



TUGAS AKHIR - KS141501

**PERANCANGAN SISTEM OPTIMASI RUTE DISTRIBUSI
PENGANGKUTAN SAMPAH DI KABUPATEN SIDOARJO
MENGUNAKAN ALGORITMA *ANT COLONY*
OPTIMIZATION (ACO)**

***THE DESIGN OF OPTIMIZATION SYSTEM OF WASTE
TRANSPORT DISTRIBUTION ROUTE IN SIDOARJO
REGENCY USING ANT COLONY OPTIMIZATION (ACO)
ALGORITHM***

Kirana Gita Larasati
5213 100 056

Dosen Pembimbing :
Edwin Riksakomara, S.Kom., M.T.

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KS141501

**PERANCANGAN SISTEM OPTIMASI RUTE DISTRIBUSI
PENGANGKUTAN SAMPAH DI KABUPATEN SIDOARJO
MENGUNAKAN ALGORITMA ANT COLONY
OPTIMIZATION (ACO)**

***THE DESIGN OF OPTIMIZATION SYSTEM OF WASTE
TRANSPORT DISTRIBUTION ROUTE IN SIDOARJO
REGENCY USING ANT COLONY OPTIMIZATION (ACO)
ALGORITHM***

**Kirana Gita Larasati
5213 100 056**

**Dosen Pembimbing :
Edwin Riksakomara, S.Kom., M.T.**

**DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017**

LEMBAR PENGESAHAN
PERANCANGAN SISTEM OPTIMASI RUTE
DISTRIBUSI PENGANGKUTAN SAMPAH DI
KABUPATEN SIDOARJO MENGGUNAKAN
ALGORITMA *ANT COLONY OPTIMIZATION* (ACO)

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Departemen Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

KIRANA GITA LARASATI

NRP. 5213100056

Surabaya, Juli 2017

KEPALA
DEPARTEMEN SISTEM INFORMASI

Dr. Ir. Aris Tjahyanto, M.Kom

NIP. 196503101991021001



LEMBAR PERSETUJUAN

PERANCANGAN SISTEM OPTIMASI RUTE DISTRIBUSI PENGANGKUTAN SAMPAH DI KABUPATEN SIDOARJO MENGGUNAKAN ALGORITMA ANT COLONY OPTIMIZATION (ACO)

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Departemen Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

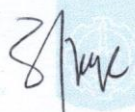
Oleh:

KIRANA GITA LARASATI


NRP. 5213100056

Disetujui Tim Penguji : Tanggal Ujian: 6 Juli 2017
Periode Wisuda : September 2017

Edwin Riksakomara, S.Kom., M.T.


(Pembimbing I)

Wiwik Anggraeni, S.Si., M.Kom.


(Penguji I)

Faizal Mahananto, S.Kom, M.Eng., Ph.D


(Penguji II)

PERANCANGAN SISTEM OPTIMASI RUTE DISTRIBUSI PENGANGKUTAN SAMPAH DI KABUPATEN SIDOARJO MENGGUNAKAN ALGORITMA ANT COLONY OPTIMIZATION (ACO)

Nama Mahasiswa : Kirana Gita Larasati
NRP : 5213100056
Jurusan : Sistem Informasi FTIf - ITS
Dosen Pembimbing : Edwin Riksakomara, S.Kom., M.T.

ABSTRAK

Permasalahan sampah merupakan permasalahan umum yang terjadi di setiap kota dan sulit untuk dihindari, tak terkecuali Kabupaten Sidoarjo. Banyak faktor yang menjadi penyebab terjadinya permasalahan sampah di setiap kota/kabupaten, baik dari kurangnya kesadaran masyarakat dalam menjaga kebersihan lingkungan, juga kurang maksimalnya kinerja pemerintah, dalam hal ini Dinas Lingkungan Hidup dan Kebersihan setempat dalam melakukan pengelolaan sampah. Sehingga, pengangkutan sampah menjadi proses yang penting dalam melakukan pengelolaan sampah setempat

Dengan luas wilayah Kabupaten Sidoarjo sebesar 714,24 km² atau seluas dua kali luas wilayah Kota Surabaya, Dinas Lingkungan Hidup dan Kebersihan setempat harus melakukan pengangkutan sampah dengan jarak yang cukup jauh dari setiap kecamatan menuju Tempat Pembuangan Akhir (TPA) yang terletak di Jabon. Sehingga, diperlukannya suatu sistem yang dapat membantu pemerintah setempat dalam menentukan rute pengangkutan sampah yang mencakup seluruh titik Tempat Pembuangan Sementara (TPS) yang ada di setiap kecamatan.

Oleh Karena itu, penelitian ini bertujuan untuk membentuk sebuah sistem optimasi rute distribusi pengangkutan sampah yang dapat membantu dinas kebersihan di pemerintahan setempat dalam menentukan rute yang optimal dalam pengangkutan sampah. Penelitian tugas akhir ini menggunakan Algoritma Ant Colony Optimization dalam menentukan pencarian rute di setiap TPS. Sedangkan aplikasi sistem yang memberikan visualisasi hasil rute yang direkomendasikan yang akan dikembangkan adalah aplikasi berbasis web dengan menggunakan bahasa pemrograman HTML5, CSS3, Javascript dan PHP.

Dalam penelitian ini didapatkan rute baru dengan menggunakan algoritma ACO yang memangkas total jarak tempuh untuk 40 armada sebesar 160.97 km.

Kata Kunci: rute pengangkutan sampah, optimasi rute, optimasi TSP, Algoritma Koloni Semut

THE DESIGN OF OPTIMIZATION SYSTEM OF WASTE TRANSPORT DISTRIBUTION ROUTE IN SIDOARJO REGENCY USING ANT COLONY OPTIMIZATION (ACO) ALGORITHM

Nama Mahasiswa : Kirana Gita Larasati
NRP : 5213100056
Jurusan : Sistem Informasi FTIf - ITS
Dosen Pembimbing : Edwin Riksakomara, S.Kom., M.T.

ABSTRACT

Waste is a public issue that always happen in every city and hard to avoid. Sidoarjo also faces the same issue. Many factors that cause the waste issue such as lack of public hygiene awareness and the lack of government's performance in waste management. So, waste transferring become an important process in the cycle of waste management.

Sidoarjo District area is around 714,24 km², twice of Surabaya. The government of Sidoarjo District should transfer the waste far from every sub-district to the final landfills in Jabon. Therefore the government requires a system to determine the waste transfer routing that covers all landfills in every sub-district.

This research aims to develop an optimization system of waste transfer route to help the government to determine the optimum route. We use Ant Colony Optimization Algorithm to determine the shortest route from every sub-district's landfills. The visualisation application that show the recommended routes that will be developed is a web-based application using HTML5, CSS3, Javascript and PHP.

The output in this research is new routes using the Ant Colony Optimization (ACO) algorithm that reduces mileage about 160.97 km.

Keywords: waste transport route, route optimization, TSP optimization, Ant Colony Optimization

KATA PENGANTAR

Segala puji dan syukur tak lupa penulis panjatkan kepada Allah SWT yang telah memberikan rahmat dan kekuatan pada penulis sehingga dapat menyelesaikan buku tugas akhir dengan judul:

**PERANCANGAN SISTEM OPTIMASI RUTE
DISTRIBUSI PENGANGKUTAN SAMPAH DI
KABUPATEN SIDOARJO MENGGUNAKAN
ALGORITMA *ANT COLONY OPTIMIZATION* (ACO)**

Pada kesempatan ini, saya ingin menyampaikan terima kasih kepada semua pihak yang telah memberikan dukungan, bimbingan, arahan, bantuan, dan semangat dalam menyelesaikan tugas akhir ini, yaitu kepada:

1. Allah SWT yang telah memberikan rahmat dan karuniaNya sehingga penulis dapat menyelesaikan tugas akhir tepat waktu.
2. Orangtua dan adik-adik penulis yang telah selalu mendoakan dan senantiasa mendukung penulis.
3. Bapak Edwin Riksakomara, selaku dosen pembimbing yang telah meluangkan waktu untuk membimbing penulis dalam pembuatan tugas akhir ini.
4. Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo, khususnya bapak Nur Achmad yang senantiasa meluangkan waktu untuk memberikan wawancara dan data terkait tugas akhir ini.
5. Wildan Fajria Lazuardy, selaku sosok terdekat yang sudah senantiasa membantu dan menemani selama pengerjaan tugas akhir.
6. Sahabat-sahabat penulis selama 4 tahun di Jurusan ini : Oriehanna Esesiawati, Fitri Larasati, Friska Izza Amalia dan Alitya Novianda Adityaputri yang selalu berhasil menggagalkan pengerjaan tugas akhir ini ☺
7. Harun Rizal yang sudah membantu pembuatan beberapa bagian dalam tugas akhir ini.

8. Fahrizal Adi Wibowo dan teman sekelompoknya, yang sudah membantu beberapa bagian kecil dalam tugas akhir ini.
9. Mas Ricky, selaku admin lab RDIB yang sudah membantu penulis dalam hal administrasi penyelesaian tugas akhir.
10. Oryza Khairunnisa dan Ashma Hanifah sahabat super yang telah senantiasa mendengarkan keluh kesah penulis ☺
11. Teman-teman lab RDIB dan Beltranis yang telah menemani penulis berjuang sampai pada penyelesaian tugas akhir ini.
12. Pihak-pihak lain yang telah mendukung dan membantu dalam kelancaran penyelesaian tugas akhir.

Penyusunan laporan ini masih jauh dari sempurna, untuk itu saya menerima adanya kritik dan saran yang membangun untuk perbaikan di masa mendatang. Semoga buku tugas akhir ini dapat memberikan manfaat pembaca.

Surabaya, 6 Juli 2017

Penulis

DAFTAR ISI

ABSTRAK	i
ABSTRACT	iii
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiii
DAFTAR PERSAMAAN	xiv
DAFTAR KODE PROGRAM	xv
DAFTAR DIAGRAM	xvi
BAB I PENDAHULUAN	1
1.1. Latar Belakang Masalah	1
1.2. Perumusan Masalah	4
1.3. Batasan Masalah	4
1.4. Tujuan Penelitian Tugas Akhir	4
1.5. Manfaat Penelitian Tugas Akhir	5
1.6. Relevansi Tugas Akhir	5
BAB II TINJAUAN PUSTAKA	7
2.1. Penelitian Sebelumnya	7
2.2. Dasar Teori	10
2.2.1. Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo	10
2.2.2. Teknik Pengelolaan dan Pengangkutan Sampah di Perkotaan	10

2.2.3.	Optimasi	13
2.2.4.	Ant Colony Optimization (ACO)	15
BAB III METODOLOGI PENELITIAN		25
3.1.	Diagram Metodologi.....	25
3.2.	Uraian Metodologi.....	26
3.2.1.	Identifikasi dan Perumusan Masalah	26
3.2.2.	Tinjauan Pustaka.....	26
3.2.3.	Pengumpulan dan pengolahan Data.....	26
3.2.4.	Pemodelan Permasalahan Rute Pengangkutan Sampah	27
3.2.5.	Penerapan Algoritma Ant Colony Optimization (ACO)	27
3.2.6.	Visualisasi Hasil Algoritma Ant Colony Optimization pada Peta.....	27
3.2.7.	Evaluasi	28
3.2.8.	Penyusunan Laporan Tugas Akhir.....	28
BAB IV PERANCANGAN		31
4.1.	Pengumpulan Data.....	31
4.1.1.	Data Pembagian Armada dan Supir Angkutan Sampah yang Bertugas	31
4.1.2.	Potret Pengelolaan Sampah di Kabupaten Sidoarjo	33
4.2.	Pengembangan Algoritma ACO pada Program Pencarian Rute Pengangkutan Sampah	34
4.3.	Proses Perancangan Use Case Diagram Aplikasi ...	34
4.3.1.	Penentuan Actor, Use Case, dan Relation	34
4.3.2.	Pembuatan Use Case Diagram	36

4.3.3.	Penjelasan Use Case Diagram.....	37
BAB V IMPLEMENTASI.....		41
5.1.	Lingkungan Uji Coba.....	41
5.2.	Persiapan Pengembangan Program	41
5.3.	Implementasi ACO pada Program	42
5.3.1.	Inisialisasi Package ke dalam Project Program 42	
5.3.2.	Input Data.....	42
5.3.3.	Inisiasi Jarak Antar Titik	43
5.3.4.	Pencarian Solusi Rute Terpendek.....	44
5.3.5.	Menampilkan Hasil	45
5.3.6.	Fungsi Penghitungan Running Time.....	45
5.3.7.	Contoh Hasil Keluaran	46
5.4.	Pengembangan Antar Muka Pencarian Rute Pengangkutan Sampah	46
5.4.1.	Inisialisasi library ke dalam project folder	46
5.4.2.	Input Data Titik TPS Sidoarjo.....	49
5.4.3.	Desain Antar Muka dan Fitur Aplikasi	50
5.5.	Penjelasan Alur Kerja Aplikasi Sistem Optimasi...60	
BAB VI HASIL DAN PEMBAHASAN		63
6.1.	Lingkungan Uji Coba.....	63
6.2.	Uji Coba Program Pencarian Rute	64
6.3.	Hasil Tampilan Visualisasi Rekomendasi Rute	66
6.4.	‘ Hasil Rekomendasi Rute Pengangkutan Sampah	67
6.5.	Analisa Hasil Pengujian	71
BAB VII KESIMPULAN DAN SARAN		73

7.1. KESIMPULAN	73
7.2. SARAN.....	73
DAFTAR PUSTAKA.....	75
LAMPIRAN A	79
LAMPIRAN B.....	93
LAMPIRAN C.....	95

DAFTAR GAMBAR

Gambar 2.1 Ilustrasi pencarian makanan pada Ant Colony Optimization.....	16
Gambar 2.2 Bentuk komunikasi stigmergy pada Ant Colony Optimization.....	17
Gambar 4.1 Aktor yang terlibat dalam Use Case adalah User	35
Gambar 4.2 Salah satu contoh Use Case dalam aplikasi.....	35
Gambar 4.3 Salah satu bentuk relasi pada Diagram.....	36
Gambar 4.4 Use Case Diagram dari Aplikasi Sistem Optimasi Rute	36
Gambar 5.1 Contoh keluaran yang dihasilkan	46
Gambar 5.2 Inisialisasi library ke dalam project folder	47
Gambar 5.3 Halaman Dasbor Aplikasi	51
Gambar 5.4 Fitur memilih armada	53
Gambar 5.5 Fitur drop-down menu	53
Gambar 6.1 Hasil visualisasi rute.....	66
Gambar C.1 Visualisasi Rute Armada1	95
Gambar C.2 Visualisasi Rute Armada2	96
Gambar C.3 Visualisasi Rute Armada3	97
Gambar C.4 Visualisasi Rute armada4	97
Gambar C.5 Visualisasi Rute Armada5	98
Gambar C.6 Visualisasi Rute Armada6	99
Gambar C.7 Visualisasi Rute Armada7	100
Gambar C.8 Visualisasi Rute Armada8	101
Gambar C.9 Visualisasi Rute sub-wilayah Armada9.....	102
Gambar C.10 Visualisasi Rute Armada10	103
Gambar C.11 Visualisasi Rute armada11.....	104
Gambar C.12 Visualisasi Rute armada12.....	105
Gambar C.13 Visualisasi Rute armada13.....	106
Gambar C.14 Visualisasi Rute armada14.....	107
Gambar C.15 Visualisasi Rute armada15.....	108
Gambar C.16 Visualisasi Rute armada16.....	109

Gambar C.17 Visualisasi Rute armada17.....	110
Gambar C.18 Visualisasi Rute armada18.....	111
Gambar C.19 Visualisasi Rute Armada 19.....	112
Gambar C.20 Visualisasi Rute Armada20.....	113
Gambar C.21 Visualisasi Rute Armada 21.....	114
Gambar C.22 Visualisasi Rute Armada 22.....	115
Gambar C.23 Visualisasi Rute Armada 23.....	116
Gambar C.24 Visualisasi Rute Armada 24.....	117
Gambar C.25 Visualisasi Rute Armada 25.....	118
Gambar C.26 Visualisasi Rute Armada 26.....	119
Gambar C.27 Visualisasi Rute Armada27.....	120
Gambar C.28 Visualisasi Rute Armada28.....	121
Gambar C.29 Visualisasi Rute Armada29.....	122
Gambar C.30 Visualisasi Rute Armada30.....	123
Gambar C.31 Visualisasi Rute Armada31.....	123
Gambar C.32 Visualisasi Rute Armada32.....	124
Gambar C.33 Visualisasi Rute Armada33.....	125
Gambar C.34 Visualisasi Rute Armada34.....	126
Gambar C.35 Visualisasi Rute Armada35.....	127
Gambar C.36 Visualisasi Rute Armada36.....	128
Gambar C.37 Visualisasi Rute Armada37.....	128
Gambar C.38 Visualisasi Rute Armada38.....	129
Gambar C.39 Visualisasi Rute Armada39.....	130
Gambar C.40 Visualisasi Rute Armada40.....	131

DAFTAR TABEL

Tabel 4.1 Data Pengangkutan Sampah Kabupaten Sidoarjo ..	32
Tabel 4.2 Penjelasan Use Case Melihat Dashboard Aplikasi Sistem Optimasi Rute.....	37
Tabel 4.3 Penjelasan Use Case Memilih Titik Berdasarkan Armada.....	38
Tabel 4.4 Penjelasan Use Case Mendapatkan Rute Terpendek	39
Tabel 6.1 Perangkat Keras Pengujian	63
Tabel 6.2 Perangkat Lunak Pengujian.....	63
Tabel 6.3 Hasil Pengujian Berdasarkan Parameter	65
Tabel 6.4 Perbandingan Jarak Tempuh	67
Tabel 6.5 Hasil Rekomendasi Rute Baru	68
Tabel B.1 Hasil Pengujian Berdasarkan Perubahan Parameter	93

DAFTAR PERSAMAAN

Persamaan 4.1.....	21
Persamaan 4.2.....	22
Persamaan 4.3.....	23
Persamaan 4.4.....	23

DAFTAR KODE PROGRAM

Kode Program 5.1 Inisialisasi package dalam program	42
Kode Program 5.2 Input Data Titik Koordinat.....	42
Kode Program 5.3 Rumus Penghitungan Jarak Antar Dua Titik	43
Kode Program 5.4 Fungsi Penghitungan Jarak Antar Titik ...	44
Kode Program 5.5 Fungsi Input Program Pencarian Rute	44
Kode Program 5.6 Fungsi Pencarian Rute	44
Kode Program 5.7 Output Rekomendasi Rute Terpendek	45
Kode Program 5.8 Fungsi Memulai Running Time Program	45
Kode Program 5.9 Fungsi Menghitung Lama Running Time	45
Kode Program 5.10 Inisialisasi library ke dalam kode program	48
Kode Program 5.11 Input data dalam file php	50
Kode Program 5.12 Fungsi menampilkan peta google maps.	52
Kode Program 5.13 kode program kerangka memilih kecamatan.....	55
Kode Program 5.14 Kode program fungsi submit	55
Kode Program 5.15 Kode program kerangka tombol Lihat Rute	55
Kode Program 5.16 Koneksi php dengan python.....	56
Kode Program 5.17 Ubah format data JSON.....	57
Kode Program 5.18 Membaca kode program python di php .	57
Kode Program 5.19 Membaca node yang diinputkan dari PHP	58
Kode Program 5.20 Menampilkan rute dalam peta.....	58
Kode Program 5.21 Ubah array menjadi format JSON.....	58
Kode Program 5.22 Visualisasi Rute	59
Kode Program 5.23 Visualisasi Rute	60

DAFTAR DIAGRAM

Diagram 2.1 Teknik operasional pengelolaan persampahan [14]	
.....	11
Diagram 2.2 Pengangkutan sampah dengan metode HCS [14]	
.....	12
Diagram 2.3 Pengangkutan sampah dengan metode SCS [14]	
.....	13
Diagram 3.1 Metodologi Penelitian.....	25
Diagram 5.1 Diagram aplikasi sistem.....	61

BAB I

PENDAHULUAN

Pada bab pendahuluan dibahas mengenai permasalahan yang di selesaikan pada tugas akhir ini meliputi latar belakang, perumusan masalah, batasan masalah, tujuan tugas akhir, dan manfaat kegiatan tugas akhir. Dengan adanya bab ini, diharapkan dapat mempermudah pembaca dalam memahami masalah yang ada dalam tugas akhir ini.

1.1. Latar Belakang Masalah

Sampah kota secara sederhana diartikan sebagai sampah organik maupun anorganik yang dibuang oleh masyarakat dari berbagai lokasi di kota tersebut. Sumber sampah umumnya berasal dari perumahan dan pasar. Permasalahan sampah merupakan hal yang krusial. Bahkan, sampah dapat dikatakan sebagai masalah kultural karena dampaknya terkena pada berbagai sisi kehidupan, terutama kota-kota besar seperti Jakarta, Semarang, Surabaya, Bandung, Palembang dan Medan. Menurut prakiraan, volume sampah yang dihasilkan per orang rata-rata sekitar 0,5kg/kapita/hari [1].

Kabupaten Sidoarjo merupakan salah satu kabupaten yang ada di Provinsi Jawa Timur. Dibandingkan ke-28 kabupaten lainnya yang ada di Provinsi Jawa Timur, Kabupaten Sidoarjo merupakan kabupaten terkecil dengan jumlah penduduk terpadat [2]. Kabupaten Sidoarjo terdiri dari 18 kecamatan dan memiliki luas wilayah 714,24 km² dengan kepadatan penduduk 3.026,39 jiwa per km². Pertumbuhan penduduk di pertengahan tahun 2015 tercatat sebesar 14,343% [3].

Permasalahan sampah menjadi fokus utama yang masih saja terjadi di kota dan kabupaten di Indonesia, tak terkecuali Kabupaten Sidoarjo. Pertumbuhan penduduk yang kian pesat menjadi salah satu faktor meningkatnya tumpukan sampah di Kabupaten Sidoarjo. Selain itu, minimnya kesadaran warga dalam menjaga kebersihan lingkungan dan kurang maksimalnya kinerja petugas Dinas Kebersihan juga

menambah faktor permasalahan persampahan. Jumlah desa yang terlampaui banyak dengan jarak tempuh yang cukup jauh dan luas wilayah yang bedar menyebabkan pengangkutan sampah tidak bisa dilaksanakan tepat waktu sehingga menyebabkan penumpukan sampah yang mengganggu aktivitas warga setiap harinya [4].

Kegiatan pengelolaan sampah adalah kegiatan yang sistematis, menyeluruh, dan berkesinambungan yang meliputi perencanaan, pengurangan dan penanganan sampah. Kegiatan penanganan sampah yang dilakukan pemerintah daerah Kabupaten Sidoarjo dilakukan dengan cara pemilahan, pengumpulan, pengangkutan, pengolahan dan pemrosesan akhir sampah. Kegiatan pengumpulan sampah dilakukan sejak pemindahan sampah dari tempat sampah rumah tangga ke TPS (Tempat Penampungan Sementara)/TPST (Tempat Pengolahan Sampah Terpadu) sampai ke TPA (Tempat Pemrosesan Akhir) dengan tetap menjamin terpisahnya sampah sesuai dengan jenis sampah.

Pola pengangkutan sampah dibagi menjadi 2 metode, yakni *Stationary Container System* (SCS) dan *Hauled Container System* (HCS). Perbedaan dari kedua metode tersebut terdapat pada kendaraan pengangkut. Jika metode SCS menggunakan *dump truck* atau *compactor truck*, sedangkan metode HCS menggunakan *armroll truck* [5]. Pengangkutan sampah yang dilakukan di Kabupaten Sidoarjo menggunakan kedua metode tersebut, dengan menggunakan armada pengangkutan sampah *dump truck* dan *armroll truck* yang melayani pengangkutan sampah di TPS maupun TPST berbeda-beda sesuai dengan lokasi dan jarak tempuh pengangkutan sampai ke TPA Jabon [6]. Proses pengangkutan sampah merupakan salah satu proses yang berperan penting dalam pengelolaan sampah, khususnya dalam sektor biaya. Biaya yang terdapat pada pengangkutan sampah bisa berupa bensin untuk menjalankan armada angkutan sampah, dan upah supir dan tenaga kerja kebersihan pada jam kerja aktif saat melakukan pengangkutan sampah.

Lokasi kecamatan di Kabupaten Sidoarjo yang cukup jauh menjadi salah satu penyebab besarnya biaya pengangkutan sampah yang dilakukan. Sehingga, penentuan rute pengangkutan sampah merupakan faktor penting dalam menentukan biaya pengelolaan sampah yang paling optimum.

Permasalahan pengangkutan sampah termasuk ke dalam permasalahan rute kendaraan atau *Vehicle Routing Problem* (VRP) yang bertujuan untuk menentukan rute yang paling efisien dalam suatu kegiatan distribusi dengan jumlah kendaraan yang ditentukan [7]. Batasan-batasan pada penyelesaian VRP antara lain kapasitas kendaraan, waktu pengantaran dan persebaran depot dan pelanggan. Dan solusi dari penyelesaian masalah VRP adalah kombinasi rute terbaik yang melayani seluruh permintaan pelanggan [8]. Dalam kasus ini, persebaran depot berupa titik-titik lokasi TPS dengan permintaan pelanggan berupa sejumlah sampah yang harus diangkut dari masing-masing lokasi TPS.

