

**ALGORITMA SIMULATED ANNEALING UNTUK
MENENTUKAN RUTE TERBAIK PADA DELIVERY
MULTIPLE CLIENT ORDER ONLINE
PADA ARMADA GOJEK**

SKRIPSI

**ANGGA RIZKY. S
0701171001**



**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA
MEDAN
2021**

**ALGORITMA SIMULATED ANNEALING UNTUK
MENENTUKAN RUTE TERBAIK PADA DELIVERY
MULTIPLE CLIENT ORDER ONLINE
PADA ARMADA GOJEK**

SKRIPSI

Diajukan Untuk Memenuhi Syarat Mencapai Gelar Sarjana Komputer

**ANGGA RIZKY. S
0701171001**



**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA
MEDAN
2021**

SURAT PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan di bawah ini,

Nama : ANGGA RIZKY.S
Nomor Induk Mahasiswa : 0701171001
Program Studi : Ilmu Komputer
Judul : **Algoritma simulated annealing untuk menentukan rute terbaik pada delivery multiple client order online pada armada gojek**

Menyatakan bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya. Apabila di kemudian hari ditemukan plagiat dalam skripsi ini maka saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dan sanksi lainnya sesuai dengan peraturan yang berlaku.

Medan, 01 November 2021



ANGGA RIZKY.S
0701171001

PERSETUJUAN SKRIPSI

Hal : Surat Persetujuan Skripsi

Lamp : -

Kepada Yth.,

Dekan Fakultas Sains dan Teknologi

Universitas Islam Negeri Sumatera Utara Medan

Assalamu'alaikum Wr. Wb.

Setelah membaca, meneliti, memberikan petunjuk dan mengoreksi serta mengatakan perbaikan, maka kami selaku pembimbing berpendapat bahwa skripsi saudara,

Nama : ANGGA RIZKY.S

Nomor Induk Mahasiswa : 0701171001

Program Studi : Ilmu Komputer

Judul : Algoritma simulated annealing untuk menentukan rute terbaik pada delivery multiple client order online pada armada gojek

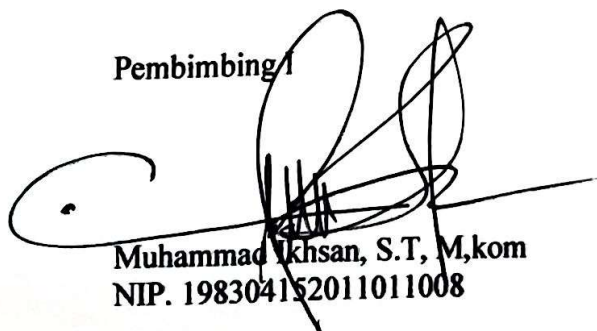
Dapat disetujui untuk segera dimunaqasyahkan. Atas perhatiannya kami ucapkan terimakasih.

Medan, 10 November 2021 M

5 Rabi'ul-Awwal 1443 H

Komisi Pembimbing,

Pembimbing I



Muhammad Ikhsan, S.T, M.kom
NIP. 198304152011011008

Pembimbing II



Armansyah, M.kom
NIB. 1100000074



PENGESAHAN SKRIPSI

Nomor : B.005/ST/ST.V.2/PP.01.1/01/2022

Judul : Algoritma simulated annealing untuk menentukan rute terbaik pada delivery multiple client order online pada armada gojek
Nama : ANGGA RIZKY.S
Nomor Induk Mahasiswa : 0701171001
Program Studi : Ilmu Komputer
Fakultas : Sains dan Teknologi

Telah dipertahankan di hadapan Dewan Penguji Skripsi Program Studi Ilmu Komputer Fakultas Sains dan Teknologi UIN Sumatera Utara Medan dan dinyatakan LULUS.

Pada hari/tanggal : Kamis, 11 Novemver 2021
Tempat : Ruang Sidang Fakultas Sains dan Teknologi

Tim Ujian Munaqasyah
Ketua,

Ilka Zuffria, S.Kom., M.Kom
NIP. 198506042015031006

Dewan Penguji



Penguji I
Muhammad Ikhsan, S.T, M,kom
NIP. 198304152011011008



Penguji III
Sriani, M.kom
NIB. 1100000108

Penguji II

Annansyah, M.kom
NIB. 1100000074

Penguji IV



Muhammad Siddik Hasibuan, M.kom
NIP. 198611152019031008

Mengesahkan,

Dekan Fakultas Sains dan Teknologi
UIN Sumatera Utara Medan



Dr. M. Syahman, MA
NIP. 198609051991031002

ABSTRAK

Berbelanja online bukanlah hal yang asing lagi bagi masyarakat sekarang ini. Kesibukan dan padatnya aktivitas menjadi pendorong seseorang untuk melakukan pemesanan online untuk memenuhi kebutuhannya. Misalnya seorang pekerja yang sudah lelah bekerja seharian melakukan pemesanan makanan melalui online untuk menu makan malam dibanding memasak. Simulated annealing (SA) merupakan algoritma untuk melakukan optimisasi yang bersifat umum. Pada penelitian ini peneliti membangun sistem yang menentukan rute pengiriman terbaik untuk mengantarkan pesanan online pelanggan dengan menggunakan algoritma simulated annealing. Dari hasil penelitian didapat bahwa penerapan algoritma simulated annealing dalam menemukan rute terbaik pada delivery multiple client order online armada gojek di Stabat dengan menggunakan data yang dikumpulkan dari driver armada gojek mampu menemukan rute terbaik pada delivery multiple client order online dengan cukup baik.

Kata kunci : *Multiple Client Order Online, Algoritma Simulated Annealing, Gojek.*

ABSTRACT

Online shopping is not a new thing for today's society. Busyness and busyness of activity motivates someone to make an online order to fulfill their needs. For example, a worker who is tired of working all day orders food online for a dinner menu instead of cooking. Simulated annealing (SA) is an algorithm to perform general optimization. In this study, researchers built a system that determines the best shipping route to deliver customer online orders using a simulated annealing algorithm. From the results of the study, it was found that the application of the simulated annealing algorithm in finding the best route for the delivery of multiple client orders online for the Gojek fleet in Stabat using data collected from the Gojek fleet drivers was able to find the best route for the delivery of multiple client orders online quite well.

Keywords : *Multiple Client Order Online, Simulated Annealing Algorithm, Gojek.*

KATA PENGANTAR

Assalamualaikum warahmatullahi wabarakatu Alhamdulillahirobbil'alamin.

Segala puji & syukur penulis panjatkan kehadiran Allah SWT, lantaran menggunakan izin-Nya lah penulis bisa menuntaskan Skripsi ini. Shalawat bersama salam semoga senantiasa tercurah pada junjungan penulis, yakni Nabi Muhammad SAW.

Skripsi ini buat melengkapi dan memenuhi kondisi buat menjalankan acara mata kuliah yg terdapat pada semester VIII Ilmu Komputer Universitas Islam Negeri Sumatera Utara buat menerima gelar Strata-1. Dalam penyusunan Skripsi ini penulis menyadari adanya keterbatasan, kemampuan, & wawasan penulis, maka penulis mendapat masukan & saran berdasarkan seluruh pihak. Untuk itu penulis membicarakan rasa terima kasih pada :

1. Bapak Prof. Dr. Syahrin Harahap, MA selaku Rektor Universitas Islam Negeri Sumatera Utara Medan.
2. Bapak Dr. Mhd Syahnan, M.A selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara Medan.
3. Bapak Ilka Zufria, M.Kom selaku Ketua Program Studi Ilmu Komputer Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara Medan.
4. Bapak Rakhmat Kurniawan R, S.T., M.Kom selaku Sekretaris Program Studi Ilmu Komputer Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara Medan.
5. Bapak Muhammad Ikhsan, S.T., M.Kom selaku Pembimbing 1 Skripsi Saya yang selalu memberikan bimbingan selama menyelesaikan skripsi ini.
6. Bapak Armansyah., M.Kom selaku Pembimbing 2 Skripsi Saya yang selalu memberikan bimbingan selama menyelesaikan skripsi ini.
7. Kedua orang tua penulis yang tiada hentinya mendoakan di setiap sujud sholatnya, mengingatkan di sepanjang waktu dan memberi motivasi bagi penulis.

8. Seluruh dosen dan pegawai Program Studi Ilmu Komputer maupun di luar Ilmu Komputer yang pernah memberikan ilmunya kepada penulis.
9. Teman-teman seperjuangan kelas Ilmu Komputer 1 angkatan 2017 dan seluruh teman-teman angkatan 2017 yang sedang berjuang untuk menyelesaikan tugas akhir ini.

Dalam penulisan ini banyak kesulitan yang penulis hadapi, namun demikian berkat bimbingan dan bantuan dari berbagai pihak akhirnya penulis dapat menyelesaikan. Demi kelancaran dalam penyelesaian laporan proposal skripsi ini tidak terlepas dari bantuan pihak terutama kepada Ayah dan Ibu beserta keluarga besar saya yang telah memberikan bantuan moril maupun materil, semangat dan do'a yang begitu besar kepada penulis.

Medan, 01 November 2021

Penulis,

Angga Rizky S

DAFTAR ISI

	Halaman
ABSTRAK.....	i
ABSTRACT	ii
KATA PENGANTAR.....	iii
DAFTAR ISI	v
DAFTAR GAMBAR.....	vii
DAFTAR TABEL	ix
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian.....	4
BAB II TINJAUAN PUSTAKA	5
2.1 Optimalisasi	5
2.2 <i>Multiple Client Order</i>	5
2.3 Kecerdasan Buatan	5
2.4 Algoritma	6
2.4.1 Struktur Algoritma.....	7
2.5 Algoritma <i>Simulated Annealing</i>	7
2.6 Graf.....	12
2.6.1 Definisi Graf.....	12
2.6.2 Jenis-jenis Graf.....	12
2.7 <i>Google Maps</i>	15
2.8 <i>Java Script</i>	16
2.9 <i>Visual Studio Code</i>	16
2.10 <i>PHP</i>	17
2.11 Pengertian Basis Data.....	Error! Bookmark not defined.

2.12 Database <i>MySQL</i>	17
2.13 <i>Unified Modeling Language (UML)</i>	18
2.14 Riset Terkait.....	22
BAB III METODE PENELITIAN	25
3.1 Tempat dan Waktu Penelitian.....	25
3.2 Bahan dan Alat Penelitian	25
3.3 Cara Kerja.....	26
3.3.1 Perencanaan	26
3.3.2 Pengumpulan Data.....	27
3.3.3 Analisis Kebutuhan	27
3.3.4 Prosedur Kerja Algoritma <i>Simulated Annealing</i>	28
3.3.5 Perancangan <i>Interface</i>	29
3.3.6 Perancangan Sistem.....	32
3.3.7 Pengujian Sistem.....	37
3.3.8 Penerapan.....	37
BAB IV HASIL DAN PEMBAHASAN.....	39
4.1 Pembahasan	39
4.1.1 Analisis Data	39
4.1.2 Representasi Data	42
4.2 Hasil.....	52
4.2.1 Analisis Data	52
BAB V KESIMPULAN DAN SARAN	58
5.1 Kesimpulan	58
5.2 Saran.....	58
DAFTAR PUSTAKA.....	59

DAFTAR GAMBAR

Gambar	Judul Gambar	Halaman
Gambar 2.1	Kedudukan Kecerdasan Buatan.....	6
Gambar 2.2	Contoh Pengiriman Barang dengan SA	11
Gambar 2.3	Hasil Pengiriman Barang dengan SA	13
Gambar 2.4	Graf Sederhana	15
Gambar 2.5	Graf Tak Sederhana	15
Gambar 2.6	Graf tidak berarah.....	16
Gambar 2.7	Graf berarah.....	16
Gambar 2.8	Graf berbobot	17
Gambar 2.9	Graf tak berbobot.....	17
Gambar 2.10	Visual Studio Code.....	19
Gambar 3.1	Tahap Perencanaan	31
Gambar 3.2	Prosedur Kerja Algoritma Simulated Annealing.....	34
Gambar 3.3	Rancangan Tampilan Login	35
Gambar 3.4	Rancangan Tampilan Dashboard.....	36
Gambar 3.5	Rancangan Tampilan Menu Tentang.....	36
Gambar 3.6	Rancangan Tampilan Menu Algoritma.....	37
Gambar 3.7	Diagram Use Case	38
Gambar 3.8	Diagram Activity Login.....	39
Gambar 3.9	Diagram Activity Dashboard.....	39
Gambar 3.10	Diagram Activity Tentang.....	40
Gambar 3.11	Diagram Activity Algoritma	41
Gambar 3.12	Diagram Sequence Login.....	41

Gambar 3.13 Diagram Sequence Menu Dashboard	42
Gambar 3.14 Diagram Sequence Menu Tentang.....	42
Gambar 3.15 Diagram Sequence Menu Algoritma	43
Gambar 4.1 Tampilan Graph Resto-resto di kota Stabat	47
Gambar 4.2 Tampilan Menu Login	58
Gambar 4.3 Tampilan Menu Dashboard.....	58
Gambar 4.4 Tampilan Menu Tentang	58
Gambar 4.5 Tampilan Menu Tentang	59
Gambar 4.6 Tampilan Add New Data	59
Gambar 4.7 Tampilan Lokasi Tujuan	60
Gambar 4.8 Tampilan View Lokasi.....	60
Gambar 4.9 Hasil Pengujian.....	61

DAFTAR TABEL

Tabel	Judul Tabel	Halaman
Tabel 2.1	Input Custemor Depot	11
Tabel 2.2	Hasil Pemindahan Pertama	12
Tabel 2.3	Hasil Pemindahan Kedua	12
Tabel 2.4	Simbol Diagram Use Case	23
Tabel 2.5	Simbol Diagram Activity	25
Tabel 2.6	Simbol Diagram Sequence	26
Tabel 2.7	Simbol Class Diagram	26
Tabel 3.1	Waktu dan Jadwal Penelitian	27
Tabel 3.2	Definisi Aktor	38
Tabel 3.3	Definisi Use Case	38
Tabel 4.1	Nama-Nama resto / Café di Stabat	43
Tabel 4.2	Simulasi Rute	47
Tabel 4.3	Hasil Pengujian Rute Terbaik Algoritma Simulated Annealing	54

BAB I

PENDAHULUAN

1.1 Latar Belakang

Berbelanja *online* bukanlah hal yang asing lagi bagi masyarakat sekarang ini. Kesibukan dan padatnya aktivitas menjadi pendorong seseorang untuk melakukan pemesanan *online* untuk memenuhi kebutuhannya. Misalnya seorang pekerja yang sudah lelah bekerja seharian melakukan pemesanan makanan melalui *online* untuk menu makan malam dibanding memasak (Bent & Hentenryck, 2006). Hal ini menginspirasi pengusaha armada ojek *online* untuk membuka layanan jasa pengantaran. Untuk memberi pelayanan terbaik terhadap pesanan yang meningkat dibanding dengan jumlah armada yang tersedia, jasa layanan pengantaran dapat dilakukan untuk melayani lebih dari satu pemesan (Dridi et al., 2011). Permasalahan yang timbul dari proses pengiriman jumlah pesanan yang banyak adalah menentukan rute terbaik untuk meminimalkan biaya pengantaran (Fajarwati & Wiwik, 2012). Rute terbaik di pilih agar waktu dan biaya perjalanan yang digunakan rendah. Maka dalam islam ada firman Allah SWT menyebutkan dalam surat Al-Maidah ayat 2 untuk tolong menolong dalam sebuah kebaikan.

