

**APLIKASI PENGOLAHAN DAN VISUALISASI
DATA PANTAUAN KEJADIAN LALULINTAS DI SURABAYA TIMUR
(STUDI KASUS : RADIO SUARA SURABAYA)**



Nama : Ferdian Kelana
NIM : 01.41010.0249
Program : S1 (Strata Satu)
Jurusan : Sistem Informasi

**SEKOLAH TINGGI
MANAJEMEN INFORMATIKA & TEKNIK KOMPUTER
SURABAYA**

2006

**APLIKASI PENGOLAHAN DAN VISUALISASI
DATA PANTAUAN KEJADIAN LALULINTAS DI SURABAYA TIMUR
(STUDI KASUS : RADIO SUARA SURABAYA)**

SKRIPSI

Diajukan sebagai syarat untuk menyelesaikan
Program Sarjana Komputer



Nama : Ferdian Kelana
NIM : 01.41010.0249
Program : S1 (Strata Satu)
Jurusan : Sistem Informasi

**SEKOLAH TINGGI
MANAJEMEN INFORMATIKA & TEKNIK KOMPUTER
SURABAYA
2006**

**APLIKASI PENGOLAHAN DAN VISUALISASI
DATA PANTAUAN KEJADIAN LALULINTAS DI SURABAYA TIMUR
(STUDI KASUS : RADIO SUARA SURABAYA)**

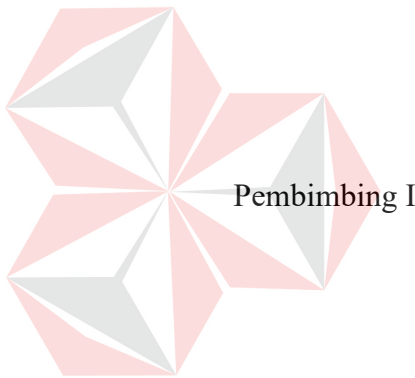
Disusun Oleh :

Nama : Ferdian Kelana

NIM : 01.41010.0249

Surabaya, Agustus 2006

Telah diperiksa, diuji dan disetujui :



Basuki Rahmat, S.Si, MT
NIDN. 0723076902

UNIVERSITAS
Dinamika

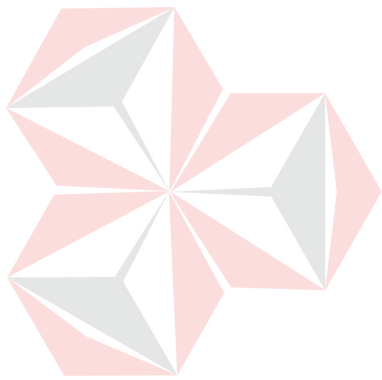
Pembimbing II

Maria Irmina Prasetyowati, S.Kom, MT.
NIDN. 0725057201

Mengetahui :

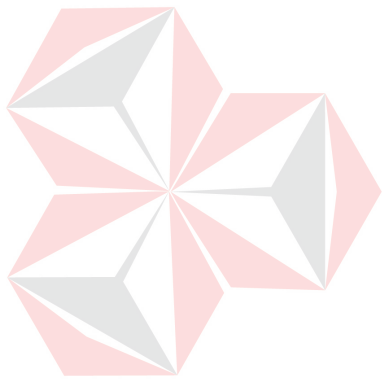
Wakil Ketua Bidang Akademik

Drs. Antok Supriyanto, M.MT
NIDN. 0726106201



UNIVERSITAS
Dinamika

*Jika Kita Mampu Memimpin Diri kita Sendiri,
Maka Akan Terbuka Jalan Menuju Kesuksesan.*



UNIVERSITAS
Dinamika

Kupersembahkan kepada:

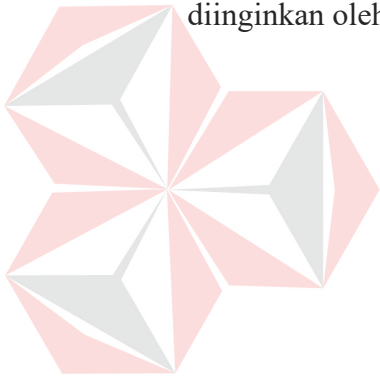
Ibu, Bapak, dan seluruh saudara ku tercinta di dunia

Alamamaterku STIKOM Surabaya

ABSTRAKSI

Banyak laporan kemacetan yang masuk setiap harinya ke radio Suara Surabaya, banyak pula pendengarnya yang bertanya rute mana yang harus dilalui untuk dapat segera mencapai tujuan. Penyiar radio yang mendapatkan pertanyaan seperti itu, biasanya akan bertanya kembali kepada pendengar yang lain, rute mana yang bisa ditempuh lebih cepat.

Dengan mengolah laporan-laporan yang masuk, aplikasi ini dapat mencari rute tercepat yang ditanyakan oleh pendengar dengan menggunakan algoritma ant colony optimization. Jadi, penyiar dapat langsung memberikan jawaban rute yang diinginkan oleh pendengar.



UNIVERSITAS
Dinamika

KATA PENGANTAR

Puji syukur kehadiran Allah SWT karena hanya berkat rahmat dan hidayah-Nya penulis akhirnya dapat menyelesaikan Tugas Akhir yang berjudul Aplikasi Pengolahan dan Visualisasi Data Pantauan Kejadian Lalulintas Di Surabaya Timur.

Dalam kesempatan ini penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Bapak, dan Ibu atas doa dan dukungannya sehingga tugas akhir ini dapat terselesaikan.
2. Ibu Maria Irminda Prasetyowati, S.Kom, MT selaku dosen pembimbing kedua atas bimbingan dan bantuannya selama pengerjaan tugas akhir ini.
3. Bapak Basuki Rahmat, S.Si, M.MT selaku dosen pembimbing pertama atas bimbingan, saran-saran dan bantuannya selama pengerjaan tugas akhir ini.
4. Bapak I Putu Agus Swastika. M.Kom untuk kebijakannya.
5. Bapak Wido untuk ide dan sarannya.
6. Samsul dan Fiqriy untuk penjelasan *Ant Colony*-nya.
7. Budi, Febri, Yudi, Pras untuk MapObject-nya.
8. Miranti, Kiki untuk statistik-nya.
9. Wawan, Didik, Anger61, Songot, Mojo, Risky, plus kacel, teman-teman kontrakan semampir yang banyak membantu dalam memberi *support*.
10. Fanny, Faris, Yoga, Dian, Dewi, dan seluruh mitra tim Visi 21 yang banyak memberi *support*.
11. Kaskuser Regional Surabaya, atas bantuan “data” dan *support*-nya.
12. Rekan-rekan STIKOM Surabaya khususnya angkatan 2001.

13. Semua pihak yang namanya tidak bisa dicantumkan satu persatu disini terima kasih banyak atas dukungannya.

Penulis sadar bahwa tugas akhir ini masih jauh dari sempurna. Untuk itu saran dan kritik sangat diharapkan untuk perbaikan tugas akhir ini.

Surabaya, Agustus 2006

Penulis



UNIVERSITAS
Dinamika

DAFTAR ISI

	Halaman
ABSTRAKSI	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah	2
1.3 Pembatasan Masalah	2
1.4 Tujuan	3
1.5 Sistematika Penulisan	3
BAB II LANDASAN TEORI.....	5
2.1 Visualisasi	5
2.2 Ant Colony	5
BAB III METODE PENELITIAN	13
3.1 Analisa Sistem	13
3.2 Perancangan dan Pemodelan Sistem Pencarian Rute	15
3.3 Perancangan Sistem	21
3.4 Disain Input dan Output	32
BAB IV IMPLEMENTASI DAN EVALUASI.....	45
4.1 Kebutuhan Sistem	45
4.2 Uji Coba Program	46

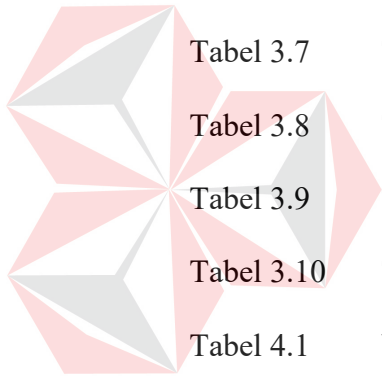
4.3	Evaluasi.....	58
BAB V PENUTUP.....		64
5.1	Kesimpulan	64
5.2	Saran.....	64
DAFTAR PUSTAKA		66
BIODATA PENULIS		67
LAMPIRAN.....		68



UNIVERSITAS
Dinamika

DAFTAR TABEL

	Halaman
Tabel 2.1	Lebar Jalan berdasarkan fungsi jalan 8
Tabel 2.2	Pembobotan Jalan berdasarkan lebar max. fungsi jalan 8
Tabel 3.1	Tabel Node 29
Tabel 3.2	Tabel Master_Jalan 29
Tabel 3.3	Tabel Master_Keadaan 30
Tabel 3.4	Tabel Master_Kejadian 30
Tabel 3.5	Tabel Master_User 30
Tabel 3.6	Tabel Master_Pendengar 31
Tabel 3.7	Tabel Bobot_Kelas 31
Tabel 3.8	Tabel Aturan Khusus 31
Tabel 3.9	Tabel Laporan 32
Tabel 3.10	Tabel Detil_Laporan 32
Tabel 4.1	Uji Coba Waktu Proses Pencarian Rute Tercepat 63



UNIVERSITAS
Dinamika

DAFTAR GAMBAR

Halaman

Gambar 2.1	Dasar perilaku semut untuk mencari jalur alternatif	6
Gambar 2.2.	Alur kerja semut buatan.	7
Gambar 3.1	Diagram blok secara umum Aplikasi Pengolahan dan Visualisasi Data Pantauan Kejadian Lalulintas	13
Gambar 3.2.	Diagram alir secara umum tentang system	14
Gambar 3.3	Alur Proses Pencarian Rute	15
Gambar 3.4	Flowchart Algoritma Ant Colony Optimation	17
Gambar 3.5	Flowchart Proses Simulate Artificial Ant	18
Gambar 3.6	Flowchart Proses pergerakan ke jalan selanjutnya	19
Gambar 3.7	Flowchart Proses Back Track	20
Gambar 3.8	Flowchart Proses Update feromon	21
Gambar 3.9	Contex Diagram	22
Gambar 3.10	Data Flow Diagram Level 0	23
Gambar 3.11	DFD Level 1 Maintenance Data	24
Gambar 3.12	DFD Level 1 Pengolahan dan Visualisasi	25
Gambar 3.13	DFD Level 2 Pengolahan Pencarian Rute	26
Gambar 3.14	Conseptual Data Model	27
Gambar 3.15	Phisical Data Model	28
Gambar 3.16	Desain Form Login	33
Gambar 3.17	Desain Form Utama	34
Gambar 3.18	Desain Form Ganti Password	35
Gambar 3.19	Desain Form Masukkan Laporan	37

Gambar 3.20	Desain Form Maintenance Data Jalan	38
Gambar 3.21	Desain Form Maintenance Data Pendengar	39
Gambar 3.22	Desain Form Maintenance Aturan Khusus	40
Gambar 3.23	Desain Form Maintenance Data User	41
Gambar 3.24	Desain Form Pencarian Rute Tercepat Alternatif (Untuk Gatekeeper)	42
Gambar 3.25	Desain Form Pencarian Rute Tercepat Alternatif (Untuk Penyiar)	42
Gambar 3.26	Desain Form Visualisasi Pantauan Kejadian Lalulintas	43
Gambar 4.1	Tampilan Form <i>Login</i>	46
Gambar 4.2	Tampilan Form Utama	48
Gambar 4.3	Tampilan Form Maintenance Data User	49
Gambar 4.4	Tampilan Form Ganti Password	50
Gambar 4.5	Tampilan Form Masukan Laporan	51
Gambar 4.6	Tampilan Form Maintenance Data Jalan	53
Gambar 4.7	Tampilan Form Maintenance Data Pendengar	54
Gambar 4.8	Tampilan Form Maintenance Data Aturan Khusus	55
Gambar 4.9	Tampilan Form Pengolahan Pencarian Rute (Untuk Gatekeeper)	56
Gambar 4.9	Tampilan Form Pengolahan Pencarian Rute (Untuk Penyiar)	57
Gambar 4.10	Tampilan Form Visualisasi Pantauan Kejadian	58
Gambar 4.11	Sketsa Salah Satu Persimpangan Jalan	59
Gambar 4.12	Hasil Pencarian dengan Algoritma <i>Ant Colony Optimization</i>	61

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Kejadian yang terjadi di jalan, perubahannya sangat cepat sekali dan bisa tidak terduga sama sekali, seperti kebakaran, kecelakaan, kerusakan, dan yang lainnya. Info kejadian lalulintas ini sangat dibutuhkan bagi pengguna jalan, khususnya pengendara kendaraan bermotor. Bila pengendara tahu kejadian kerusakan atau kemacetan di jalan A, maka pengendara bisa mencari jalan alternatif yang lain.

Untuk penyampaian informasi ini, para pengguna jalan dapat mendengarkan siaran radio di tape mobil maupun di *FM tuner* (radio mini yang bisa dibawa pengendara sepeda motor). Radio yang menyediakan informasi ini salah satunya adalah Radio Suara Surabaya (Radio SS).

Sumber data lalulintas radio SS berasal dari reporter radio SS yang terjun langsung ke jalan maupun laporan pendengar yang menyaksikan secara langsung kejadian yang terjadi di jalan. Dari laporan-laporan ini, penyiar akan menginformasikannya kepada pendengar secara langsung tanpa bantuan ataupun panduan dari sebuah peta. Dengan belum digunakannya peta sebagai panduan penyiar, maka bisa saja penyiar salah menyebutkan nama daerah atau nama jalan tempat terjadinya sesuatu kejadian lalulintas. Untuk membantu penyiar menyampaikan laporan kejadian lalulintas ke para pendengar, setidaknya penyiar memiliki gambaran atau dalam konkritnya, sebuah peta yang dapat ditandai secara digital atas laporan-laporan yang masuk dan kemudian di visualisasikan dalam sebuah peta.

Dengan di visualisasikan maka kejadian–kejadian yang terjadi di jalan atau lalu lintas akan tampak jelas, dengan begitu penyiari dapat dengan mudah memberikan jalan alternatif untuk menghindari kejadian yang buruk. Dari data data yang masuk, akan diolah untuk mendapatkan jalan alternatif, agar pendengar dapat menghindari kemacetan atau kesulitan dalam perjalanan dikarenakan kejadian yang terjadi di jalan tersebut. Selain itu perubahan apa saja yang terjadi di suatu jalan dapat di informasikan dengan mudah, karena historinya bisa ditampilkan kembali.

1.2 Perumusan Masalah

Dari latar belakang di atas, dapat dirumuskan masalah yang dihadapi adalah bagaimana membuat Aplikasi Pengolahan dan Visualisasi Data Pantauan Kejadian Lalulintas Di Surabaya Timur

1.3 Pembatasan Masalah

Batasan masalah dalam pembuatan sistem ini adalah sebagai berikut:

- a. Sumber data yang digunakan untuk sistem ini adalah, laporan dari pendengar yang melihat langsung dan reporter yang secara langsung terjun ke tempat kejadian. Dan dimasukkan ke dalam aplikasi secara manual oleh *gatekeeper* atau penyiari.
- b. Sistem ini hanya digunakan untuk mengolah data laporan di wilayah Surabaya Timur.
- c. Jalan yang dipantau hanya jalan besar(utama), Berdasarkan Buku Peta “Surabaya & Perkembangannya” edisi Terbaru, terbitan PT. Karya Pembina Swajaya, Surabaya. Dan tambahan jalan yang bisa dijadikan jalan alternatif hasil pemantauan langsung.