VRP adalah permasalahan optimasi yang termasuk ke dalam kelas *combinatorial optimization* (CO). CO adalah masalah optimasi yang bertujuan untuk meminimalkan suatu biaya [7]. VRP adalah permasalahan kombinasi yang secara umum termasuk dalam masalah *non-polynomial hard* (NP-hard). Permasalahan rute kendaraan dapat diselesaikan dengan menggunakan metode heuristik dan metaheuristik [9]. Metode metaheuristik antara lain adalah *iterated local search*, *simulated annealing*, *tabu search*, algoritma genetika dan optimasi *Ant Colony Optimization*. Sedangkan metode metaheuristik yang telah digunakan dalam penelitian untuk menyelesaikan permasalahan VRP antara lain *simulated annealing*, *tabu search*, *granular tabu search*, algoritma genetika, *variable neighborhood search* dan optimasi *Ant Colony Optimization*.

Tugas akhir ini akan menggunakan algoritma optimasi *Ant Colony Optimization* dalam menentukan rute pengangkutan sampah yang paling efisien dalam meminimumkan biaya dari salah satu proses dalam pengelolaan sampah yang dilakukan

oleh Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo.

1.2. Perumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan sebelumnya, rumusan masalah dari penelitian ini antara lain:

1. Bagaimana penerapan Algoritma *Ant Colony Optimization* ke dalam sistem optimasi rute pengangkutan sampah di Kabupaten Sidoarjo?
2. Bagaimana penerapan hasil optimasi rute yang dihasilkan dari sistem divisualisasikan ke dalam tampilan peta

1.3. Batasan Masalah

Batasan masalah dari penelitian ini adalah:

1. Studi kasus yang digunakan pada tugas akhir ini adalah permasalahan pengangkutan sampah di Kabupaten Sidoarjo
2. Data mengenai lokasi Tempat Pembuangan Sementara dan daftar armada yang bertugas melayani pengangkutan sampah di titik-titik yang sudah ditentukan didapatkan dari laporan tahunan Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo tahun 2016
3. Pencarian rute pengangkutan sampah yang optimum dilakukan dengan menggunakan Algoritma *Ant Colony Optimization* dengan tidak mempertimbangkan kapasitas pengangkutan dan volume sampah yang dihasilkan di masing-masing TPS
4. Hasil dari sistem pencarian rute pengangkutan sampah yang optimum ini divisualisasikan dalam tampilan peta

1.4. Tujuan Penelitian Tugas Akhir

Tujuan dilakukannya tugas akhir ini adalah menerapkan Algoritma *Ant Colony Optimization* dalam sistem optimasi rute

pengangkutan sampah di Kabupaten Sidoarjo yang dapat membantu Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo dalam menentukan rute pelayanan pengangkutan sampah yang optimum untuk meminimalisir biaya pengangkutan sampah.

1.5. Manfaat Penelitian Tugas Akhir

Manfaat yang diberikan dengan adanya tugas akhir ini adalah:

1. Memberikan gambaran kepada Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo mengenai referensi solusi rute pengangkutan sampah alternatif yang lebih optimum untuk mengatasi permasalahan pengangkutan sampah di Kabupaten Sidoarjo
2. Menjadi bahan pertimbangan dalam pengambilan keputusan berupa solusi yang tepat untuk mengatasi permasalahan pengangkutan sampah di Kabupaten Sidoarjo.

1.6. Relevansi Tugas Akhir

Tugas akhir ini berkaitan dengan mata kuliah Bahasa Pemrograman, Riset Operasi dan Riset Operasi Lanjut yang tercakup pada Laboratorium Rekayasa Data Dan Intelegensi Bisnis.

[halaman ini sengaja dikosongkan]

BAB II TINJAUAN PUSTAKA

Dalam bab ini akan menjelaskan mengenai penelitian sebelumnya, profil daerah yang digunakan dalam studi kasus penelitian dan tinjauan pustaka yang digunakan sebagai dasar acuan dalam menyelesaikan tugas akhir.

2.1. Penelitian Sebelumnya

Judul Paper	An improved ant colony optimization for vehicle routing problem [10]
Penulis ; Tahun	Yu Bin, Yang Zhong-Zhen, Yao Baozhen; 2009
Deskripsi Umum Penelitian	Permasalahan rute kendaraan atau <i>Vehicle Routing Problem</i> (VRP) adalah permasalahan optimasi kombinatorial yang cukup terkenal yang memegang peranan penting dalam manajemen logistik. Paper ini mengusulkan algoritma <i>Ant Colony Optimization</i> dengan improvisasi untuk menyelesaikan beberapa permasalahan rute kendaraan dan membandingkan hasil penyelesaian yang didapatkan dengan hasil penyelesaian dengan metode yang digunakan sebelumnya. Hasil pada paper ini menunjukkan algoritma <i>Ant Colony Optimization</i> dengan improvisasi yang diusulkan memberikan hasil yang lebih efektif dan efisien dibandingkan dengan algoritma yang digunakan sebelumnya.
Keterkaitan Penelitian	Dasar penggunaan algoritma pada paper ini menjadi acuan dalam pengerjaan tugas akhir

Judul Paper	Integrating a heterogeneous fixed fleet and a flexible assignment of destination depots in the waste collection VRP with intermediate facilities [11]
--------------------	---

Penulis Tahun ;	Iliya Markov, Sacha Varone, Michael Bierlaire ; 2015
Deskripsi Umum Penelitian	Permasalahan kegiatan pengumpulan sampah dipertimbangkan sebagai permasalahan rute kendaraan dalam kelas yang lebih luas dengan mengintegrasikan armada tetap heterogen dan tugas fleksibel di depot tujuan menggunakan fasilitas menengah. Dengan beberapa batasan tambahan yang ada di lapangan, seperti lama waktu istirahat yang ditentukan berdasarkan waktu mulai perjalanan, kapasitas beberapa kendaraan, dan lainnya. Penelitian ini merekomendasikan penyelesaian menggunakan lingkungan pencarian heuristik untuk menyelesaikan beberapa contoh permasalahan di dunia nyata dengan karakteristik yang berbeda-beda. Hasil penelitian ini menunjukkan bahwa metode heuristik mampu menyelesaikan beberapa contoh kecil mendekati optimal, dengan menunjukkan kinerja yang kompetitif jika dibandingkan dengan metode solusi yang paling mutakhir dalam beberapa kasus tertentu dan mengarah pada penghematan paling penting. Bahkan, metode heuristik juga menyoroti dan mengukur penghematan dengan dilakukannya tugas depot yang fleksibel.
Keterkaitan Penelitian	Permasalahan pada penelitian ini menjadi acuan dalam pengerjaan tugas akhir.

Judul Paper	An Ant Colony system empowered variable neighborhood search algorithm for the vehicle routing problem with simultaneous pickup and delivery
Penulis Tahun ;	Can B. Kalayci, Can Kaya; 2016 [12]

<p>Deskripsi Umum Penelitian</p>	<p>Penggunaan Algoritma <i>Ant Colony Optimization</i> (ACS – <i>Ant Colony System</i>) yang mendukung penggunaan algoritma Pencarian Lingkungan (VNS – <i>Variable Neighborhood Search</i>) dalam permasalahan rute kendaraan dengan mempertimbangkan pelayanan campuran, yaitu pengiriman dan pengambilan barang yang biasa disebut <i>Vehicle Routing Problem with Simultaneous Pickup and Delivery</i> (VRSPD) yang tergolong ke dalam permasalahan NP-Hard. VNS adalah algoritma optimasi yang kuat yang menyediakan pencarian lokal yang intensif, namun sayangnya algoritma ini tidak memiliki struktur memori. Oleh karena itu, digunakan ACS untuk meminimalisir kekurangan tersebut dengan memanfaatkan struktur memori jangka panjang yang dimiliki. Sehingga, kinerja keseluruhan algoritma bisa meningkat. Percobaan penggunaan algoritma ini diterapkan pada beberapa penelitian dengan permasalahan yang sama untuk menjadi perbandingan hasil dari algoritma ini dengan algoritma lainya pada penelitian sebelumnya. Hasilnya, didapatkan bahwa penelitian ini menghasilkan nilai yang kuat dan sangat efisien baik dari segi kualitas solusi dan waktu <i>running</i> CPU karena nilai yang lebih baik mampu dihasilkan dalam kurun waktu yang cukup singkat.</p>
<p>Keterkaitan Penelitian</p>	<p>Referensi algoritma pada penelitian ini akan menjadi acuan pengerjaan tugas akhir.</p>

2.2. Dasar Teori

2.2.1. Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo

Berdasarkan Pasal 4 dan Pasal 5 Peraturan Bupati Sidoarjo Nomor 77 Tahun 2016, menjelaskan bahwa pada Pasal 4, Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo mempunyai tugas membantu Bupati melaksanakan urusan pemerintahan di bidang lingkungan hidup dan tugas pembantuan yang diberikan kepada kabupaten.

Dan untuk melaksanakan tugas sebagaimana dimaksud dalam Pasal 4, Dinas Lingkungan Hidup dan Kebersihan menyelenggarakan fungsi sebagai berikut:

- a) Perumusan kebijakan bidang lingkungan hidup
- b) Pelaksanaan kebijakan bidang lingkungan hidup
- c) Pelaksanaan evaluasi dan pelaporan bidang lingkungan hidup
- d) Pelaksanaan administrasi dinas lingkungan hidup
- e) Pelaksanaan tugas lain yang diberikan oleh Bupati sesuai dengan tugasnya [13].

2.2.2. Teknik Pengelolaan dan Pengangkutan Sampah di Perkotaan

Berdasarkan Standar Nasional Indonesia mengenai tata cara pengelolaan sampah perkotaan, teknik operasional pengelolaan persampahan terdiri dari kegiatan pewardahan sampai dengan pembuangan akhir sampah dan harus bersifat terpadu dengan melakukan pemilahan sejak dari sumbernya [14]. Teknik operasional pengelolaan persampahan dapat dilihat pada skema berikut:

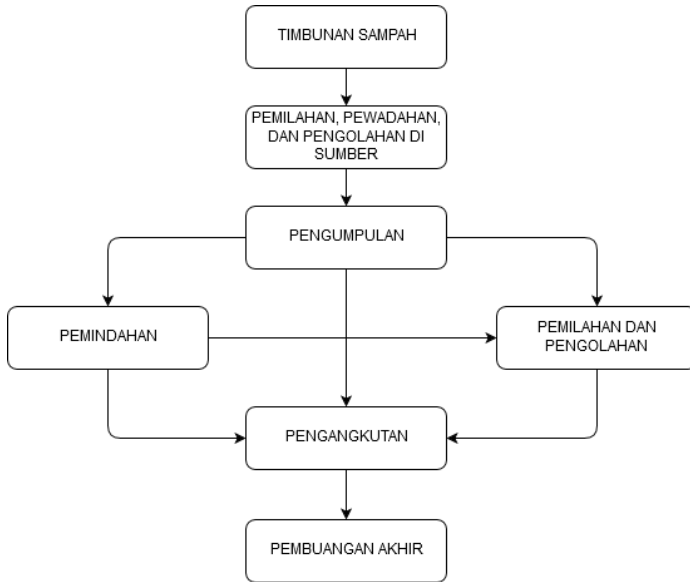


Diagram 2.1 Teknik operasional pengelolaan persampahan [14]

Terdapat dua jenis sistem pengangkutan sampah yang dapat dilakukan oleh Dinas Kebersihan di setiap kota/kabupaten yaitu dengan menggunakan sistem pemindahan (transfer depo), dengan proses pengangkutan menggunakan sistem kontainer angkat / *Hauled Container System* (HCS) dan sistem kontainer tetap / *Stationary Container System* (SCS).

Metode HCS menggunakan armada *arm roll truck*, yaitu truk dengan kontainer yang dapat diangkat atau diturunkan. Sistem pada metode HCS ini yaitu truk membawa kontainer kosong dari pool, menurunkan kontainer kosong ke TPS dengan membawa kontainer berisi sampah untuk diangkat ke TPA. Dari TPA kemudian akan membawa kontainer kosong menuju TPS dan kegiatan terus berulang sampai layanan TPS yang dijadwalkan di masing-masing truk terpenuhi.

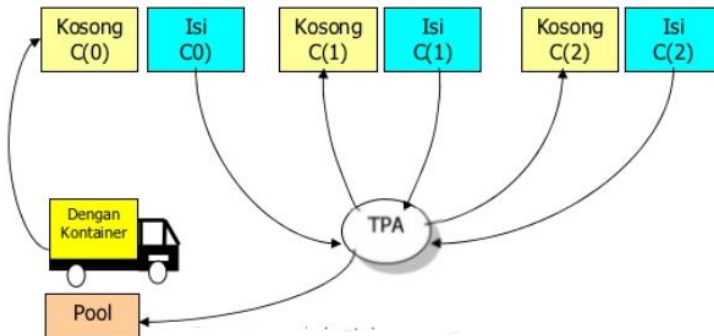


Diagram 2.2 Pengangkutan sampah dengan metode HCS [14]

Dengan proses pengangkutan pada metode HCS adalah sebagai berikut:

- Kendaraan dari pool dengan membawa kontainer kosong menuju lokasi kontainer isi untuk mengganti atau mengambil dan langsung membawanya ke TPA
- Kendaraan dengan membawa kontainer kosong dari TPA menuju kontainer isi berikutnya
- Demikian seterusnya sampai ritasi terakhir

Sedangkan untuk metode SCS menggunakan armada *dump truck*, yang merupakan truk yang mengangkut sampah dengan memindahkan sampah-sampah yang terdapat di masing-masing titik TPS ke dalam angkutan truk sampai penuh untuk kemudian dibawa ke TPA.

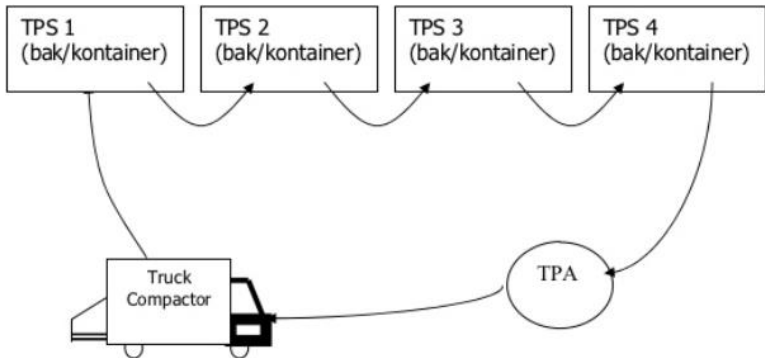


Diagram 2.3 Pengangkutan sampah dengan metode SCS [14]

Dengan proses pengangkutan pada metode SCS adalah sebagai berikut:

- Kendaraan dari *pool* menuju TPS pertama, sampah dimuat ke dalam truk
- Kendaraan menuju TPS berikutnya sampai truk penuh untuk kemudian menuju TPA
- Demikian seterusnya sampai ritasi terakhir

2.2.3. Optimasi

2.2.3.1. Definisi Masalah Optimasi

Optimisasi adalah suatu proses untuk mencapai hasil yang optimal (nilai efektif yang dapat dicapai). Dalam disiplin matematika optimisasi merujuk pada studi permasalahan yang mencoba untuk mencari nilai minimal atau maksimal dari suatu fungsi riil. Untuk dapat mencapai nilai optimal baik minimal atau maksimal tersebut, secara sistematis dilakukan pemilihan nilai variabel integer atau riil yang akan memberikan solusi optimal.

2.2.3.2. Macam-Macam Permasalahan Optimasi

Permasalahan yang berkaitan dengan optimisasi sangat kompleks dalam kehidupan sehari-hari. Nilai optimal yang didapat dalam optimisasi dapat berupa besaran panjang, waktu, jarak, dan lain-lain. Berikut ini adalah termasuk beberapa persoalan optimisasi:

1. Menentukan lintasan terpendek dari suatu tempat ke tempat yang lain.
2. Menentukan jumlah pekerja seminimal mungkin untuk melakukan suatu proses produksi agar pengeluaran biaya pekerja dapat diminimalkan dan hasil produksi tetap maksimal.
3. Mengatur rute kendaraan umum agar semua lokasi dapat dijangkau.
4. Mengatur *routing* jaringan kabel telepon agar biaya pemasangan kabel tidak terlalu besar dan penggunaannya tidak boros.

Selain beberapa contoh di atas, masih banyak persoalan lainnya yang terdapat dalam berbagai bidang.

2.2.3.3. Penyelesaian Masalah Optimasi

Secara umum, penyelesaian masalah pencarian jalur terpendek dapat dilakukan dengan menggunakan dua metode, yaitu metode konvensional dan metode heuristik. Metode konvensional diterapkan dengan perhitungan matematis biasa, sedangkan metode heuristik diterapkan dengan perhitungan kecerdasan buatan, dengan menentukan basis pengetahuan dan perhitungannya.

1. Metode Konvensional
Metode konvensional berupa algoritma yang menggunakan perhitungan matematis biasa. Ada beberapa metode yang biasa digunakan untuk melakukan pencarian jalur terpendek, diantaranya

algoritma Dijkstra, algoritma Floyd-Warshall, dan algoritma Bellman-Ford.

2. Metode Heuristik

Metode Heuristik adalah sub bidang dari kecerdasan buatan yang digunakan untuk melakukan pencarian dan optimasi. Ada beberapa algoritma pada metode heuristik yang biasa digunakan dalam permasalahan optimasi, diantaranya algoritma genetika, algoritma semut, logika fuzzy, jaringan syaraf tiruan, pencarian tabu, *simulated annealing*, dan lain-lain [17].

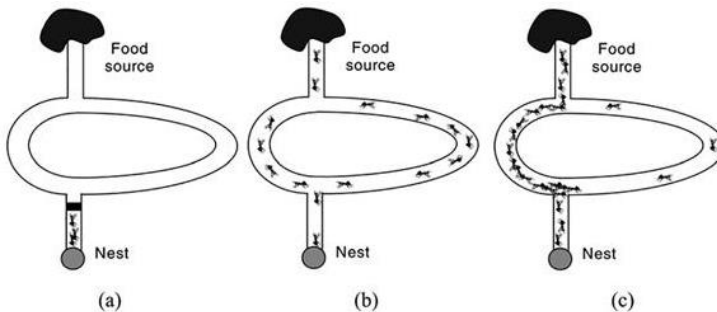
2.2.4. Ant Colony Optimization (ACO)

Pada kehidupan sebenarnya, semut – semut meninggalkan sarang untuk mencari makanan dan harus mencari kembali sarang mereka. Misalkan ada segerombolan semut yang mencari makanan, maka semut yang berada di depan harus memilih lintasan tertentu untuk dilewati. Pada saat semut pertama berjalan, semut tersebut meninggalkan *pheromone* yang dapat dicium oleh semut berikutnya, sehingga semut-semut berikutnya tahu apakah tempat tersebut sudah dilewati atau belum.

Algoritma semut diperkenalkan oleh Moyson dan Mendrik dan secara meluas dikembangkan oleh Macro Dorigo, merupakan teknik probalistik untuk menyelesaikan masalah komputasi dengan menemukan jalur terbaik melalui grafik. Algoritma ini terinspirasi oleh perilaku semut dalam menemukan jalur dari koloninya menuju makanan.

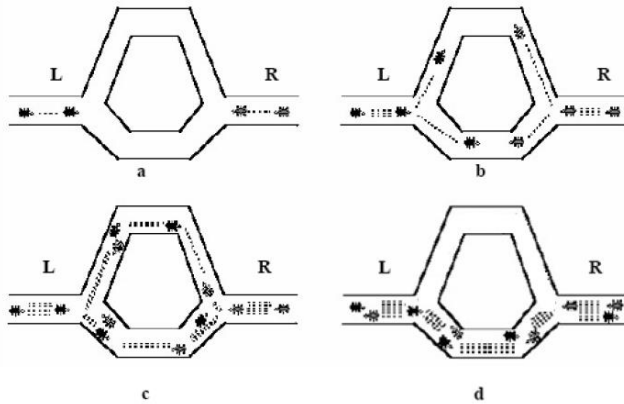
Algoritma *Ant Colony Optimization* (ACO) adalah algoritma yang bersifat heuristik untuk menyelesaikan masalah optimasi. Algoritma *Ant Colony Optimization* terinspirasi dari perilaku *Ant Colony Optimization* dalam mencari makan dan menargetkan masalah optimasi diskrit. Secara alamiah, *Ant Colony Optimization* mampu menemukan rute terpendek dalam

perjalanan dari sarang ke tempat-tempat sumber makanan berdasarkan jejak kaki yang mengandung *pheromone* pada lintasan yang telah dilalui oleh semut-semut lainnya. Semakin banyak semut yang melalui suatu lintasan, maka semakin jelas bekas jejak kakinya. Hal ini kemudian mengakibatkan, lintasan yang jarang dilalui semut pun menjadi berkurang kandungan *pheromone*-nya, dan kepadatan semut yang melalui lintasan tersebut akhirnya semakin berkurang atau bahkan tidak dilewati sama sekali. Dan sebaliknya, lintasan yang banyak dilalui semut akan bertambah kandungan *pheromone*-nya, sehingga lintasan tersebut akan semakin padat dan mungkin semua *Ant Colony Optimization* hanya akan melewati lintasan tersebut. Gambar berikut merupakan ilustrasi dari konsep *Ant Colony Optimization* [18].



Gambar 2.1 Ilustrasi pencarian makanan pada Ant Colony Optimization

Pada saat semut berjalan ia meninggalkan sejumlah informasi, yang disebut *pheromone* (dalam jumlah tertentu), ditempat yang dilaluinya dan menandai jalur tersebut. Dengan perantara *pheromone* inilah terjadi komunikasi tidak langsung dan juga pertukaran informasi antar semut selagi membangun suatu solusi. Bentuk komunikasi tidak langsung yang diperlihatkan oleh semut ini disebut *stigmergy*.



Gambar 2.2 Bentuk komunikasi stigmergy pada Ant Colony Optimization

Gambar 2a diatas menunjukkan ada dua kelompok semut yang akan melakukan perjalanan. Satu kelompok bernama L yaitu kelompok yang berangkat dari arah kiri yang merupakan sarang semut dan kelompok lain yang bernama kelompok R yang berangkat dari kanan yang merupakan sumber makanan. Kedua kelompok semut dari titik berangkat sedang dalam posisi pengambilan keputusan jalan sebelah mana yang akan diambil. Kelompok semut L membagi dua kelompok lagi. Sebagian melalui jalan atas dan sebagian melalui jalan bawah. Hal ini juga berlaku pada kelompok semut R. Gambar 2b dan gambar 2c menunjukkan bahwa kelompok semut berjalan pada kecepatan yang sama dengan meninggalkan *pheromone* atau jejak kaki di jalan yang telah dilalui. *Pheromone* yang ditinggalkan oleh kumpulan semut yang melalui jalan atas telah mengalami banyak penguapan karena semut yang melalui jalan atas berjumlah lebih sedikit dari pada jalan yang di bawah. Hal ini dikarenakan jarak yang ditempuh lebih panjang daripada jalan bawah. Sedangkan *pheromone* yang berada di jalan bawah, penguapannya cenderung lebih lama. Karena semut yang melalui jalan bawah lebih banyak daripada semut yang

melalui jalan atas. Gambar 2d menunjukkan bahwa semut-semut yang lain pada akhirnya memutuskan untuk melewati jalan bawah karena *pheromone* yang ditinggalkan masih banyak. Sedangkan *pheromone* pada jalan atas sudah banyak menguap sehingga semut-semut tidak memilih jalan atas tersebut. Semakin banyak semut yang melalui jalan bawah maka semakin banyak semut yang mengikutinya. Demikian juga dengan jalan atas, semakin sedikit semut yang melalui jalan atas, maka *pheromone* yang ditinggalkan semakin berkurang bahkan hilang. Dari sinilah kemudian terpilihlah jalur terpendek antara sarang dan sumber makanan. Hal ini berarti bahwa semakin banyak semut yang mengikuti sebuah jalur maka semakin bertambah menariklah jalur tersebut untuk dilalui. Probabilitas dimana seekor semut memutuskan untuk mengikuti suatu jalur meningkat dengan banyaknya semut yang lebih dulu menggunakan jalur tersebut [19].

Dalam algoritma *Ant Colony System* (ACS) terdapat jumlah semut yang masing-masing akan melakukan perjalanan dari satu kota ke kota yang lainnya. Banyaknya semut yang akan melakukan perjalanan ditentukan oleh pengguna ACS. Semakin banyak jumlah semut pada ACS, maka solusi yang akan diperoleh akan lebih baik dibandingkan dengan jumlah semut yang lebih sedikit pada jumlah iterasi yang sama, jumlah semut yang semakin banyak akan mengakibatkan waktu komputasi untuk memperoleh solusi menjadi lebih lama. Pada saat semut melakukan perjalanan untuk mencari solusi dari TSP ia akan dipengaruhi oleh jejak *pheromone* dan jarak antar kota. Jumlah *pheromone* antara kota i dengan kota j akan dilambangkan dengan $\tau(i, j)$. Nilai $\tau(i, j)$ akan selalu dimodifikasi selama algoritma ACS dijalankan [18].