يَا أَيُّهَا الَّذِينَ آمَنُوا لَا تَجْلُوا شَعَائِرَ اللَّهِ وَلَا الشُّهُرَ الْحَرَامَ وَلَا الْهَدْيَ وَلَا الْقَلَائِدَ وَلَا آمِينَ الْبَيْتِ الْحَرَامَ يَبْتَغُونَ فَضْلًا مِّن رَّبِّهِمْ وَرِضْوَانًا وَإِذَا حَلَلْتُمْ فَاصْطَادُوا وَلَا يَجْرِمَنَّكُمْ شَنَاٰنُ قَوْمٍ أَن صَدَّقْتُم مِّنَ الْمَسْجِدِ الْحَرَامِ أَن تَعْتَدُوا وَتَعَاوَنُوا عَلَى الْبِرِّ وَالتَّقْوَىٰ وَلَا تَعَاوَنُوا عَلَى الْإِثْمِ وَالْعُدْوَانِ وَاتَّقُوا اللَّهَ إِنَّ اللَّهَ شَدِيدُ الْعِقَابِ

Artinya: Hai orang-orang yang beriman agar tidak melanggar syiarsyiar (kesucian) Allah, 193) Bulan Terlarang (melanggar kehormatan) 194) (mengganggu) hadyu (hewan qurban) bukan 195) dan qalā'id (Hewan Bertanda Kurban), 196) Dan janganlah kamu mengganggu para pengunjung Rumah Suci saat mereka mencari kemurahan hati dan sukacita Tuhan! 197) Jika Anda memiliki Bertahalul (Ihram lengkap), berburu (jika diperlukan). Mereka menjauhkanmu dari masjid suci, jadi mereka tidak pernah membenci orang. Jangan saling tolong menolong dengan kebajikan dan ketaqwaan, dan jangan saling tolong menolong dengan dosa dan permusuhan. Takutlah kepada Allah. Allah tegas dalam hukuman.

Simulated annealing (SA) merupakan algoritma untuk melakukan optimisasi yang bersifat umum. Berdasarkan probabilitas dan mekanika statistik,

algoritma ini dapat digunakan untuk mencari pendekatan terhadap solusi optimum global dari suatu permasalahan. Masalah yang memerlukan pendekatan algoritma ini adalah permasalahan optimisasi kombinasi, dimana ruang pencarian solusi yang ada terlalu luas, sehingga hampir tidak mungkin diperoleh solusi eksak terhadap permasalahan tersebut (Gea, 2014).

Publikasi pertama dari pendekatan ini adalah S. Kirkpatrick, C. D. Oleh Gelatt, dan M.P. Vecchi berlaku untuk desain peralatan komputer yang optimal, tetapi juga untuk salah satu masalah klasik ilmu komputer: masalah penjual keliling. Annealing adalah teknik yang dikenal dalam metalurgi dan digunakan untuk mempelajari proses pembentukan kristal bahan (Rere et al., 2015). Beberapa pemanasan diperlukan untuk membentuk struktur kristal yang sempurna. Kemudian perlahan-lahan kendalikan dan dinginkan bahan tersebut. Pemanasan material pada awal proses annealing memungkinkan atom-atom dalam material untuk bergerak bebas, mengingat tingkat energi yang sangat tinggi dalam keadaan panas ini (Kallehauge B. & OBG, 2001).

Sebagai hasil dari proses pendinginan yang lambat ini, atom yang sebelumnya bergerak bebas akhirnya menemukan lokasi optimal yang meminimalkan energi internal yang dibutuhkan atom untuk mempertahankan posisinya. Simulated annealing (SA) bekerja seperti proses annealing. Pada awal proses, solusi awal dipilih yang mencerminkan keadaan material sebelum proses dimulai. Pergerakan bebas atom dalam materi dinyatakan dalam bentuk modifikasi solusi awal/ sementara (Firdaus et al., 2015). Menyetel parameter suhu tinggi (T) di awal proses SA memberi Anda kebebasan untuk memodifikasi solusi sementara yang ada. Derajat kebebasan ini diukur relatif terhadap nilai fungsi tertentu yang mengevaluasi seberapa optimal solusi sementara yang diperoleh. Jika nilai fungsi skor yang dimodifikasi ditingkatkan (dalam kasus masalah optimasi yang mencoba mencari minimum, nilainya akan lebih kecil), solusi yang dimodifikasi akan digunakan sebagai solusi berikutnya. Jika nilai merit function perubahan ini memburuk saat suhu annealing masih tinggi (Noviardiando, G et al., 2019).

Larutan inferior ini masih dapat diterima, tetapi jika suhu annealing relatif rendah, larutan modifikasi inferior ini mungkin tidak dapat diterima. Pada tahap berikutnya, seiring dengan penurunan suhu secara bertahap, kemungkinan kecil untuk menerima langkah perubahan yang tidak meningkatkan nilai fungsi merit. Artinya kebebasan untuk mengubah solusi menjadi semakin sempit hingga akhirnya kita ingin mendapatkan solusi yang mendekati solusi optimal (Pangestuti, R, 2020).

Berdasarkan kemampuan algoritma *simulasi annealing* untuk menemukan solusi terbaik dari berbagai macam kombinasi kasus dan masalah yang dihasilkan dari proses pelayanan antara banyak pesanan pelanggan berjudul **ALGORITMA SIMULATED ANNEALING UNTUK MENENTUKAN RUTE TERBAIK PADA DELIVERY MULTIPLE CLIENT ORDER ONLINE PADA ARMADA GOJEK** untuk mendapatkan solusi rute pengiriman pesanan terbaik untuk mengurangi biaya dan waktu pengiriman.

1.2 Rumusan Masalah

Adapun rumusan masalah dalam skripsi ini adalah sebagai berikut:

1. Bagaimana menentukan rute pengiriman terbaik untuk mengantarkan pesanan online pelanggan dengan menggunakan algoritma *simulated annealing*?
2. Bagaimana mengimplementasikan program yang dibangun terhadap penentuan rute pengiriman terbaik untuk mengantarkan pesanan online pelanggan dengan menggunakan algoritma *simulated annealing* pada Kota Stabat Kabupaten Langkat?

1.3 Batasan Masalah

Adapun batasan masalah untuk tugas akhir ini adalah sebagai berikut:

1. *Simulated annealing* sebagai algoritma penentu solusi optimalisasi rute terbaik.
2. Penelitian ini dilakukan di Kota Stabat Kabupaten Langkat, sehingga rute yang digunakan ialah Kota Stabat saja.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini antara lain:

1. Agar dapat menentukan rute pengiriman terbaik untuk mengantarkan pesanan online pelanggan dengan menggunakan algoritma *simulated annealing*.
2. Agar program yang dibangun terhadap penentuan rute pengiriman terbaik untuk mengantarkan pesanan online pelanggan dengan menggunakan algoritma *simulated annealing* pada Kota Stabat Kabupaten Langkat dapat diimplementasikan dengan baik.

1.5 Manfaat Penelitian

Manfaat yang diperoleh dalam penelitian ini adalah sebagai berikut:

1. Sebagai sarana yang dapat membantu driver gojek dalam menentukan rute pengiriman terbaik untuk mengantarkan pesanan *online* pelanggan dengan menggunakan algoritma *simulated annealing*.
2. Mengetahui jarak yang cepat dan tidak ada hambatan dalam mengantar pesanan *online* pada Kota Stabat Kabupaten Langkat.

BAB II

TINJAUAN PUSTAKA

2.1 Optimalisasi

Optimasi adalah suatu cara untuk menyatakan hasil solusi terbaik dari suatu model matematika masalah (Santoso, 2016). Hal ini dapat diartikan bahwa bisnis beroperasi untuk memaksimalkan keuntungan dan efisiensi dan meminimalkan kerugian, biaya, atau risiko. Secara umum, keinginan untuk memecahkan masalah model optimasi dapat digunakan di hampir semua area aplikasi.

Model optimasi telah digunakan selama berabad-abad. Hari ini, menjadi sangat terlihat ketika Anda ingin membuat perusahaan yang sangat besar dan kompleks. Para peneliti menjadi semakin ambisius. Dalam banyak kasus, tidak mungkin lagi membuat keputusan tanpa model optimasi. Misalnya, dalam kemitraan multinasional yang besar, persentase kecil dari peningkatan operasional dapat menambah jutaan dolar untuk peningkatan laba. Untuk model yang lebih besar, semua kerumitan terlibat, jadi jika Anda tidak dapat menyelesaikannya, itu bisa menjadi masalah kecil. Selama beberapa dekade terakhir, ada perkembangan luar biasa dalam perangkat keras dan perangkat lunak komputer, dan kemajuan ini telah menjadikan model pengoptimalan alat yang populer dalam bisnis, akademisi, dan bisnis.

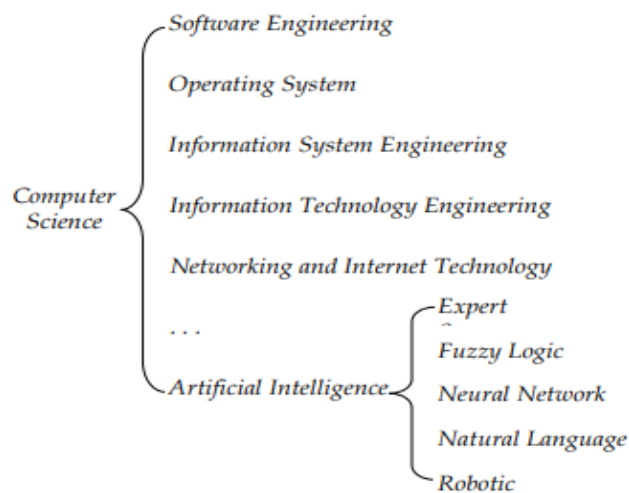
2.2 *Multiple Client Order*

Multiple client order merupakan persoalan optimasi yang memiliki lebih dari satu fungsi tujuan pemesanan, yang memungkinkan terjadi konflik antar fungsi tujuan tersebut (Santosa & Willy, 2011). Klasifikasi teknik yang populer di *Operations Research* yang fokus untuk mengatasi.

2.3 Kecerdasan Buatan

Kecerdasan buatan adalah bidang ilmu komputer yang secara khusus ditujukan untuk mengembangkan perangkat lunak dan perangkat keras yang

sepenuhnya dapat meniru beberapa fungsi otak manusia, atau bidang ilmu komputer yang mempelajari otomatisasi gerakan cerdas.



Gambar 2.1 Kedudukan Kecerdasan Buatan

2.4 Algoritma

Algoritma adalah sebuah metode dalam mengurutkan langkah-langkah yang bersifat logis dalam menyelesaikan suatu masalah dan tersusun secara logis dan sistematis. Kata-kata logis adalah kata kunci Algoritma. Algoritma merupakan metode efektif dalam penyelesaian masalah yang kompleks pada dunia informatika. Algoritma juga merupakan kumpulan dari beberapa perintah dalam menyelesaikan masalah. Perintah-perintah tersebut akan diterjemahkan secara bertahap. Semua masalah yang ada harus memiliki kriteria yang berbeda dalam semua kondisi awal yang harus ada, agar ketika menjalankan proses dari algoritma lebih mudah dalam penyelesaiannya.

Algoritma kriptografi akan melakukan perulangan hingga sebuah masalah terselesaikan dengan baik, apabila belum selesai atau dalam katagori jauh maka algoritma terus mengulangi hingga mendekati atau masalah tersebut terpecahkan (Maulana, 2017). Algoritma akan dipelajari di setiap cabang ilmu secara jelas, sehingga dapat digabungkan ke dalam sistem komputer yang dibangun dan bahasa pemrograman yang digunakan. Selain itu beberapa masalah bisa diselesaikan dengan algoritma yang berbeda dengan catatan harus memiliki kriteria yang sama

dari setiap algoritma tersebut. Ukuran dari kompleksitas algoritma adalah seberapa banyak komputasi yang akan diaplikasikan dalam setiap algoritma tersebut, untuk menyelesaikan sebuah masalah. Selain itu masalah yang diselesaikan dengan waktu yang singkat merupakan tolak ukur keberhasilan dan ketepatan dari sebuah algoritma. Semakin lama penyelesaian masalah maka semakin buruk algoritma dan begitu juga sebaliknya.

2.4.1 Struktur Algoritma

Sebagai ilmu atau seni dalam memecahkan sebuah permasalahan yang kompleks algoritma memiliki struktur – struktur yang mendasar yaitu sebagai berikut:

1. Sekuensial (runtunan)

Langkah - langkah yang dilakukan pada struktur sekuensial ini dalam menyelesaikan masalah adalah melakukan proses algoritma secara berurutan.

2. Struktur seleksi

Langkah struktur seleksi memilih pesan berdasarkan sebuah kondisi dari masalah tersebut dalam menentukan sebuah kondisi berdasarkan pengambilan keputusan. Banyak contoh yang dapat kita terapkan pada struktur jenis ini jika itu menyangkut keputusan, diantaranya: nilai siswa berdasarkan kemampuan pengetahuan dan keterampilan.

3. Struktur perulangan

Dalam struktur perulangan intruksi dapat dilakukan dengan beberapa kali agar masalah tersebut dapat selesai dengan baik.

2.5 Algoritma *Simulated Annealing*

Simulated Annealing adalah suatu metode optimasi yang berdasarkan pada proses pendinginan yang digunakan dalam Metalurgi (Nurhanivah & Widita, 2018). Umumnya, ketika suatu zat didinginkan, pertama-tama dipanaskan sampai titik leleh di mana ia meleleh, dan kemudian didinginkan secara perlahan dengan mengendalikan proses pendinginan sampai zat itu membeku kembali. Sifat akhir dari bahan ini sangat tergantung pada rencana pendinginan yang digunakan. Jika

suhu pendinginan turun tajam, produk rentan terhadap kerusakan karena ketidaksempurnaan struktur.

Sebaliknya, menurunkan suhu pendinginan secara perlahan akan memperbaiki struktur dan meningkatkan kekuatan.

Kelebihan *simulated annealing* dibandingkan dengan metode lainnya adalah kemampuannya untuk menghindari jebakan optimal local (Hidayati et al., 2009). Algoritmanya merupakan algoritma pencarian acak, tetapi tidak hanya menerima nilai obyektif yang selalu turun, melainkan terkadang menerima nilai obyektif yang naik juga. Dalam algoritma *simulated annealing* itu, suatu state (kombinasi dari satu solusi) dapat diterima dalam persamaan 2.1, seperti dibawah ini:

$$P(E) = e^{-\frac{\Delta E}{kT}} \dots\dots\dots(2.1)$$

Dimana :

ΔE = Selisih energi saat ini dan energi sebelumnya

K = Konstanta Boltzmann

T = Temperatur

Saat suhu perlahan menurun, kemungkinan menerima konfigurasi eksperimental dengan fungsi tujuan yang lebih tinggi berkurang. Suhu awal dipilih lebih tinggi dari maksimum yang dihitung secara acak dari fungsi tujuan. Suhu berhenti ditentukan dengan memantau fungsi obyektif sebagai fungsi perubahan suhu. Perbedaan fungsi obyektif antara temperatur yang berbeda harus berhenti ketika sistem mencapai optimum. Fungsi obyektif diturunkan untuk mengoptimalkan orientasi file efek saat menghitung distribusi dosis (Nurhanivah & Widita, 2018). Hal ini dapat dikonfirmasi dengan Persamaan 2.2 di bawah ini.:

$$F = \iint [D_i(x,y) - D_i^D(x,y)]^2 dx dy = \min! \dots\dots\dots(2.2)$$

Dimana :

D_i = Dosis yang diterima pada voxel ke i

D_i^D = Dosis yang direncanakan pada saat TPS

Selanjutnya dengan menggunakan transformasi Fourier diskrit dan memasukkan beberapa syarat batas dapat diperoleh fungsi objektif sebagai persamaan 2.3, seperti berikut:

$$F = \iint [D_i(w, \theta) - D_i^D(w, \theta)]^2 dw d\theta = \min! \dots \dots \dots (2.3)$$

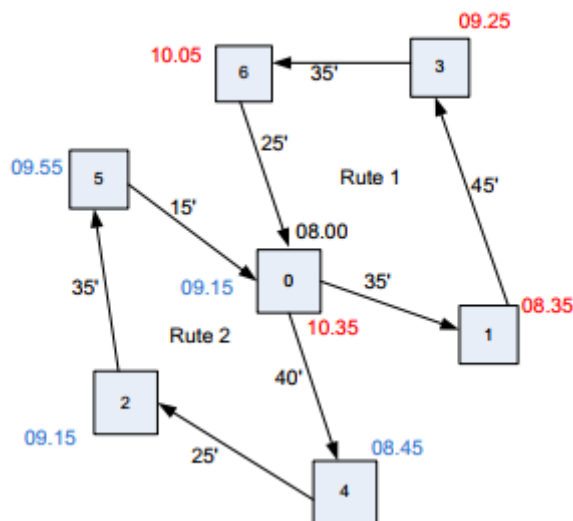
Dan dapat disederhanakan menjadi persamaan 2.4, seperti dibawah ini:

$$F = \sum_i (D_i - D_i^D)^2 \dots \dots \dots (2.4)$$

Dimana :

W_i = faktor bobot berkas radiasi.