- d. Pengolahan laporan yang dihasilkan dari sistem ini adalah pembuatan jalur alternatif, dan pencarian jalur tercepat.
- e. Metode yang digunakan untuk membuat jalur alternatif dan mencari jalur tercepat, menggunakan metode *Ant Colony*.
- f. Program diaplikasikan pada *computer desktop*, tidak berbasis *web*.
- g. Hanya ada maintenance perubahan data ruas jalan, tidak ada maintenance penambahan ruas jalan.
- h. Traffic Light, kerusakan jalan diabaikan.

1.4 Tujuan

Tujuan dari sistem ini adalah membuat Aplikasi Pengolahan dan Visualisasi Data Pantauan Kejadian Lalulintas Di Surabaya Timur, agar mempermudah penyampaian kejadian yang terjadi di jalan oleh penyiar radio SS kepada pendengar.

1.5 Sistematika Penulisan

Sistematika penulisan pada laporan ini adalah sebagai berikut:

BAB I : PENDAHULUAN

Pada bab ini membahas tentang latar belakang masalah dan penjelasan permasalahan secara umum, perumusan masalah serta batasan masalah yang dibuat, tujuan dari pembuatan tugas akhir dan sistematika penulisan buku ini.

BAB II : LANDASAN TEORI

Pada bab ini membahas secara singkat teori-teori yang berhubungan dan mendukung dalam pembuatan tugas akhir ini.

BAB III : PERANCANGAN SISTEM

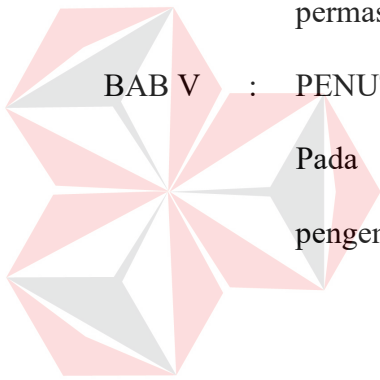
Pada bab ini membahas tentang perancangan sistem, analisa sistem, Diagram Alir, *System Flow*, *Data Flow Diagram* (DFD), *Entity Relationship Diagram* (ERD), struktur *database* yang digunakan dalam pembuatan aplikasi serta desain *input* dan *output*.

BAB IV : IMPLEMENTASI DAN EVALUASI

Pada bab ini membahas tentang implementasi dari aplikasi yang dibuat secara keseluruhan. Serta melakukan pengujian terhadap aplikasi yang dibuat untuk mengetahui aplikasi tersebut telah dapat menyelesaikan permasalahan yang dihadapi sesuai dengan yang diharapkan.

BAB V : PENUTUP

Pada bab ini berisikan kesimpulan dari sistem dan saran untuk pengembangan sistem.



UNIVERSITAS
Dinamika

BAB II

LANDASAN TEORI

2.1 Visualisasi

Visualisasi adalah menampilkan data-data dalam bentuk gambar agar dapat mempermudah untuk dibaca. Data-data yang diterima oleh *gatekeeper*, dimasukkan ke dalam sistem, dan akan di tampilkan dalam peta. Dengan dimasukkannya data ke dalam peta, maka penyiar maupun *gatekeeper* akan lebih mudah untuk membaca data laporan yang masuk. Setiap keadaan dan kejadian lalu lintas akan di wakili oleh simbol maupun oleh perbedaan warna garis.

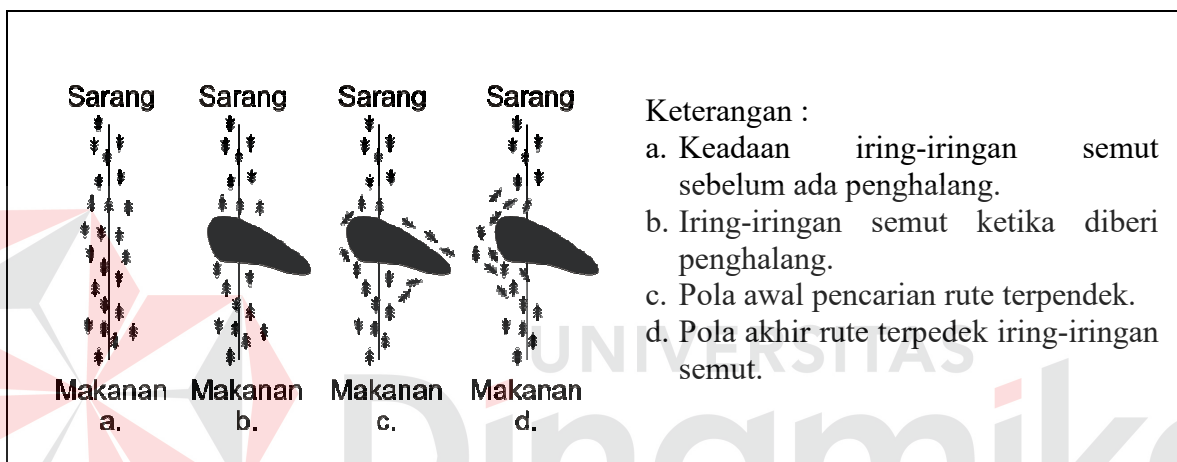
2.2 Ant Colony

Menurut K. Doerner dalam (2000:3) menyebutkan bahwa optimasi *ant colony* (*Ant Colony Optimization-ACO*) merupakan sistem cerdas yang diinspirasi oleh perilaku semut dan koloninya, yang digunakan untuk menyelesaikan permasalahan optimasi diskrit. Sistem ACO pertama kali diperkenalkan oleh Marco Dorigo (1992), dan dinamakan sistem semut (*Ant System-AS*). AS sebagai hasil penelitian dalam bidang komputasi cerdas yang digunakan untuk mendekati permasalahan optimasi kombinator.

Sebuah analogi terhadap bagaimana koloni semut berfungsi memberikan sebuah definisi baru model perhitungan, yang dinamakan Sistem Semut (*Ant System-AS*). AS dikemukakan sebagai sebuah pendekatan baru untuk optimasi kombinatorial.

Perilaku semut yang di digunakan dalam *Ant Colony* adalah tingkah laku semut dalam mencari makanan, dimana semut-semut tersebut secara berkelompok mencari makanan dari sarang mereka menuju sumber makanan dengan cara mencari jalan

terpendek dari sarang mereka menuju sumber makanan dan setelah ditemukan rute terpendek ke sumber makanan dari hasil pencarian beberapa rute maka semut-semut yang lain akan mengikuti rute terpendek yang berhasil ditemukan. Berdasarkan hal tersebut maka dibuat suatu algoritma yang meniru tingkah laku semut dalam mencari makanan dengan cara menentukan rute yang paling pendek dari beberapa rute yang dapat ditempuh untuk mencapai tujuan yang diinginkan.

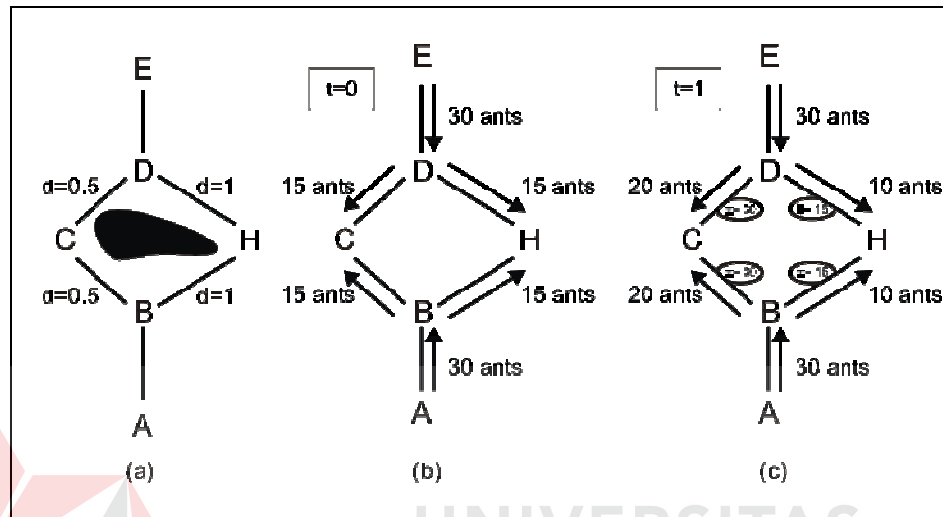


Gambar 2.1. Dasar perilaku semut untuk mencari jalur alternatif

Dasar perilaku semut di atas dapat digunakan untuk menjelaskan bagaimana mereka dapat menemukan rute terpendek dan menyambung kembali rute yang terputus akibat adanya penghalang. Iring-iringan semut ketika diberi penghalang, pada gambar 2.1.b. Ketika ada penghalang, maka semut yang berada di depan penghalang tidak dapat melanjutkan dan mengikuti jejak feromon. Yang terjadi kemudian, semut-semut itu bisa memilih untuk belok ke kanan atau ke kiri dengan probabilitas yang sama. Keadaan ini digambarkan seperti pada gambar 2.1. Dengan asumsi semua semut bergerak dengan kecepatan yang hampir sama dan tumpukan jejak feromon rata-rata sama, maka waktu

yang dibutuhkan akan lebih cepat jika melewati rute yang sebelah kiri, dan tumpukan feromon akan lebih tebal dibanding yang sebelah kanan.

Menurut Dorigo dalam (1996) Jika diimplementasikan dengan penghitungan dan gambaran yang lebih sederhana, diperoleh gambaran seperti gambar 2.2.



Gambar 2.2. Alur kerja semut buatan.

Penjelasan gambar 2.2:

- (a). Penginisialisasian jarak pada *graph*.
- (b). Saat $t=0$ belum ada jejak feromon pada semua jalur, maka semut masih memilih lewat jalur kiri atau kanan dengan probabilitas yang sama.
- (c). Saat $t=1$ jalur dengan rute tercepat, memiliki intensitas feromon yang lebih besar, sehingga sebagian semut sudah memulai banyak yang memilih jalur tercepat.

2.2.1 Ant Colony Optimization Algorithm

Diketahui satu set n *Node*, misalkan d_{ij} adalah lama perjalanan dari *node* i ke j . (dengan nilai $d_{ij} = \text{Jarak} / \text{Kecepatan} * \text{Bobot kelas jalan}$). Pembobotan kelas jalan

diberikan berdasarkan lebar jalan *max.* fungsi jalan sesuai dengan tabel 2.1, dan dibobotkan seperti tabel 2.2.

Tabel 2.1 Lebar Jalan berdasarkan fungsi jalan

No	Fungsi Jalan	Lebar (m)
1	Arteri Primer	20 - 50
2	Arteri Sekunder	20 - 50
3	Kolektor Primer	12 - 45
4	Kolektor Sekunder	12 - 25
5	Lokal	6 - 12
<i>Sumber : Dinas Perhubungan Kota Surabaya</i>		

Tabel 2.2 Pembobotan Jalan berdasarkan lebar *max.* fungsi jalan

No	Fungsi Jalan	Pembobotan Fungsi Jalan
1	Arteri Primer	1
2	Arteri Sekunder	1
3	Kolektor Primer	2
4	Kolektor Sekunder	3
5	Lokal	4

$b_i(t)$ ($i=1, \dots, n$) Adalah jumlah semut pada kota i pada saat t dan $m = \sum_{i=1}^n b_i(t)$ adalah jumlah total semut. Tiap semut adalah agen sederhana dengan karakter sebagai berikut :

- Jika memilih kota tujuan dengan suatu probabilitas itu adalah sebuah fungsi dari lama perjalanan di ruas jalan dan jumlah unit feromon pada ruas jalan tersebut.
- Untuk memaksa semut untuk melakukan legal tur, yaitu transisi ke kota yang telah dikunjungi tidak diperbolehkan, sampai tur selesai (yang dikontrol oleh *tabu list*).
- Ketika tur selesai, semut meninggalkan substansi yang disebut jejak feromon pada setiap ruas jalan yang dilewatinya.

Misalkan $\tau_{ij}(t)$ = intensitas feromone pada ruas (i,j) pada waktu ke t . Masing-masing semut pada waktu t memilih kota berikutnya, dimana itu terjadi pada waktu $t+1$.

Oleh karena itu jika melakukan sebuah iterasi dari algoritma AS, m gerakan di bawa oleh m semut dalam interval $(t, t+1)$ kemudian setiap n iterasi dari algoritma (satu *cycle*) setiap semut harus sudah menyelesaikan sebuah tur. Dalam hal ini, intensitas jalur feromon diperbaharui menurut rumus (2.1).

$$\tau_{ij}(t+n) = \rho\tau_{ij}(t) + \Delta\tau_{ij} \dots\dots\dots \text{Rumus (2.1)}$$

dimana, ρ adalah koefisien seperti $(1 - \rho)$ mewakili penguapan (*evaporation*) dari jalur antara waktu t dan $t+n$,

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k \dots\dots\dots \text{Rumus (2.2)}$$

dimana, $\Delta\tau_{ij}^k$ adalah jumlah perunit feromon dari panjang tour yang diletakkan pada ruas (i,j) oleh semut ke k antara waktu t dan $t+n$, seperti rumus (2.3).

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k} \\ 0 \end{cases} \dots\dots\dots \text{Rumus (2.3)}$$

dimana Q adalah konstan dan L_k adalah lama tour dari semut ke k .

Koefisien ρ harus diset dengan nilai < 1 untuk menghindari penimbunan jalur yang tak terbatas.

Dengan tujuan memenuhi batasan dimana semut mengunjungi semua n *node* yang berbeda, dihubungkan dengan masing–masing semut sebuah struktur data bernama *tabu list*, yang menyimpan kota yang sudah dikunjungi sampai waktu t dan melarang semut mengunjunginya lagi sebelum satu *cycle* telah terselesaikan. Setelah satu *cycle* selesai, *tabu list* digunakan untuk menghitung hasil semut saat ini (lama tempuh dari jalur yang dilalui semut). *Tabu list* lalu dikosongkan dan semut bebas untuk memilih lagi. Definisi $tabu_k$ adalah vektor yang tumbuh secara dinamik dimana mengandung *tabu list* dari

semut ke-k, dan $\text{tabu}_k(s)$ adalah elemen ke-s dari daftar (yaitu kota ke-s yang telah dikunjungi oleh semut ke-k pada tur saat ini).

Didefinisikan probabilitas transisi dari kota i ke kota j untuk semut ke-k sebagai rumus (2.4) :

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{k \in \text{allowed}_k} [\tau_{ik}(t)]^\alpha \cdot [\eta_{ik}]^\beta} & \text{if } j \in \text{allowed}_k \\ 0 & \text{otherwise} \end{cases} \quad \dots\dots\dots \text{Rumus (2.4)}$$

dimana, $\text{allowed}_k = \{N - \text{tabu}_k\}$ dan dimana α dan β adalah parameter yang mengontrol kepentingan relatif jalur melawan *visibility*. Oleh karena itu probabilitasnya transisi adalah sebuah pertukaran antara *visibility* (yang mana dikatakan bahwa kota terdekat harus dipilih dengan probabilitas tinggi) dan intensitas jalur pada waktu t.