Ada tiga buah aturan yang akan digunakan dalam ACS yaitu:

1. Aturan *State Transition*

Aturan ini digunakan oleh semut untuk memutuskan ke kota mana ia akan pergi. Dengan menerapkan aturan ini, semut dapat memilih untuk pergi kesisi yang baru (sisi yang belum dilewati oleh semut) atau sisi yang terbaik (sisi yang memiliki jumlah *pheromone* terbanyak dan jarak terpendek) secara probabilitas.

2. Aturan *Local updating*

Setelah semut melalui sisi tertentu, maka jumlah *pheromone* pada sisi tersebut akan berkurang. Jumlah *pheromone* pada sisi yang baru saja dikunjungi oleh semut dapat diperoleh dengan menerapkan *local updating*.

3. Aturan *Global Updating*

Aturan ini digunakan setelah semua semut membentuk jalur perjalanan. Pada aturan *global updating* akan dilakukan pengurangan jumlah *pheromone* pada semua sisi, kemudian dilakukan penambahan jumlah *pheromone* pada sisi-sisi yang termasuk dalam perjalanan dengan jarak yang terpendek.

Dalam algoritma ACS akan dilakukan proses iterasi (pengulangan dalam mencari jalur terpendek dan jarak perjalanan yang terpendek). Banyak iterasi yang dilakukan sesuai dengan banyaknya iterasi yang diinginkan oleh pengguna. Dalam sebuah proses iterasi, semua semut akan melakukan perjalanan sampai semua kota dikunjungi, ketika semut melakukan perjalanannya, semut akan melakukan *state transition* untuk memilih kota yang akan dituju dan aturan *local updating* untuk memodifikasi jumlah *pheromone* pada sisi yang telah dikunjungi oleh semut. Setelah semua kota dikunjungi oleh semut, maka digunakan aturan *global updating* untuk mengurangi jumlah *pheromone* pada semua sisi dan menambah

jumlah *pheromone* pada sisi-sisi yang termasuk dalam perjalanan dengan jarak terpendek.

Cara kerja ACS adalah sebagai berikut:

- a) Masing-masing semut mula-mula ditempatkan pada sebuah kota yang diperoleh dari hasil inisialisasi.
- b) Setiap semut melakukan perjalanan dengan menerapkan aturan *state transition* untuk mencari kota yang ditujunya.
- c) Ketika semut sudah menggunakan aturan *state transition*, ia langsung menerapkan aturan *local updating* untuk mengurangi jumlah *pheromone* pada sisi yang telah dikunjunginya.
- d) Penggunaan aturan *state transition* dan *local updating* ini terus dilakukan berulang-ulang sampai semua semut mengunjungi semua kota yang ada pada permasalahan TSP.
- e) Setelah semua semut menyelesaikan perjalanannya maka dilakukan perhitungan panjang perjalanan untuk semua semut dan dilakukan pencarian semut memiliki panjang perjalanan terpendek, kemudian jumlah *pheromone* pada semua sisi dimodifikasi lagi dengan menerapkan aturan *global updating*.
- f) Langkah-langkah tersebut dilakukan berulang kali sampai jumlah iterasi yang diinginkan oleh pengguna program [18].

2.2.2.1. Aturan State Transition

Aturan *state transition* dalam ACS adalah sebagai berikut: sebuah semut yang berada di kota i memilih untuk pergi ke kota s dengan menerapkan persamaan berikut:

$$s = \begin{cases} \arg \max_{i \in J_k(i)} \{ \tau(i, l) \cdot [\eta(i, l)]^\beta \} & \text{jika } q \leq q_0 \text{ (exploitasi)} \\ \text{Pilih menurut persamaan berikutnya} & \text{jika } q \geq q_0 \text{ (eksplorasi)} \end{cases}$$

Persamaan 4.1

dimana :

q_0 = Parameter yang menunjukkan hubungan antara eksploitasi dan eksplorasi ($0 < q_0 < 1$)

q = Parameter bernilai acak antara 0 sampai 1 ($0 < q < 1$)

β = Parameter yang mengatur bobot intensitas relative jejak *pheromone*. Jika $\beta = 0$ jarak antar kota tidak berpengaruh

$\tau(i, l)$ = Jumlah *pheromone* pada sisi yang menghubungkan kota i dengan kota l

$\eta(i, l)$ = Panjang sisi dari kota i ke kota l

$J_k(i)$ = Daftar sekumpulan kota yang masih harus dikunjungi oleh semut k ketika semut masih di kota i , lengkap dengan daftar semua kota yang akan dikunjungi, $J_k(i)$ disebut juga sebagai tempat penyimpanan kota-kota yang akan dikunjungi semut

j = kota yang berada dalam $J_k(i)$

Eksploitasi: semut mengeksploitasi kota yang telah dikunjungi sebelumnya

Eksplorasi: semut mengeksplorasi kota yang belum pernah dikunjungi sebelumnya

Jika $q \leq q_0$ (*exploitasi*) yaitu semut akan memilih jalur yang terbaik menurut nilai *pheromone* dan dari heuristic.

Jika $q \geq q_0$ (*explorasi*) yaitu semut akan memilih kota berikutnya menurut persamaan probabilitas berikut ini:

$$Pk(i, j) = \begin{cases} \frac{[\tau(i, j)] \cdot [\eta(i, j)]^\beta}{\sum_{j \in Jk(i)} [\tau(i, l)] \cdot [\eta(i, l)]^\beta} & \text{jika } J \in Jk(i) \\ 0 & \text{jika } J \notin Jk(i) \end{cases}$$

Persamaan 4.2

dimana:

$Pk(i, j)$ = Probabilitas semut k memilih untuk berpindah dari kota i ke kota j

$\tau(i, j)$ = Jumlah *pheromone* pada sisi dari kota i ke kota j

$\eta(i, j)$ = Panjang sisi dari kota i ke kota j

β = Parameter yang mengatur bobot intensitas relative jejak *pheromone*. Jika $\beta = 0$ jarak antar kota tidak berpengaruh

$\tau(i, l)$ = Jumlah *pheromone* pada sisi dari kota i ke kota l

$\eta(i, j)$ = Panjang sisi dari kota i ke kota l

2.2.2.2. Aturan Local Updating

Ketika semut sedang mencari solusi dari Travelling Salesman Problem. Semut mengunjungi sisi yang menghubungkan satu kota dengan kota yang lainnya dan langsung memodifikasi jumlah *pheromone* pada sisi yang baru saja dikunjunginya dengan menerapkan persamaan berikut:

$$\tau(i, j)_{\text{baru}} \leftarrow (1 - \rho) \cdot \tau(i, j)_{\text{lama}} + \rho \cdot \tau_0$$

Persamaan 4.3

dimana:

ρ = parameter yang merepresentasikan *evaporation pheromone local* yang bernilai antara 0 sampai 1 ($0 < \rho < 1$)

$$\tau_0 = (\text{jumlah pheromone awal}) = \frac{1}{\text{jarakawal} * \text{jumlahkota}}$$

Parameter ρ akan menentukan besarnya pengurangan *pheromone* pada sisi yang baru saja dikunjungi oleh semut. Semakin besar nilai maka pengurangan jumlah *pheromone* akan semakin banyak, sehingga perjalanan semut akan cepat konvergen pada jalur tertentu.

Dengan adanya *local updating*, maka jumlah *pheromone* pada sisi yang baru saja dikunjungi menjadi berkurang sehingga kemungkinan terpilihnya sisi tersebut oleh semut yang lain pada langkah iterasi yang sama menjadi berkurang. Akibatnya semut tidak pernah konvergen pada sisi tertentu saja, tetapi semut lebih suka memilih sisi-sisi yang belum pernah dikunjungi.

2.2.2.3. Aturan Global Updating

Aturan *global updating* dilakukan setelah semua semut selesai melakukan perjalanannya. Jumlah *pheromone* pada setiap sisi dimodifikasi dengan menerapkan persamaan berikut:

$$\tau(i, j) \leftarrow (1 - \rho) \cdot \tau(i, j) + \rho \cdot \frac{1}{L_{gb}}$$

Persamaan 4.4

dimana:

ρ = parameter yang merepresentasikan *evaporation pheromone global*; yang bernilai antara 0 sampai 1 ($0 < \rho < 1$)

L_{gb} = Jarak tempuh terpendek yang di peroleh sejak awal iterasi sampai dengan iterasi yang sedang berlangsung.

Global updating dimaksudkan mengurangi jumlah *pheromone* pada semua sisi sebesar ρ kemudian dilakukan penambahan jumlah *pheromone* pada sisi yang termasuk dalam perjalanan dengan jarak yang paling pendek.

Parameter ρ pada persamaan diatas untuk menentukan besarnya penguapan *pheromone*. Semakin besar nilai ρ maka penguapan *pheromone* akan semakin banyak. Penguapan *pheromone* yang terlalu banyak akan menyebabkan perjalanan semut menjadi cepat konvergen pada jalur tertentu [18].

BAB III METODOLOGI PENELITIAN

Bab metodologi membahas mengenai alur penyelesaian penelitian. Tujuan dari bab ini adalah agar penyusunan dan penyelesaian penelitian dapat terstruktur dan selesai tepat pada waktu yang telah direncanakan.

3.1. Diagram Metodologi

Berikut ini disajikan metodologi pengerjaan tugas akhir Perancangan Sistem Optimasi Rute Pengangkutan Sampah di Kabupaten Sidoarjo menggunakan Algoritma *Ant Colony Optimization* (ACO):

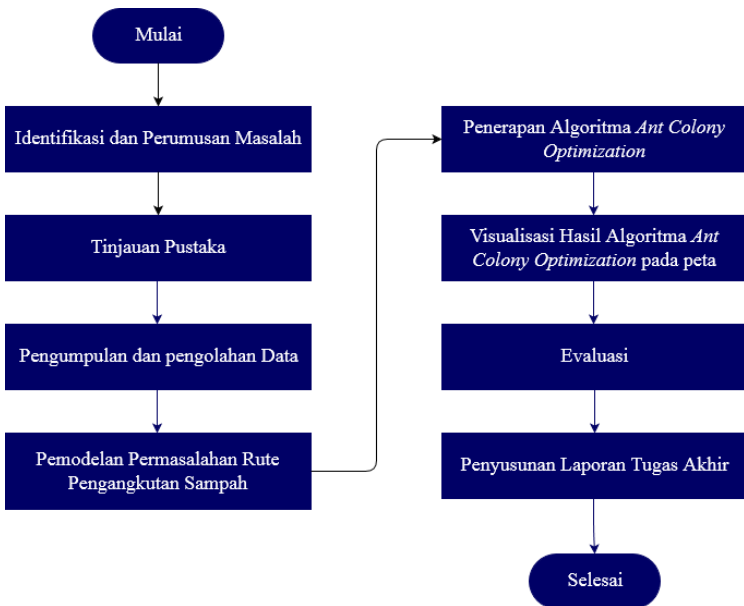


Diagram 3.1 Metodologi Penelitian

3.2. Uraian Metodologi

Berdasarkan pada diagram alur metodologi yang telah dikemukakan pada sub bab sebelumnya, di bawah ini adalah penjelasan pada setiap proses pengerjaan tugas akhir sesuai dengan metodologi yang telah dibuat:

3.2.1. Identifikasi dan Perumusan Masalah

Pada tahapan pertama ini dilakukan identifikasi serta perumusan masalah yang terkait dengan latar belakang, tujuan serta manfaat yang akan didapat dalam penelitian. Tugas akhir ini membahas mengenai permasalahan pengangkutan sampah yang terjadi di Kabupaten Sidoarjo, dimana Kabupaten Sidoarjo memiliki luas wilayah yang lebih besar jika dibandingkan dengan Kota Surabaya. Tahapan ini juga menentukan metode yang akan digunakan untuk menyelesaikan permasalahan yang didapat. Pada tugas akhir ini, pencarian rute pengangkutan sampah yang paling optimum diselesaikan dengan menggunakan Algoritma *Ant Colony Optimization*.

3.2.2. Tinjauan Pustaka

Pada tahapan ini dilakukan studi literatur terkait penelitian yang telah dilakukan oleh peneliti lain dan teori-teori yang dibutuhkan dalam pengerjaan tugas akhir. Tahapan ini dimulai dengan pencarian literatur mengenai permasalahan terkait pengangkutan sampah, yang umumnya dimodelkan sebagai permasalahan rute kendaraan atau *Vehicle Routing Problem* (VRP) dan literatur mengenai metode yang akan digunakan, yaitu Algoritma *Ant Colony Optimization* (ACO) dalam menyelesaikan permasalahan rute kendaraan. Literatur pendukung lainnya terkait kondisi eksisting sistem pengolahan dan pengangkutan sampah.

3.2.3. Pengumpulan dan pengolahan Data

Data merupakan hal penting dalam sebuah penelitian. Data yang dibutuhkan dalam penelitian berupa data masa lalu

sebagai acuan dalam mengerjakan penelitian. Data yang digunakan adalah informasi kondisi eksisting pengolahan dan pengangkutan sampah di Kabupaten Sidoarjo saat ini, data armada dan rute pengangkutan sampah di tahun 2016. Dalam tugas akhir ini, data didapatkan dari Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo.

3.2.4. Pemodelan Permasalahan Rute Pengangkutan Sampah

Pada tahapan ini dilakukan pemodelan dari permasalahan rute pengangkutan sampah, yang terdiri dari:

- Titik-titik koordinat lokasi setiap TPS/TPST di Kabupaten Sidoarjo
- Rute pengangkutan sampah di setiap layanan
- Jarak antar titik TPS/TPST di setiap layanan

3.2.5. Penerapan Algoritma *Ant Colony Optimization* (ACO)

Pada tahapan ini dilakukan penerapan *Ant Colony Optimization* untuk menyelesaikan permasalahan rute pengangkutan sampah yang sudah dimodelkan di tahapan sebelumnya. Titik-titik yang akan dioptimasi pada penelitian ini tidak mempertimbangkan titik-titik TPS yang dilayani setiap armada pengangkutannya, namun berdasarkan keseluruhan titik TPS yang dilayani oleh Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo dan menghasilkan rekomendasi rute perjalanannya yang lebih ringkas dari sebelumnya.

3.2.6. Visualisasi Hasil Algoritma *Ant Colony Optimization* pada Peta

Pada tahapan ini dilakukan visualisasi dari hasil yang telah didapatkan dari tahapan sebelumnya ke dalam bentuk peta. Penghitungan jarak dilakukan sesuai dengan kondisi jalan yang ada di kondisi eksisting dengan ketentuan-ketentuan jalan yang

digunakan sesuai dengan kondisi jalan yang dapat dilalui oleh armada pengangkutan sampah.

3.2.7. Evaluasi

Pada tahapan ini dilakukan evaluasi dari hasil keseluruhan sistem yang telah di buat sebelumnya dengan menganalisa besar optimalisasi yang dihasilkan dari kondisi eksisting yang sudah ada.

3.2.8. Penyusunan Laporan Tugas Akhir

Tahapan akhir adalah penyusunan laporan tugas akhir. Laporan ini bertujuan untuk menjelaskan dengan rinci hasil dari penelitian yang dilakukan pada tugas akhir. Laporan tugas akhir meliputi:

Bab I Pendahuluan

Bab ini menjelaskan tentang latar belakang mengapa tugas akhir ini membahas mengenai permasalahan pengangkutan sampah di Kabupaten Sidoarjo, rumusan masalah dan batasan masalah yang akan dibahas di dalam tugas akhir ini, tujuan dan manfaat dari diselesaikannya tugas akhir mengenai rute optimum pengangkutan sampah dan relevansi dalam pengerjaan tugas akhir mengenai permasalahan rute pengangkutan sampah di Kabupaten Sidoarjo ini.

Bab II Tinjauan Pustaka dan Dasar Teori

Bab ini menjelaskan tentang penelitian-penelitian sebelumnya yang terkait dengan permasalahan penentuan rute optimum pengangkutan sampah, yaitu permasalahan rute kendaraan dan dasar-dasar teori dari metode Algoritma *Ant Colony Optimization* yang digunakan sebagai acuan dalam pengerjaan tugas akhir ini.

Bab III Metodologi Penelitian

Bab ini menjelaskan mengenai langkah-langkah pengerjaan tugas akhir Perancangan Sistem Optimasi Rute Pengangkutan Sampah di Kabupaten Sidoarjo menggunakan Algoritma *Ant Colony Optimization* (ACO) beserta penjelasannya.

Bab IV Rancangan

Bab ini menjelaskan tentang rancangan tugas akhir untuk pembuatan sistem optimasi rute pengangkutan sampah di Kabupaten Sidoarjo meliputi proses pengumpulan dan pengolahan data, identifikasi variabel dan proses pengolahan data.

Bab V Implementasi

Bab ini menjelaskan tentang proses optimasi yang dilakukan, seperti jarak optimum yang didapatkan dan hasil analisa dari optimasi biaya pengangkutan sampah.

Bab VI Evaluasi

Bab ini menjelaskan tentang hasil dan pembahasan yang didapatkan dari proses optimasi yang dilakukan berupa hasil ujicoba dan rekomendasi rute alternatif yang bisa ditawarkan.

Bab VII Kesimpulan dan Saran

Bab ini menjelaskan tentang kesimpulan yang dapat ditarik dari hasil pembahasan serta saran yang dapat ditujukan untuk menyempurnakan tugas akhir dan saran yang ditujukan untuk penelitian selanjutnya.

[halaman ini sengaja dikosongkan]

BAB IV PERANCANGAN

Pada bab ini akan dijelaskan bagaimana rancangan dari penelitian tugas akhir yang meliputi subyek dan obyek dari penelitian, pemilihan subyek dan obyek penelitian dan bagaimana penelitian akan dilakukan.

4.1. Pengumpulan Data

Pada tahapan ini dilakukan pengumpulan data yang dibutuhkan dalam pengerjaan tugas akhir. Proses pengumpulan data dilakukan dengan melakukan permintaan data kepada Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo. Dari proses tersebut didapatkan beberapa data sebagai berikut:

4.1.1. Data Pembagian Armada dan Supir Angkutan Sampah yang Bertugas

Tabel 4.1 merupakan data tabel pembagian armada dan supir angkutan sampah yang bertugas di Dinas Lingkungan Hidup dan Kebersihan (DLKH) Kabupaten Sidoarjo, yang didapatkan dari Laporan Angkutan Sampah Kabupaten Sidoarjo Tahun 2017 oleh DLHK.

Data pada Tabel 4.1 berikut hanya sebagian data yang ditampilkan dari total keseluruhan data yang tercantum pada Lampiran 1.

4.1.2. Potret Pengelolaan Sampah di Kabupaten Sidoarjo

Berdasarkan Laporan 6 Bulanan Dinas Lingkungan Hidup dan Kebersihan Kabupaten Sidoarjo tahun 2017, potret pengelolaan sampah di Kabupaten Sidoarjo di gambarkan dalam diagram

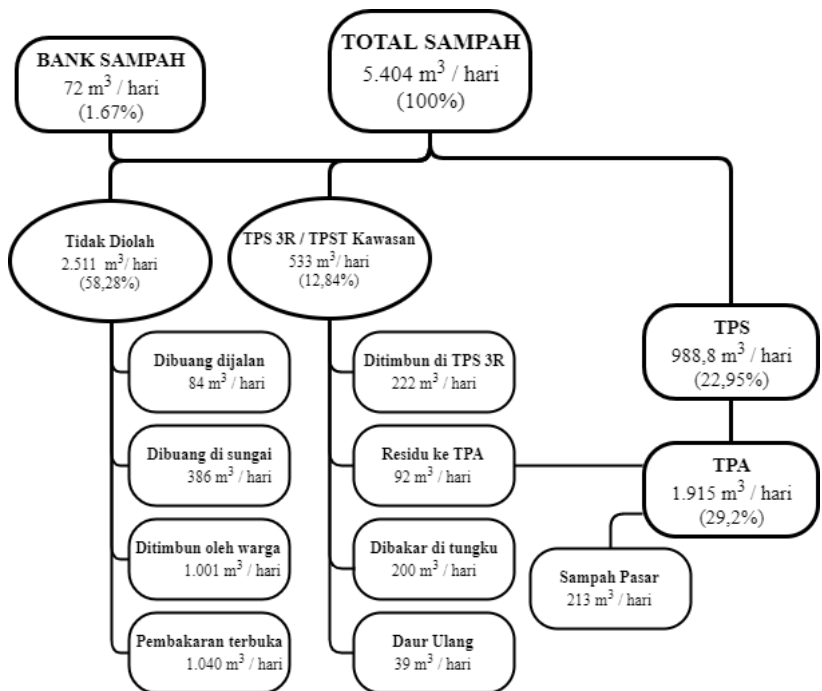


Diagram 4.1 Potret Pengelolaan Sampah Kabupaten Sidoarjo

Adapun proses pengangkutan sampah yang dilakukan oleh Dinas Lingkungan Hidup dan Kebersihan di Kabupaten Sidoarjo adalah dengan menggunakan metode *Stationary Container System* (SCS) dimana truk akan berkeliling menuju

TPS untuk mengangkut sampah ke dalam kontainer sampai penuh, baru kemudian dibawa ke Tempat Pembuangan Akhir yang terletak di Jabon.

4.2. Pengembangan Algoritma ACO pada Program Pencarian Rute Pengangkutan Sampah

Pada penelitian ini akan dikembangkan langkah-langkah penyelesaian untuk permasalahan pencarian rute pengangkutan sampah berdasarkan algoritma ACO:

1. Menentukan parameter yang digunakan, yaitu titik-titik rute tujuan, jarak antar titik, nilai alpha sebagai parameter pengendali intensitas jejak semut, nilai beta sebagai parameter pengendali visibilitas jarak, nilai rho sebagai parameter penguapan *pheromone*.
2. Menerapkan aturan *State Transition* untuk semut menentukan titik mana yang akan dikunjungi pada iterasi yang akan dilakukan.
3. Menetapkan aturan *Local Updating* untuk memodifikasi jumlah *pheromone*
4. Ulangi langkah 1-3 hingga semua semut mendapatkan rute untuk mengunjungi semua titik
5. Hitung total jarak terpendek di setiap iterasi
6. Menetapkan aturan *Global Updating*
7. Menentukan solusi terbaik yang didapatkan

4.3. Proses Perancangan Use Case Diagram Aplikasi

4.3.1. Penentuan Actor, Use Case, dan Relation

Untuk merancang sebuah *use case diagram*, perlu untuk menentukan terlebih dahulu *actor*, *use case* dan *relation* dalam aplikasi.

- *Actor*

Pada aplikasi ini, ditentukan hanya ada satu aktor yaitu *user*. User akan melakukan interaksi secara langsung dengan sistem tanpa ada keterlibatan dari aktor lainnya.



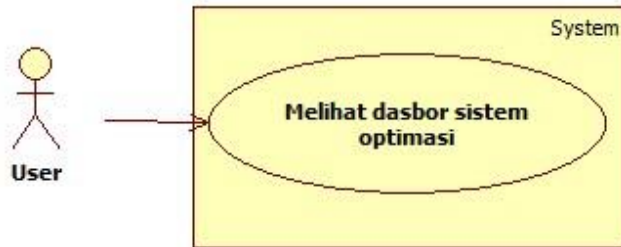
Gambar 4.1 Aktor yang terlibat dalam Use Case adalah User

- *Use Case*
Use Case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling berinteraksi atau bertukar pesan antar unit maupun aktor. *Use Case* digambarkan dalam bentuk oval.



Gambar 4.2 Salah satu contoh Use Case dalam aplikasi

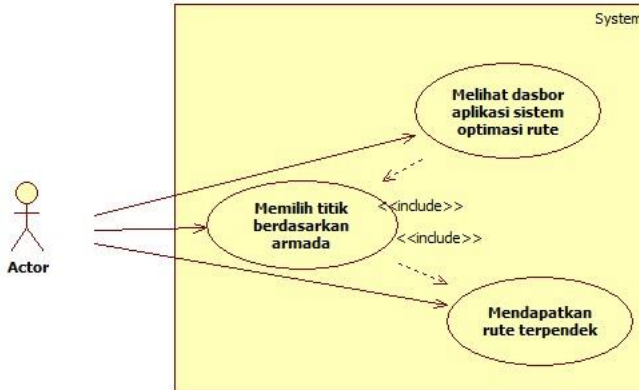
- *Relation*
Relation atau relasi merupakan hubungan yang terjadi pada sistem baik antar aktor maupun antar usecase maupun antara usecase dan aktor. Salah satu bentuk relasi antara *actor* dan *use case* digambarkan pada Gambar



Gambar 4.3 Salah satu bentuk relasi pada Diagram

4.3.2. Pembuatan Use Case Diagram

Setelah menentukan *Actor*, *Use Case* dan *Relation* pada tahap sebelumnya, selanjutnya adalah pembuatan *Use Case Diagram*. Untuk *Use Case Diagram* pada sistem ini digambarkan pada Gambar



Gambar 4.4 Use Case Diagram dari Aplikasi Sistem Optimasi Rute

Berdasarkan gambar tersebut, dapat dilihat bahwa *user* dapat berinteraksi langsung dengan sistem dengan total 3 *use case* utama.