Dalam penerapan algoritma *simulated annealing* dalam menyelesaikan multi depot *vehicle routing problem* dengan variabel *travel time* (Sinaga, 2015), seperti gambar dibawah ini:



Gambar 2.2 Contoh Pengiriman Barang dengan SA

Tabel 2.1 Input Custemor Depot

Dari node	Ke node	Travel time
3	0	25
2	6	12
6	5	15
6	5	10
5	6	10

Langkah pertama, kita harus memastikan apakah rute baru memenuhi *capacity constraint* untuk kedua kemungkinan pemindahan, maka perhitungannya didapatkan:

$$q1\text{-new} = q1\text{-old} - q5 = 57 - 25 = 32 \text{ galon}$$

$$q2\text{-new} = q2\text{-old} + q5 = 56 + 25 = 81 \text{ galon}$$

Karena kapasitas kendaraan (100 galon) tidak terlampaui, maka *insertion* dapat dilakukan dengan memindahkan node 5 ke tour 2.

Langkah kedua, memastikan apakah pada tour baru *time window constraint* masing-masing node yang dikunjungi terpenuhi.

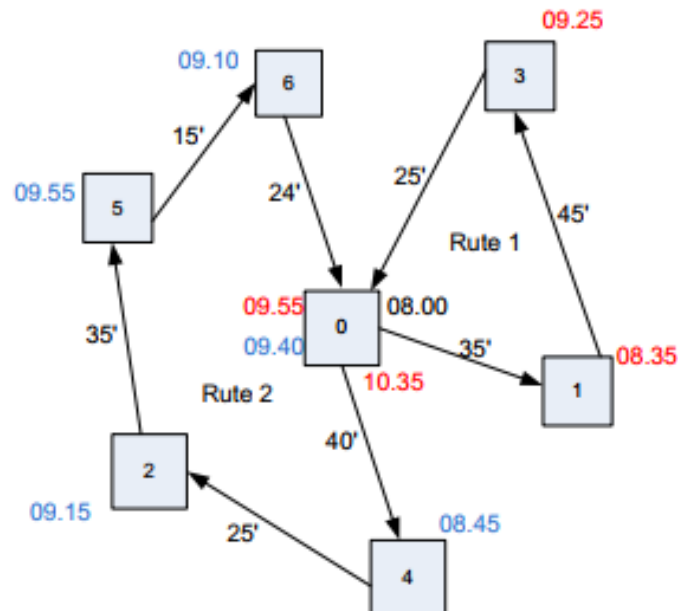
Tabel 2.2 Hasil Pemindahan Pertama

Rute 1	0	1	3	0
Arrival	08.00	08.35	09.25	09.55
Rute 2	0	4	2	5
Arrival	08.00	08.45	09.15	09.55

Tabel 2.3 Hasil Pemindahan Kedua

Rute 1	0	1	3	0
Arrival	08.00	08.35	09.25	09.55
Rute 2	0	4	2	6
Arrival	08.00	08.45	09.15	09.32

Berdasarkan tabel diatas terdapat beberapa pemindahan rute dari rute pertama hingga rute kedua dengan penerapan *simulated annealing*. Terdapat pada gambar berikut:



Gambar 2.3 Hasil Pengiriman Barang dengan SA

Berikut *pseudocode* algoritma *simulated annealing* adalah :

T_0 (T and a lowercase 0) Starting temperature

Iter Number of iterations

λ The cooling rate

1. Set $T = T_0$ (T and a lowercase 0)
 2. Let x = a random solution
 3. For $i = 0$ to $\text{Iter}-1$
 4. Let f = fitness of x
 5. Make a small change to x to make x'
 6. Let f' = fitness of new point
 7. If f' is worse than f then
 8. Let $p = \text{PR}(f', f, T_i)$ (T with a lowercase i)
 9. If $p > \text{UR}(0,1)$ then
 10. Undo change (x and f)
 11. Else
 12. Let $x = x'$
 13. End if
 14. Let $T_i(T \text{ with a lowercase } i) + 1 = \lambda T_i$ (λ and T with a lowercase i)
 15. End for
- Output: The solution x

2.6.2 Istilah dalam Algoritma *Simulated Annealing*

Penerapan *Simulated Annealing* dalam permasalahan optimalisasi menggunakan beberapa istilah sebagai berikut.

1. Simulasi termodinamika adalah masalah optimasi yang dilakukan oleh algoritma simulasi annealing.
2. Status sistem adalah solusi yang layak yang dipilih berdasarkan perubahan status.
3. Energi adalah suatu usaha atau tujuan untuk mencari solusi dari masalah optimasi seperti Makespan, delay, dll.
4. Perubahan status adalah perpindahan untuk menggunakan solusi swap dan insert.
5. Temperatur adalah parameter yang digunakan untuk mengontrol masalah optimasi. Tanggal.
6. Status beku adalah solusi akhir dimana solusi tersebut optimal atau hampir optimal.

2.6 Graf

Dalam menentukan lintasan terbaik dipenelitian ini diperlukan graf sebagai representasi lokasi-lokasi yang tersedia dalam aplikasi yang akan dibuat. Graf merupakan pokok bahasan yang sudah tua dan lama dalam penggunaannya namun memiliki banyak terapannya sampai saat ini.

2.6.1 Definisi Graf

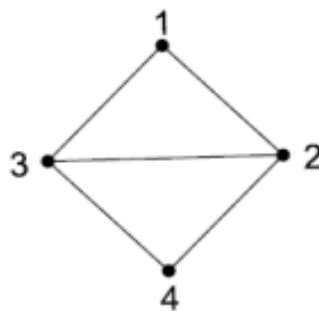
Secara matematis, Graf (G) didefinisikan sebagai pasangan himpunan (V,E), ditulis dengan notasi $G = (V,E)$ dimana V ialah simpul atau titik (bukan himpunan kosong) dan E ialah sisi atau garis penghubung (Prasatria, 2017).

2.6.2 Jenis-jenis Graf

Graf secara umum dapat dibedakan menjadi dua jenis, berdasarkan ada tidaknya loop atau busur ganda pada graf tersebut:

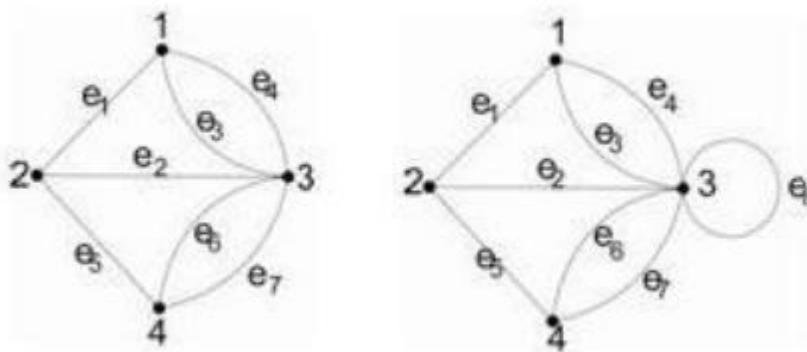
1. Graf sederhana (*simple graph*) yaitu yang tidak mengandung gelang maupun sisi ganda. Pada graf sederhana, sisi adalah pasangan tak-terurut (*undirected*

pairs). Oleh karena itu, deskripsi halaman (u, v) sama dengan (v, u) . Graf juga dapat dikatakan sederhana dengan $G = (V, E)$ terdiri dari himpunan tidak kosong simpul-simpul dan E adalah himpunan pasangan tak berurut yang berbeda disebut dengan sisi, graf sederhana dapat dilihat pada gambar dibawah ini:



Gambar 2.4 Graf Sederhana

2. Graf tidak sederhana (*unsimple graph*) yaitu graf yang mengandung sisi ganda atau gelang dinamakan graf tidak-sederhana (*unsimple graph*). Ada dua macam tidak sederhana, yaitu graf ganda dan graf semu. Sisi-sisi dari graf dapat berada di sembarang arah. Gambar graf tak sederhana dapat dilihat pada gambar dibawah:

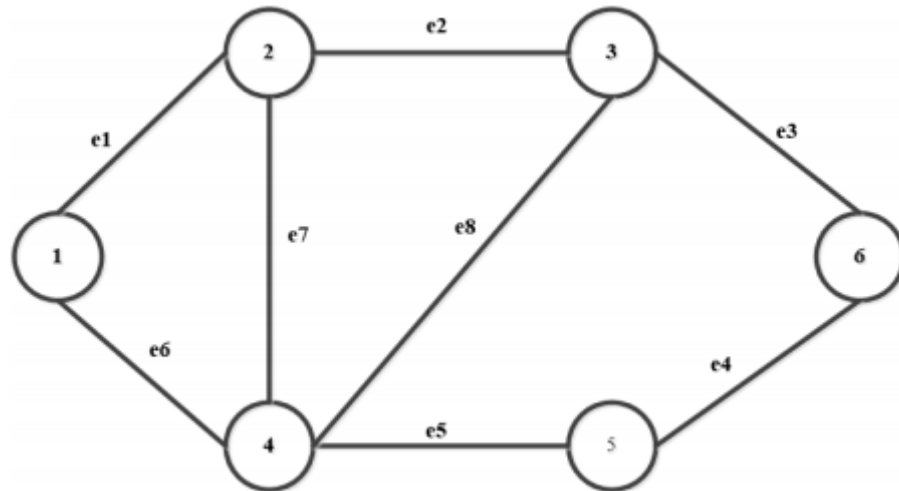


Gambar 2.5 Graf Tak Sederhana

Menurut orientasi arah pada sisinya, graf dibagi menjadi dua jenis, yaitu:

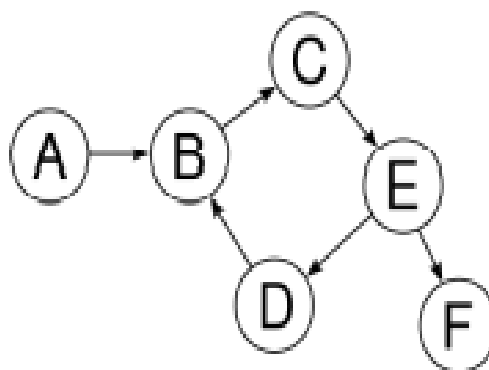
1. Graf tidak berarah (*undirect graph*) adalah graf yang sisinya tidak mempunyai orientasi arah, pada graf ini, urutan pasangan simpulnya dihubungkan oleh sisi

dan tidak diperhatikan. Sebagai contoh graf tidak berarah dapat dilihat pada gambar dibawah:



Gambar 2.6 Graf tidak berarah

2. Graf berarah adalah graf yang setiap sisinya diberi arah. Kami lebih suka menyebut sisi yang berlawanan sebagai busur (arc). Pada graf berarah, (u, v) dan (v, u) menunjukkan dua buah busur yang berbeda. Untuk busur (u, v) , simpul u diberi nama simpul asal (initial vertex) dan simpul v diberi nama simpul terminal (terminal vertex). Contoh graf berarah ditunjukkan pada gambar dibawah ini:

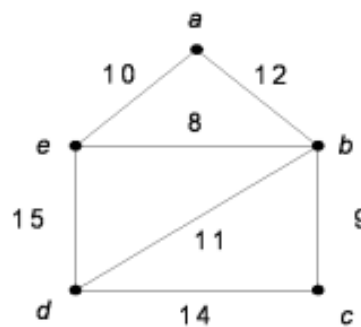


Gambar 2.7 Graf berarah

Selain dapat mempunyai orientasi arah, sisi pada graf juga dapat membedakan graf berdasarkan bobotnya menjadi 2 jenis (Ginting, G, F, 2019):

1. Graf Berbobot (*weighted graph*)

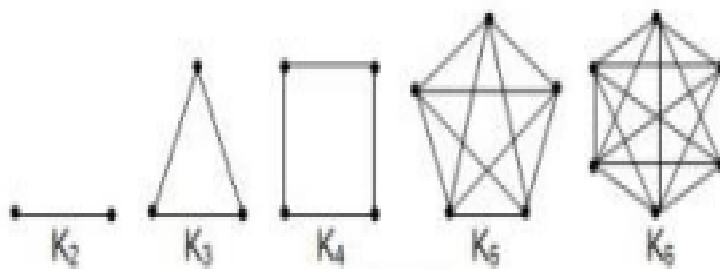
Graf berbobot adalah graf yang setiap sisinya diberi nilai (bobot). Bobot pada tiap sisi dapat berbeda-beda bergantung pada kasus yang dimodelkan dengan graf. Bobot dapat menyatakan jarak antara dua buah kota, biaya perjalanan antara dua buah kota, Penerusan pesan dari satu simpul komunikasi ke simpul lain (dalam jaringan komputer), biaya produksi, ongkos produksi, dan sebagainya. Contoh graf berbobot ditunjukkan pada gambar dibawah ini:



Gambar 2.8 Graf berbobot

2. Graf Tak Berbobot (*unweighted graph*)

Graf Tak Berbobot berarti semua jenis graf yang tidak memiliki harga pada setiap sisinya. Contoh graf tak berbobot ditunjukkan pada gambar dibawah ini:



Gambar 2.9 Graf tak berbobot

2.7 Google Maps

Google Maps adalah layanan gratis dari *Google* dan sangat populer. *Google Maps* adalah peta dunia yang kami gunakan Lihat daerah. Dengan kata lain, *Google Maps* dapat melihatnya di *browser*. Dapat menambahkan fungsi *Google Maps*

untuk web atau blog berbayar yang kami buat atau dapat menggunakannya secara gratis menggunakan *Google Maps API*. *Google Maps API* Pustaka *JavaScript* (Rismayani & Ardimansyah, 2015).

Hanya memerlukan layanan untuk dapat mengakses data dari *Google Maps* Pemetaan web gratis dari google dengan kapasitas Banyak layanan pemetaan web. *Google Maps* juga memiliki properti sisi server. Artinya, pengguna dapat menggunakan peta yang disimpan di *server Google*.

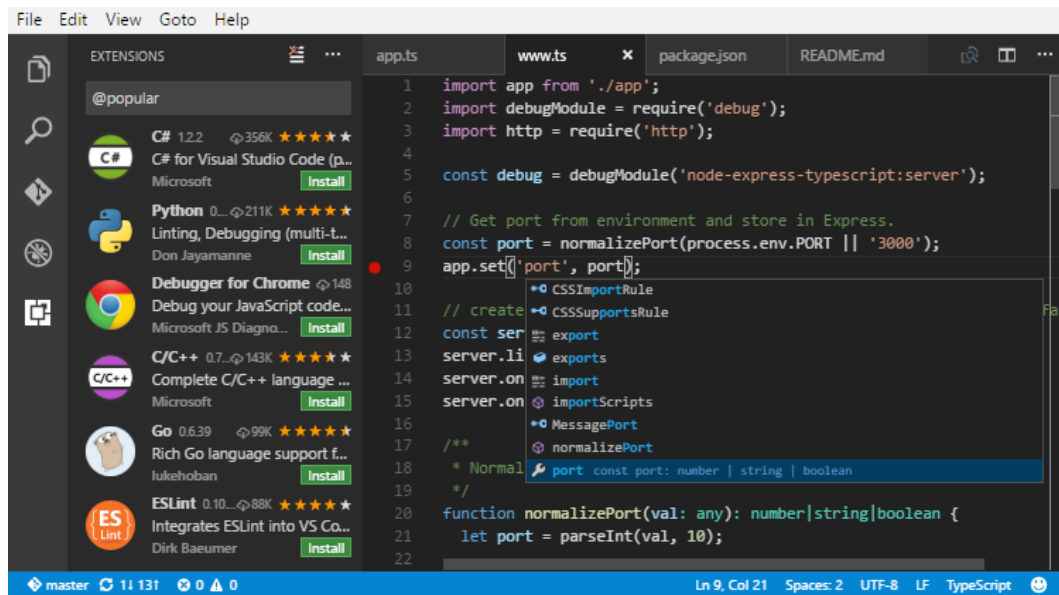
2.8 Java Script

Menurut Irawan (2012) “*Java Script* merupakan bahasa pemrograman web yang berbasis *script*. Dapat menggunakan Javascript dalam pemrograman web untuk membuat web dengan tampilan dan nuansa serta kinerja yang dinamis dan interaktif.