Algoritma ant cycle ditetapkan, pada saat $t=0$. fase inialisasi dilakukan pada waktu semut diposisikan pada *node-node* yang berbeda dan inisial values $\tau_{ij}(0)$ untuk intensitas jalur diset pada ruas-ruas. Elemen pertama dari *tabu list* masing-masing semut diset sama dengan kota awalnya. Kemudian setiap semut bergerak dari *node* i ke *node* j memilih *node* tujuan dengan probabilitas yang berupa fungsi (dengan parameter α dan β , lihat rumus (2.4)) dari 2 perhitungan yang diinginkan. Yang pertama, jalur $\tau_{ij}(t)$, memberikan informasi tentang berapa banyak semut sebelumnya telah memilih ruas(i,j) yang sama; yang kedua, *visibility* η_{ij} , mengatakan bahwa semakin cepat waktu tempuh ke sebuah *node*, maka semakin di inginkan untuk dipilih.

Setelah n iterasi semua semut telah menyelesaikan tur, dan *tabu list*-nya akan penuh. Pada keadaan ini untuk setiap semut k nilai L_k akan dihitung dan nilai dari $\Delta\tau_{ij}^k$

diperbarui sesuai dengan rumus (2.3). Dan rute tercepat yang ditemukan oleh semut (yaitu L_k , $k=1, \dots, m$) disimpan dan semua *tabu list* dikosongkan. Proses ini teriterasi sampai counter tur mencapai maksimum(*user-defined*) jumlah cycles NC_{MAX} , atau semua semut membuat tur yang sama yang disebut sebagai *stagnation behaviour* karena ia merupakan suatu situasi dimana algoritma berhenti mencari solusi alternatif.

Secara garis besar algoritma *ant cycle* adalah sebagai berikut :

1. Inisialisasi :

Set $t = 0$ {iterasi}

Set $NC = 0$ {cycles iterasi}

Untuk setiap ruas, set nilai initial $\tau_{ij}(t) = C$ untuk intensitas feromon dan $\Delta\tau_{ij} = 0$

Tempatkan m semut pada kota n yang dikehendaki.

2. Set $s = 1$ { s adalah *tabu list* index}

For $k = 1$ to m do

Tempatkan kota asal pada semut ke- k pada $tabu_k(s)$.

3. Ulangi sampai semua semut menemukan solusi

Set $s = s+1$

For $k = 1$ to m do

pilih *node* j sebagai *node* tujuan, dengan probabilitas $\Delta\tau_{ij}^k(t)$.

Pindahkan semut ke k ke dalam kota j .

Masukkan kota j ke dalam $tabu_k$.

4. For $k = 1$ to m do

Pindahkan semut ke k dari $tabu_k(n)$ ke $tabu_k(1)$. (Kembali ke asal)

Hitung panjang L_k perjalanan yang telah dilakukan oleh semut k .

Update perjalanan yang tercepat yang ditemukan.

5. Update intensitas feromon pada setiap jalur.

Set $t = t+n$

Set $NC = NC+1$

6. If ($NC < NC_{MAX}$) and (not stagnation behavior) then

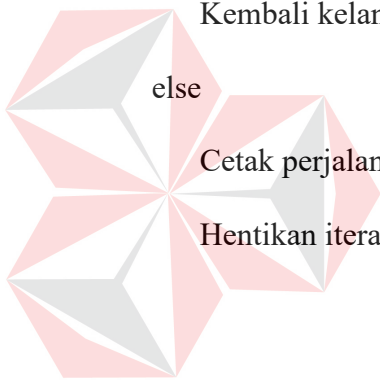
Kosongkan semua *tabu list*.

Kembali kelangkah 2

else

Cetak perjalanan tercepat.

Hentikan iterasi.



UNIVERSITAS
Dinamika

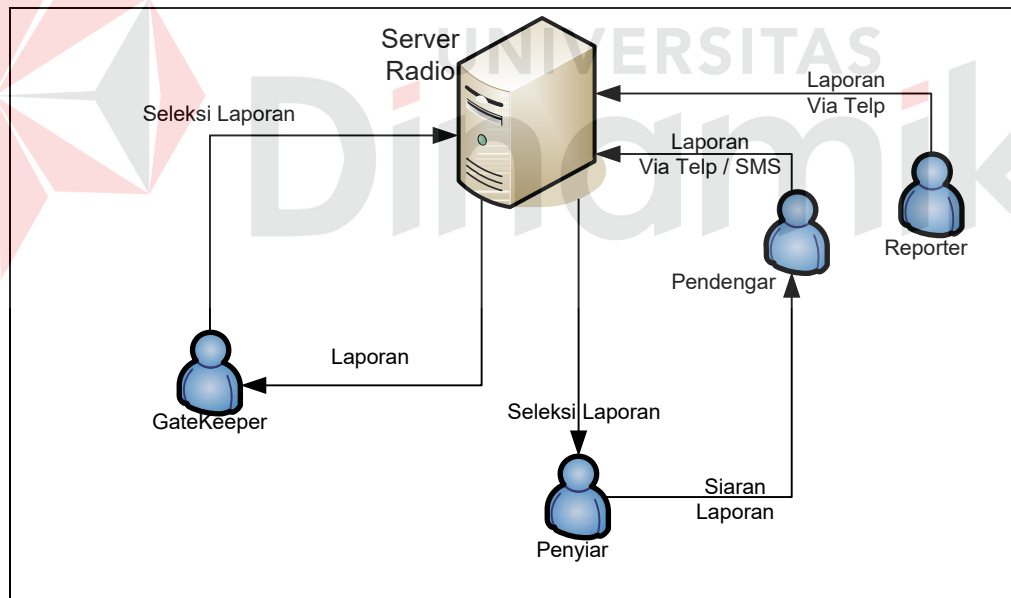
BAB III

METODE PENELITIAN

3.1 Analisa Sistem

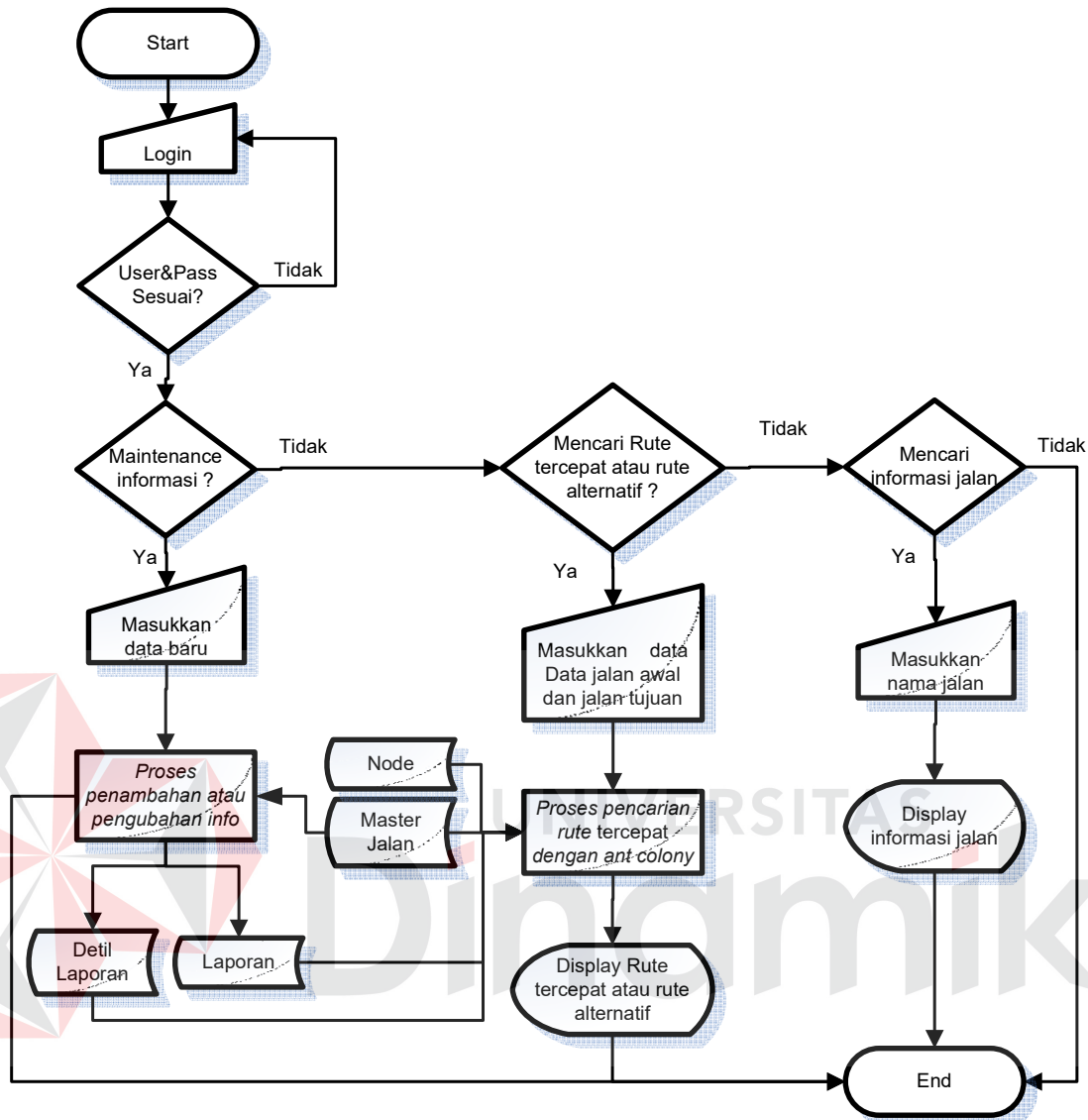
Sistem ini dirancang untuk membantu penyiar untuk menyampaikan informasi kejadian di jalan yang diberikan pendengar lain maupun reporter radio yang turun langsung ke lokasi termasuk efek yang terjadi dari kejadian tersebut, seperti pemblokiran jalur, karena ada kebakaran sehingga jalur itu dialihkan ke jalan lain untuk menghindari kemacetan dan juga untuk membantu PMK dalam memadamkan api.

Sistem aliran data yang ada dalam radio Suara Surabaya, dapat dilihat pada diagram blok gambar 3.1. di bawah ini :



Gambar 3.1 Diagram blok secara umum Sistem aliran data di radio Suara Surabaya.

Langkah-langkah yang dilakukan pada pengembangan sistem ini adalah dengan membuat perancangan sistem dengan merancang diagram alir yang menunjukkan proses pengolahan laporan lalu lintas dan pencarian rute.



Gambar 3.2. Diagram alir secara umum tentang sistem.

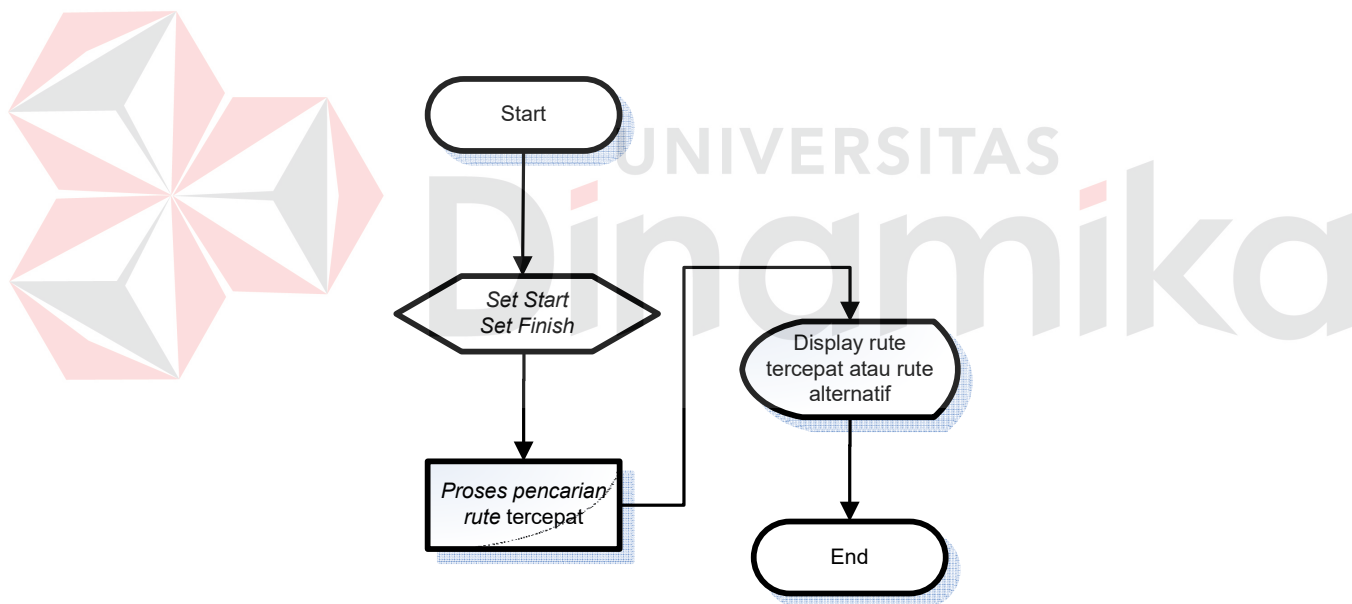
Dalam gambar 3.2. dijelaskan mengenai *maintenance* informasi jalan, pencarian rute tercepat dan pencarian informasi jalan. Dalam *maintenance* informasi jalan pengguna dapat memasukkan data baru (*input*) dan mengubah data lama (*update*). Dalam proses pembuatan rute tercepat dan rute alternatif jalan dilakukan dengan menggunakan metode *ant colony*. Selain itu, pengguna juga dapat melakukan pencarian informasi jalan.

3.2 Perancangan dan Pemodelan Sistem Pencarian Rute

Ada beberapa bagian penting dalam melakukan perancangan dan pemodelan sistem, yaitu melakukan perancangan dan pembuatan alur sistem, DFD dan penyusunan struktur database. Pada bagian selanjutnya, dilakukan pembuatan rancangan entitas-entitas, yang saling berhubungan yang membentuk suatu database relasional.

3.2.1 Alur Proses Pencarian Rute

Untuk lebih memperjelas alur sistem, maka alur proses aplikasi dapat digambarkan pada gambar 3.3, yang menjelaskan rancangan urutan proses yang terjadi pada aplikasi, yaitu:



Gambar 3.3 Alur Proses Pencarian Rute

Pada gambar 3.3 diatas dapat dijelaskan sebagai berikut:

- Pengsetan letak awal dan tujuan semut, sesuai dengan inputan yang diberikan.
- Melakukan proses pencarian rute, dengan membangun rute perjalanan dari letak awal ke tempat tujuan menggunakan peta *virtual*. Peta *virtual* ini terbentuk dari semua

edge (ruas jalan) dan *node* yang ada didalam *database*, dimana *node* berfungsi untuk menghubungkan *edge* yang satu dengan *edge* yang lain.

- c. Mengambil satu rute yang tercepat, dari semua rute optimal yang telah ditemukan.
- d. Setelah rute ditemukan, sistem akan menampilkan informasi rute tercepat kepada pemakai.