4.3.3. Penjelasan Use Case Diagram

Bagian ini akan menjelaskan lebih detail mengenai masing-masing *use case* yang terdapat dalam *Use Case Diagram* di bagian 4.4.2. penjelasan tersebut dapat dilihat dalam *Use Case Description* di Tabel berikut.

Tabel 4.2 Penjelasan *Use Case* Melihat Dashboard Aplikasi Sistem Optimasi Rute

ID :	1
Use Case :	Melihat Dashboard Aplikasi Sistem Optimasi Rute
Primary Actor	<i>User</i>
Brief Description	Menjelaskan tentang bagaimana <i>User</i> dapat mengakses dashboard aplikasi
Pre Condition	-
Normal Flow of Event	1. <i>User</i> dapat mengakses dan menjalankan fitur yang terdapat pada aplikasi melalui halaman dashboard.
Alternate Flow of Event	-
Exeptional Flow of Event	-
Post Condition	<i>User</i> telah berada pada halaman dashboard sistem

Tabel 4.3 Penjelasan Use Case Memilih Titik Berdasarkan Armada

ID :	2
Use Case :	Memilih Titik Berdasarkan Armada
Primary Actor	<i>User</i>
Brief Description	Menjelaskan tentang bagaimana <i>User</i> dapat memilih titik di suatu armada untuk melihat rute berdasarkan armada yang dimaksud
Pre Condition	<i>User</i> telah berada pada halaman dashboard terlebih dahulu
Normal Flow of Event	<ol style="list-style-type: none"> 1. <i>User</i> telah berada pada halaman dashboard aplikasi 2. <i>User</i> memilih armada yang terdapat pada <i>drop down menu</i> “Armada” yang tersedia 3. <i>Field</i> pada tampilan <i>drop down menu</i> “Armada” akan berubah menjadi armada yang telah dipilih
Alternate Flow of Event	-
Exceptional Flow of Event	<ol style="list-style-type: none"> 1. <i>User</i> telah berada pada halaman dashboard aplikasi 2. <i>User</i> tidak memilih armada yang terdapat pada <i>drop down menu</i> “Armada” yang tersedia 3. Tidak ada perubahan pada <i>field</i> di tampilan <i>drop down menu</i> “Armada”
Post Condition	<i>User</i> dapat memilih armada yang diinginkan

Tabel 4.4 Penjelasan *Use Case* Mendapatkan Rute Terpendek

ID :	3
Use Case :	Mendapatkan Rute Terpendek
Primary Actor	<i>User</i>
Brief Description	Menjelaskan tentang bagaimana <i>User</i> dapat melihat rute terpendek yang dihasilkan
Pre Condition	<ol style="list-style-type: none"> 1. <i>User</i> telah berada dalam dashboard aplikasi 2. <i>User</i> telah memilih salah satu kecamatan yang ingin dilihat rutenya
Normal Flow of Event	<ol style="list-style-type: none"> 1. <i>User</i> memilih tombol “Lihat Rute” 2. Sistem akan menampilkan peta yang menggambarkan rute terpendek yang dihasilkan dan rincian detail jarak yang dihasilkan.
Alternate Flow of Event	-
Exceptional Flow of Event	<ol style="list-style-type: none"> 1. <i>User</i> memilih tombol “Lihat Rute” 2. Sistem tidak berhasil menampilkan peta rute terpendek, ditandai dengan tidak munculnya peta maupun rincian detail jarak.
Post Condition	<i>User</i> dapat melihat rekomendasi rute terpendek yang dihasilkan

[halaman ini sengaja dikosongkan]

BAB V

IMPLEMENTASI

Bab ini menjelaskan tentang proses implementasi model ke dalam sistem untuk mendapatkan hasil optimasi terbaik.

5.1. Lingkungan Uji Coba

Dalam tugas akhir ini, implementasi pemodelan optimasi dilakukan menggunakan PC berspesifikasi *processor* Intel Core i5-3317U dengan RAM 4GB.

5.2. Persiapan Pengembangan Program

Dalam proses implementasi ini, pembuatan program pencarian rute menggunakan bahasa pemrograman Python 3. Pembuatan program menggunakan bantuan *tools Pants* yang merupakan *tools* bawaan dari Python3 untuk menyelesaikan permasalahan TSP dengan mengimplementasikan Algoritma *Ant Colony Optimization*.

Dalam pengembangan program, dengan menggunakan *tools Pants* Python3 dibutuhkan beberapa *package* sebagai berikut:

- **Pants**
Package Pants digunakan untuk menjalankan fungsi-fungsi yang dibutuhkan dalam *tools Pants*.
- **Math**
Package Math digunakan untuk menjalankan fungsi perhitungan matematis yang terdapat dalam pengembangan aplikasi
- **Timeit**
Package Timeit digunakan untuk menjalankan fungsi perhitungan waktu. Pada pengembangan aplikasi, *package* ini digunakan untuk menghitung waktu running aplikasi.

Setelah seluruh *package* yang dibutuhkan sudah tersedia di dalam *project folder*, maka selanjutnya program akan siap untuk dikembangkan.

5.3. Implementasi ACO pada Program

Pada tahap ini akan dijelaskan pengembangan program menggunakan algoritma ACO dalam beberapa fase sebagai berikut:

5.3.1. Inisialisasi Package ke dalam Project Program

Seperti yang sudah dijelaskan di bagian 4.2, terdapat 3 *package* yang digunakan yaitu *Pants*, *Math* dan *Timeit* dengan inisialisasi di dalam program yang dapat dilihat dalam kode program 5.1.

```
import pants
import math
import timeit
```

Kode Program 5.1 Inisialisasi package dalam program

5.3.2. Input Data

Dalam *tools* *Pants* ini, terdapat beberapa cara untuk memasukkan data pencarian jarak berupa titik / node. Dalam tugas akhir ini, program mendapatkan input berupa titik koordinat yang terdiri dari *longitude* dan *latitude*. Inputan data yang dilakukan pada program dapat dilihat dalam kode program 5.2.

```
buduran_nodes = [(-7.427615, 112.711109), (-7.420040, 112.717253), (-7.432599, 112.7254143), (-7.432267, 112.728402), (-7.431704, 112.726543), (-7.428407, 112.725056), (-7.4015367, 112.7268687), (-7.4307691, 112.7228216), (-7.415761, 112.724974), (-7.432447, 112.720377)]
```

Kode Program 5.2 Input Data Titik Koordinat

5.3.3. Inisiasi Jarak Antar Titik

Untuk bisa menemukan rute terpendek menggunakan ACO, dibutuhkan informasi mengenai jarak antar titik yang akan dicari. Dalam program, pencarian jarak antar dua titik dilakukan dengan menginputkan data jarak antar titik secara manual yang dengan menggunakan fitur *dictionary* yang dapat dilihat dalam kode program 5.3:

```
def jarak(a,b):
    a0 = str(a[0])
    a1 = str(a[1])
    b0 = str(b[0])
    b1 = str(b[1])
    kordinat = a0 + ', ' + a1 + ', ' + b0 + ', '
    + b1
    node = next((item for item in
dictionary.dict_jarak if item["titik"] ==
kordinat))

    return node["jarak"]
```

Kode Program 5.3 Rumus Penghitungan Jarak Antar Dua Titik

Berdasarkan kode program 5.3, fitur *dictionary* didefinisikan dalam variable **jarak**. Adapun inputan jarak antar titik yang akan dicari dimasukkan ke dalam file kode program terpisah yang terdapat dalam file *dictionary* dengan nama **dict_jarak**. Kode program 5.4 merupakan contoh potongan inputan jarak pada *dictionary* yang dimaksud.

```
dict_jarak = [
    # Armada1
    {"titik": "-7.427615, 112.711109, -7.42004,
112.717253", "jarak": 3200}, #v01-v02
    {"titik": "-7.42004, 112.717253, -7.427615,
112.711109", "jarak": 2900}, #v02-v01
    {"titik": "-7.427615, 112.711109, -7.432599,
112.7254143", "jarak": 2700}, #v01-v03
    {"titik": "-7.432599, 112.7254143, -7.427615,
112.711109", "jarak": 2700}, #v03-v01
    {"titik": "-7.427615, 112.711109, -7.432267,
112.728402", "jarak": 2900}, #v01-v04
    {"titik": "-7.432267, 112.728402, -7.427615,
```

```

112.711109", "jarak": 2900},           #v04-v01
  {"titik": "-7.427615, 112.711109, -7.431704,
112.726543", "jarak": 2700},         #v01-v05
  {"titik": "-7.431704, 112.726543, -7.427615,
112.711109", "jarak": 2700},         #v05-v01

```

Kode Program 5.4 Fungsi Penghitungan Jarak Antar Titik

Format input data yang terdapat pada kode program 5.4 diatas terdiri dari input titik yang dibaca dalam format *string* dengan nama variabel “titik”, dan input panjang jarak dalam satuan meter dengan nama variabel “jarak”.

Adapun titik-titik yang dipilih dengan panjang jarak antar titik akan menjadi inputan dalam program pencarian rute yang dimasukkan dalam variable world seperti pada contoh kode program 5.5.

```

world = pants.World(sidoarjo_nodes, jarak)

```

Kode Program 5.5 Fungsi Input Program Pencarian Rute

5.3.4. Pencarian Solusi Rute Terpendek

Tools Pants menyediakan fungsi untuk mencari solusi rute terpendek dengan menggunakan algoritma ACO, yaitu fungsi Solver. Kode pada program dapat dilihat pada kode program 5.5.

```

solver =
pants.Solver(alpha=0.1,beta=2,rho=0.8,q=1)
solution = solver.solve(world)

```

Kode Program 5.6 Fungsi Pencarian Rute

Dalam atribut “solver”, user bisa mengubah parameter yang dibutuhkan sesuai keinginan. Parameter yang digunakan dapat mempengaruhi keluaran yang dihasilkan.

Sedangkan atribut “solution” berfungsi untuk menghitung rute terpendek menggunakan ACO.

5.3.5. Menampilkan Hasil

Untuk menampilkan hasil rute terpendek yang telah dilakukan di bagian sebelumnya, gunakan fungsi “Print”. Terdapat dua keluaran yang dihasilkan, diantaranya adalah total jarak yang ditunjukkan dalam fungsi *solution.distance* dan rekomendasi urutan rute dalam fungsi *solution.tour*. fungsi keluaran dapat dilihat dalam kode program 5.6 berikut.

```
print(solution.distance)
print(solution.tour)
```

Kode Program 5.7 Output Rekomendasi Rute Terpendek

5.3.6. Fungsi Penghitungan *Running Time*

Program ini menggunakan fungsi dari *package* *timeit* untuk menghitung lama *running time* setiap program dijalankan.

Waktu dimulainya penghitungan *running time* program dimulai sebelum program di jalankan, yaitu setelah dilakukan inisiasi *package*. Pada program ini, waktu dimulainya program ditunjukkan dalam atribut “start” yang dapat dilihat dalam kode program 5.7 berikut

```
import timeit
start = timeit.default_timer()
```

Kode Program 5.8 Fungsi Memulai *Running Time* Program

Setelah keseluruhan fungsi inti pada program telah dijalankan, maka diberikan fungsi untuk memberhentikan *running time* pada program. Waktu berhenti ditunjukkan dalam atribut “Stop”. Hasil dari besar *running time* ditunjukkan dalam fungsi `print(stop-start)` yang merupakan hasil pengurangan waktu selesai dengan waktu dimulai, yang dapat dilihat dalam kode program 5.8 berikut

```
stop = timeit.default_timer()
print(stop-start)
```

Kode Program 5.9 Fungsi Menghitung Lama *Running Time*

5.3.7. Contoh Hasil Keluaran

Berikut adalah contoh keluaran yang dihasilkan setelah program dijalankan

```
C:\Users\Kirana\AppData\Local\Programs\Python\Python36\python.exe
C:/Users/Kirana/PycharmProjects/AC01/coba09_jarakwithdict
9800
[(-7.4308769, 112.7217183), (-7.424268, 112.704583), (-7.439658, 112.703718),
(-7.4490591, 112.7092392)]
0.04139720549241967
```

Gambar 5.1 Contoh keluaran yang dihasilkan

5.4. Pengembangan Antar Muka Pencarian Rute Pengangkutan Sampah

Dalam tahapan ini, dijelaskan langkah-langkah dalam memvisualisasikan hasil yang didapatkan dalam langkah 5.3 ke dalam tampilan peta yang berada di dalam sistem optimasi rute distribusi pengangkutan sampah. Pada tugas akhir ini, visualisasi peta di tampilkan dalam aplikasi berbasis web.

Dalam proses implementasi, pembuatan aplikasi menggunakan bahasa pemrograman *HTML5*, *CSS3*, *Javascript* dan *PHP* dengan bantuan *tools* XAMPP, Notepad++, dan Google Chrome sebagai *tools* pendukung agar mempermudah proses pengerjaan aplikasi.

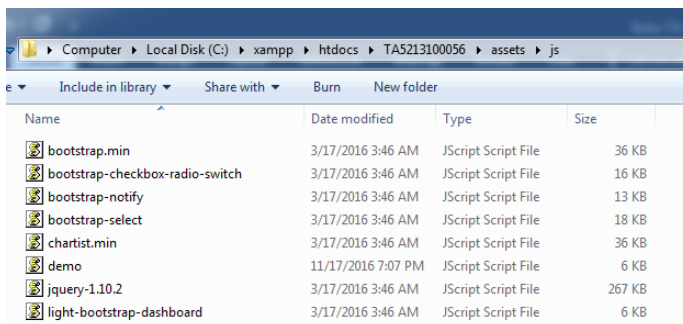
Adapun langkah-langkah pengembangan antar muka aplikasi pencarian rute pengangkutan sampah ini dijelaskan dalam beberapa fase sebagai berikut:

5.4.1. Inisialisasi *library* ke dalam project folder

Dalam pembuatan aplikasi, penelitian ini menggunakan beberapa *library* dengan tujuan mempermudah proses pembuatan aplikasi. Penelitian ini menggunakan *library* diantaranya adalah *jQuery (offline)* untuk mempermudah penggunaan kode *Javascript*, *Bootstrap (offline)* untuk

mempermudah pembuatan desain tampilan website, dan *Google Maps API* untuk menampilkan rute yang dihasilkan dalam bentuk peta.

Pada pengerjaan tugas akhir ini, *project* ditempatkan dalam folder yang bernama “TA5213100056” yang berlokasi di drive *C:\xampp\htdocs\TA5213100056*. *Library* yang digunakan dalam *project* ini ditempatkan dalam drive *C:\xampp\htdocs\TA5213100056\assets\js* seperti gambar 5.1 berikut :



Gambar 5.2 Inisialisasi *library* ke dalam *project folder*

Library yang sudah tersedia di dalam *project folder* kemudian dihubungkan dengan aplikasi dengan cara dipanggil dalam kode program aplikasi. Penulisan kode program yang memanggil *library* tersebut dapat dilihat dalam kode program 5.9 berikut:

```

<script
src="https://maps.googleapis.com/maps/api/js?key=
AIzaSyBHbWWn7TmJkSWdSXZffZu-
_qbsw7tRLDk&libraries=places&callback=initMap"
  async defer></script>

  <!-- Core JS Files -->
  <script src="assets/js/jquery-1.10.2.js"
type="text/javascript"></script>
  <script src="assets/js/bootstrap.min.js"
type="text/javascript"></script>

  <!-- Checkbox, Radio & Switch Plugins -->
  <script src="assets/js/bootstrap-checkbox-
radio-switch.js"></script>

  <!-- Charts Plugin -->
  <script
src="assets/js/chartist.min.js"></script>

  <!-- Notifications Plugin -->
  <script src="assets/js/bootstrap-
notify.js"></script>

  <!-- Google Maps Plugin -->
  <script type="text/javascript"
src="https://maps.googleapis.com/maps/api/js?sens
or=false"></script>

  <!-- Light Bootstrap Table Core javascript
and methods for Demo purpose -->
  <script src="assets/js/light-bootstrap-
dashboard.js"></script>

  <!-- Light Bootstrap Table DEMO methods,
don't include it in your project! -->
  <script src="assets/js/demo.js"></script>

```

Kode Program 5.10 Inisialisasi *library* ke dalam kode program

5.4.2. Input Data Titik TPS Sidoarjo

Tahapan ini menjelaskan bagaimana data berupa titik koordinat TPS yang berada di Kabupaten Sidoarjo yang digunakan dalam penelitian ini dimasukkan ke dalam kode program. Data yang diinputkan inilah yang kemudian digunakan sebagai masukan dalam pencarian optimasi rute oleh program *python* yang terdapat di bagian 5.3.

Karena data titik koordinat yang digunakan bersifat statis, data dimasukkan ke dalam kode program php dalam bentuk *JSON* sebagai variabel “locations”. Data di dalam *JSON* ini diinisiasikan dalam nama unik sesuai nama masing-masing wilayah yang nantinya dipanggil oleh kode program *python*. Kode program 5.10 merupakan salah satu contoh data *JSON* yang diinputkan dalam kode program php.

```
var locations = [
    locations['buduran'] = [
        ['TPS Sidokerto', -7.427615, 112.711109,
1],
        ['TPS Sukorejo', -7.420040, 112.717253,
2],
        ['TPST Siwalanpanji', -7.432599,
112.7254143, 3],
        ['TPS SDN Siwalanpanji', -7.432267,
112.728402, 4],
        ['SMA Antartika', -7.431704, 112.726543,
5],
        ['TPS Jl. KH. Khamdani', -7.428407,
112.725056, 6],
        ['TPS Bursa Kupang', -7.4015367,
112.7268687, 7],
        ['TPS Jl. Antartika', -7.4307691,
112.7228216, 8],
        ['TPS Jl. Raya Buduran', -7.415761,
112.724974, 9],
        ['TPS Sono (Sidokerto)', -7.432447,
112.720377, 10],
```

```

11],
    ['TPS Ps Krempyeng Sugihwaras', -
7.477367, 112.701596, 7],
    ['TPS Jl Raya Bligo', -7.475349,
112.71509, 8],
    ['TPS Kir Candi', -7.4826914,
112.7105095, 9],
    ['TPS Jl. Karang Tanjung', -7.487192,
112.690701, 10],
    ['TPS Bligo', -7.474292, 112.717225, 11],
    ['TPS Desa Klurak', -7.479375,
112.725452, 12],
    ['TPS Jembatan Candi', -7.494304,
112.732967, 13],
    ['TPS Pondok Millenium', -7.469346,
112.706768, 14],
    ['TPS Perum Sidodadi', -7.457085,
112.6791, 15],
    ['TPS Desa Larangan', -7.46912,
112.720876, 16]
];

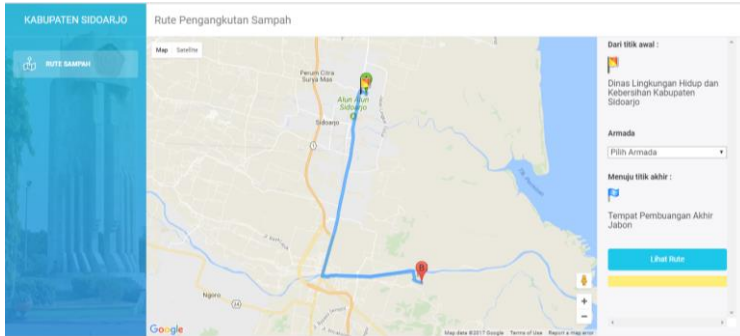
```

Kode Program 5.11 Input data dalam file php

5.4.3. Desain Antar Muka dan Fitur Aplikasi

Pada tahap ini, dilakukan pengembangan antar muka aplikasi yang bertujuan untuk mempermudah *user* dalam mendapatkan informasi rute yang direkomendasikan dalam penelitian ini. Adapun aplikasi sistem optimasi rute distribusi pengangkutan sampah ini terdiri dari beberapa fitur sebagai berikut:

- **Halaman Dasbor Utama**
 Aplikasi Sistem Optimasi Rute Pengangkutan Sampah pada penelitian ini bersifat *Open* sehingga untuk bisa menggunakan fitur yang ada *user* tidak membutuhkan akses khusus seperti login. Halaman dasbor utama pada Aplikasi dapat dilihat di gambar 5.3



Gambar 5.3 Halaman Dasbor Aplikasi

Dalam tampilan dasbor, terdapat tiga bagian utama. Bagian di sisi kiri berisi fitur untuk memilih menu apakah user ingin mencari rute pengangkutan sampah, atau melihat titik daftar TPS yang ada di Sidoarjo. Bagian di sisi tengah merupakan tampilan peta yang menjadi media visualisasi rute ataupun titik TPS sesuai fungsi yang dipilih. Sedangkan bagian di sisi kanan berisi fitur untuk memilih kecamatan/wilayah/armada yang bertugas yang nantinya akan divisualisasikan rute/titik TPS nya. Tampilan peta merupakan fitur utama yang terdapat dalam dasbor aplikasi. Kode program 5.11 merupakan tampilan kode program dari fungsi menampilkan peta dari *Google Maps*.

```

var myLatLng = {lat: -7.4540353, lng:
112.6316769};
    var map = new
google.maps.Map(document.getElementById('map'), {
    zoom: 14,
    center: myLatLng
});

    var image =
'https://developers.google.com/maps/documentation
/javascript/examples/full/images/beachflag.png';
    var beachMarker = new
google.maps.Marker({
    position: {lat: -7.4325987, lng:
112.7254143},
    map: map,
    icon: image
});

    var image = 'assets/img/flag2.png';
    var beachMarker = new
google.maps.Marker({
    position: {lat: -7.5489852, lng:
112.76136},
    map: map,
    icon: image
});

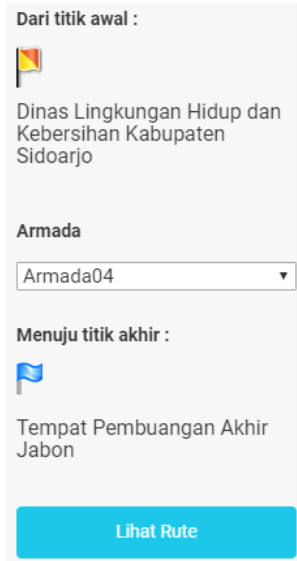
directionsDisplay.setMap(map);

```

Kode Program 5.12 Fungsi menampilkan peta google maps

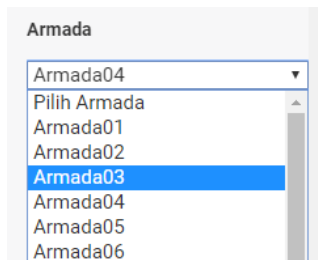
- **Fitur Memilih Kecamatan**

Dalam aplikasi ini, titik-titik TPS sudah terbagi berdasarkan wilayah kecamatan. Sehingga, untuk bisa melihat rekomendasi rute, *user* terlebih dahulu harus memilih salah satu wilayah yang tersedia. Tampilan fitur untuk memilih Armada dapat dilihat dalam gambar 5.4.



Gambar 5.4 Fitur memilih armada

Daftar pilihan wilayah armada berada dalam tampilan *drop-down* menu seperti pada gambar 5.5.



Gambar 5.5 Fitur *drop-down* menu

Tampilan kode program 5.12 dibawah merupakan kode program kerangka *HTML* dari fitur pemilihan armada.

```

<div id="content-panel">
  <div id="map"></div>
  <div id="right-panel">
    <div id="right-content">
      <form action="" method="post">

        <div class="form-group">
          <label>Dari titik awal :</label>
          <label><h5>Dinas Lingkungan Hidup dan Kebersihan
Kabupaten Sidoarjo</h5></label>
          </div>

          <div class="form-group">
            <label>Armada</label>
            <select id="namaarmada"
name="namaarmada">
              <option value="">Pilih Armada</option>
              <option value="armada01"<?php echo
$active == 'armada01' ? ' selected' : ''
?>>Armada01</option>
              <option value="armada02"<?php echo
$active == 'armada02' ? ' selected' : ''
?>>Armada02</option>
              <option value="armada03"<?php echo
$active == 'armada03' ? ' selected' : ''
?>>Armada03</option>
              <option value="armada04"<?php echo
$active == 'armada04' ? ' selected' : ''
?>>Armada04</option>
            </select>
          </div>

          <div class="form-group">
            <label>Menuju titik akhir :</label>
            <label><h5>Tempat Pembuangan
Akhir Jabon</h5></label>
          </div>

          <input type="hidden" name="titik">
          <button id="submit" type="submit"
style="width:100%;" class="btn btn-info btn-
fill">Lihat Rute</button>

```

```

        </form>

        <div id="directions-panel"></div>
    </div>
</div>

```

Kode Program 5.13 kode program kerangka memilih kecamatan

Nama kecamatan yang dipilih kemudian akan terhubung dengan data titik koordinat yang telah dijelaskan di bagian 5.4.2 dengan menggunakan fungsi dari tipe tombol yang digunakan yaitu “submit”. Tampilan kode program dari fungsi menghubungkan nama kecamatan yang dipilih dengan data yang dimaksud terdapat pada kode program 5.13.

```

$( "#submit" ).click( function() {
    var value = $( 'select' ).val();
    $( 'input[name="titik"]' ).val(
    JSON.stringify( locations[value] ) );
} );

```

Kode Program 5.14 Kode program fungsi submit

Proses pencarian rute akan berjalan jika *user* menekan tombol “Lihat Rute” yang tepat berada dibawah *field drop-down menu* pemilihan nama kecamatan. Kode program 5.14 berikut adalah tampilan kode program *HTML* dari kerangka tombol yang digunakan pada aplikasi.

```

<input type="hidden" name="titik">
<button id="submit" type="submit"
style="width:100%;" class="btn btn-info btn-
fill">Lihat Rute</button>

```

Kode Program 5.15 Kode program kerangka tombol Lihat Rute

- Fungsi Pencarian Rute

Fungsi pencarian rute akan berjalan ketika *user* melakukan aksi *POST* setelah menekan tombol *Lihat Rute*.