Menurut Abdul Kadir dan Terra (2013), “*JavaScript* adalah bahasa pemrograman yang biasanya ditempatkan dengan kode HTML untuk menentukan tindakan.” Menurut Priyanto dan Jauhari (2014), “*JavaScript* adalah bahasa scripting yang digunakan sebagai fitur saat membuat Web. Menggunakan *javascript* dalam pemrograman web dapat menciptakan sebuah web dengan tampilan dan kinerja yang dinamis dan interaktif”. Menurut Abdul kadir dan Terra (2013) “*Java Script* adalah bahasa pemrograman yang biasa diletakkan bersama kode HTML untuk menentukan suatu tindakan”.

2.9 Visual Studio Code

Visual Studio Code (VS Code) adalah editor kode sumber terbuka gratis untuk pengembangan dan *debugging* aplikasi *cloud* dan *web modern* yang tersedia secara gratis di Linux, OS X dan Windows. Visual Studio Code mendukung lebih dari 30 bahasa pemrograman, markup, dan basis data yang berbeda, beberapa di antaranya *JavaScript*, *C #*, *C ++*, *PHP*, *Java*, *HTML*, *R*, *CSS*, *SQL*, *Markdown*, *TypeScript*, *Less*, *Sass*, *JSON*, *XML* dan *Python*. (Microsoft.com)



Gambar 2.10 Visual Studio Code

2.10 PHP

PHP adalah bahasa pemrograman yang berjalan dalam sebuah *webs-server* (*serverside*). PHP dibuat oleh seorang programmer Unix dan Perl bernama Rasmus Lerdoft dari bulan Agustus sampai September 1994. Script PHP adalah bahasa pemrograman yang berjalan di web server atau di sisi server. Oleh karena itu, PHP dapat melakukan apa saja yang bisa dilakukan program CGI lain, yaitu semua jenis dapat membuat halaman web dinamis, menerima dan membuat cookie, dan bahkan PHP dapat melakukan lebih banyak lagi.

2.11 Database MySQL

MySQL adalah DBMS yang didistribusikan secara gratis di bawah lisensi General Public License (GPL) dan bebas digunakan oleh siapa saja, tetapi tidak sebagai program induk eksplisit (komersial). MySQL sebenarnya berasal dari salah satu konsep terpenting dari baris untuk waktu yang lama: SQL (Structured Query Language). SQL merupakan konsep operasi basis data, terutama untuk proses pemilihan, pemasukan, pengubahan, dan penghapusan data, yang dapat dilakukan dengan mudah dan otomatis.” (Syahputra, 2020)

2.12 *Unified Modeling Language (UML)*

1. Pengenalan UML

Unified Modeling Language (UML) adalah alat untuk memvisualisasikan dan mendokumentasikan analisis dan hasil desain, termasuk sintaks pemodelan visual sistem." (Haviluddin, 2011) Banyak orang mengembangkan teknik pemrograman pada saat itu. Saya membuat bahasa pemodelan untuk pengembangan perangkat lunak menurut Misalnya, diagram aliran data (DFD) dikembangkan dan digunakan oleh banyak pemangku kepentingan untuk memodelkan perangkat lunak dalam pemrograman prosedural atau terstruktur. Transisi keadaan (STD) digunakan untuk pemodelan waktu nyata.

Selama pengembangan teknologi pemrograman berorientasi objek, bahasa pemodelan standar untuk pengembangan perangkat lunak telah dibuat. Itu dibuat menggunakan *Unified Modeling Language (UML)*, sebuah teknologi pemrograman berorientasi objek.

2. *Diagram Use Case*

Diagram yang menunjukkan aktor, kasus penggunaan, dan hubungan mereka sebagai serangkaian tindakan yang memberikan nilai terukur kepada aktor. "Use case direpresentasikan sebagai elips horizontal dalam diagram use case UML." (Haviluddin, 2011)

Tabel 2.4 Simbol *Diagram Use Case*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Sumber: (Syahputra, 2020)

3. Diagram Activity

Diagram aktivitas atau diagram aktivitas menggambarkan alur kerja atau aktivitas dari sistem atau proses bisnis atau menu dalam perangkat lunak. Perhatikan bahwa diagram aktivitas menunjukkan aktivitas sistem, bukan apa yang dilakukan aktor, yaitu aktivitas yang dapat dilakukan sistem.

Tabel 2.5 Simbol *Diagram Activity*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi

3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Sumber: (Syahputra, 2020)

4. Diagram Sequence

Diagram urutan menggambarkan bagaimana suatu objek berperilaku dalam kasus penggunaan dengan menjelaskan masa pakai objek dan pesan yang dikirim dan diterima di antara objek. Oleh karena itu, untuk menggambar diagram urutan, Anda perlu mengetahui objek yang terkandung dalam use case dan metode kelas yang diinstansiasi oleh objek ini. Anda juga perlu membuat diagram urutan untuk melihat skenario use case.

Tabel 2.6 Simbol *Diagram Sequence*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Sumber: (Syahputra, 2020)

5. Class Diagram

Class diagram menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut, operasi dan hubungan antara kelas. Class diagram membantu dalam memvisualisasikan struktur kelaskelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Selama tahap desain, class diagram berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat.

Tabel 2.7 Simbol *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi
2		<i>dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung padanya
3		<i>extend</i>	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan.

Sumber: (Syahputra, 2020)

2.13 Riset Terkait

Ada beberapa riset yang terkait dengan penelitian ini, beberapa di antaranya yaitu riset dari L.M. Rasdi Rere, Mohamad Ivan Fanany, Aniati Murni Arymurthy (2015) yang berjudul “*Simulated Annealing Algorithm for Deep Learning*”. Penelitian ini meneliti tentang algoritma *Simulated Annealing* untuk *deep learning*. *Deep learning* adalah area penelitian baru dalam pembelajaran mesin, di mana tujuannya membawa lebih dekat ke tujuan kecerdasan buatan (Rere et al., 2015).

Metode ini dapat mempelajari banyak level abstraksi dan representasi untuk menciptakan kesamaan rasa data seperti teks, suara dan gambar. Meskipun *deep learning* berguna untuk berbagai tugas, sulit untuk dilatih. beberapa metode dalam melatih *deep learning* agar optimal telah diusulkan. Dalam penelitian ini, penulis mengusulkan *Simulated Annealing* (SA) untuk meningkatkan kinerja *Convolution Neural Network* (CNN), sebagai pendekatan alternatif untuk *deep learning* optimal menggunakan teknik optimasi modern, yaitu algoritma *metaheuristik*. Dataset MNIST digunakan untuk memastikan akurasi dan efisiensi metode yang diusulkan. Selain itu, peneliti juga membandingkan metode yang diusulkan dengan yang asli dari CNN. Meskipun ada peningkatan waktu komputasi, hasil percobaan menunjukkan bahwa metode yang diusulkan dapat meningkatkan kinerja CNN asli.

Penelitian terkait berikutnya adalah penelitian dari Asaziduhu Gea dengan judul “*Optimasi Turn Around Time Pada Penjadwalan Round Robin Dengan Mencari Quantum Time Optimal Menggunakan Algoritma Simulated Annealing*”. Perencanaan proses dalam antrian harus dikelola oleh sistem operasi agar semua proses dapat dilayani tanpa proses yang di tolak atau terlalu lama untuk di respon sehingga berakibat pada waktu yang dialokasikan pada proses tersebut (Gea, 2014). Karena lamanya waktu respons, waktu burst yang ditetapkan untuk masing-masing proses ini telah berlalu, menyebabkan kesalahan proses pada proses berikutnya. Salah satu tanggung jawab sistem operasi adalah mengelola proses ini melalui penjadwalan. Algoritma penjadwalan yang digunakan tidak hanya menyelesaikan masalah turnaround time. Turnaround time yang diperoleh diharapkan lebih singkat setiap kali beberapa proses dijalankan. Oleh karena itu, Anda perlu mengoptimalkan lead time. Salah satu algoritma penjadwalan yang digunakan oleh banyak proses adalah round robin. Penelitian ini menggunakan algoritma simulasi annealing. Simulated annealing adalah algoritma berorientasi heuristik untuk menemukan solusi untuk masalah yang menghasilkan hasil yang cukup besar dengan energi yang lebih sedikit. Pengujian dijalankan dalam jumlah yang berbeda, 100, 150, dan 200 proses. Waktu burst untuk setiap proses berbeda, variabilitas alpha diuji pada 0,90 dan 0,99, dengan nilai suhu awal 110 dan nilai suhu akhir 46.

0,5. Pengujian setiap kelompok proses telah menunjukkan bahwa waktu tunggu rata-rata dari awal iterasi hingga iterasi terakhir berkurang secara signifikan.

Kajian terkait yang paling dekat adalah penelitian oleh Raymond Lamhot Sinaga (2015) yang berjudul "Simulated Annealing Algorithms for Solving MultiDepot Vehicle Routing Problem Using Travel Time Variables". Dalam beberapa tahun terakhir, perusahaan besar telah memperhatikan manajemen pengiriman kargo (Sinaga, 2015). Oleh karena itu, diperlukan sistem distribusi yang efisien untuk memenuhi permintaan konsumen secara tepat waktu. Tugas akhir ini membahas tentang perencanaan rute dan optimasi sistem distribusi dalam perencanaan. Masalah ini dapat dimodelkan sebagai masalah alokasi rute kendaraan (VRP). VRP sendiri memiliki beberapa perkembangan, seperti MultiDepot Vehicle Routing Problem (MDVRP) yang memiliki beberapa depot di dalam perusahaan. Berdasarkan hasil penelitian ini, algoritma simulasi annealing dapat menyelesaikan masalah MDVRP dalam menghasilkan rute kendaraan dengan waktu tempuh yang minimal. Kajian terkait yang paling dekat adalah Raymond Lamhot Sinaga (2019) yang berjudul "Solving the Travelling Salesman Problem Using Simulated Annealing". Travelling Salesman Problem (TSP) merupakan salah satu cara untuk menentukan rute atau rute terpendek bagi seorang tenaga penjualan. TSP digunakan dengan mengakses semua lokasi satu demi satu dan kemudian kembali ke lokasi semula. Proses Simulated Annealing adalah salah satu proses optimasi dengan ide dasar pendinginan dan pemadatan logam. Dengan menghitung nilai probabilitas emisi simulasi, metode ini dapat digunakan untuk menghindari minima lokal dan memungkinkan sinar yang disimulasikan menyelesaikan masalah TSP dan mendapatkan jalur yang optimal. Berdasarkan hasil pengujian yang dilakukan, dapat disimpulkan bahwa simulasi anil dapat mengatasi TSP. Jika kota awal adalah 1 dan kota tujuan adalah 10 dengan menggunakan 10 titik, maka urutan titik kota yang optimal adalah 1846275310, dan total jarak antar titik adalah 234. Selain itu, batas maksimum iterasi juga mempengaruhi probabilitas berikut: deteksi. Solusi terbaik. Semakin tinggi batas iterasi, semakin besar kemungkinan Anda untuk mendapatkan solusi terbaik.

BAB III

METODE PENELITIAN

3.1 Tempat dan Waktu Penelitian

Penelitian penerapan algoritma *simulated annealing* untuk menentukan rute terbaik pada *delivery multiple client* order online pada armada gojek dilakukan di Kota Stabat Kabupaten Langkat. Penelitian ini dilaksanakan pada semester genap sampai semester ganjil tahun ajaran 2020/2021.

Tabel 3.1 Waktu dan Jadwal Penelitian

No.	Waktu	Jadwal Penelitian					
		Februari	Maret	April	Mei	Juni	Juli
1.	Perencanaan						
2.	Pengumpulan Data						
3.	Analisis Data dan Perencanaan Sistem						
4.	Implementasi Sistem						
5.	Pengujian						

3.2 Bahan dan Alat Penelitian

Terdapat dua alat penelitian dalam penyelesaian penelitian algoritma *simulated annealing* untuk menentukan rute terbaik pada *delivery multiple client* order online pada armada gojek, yaitu:

1. Analisis Kebutuhan Hardware

Hardware yang digunakan memiliki spesifikasi sebagai berikut:

- 1) *Processor*: Intel® Core™ i3
- 2) *Memory*: 4 GB DDR 3
- 3) *Harddisk*: 500 GB
- 4) *Display*: 14 Inch WXGA (1366 x 768)
- 5) *Sound Card*: Integrated
- 6) *Video Type*: Intel®HD Graphics
- 7) *Keyboard, Mouse, Speaker, Headset*.

2. Analisis Kebutuhan Software

Software yang digunakan adalah:

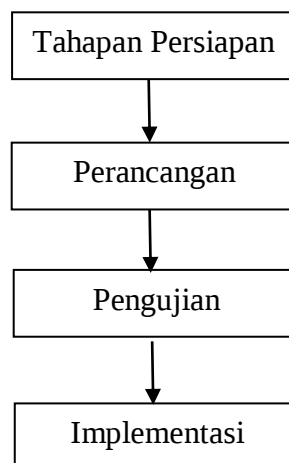
- 1) *Sistem Operasi Windows 10*
- 2) *Visual Studio Code*
- 3) *Xampp*
- 4) *Framework CodeIgniter 4.0*
- 5) *Google Maps API*

3.3 Cara Kerja

Dalam penerapan algoritma *simulated annealing* untuk menentukan rute terbaik pada *delivery multiple client* order online pada armada gojek, memiliki cara kerja sebagai berikut:

3.3.1 Perencanaan

Pada tahap ini merupakan proses awal dalam penelitian ini yang memiliki beberapa tahapan antara lain, tahap persiapan, tahap desain, tahap pengujian desain dan tahap implementasi. Tahapan dalam perencanaan digambarkan dalam bentuk diagram blok agar memudahkan dalam penyampain, adapun gambaran tersebut dapat dilihat sebagai berikut:



Gambar 3.1 Tahap Perencanaan

- a. Pada tahap persiapan merupakan tahapan yang harus dilakukan dalam menyiapkan segala sesuatu yang dibutuhkan dalam mendukung penelitian ini.
- b. Perancangan dilakukan agar memiliki skema awal dalam rancangan sistem yang akan dibangun.
- c. Pengujian merupakan tahapan yang dilakukan setelah hasil dari perencanaan selesai.
- d. Implementasi merupakan tahapan yang dilalui setelah pengujian agar menghasilkan perencanaan.

3.3.2 Pengumpulan Data

Tahap akuisisi data digunakan untuk mencari referensi teoritis terkait dengan kasus atau masalah langsung (Nasution, 2018). Menyelidiki literatur dari beberapa jurnal ilmiah dan tugas akhir yang terkait dengan penerapan algoritma cahaya simulasi untuk menemukan informasi dan menentukan rute terbaik untuk mengirimkan beberapa pesanan pelanggan secara online dengan armada Gojek. Hal ini dapat memperkuat alasan masalah dan implementasi dari penelitian.

3.3.3 Analisis Kebutuhan

Analisis kebutuhan sistem menggunakan beberapa kebutuhan material dalam sistem untuk menambah dan mendukung proses pembuatan objek (Nurul Fitria Insani, n.d.). Analisis kebutuhan sistem menjelaskan kebutuhan sistem. Tujuannya adalah untuk benar-benar memahami persyaratan dari sistem yang dibuat.

