekatnya, karena untuk tempat wisata yang keluar juga muncul secara acak.

Tabel 4.3 Pengujian Jarak antar GWO dengan Tuning Parameter 100 iterasi dan 100 populasi

No	Rangkaian Destinasi	Total Jarak GWO
1	Grand Mercure Malang Mirama → Bataputi Coffeehouse → Patung	37.36 KM

	Ken Dedes → Monumen Melati → Taman Edukasi Rolak → Es Talun → Vihara Vajra Earth Kertanegara → Masjid Ki Ageng Gribig → Taman Kunang-Kunang	
2	Grand Mercure Malang Mirama → Patung Hamid Rusdi → Vihara Vajra Earth Kertanegara → Kampung Warna Warni Jodipan → Es Talun → Gereja Hati Kudus Yesus Kayutangan → Patung Panglima Sudirman → Kampung Wisata Keramik Dinoyo → MIE T-Rex	33.59 KM
3	Grand Mercure Malang Mirama → Monumen Melati → Alun-Alun Merdeka → Seoulscent → Monumen Tugu → Kajoetangan Street → Museum Pendidikan Indonesia → Kampung Terapi Hijau → Vihara Vajra Earth Kertanegara	24.11 KM
4	Grand Mercure Malang Mirama → Hutan Kota Malabar → Taman Kota Terusan Diengt → Museum Brawijaya → Taman Dempo → Klenteng Eng An Kiong → Kampung Tridi → Bakso Bakar Pak Man → Museum Mpu Purwa	23.67 KM
5	Grand Mercure Malang Mirama → Kampung Tridi → Loteng Gereja GPIB Immanuel → Museum Brawijaya → Hutan Kota Malabar → Gereja Katedral Ijen → Masjid Jami → Taman Edukasi Rolak	14.21 KM

Hasil yang didapatkan pada Tabel 4.3 adalah dengan dilakukannya pada iterasi 50 dan populasi 50 dan dilakukan percobaan berkali-kali maka akan menghasilkan Hasil yang random atau acak untuk hasil jarak terdekatnya, karena untuk tempat wisata yang keluar juga muncul secara acak.

Tabel 4.4 Pengujian Jarak antar GWO dengan Tuning Parameter 150 iterasi dan 150 populasi

No	Rangkaian Destinasi	Total Jarak GWO
1	Swiss-Belinn → Kampung Wisata Topeng Desaku Menanti → Monumen Tugu → Patung Pahlawan Trip → Museum Motor Klasik → Taman Kunang-Kunang → Bakso Bakar Pak Man Kampung Tridi	37.11 KM

	→ Toko Oen	
2	Swiss-Belinn → Golden Heritage Koffie → Java Dancer Coffee → Madam Wang Secret Garden → Kampung Terapi Hijau → Monumen TGP → Taman Dempo → Monumen Pesawat MIG 17 Fighter → Masjid Ki Ageng Gribig	33.09 KM
3	Swiss-Belinn → Masjid Ki Ageng Gribig → Cwi Mie Malang Mie Amor → Taman Singhasari Merjosari → Monumen Melati → Javanine → Patung Chairil Anwar → Melati Restaurant → Loteng	23.11 KM
4	Swiss-Belinn → Taman Mojolangu → Taman Edukasi Rolak → Kampung Wisata Topeng Desaku Menanti → Taman Cerme → Kampung Tridi → Monumen TGP → Es Talun → Pawvilion Dog Cafe	23.6 KM
5	Swiss-Belinn → Gereja GPIB Immanuel → Loteng → Taman Cerme → Wisma Tumapel → Mie T-Rex → Hawaii Waterpark → Kampung Warna Warni Jodipan → Patung Hamid Rusdi	16.24 KM

Hasil yang didapatkan pada Tabel 4.4 adalah dengan dilakukannya pada iterasi 50 dan populasi 50 dan dilakukan percobaan berkali-kali maka akan menghasilkan Hasil yang random atau acak untuk hasil jarak terdekatnya, karena untuk tempat wisata yang keluar juga muncul secara acak.