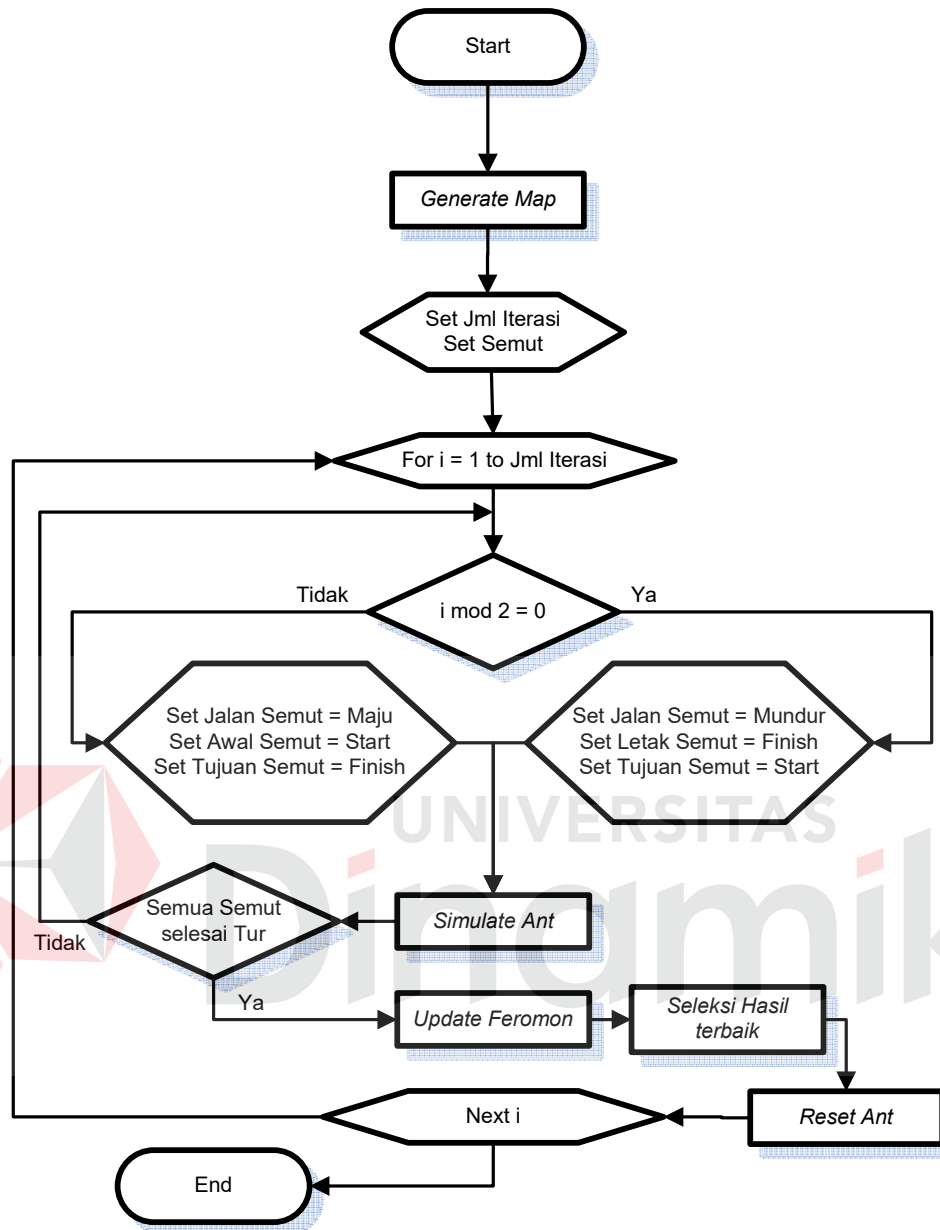
3.2.2 Penerapan Algoritma Ant Colony Optimization

Dalam pencarian rute terpendek atau rute alternatif, akan menggunakan algoritma *Ant Colony Optimization*. Pencarian rute ini didasarkan pada waktu tempuh yang diperlukan untuk mencapai tempat tujuan. Pada sub bab ini akan dijelaskan secara praktis tahap-tahap proses pencarian rute yang dilakukan aplikasi menggunakan *Ant Colony Optimization*.

A. Proses Pencarian Rute

Dimulai dari proses *generate map*, yaitu proses untuk membangun peta secara virtual yang nanti akan digunakan oleh *artificial ant* dalam membangun perjalanannya. Kemudian mengeset jumlah iterasi dan mengeset semut, yaitu menginisialisasi letak awal perjalanan dan letak tujuan.

Proses selanjutnya yaitu proses *simulate ant* yaitu mensimulasikan semut sampai semua semut sampai pada tujuan. Kemudian dilanjutkan dengan proses meng-*update* feromon, menyimpan *result* yang terbaik dari hasil yang didapatkan oleh semua semut dan me-*reset* ulang semut dengan mengeset ulang semut seperti kondisi awal. Proses ini terus dilakukan sebanyak jumlah iterasi.

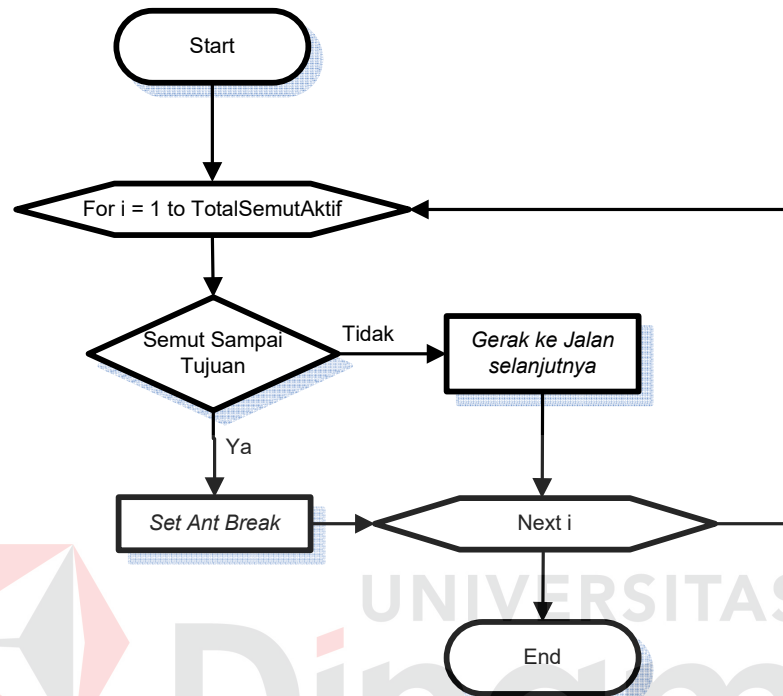


Gambar 3.4 Flowchart Algoritma Ant Colony Optimization

B. Proses Simulate Ant

Proses ini menunjukkan bagaimana artificial ant membangun rute perjalanannya untuk menemukan tujuan yang telah ditentukan. Dimulai dari proses inialisasi *Start* yang menunjukkan posisi awal dimana lokasi semut memulai turnya dan inialisasi *Finish* yang menunjukkan lokasi tujuan akhir tur semut. Proses ini

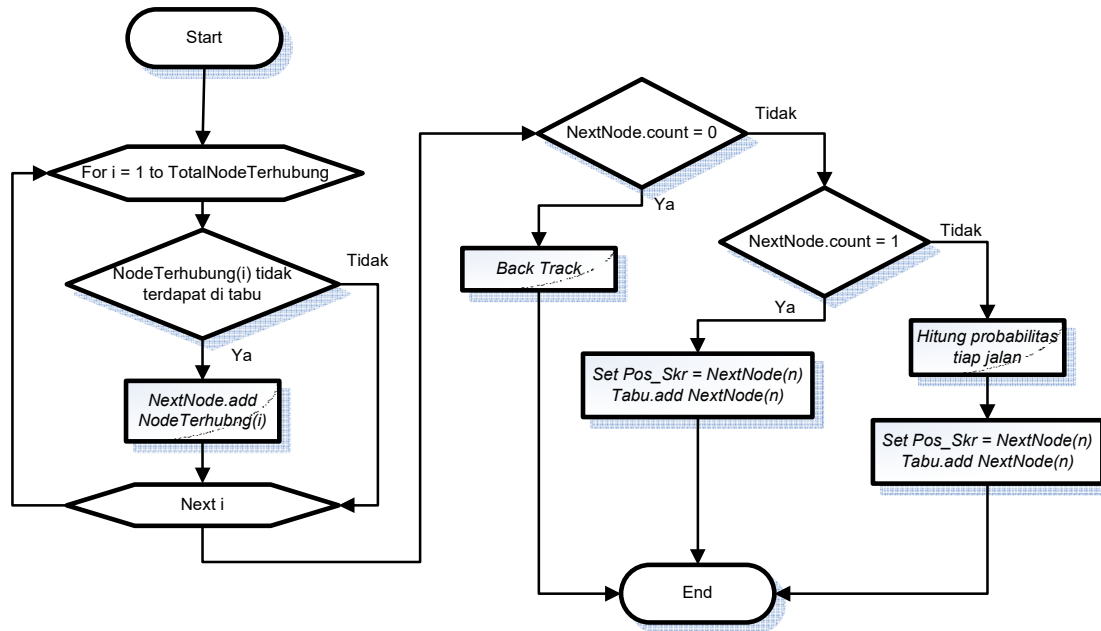
menjalankan semua semut satu-persatu menuju jalan berikutnya. Apabila semut sudah sampai pada tujuan maka semut akan diset *Break*, dimana semut tidak melakukan pergerakan lagi.



Gambar 3.5 Flowchart Proses Simulate Artificial Ant

C. Proses Pergerakan ke Jalan Selanjutnya

Proses ini menunjukkan bagaimana setiap *artificial ant* melakukan perhitungan probabilitas dan eksplorasi memori untuk memilih jalur yang harus ditempuh dalam membangun perjalanannya.



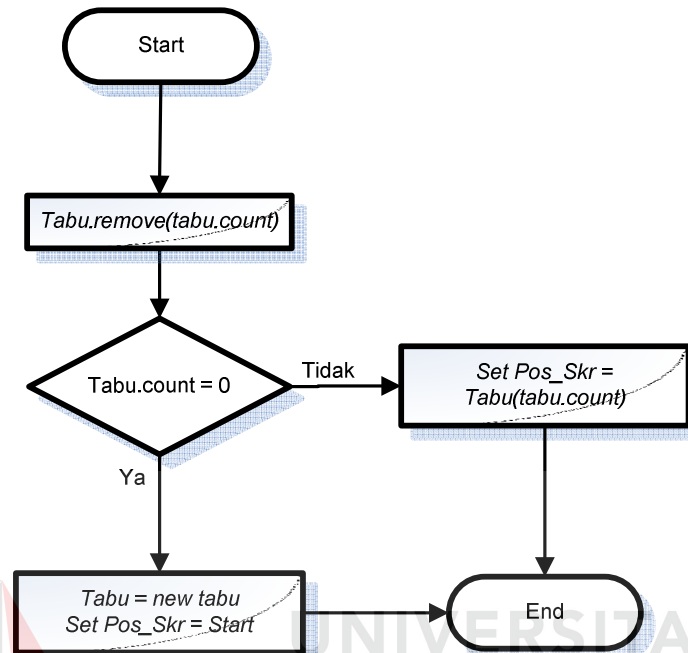
Gambar 3.6 Flowchart Proses pergerakan ke jalan selanjutnya.

Perobabilitas pemilihan jalan dihitung dengan formula *Random Propotional Rule*. Parameter yang berpengaruh dalam perhitungan probabilitas adalah tingkat intensitas feromon dan lama tempuh jalan itu sendiri. Proses dimulai dari inialisasi Tabu yang digunakan sebagai memori semut untuk menyimpan data jalan yang pernah dilalui oleh semut. Hal ini dilakukan agar semut tidak melewati jalan yang sama. Ketika semut menemui percabangan jalan , maka akan dilakukan perhitungan probabilitas *Random Propotional Rul*, jika tidak ada percabangan, maka akan langsung ke jalan selanjutnya, tetapi jika tidak ditemukan jalan berikutnya (Jalan buntu), maka semut akan kembali ke jalan sebelumnya (*Back Track*).

D. Proses Back Track

Proses ini digunakan untuk menangani masalah ketika suatu semut menemui jalan buntu yaitu dengan memindahkan semut ke langkah sebelumnya dan memilih jalur

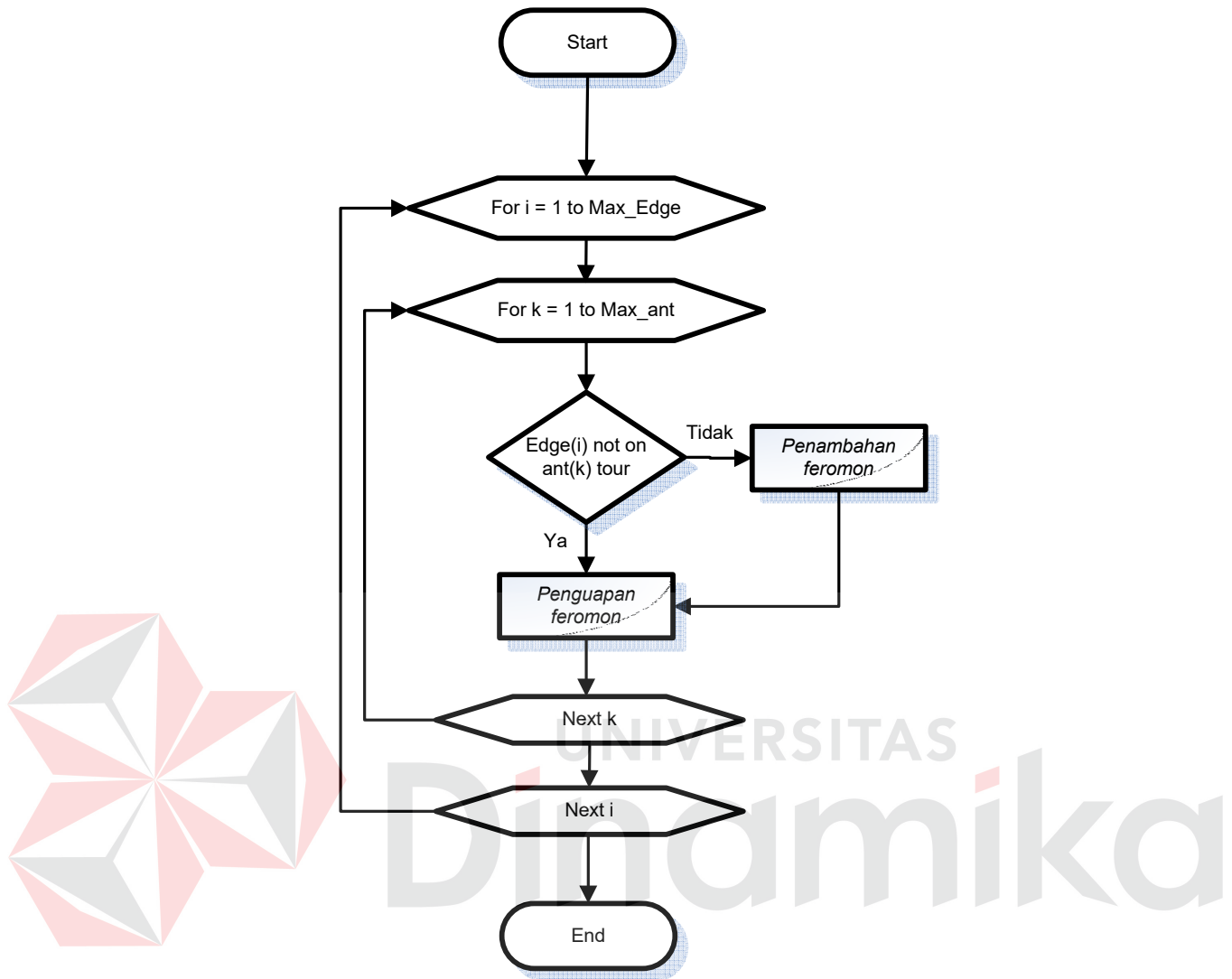
lain yang belum pernah dilalui. Apabila sampai kembali ke posisi awal, maka semut akan diset ulang dari awal.