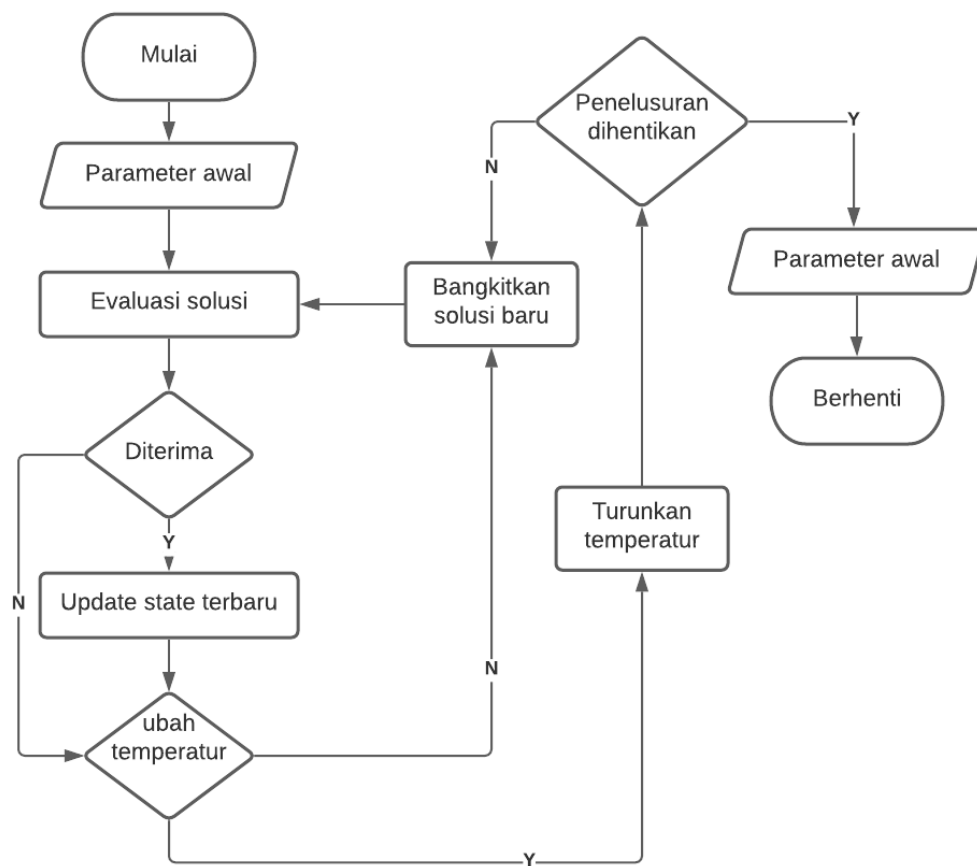
Sistem yang dianalisis adalah sistem yang berisi semua informasi yang terkait dengan implementasi algoritma simulasi annealing. Prosedur perancangan system ini terdiri dari memasukkan titik awal dan jarak klinik serta peta yang akan dianalisis. Kemudian diproses, algoritma serakah digunakan untuk menentukan rute terpendek, dan hasilnya ditampilkan.

Hasil akhir dari algoritma diimplementasikan untuk mendapatkan hasil yang baik dari graf yang terbentuk.

1. Aplikasi yang akan dibuat harus siap input sebagai berikut.
 - a. Pengguna memilih lokasi pemesanan pertama yang ditawarkan.
 - b. Dari titik awal, pengguna bisa mendapatkan informasi tentang lokasi yang paling dekat dengan pesanan.
2. Aplikasi yang Anda buat harus dapat melakukan proses sebagai berikut:
 - a. Sistem dapat menemukan rute terpendek ke hasil Pengguna ke lokasi pemesanan terdekat lahir.
 - b. Sistem dapat memproses rute pada peta sebagai sumber untuk menemukan rute terpendek bagi pengguna untuk mencapai lokasi pesanan berikutnya.
 - c. Sistem harus dapat menentukan panjang rute dan waktu tempuh.
3. Aplikasi yang sedang dibangun harus dapat melakukan hal berikut:
 - a. Sistem akan menampilkan peta yang berisi hasil pencarian rute terpendek Dari titik awal kota Medan ke tempat terdekat untuk memesan.
 - b. Sistem akan menampilkan rute pada peta sebagai rute terpendek.

3.3.4 Prosedur Kerja Algoritma *Simulated Annealing*

Untuk memecahkan suatu masalah, Anda memerlukan alur kerja sebagai pedoman penelitian utama untuk hasil yang baik. Alur kerja untuk penelitian ini ditunjukkan dalam diagram alir berikut:



Gambar 3.2 Prosedur Kerja Algoritma *Simulated Annealing*

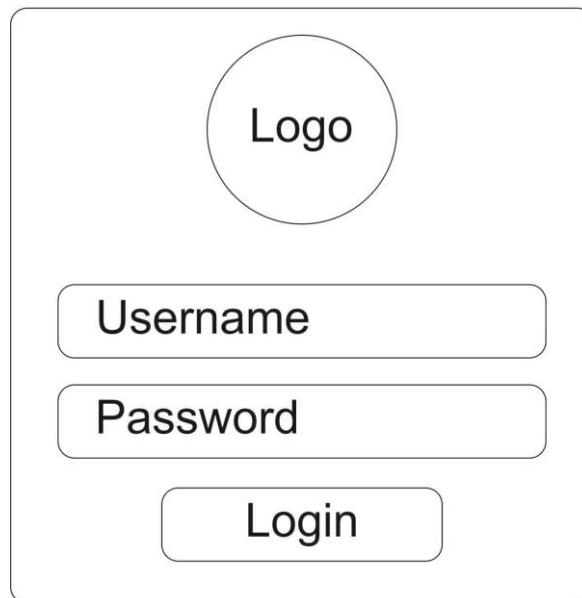
3.3.5 Perancangan *Interface*

Perancangan antar muka dirancang untuk memberikan gambaran suatu aplikasi yang dapat dengan mudah disajikan kepada pengguna. Pengguna aplikasi ini diharapkan dapat dengan mudah memahami fungsi tombol-tombol yang ada di dalam aplikasi. Saat menerapkan simulasi algoritma glow untuk menentukan rute terbaik untuk pengiriman beberapa pesanan pelanggan secara online dengan armada Gojek ini, terdapat beberapa bagian tampilan dengan kemampuan yang berbeda untuk setiap tombolnya. Fungsi tombol di setiap area tampilan dijelaskan dan ditunjukkan pada gambar berikut:

1. Rancangan Tampilan Login

Rancangan tampilan menu login merupakan tampilan awal setelah program dijalankan yang berfungsi sebagai akses masuk awal pada aplikasi yang dibangun. Tampilan login berisi teks judul aplikasi, logo Universitas, form *username*, form password dan tombol login sebagai akses masuk aplikasi, tampilan login sebagai berikut:

JUDUL APLIKASI



Gambar 3.3 Rancangan Tampilan Login

2. Rancangan Tampilan Dashboard

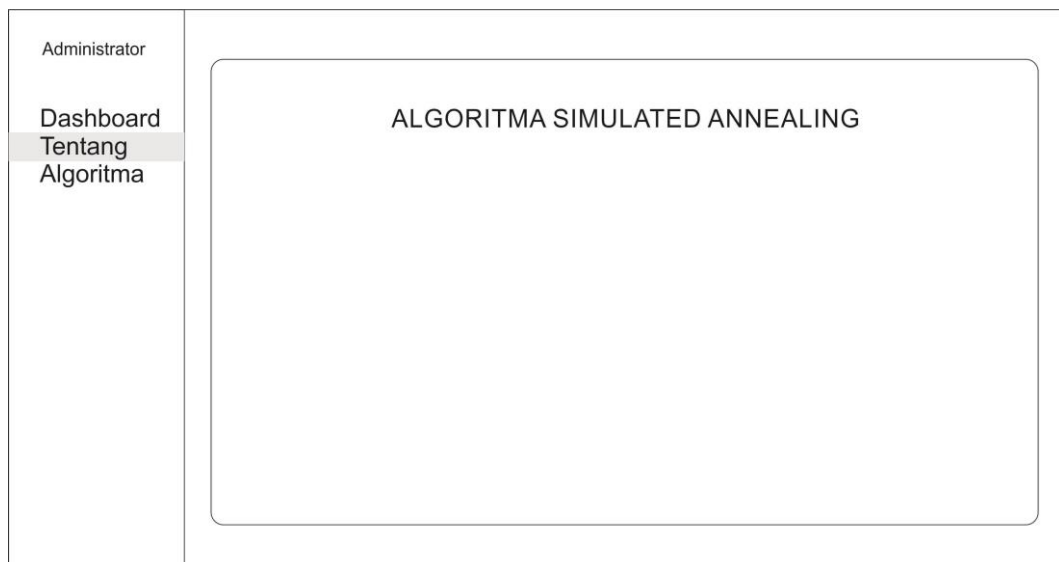
Rancangan tampilan dashboard merupakan tampilan setelah login yang berisi tentang judul aplikasi, logo, nama, nim dan program studi ilmu komputer Universitas Islam Negeri Sumatera Utara. Selain itu terdapat navbar dashboard, tentang dan algoritma, sedangkan di atas kanan terdapat tombol logout, tampilan dashboard sebagai berikut:



Gambar 3.4 Rancangan Tampilan Dashboard

3. Rancangan Tampilan Menu Tentang

Rancangan tampilan menu tentang adalah tampilan yang berisikan tentang algoritma *simulated annealing*. Tampilan menu tentang seperti dibawah ini:



Gambar 3.5 Rancangan Tampilan Menu Tentang

4. Rancangan Tampilan Menu Algoritma

Rancangan tampilan menu algoritma adalah tampilan yang berisikan tentang penerapan algoritma *simulated annealing* dalam menentukan rute terbaik pada

delivery multiple client order online pada armada gojek. Tampilan menu algoritma seperti dibawah ini:

Administrator Dashboard Tentang Algoritma	Lokasi Awal	Lokasi Akhir
	Lokasi Terbaik	

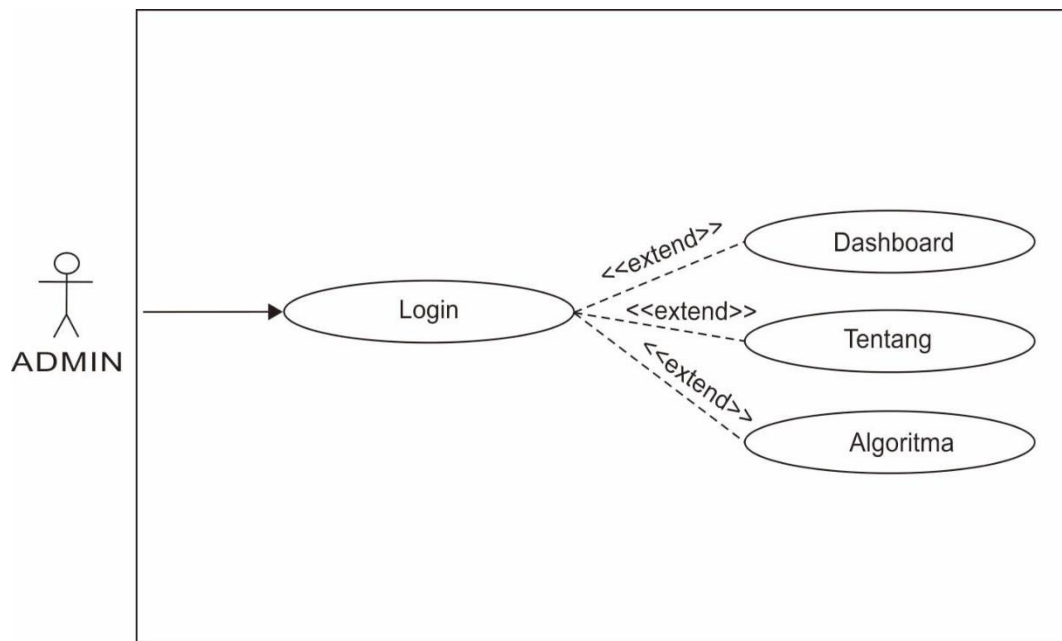
Gambar 3.6 Rancangan Tampilan Menu Algoritma

3.3.6 Perancangan Sistem

Desain sistem adalah bagian penting dari manufaktur sistem, struktur sistem ini harus memberikan gambaran tentang aplikasi yang diajukan. Perancangan sistem merupakan tahap awal dalam perancangan perangkat lunak. Ukuran ini dilakukan untuk menentukan keadaan umum sistem.

4. *Diagram Use Case*

Untuk mendapatkan informasi dari sistem yang dibuat, penulis menggunakan use case diagram. Angka ini menunjukkan proses yang sedang berlangsung di aplikasi:



Gambar 3.7 Diagram Use Case

a. Definisi Aktor

Berikut adalah deskripsi pendefinisian aktor pada sistem penerapan algoritma *simulated annealing* untuk menentukan rute terbaik pada *delivery multiple client* order online pada armada gojek:

Tabel 3.2 Definisi Aktor

Aktor	Deskripsi
Admin	Pemilik aplikasi.

b. Definisi Use Case

Berikut adalah deskripsi pendefinisian *Use case* pada penerapan algoritma *simulated annealing* untuk menentukan rute terbaik pada *delivery multiple client* order online pada armada gojek:

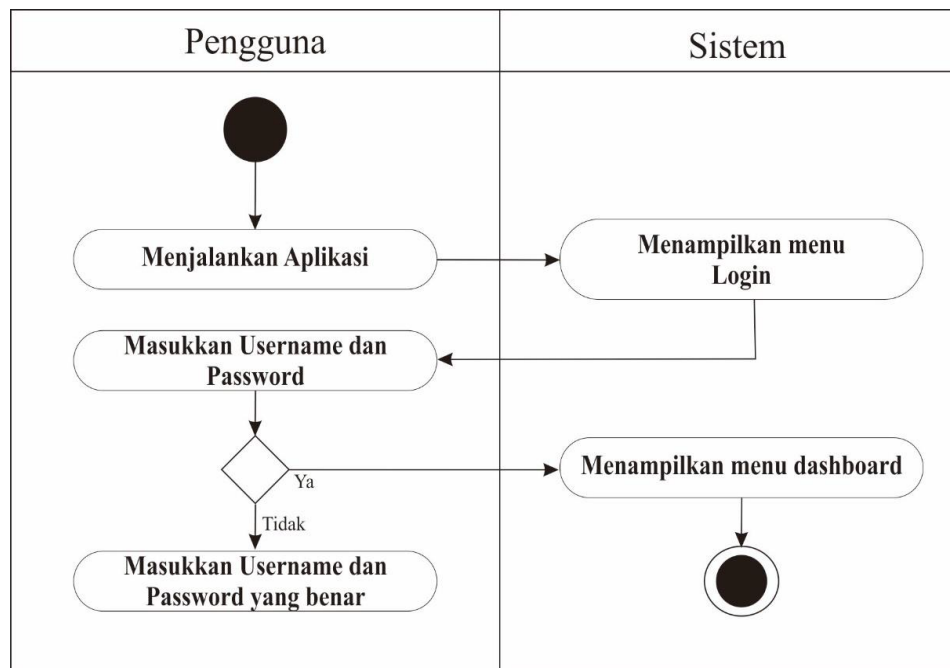
Tabel 3.3 Definisi Use Case

No	Use case	Deskripsi
1.	Login	Merupakan menu awal untuk login admin
2.	Dashboard	Merupakan menu utama aplikasi