Seperti yang sudah dijelaskan di bagian 3.5, fungsi pencarian rute berjalan dengan menggunakan kode program *python*. Sedangkan data yang digunakan dan fungsi untuk memasukkan data berada dalam kode program *PHP*. Oleh karena itu, dibutuhkan penghubung antara kode program *PHP* dengan kode program *python* yang bertugas mencari rute.

Kode program 5.15 dibawah adalah tampilan kode program dari inialisasi yang harus dilakukan sebelum menghubungkan kode program *PHP* dengan *Python*. Program menginisialisasi *installer* dari *Python* dalam variabel *\$python* dan kode program *python* yang akan dijalankan dalam variabel *\$path*.

```

if ( $_POST ) {

    $active = $_POST['namaarmada'];
    $python =
'C:\Users\Kirana\AppData\Local\Programs\Python\Py
thon36\python.exe';
    $path =
'C:\Users\Kirana\PycharmProjects\AC01\coba05.py';
    $args = '';

```

Kode Program 5.16 Koneksi php dengan python

Selain itu, *Python* membutuhkan data dalam format *array python* untuk menjalankan fungsi pencarian rute. Supaya data *JSON* yang terdapat dalam kode program *PHP* bisa dibaca oleh file *python*, dibutuhkan modifikasi dalam pembacaan data sebelum kode program *PHP* terhubung dengan kode program *Python*. Kode program 5.16 berikut menunjukkan kode program yang mengubah data *JSON* dalam php ke dalam format *array python* yang

sesuai sehingga bisa dijalankan di dalam fungsi pencarian rute.

```
$titik = json_decode( $_POST['titik'] );
if ( is_array( $titik ) ) foreach ( $titik as
$value) {
    $array = array( $value[1], $value[2] );
    $campur = implode( ',', $array );
    $apaja[] = '' . $campur . '';
}
}
```

Kode Program 5.17 Ubah format data JSON

Kode program 5.17 berikut adalah tampilan kode program dari fungsi menghubungkan kode program *PHP* dengan *Python*.

```
$args = implode( ' ', $apaja );
$cmd = "$path $python $args";
$py = exec( $cmd );
```

Kode Program 5.18 Membaca kode program *python* di *php*

- Fungsi Membaca Data Inputan *PHP* pada Kode Program *Python*

Beralih pada kode program *Python*, disini *Python* juga membutuhkan modifikasi untuk membaca data yang diinputkan dari kode program *PHP*. Modifikasi dilakukan di dalam variabel *nodes* sebagai masukan dalam fungsi pencarian solusi rute terpendek seperti yang telah dijelaskan di bagian 5.3.4.

Kode program 5.18 berikut adalah tampilan kode program dari modifikasi pada variabel *nodes*.

```

nodes = []
isinya = sys.argv
berapax = len( sys.argv )

for i in range(0, berapax):
    pisahin = isinya[i].split(",")

    if len(pisahin) < 2:
        continue
    x = float(pisahin[0])
    y = float(pisahin[1])
    nodes.append((x, y))

```

Kode Program 5.19 Membaca *node* yang diinputkan dari PHP

- Fungsi Menampilkan Rute Dalam Peta

Hasil rekomendasi rute terpendek yang dikeluarkan oleh *Python* kemudian akan kembali dibaca oleh kode program *PHP*. Kode program 5.19 berikut merupakan kode program dari fungsi modifikasi data hasil keluaran *Python* menjadi *array PHP*.

```

preg_match_all('#\((.*?)\)#', $py, $match);

$isip = [];
foreach ( $match[1] as $value ) {
    $pecah = explode( ',', $value );
    $isip[] = $pecah;
}

} else $isip = [];

```

Kode Program 5.20 Menampilkan rute dalam peta

Sama seperti data yang diinputkan di bagian 5.4.2 sebelumnya, *google maps* membutuhkan data dengan format *JSON* untuk bisa dibaca dalam peta. Kode program 5.20 berikut merupakan kode program dari fungsi merubah data *array PHP* menjadi *JSON*.

```

var locationt = <?= json_encode( $isip ) ?>;

```

Kode Program 5.21 Ubah *array* menjadi format *JSON*

Kemudian, data dimasukkan ke dalam fungsi untuk dibaca dalam peta, dan kemudian digambarkan visualisasi rute sesuai dengan titik-titik yang dimasukkan. Fungsi tersebut dapat dilihat dalam kode program 5.21 berikut.

```
var directionsDisplay = new
google.maps.DirectionsRenderer;
    var directionsService = new
google.maps.DirectionsService;

calculateAndDisplayRoute(directionsService,
directionsDisplay);
```

Kode Program 5.22 Visualisasi Rute

```
function
calculateAndDisplayRoute(directionsService,
directionsDisplay) {
    var waypts = [];
    var checkboxArray =
document.getElementById('waypoints');
    for (var i = 0; i < locationt.length;
i++) {
        waypts.push({
            location: new google.maps.LatLng(
locationt[i][0], locationt[i][1] ),
            stopover: true
        });
    }

    directionsService.route({
        origin: new google.maps.LatLng(-
7.4325987,112.7254143),
        destination: new google.maps.LatLng(-
7.5489852,112.76136),
        waypoints: waypts,
        optimizeWaypoints: false,
```

```

        travelMode: 'DRIVING'
    }, function(response, status) {
        if (status === 'OK') {

directionsDisplay.setDirections(response);
            var route = response.routes[0];
            var summaryPanel =
document.getElementById('directions-panel');
            summaryPanel.innerHTML = '';

                // For each route, display summary
information.
                    for (var i = 0; i <
route.legs.length; i++) {
                        var routeSegment = i + 1;
                        summaryPanel.innerHTML += '<b>Route
Segment: ' + routeSegment +
                            '</b><br>';
                            summaryPanel.innerHTML +=
route.legs[i].start_address + ' to ';
                            summaryPanel.innerHTML +=
route.legs[i].end_address + '<br>';
                            summaryPanel.innerHTML +=
route.legs[i].distance.text + '<br><br>';
                        }
                    } else {
                        window.alert('Directions request
failed due to ' + status);
                    }
                });
        }
    }
}

```

Kode Program 5.23 Visualisasi Rute

5.5. Penjelasan Alur Kerja Aplikasi Sistem Optimasi

Tahap ini berisi penjelasan alur kerja dari aplikasi Sistem Optimasi Rute Pengangkutan Sampah yang telah dibuat di bagian sebelumnya. Aplikasi Sistem Optimasi Rute yang dikembangkan pada penelitian ini berbasis website dengan menggunakan bahasa pemrograman *HTML*, *PHP*, *CSS* dan *Javascript*.

Alur dan cara kerja dari aplikasi dapat dilihat pada diagram 5.1 berikut:

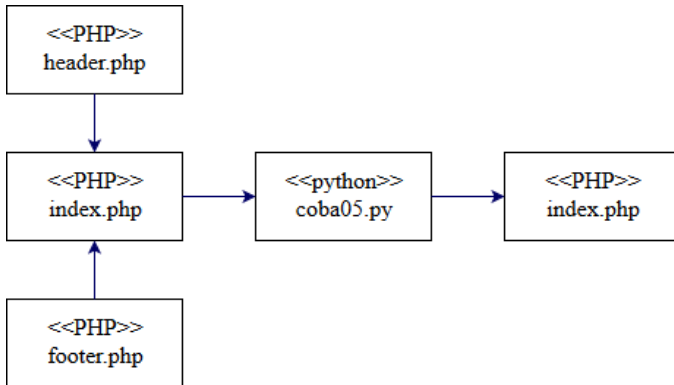


Diagram 5.1 Diagram aplikasi sistem

Aplikasi terbagi ke dalam dua kode program utama, yaitu:

1. Bahasa Pemrograman *PHP*, *HTML*, *CSS* dan *Javascript*
 - a) Header.php
Berisi kode program dari desain kerangka website bagian *header* dan *form* di sisi kiri.
 - b) Footer.php
Berisi kode program dari inialisasi *library* yang digunakan dalam aplikasi dan desain kerangka website bagian *form* di sisi kanan.
 - c) Index.php
Berisi kode program dari fungsi utama yang terdapat dalam aplikasi website seperti yang telah digambarkan di bagian sebelumnya.
2. Bahasa Pemrograman *Python*
 - a) Coba05.py
Berisi kode program *python*

[halaman ini sengaja dikosongkan]

BAB VI

HASIL DAN PEMBAHASAN

Pada bagian ini akan dijelaskan hasil dari perancangan dan implementasi yang telah disusun dalam tugas akhir ini. Bagian ini akan menjelaskan hasil dan pembahasan terkait optimasi dan visualisasi rute pengangkutan sampah di Kabupaten Sidoarjo dengan menggunakan Algoritma *Ant Colony Optimization* (ACO).

6.1. Lingkungan Uji Coba

Lingkungan uji coba merupakan perangkat yang digunakan dalam menjalankan program untuk menyelesaikan permasalahan dan penelitian ini. Lingkungan uji coba terdiri dari perangkat keras dan perangkat lunak. Adapun tabel 6.1 dibawah menampilkan informasi perangkat keras yang digunakan.

Tabel 6.1 Perangkat Keras Pengujian

Perangkat Keras	Spesifikasi
Jenis	Mobile Computer / Laptop
Processor	Intel Core i5-33170
RAM	4GB

Sedangkan tabel 6.2 berikut menampilkan informasi perangkat lunak yang digunakan dalam penelitian ini.

Tabel 6.2 Perangkat Lunak Pengujian

Perangkat Lunak	Fungsi
Windows 10	Sistem operasi
Microsoft Excel 2016	Merangkum data
Jetbrains PyCharm Community Edition 2016.1	Mengolah kode program <i>python</i>
Notepad++	Mengolah kode program aplikasi berbasis website

XAMPP Control Panel	Server yang digunakan untuk melakukan uji coba aplikasi
Google Chrome	<i>Browser</i> yang digunakan untuk melakukan uji coba aplikasi

6.2. Uji Coba Program Pencarian Rute

Bagian ini menjelaskan mengenai pengujian terhadap program pencarian rute yang telah dirancang dengan menggunakan pemrograman *python* sehingga dapat dipastikan tingkat optimalisasi dari hasil yang dikeluarkan dan dapat dijadikan rekomendasi dalam pengambilan keputusan. Pengujian dilakukan dengan menggunakan aplikasi *Jetbrains PyCharm Community Edition 2016.1*. Pengujian dilakukan dengan melakukan perubahan pada nilai parameter *alpha* dan *beta* karena kedua parameter tersebut yang berperan dalam memberikan perubahan hasil pada pencarian rute yang optimum. Berikut adalah skenario pengujian yang akan dijalankan pada data titik-titik TPS di tiap armada:

- a. Skenario 1
Mengubah parameter *alpha* (α) yang digunakan dengan nilai: 0,1, 0,5 dan 0,8.
- b. Skenario 2
Mengubah parameter *beta* (β) yang digunakan dengan nilai: 2.25 dan 4.

Tabel 6.3 berikut adalah hasil pengujian untuk data seluruh titik TPS yang ada di Kabupaten Sidoarjo.

Tabel 6.3 Hasil Pengujian Berdasarkan Parameter

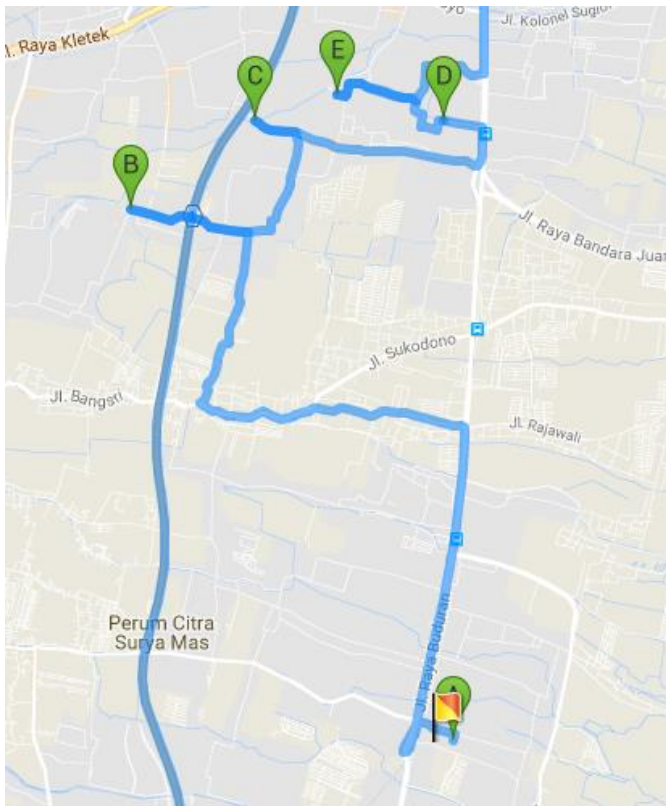
Alpha	Beta	Distance	Running time
0.1	2.25	5600	0.09659971794116369
	4	5600	0.09976322030050858
0.5	2.25	5600	0.09332090471337703
	4	5600	0.0936058614144478
0.8	2.25	5600	0.09622903274104198
	4	5600	0.09176269868485239

Untuk hasil pengujian untuk data di tiap armada dapat dilihat dalam lampiran 2.

6.3. Hasil Tampilan Visualisasi Rekomendasi Route

Bagian ini akan dijelaskan visualisasi yang dihasilkan dari pencarian rute dalam aplikasi sistem optimasi pencarian rute pengangkutan sampah.

Gambar 6.1 menampilkan contoh visualisasi rute yang dihasilkan dari rekomendasi rute optimum di salah satu armada.



Gambar 6.1 Hasil visualisasi rute

Hasil rekomendasi rute di setiap armada dapat dilihat di Lampiran 3.

6.4. Hasil Rekomendasi Rute Pengangkutan Sampah

Berdasarkan hasil yang didapatkan dalam pencarian rute optimal yang telah diujikan sebelumnya, didapatkan rekomendasi rute baru dengan perbedaan jarak seperti pada tabel

Tabel 6.4 Perbandingan Jarak Tempuh

Armada	Jarak Tempuh	
	Rute Lama	Rute Baru
1	60.1	48.8
2	65.3	61.2
3	54.5	56.9
4	87.7	57.9
5	50.8	53.6
6	61.8	53.4
7	59.5	58.9
8	70.2	88.4
9	62.5	49.1
10	61.6	58.2
11	65.9	61.1
12	74.8	70.9
13	76.9	69.3
14	90.9	51.4
15	71.8	76.6
16	68.8	61.8
17	93.8	84.4
18	80.4	70.5
19	72.4	88.5
20	76.2	74
21	70.4	72.3
22	77.8	56.9
23	93.8	55.3
24	69.3	63.7
25	71.2	60.8
26	84.4	88.7

Armada	Jarak Tempuh	
	Rute Lama	Rute Baru
27	84.3	56.4
28	56.9	70.7
29	79.7	86.4
30	54.5	59.5
31	61.2	81.9
32	60.1	72.8
33	65.8	51.6
34	65.4	95.7
35	67.5	73.1
36	66.8	86.6
37	54.1	58.5
38	65.9	50.7
39	61.97	55.4
40	72.7	56.7
TOTAL	2789.67	2628.7
SELISIH	160.97	

Berdasarkan tabel di atas, didapatkan total selisih jarak tempuh antara rute lama dengan rute baru sebesar sebesar 160.97 km. Adapun tabel 6.5 menampilkan rekomendasi rute baru tersebut sebagai berikut

Tabel 6.5 Hasil Rekomendasi Rute Baru

Armada ke-	Rekomendasi Rute
1	Pool – TPST Siwalanpanji – TPS SDN Siwalanpanji – TPS SMA Antartika – TPS SMP PGRI 1 - TPS JI Antartika – TPA Jabon
2	Pool – TPS Sukorejo – TPS PT Avian – TPS JI Raya Buduran – TPS Pasar Sono – TPA Jabon
3	Pool – TPS JI Mangundiprojo – TPS Sidokerto – TPS DS Ental Sewu – TPS Sono – TPS Gading Fajar – TPA Jabon

Armada ke-	Rekomendasi Rute
4	Pool – TPS Citra Garden – TPS Pondok Jati – TPS Perum Angrek (Pagerwojo) – TPS Gajah Magersari – TPS Perum Taman Tiara – TPA Jabon
5	Pool – TPS Bligo – TPS Desa Larangan – TPS Jl Raya Bligo – TPS Desa Klurak – TPA Jabon
6	Pool – TPS Graha Candi Mas – TPS Candi Asri – TPS Perum Candiloka – TPS Jembatan Candi – TPA Jabon
7	Pool – TPS Sungai Seruni – TPS Indomart – TPS Bursa Kupang – TPA Jabon
8	Pool – TPS Bakalan Keterangan – TPS Perumtas III (Grabagan) – TPS Jl Raya Wonoayu – TPS Pondok Ketimang – TPS Jl Raya Pilang – TPA Jabon
9	Pool – TPS Pondok Al Ghozini – TPS Jl. KH Khamdani – TPA Jabon
10	Pool – TPS Komplek AL Tebel – TPS Karangbong – TPA Jabon
11	Pool – TPS Alloy Steel – TPST Kebonsikep – TPA Jabon
12	Pool – TPS Jembatan Layang Waru – TPS KSM Janti – TPA Jabon
13	Pool – TPS Perum Surya Asri – TPS Perum Putri Juanda – TPS Jl. Satria – TPS Lesen – TPA Jabon
14	Pool – TPS Kir Candi – TPS Ps Krempeyeng – TPS Perum AL – TPA Jabon
15	Pool – TPS Delta Sari – TPS Sawotratap – TPA Jabon
16	Pool – TPS SMA Negeri Gedangan – TPS Hair Star – TPS Desa Ketajen – TPA Jabon
17	Pool – TPS Perum Bhayangkara – TPS Taman Aloha – TPS Perum PPLegi – TPS Pepe Legi – TPA Jabon
18	Pool – TPS Mc Donald Waru – TPS Kantor BKKN – TPS Mitra Keluarga – TPA Jabon
19	Pool – TPS Sambibulu – TPS Phok Pan – TPS Fuboru – TPA Jabon
20	Pool – TPS Gudang Garam – TPS Bungurasih – TPS Brimob Medaeng – TPS PLN Sepanjang – TPA Jabon
21	Pool – TPS Aloha – TPS Restoran Aloha – TPS Marinir Juanda – TPS Auri- TPS Jl Raya Juanda – TPA Jabon

Armada ke-	Rekomendasi Rute
22	Pool – TPS Stadion GOR – TPS Mall Giant – TPS Alun-Alun – TPS Pucang RW I / RT III – TPA Jabon
23	Pool – TPS Terminal Larangan – TPS Jl KH Mukmin – TPS Pondok Millenium – TPS Tenggulunan – TPA Jabon
24	Pool – Jl Raya Cemengkalang – TPS PT Burung Dara – TPS Jl Raya Suko – TPS Pasar Suko – TPS Perum Bumi Suko Indah – TPS Villa Jasmin – TPS Jl Raya Lebo – TPA Jabon
25	Pool – TPST Jari – TPS Praloyo – TPS Rangka – TPS SMPN 6 Sidoarjo – TPS Lingkar Timur – TPA Jabon
26	Pool – TPS Kampung Sukodono – TPS Perum Taman Puspa – TPS Sawo Cangkring – TPS Jl Raya Krian – TPA Jabon
27	Pool – TPS Jl Karang Tanjung – TPS Pasar Wates – TPS Perum Permata – TPS Perumtas II Tanggulangin – TPS Perum Tanggulangin – TPS Pondok Putat – TPA Jabon
28	Pool – TPS Kepuh Kemiri – TPS Perum Sidodadi – TPS Perum CSM – TPS Surya Kencana – TPA Jabon
29	Pool – TPS Jl Raya Tropodo – TPS Pondok Chandra – TPA Jabon
30	Pool – TPS Brimob Porong – TPS Pondok Pesantren Porong – TPS Porong – TPS Jl Raya Lajuk
31	Pool – TPS Perum Citra Diamon – TPS Perumtas IV – TPS Puskesmas Prambon – TPA Jabon
32	Pool – TPS Sungai Depan Giant Waru – TPS Gate Way – TPS Makro – TPA Jabon
33	Pool – TPS Jl Jasem – TPS Sekardangan – TPS Gebang – TPA Jabon
34	Pool – TPS Angkasa Pura – TPS Bandara Juanda – TPA Jabon
35	Pool – TPS Bea Cukai Juanda – TPS Pasar Wisata Juanda – TPA Jabon
36	Pool – TPS Ponokowan – TPS Kemangsen – TPA Jabon
37	Pool – TPS Graha Kota – TPS SMP 4 – TPA Jabon

Armada ke-	Rekomendasi Rute
38	Pool – TPS Pabrik Saos – TPS JI Kartini – TPS JI Monginsidi – TPS Perum BCF – TPS RS Siti Hajar – TPA Jabon
39	Pool – TPS RS Delta Surya – TPS Pondok Mutiara – TPS JI Raya Jati
40	Pool – TPS Perum Taman Pinang Indah – TPS Hutan Putri – TPS Mall Ramayana – TPA Jabon

6.5. Analisa Hasil Pengujian

- Berdasarkan hasil pengujian dengan menggunakan scenario perubahan nilai parameter nilai alfa dan beta didapatkan bahwa hasil paling optimum dengan menggunakan nilai alfa sebesar 0.8 dan nilai beta sebesar 4 yang memberikan hasil running time tercepat. Namun tidak memberikan perbedaan hasil pada jarak karena jumlah inputan titik TPS yang dicari untuk setiap armada yang terlalu sedikit.
- Dengan menggunakan metode ACO ini didapatkan pemangkasan jarak tempuh sebesar 160.97 km.

[halaman ini sengaja dikosongkan]

BAB VII

KESIMPULAN DAN SARAN

Pada bagian ini berisi kesimpulan dari semua proses yang telah dilakukan dalam tugas akhir serta saran yang dapat diberikan untuk pengembangan kedepannya yang lebih baik.

7.1. KESIMPULAN

Dari penelitian yang telah dilakukan dapat disimpulkan bahwa:

1. Algoritma *Ant Colony Optimization* dapat digunakan dalam menentukan rute pengangkutan sampah.
2. Rekomendasi rute yang dihasilkan dapat divisualisasikan dalam rute peta dengan menggunakan fitur pada *google maps API*

7.2. SARAN

Adapun saran yang dapat diberikan berdasarkan pengujian, penarikan kesimpulan dan batasan masalah dari penelitian tugas akhir ini adalah:

1. Data pada studi kasus yang digunakan hanya dalam lingkup satu Kabupaten/Kota saja yaitu Kabupaten Sidoarjo. Aplikasi dapat dikembangkan dalam permasalahan serupa dengan cakupan yang lebih luas, seperti rute distribusi barang atau sejenisnya.
2. Penelitian ini hanya menerapkan penentuan rute pengangkutan sampah untuk metode *Stationary Container System (SCS)* saja, sehingga sangat disarankan untuk bias menerapkan penggunaan aplikasi pada metode pengangkutan sampah lainnya yaitu *Hauled Container System (HCS)*
3. Dengan menggunakan studi kasus yang sama, bisa diterapkan penggunaan algoritma lain untuk mencari rute pengangkutan sampah yang optimum.

[halaman ini sengaja dikosongkan]

DAFTAR PUSTAKA

- [1] M. P. D. I. H. R. Sudrajat, "BAB I - Mengelola Sampah Kota," in *Mengelola Sampah Kota - Seri Agritekno*, Bogor, Penebar Swadaya, 2006, pp. 5-6.
- [2] D. P. K. P. d. O. R. K. Sidoarjo, "Sidoarjo Selayang Pandang," Dinas Pariwisata Kebudayaan Pemuda dan Olah Raga Kabupaten Sidoarjo, [Online]. Available: <http://pariwisata.sidoarjokab.go.id/profil.php>. [Accessed 19 Januari 2017].
- [3] B. P. S. K. Sidoarjo, "Kabupaten Sidoarjo dalam Angka 2016," Badan Pusat Statistik Kabupaten Sidoarjo, Sidoarjo, 2016.
- [4] S. Hadi, "Pemkab Sidoarjo Salahkan Warga Soal Menumpuknya Sampah," MetroTV News, 23 Juli 2016. [Online]. Available: <http://m.metrotvnews.com/jatim/peristiwa/eN4vz8ok-pemkab-sidoarjo-salahkan-warga-soal-menumpuknya-sampah>. [Accessed 18 Januari 2017].
- [5] H. T. S. V. G. Tchobanoglous, "Engineering Principles and Management Issues," in *Integrated Solid Waste Management*, New York, USA, McGraw-Hill, Inc., 1993.
- [6] N. Achmad, Interviewee, *Kegiatan Pengangkutan Sampah di Kabupaten Sidoarjo*. [Interview]. 19 Februari 2017.
- [7] Rizzoli, "Ant Colony Optimization for Real-World Vehicle Routing Problem," *Swarm Intell*, pp. 135-151, 2007.