Gambar 3.7 Flowchart Proses Back Track

E. Proses Update Feromon

Roses ini menggambarkan bagaimana *artificial ant* menandai rute yang telah dilewati dengan menambahkan sejumlah feromon ke jalur yang hanya dilewati ketika membangun perjalanannya. feromon ini digunakan sebagai petunjuk bagi semut yang lainnya dalam memilih jalan yang akan ditempuh dan juga terdapat proses *evaporation* (penguapan) feromon yang membuat kadar feromon disuatu *edge* (ruas jalan) semakin menipis. Proses penguapan terjadidisetiap *edge*. Proses ini dijalankan setelah semua semut telah menyelesaikan perjalanannya.



Gambar 3.8 Flowchart Proses Update feromon

3.3 Perancangan Sistem

Sebelum membuat program aplikasi, terlebih dahulu dilakukan proses perancangan sistem. Hal ini dilakukan supaya aplikasi yang dibuat dapat berfungsi sesuai dengan yang diharapkan sehingga mampu menghasilkan pencarian rute yang tercepat.

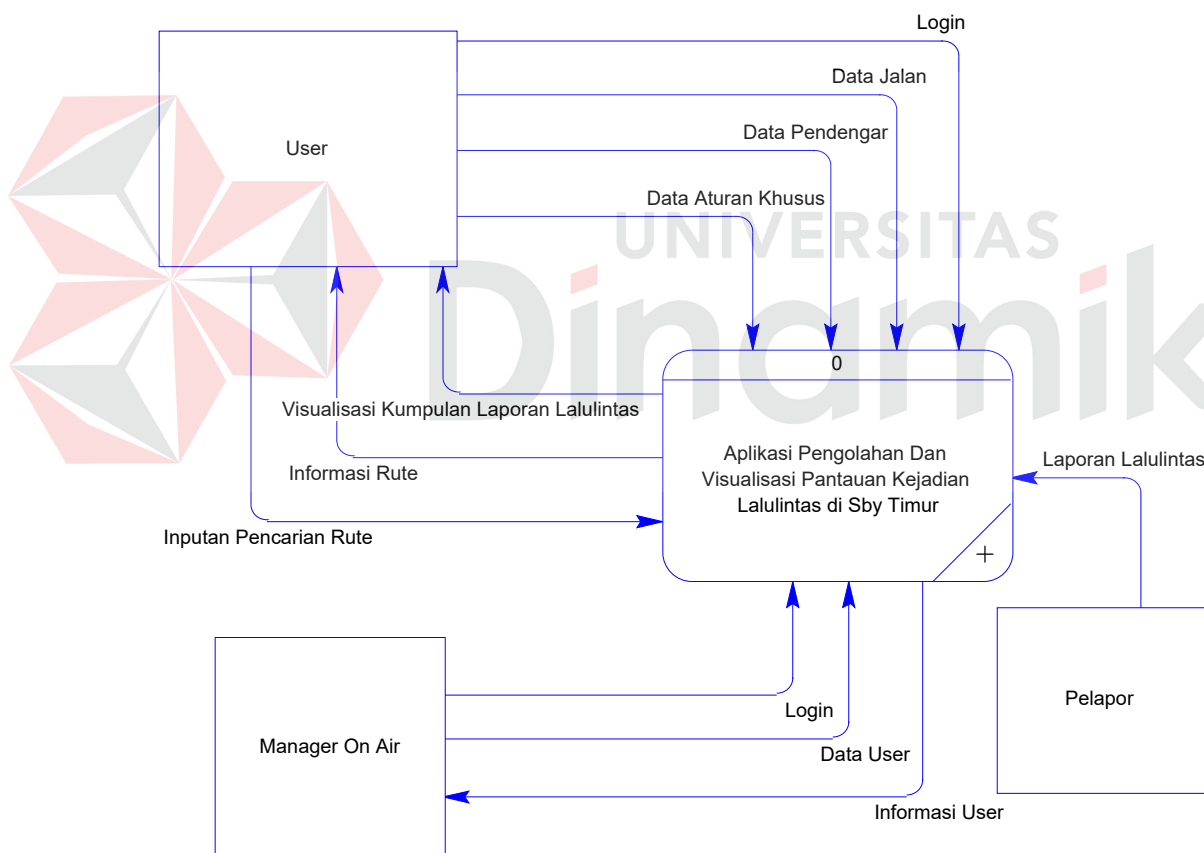
Dalam perancangan sistem ini ada beberapa tahapan-tahapan yang harus dilakukan. Adapun tahapan-tahapan dalam perancangan sistem yang dilakukan adalah

pembuatan *Data Flow Diagram* (DFD), *Entity Relationship Diagram* (ERD), dan *Struktur Database*.

3.3.1 Data Flow Diagram

Data flow diagram digunakan untuk analisa desain sistem yang menggambarkan sistem secara garis besar dan memecahnya menjadisub bagian-sub bagian yang lebih terperinci.

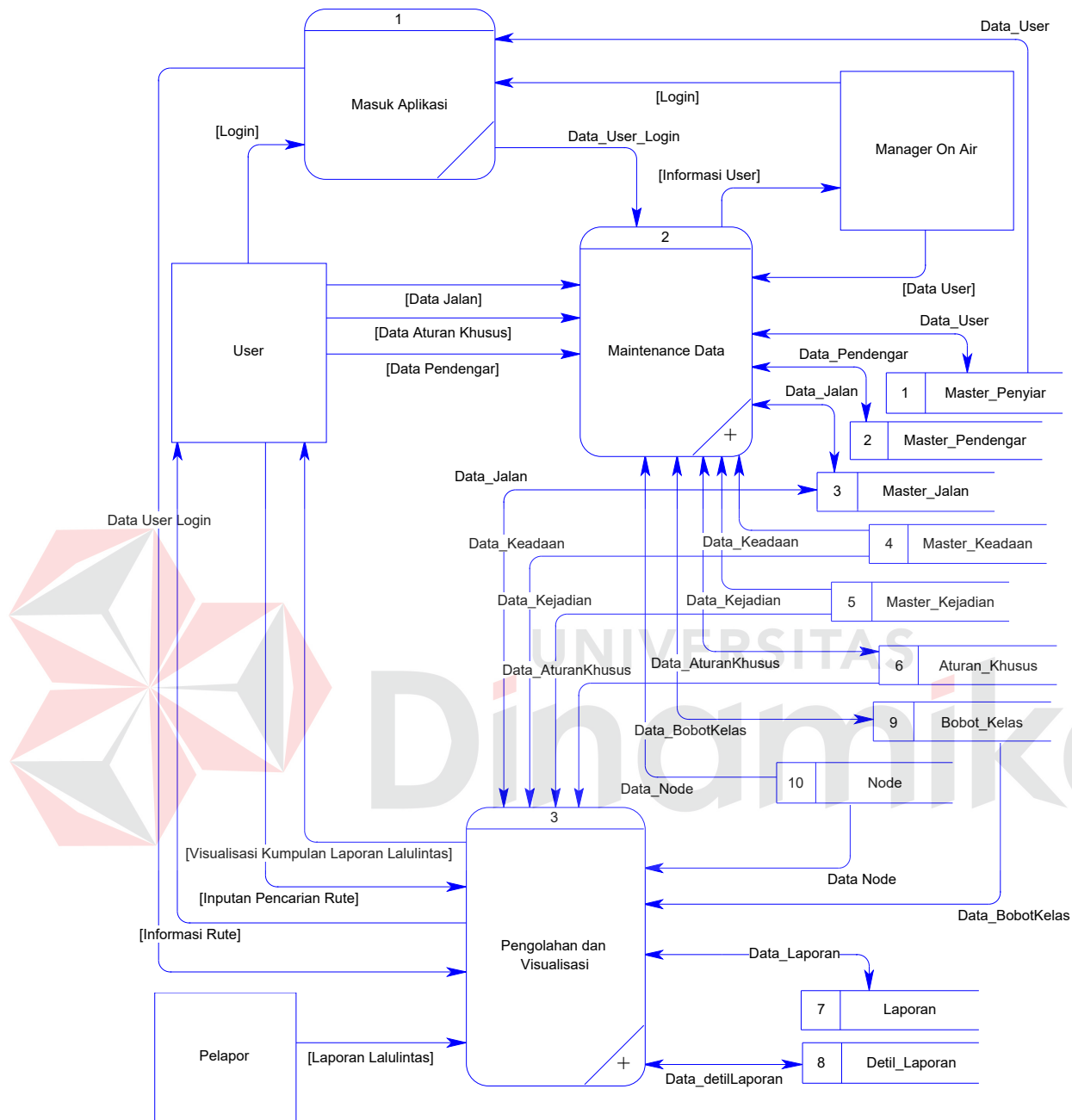
A. Context Diagram



Gambar 3.9 Context Diagram

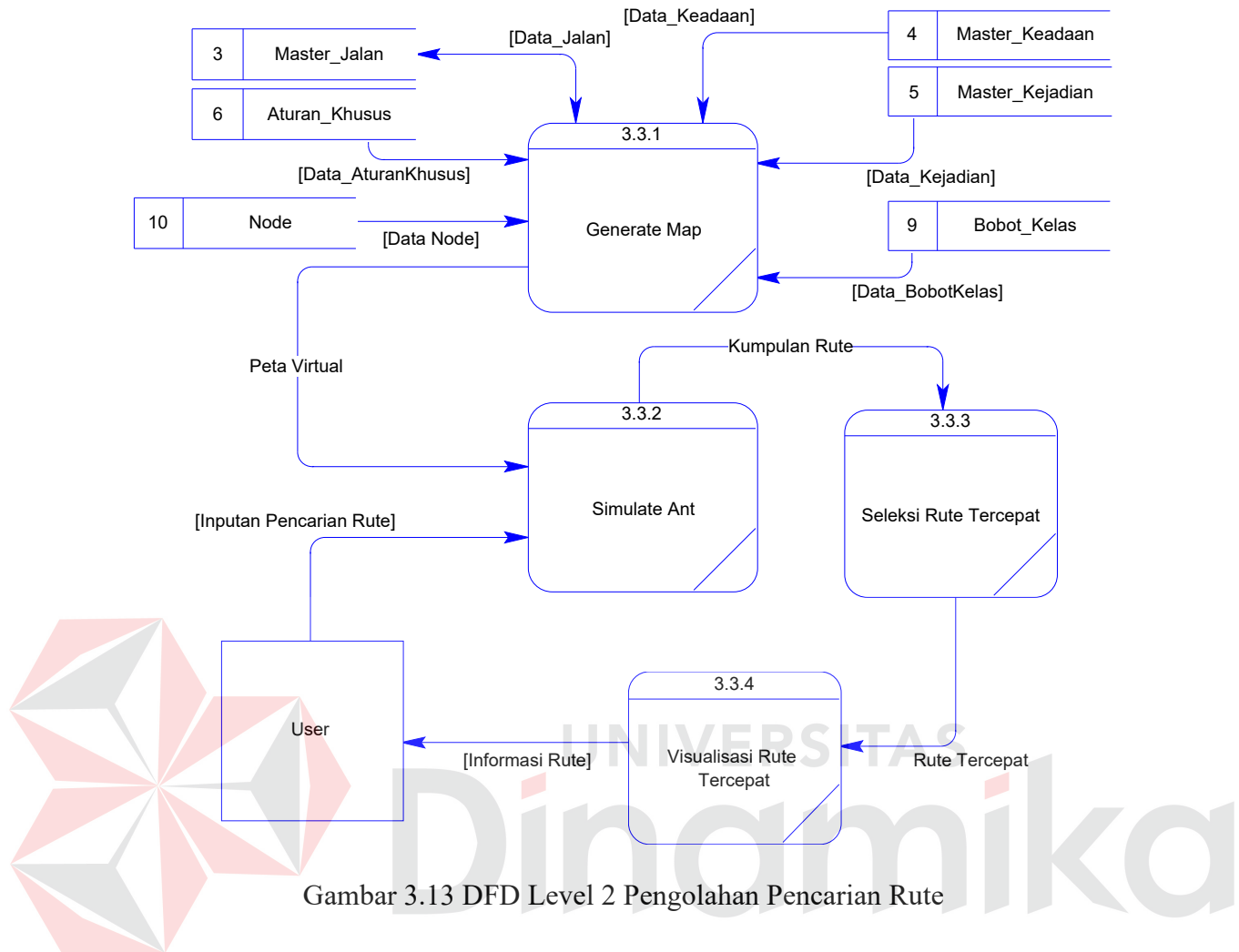
Context diagram pada gambar 3.9 menggambarkan hubungan sistem dengan tiga *entity*, yaitu Pelapor, *User*, dan *Manager OnAir*.

B. Data Flow Diagram Level 0



Gambar 3.10 Data Flow Diagram Level 0

Hasil penurunan proses dari *context diagram*, terdapat tiga proses yaitu masuk aplikasi, *maintenance data*, dan proses pengolahan dan visualisasi.

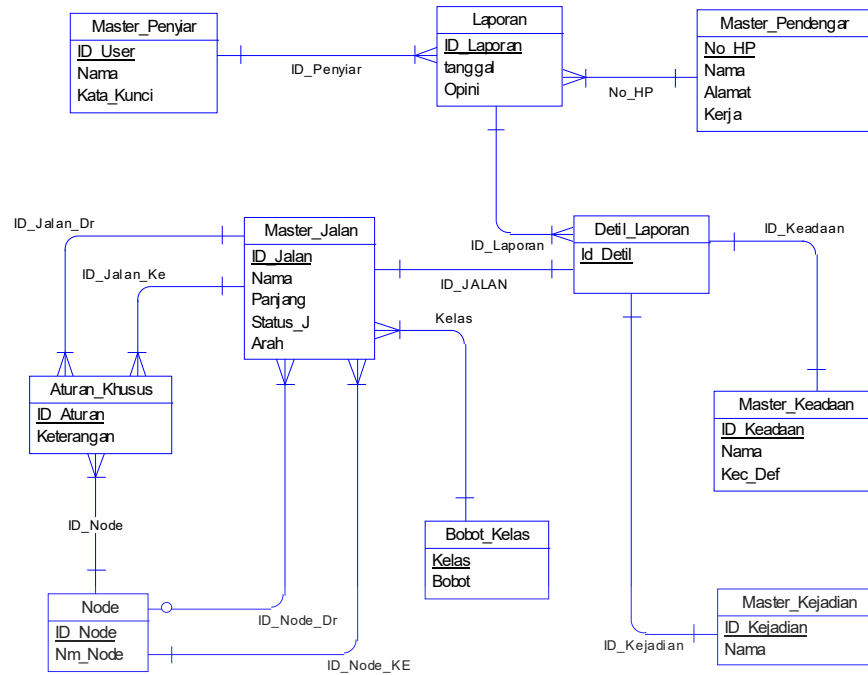


Gambar 3.13 DFD Level 2 Pengolahan Pencarian Rute

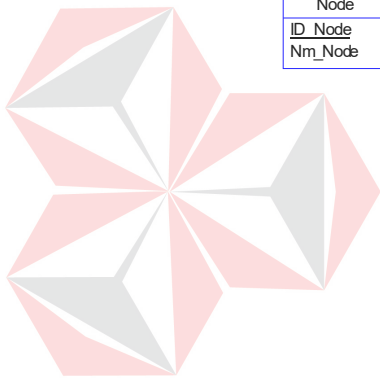
3.3.2 Entity Relational Diagram

Entity Relational Diagram adalah diagram yang menggambarkan hubungan antara tabel–tabel.