3.	Tentang	Merupakan menu judul dan nama pembuat aplikasi.
4.	Algoritma	Merupakan menu pengaplikasian algoritma

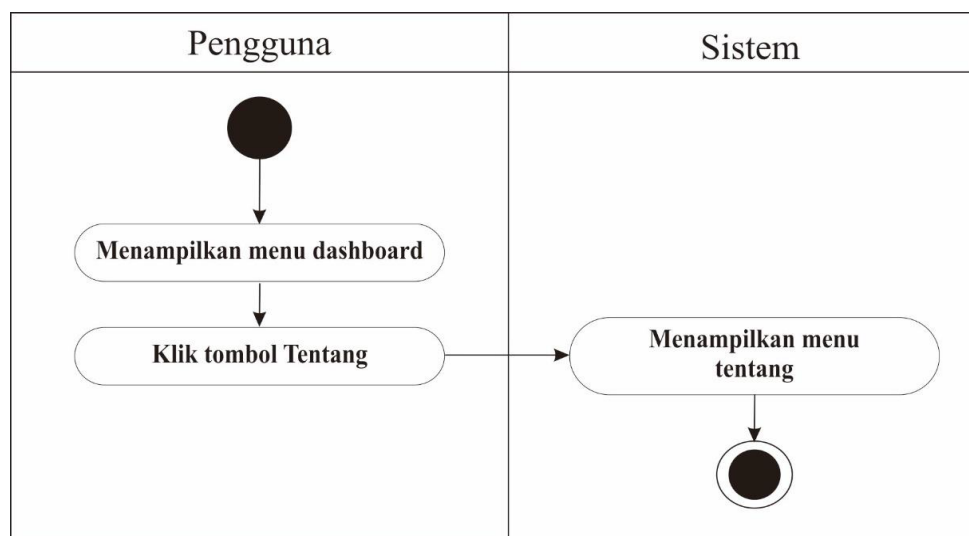
5. Diagram Activity

a. Diagram Activity Login



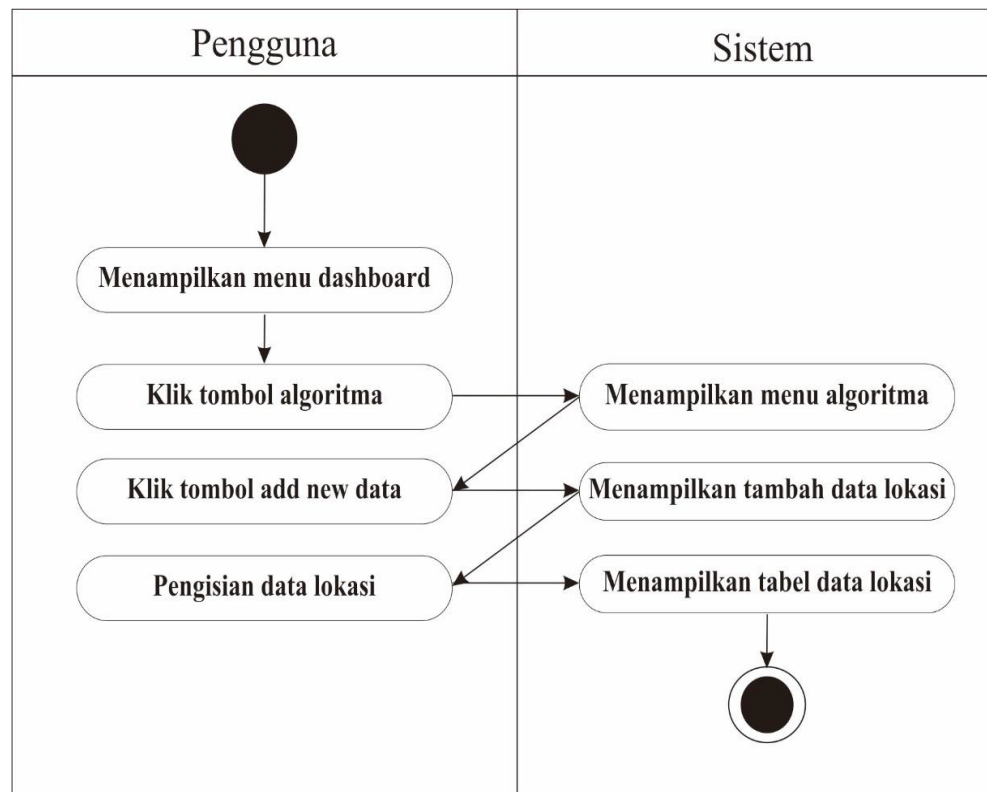
Gambar 3.8 Diagram Activity Login

b. Diagram Activity Tentang



Gambar 3.9 Diagram Activity Tentang

c. *Diagram Activity Algoritma*

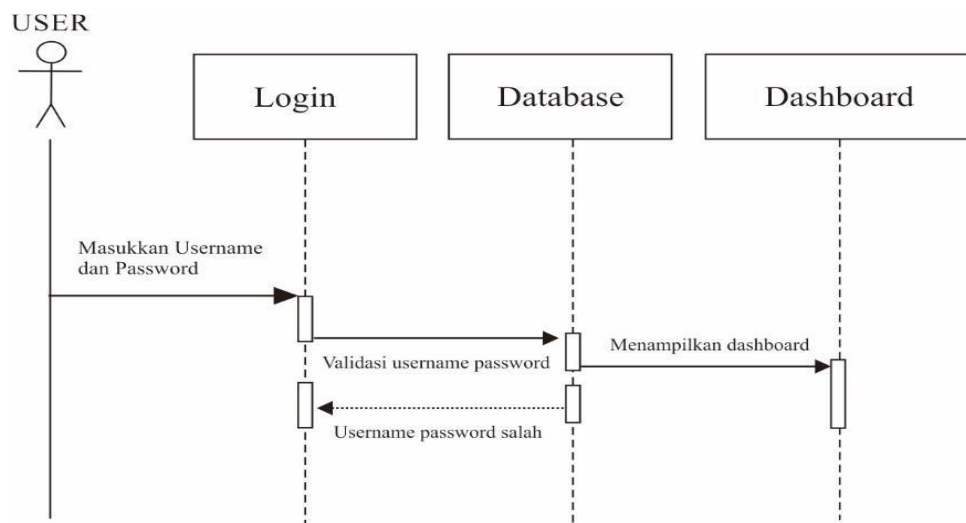


Gambar 3.10 *Diagram Activity Algoritma*

6. *Diagram Sequence*

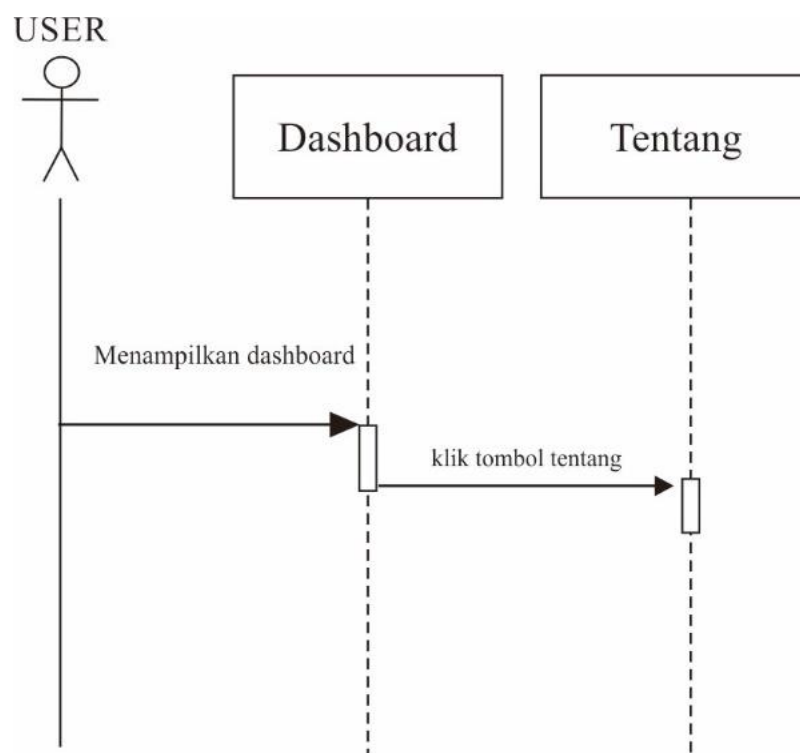
Berikut adalah *diagram sequence* pada penerapan algoritma *simulated annealing* untuk menentukan rute terbaik pada *delivery multiple client order online* pada armada gojek:

a. *Diagram Sequence Login*



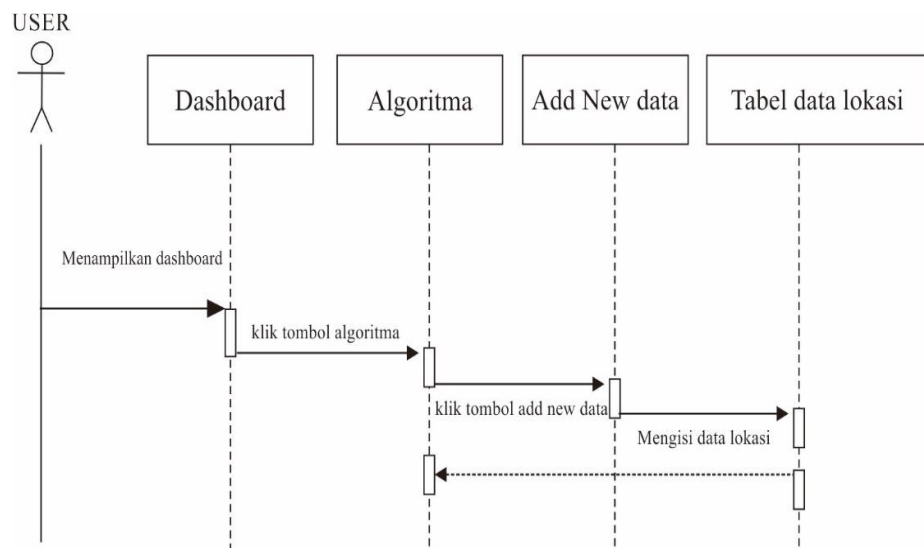
Gambar 3.11 *Diagram Sequence Login*

b. *Diagram Sequence Tentang*



Gambar 3.12 *Diagram Sequence Menu Tentang*

c. *Diagram Sequence Algoritma*



Gambar 3.13 *Diagram Sequence Menu Algoritma*

3.3.7 Pengujian Sistem

Proses pengujian menggambarkan hasil pengujian dari aplikasi yang dirancang dan diimplementasikan. Tesnya adalah, Aplikasi mungkin berperilaku seperti yang diharapkan dan ini akan terdeteksi dan ini berjalan sesuai dengan desain Anda.

Untuk menjalankan aplikasi yang mengimplementasikan algoritma simulasi annealing untuk menentukan rute terbaik untuk mengirimkan beberapa pesanan pelanggan secara online di armada Gojek, yang sebelumnya diprogram pada persiapan perangkat keras dan perangkat lunak komputer, saya harus menerapkan persyaratan. Pada penelitian ini, aplikasi dibuat dengan menggunakan bahasa pemrograman PHP7 dan software Visual Studio Code.

3.3.8 Penerapan

Dengan menerapkan algoritma simulasi annealing berbasis website, penerapan pencarian rute yang optimal untuk pengiriman beberapa pesanan pelanggan secara online ke armada Gojek akan diimplementasikan nantinya. Aplikasi ini membahas tentang cara kerja simulasi algoritma glow dan apakah

algoritma terbaik untuk menentukan rute terbaik untuk melayani beberapa pesanan pelanggan secara online di Armada Gojek..

BAB IV

HASIL DAN PEMBAHASAN

4.1 Pembahasan

Beberapa langkah yang dibahas dalam penelitian ini adalah analisis data, representasi data, hasil analisis data dan desain berikut ini.

4.1.1 Analisis Data

Pada aplikasi yang telah dirancang maka terdapat beberapa data proses. Data tersebut berupa pertanyaan yang diperoleh dari 10 orang responden yang masing-masing menjawab beberapa pertanyaan yang sudah disediakan. Responden merupakan *driver* ojek *online* pada armada gojek yang biasa menerima orderan *Gofood* di Kota Stabat Kabupaten Langkat. Dalam penerapan algoritma *simulated annealing* untuk menentukan rute terbaik pada *delivery multiple client* order online memerlukan analisis masalah, analisis persyaratan dan analisis proses.

1. Analisis Masalah

Analisis adalah langkah yang bertujuan untuk mengidentifikasi suatu masalah

Ini akan ditampilkan ketika aplikasi diajukan. Ini untuk mencegah kesalahan dalam aplikasi Anda selama proses desain. Ini akan memungkinkan sistem berfungsi dengan baik dan selesai tepat waktu. Dalam analisis sistem ini, sistem yang akan dianalisis.

- a. Bagaimana menentukan rute pengiriman terbaik untuk mengantarkan pesanan online pelanggan dengan menggunakan algoritma *simulated annealing*.
- b. Bagaimana mengimplementasikan program yang dibangun terhadap penentuan rute pengiriman terbaik untuk mengantarkan pesanan online pelanggan dengan menggunakan algoritma *simulated annealing* pada Kota Stabat Kabupaten Langkat.

Ada 10 nama-nama resto di Stabat yang sering dipesan melalui aplikasi gojek dan akan diterapkan dalam maps yang terdapat pada sistem yang dibangun. Nama-nama resto dapat dilihat pada tabel dibawah ini:

Tabel 4.1 Nama-nama Resto/Cafe di Stabat

NO	NAMA-NAMA RESTO/CAFE	NOTE	ALAMAT
1	AYAM PENYET CINDELARAS STABAT	A	JL. PROKLAMASI NO 9, KWALA BINGAI, KEC. STABAT, KAB.LANGKAT, SUMATERA UTARA
2	AYAM PRESTO CABE IJO STABAT	B	JL. PERDAMAIAN, KEC. STABAT, KAB. LANGKAT, SUMATERA UTARA
3	PONDOK LESEHAN PRABU DAN TOMBOLUWE	C	JL. JEND SUDIRMAN NO 9, PERDAMAIAN KEC.STABAT, KAB LANGKAT, SUMATERA UTARA
4	DAPOER ALUN-ALUN	D	JL. KWALA BINGAI, KABUPATEN LANGKAT, SUMATERA UTARA
5	RESTORAN STABAT SEAFOOD	E	JL.KH.ZAINAL ARIFIN, NO,24 STABAT, KAB LANGKAT, SUMATERA UTARA
6	KAFE SIRIP BIRU	F	JL. PROKLAMASI, KWALA BINGAI, KEC.STABAT, KAB LANGKAT, SUMATERA UTARA
7	KAFE UNCLE SIX	G	JL. PROKLAMASI, NO 10 KWALA BINGAI, KEC STABAT, KAB.LANGKAT, SUMATERA UTARA

8	AYAM BAKAR PUTRA SOLO	H	JL. PROKLAMASI, KWALA BINGAI, KEC STABAT, KAB. LANGKAT, SUMATERA UTARA
9	BAKSO PUTRA BENGAWAN SOLO	I	JL. PROKLAMASI, KWALA BINGAI, KEC. STABAT, KAB.LANGKAT, SUMATERA UTARA
10	LESEHAN LEMBUR KEMUNING STABAT	J	JL. KWALA BINGAI, KEC STABAT, KAB LANGKAT, SUMATERA UTARA

2. Analisis Kebutuhan

Analisis kebutuhan dibagi menjadi dua bagian: analisis kebutuhan. Analisis kebutuhan fungsional dan non-fungsional.

A. Persyaratan fungsional

Persyaratan fungsional di sini untuk menggambarkan aplikasi seperti: disponsori. Aplikasi ini melakukan perhitungan dan membuat rute terbaik saat mengantarkan pesanan dari banyak pelanggan di kota Stabat. Dengan menggunakan algoritma simulasi anil, ada persyaratan fungsional tertentu seperti:

- b. Aplikasi ini menggunakan peta restoran Kota Stabat yang dipilih dan dapat ditampilkan di aplikasi sesuai dengan tampilan peta yang ditentukan.
- c. Aplikasi dibangun dengan menggunakan node dan edge pada peta dan dapat dilihat berdasarkan nama dan rute dalam aplikasi untuk menemukan rute terbaik untuk mengantarkan pesanan pelanggan. Beberapa rute terhubung ke setiap sisi (tepi). Simpul (simpul).
- d. Aplikasi ini menggunakan algoritma simulasi annealing untuk menghasilkan pencarian rute yang optimal dari sebuah peta.

3. Persyaratan non-fungsional

Persyaratan non-fungsional adalah persyaratan yang mencakup kinerja operasional dan kinerja aplikasi. Ada beberapa persyaratan non-fungsional yang harus dipenuhi.

4. Persyaratan Non-Fungsional

Persyaratan non-fungsional adalah persyaratan yang berisi kinerja operasional dan kinerja aplikasi. Ada beberapa persyaratan non-fungsional yang harus dipenuhi yaitu:

A. Kemudahan penggunaan

Untuk membuat aplikasi ini harus mudah digunakan (user friendly). Singkatnya, sistem ini harus mudah digunakan bagi pengguna dengan antarmuka yang sederhana dan mudah dipahami.

B. Kinerja

Aplikasi yang akan dibangun ini harus mampu menampilkan hasil rute terbaik dalam pengiriman pesanan pelanggan, yaitu dari beberapa algoritma yang digunakan yaitu algoritma simulasi annealing.

C. Manajemen mutu

Aplikasi yang dibangun harus berkualitas tinggi, menampilkan peta, menampilkan jarak dan memberikan hasil rute terbaik saat memesan.