- [8] P. & V. D. Toth, "The Vehicle Routing Problem," *Philadelphia : Society for Industrial and Applied Mathematics*, 2002.
- [9] C. Blum, "Ant Colony Optimization : Introduction and recent trends," *Physic of Live Reviews* 2, pp. 353-373, 2005.
- [10] Y. Z.-Z. Y. B. Yu Bin, "An improved ant colony optimization for vehicle routing problem," *European Journal of Operational Research*, vol. 196, pp. 171-176, 2009.
- [11] S. V. M. B. Iliya Markov, "Integrating a heterogeneous fixed fleet and a flexible assignment of destination depots in the waste collection VRP with intermediate facilities," *Transportation Research Part B*, pp. 1-18, 2015.
- [12] C. K. Can B. Kalayci, "An Ant Colony System Empowered Variable Neighborhood Search Algorithm for the Vehicle Routing Problem with Simultaneous Pickup and Delivery," *Expert System with Application*, vol. 66, pp. 163-175, 2016.
- [13] D. L. H. d. K. K. Sidoarjo, "Peraturan Bupati Sidoarjo Nomor 77 Tahun 2016," Sidoarjo, 2016.
- [14] S. N. Indonesia, "Tata cara teknik operasional pengelolaan sampah perkotaan SNI 19-2454-2002," Badan Standarisasi Nasional, 2002.
- [15] N. Achmad, Interviewee, *Kegiatan Pengangkutan Sampah di Kabupaten Sidoarjo*. [Interview]. 19 Februari 2017.

- [16] I. Setiawan, "Antara Jatim," 25 Juli 2016. [Online]. Available:
<http://www.antarajatim.com/lihat/berita/181304/pemkab-sidoarjo-terus-kembangkan-tpst>. [Accessed 12 Februari 2017].
- [17] H. P. Sri Kusumadewi, "Penyelesaian Masalah Optimasi," Graha Ilmu, Yogyakarta, 2005.
- [18] M. d. S. T. Dorigo, Ant Colony Optimization, London: MIT Press, 2004.
- [19] F. H. S.T., "Penerapan Algoritma Ant Colony System (ACS) dan Algoritma Generate And Test Dalam Pemecahan Solusi Travelling Salesman Problem (TSP)," Tugas Akhir Teknik Informatika UNIKOM, Bandung, 2005.

[halaman ini sengaja dikosongkan]

LAMPIRAN A

Data yang digunakan dalam tugas akhir

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
01	Nanang Iskandar	W 8054 PP	TPS Sidokerto	Buduran	-7.427615	112.711109	60.1
			TPS Sukorejo	Buduran	-7.42004	112.717253	
			TPST Siwalanpanji	Buduran	-	112.7254143	
			TPS SDN Siwalanpanji	Buduran	-7.432267	112.728402	
			SMA Antartika	Buduran	-7.431704	112.726543	
02	Joko Prayitno	W 8076 NP	TPST PT Avian	Buduran	-7.418909	112.725523	65.3
			SMP PGRI 1	Buduran	-7.431318	112.724925	
			TPS Pondok Al-Ghozini	Buduran	-7.4281	112.724118	
			TPS Perum Surya Asri (Dapurno)	Buduran	-7.417853	112.757257	
03	Arsiadi	W 8072 PP	Jl. KH Khamdani	Buduran	-7.428407	112.725056	54.5

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS Bursa Kupang	Buduran	- 7.4015367	112.7268687	
			TPS Jl. Antartika	Buduran	- 7.4307691	112.7228216	
			TPS Jl. Raya Buduran	Buduran	-7.415761	112.724974	
			TPS Sono	Buduran	-7.432447	112.720377	
04	M. Assykun Nahdi	W 8303 PP	TPS Perum Candiloka	Candi	-7.494257	112.719477	87.7
			TPS Candi Asri	Candi	-7.493967	112.71687250.8	
			TPS Perum CSM	Candi	-7.460478	112.687796	
			Terminal Larangan	Candi	-7.466627	112.71302	
			TPS Tenggulunan	Candi	-7.46971	112.71078	
05	Kamim	W 8300 PP	TPS Perum AL	Candi	-7.477639	112.702337	50.8
			TPS Ps. Krempyeng Sugiharwas	Candi	-7.477367	112.701596	

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS Jl. Raya Bligo	Candi	-7.475349	112.71509	
			TPS Kir Candi	Candi	-7.4826914	112.7105095	
06	Aminin	W 8250 PP	TPS Jl Karang Tanjung	Candi	-7.487192	112.690701	61.8
			TPS Bligo	Candi	-7.474292	112.717225	
			TPS Desa Klurak	Candi	-7.479375	112.725452	
			TPS Jembatan Candi	Candi	-7.494304	112.732967	
07	Mulyono		Pondok Millenium	Candi	-7.469346	112.706768	59.5
			TPS Perum Sidodadi	Candi	-7.457085	112.6791	
			TPS Desa Larangan	Candi	-7.46912	112.720876	
08	Moh. Maslukhin	W 8255 PP	TPS Desa Ketajen	Gedangan	-7.387379	112.732604	70.2
			TPS Aloha	Gedangan	-7.373912	112.728558	

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS Sungai Seruni	Gedangan	-7.399051	112.724847	
			TPS Makro	Gedangan	-7.367755	112.727134	
			TPS Hair Star	Gedangan	-7.387904	112.744354	
09	M Arif Wahyudi	W 8072 NP	TPS Indomart	Gedangan	-7.399895	112.727094	62.5
			TPS Restoran Aloha	Gedangan	-7.3742506	112.7282426	
10	Aris Sudrajat	W 8248 PP	TPS Karangbong	Gedangan	-7.403149	112.714927	61.6
			TPST Kebonsikep	Gedangan	-7.390629	112.723074	
11	Sugeng Priambodo	W 8249 PP	TPS Marinir Juanda	Gedangan	-7.3798339	112.7390049	65.9
			TPS Alloy Stell	Gedangan	-7.3839035	112.7242795	
12	Rahmat Wahyudi	W 8108 PP	TPS Sawotratap	Gedangan	-7.367898	112.72927	74.8
			TPS Gudang Garam	Gedangan	-7.3543774	112.7210301	
13	Djainul Ma'arif	W 8053 PP	TPS Komplek AL Tebel	Gedangan	-7.4042878	112.7248046	76.9

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS SMA Negeri Gedangan	Gedangan	-7.387845	112.741398	
			TPS Perum Putri Juanda	Waru	-7.3999753	112.7623732	
			TPS Taman Aloha	Waru	-7.367158	112.704671	
14	Jiran	W 8306 PP	TPS Jl Raya Krian	Krian	-7.406027	112.585173	84.4
			TPS Perum Citra Diamon	Krian	-7.4462837	112.5687863	
			TPS Kemangsen	Krian	-7.411169	112.569028	
15	Tarmuji	W 8071 PP	TPS Ponokowan	Krian	-7.396165	112.59625	84.3
			TPS Bakalan Keterungan	Krian	-7.418858	112.58401	
16	Djakfar Sodiq	W 8302 PP	TPS Kampung Sukodono	Sukodono	-7.394199	112.673745	77.8
			TPS Perum Taman Puspa	Taman	-7.423074	112.687388	

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS PLN Sepanjang	Taman	-7.349824	112.707393	
17	Sugianto	W 8161 NP	TPS Perum Bhayangkara	Sukodono	-7.37632	112.691748	93.8
			TPS Sambibulu	Taman	-7.377696	112.661428	
			TPS Phok Pan	Taman	-7.366488	112.661791	
			TPS Fuboru	Taman	-7.379186	112.627686	
18	Mulyono	W 8075 PP	TPS Jl Raya Wonoayu	Wonoayu	-7.439811	112.628394	71.2
			TPS Sawo Cangkring	Wonoayu	-7.418083	112.65737	
			TPS Pondok Ketimang	Wonoayu	-7.443507	112.645163	
19	Abdul Munib	W 8251 PP	TPS Surya Kencana	Tulangan	-7.457918	112.660534	69.3
			TPS Kepuh Kemiri	Tulangan	-7.455947	112.648309	
			TPS Perumtas III	Wonoayu	-7.442406	112.629459	
20	Agus Mulyanto	W 8305 PP	TPS PT. Burung Dara (Suprama)	Sidoarjo	-7.443566	112.6823066	61.2

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS Jl. Raya Cemengkalang	Sidoarjo	-7.442641	112.68678	
			TPS Pabrik Saos	Sidoarjo	-7.440248	112.721446	
			TPS Rangka	Sidoarjo	-7.462986	112.734459	
21	M Sulkan	W 8051 PP	TPS Gebang	Sidoarjo	-7.468754	112.73086	60.1
			TPS Alun-Alun	Sidoarjo	-7.445191	112.715681	
			TPS Citra Garden	Sidoarjo	-7.43793	112.700073	
			Stadion GOR	Sidoarjo	-7.44711	112.708629	
			TPS Jl. SMPN 6 Sidoarjo	Sidoarjo	-7.451971	112.736483	
22	Moch Soleh	W 8028 NP	TPS Perum BCF	Sidoarjo	-7.460073	112.725918	65.8
			TPS Jl. KH Mukmin	Sidoarjo	-7.459655	112.714201	
			TPS Perum Bumi Suko Indah	Sidoarjo	-7.445155	112.676699	
			TPS Perum Taman Tiara	Sidoarjo	-7.437414	112.713055	
23	Febri Dwi	W 8306 PP	TPS Jl. Raya Suko	Sidoarjo	-7.44429	112.682325	65.4

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS Jl. Raya Lebo	Sidoarjo	-7.45062	112.674242	
			TPS Jl. Raya Pilang	Sidoarjo	-7.446362	112.652786	
			TPS Pondok Mutiara	Sidoarjo	-7.448426	112.702032	
24	Warni	W 8076 NP	TPS Hutan Putri	Sidoarjo	- 7.4516023	112.7132743	67.5
			TPS Perum Anggrek (Pagerwojo)	Sidoarjo	-7.439588	112.705027	
			Sampah Lingkar Timur	Sidoarjo	-7.437844	112.735044	
			TPST Jari (Lingkar Timur)	Sidoarjo	-7.457315	112.736602	
			TPS Jl. Kartini	Sidoarjo	-7.445803	112.722443	
			TPS Jl. Jasem	Sidoarjo	-7.459862	112.718077	
			TPS Praloyo	Sidoarjo	-7.472016	112.739199	
			Perum BCF	Sidoarjo	-7.460043	112.725914	
25	M. Fatakha	W 8069 PP	TPS Jl. Mangundiprojo	Sidoarjo	-7.431694	112.721631	66.8

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS Pasar Sono	Sidoarjo	- 7.4232619	112.7145135	
			TPS Jl. Raya Jati	Sidoarjo	-7.44563	112.696802	
			TPS Pasar Suko	Sidoarjo	- 7.4454797	112.6793353	
			TPS Sekardangan	Sidoarjo	-7.464231	112.723405	
26	Solikin	W 8253 PP	TPS DS Ental Sewu	Sidoarjo	-7.424268	112.704583	54.1
			Pondok Jati	Sidoarjo	-7.439658	112.703718	
			TPS Mall Giant	Sidoarjo	- 7.4490591	112.7092392	
			TPS Gading Fajar	Sidoarjo	- 7.4308769	112.7217183	
27	Nur Cholis	W 8070 PP	TPS Pucang RW I / RT III	Sidoarjo	-7.438014	112.720487	65.9
			TPS Graha Candi Mas	Sidoarjo	- 7.4879416	112.7138472	
			TPS Graha Kota	Sidoarjo	- 7.4505016	112.6804612	

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS RS Siti Hajar	Sidoarjo	- 7.4576371	112.7198672	
			TPS Jl. Monginsidi	Sidoarjo	-7.449731	112.721444	
			TPS Mall Ramayana	Sidoarjo	- 7.4517156	112.7133429	
28	Pujo Hariyanto	W 8256 PP	TPS SMP 4	Sidoarjo	- 7.4465468	112.6838928	61.9
			TPS RS Delta Surya	Sidoarjo	-7.44709	112.6993708	
			TPS Villa Jasmin	Sidoarjo	-7.444804	112.671146	
			TPS Gajah Magersari	Sidoarjo	-7.44045	112.707887	
29	Abdul Jailani	W 8247 PP	TPS Jl. Satria (Zeni)	Sidoarjo	-7.388949	112.760555	72.7
			TPS Perum Taman Pinang Indah	Sidoarjo	-7.458578	112.703502	
30	Allen	W 8562 PP	TPS Pondok Chandra	Waru	- 7.3445212	112.7750117	90.9

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS Pepe Legi	Waru	-7.364312	112.71352	
			TPS Kantor BKKN	Waru	-7.36404	112.728724	
			TPS KSM Janti	Waru	-7.350167	112.727733	
31	Suwanto	W 8246 PP	TPS Jembatan Layang Waru	Waru	-7.355767	112.728979	71.8
			TPS Perum PP Legi	Waru	-7.367026	112.724937	
			Sungai Depan Giant Waru	Waru	-7.367402	112.728905	
32	Supri		TPS Mc Donald Waru	Waru	-7.36513	112.729008	68.8
			TPS Delta Sari	Waru	-7.362546	112.745533	
			TPS Gate Way	Waru	-7.36775	112.72893	
33	Agus Sucipto	W 8013 PP	TPS Mitra Keluarga	Waru	-7.363076	112.729062	74
			TPS Brimob Medaeng	Waru	-7.352494	112.710016	
			TPS Bungurasih	Waru	-7.353467	112.722571	
34	Sutrisno	W 8301 PP	TPS Angkasapura	Sedati	-7.373831	112.800322	80.4

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS Bandara Juanda	Sedati	-7.371533	112.780227	
35	Mudakir	W 8117 PP	TPS Jl. Raya Juanda	Sedati	-7.380967	112.749007	72.4
			TPS Pasar Wisata Juanda	Sedati	-7.373621	112.763504	
36	Miskan		TPS Bea Cukai Juanda	Sedati	-7.381391	112.758408	76.2
			TPS Jl. Raya Tropodo	Sedati	-7.359124	112.764548	
37	Choirul Anan		TPS Lesen	Sedati	-7.390902	112.759091	70.4
			TPS AURI	Sedati	-7.379465	112.750342	
38	Ami Yudhi Arno	W 8307 PP	TPS Perumtas II Tanggulangin	Tanggulangin	-7.503989	112.703893	54.5
			TPS Pasar Wates	Tanggulangin	-7.49824	112.692704	
			TPS Perum Permata	Tanggulangin	-7.503157	112.703059	
			Perum Tanggulangin	Tanggulangin	-7.509316	112.710421	

Armada ke-	Nama Supir	Plat Nomor	TPS/TPST	Kecamatan	Longitude	Latitude	Jarak Tempuh
			TPS Pondok Putat	Tanggulangin	-7.51654	112.731957	
39	Ngaderi	W 8304 PP	Jl. Raya Lajuk (Porong)	Porong	-7.527082	112.674288	56.9
			Pondok Pesantren	Porong	-7.543947	112.697443	
			TPS Porong	Porong	-7.539067	112.68385	
40	Andri Mudjiyanto	W 8309 PP	Brimob Porong	Porong	-7.54151	112.6944289	79.7
			TPS Prumtas IV	Prambon	-7.448795	112.570337	
			TPS Puskesmas Prambon	Prambon	-7.474034	112.560629	

LAMPIRAN B

Hasil Uji Coba dengan Skenario

Tabel B.1 berikut adalah hasil pengujian berdasarkan perubahan parameter alfa dengan nilai 0.8, dan beta dengan nilai 4 untuk semua rute di setiap armada

Tabel B.1 Hasil Pengujian Berdasarkan Perubahan Parameter

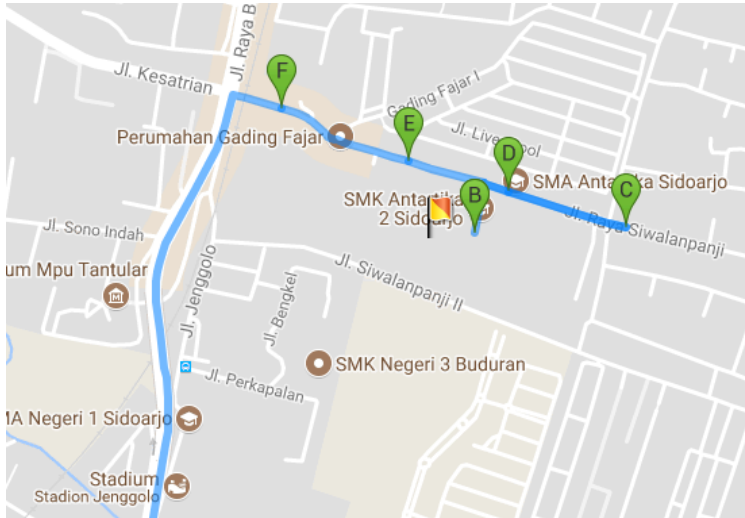
Armada	Running time
1	0.07548937691077957
2	0.04230339195069766
3	0.05787941501770112
4	0.06355138155702274
5	0.042026283633342834
6	0.042746523769735666
7	0.03018609121512299
8	0.061894768871136784
9	0.017758477461856853
10	0.019746533429284854
11	0.018411100753927812
12	0.0175978874565924
13	0.04909948852687045
14	0.03541915198817671
15	0.020701017633507044
16	0.029165801332051832
17	0.04279965129027428
18	0.029698283981086603
19	0.029909586619592463
20	0.042770672642707765
21	0.05746224323710812
22	0.04259559331366005
23	0.04229976961975185
24	0.09121270810291285
25	0.05822232901390492

Armada	Running time
26	0.04606276775062908
27	0.07391366294935016
28	0.04195383701442654
29	0.0183090717656207
30	0.04224664209921323
31	0.02961195176021135
32	0.028977440122869465
33	0.02918995020502393
34	0.017787456109423373
35	0.017073253191273563
36	0.017802549155030933
37	0.01783756502084048
38	0.05888461185516472
39	0.03850175562306507
40	0.028923105158682242