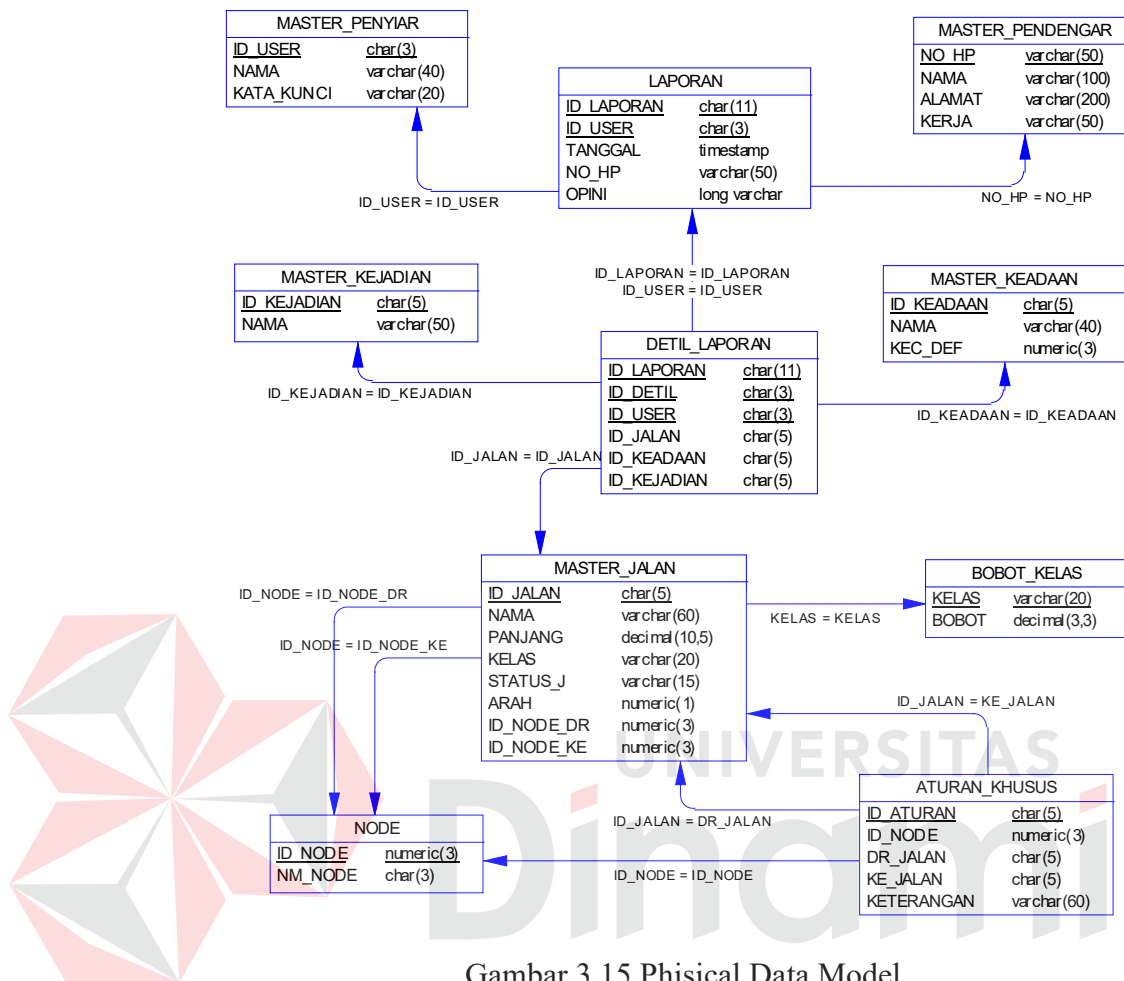
A. Conceptual Data Model (CDM)



Gambar 3.14 Conceptual Data Model



B. Physical Data Model (PDM)



Gambar 3.15 Physical Data Model

3.3.3 Struktur Table

Rincian dari ERD digambarkan dengan struktur database yang terdiri atas kolom-kolom yang memiliki atribut berupa nama *field*, tipe data, ukuran dan keterangan. Struktur database menunjukkan daftar kebutuhan tabel yang digunakan untuk menyimpan data yang diperlukan dalam sistem.

1. Tabel *Node*

Primary Key : *ID_Node*

Foreign Key : -

Fungsi : Menyimpan *node* yang ada.

Tabel 3.1 Tabel *Node*

Field	Tipe Data	Ukuran	Keterangan
<i>ID_Node</i>	Number	3	PK
<i>Nm_Node</i>	Char	3	NN

2. Tabel *Master_Jalan*

Primary Key : *ID_Jalan*

Foreign Key : *T_Awal - Node, T_Akhir - Node*

Fungsi : Menyimpan data jalan.

Tabel 3.2 Tabel *Master_Jalan*

Field	Tipe Data	Ukuran	Keterangan
<i>ID_Jalan</i>	Char	5	PK
Nama	Varchar	60	NN
Panjang	Decimal	10.5	NN
Kelas	Varchar	20	NN
Status_J	Varchar	15	NN
Arah	Int	1	NN
<i>T_Awal</i>	Int	3	FK, NN
<i>T_Akhir</i>	Int	3	FK, NN

3. Tabel *Master_Keadaan*

Primary Key : *ID_Keadaan*

Foreign Key : -

Fungsi : Menyimpan data keadaan jalan.

Tabel 3.3 Tabel Master_Keadaan

Field	Tipe Data	Ukuran	Keterangan
ID Keadaan	Char	5	PK
Nama	Varchar	50	NN
Kec_Def	Int	3	NN

4. Tabel Master_Kejadian

Primary Key : ID_Kejadian

Foreign Key : -

Fungsi : Menyimpan data kejadian jalan.

Tabel 3.4 Tabel Master_Kejadian

Field	Tipe Data	Ukuran	Keterangan
ID_Kejadian	Char	5	PK
Nama	Varchar	50	NN

5. Tabel Master_User

Primary Key : ID_User

Foreign Key : -

Fungsi : Menyimpan Data User

Tabel 3.5 Tabel Master_User

Field	Tipe Data	Ukuran	Keterangan
ID_User	Char	3	PK
Nama	Varchar	40	NN
Kata_Kunci	Varchar	20	NN

6. Tabel Master_Pendengar

Primary Key : No_HP

Foreign Key : -

Fungsi : Menyimpan Data Pendengar

Tabel 3.6 Tabel Master_Pendengar

Field	Tipe Data	Ukuran	Keterangan
No HP	Varchar	50	PK
Nama	Varchar	100	NN
Alamat	Varchar	200	NN
Kerja	Varchar	50	NN

7. Tabel Bobot_Kelas

Primary Key : Kelas

Foreign Key : Kelas – Master_Jalan

Fungsi : Menyimpan Data Jalan.

Tabel 3.7 Tabel Bobot_Kelas

Field	Tipe Data	Ukuran	Keterangan
Kelas	Char	20	PK, FK
Bobot	Double		NN

8. Tabel Aturan_Khusus

Primary Key : ID_Aturan

Foreign Key : *Node* – *Node*, Dr_Jalan – Master_Jalan, Ke_Jalan – Master_Jalan

Fungsi : Menyimpan data aturan khusus melintas dari ruas jalan satu ke ruas jalan yang lain.

Tabel 3.8 Tabel Aturan Khusus

Field	Tipe Data	Ukuran	Keterangan
ID_Aturan	Char	5	PK
<i>Node</i>	Number	3	NN, FK
Dr Jalan	Number	5	NN, FK
Ke Jalan	Number	5	NN, FK
Keterangan	Varchar	60	NN

9. Tabel Laporan

Primary Key : ID_Laporan, GKeeper

Foreign Key : GKeeper – Master_Penyiar, ID_HP – Master_Pendengar

Fungsi : Menyimpan Data Laporan

Tabel 3.9 Tabel Laporan

Field	Type Data	Ukuran	Keterangan
ID_Laporan	Char	11	PK
ID_User	Char	3	PK, FK
Tanggal	Time		NN
ID_HP	Varchar	50	NN, FK
Opini	Text		

10. Tabel Detil_Laporan

Primary Key : ID_Laporan, ID_Detil, GKeeper

Foreign Key : (ID_Laporan, GKeeper)-Laporan, ID_Jalan-Master_Jalan,
ID_Keadaan-Master_Keadaan, ID_Kejadian-Master_Kejadian

Fungsi : Menyimpan Data Detil Laporan

Tabel 3.10 Tabel Detil_Laporan

Field	Type Data	Ukuran	Keterangan
ID_Laporan	Char	11	PK, FK
ID_Detil	Char	3	NN
ID_User	Char	3	NN, FK
ID_Jalan	Char	5	FK
ID_Keadaan	Char	5	FK
ID_Kejadian	Char	5	FK

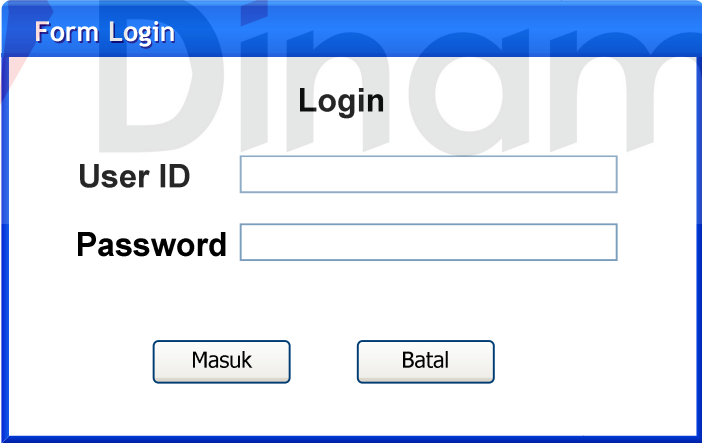
3.4 Disain Input dan Output

Untuk menjalankan aplikasi pengolahan dan visualisasi ini dibutuhkan beberapa form yang digunakan sebagai sarana untuk melakukan proses prediksi.

3.4.1 Desain Form Login

Form ini digunakan untuk mengotorisasi user yang akan menggunakan aplikasi ini. Dalam aplikasi ini, user hanya dibagi menjadidua bagian, bagian pertama adalah Manager Online, dan bagian user umum (Penyiar dan *Gatekeeper*). Setelah login dengan form ini, maka user akan masuk ke form utama, Seperti pada gambar 3.20. Langkah-langkah penggunaan form ini adalah sebagai berikut:

1. Isi kotak isian *User ID* dan *Password* sesuai dengan *User ID* dan *Password* yang telah diberikan oleh *Manager Online*.
2. Setelah kedua kotak isian terisi, tekan tombol “Masuk” untuk menjalankan aplikasi ini.
3. Tombol “Batal” digunakan jika *user* ingin keluar dari form *login*

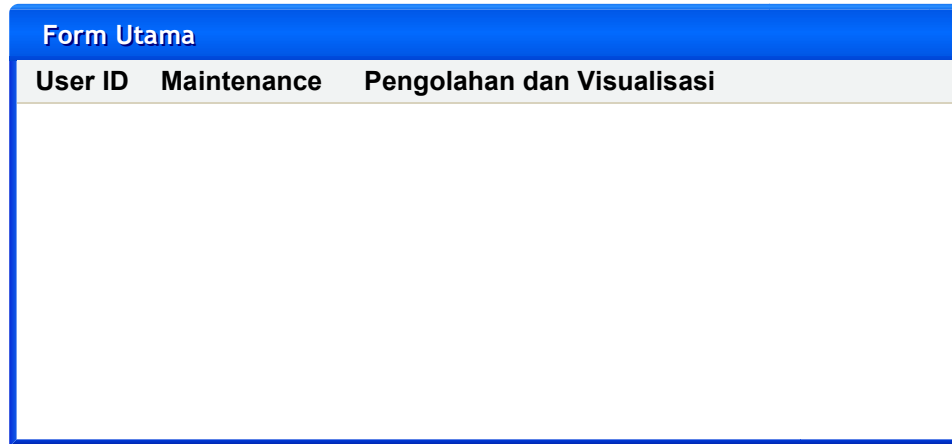


The image shows a screenshot of a login form window. The window has a blue title bar with the text "Form Login". Inside the window, the word "Login" is centered at the top. Below it, there are two input fields: "User ID" and "Password". At the bottom of the form, there are two buttons: "Masuk" (Login) and "Batal" (Cancel). The form is overlaid on a background that includes a watermark of a red and white geometric logo and the text "UNIVERSITAS Dinamika".

Gambar 3.16 Desain Form Login

3.4.2 Desain Form Utama

Form ini muncul, setelah user berhasil masuk lewat form login. Form ini dapat menghubungkan *user* ke form-form yang lain, melalui *toolbars*-nya.



User ID	Maintenance	Pengolahan dan Visualisasi
---------	-------------	----------------------------

Gambar 3.17 Desain Form Utama

3.4.3 Desain Form Ganti Password

Form ini digunakan jika user ingin mengganti pasword. User hanya bisa mengganti *password*-nya sendiri. Langkah-langkah penggunaan form ini adalah sebagai berikut:

1. Isi password yang lama yang ingin diganti, ke dalam kotak isian Password lama.
2. Isi password pengganti, kedalam kotak isian Password Baru dan Password Conf, dengan isian yang sama.
3. Setelah terisi semua, tekan tombol “Ganti”.
4. Setelah selesai, tekan tombol “Selesai”, untuk kembali ke form utama.

Gambar 3.18 Desain Form Ganti Password

3.4.4 Desain Form Masukkan Laporan

Form ini digunakan oleh user untuk memasukkan laporan-laporan yang masuk, baik dari pendengar langsung maupun dari reporter. Langkah-langkah penggunaan form ini adalah sebagai berikut:

1. Isikan Nomer telepon dari pelapor dikotak isian Telepon/HP, atau tekan tombol "..."
(*list view*) yang ada disebelah kanan kotak isian tersebut. Bila data nomer telepon ada didalam database, maka otomatis data pelapor akan keluar. Jika tidak ada maka, user bisa langsung mengisi keterangan pelapor langsung diform ini, dan otomatis data pelapor akan tersimpan.
2. Setelah data pelapor terisi, user bisa mengisi langsung keadaan dan kejadian lalulintas. user dapat memilih keadaan dan kejadian lalulintas yang telah disediakan pada *combobox*, sebelum memilih jalan yang dimaksud didalam peta yang telah disediakan, dengan sebelumnya menekan tombol "i", untuk mengambil data jalan dari

dalam peta. Jalan-jalan yang kita pilih akan langsung tertulis lengkap beserta keadaan dan kejadian lalulintas sesuai dengan yang dilaporkan.