5. Analisis Proses

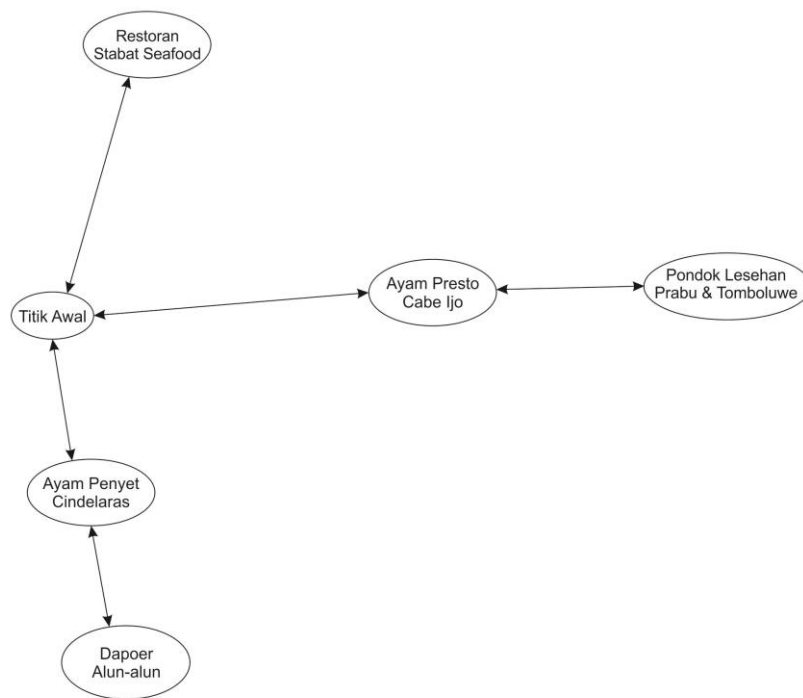
Aplikasi yang akan dibuat bahasa pemrogramannya adalah PHP sebagai software *Visual Studio Code* sebagai *text editor*. Algoritma yang digunakan untuk menentukan rute rute terbaik dalam pengantaran orderan pelanggan lebih dari satu adalah Algoritma *simulated annealing*.

4.1.2 Representasi Data

1. Hasil Pengujian Proses Algoritma *Simulated Annealing*

Pada algoritma *simulated annealing* langkah awal yang harus ditentukan adalah menentukan titik awal dengan cara klik tombol “*Add New Data*”, selanjutnya pilih tanggal pengiriman, masukkan alamat awal, pilih nama tempatnya dan

pilih titik *latitude* dan *longitude*. Setelah dipilih titik awal yang berupa tempat awal *driver*, kemudian masukkan nama resto-resto yang akan dituju klik tombol “Tambah Lokasi Tujuan” dimana tombol akan berfungsi untuk menambahkan rute-rute tempat pemesanan resto. Selanjutnya menampilkan hasil rute terbaik pada resto-resto yang akan dipesan. Hasil pencarian menunjukkan rute dalam bentuk peta, menunjukkan jarak dan titik yang diperlukan oleh algoritma. Dalam pengujian ini diambil contoh 6 lokasi yang merupakan 1 lokasi awal rute dan 5 lokasi resto yang ada di Stabat. Berikut tampilan *graph* resto-resto di kota Stabat:



Gambar 4.1 Tampilan *Graph* Resto-resto di kota Stabat

2. Hasil Pengujian Proses Algoritma *Simulated Annealing*

Langkah selanjutnya adalah melakukan proses perhitungan dengan menggunakan algoritma *simulated annealing*, perhitungan dilakukan secara manual dengan 6 titik lokasi yaitu 1 lokasi awal dan 5 lokasi resto, berikut tabel simulasi rute:

Tabel 4.2 Simulasi Rute

Index	Nama lokasi	Alamat	Kordinat
0	Bundaran Stabat	Jl. Lintas Sumatera	3.7508700241701685, 98.45308786839799
1	Ayam Penyet Cindelaras Stabat	Jl. Proklamasi No 9, Kwala Bingai, Kec Stabat, Kab.Langkat, Sumatera Utara	3.7479841233743536, 98.45023036833598
2	Ayam Presto Cabe Ijo Stabat	Jl. Perdamaian, Kec Stabat, Kab. Langkat, Sumatera Utara	3.751006508493181, 98.46381212649946
3	Pondok Lesehan Prabu Dan Tomboluwe	Jl. Jend Sudirman No 9, Perdamaian Kec.Stabat, Kab Langkat, Sumatera Utara	3.7509685850821612, 98.46497528232369
4	Dapoer Alun-Alun	Jl. Kwala Bingai, Kabupaten Langkat, Sumatera Utara	3.7449769268768156, 98.44572123696118
5	Restoran Stabat Seafood	Jl.Kh. Zainal Arifin, No,24 Stabat, Kab Langkat, Sumatera Utara	3.75809640622375, 98.45211292447851

Penyelesaian untuk *scenario* bahwa *driver* memilih 6 (enam) destinasi kunjungan termasuk lokasi awal keberadaan, dengan proses iterasi sebanyak 10 kali. Ada beberapa operator yang digunakan pada algoritma *simulate annealing*, diantaranya:

- Node lokasi disimpan pada list atau Larik L
- Bangkitkan 2 (dua) bilangan random 1 sampai NC, dengan $N1 < N2$
- Baris depan = L (1) sampai L(N1-1)
- Baris tengah = L(N1) sampai L(N2)
- Baris belakang = L(N2+1) sampai L(NC)
- Bangkitkan bilangan random r
 - Apabila $r < 0.5$, maka
 - a. DepanBaru = Baris Depan

- b. $TengahBaru = \text{Baris tengah dengan urutan dibalik dari belakang ke depan}$
- c. $BelakangBaru = \text{Baris Belakang}$
 - Apabila $r > 0.5$ maka
- a. $Baris sementara = [\text{Depan belakang}]$
- b. Misalkan baris sementara memiliki M elemen, bangkitkan bilangan random r dengan nilai antara 1 sampai M
- c. $DepanBaru = \text{Baris sementara}(1:r)$
- d. $TengahBaru = \text{Tengah}$
- e. $BelakangBaru = \text{Sementara}(r+1:M)$

Misalkan asumsi rute terbaik sementara adalah

$L_{min} [0-1-2-3-4-5]$ dengan energi = $450 + 2300 + 140 + 2400 + 1900 = 7.19 \text{ KM}$

Parameter:

$NC = 5$ (titik yang dikunjungi)

$M_{iterasi} = 2$

$MaxIterasi = M_{iterasi} * NC = 10$

Iterasi 1:

Bilangan random $N1 = 3, N2 = 4$

Baris depan = $[1, 2]$

Baris tengah = $[3, 4]$

Baris belakang = $[5]$

Bilangan random $r = 0.05$ (kurang dari 0.5), maka

Baris depan baru = $[1, 2]$

Baris tengah baru = $[4, 3]$

Baris belakang baru = $[5]$

Diperoleh jalur baru: $[0-1-2-4-3-5]$

Energi jalur baru = 9.55

9.55 km > 7.19 km, maka jalur baru ditolak

Iterasi 2:

Bilangan random $N1 = 2$, $N2 = 4$

Baris depan = [1]

Baris tengah = [2, 3, 4]

Baris belakang = [5]

Bilangan random $r = 0.68$ (lebih dari 0.5), maka buat dulu baris sementara

Baris sementara = [1,5]

Baris depan baru = [1,5]

Baris tengah baru = [2, 3, 4]

Baris belakang baru = []

Diperoleh jalur baru: [0-1-5-2-3-4]

Energi jalur baru = 6.29

6.29 km < 7.19 km, maka jalur baru diterima

Iterasi 3:

Bilangan random $N1 = 1$, $N2 = 3$

Baris depan = []

Baris tengah = [1, 2, 3]

Baris belakang = [4, 5]

Bilangan random $r = 0.58$ (lebih dari 0.5), maka buat dulu baris sementara

Baris sementara = [4,5]

Baris depan baru = [4]

Baris tengah baru = [1,2,3]

Baris belakang baru = [5]

Diperoleh jalur baru: [0-4-1-2-3-5]

Energi jalur baru = 6.34

6.34 km < 7.19 km, maka jalur baru diterima

Iterasi 4:

Bilangan random N1 = 1, N2 = 3

Baris depan = []

Baris tengah = [1, 2, 3]

Baris belakang= [4, 5]

Bilangan random r = 0.71 (lebih dari 0.5), maka buat dulu baris sementara

Baris sementara = [4,5]

Baris depan baru = [4, 5]

Baris tengah baru = [1,2,3]

Baris belakang baru = []

Diperoleh jalur baru: [0-4-5-1-2-3]

Energi jalur baru = 6.74

6.74 km < 7.19 km, maka jalur baru diterima

Iterasi 5:

Bilangan random N1 = 4, N2 = 5

Baris depan = [1, 2, 3]

Baris tengah = [4, 5]

Baris belakang= []

Bilangan random $r = 0.27$ (kurang dari 0.5), maka

Baris depan baru = [1,2,3]

Baris tengah baru = [5, 4]

Baris belakang baru = []

Diperoleh jalur baru: [0-1-2-3-5-4]

Energi jalur baru = 6.89

$6.89 \text{ km} < 7.19 \text{ km}$, maka jalur baru diterima

Iterasi 6:

Bilangan random $N1 = 1, N2 = 2$

Baris depan = []

Baris tengah = [1, 2]

Baris belakang= [3, 4, 5]

Bilangan random $r = 0.67$ (lebih dari 0.5), maka buat dulu baris sementara

Baris Sementara = [3, 4, 5]

Baris depan baru = [3]

Baris tengah baru = [1, 2]

Baris belakang baru = [4, 5]

Diperoleh jalur baru: [0-3-1-2-4-5]

Energi jalur baru = 9.6

$9.6 \text{ km} > 7.19 \text{ km}$, maka jalur baru ditolak

Iterasi 7:

Bilangan random $N1 = 1, N2 = 2$

Baris depan = []

Baris tengah = [1, 2]

Baris belakang = [3, 4, 5]

Bilangan random $r = 0.4$ (kurang dari 0.5), maka

Baris depan baru = []

Baris tengah baru = [2, 1]

Baris belakang baru = [3, 4, 5]

Diperoleh jalur baru: [0-2-1-3-4-5]

Energi jalur baru = 9.5

$9.5 \text{ km} > 7.19 \text{ km}$, maka jalur baru ditolak

Iterasi 8:

Bilangan random $N1 = 1, N2 = 5$

Baris depan = []

Baris tengah = [1, 2, 3, 4, 5]

Baris belakang = []

Bilangan random $r = 0.09$ (kurang dari 0.5), maka

Baris depan baru = []

Baris tengah baru = [5, 4, 3, 2, 1]

Baris belakang baru = []

Diperoleh jalur baru: [0-5-4-3-2-1]

Energi jalur baru = 7.44

$7.44 \text{ km} > 7.19 \text{ km}$, maka jalur baru ditolak

Iterasi 9:

Bilangan random $N1 = 3, N2 = 5$

Baris depan = [1, 2]

Baris tengah = [3, 4, 5]

Baris belakang= []

Bilangan random $r = 0.15$ (kurang dari 0.5), maka

Baris depan baru = [1, 2]

Baris tengah baru = [5, 4, 3]

Baris belakang baru = []

Diperoleh jalur baru: [0-1-2-5-4-3]

Energi jalur baru = 9.05

$9.05 \text{ km} > 7.19 \text{ km}$, maka jalur baru ditolak

Iterasi 10:

Bilangan random $N1 = 2$, $N2 = 3$

Baris depan = [1]

Baris tengah = [2, 3]

Baris belakang= [4, 5]

Bilangan random $r = 0.17$ (kurang dari 0.5), maka

Baris depan baru = [1]

Baris tengah baru = [3, 2]

Baris belakang baru = [4, 5]

Diperoleh jalur baru: [0-1-3-2-4-5]

Energi jalur baru = 7.20

$7.20 \text{ km} > 7.19 \text{ km}$, maka jalur baru ditolak

Hasil uji coba yang dilakukan menghasilkan rute terbaik pada *delivery multiple client* order dengan menerapkan algoritma *simulated annealing*, berikut tabel hasil rute terbaik algoritma *simulated annealing*:

Tabel 4.3 Hasil Pengujian Rute Terbaik Algoritma *Simulated Annealing*

Iterasi	N1	N2	Depan	Tengah	Belakang	r	Jalur Baru	Energy
1	3	4	1, 2	3, 4	5	0.05	0-1-2-4-3-5	9.55 km > 7.19 km Maka solusi rute ditolak
2	2	4	1	2, 3, 4	5	0.68	0-1-5-2-3-4	6.29 km < 7.19 km Maka solusi rute diterima
3	1	3		1, 2, 3	4, 5	0.58	0-4-1-2-3-5	6.34 km < 7.19 km Maka solusi rute diterima
4	1	3		1, 2, 3	4, 5	0.71	0-4-5-1-2-3	6.74 km < 7.19 km Maka solusi diterima rute diterima
5	4	5	1, 2, 3	4, 5		0.27	0-1-2-3-5-4	6.89 km < 7.19 km Maka solusi rute diterima
6	1	2		1, 2	3, 4, 5	0.67	0-3-1-2-4-5	9.6 km > 7.19 km Maka solusi rute ditolak
7	1	2		1, 2	3, 4, 5	0.4	0-2-1-3-4-5	9.5 km > 7.19 km Maka solusi rute ditolak
8	1	5		1, 2, 3, 4, 5		0.09	0-5-4-3-2-1	7.44 km

								> 7.19 km Maka solusi rute ditolak
9	3	5	1, 2	3, 4, 5		0.15	0-1-2-5-4- 3	9.05 km > 7.19 km Maka solusi rute ditolak
10	2	3	1	2, 3	4, 5	0.17	0-1-3-2-4- 5	7.20 km > 7.19 km Maka solusi rute ditolak

Dari hasil Pengujian diatas, ditemukan Rute **[0-1-5-2-3-4]** dengan Panjang **6.29 km** adalah rute terbaik dari lainnya.

4.2 Hasil

4.2.1 Analisis Data

Implementasi sistem merupakan lanjutan dari tahap analisis dan desain sistem. Sistem dibangun menggunakan bahasa pemrograman PHP. Pada sistem terdapat 4 (empat) tampilan menu yaitu menu login, menu dashboard, menu tentang dan menu algoritma.

1. Menu *Login*

Menu *login* merupakan menu awal dalam menggunakan sistem penerapan algoritma *simulated annealing* untuk menentukan rute terbaik pada *delivery multiple client order*, adapun tampilan dari menu *login* terdiri dari judul sistem, logo UIN-SU, *form username*, *form password* dan *button login*.

ALGORITMA SIMULATED ANNEALING UNTUK
MENENTUKAN RUTE TERBAIK PADA DELIVERY
MULTIPLE CLIENT ORDER ONLINE
PADA ARMADA GOJEK



Username

Password

Login

Gambar 4.2 Tampilan Menu *Login*

2. Menu *Dashboard*

Menu *dashboard* merupakan menu yang ditemukan setelah *user* melakukan login, adapun tampilan dari menu *dashboar* terdiri dari 2 bagian yaitu bagian kanan *user* yang terdiri dari judul sistem, nama dan nim mahasiswa, logo UIN-SU dan nama instansi, sedangkan bagian kiri *user* merupakan *main navigation* yang terdiri dari menu dashboard, menu tentang, menu algoritma.



Gambar 4.3 Tampilan Menu *Dashboard*

3. Menu Tentang

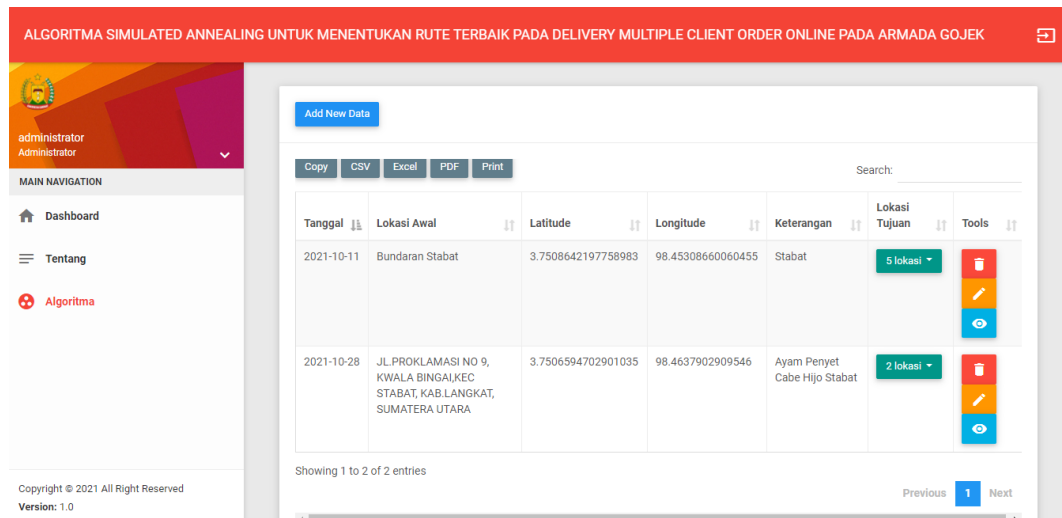
Menu tentang merupakan menu yang berisi pengetahuan dasar terkait algoritma *simulated annealing*.