LAMPIRAN C

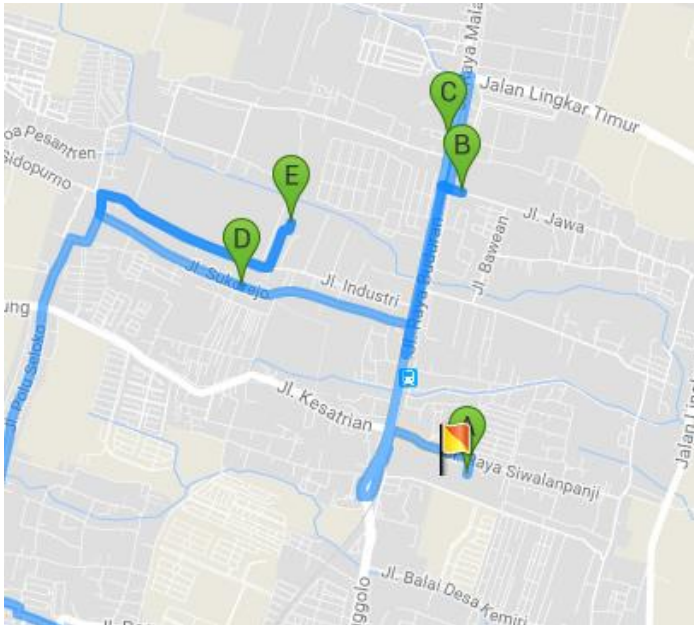
Visualisasi Rekomendasi Rute Optimum

Gambar C.1 berikut adalah visualisasi rute untuk Armada 1



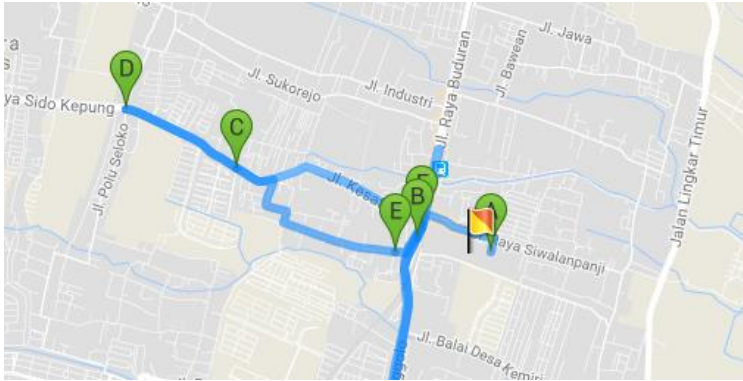
Gambar C.1 Visualisasi Rute Armada1

Gambar C.2 berikut adalah visualisasi rute untuk Armada2



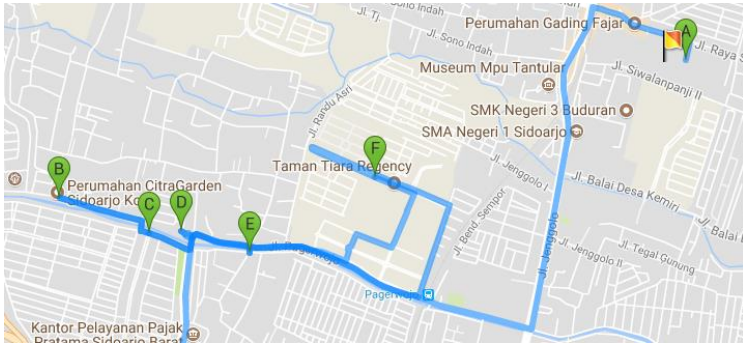
Gambar C.2 Visualisasi Rute Armada2

Gambar C.3 berikut adalah visualisasi rute untuk Armada3



Gambar C.3 Visualisasi Rute Armada3

Gambar C.4 berikut adalah visualisasi rute untuk Armada4



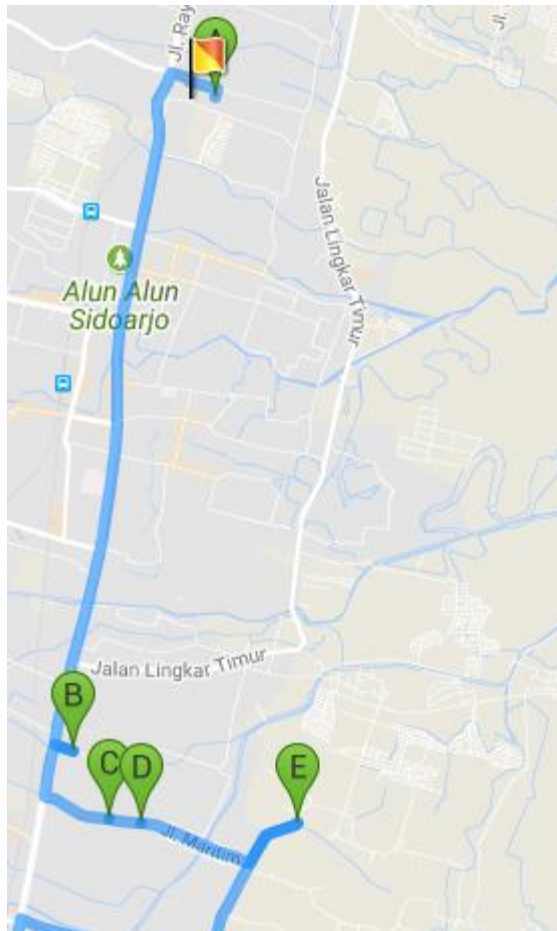
Gambar C.4 Visualisasi Rute armada4

Gambar C.5 berikut adalah visualisasi rute untuk armada5



Gambar C.5 Visualisasi Rute Armada5

Gambar C.6 berikut adalah visualisasi rute untuk armada6



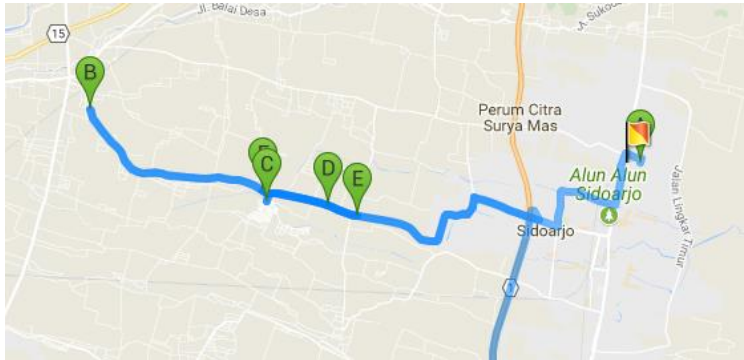
Gambar C.6 Visualisasi Rute Armada6

Gambar C.7 berikut adalah visualisasi rute untuk armada7



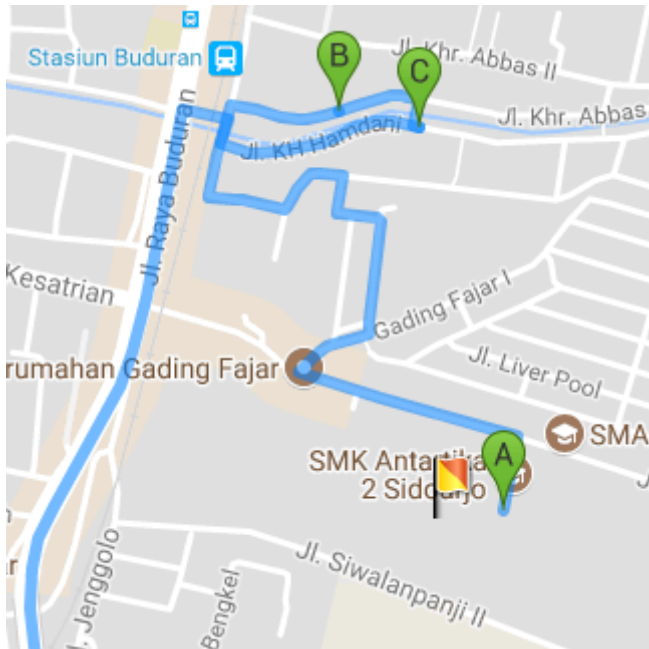
Gambar C.7 Visualisasi Rute Armada7

Gambar C.8 berikut adalah visualisasi rute untuk armada8



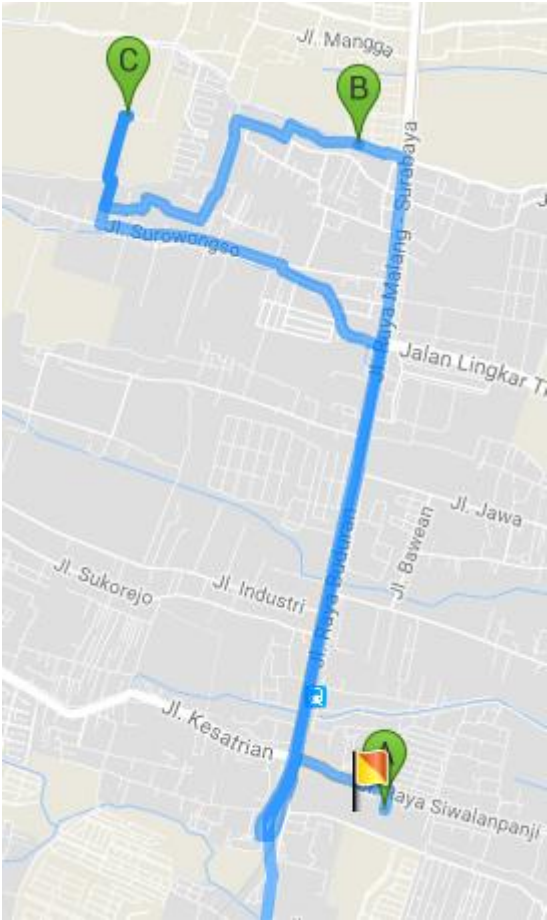
Gambar C.8 Visualisasi Rute Armada8

Gambar C.9 berikut adalah visualisasi rute untuk Armada9



Gambar C.9 Visualisasi Rute sub-wilayah Armada9

Gambar C.10 berikut adalah visualisasi rute untuk armada10



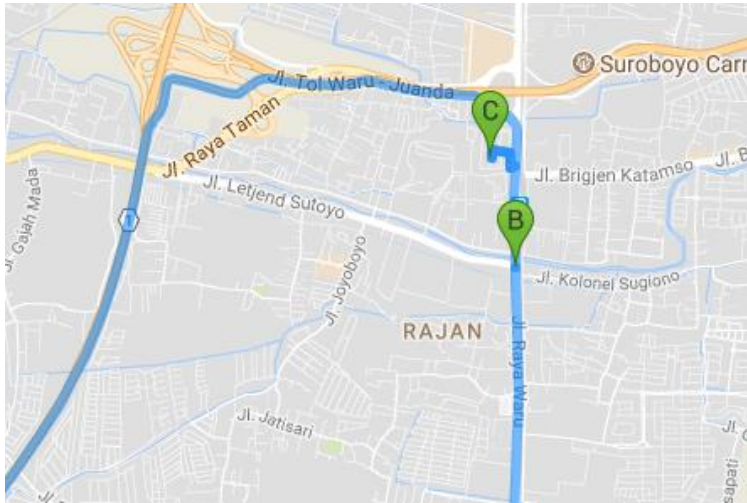
Gambar C.10 Visualisasi Rute Armada10

Gambar C.11 berikut adalah visualisasi rute untuk armada11



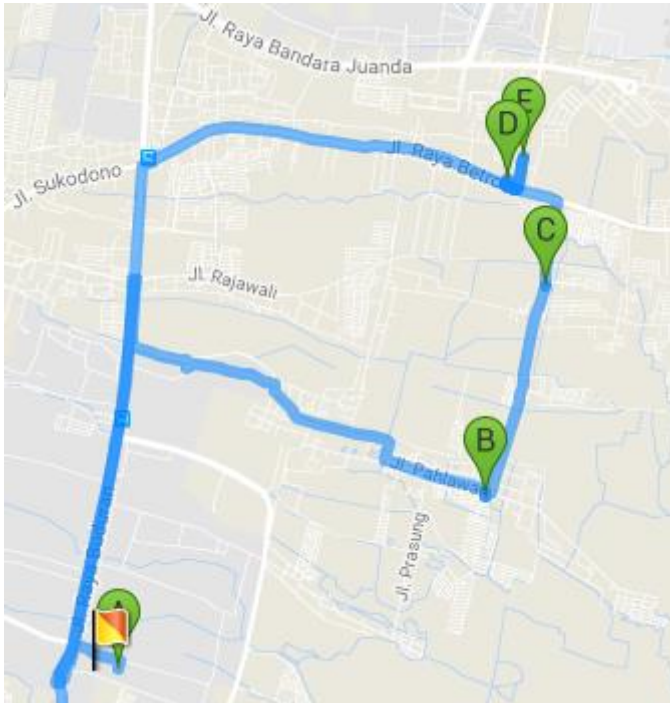
Gambar C.11 Visualisasi Rute armada11

Gambar C.12 berikut adalah visualisasi rute untuk armada12



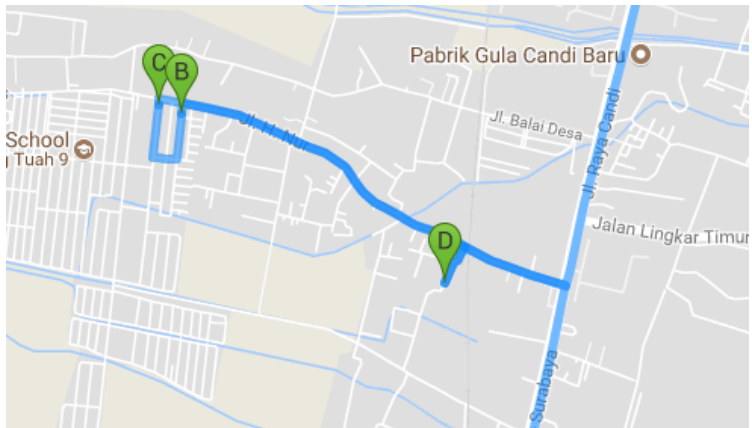
Gambar C.12 Visualisasi Rute armada12

Gambar C.13 berikut adalah visualisasi rute untuk armada13



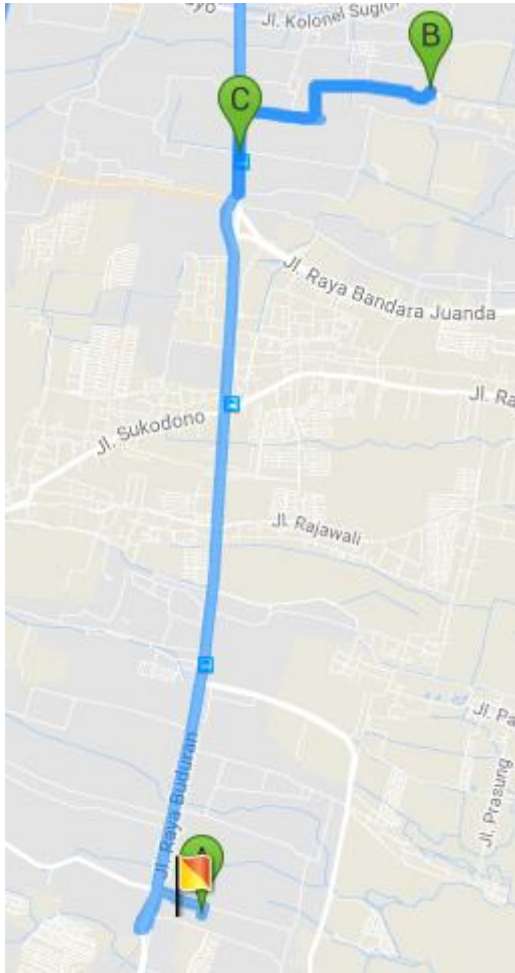
Gambar C.13 Visualisasi Rute armada13

Gambar C.14 berikut adalah visualisasi rute untuk armada14



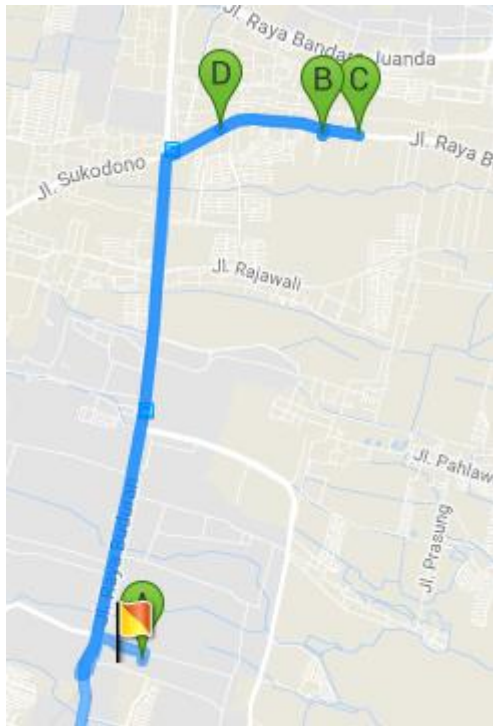
Gambar C.14 Visualisasi Rute armada14

Gambar C.15 berikut adalah visualisasi rute untuk armada15



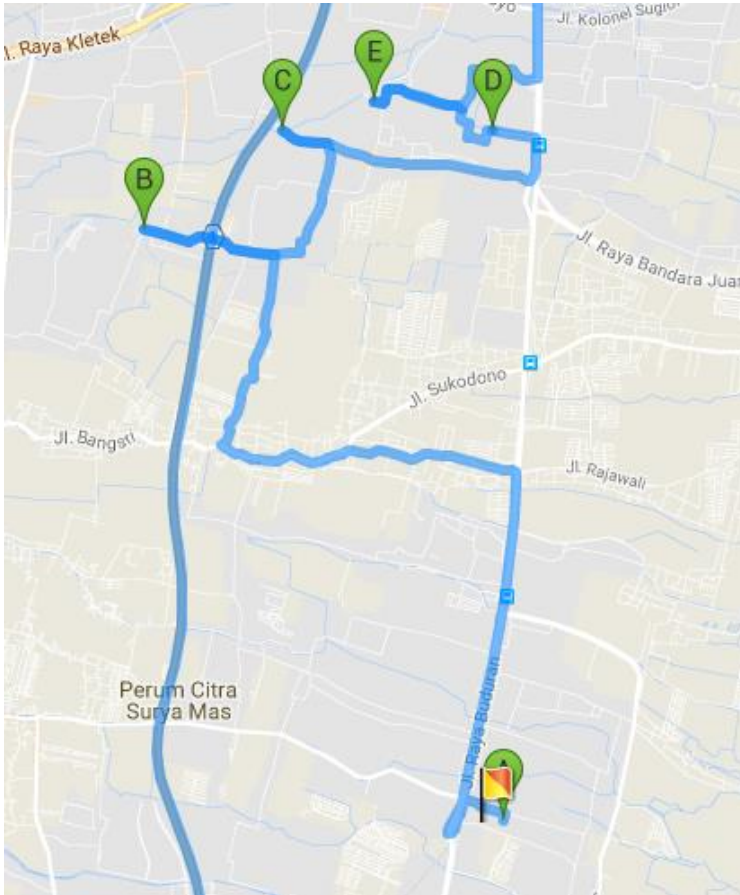
Gambar C.15 Visualisasi Rute armada15

Gambar C.16 berikut adalah visualisasi rute untuk armada16



Gambar C.16 Visualisasi Rute armada16

Gambar C.17 berikut adalah visualisasi rute untuk armada17



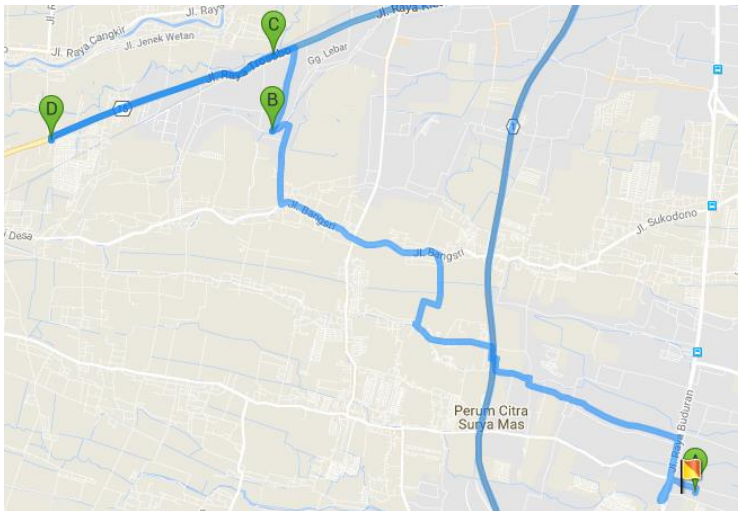
Gambar C.17 Visualisasi Rute armada17

Gambar C.18 berikut adalah visualisasi rute untuk armada18



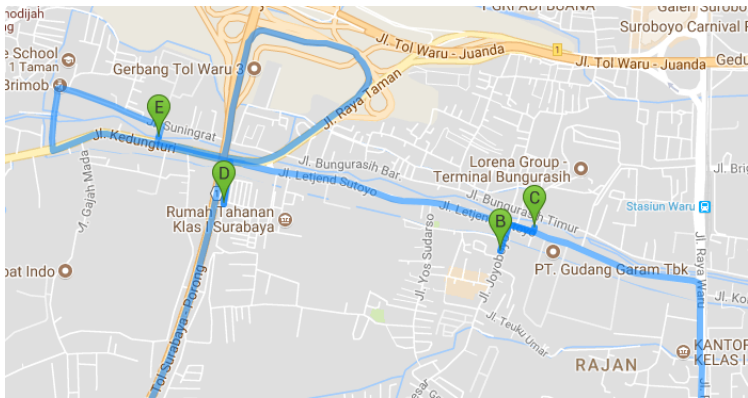
Gambar C.18 Visualisasi Rute armada18

Gambar C.19 berikut adalah visualisasi rute untuk armada19



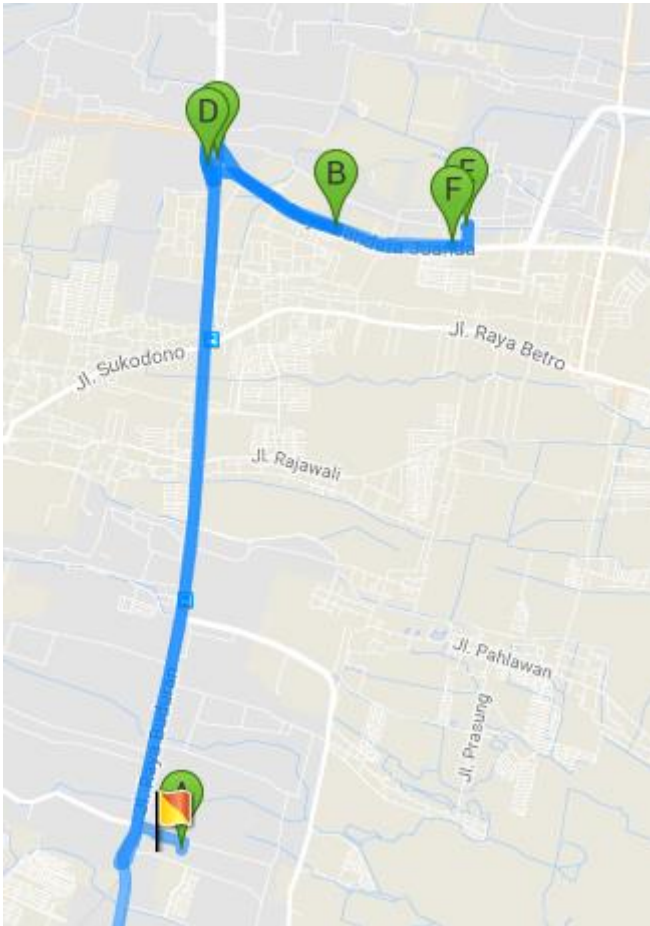
Gambar C.19 Visualisasi Rute Armada 19

Gambar C.20 berikut adalah visualisasi rute untuk armada20



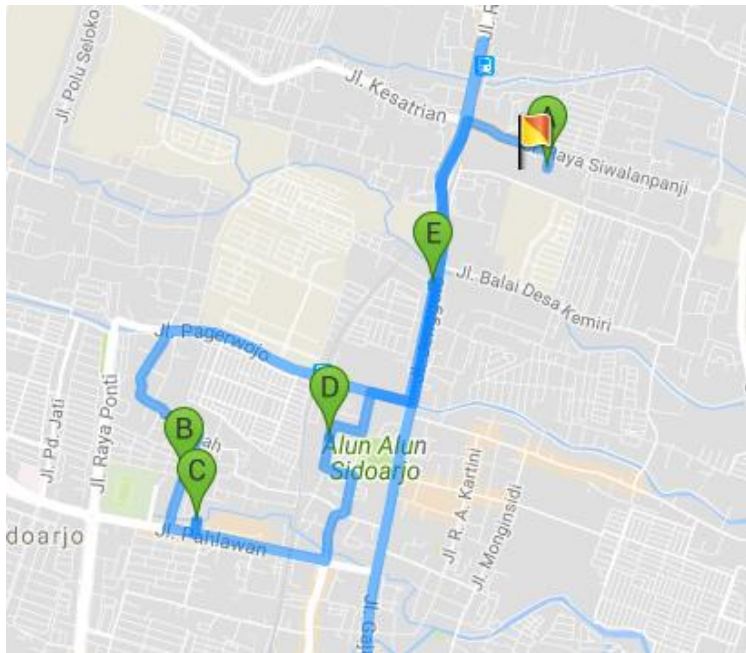
Gambar C.20 Visualisasi Rute Armada20

Gambar C.21 berikut adalah visualisasi rute untuk armada21



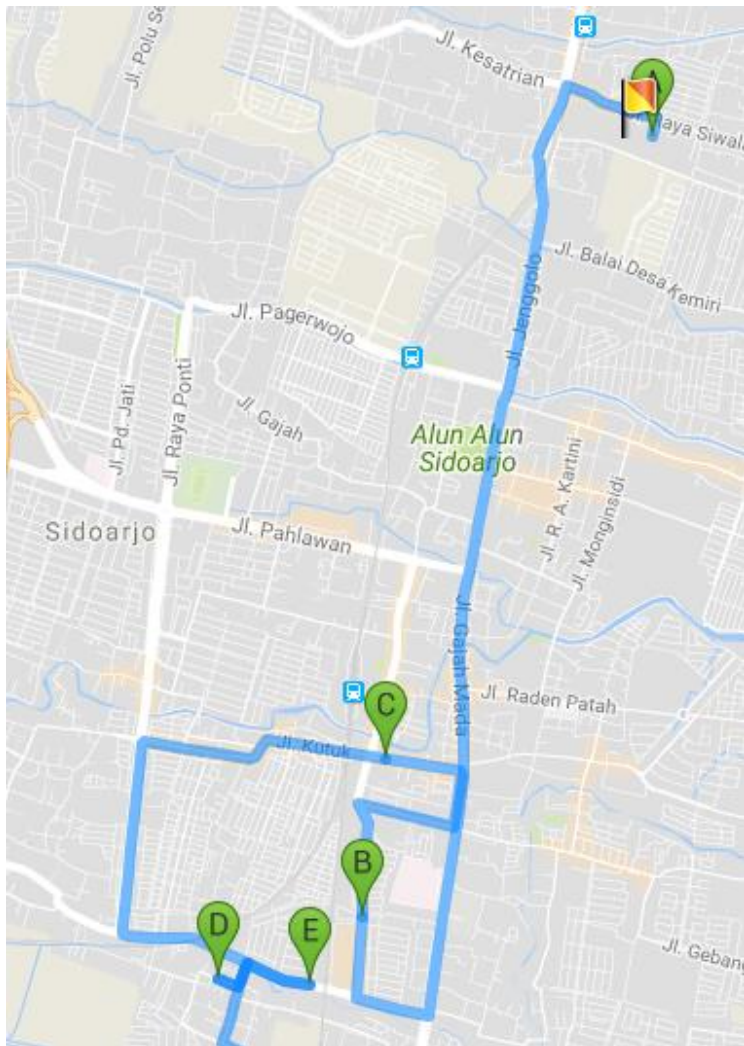
Gambar C.21 Visualisasi Rute Armada 21

Gambar C.22 berikut adalah visualisasi rute untuk armada22



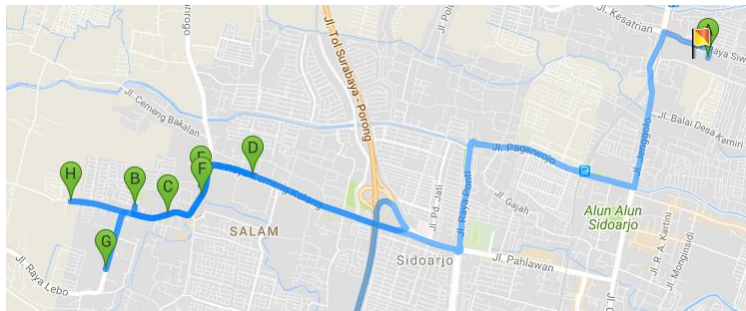
Gambar C.22 Visualisasi Rute Armada 22

Gambar C.23 berikut adalah visualisasi rute untuk armada23



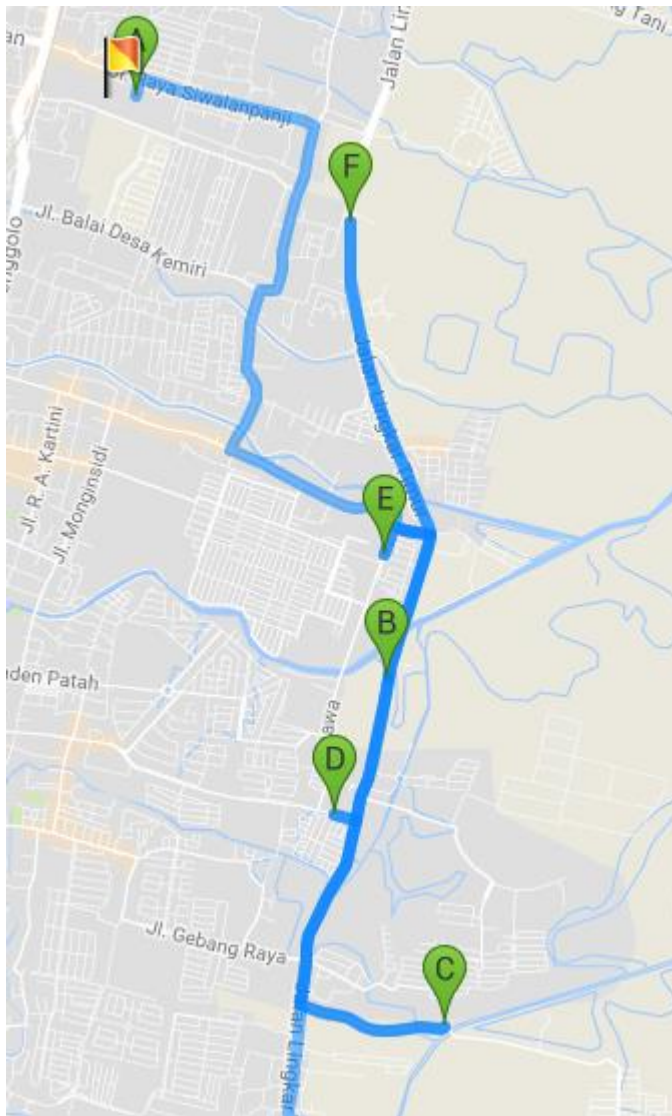
Gambar C.23 Visualisasi Rute Armada 23

Gambar C.24 berikut adalah visualisasi rute untuk armada24



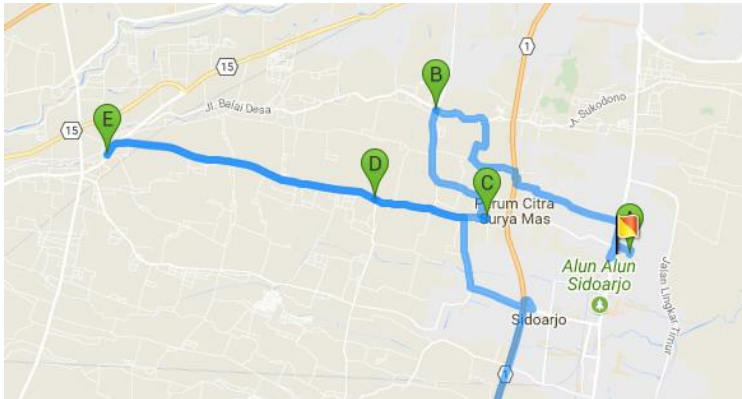
Gambar C.24 Visualisasi Rute Armada 24

Gambar C.25 berikut adalah visualisasi rute untuk armada25



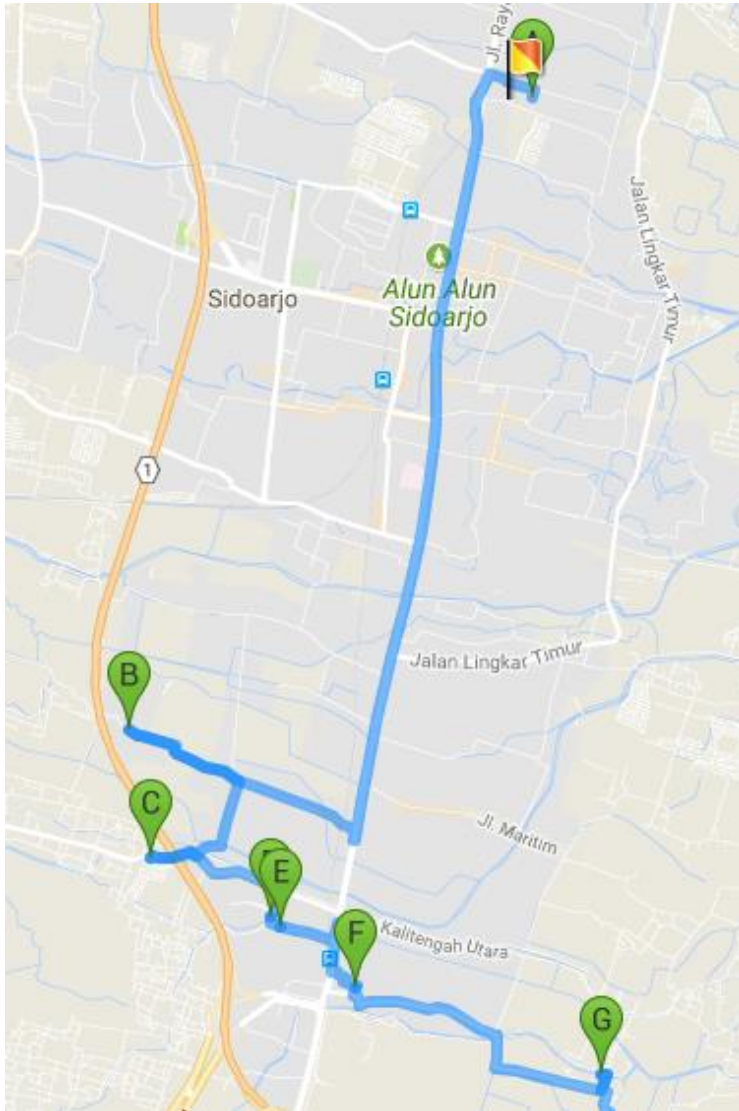
Gambar C.25 Visualisasi Rute Armada 25