3. Tombol “<”, berfungsi untuk menghapus pilihan jalan terakhir, yang telah user pilih tadi. dan Tombol “Bersih” berfungsi untuk menghapus semua pilihan jalan yang telah dipilih.
4. Bila pelapor ingin memberikan opini diluar masalah lalulintas, user dapat mengisinya didalam kotak isian “Opini”
5. Setelah laporan selesai, tekan tombol “Simpan” untuk menyimpan laporan dedalam database.
6. Tombol “Edit” digunakan untuk mengubah laporan yang terakhir disimpan.
7. Tombol “Batal” digunakan untuk membersihkan kotak isian, dari isian yang keliru.
8. Tombol “Keluar” digunakan untuk kembali ke form utama.
9. Untuk mengeksplor peta, disediakan beberapa tombol yang berfungsi untuk memperbesar gambar peta, dangan menekan tombol “+”, memperkecil peta dengan menekan tombol “-”, sedangkan tombol “R” digunakan untuk mengembalikan sekala peta ke kondisi semula, dan bila user ingin menggeser peta, dengan menekan tombol “H”, setelah itu, user dapat menggeser peta sesuai dengan keinginan.

Gambar 3.19 Desain Form Masukan Laporan

3.4.5 Desain Form Maintenance Data Jalan

Form ini digunakan oleh user untuk mengubah data jalan, bila ada perubahan yang dilakukan oleh pihak yang berwenang. Seperti perubahan arah arus jalan, kelas jalan, dan lain-lain. Langkah-langkah penggunaan form ini adalah sebagai berikut:

1. Tekan tombol “i”, lalu pilih jalan yang ada dipeta, dengan mengklik langsung jalan yang ada dipeta.
2. Setelah itu, data jalan yang dipilih tadi akan tampil, dan user bisa merubah data jalan tersebut.
3. Tekan tombol “Edit” jika ingin menyimpan perubahan data yang telah dilakukan oleh user.

4. Tombol “Batal” digunakan jika ingin membersihkan isian yang ada dikotak isian.
5. Tombol “Keluar” digunakan untuk kembali ke form utama.

The image shows a web form titled "Form Maintenance Data Jalan". At the top, there are labels for "Tanggal" and "Jam". Below these is a large map area labeled "PETA". To the right of the main map is a smaller map window also labeled "PETA" with navigation controls (+, R, -, H, i). Below the maps are several input fields: "ID Jalan" (text), "Nama Jalan" (text), "Kelas" (dropdown), "Status" (dropdown), "Panjang" (text), "Ruas" (dropdown), "Arah" (dropdown), "Dari Jalan" (text), and "Ke Jalan" (text). At the bottom of the form are three buttons: "Edit", "Batal", and "Keluar".

Gambar 3.20 Desain Form Maintenance Data Jalan

3.4.6 Desain Form Maintenance Data Pendengar

Form ini digunakan oleh user untuk mengubah data pendengar, bila pada isian awal diform masukan laporan terjadikesalahan penulisan, atau data pendengar telah berubah. Langkah-langkah penggunaan form ini adalah sebagai berikut:

1. Isikan nomer telepon, jika nomer sudah ada dalam database maka data dari nomer itu akan keluar dan tombol “Edit” akan berubah menjaditombol “Hapus”, dan user bisa mengedit data lalu menekan tombol “Simpan” atau menghapus data pendengar dari database.

2. Tombol “Edit” digunakan untuk menampilkan list view dari data pendengar, untuk memilih salah satu record untuk diedit atau dihapus.
3. Tombol “Batal” digunakan untuk membersihkan semua kotak isian.
4. Tombol “Keluar” digunakan untuk kembali ke form utama.

The image shows a web form titled "Form Maintenance Data Pendengar". It features a blue header bar with the title in white text. Below the header, there are four input fields, each with a label to its left: "Telepon / HP", "Nama", "Alamat", and "Kerja". At the bottom of the form, there are four buttons: "Simpan", "Edit", "Batal", and "Keluar". The form is overlaid on a background that includes a large, semi-transparent watermark of the logo of Universitas Dinamika.

Gambar 3.21 Desain Form Maintenance Data Pendengar

3.4.7 Desain Form Maintenance Aturan Khusus

Form ini digunakan oleh user untuk memberikan aturan khusus dilarang melintas dari suatu jalan ke jalan yang lain. seperti disalah satu perempatan jalan besar, kita tidak boleh untuk belok kanan, tapi harus terus atau belok kiri. Langkah-langkah penggunaan form ini adalah sebagai berikut:

1. Tekan tombol “i” yang ada disebelah kanan kotak isian “Dari jalan” untuk memilih jalan yang ada dalam peta, dan dimasukkan ke dalam kotak isian “Dari jalan”.
2. Tekan tombol “i” yang ada disebelah kanan kotak isian “Ke jalan” untuk memilih jalan yang ada dalam peta, dan dimasukkan ke dalam kotak isian “Ke jalan”.
3. Setelah kedua isian terisi, tekan tombol “Simpan” untuk menyimpan aturan yang telah dibuat tadi.

4. Tekan tombol “Edit” untuk menampilkan list view, pilih salah satu record, dan tombol “Edit” akan berubah menjaditombol “Hapus”, yang digunakan untuk menghapus aturan khusus yang telah dibuat.
5. Tombol “Keluar” digunakan untuk kembali ke form utama.

Gambar 3.22 Desain Form Maintenance Aturan Khusus

3.4.8 Desain Form Maintenance Data User

Form ini digunakan oleh Manager On Air untuk mengubah, menambahkan, atau menghapus user yang dapat mengakses aplikasi ini. Status yang dapat diberikan kepada user lain adalah setatus sebagai “Penyiar dan Gatekeeper” atau status sebagai “Manager On Air”. Langkah-langkah penggunaan form ini adalah sebagai berikut:

1. Isikan kotak isian User ID, jika didalam database sudah ada data dengan user ID itu, maka semua data user itu akan keluar secara otomatis. Manager dapat mengubah isian data user yang dipilih. Tombol “Simpan” digunakan untuk menyimpan hasil *edit*-an.
2. Jika User ID yang dimasukkan tidak ada didalam database, maka manager bisa menggunakan user id tersebut untuk membuat user baru.
3. Tombol “Edit” digunakan untuk menampilkan list view dari data Penyiar, untuk memilih salah satu record untuk diedit atau dihapus.
4. Tombol “Batal” digunakan untuk membersihkan semua kotak isian.
5. Tombol “Keluar” digunakan untuk kembali ke form utama.

Gambar 3.23 Desain Form Maintenance Data User

3.4.9 Desain Form Pengolahan Pencarian Rute Tercepat Alternatif (Untuk Gatekeeper)

Form ini digunakan oleh user, khususnya gatekeeper untuk mencari rute tercepat atau rute alternatif. Langkah-langkah penggunaan form ini adalah sebagai berikut:

1. Tekan tombol “i” yang ada disebelah kanan kotak isian “Jalan Awal” untuk memilih jalan yang ada dalam peta, dan dimasukkan ke dalam kotak isian “Jalan Awal”.
2. Tekan tombol “i” yang ada disebelah kanan kotak isian “Jalan Tujuan” untuk memilih jalan yang ada dalam peta, dan dimasukkan ke dalam kotak isian “Jalan Tujuan”.
3. Setelah kedua isian terisi, tekan tombol “Pencarian Rute” untuk mencari rute yang diinginkan.

The screenshot shows a web application window titled "Form Pencarian Rute Tercepat Alternatif". The interface is divided into several sections:

- Top Left:** Fields for "Tanggal" (Date) and "Jam" (Time).
- Map Area:** A large map area on the left with the word "PETA" overlaid. A smaller map window on the right also displays "PETA".
- Search Panel (Right):**
 - Buttons for map navigation: "+", "R", "-", and "H".
 - "Jalan Awal" input field with an "i" button to the right.
 - "Jalan Tujuan" input field with an "i" button to the right.
 - "Pencarian Rute" button.
 - A scrollable area below the buttons.
- Bottom Right:** "Panjang Tempuh : Meter".

Gambar 3.24 Desain Form Pencarian Rute Tercepat Alternatif (Untuk Gatekeeper)

3.4.10 Desain Form Pengolahan Pencarian Rute Tercepat Alternatif (Untuk Penziar)

Sama seperti form pada sub bab 3.4.9, Form ini diperuntukkan khusus untuk penziar, perbedaannya ukuran font yang lebih besar untuk membantu penziar dalam membaca rute.

The image shows a web application interface for finding alternative routes. The window title is "Form Pencarian Rute Tercepat Alternatif". On the left, there are labels for "Tanggal" and "Jam" above a large map area. The word "PETA" is written in large letters across the map. On the right side, there are four buttons: "+", "R", "-", and "H". Below these are two input fields: "Jalan Awal" and "Jalan Tujuan", each with a small "i" button to its right. A "Pencarian Rute" button is positioned below the "Jalan Tujuan" field. At the bottom right, there is a "Panjang Tempuh" label followed by an input field and the word "Meter". A large, semi-transparent watermark for "UNIVERSITAS Dindamika" is visible in the background.

Gambar 3.24 Desain Form Pencarian Rute Tercepat Alternatif (Untuk Gatekeeper)

3.4.10 Desain Form Visualisasi Pantauan Kejadian Lalulintas

Form ini digunakan oleh user untuk melihat keadaan jalan berdasarkan beberapa laporan yang masuk. Langkah-langkah penggunaan form ini adalah sebagai berikut:

1. Beberapa laporan akan ditampilkan ke dalam list view, dimulai dari yang terbaru.
2. Beberapa laporan itu akan ditampilkan juga dipeta, dengan diwakilkan oleh simbol, per kejadian.

The image shows a web application window titled "Form Visualisasi Pantauan Kejadian Lalulintas". The interface includes a date and time selection section labeled "Tanggal" and "Jam". Below this is a large map area labeled "PETA". To the right of the main map is a smaller map area also labeled "PETA", which includes navigation controls (arrows and a zoom slider) and a set of buttons labeled "+", "R", "-", "H", and "i". Below the smaller map is a table titled "Laporan terakhir" with several empty rows. A large, semi-transparent watermark for "UNIVERSITAS DINDAMIKA" is overlaid on the bottom half of the image.

Gambar 3.25 Desain Form Visualisasi Pantauan Kejadian Lalulintas

BAB IV

IMPLEMENTASI DAN EVALUASI

4.1 Kebutuhan Sistem

Untuk dapat menjalankan aplikasi pengolahan dan visualisasi data pantauan kejadian lalu lintas, dibutuhkan perangkat keras dan perangkat lunak sebagai berikut:

4.1.1 Perangkat Keras

Perangkat keras komputer adalah komponen-komponen fisik peralatan yang membentuk suatu sistem komputer, serta peralatan-peralatan lain yang mendukung komputer dalam menjalankan tugasnya. Adapun perangkat keras yang diperlukan dalam aplikasi ini adalah:

1. Processor Pentium 3 atau yang lebih baik.
2. Memory 128 Mb atau yang lebih baik.
3. Harddisk 10 Gb atau yang lebih baik.
4. Monitor.
5. Mouse, dan Keyboard.

4.1.2 Perangkat Lunak

Perangkat lunak yang diperlukan adalah program komputer yang diperlukan untuk mengoperasikan fungsi dari perangkat keras. Adapun perangkat lunak yang diperlukan dalam perancangan dan pembuatan aplikasi ini adalah:

1. Microsoft Windows XP
2. Microsoft Visual Basic 6
3. Microsoft Visio Profesional 2002

4. R2V
5. Map objects 2.2
6. MySQL 5.0
7. Power Designer 6.0

4.2 Implementasi Program

Pada sub bab berikut ini akan dijelaskan tentang penggunaan aplikasi persistem menu, mulai dari tampilan aplikasi, fungsi dan cara penggunaannya.

4.2.1 Form Login

Saat program dijalankan, form yang pertama kali muncul adalah sebagai berikut:



The image shows a screenshot of a login window titled "Laporan Lalulintas dan Pengolahan". The window has a blue and white theme. At the top left, there is a small graphic of a car on a road. At the top right, there is a large blue circular logo with a musical note and the text "Suara Surabaya FM 100". Below the logo, there are two input fields: "User ID" with the text "rYG" and "Password" with the text "xxx". Below the input fields are two buttons: "Masuk" and "Batal". The background of the window is a light blue and white pattern.

Gambar 4.1 Tampilan Form *Login*

Program ini terdiri dari 2 (dua) user yang mempunyai hak akses berbeda yaitu user Manager Online dan user Penyiar/Getekeeper. user manager online hanya bisa

menggunakan menu maintenance user, sedang user penyiar/gatekeeper bisa menggunakan semua menu, kecuali menu maintenance user.

Setelah user memasukkan user ID dan password, tekan tombol "Masuk" untuk masuk ke form Utama.

4.2.2 Form Utama

Form utama ini memiliki toolbar, yang berisikan tiga grup menu, yaitu file, maintenance, pengolahan dan visualisasi. Grup menu file, terdapat menu untuk masukan laporan, ganti password, logout, dan keluar.

Menu logout digunakan jika ingin mengganti user ID tanpa harus menutup program, tetapi langsung kembali ke form login. Sedang menu keluar digunakan untuk menutup program.

Grup menu Maintenance, terdapat menu untuk maintenance jalan, maintenance pendengar, maintenance aturan khusus, dan maintenance user. Untuk grup menu pengolahan dan visualisasi, terdapat menu untuk pengolahan pencarian rute dan visualisasi kejadian lalulintas.



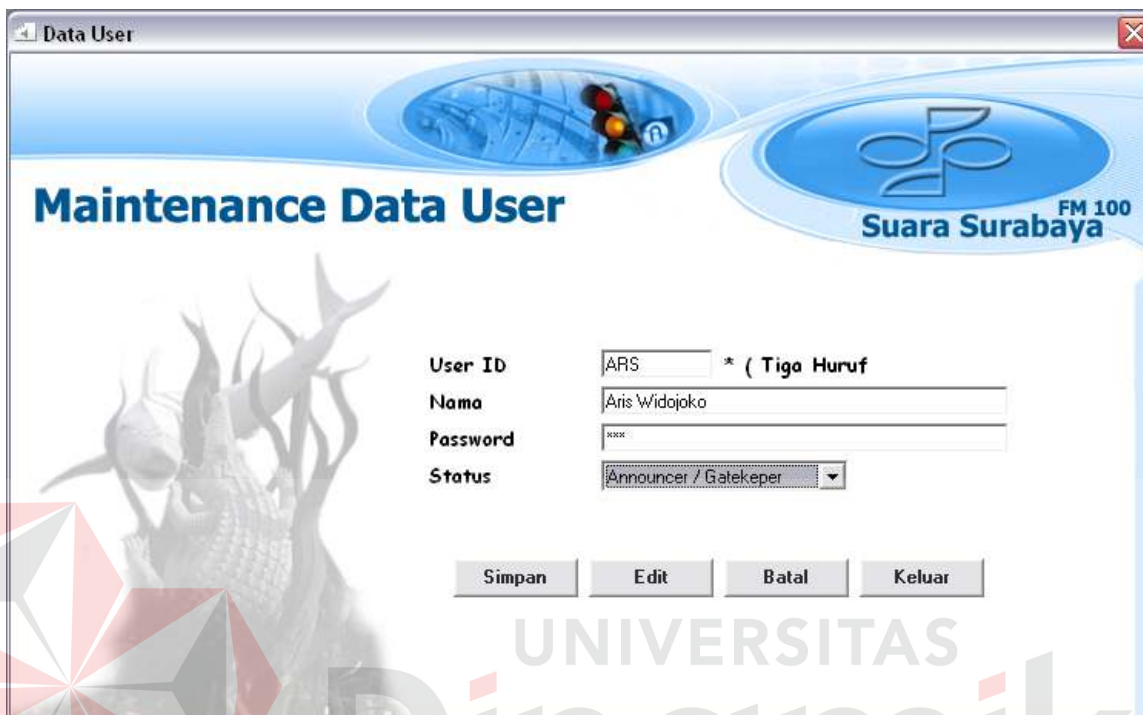
Gambar 4.2 Tampilan Form Utama

4.2.3 Form Maintenance Data User

Form ini berfungsi untuk mengolah data user, membuat user baru, dan menghapus user. Setelah kotak isian diisi semua, tekan tombol “Simpan” untuk menyimpan data, jika user id nya belum ada dalam database. jika sudah ada maka yang dilakukan adalah update data dengan user id tersebut. Pada saat user mengisikan user id, dan menekan tombol enter, maka data akan keluar otomatis jika user id sudah ada dalam database.