Tabel 4.5 Pengujian Jarak antar GWO dengan Tuning Parameter 200 iterasi dan 200 populasi

No	Rangkaian Destinasi	Total Jarak GWO
1	Shalimar Hotel → Museum Motor Klasik → Kampung Putih → Monumen Tugu → Kampung Biru Arema → Taman Dempo → Ijen Boulevard → Kampung Heritage Kajoetangan → Monumen Adipura	14.01 KM
2	Shalimar Hotel → Patung Pahlawan Trip → Habbit Eatery → Volks Waffle Stretery → Monumen Tugu → Museum Motor Klasik → Kampung Wisata Keramik Dinoyo → Seoulscent → Alun-Alun	17.77 KM

	Merdeka Malang	
3	Shalimar Hotel → Mie T-Rex → Volks Waffle Stretery → Patung Arema Singo Edan → Kampung Tridi → Monumen TGP → Kampung Glintung Go Green → Kelenteng Eng AN Kiong → Kajoetangan Street	18.05 KM
4	Shalimar Hotel → Bakso Bakar Pak Man → Monumen TGP → Museum Pendidikan Indonesia → Kampung Budaya Polowijen → Kampung Wisata Keramik Dinoyo → Alun-Alun Merdeka Malang → Monumen Pesawat MIG 17 Fighter → Masjid Jami	29.62 KM
5	Shalimar Hotel → Masjid Jami → KON.CO Coffee dan Garden → Kampung Wisata Topeng Desaku Menanti → Zoologi Frater Vianney → Taman Indie River View Resto → Masjid Ki Ageng Gribig → Taman Kota Terusan Dieng → Kajoetangan Street	35.13 KM

Hasil yang didapatkan pada Tabel 4.5 adalah dengan dilakukannya pada iterasi 50 dan populasi 50 dan dilakukan percobaan berkali-kali maka akan menghasilkan Hasil yang random atau acak untuk hasil jarak terdekatnya, karena untuk tempat wisata yang keluar juga muncul secara acak.

Tabel 4.6 Pengujian Jarak antar GWO dengan Tuning Parameter 250 iterasi dan 250 populasi

No	Rangkaian Destinasi	Total Jarak GWO
1	Hotel Aria Gajayana → Taman Cerme → Kampung Biru Arema → Java Dancer Coffee → Kampung Heritage Kajoetangan → Kajoetangan Street → Kampung Putih → Gereja Katedral Ijen → Alun-Alun Merdeka Malang	7.63 KM
2	Hotel Aria Gajayana → Monumen Melati → The Lost Temple Dinner Saigon San → Toko Oen → Es Talun → Hutan Kota Tunggul Wulung → Pawvilion Dog Cafe → Kampung Terapi Hijau → 15.18 KMKajoetangan Street	15.18 KM
3	Hotel Aria Gajayana → Kampung Glintung Go Green → Taman	17.45 KM

	Edukasi Rolak → Monumen TGP → Kampung Sejarah → KON.CO Coffee dan Garden → Seoulscent → Kampung Putih	
4	Hotel Aria Gajayana → Kampung Budaya Polowijen → Cafe Bukit Delight → Monumen Pesawat MIG 17 Fighter → The Lost Temple Dinner Saigon San → Melati Restaurant → Hutan Tugu Kota Malang → Monumen Adipura → Vihara Vajra Earth Kertanegara	22.9 KM
5	Hotel Aria Gajayana → Masjid Ki Ageng Gribig → Cwi Mie Malang Mi Amor → Masjid Jami → Angsle & Ronde Titoni → Java Dancer Coffee → Wisata Aeng Hamid Rusdi Wonokoyo → Kampung Terapi Hijau → Kajoetangan Street	42.14 KM

Hasil yang didapatkan pada Tabel 4.6 adalah dengan dilakukannya pada iterasi 50 dan populasi 50 dan dilakukan percobaan berkali-kali maka akan menghasilkan Hasil yang random atau acak untuk hasil jarak terdekatnya, karena untuk tempat wisata yang keluar juga muncul secara acak.

Dari pengujian *Grey Wolf Optimizer* bisa di ambil kesimpulan jika jumlah populasi dan jumlah iterasi ditambah maka akan semakin lama menemukan rute tercepat dari destinasi yang ada di kota malang, karena semakin banyak iterasi dan populasi maka makin banyak data yang di olah. Untuk kesimpulan jarak, tidak terlalu berpengaruh karena untuk pengukuran jarak tempat wisata yang keluar secara random atau acak jadi jika berkali-kali melakukan Itinerary hasil jarak yang keluar akan selalu acak.

Bab 5

Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan Penelitian tugas akhir yang telah dilakukan dapat ditarik kesimpulan sebagai berikut.