Gambar C.26 berikut adalah visualisasi rute untuk armada26



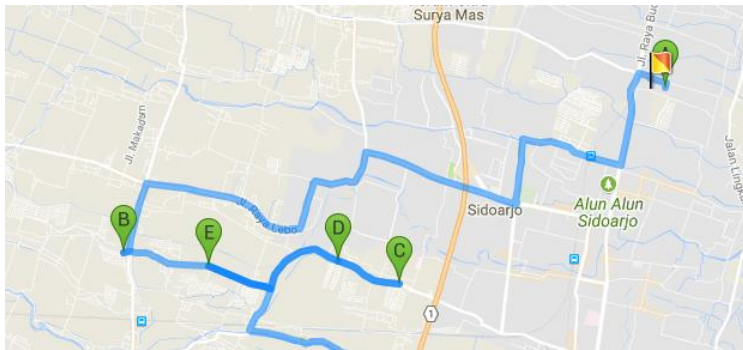
Gambar C.26 Visualisasi Rute Armada 26

Gambar C.27 berikut adalah visualisasi rute untuk armada27



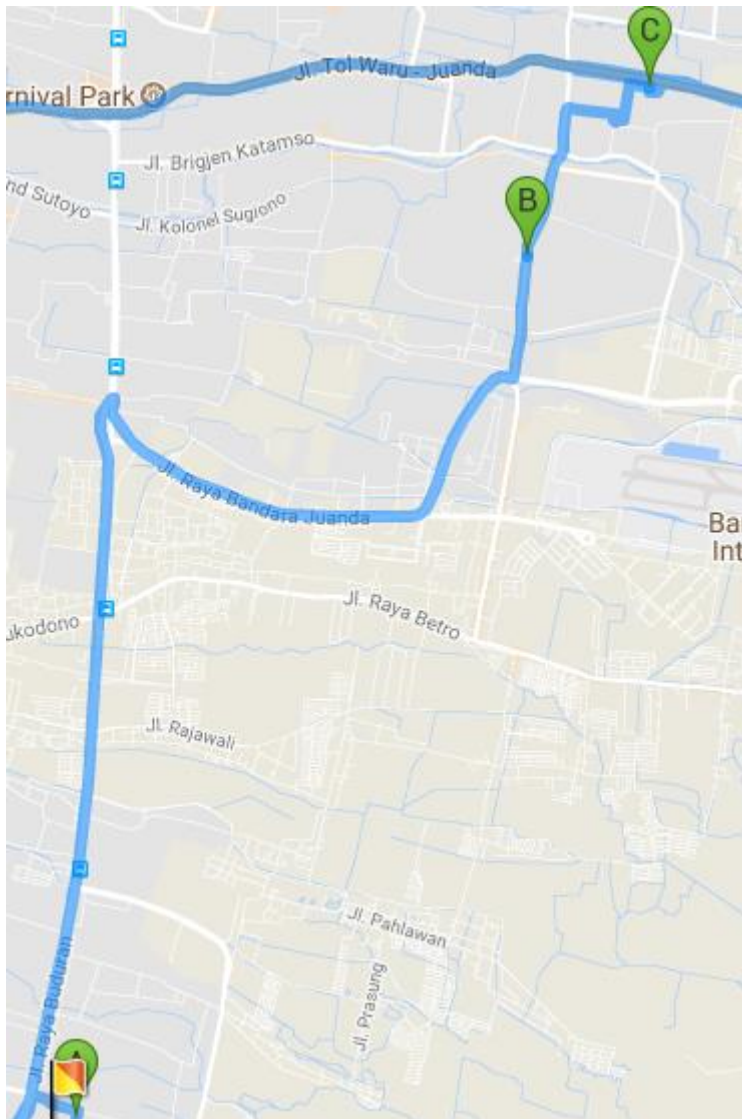
Gambar C.27 Visualisasi Rute Armada27

Gambar C.28 berikut adalah visualisasi rute untuk armada28



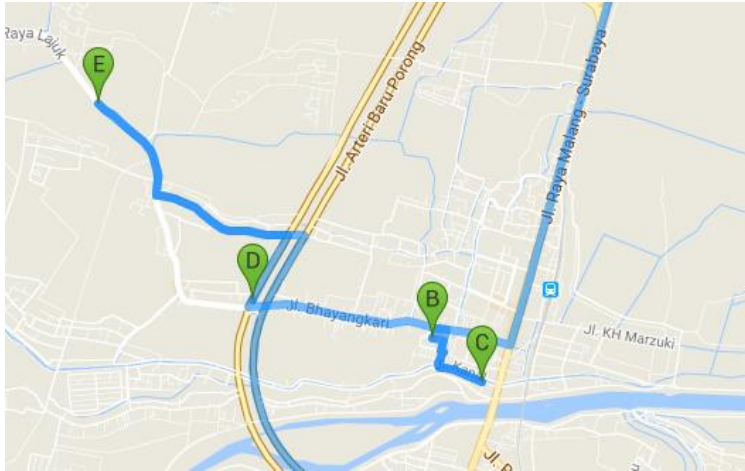
Gambar C.28 Visualisasi Rute Armada28

Gambar C.29 berikut adalah visualisasi rute untuk armada29



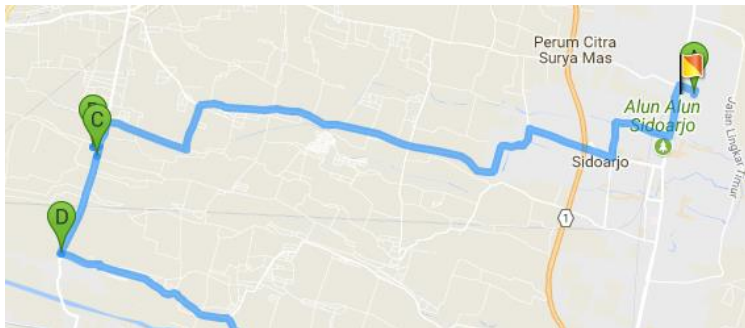
Gambar C.29 Visualisasi Rute Armada29

Gambar C.30 berikut adalah visualisasi rute untuk armada30



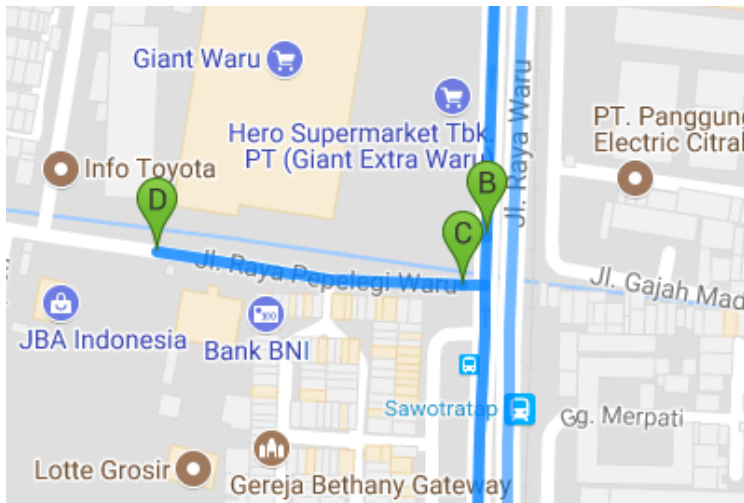
Gambar C.30 Visualisasi Rute Armada30

Gambar C.31 berikut adalah visualisasi rute untuk armada31



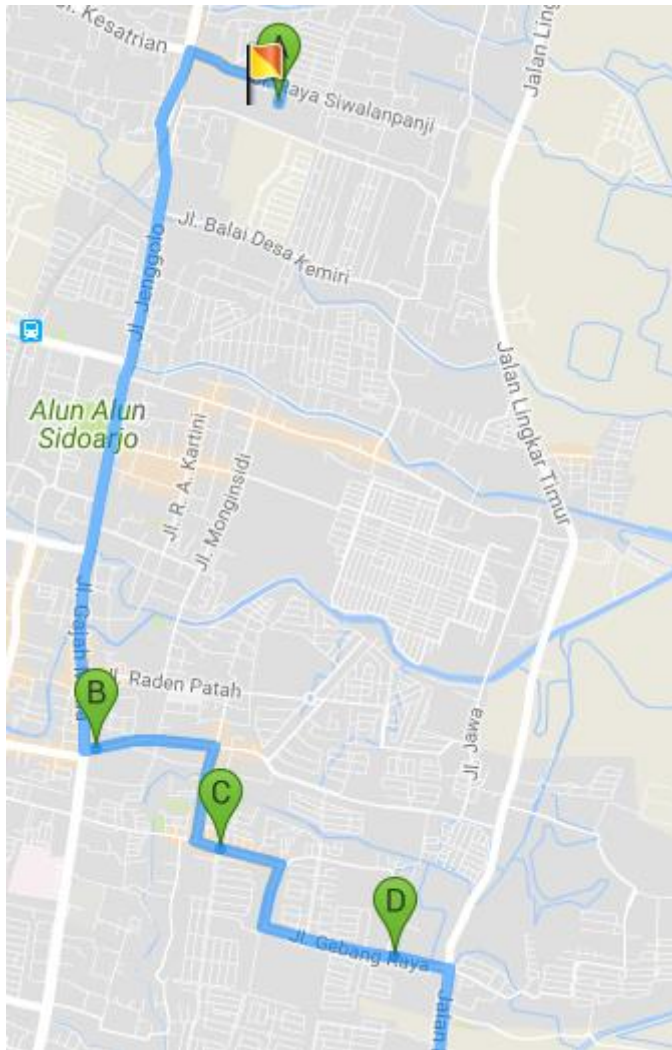
Gambar C.31 Visualisasi Rute Armada31

Gambar C.32 berikut adalah visualisasi rute untuk armada32



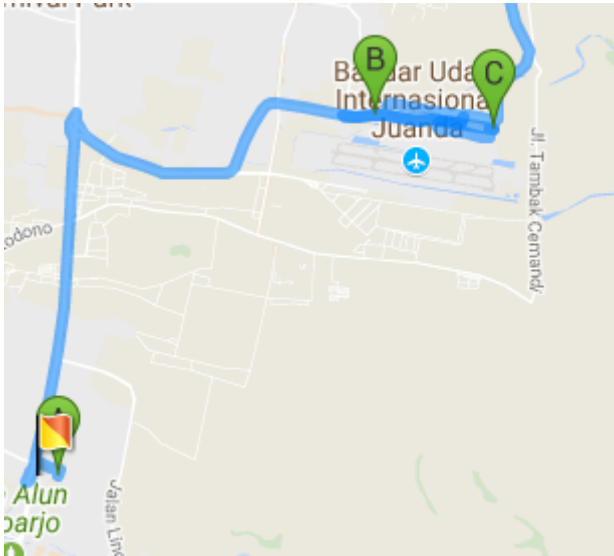
Gambar C.32 Visualisasi Rute Armada32

Gambar C.33 berikut adalah visualisasi rute untuk armada33



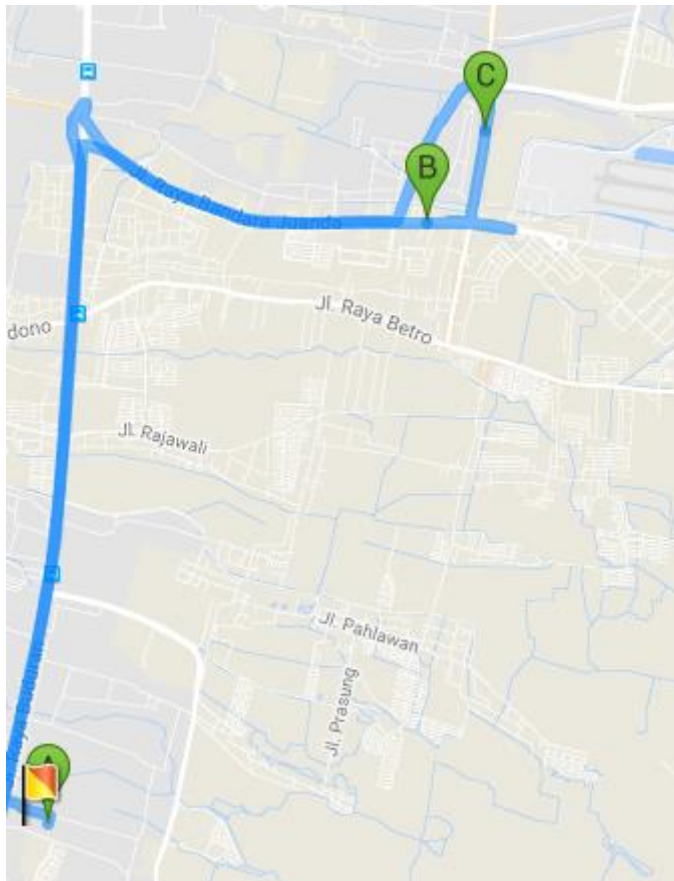
Gambar C.33 Visualisasi Rute Armada33

Gambar C.34 berikut adalah visualisasi rute untuk armada34



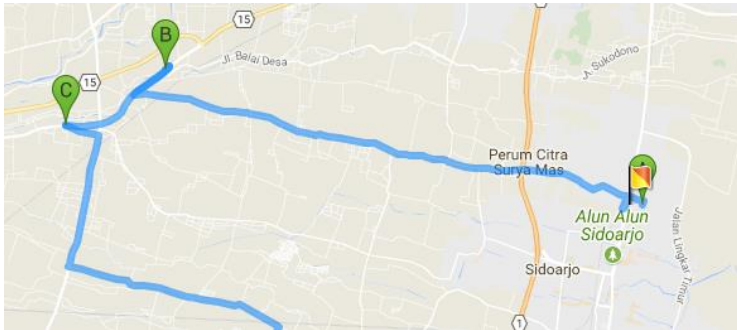
Gambar C.34 Visualisasi Rute Armada34

Gambar C.35 berikut adalah visualisasi rute untuk armada35



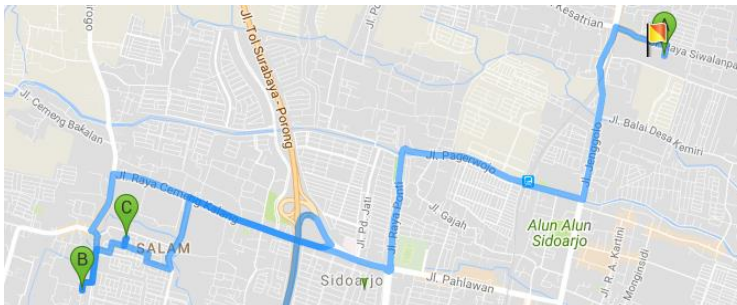
Gambar C.35 Visualisasi Rute Armada35

Gambar C.36 berikut adalah visualisasi rute untuk armada36



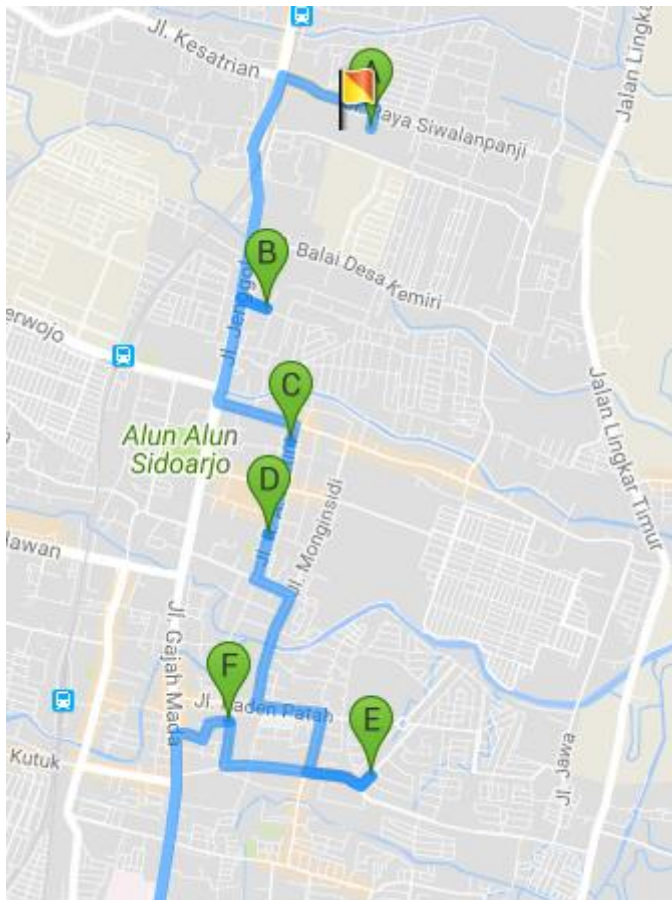
Gambar C.36 Visualisasi Rute Armada36

Gambar C.37 berikut adalah visualisasi rute untuk armada37



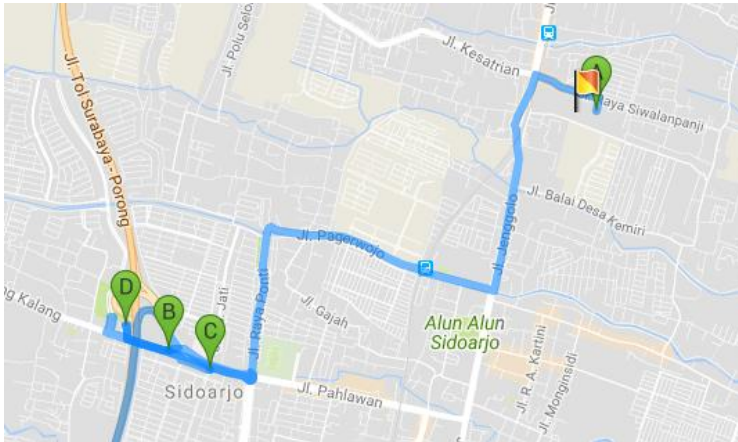
Gambar C.37 Visualisasi Rute Armada37

Gambar C.38 berikut adalah visualisasi rute untuk armada38



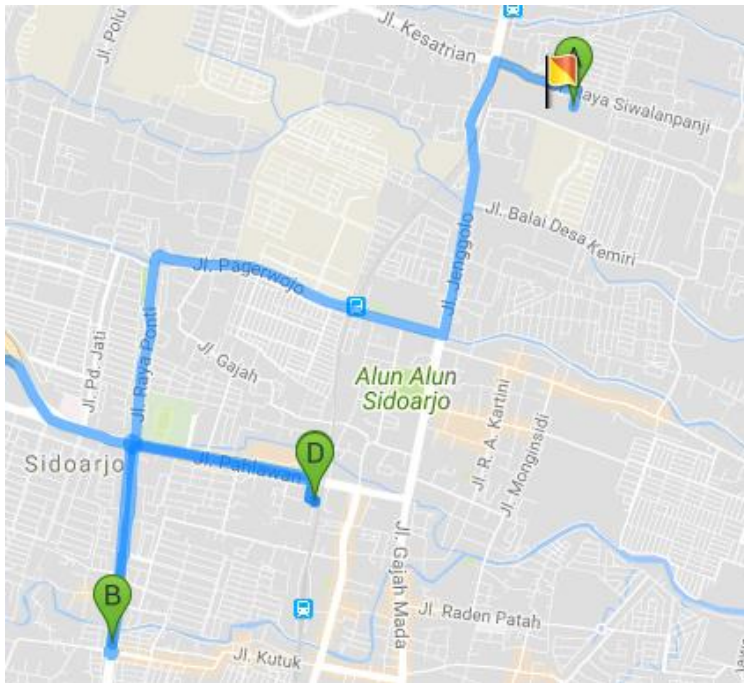
Gambar C.38 Visualisasi Rute Armada38

Gambar C.39 berikut adalah visualisasi rute untuk armada39



Gambar C.39 Visualisasi Rute Armada39

Gambar C.40 berikut adalah visualisasi rute untuk armada40



Gambar C.40 Visualisasi Rute Armada40

BIODATA PENULIS



Penulis bernama lengkap Kirana Gita Larasati, atau biasa dikenal dengan nama panggilan Kikin. Penulis lahir sebagai anak pertama dari tiga bersaudara di Surabaya pada tanggal 17 Juli 1995. Penulis telah menempuh Pendidikan formal di TK Lestari Cilegon, tamat SD dan SMP di yayasan yang sama, yaitu Islam Terpadu Raudhatul Jannah Cilegon, dan kemudian menyelesaikan pendidikan jenjang SMA di SMAN Cahaya Madani Banten Boarding School Pandeglang, dan kemudian masuk ke perguruan tinggi negeri ITS di Departemen Sistem Informasi, Fakultas Teknologi Informasi pada tahun 2013. Penulis memiliki beberapa pengalaman selama berkuliah di ITS diantaranya menjadi panitia salah satu acara terbesar di ITS yaitu ITS Expo dan sebagai anggota yang bergerak di bidang sosial masyarakat BEM FTIf. Penulis juga pernah melakukan kerja praktik sebagai bagian dari tim penelitian sementara selama dua bulan di Pusat Penelitian Informatika (P2I) Lembaga Ilmu Pengetahuan Indonesia (LIPI) yang berlokasi di Bandung.

Pada pengerjaan Tugas Akhir di Departemen Sistem Informasi ITS, penulis mengambil bidang minat Rekayasa Data dan Intelegensi Bisnis dengan topik riset operasi dan sistem pendukung keputusan, yakni mengenai sistem optimasi rute pengangkutan sampah dengan studi kasus yang dilakukan di Kabupaten Sidoarjo. Untuk menghubungi penulis, dapat melalui email: kiranagitalarasati@gmail.com.