Gambar 4.4 Tampilan Menu *Tentang*

4. Menu Algoritma

Menu algoritma merupakan menu yang berisi tentang input data lokasi, yang berbentuk tabel dan *report*.



Gambar 4.5 Tampilan Menu Tentang

Selanjutnya ketika tombol “Add New Data” maka akan muncul tampilan yang digunakan untuk mengisi lokasi awal.

The screenshot shows the 'Create New Data' form. It includes input fields for 'Tanggal Pengiriman' (dd/mm/yyyy), 'Alamat', 'Keterangan', 'latitude', and 'longitude'. Below these fields is a map titled 'Pilih Lokasi' (Choose Location) showing a street view of Stabat, Langkat. A red pin is placed on the map, and a yellow line indicates a route. The map shows various landmarks such as 'SMA Putra Jaya', 'Perumnas wonosari Srita Indah, lingkungan X', 'Ayam Presto SPBU 14 208152 Cabe Hijo Stabat Perdanaian Stabat', 'Yapim Tanja Stabat', 'Kantor koperasi', 'Alfamart', 'Komplek PNS Langkat Berseri (Perumnas)', and 'AL MAKSUM LANGKAT'.

Gambar 4.6 Tampilan Add New Data

Setelah menambahkan data awal yang merupakan rute awal selanjutnya jika ingin menambahkan rute selanjutnya pilih lokasi tujuan lalu pilih tambahkan lokasi.

ALGORITMA SIMULATED ANNEALING UNTUK MENENTUKAN RUTE TERBAIK PADA DELIVERY MULTIPLE CLIENT ORDER ONLINE PADA ARMADA GOJEK

administrator
Administrator

MAIN NAVIGATION

- Dashboard
- Tentang
- Algoritma

Copyright © 2021 All Right Reserved
Version: 1.0

Add New Data

Copy CSV Excel PDF Print

Search:

Tanggal	Lokasi Awal	Latitude	Longitude	Keterangan	Lokasi Tujuan	Tools
2021-10-11	Bundaran Stabat	3.7508642197758983	98.45308660060455	Stabat		Tambah lokasi View lokasi
2021-10-28	JL.PROKLAMASI NO 9, KWALA BINGAI,KEC STABAT, KAB.LANGKAT, SUMATERA UTARA	3.7506594702901035	98.4637902909546	Ayam Penyet Cabe Hijo Stabat	2 lokasi	

Showing 1 to 2 of 2 entries

Previous 1 Next

Gambar 4.7 Tampilan Lokasi Tujuan

Untuk melihat lokasi yang ditambahkan maka dapat klik tombol *view lokasi* sehingga akan muncul semua lokasi tujuan yang ditambahkan.

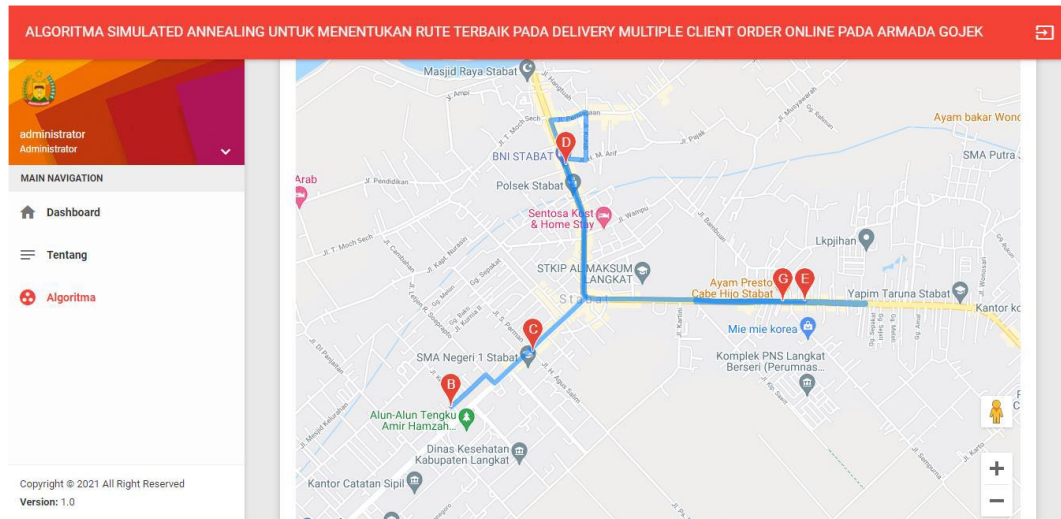
10 items/page

Pencarian

Lokasi Tujuan	Latitude	Longitude	Keterangan	Jarak	Tools
Jl. Jend Sudirman No 9, Perdamaian Kec.Stabat, Kab Langkat, Sumatera Utara	3.7508093522714314	98.46501740157892	Pondok Lesehan Prabu Dan Tomboluwe	1.3313166668819	
Jl.Kh.Zainal Arifin, No,24 Stabat, Kab Langkat,Sumatera Utara	3.7580839503706644	98.45211329400827	Restoran Stabat Seafood	0.80680293721256	
Jl.Kwala Bingai, Kabupaten Langkat, Sumatera Utara	3.7449478770768097	98.44571622550775	Dapoer Alun-Alun	1.045045891575	
Jl.Perdamaian,Kec Stabat, Kab. Langkat, Sumatera Utara	3.7507558229953593	98.46381174862672	Ayam Presto Cabe Ijo Stabat	1.1976069319968	
Jl.Perdamaian,Kec Stabat, Kab. Langkat, Sumatera Utara	3.7507558229953593	98.46381174862672	Ayam Presto Cabe Ijo Stabat	1.1976069319968	
Jl.Proklamasi No 9, Kwala Bingai,Kec Stabat, Kab.Langkat, Sumatera Utara	3.747961590233661	98.45023977101137	Ayam Penyet Cindelas Stabat	0.44795365495923	

Gambar 4.8 Tampilan View Lokasi

Untuk melakukan pengujian sistem maka klik tombol mata yang ada pada menu algoritma maka akan menghasilkan rute terbaik dalam menerapkan algoritma *simulated annealing*.



Gambar 4.9 Hasil Pengujian

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan yang didapat dari hasil pengujian dalam proses menemukan rute terbaik pada *delivery multiple client order online* pada armada gojek di Stabat menggunakan algoritma *simulated annealing* adalah sebagai berikut :

1. Penerapan algoritma *simulated annealing* dalam menemukan rute terbaik pada *delivery multiple client order online* armada gojek di Stabat dengan menggunakan data yang dikumpulkan dari driver armada gojek.
2. Sistem yang dibangun dengan menerapkan algoritma *simulated annealing* mampu menemukan rute terbaik pada *delivery multiple client order online* dengan cukup baik.

5.2 Saran

Saran yang dapat dipertimbangkan untuk penelitian dan pengembangan selanjutnya adalah:

1. Sistem ini perlu dikembangkan dengan jumlah vertex dan edge yang lebih banyak. Penelitian ini menggunakan algoritma yang merupakan algoritma simulasi annealing. Penelitian selanjutnya disarankan untuk menambahkan algoritma terbaru yang dapat menentukan rute terbaik.
2. Pada penelitian ini aplikasi berbasis website yang belum berbasis android. Investigasi lebih lanjut dianjurkan. Aplikasi harus berbasis Android secara online sehingga pengguna dapat menggunakannya kapan saja.

DAFTAR PUSTAKA

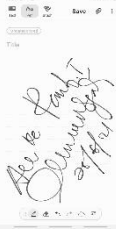
- Amrizal, V. & Aini, Q. (2013) Kecerdasan buatan. Universitas Islam Negeri Syarif Hidayatullah.
- Andriansyah, Novatama, R. , Dan Sentia, P, D. (2020). Algoritma simulasi anil (studi kasus) untuk masalah rute kendaraan yang heterogen. Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK), 7 (5), 933–942.
- Azmi, M, H., Sugiono & Tantrika, C, F, M. (2015) Rencana produksi rokok untuk meminimalkan delay maksimum menggunakan simulasi algoritma glow (studi kasus Pr. AdiBungu Malang). Jurnal Rekayasa dan Manajemen Sistem Industri, 3 (2).
- Bent, R. , Dan Hentenryck, P.V. (2006). Algoritma TwoHybrid untuk masalah perutean kendaraan penjemputan dan pengantaran menggunakan TimesWindows. Jurnal Riset Operasi, 123–137.
- Dridi, Y.I., Kammarti & R. , Ksouri, M. (2011) Optimalisasi multiguna mPDPTW. Metode agregasi menggunakan algoritma genetika dan batas bawah. Int. J. Komputer, Komunikasi & Kontrol, IV, 246-257.
- Fajarwati, I. A. & Wiwik, A. (2012) Penerapan algoritma evolusi diferensial untuk pemecahan masalah alokasi kendaraan masalah rute dengan pengiriman dan pengumpulan. Jurnal Teknik POMITS, 1 (1), 1-6.
- Firdaus, M., Masudin, I. , Dan Utama, D, M. (2015). Perencanaan flow shop menggunakan simulasi annealing [Universitas Muhammadiyah Malang]. <https://doi.org/10.16285/j.rsm.2007.10.006>
- Gea, A. (2014) Optimalkan waktu penyelesaian program round-robin dengan menggunakan algoritma simulasi anil untuk menentukan waktu kuantum yang optimal. Universitas Sumatera Utara.
- Hidayati, R., Gubtoro, I. , Dan Junianti, S. (2009). Menggunakan prosedur simulasi anil untuk memecahkan masalah salesman keliling. CESS (Journal of Computer Engineering Systems and Science, 4 (2).
- Jin Ting, G, F, L. (2019). Perbandingan pencarian luas-pertama dan algoritma pendakian paling curam ketika menemukan rute kunjungan rumah sakit. Universitas Sumatera Utara.
- Kallehauge B., L. J. & OBG, M. (2001) Dualitas Lagrange diterapkan pada masalah rute pengiriman dengan kerangka waktu. Laporan teknikal. IM. Universitas Teknik Denmark. DK 2800kg. Lyngby. [https://doi.org/10.25299/ge.2018.vol1\(1\).2258](https://doi.org/10.25299/ge.2018.vol1(1).2258)
- Maulana, G. G. (2017). Pembelajaran dasar algoritma dan pemrograman menggunakan El Goritma berbasis web. Jurnal Teknik Mesin, 6 (2), 8. <https://doi.org/10.22441/jtm.v6i2.1183>
- Noviandianto, G, E., Roman, M. , Dan Legowo, M, B. (2019). Gunakan simulasi anil untuk mengoptimalkan lokasi titik akses di jaringan WIFI Anda. Jurnal

SERI ILMU PENGETAHUAN DAN TEKNOLOGI ALAZHAR
INDONESIA, 5 (1).

- Nurhaniwa, D. & Widita, R. (2018) Optimalisasi sudut saat menentukan distribusi dosis dengan metode pendaran simulasi. Risalah SNIPS.
- Pangestuti, R, E (2020). Sebuah generator kasus uji berdasarkan model diagram urutan Unified Modeling Language (UML) menggunakan anil simulasi. Universitas Islam Negeri Maurana Mariquibrahim.
- Prasatria, A. (2017). Analisis perbandingan algoritma Floyd dan SDijkstra dalam pencarian rute terpendek ke field stop. Universitas Sumatera Utara.
- Wastafel dapur, M.R., Fanany, M. SAYA. & Arynmurthy, A, M. (2015). Algoritma anil simulasi untuk pembelajaran mendalam. Procedia Ilmu Komputer 72, 137-144.
- Santosa, B. & Willy, P (2011). Desain dan implementasi metode metaheuristik. Edisi pertama pencetakan utama. <https://doi.org/10.31004/obsesi.v4i2.483>
- Santoso, B (2016). Optimalisasi sistem penerangan dan irigasi taman kota berbasis programmable logic controller (PLC). Universitas Surabaya, 17 Agustus 1945.
- Shinaga, R.L. (2015). Sebuah algoritma cahaya simulasi untuk memecahkan masalah rute kendaraan multi-depot dengan waktu tempuh variabel. Institut Teknologi 10 November.
- Syahputra, Z. (2020) "Sistem informasi penjualan berbasis website dengan konsep Mvc (model view controller): Sistem informasi penjualan berbasis website dengan konsep Mvc (model view controller)", *Jurnal Mantik*, 4 (2) , Pp.11331137 . Doi: 10.35335 /mantik Vol4.2020. 915.p. 11331137.

KARTU BIMBINGAN SKRIPSI
PROGRAM STUDI ILMU KOMPUTER
Semester Gasal/Genap TA. 2020/2021

Nama : ANGGA RIZKY.S	Pembimbing I : Muhammad Ikhsan, S.T, M.Kom
NIM : 0701171001	Pembimbing II : Armansyah, M.Kom
Prog.Studi : ILMU KOMPUTER	SK Pembimbing : B.053/ST.IV.I/HM.03/4/2021
Judul Skripsi : Algoritma Simulated Annealing untuk menentukan Rute terbaik pada Delivery Multiple Client order pada Armada Gojek	

Pertemuan	Tgl	PEMBIMBING I	Tanda Tangan	Tgl	PEMBIMBING II	Tanda Tangan
		Materi bimbingan			Materi Bimbingan	
I	16/4 2021	Revisi BAB I		16/4 2021	Revisi BAB I	
II	23/4 2021	Revisi BAB II		23/4 2021	Revisi BAB II	
III	07/5 2021	Revisi BAB III		07/5 2021	Revisi BAB III	
IV	20/7 2021	ACC Final		20/7 2021	ACC Final	
V	23/8/ 2021	Revisi BAB IV		23/8/ 2021	Revisi BAB IV	
VI	6/9/ 2021	REVISI BAB IV		6/9/ 2021	Revisi BAB IV	
VII	27/9/ 2021	REVISI BAB V		27/9/ 2021	Revisi BAB V	
VIII		ACC FINAL			ACC FINAL	

Medan, Juni 2021

Diketahui oleh:
Ketua Program Studi
Ilmu Komputer

Ilka Zufria, S.Kom, M.Kom
NIP.198506042015031006

DAFTAR RIWAYAT HIDUP



I. DATA UTAMA

- 1 Nama Lengkap : **ANGGA RIZKY.S**
- 2 Tempat Tanggal Lahir : TAKENGON, 01 MEI 1999
- 3 Jenis kelamin : LAKI-LAKI
- 4 Agama : Islam
- 5 Status Perkawinan : Belum Kawin
- 6 Pendidikan Terakhir : S-1
- 7 Alamat Rumah
 - a. Jalan : Jln.pendidikan lk -2
 - b. Kelurahan/Desa : sidomulyo
 - c. Kecamatan :Stabat
 - d. Kabupaten/Kota :Langkat
 - e. Provinsi : Sumatera Utara
 - f. Kode Pos : 20815
 - g. No.HP : 0822-9466-0085
 - h. Email : anggarizkyajalah@gmail.com

8 Keterangan Badan

- a. Tinggi badan (cm) : 163
- b. Berat badan (kg) : 95
- c. Bentuk muka : Bulat
- d. Rambut : Lurus
- e. Warna kulit : Sawo matang
- f. Kegemaran (Hobby) : membaca

II. PENDIDIKAN

1. Pendidikan

No tingkat	Nama sekolah	Jurusan	Thn lulus	Tempat
SD	054901 sidomulyo	-	2011	Stabat
SMP	MTSN Stabat	-	2014	Stabat
SMA	MAN 1 Stabat	IPA	2017	Stabat
S-1	UINSU	ILMU KOMPUTER	2021	Medan

Stabat, 30 Desember 2021

ANGGA RIZKY.S, S.KOM