1. *Grey Wolf Optimizer* merupakan algoritma metaheuristik yang mengimplementasikan perilaku serigala di alam. Serigala abu-abu dianggap sebagai predator puncak, yang berarti bahwa serigala abu-abu berada di puncak rantai makanan. Serigala abu-abu juga memiliki hirarki dominan sosial yang tinggi.
2. *Grey Wolf Optimizer* melakukan perhitungan jarak berdasarkan koordinat atau longitude dan latitude, pada dasarnya perhitungan ini masih kurang efisien karena jika dibandingkan dengan Google Maps karena Google Maps mampu menentukan jarak sesuai kondisi jalan raya yang berlaku pada dunia nyata.
3. Jika populasi dan iterasi ditambah maka untuk pencarian rute tercepat akan semakin lama didapat, sehingga membuat kurang efisien ketika mencari *itinerary* untuk para wisatawan

5.2 Saran

Penelitian tidak selalu membuahkan hasil yang diinginkan, penelitian dilakukan untuk menguji hipotesis, demikian juga penelitian ini yang hasilnya dapat dilihat pada Bab 4 dan Bab 5 pada bagian kesimpulan, sehingga kedepannya perlu ada pertimbangan dan peningkatan sebagai berikut:

1. Menggunakan Google Maps atau fitur serupa untuk menghitung jarak sehingga bisa didapatkan jarak yang lebih baik dan lebih akurat.
2. Menambahkan checkpoints untuk penginapan jika menginap lebih dari satu hari.
3. Menambahkan sistem rating destinasi agar rekomendasi destinasi menjadi lebih baik.
4. Menambahkan kategori apa saja yang ingin dikunjungi.

DAFTAR PUSTAKA

- Lestari, V., Kamsyakawuni, A., Santoso, K,A. 2019, Implementasi Algoritma Greywolf Optimizer (GWO) di Toko Citra Tani Jember, *Majalah Ilmiah Matematika dan Statistika*, **19**, 65-74
- Oktaviani, M,A., Wibowo, R,S., Aryani, N,K. 2018, Aliran Daya dengan Efek Katup Menggunakan Grey Wolf Optimization, *Jurnal Teknik ITS*, **7**, 2337-3539
- Almufti, S,M., Ahmad, H,B., Marqas, R,B., Asaad, R,R. 2021, Grey Wolf Optimizer: Overview, Modifications, and Applications, *International Research Journal of Science, Technology, Education, and Management*, **1**, 44-56
- Ahmadi, R., Ekbatanifard, G., Bayat, P. 2021, a Modified Grey Wolf Optimizer Based Data Clustering Algorithm, *Applied Artificial Intelligence*, **35**, 63-79
- Khandelwal, A., Bhargava, A., Sharma, A., Sharma, H. 2018, Modified Grey Wolf Optimization Algorithm for Transmission Network Expansion Planning Problem, *Arabian Journal for Science and Engineering*, **43**, 2899-2908
- Natesan, G., Chokkalingam, A. 2019, Optimal Task Scheduling in the Cloud Environment Using aMean Grey Wolf Optimization Algorithm, *International Journal of Technology*, **10**, 126-136
- Luo, S., Zhang, L., Fan, Y. 2019, Energy-Efficient Scheduling for Multi-Objective Flexible Speeds by Grey Wolf Optimization, *Journal of Cleaner Production*, **234**, 1365-1384
- Hosseinzadeh, A,Z., Amiri, G,G., Abyaneh, M,J., Razzaghi, S,A,S., Hamzehkolaei, A,G. 2020, Baseline Updating Method for Structural Damage Identification Using

Modal Residual Force and Grey Wolf Optimization, *Engineering Optimization*, **52**, 549-566

Sreenu, K., Malempati, S, 2019, MFGMTS: Epsilon Constraint-Based Modified Fractional Grey Wolf Optimizer for Multi-Objective Task Scheduling in Cloud Computing, *IETE Journal of Research*, **65**, 201-215

Tripathi, A,K., Sharma, K., Bala, M. 2018, a Novel Clustering Method Using Enhanced Grey Wolf Optimizer and MapReduce, *Big Data Research*, **14**, 93-100

Abed-alguni, B,H., Alawad, N,A. 2021, Distirbuted Grey Wolf Optimizer for Scheduling of Workflow Applications in Cloud Environments, *Applied Soft Computing*, **102**, 107-113