Untuk melihat user aktif yang ada, user bisa menekan tombol edit untuk menampilkan list view. user dapat langsung memilih user yang diinginkan dengan double click dibaris user yang diinginkan, dan secara otomatis tombol edit akan berubah menjadi tombol hapus. Setelah data tampil, user bisa mengubah datanya dengan menekan tombol simpan atau menghapus user tersebut, dengan menekan tombol hapus.

Tombol “Batal” digunakan untuk membersihkan kotak isian, dan tombol “keluar” digunakan untuk keluar dari form ini dan kembali ke form utama.



User ID	ARS	* (Tiga Huruf)
Nama	Aris Widajoko	
Password	xxxx	
Status	Announcer / Gatekeeper	

Simpan Edit Batal Keluar

Gambar 4.3 Tampilan Form Maintenance Data User

4.2.4 Form Ganti Password

Form ini digunakan untuk mengganti password untuk user itu sendiri, user tidak bisa mengganti password user yang lain. setelah mengisi password lama di kotak isian password lama dan mengisi password baru yang sama ke dalam kotak isian password baru dan password conf. Tekan tombol ganti untuk menyimpan perubahan password.

Gambar 4.4 Tampilan Form Ganti Password

4.2.5 Form Masukan Laporan

Form ini digunakan untuk menyimpan laporan laporan yang masuk, yang nanti akan diolah untuk mencari rute tercepat, atau hanya sekedar untuk divisualkan didalam peta. Jika tampilan peta terlalu kecil, user bisa memperbesar peta dengan menekan tombol “+”, dan mem-*block* daerah yang ingin diperbesar. Kalau ingin mengecilkan peta, tekan tombol “-”, lalu mengklik peta. Tombol “R” digunakan untuk mengembalikan ukuran peta seperti awal. Tombol “H” digunakan untuk menggeser peta menggunakan *mouse*. dan tombol “i” digunakan untuk mengambil informasi jalan dengan mengklik langsung jalan yang ada di peta.

Untuk memulai menggunakan form ini, user harus mengisi dulu data pelapor. Apabila nomer telepon pelapor ada dalam database, maka datanya akan otomatis keluar, kalau tidak ada, data pelapor bisa diisi manual yang nantinya akan ikut tersimpan ketika

data laporan disimpan. Tekan tombol “i” untuk mengambil data jalan dari peta, dan sesuaikan keadaan jalan dan kejadian jalan dengan laporan yang diterima.

Jika ada opini tambahan diluar laporan lalulintas, dapat ditulis dikotak isian opini. tekan tombol simpan untuk menyimpan laporan ke dalam database. tombol “edit” digunakan jika ingin mengubah data laporan yang terakhir, setelah diubah datanya lalu tekan tombol “Simpan” menyimpan perubahan.

Tombol “Batal” digunakan untuk membersihkan kotak isian, dan tombol “keluar” digunakan untuk keluar dari form ini dan kembali ke form utama.

The screenshot shows a web browser window titled "SUARA SURABAYA FM 100 - [Masukan Laporan]". The page header includes the date "30/07/2006" and time "06:37:29". The main heading is "Masukan Laporan".

The central part of the form is a map showing the Nginden Intan area with various roads labeled, including "Raya Panjang Jiwo" and "Raya Nginden Intan". A red cross marker is placed on the map near "RS. HCDS".

On the right side, there is a form for the reporter's details:

- Pelapor :**
 - Telepon/HP: 03171444228
 - Nama: Ferdian Kelana
 - Alamat: Jl. Medokan Asri Barat VII / J - 19
 - Pekerjaan: Mahasiswa

At the bottom of the form, there are several input fields and buttons:

- Kejadian Lalulintas :** Nihil
- Keadaan Lalulintas :** Padat Lancar
- Km/Jam :** 30
- Opini :** Pagi yang cerah
- Buttons: Simpan, Edit, Batal, Keluar, Bersih

The bottom section also contains a text area with the following text:

```

KEJADIAN : Nihil
KEADAAN : Padat Lancar
DI JALAN :
Raya Panjang Jiwo, J_Nginden
  
```

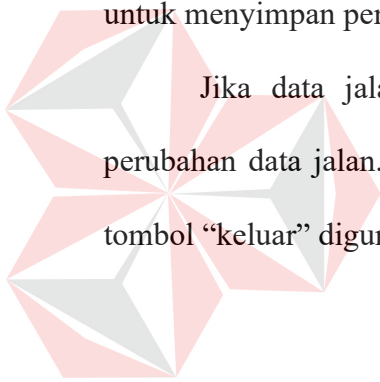
Gambar 4.5 Tampilan Form Masukan Laporan

4.2.6 Form Maintenance Data Jalan

Form ini digunakan untuk mengubah data-data ruas jalan, tekan tombol “i” untuk mengambil data gambar, klik jalan yang ingin diubah datanya. Kotak isian id jalan mengartikan grup suatu jalan, dan kotak isian Ruas mengartikan ruas jalan dari grup jalan tersebut. Seperti pada gambar 4.6, id jalan J305 mengartikan grup jalan “Raya Panjang Jiwo”, Ruas “A” mengartikan ruas jalan “A” dari grup jalan “Raya Panjang Jiwo”. user bisa berpindah ke ruas jalan lain digrup jalan itu dengan merubah pilihan ruas jalan.

Jika jalan hanya memiliki satu arah, user bisa mengubah arah arus jalan dengan menggunakan tombol ”Tukar Arah”. setelah perubahan jalan selesai, tekan tombol ”Edit” untuk menyimpan perubahan data.

Jika data jalan sudah selesai diubah, tekan tombol edit untuk menyimpan perubahan data jalan. Tombol “Batal” digunakan untuk membersihkan kotak isian, dan tombol “keluar” digunakan untuk keluar dari form ini dan kembali ke form utama.



Gambar 4.6 Tampilan Form Maintenance Data Jalan

4.2.7 Form Maintenance Data Pendengar

Form ini berfungsi untuk mengolah data pendengar, membuat data pendengar baru, dan menghapus data pendengar. Setelah kotak isian diisi semua, tekan tombol “Simpan” untuk menyimpan data, jika nomor teleponnya belum ada dalam database. Jika sudah ada maka yang dilakukan adalah update data dengan nomor telepon tersebut. Pada saat user mengisikan nomor telepon, dan menekan tombol enter, maka data akan keluar otomatis jika nomor telepon sudah ada dalam database.

Untuk melihat pendengar yang ada, user bisa menekan tombol edit untuk menampilkan *list view*. user dapat langsung memilih user yang diinginkan dengan *double click* dibaris user yang diinginkan, dan secara otomatis tombol edit akan berubah menjadi

tombol hapus. Setelah data tampil, user bisa mengubah datanya dengan menekan tombol simpan atau menghapus user tersebut, dengan menekan tombol hapus.

Tombol “Batal” digunakan untuk membersihkan kotak isian, dan tombol “keluar” digunakan untuk keluar dari form ini dan kembali ke form utama.

Telepon/HP	03171444228
Nama	Ferdian Kelana
Alamat	Jl. Medokan Asri Barat VII / J - 19
Kerja	Mahasiswa

Gambar 4.7 Tampilan Form Maintenance Data Pendengar

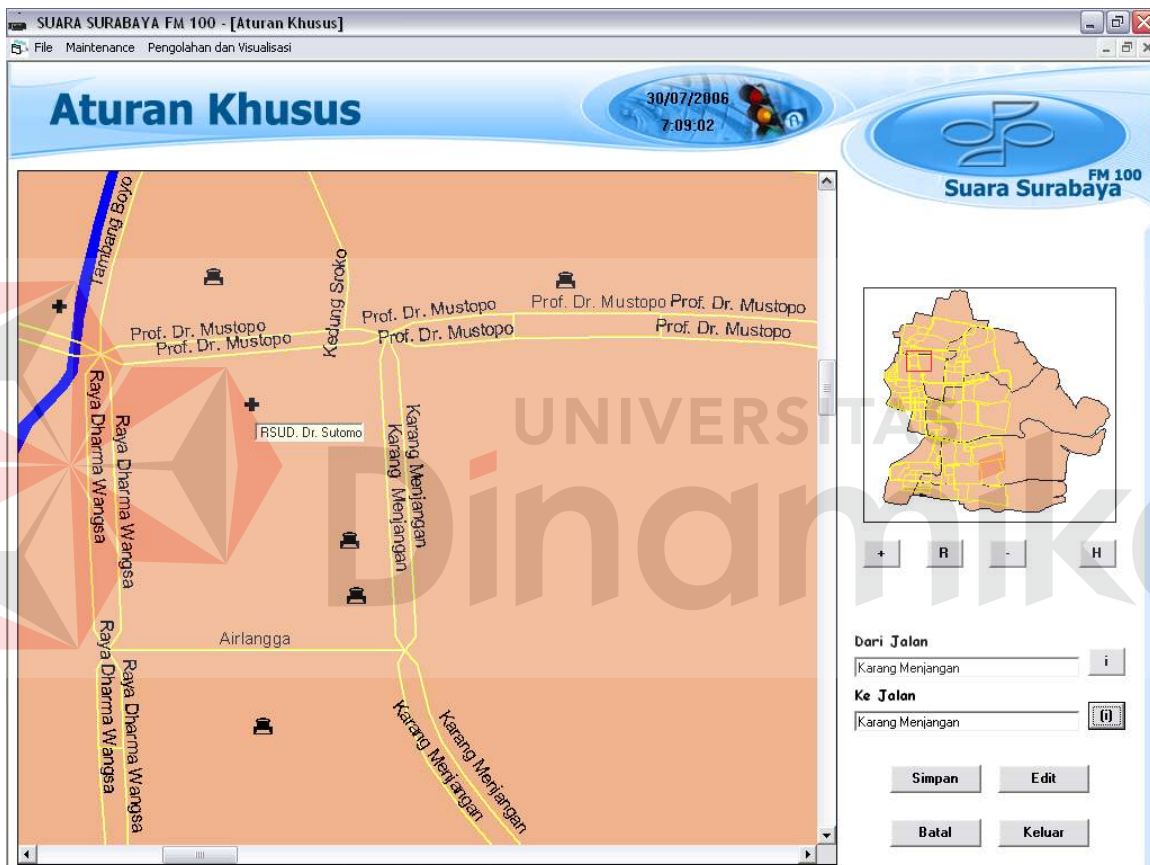
4.2.8 Form Maintenance Data Aturan Khusus

Form ini digunakan oleh user untuk memberikan aturan khusus dilarang melintas dari suatu jalan ke jalan yang lain. seperti disalah satu perempatan jalan besar, kita tidak boleh untuk belok kanan, tapi harus terus atau belok kiri.

Untuk melihat aturan khusus yang sudah ada, user bisa menekan tombol edit untuk menampilkan *list view*. user dapat langsung memilih aturan khusus yang diinginkan dengan *double click* dibaris aturan khusus yang diinginkan, dan secara

otomatis tombol edit akan berubah menjadi tombol hapus. Setelah data tampil, user bisa mengubah datanya dengan menekan tombol simpan atau menghapus user tersebut, dengan menekan tombol hapus.

Tombol “Batal” digunakan untuk membersihkan kotak isian, dan tombol “keluar” digunakan untuk keluar dari form ini dan kembali ke form utama.

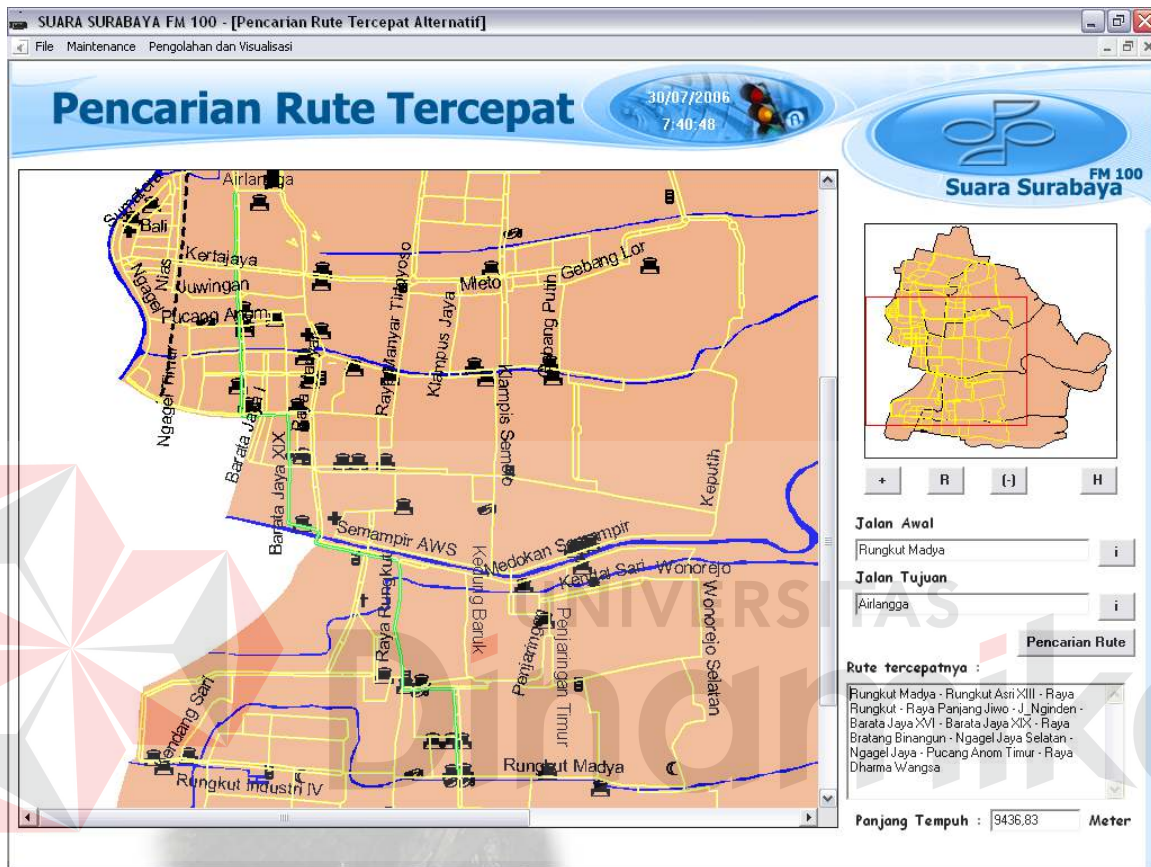


Gambar 4.8 Tampilan Form Maintenance Data Aturan Khusus

4.2.9 Form Pengolahan Pencarian Rute (Untuk Gatekeeper)

Form ini digunakan oleh user, khususnya gatekeeper untuk mencari rute tercepat atau rute alternatif. Dengan memilih jalan awal dan jalan tujuan yang sebelumnya menekan tombol “i” disebelah kanan kotak isian jalan awal dan jalan akhir untuk mengisi

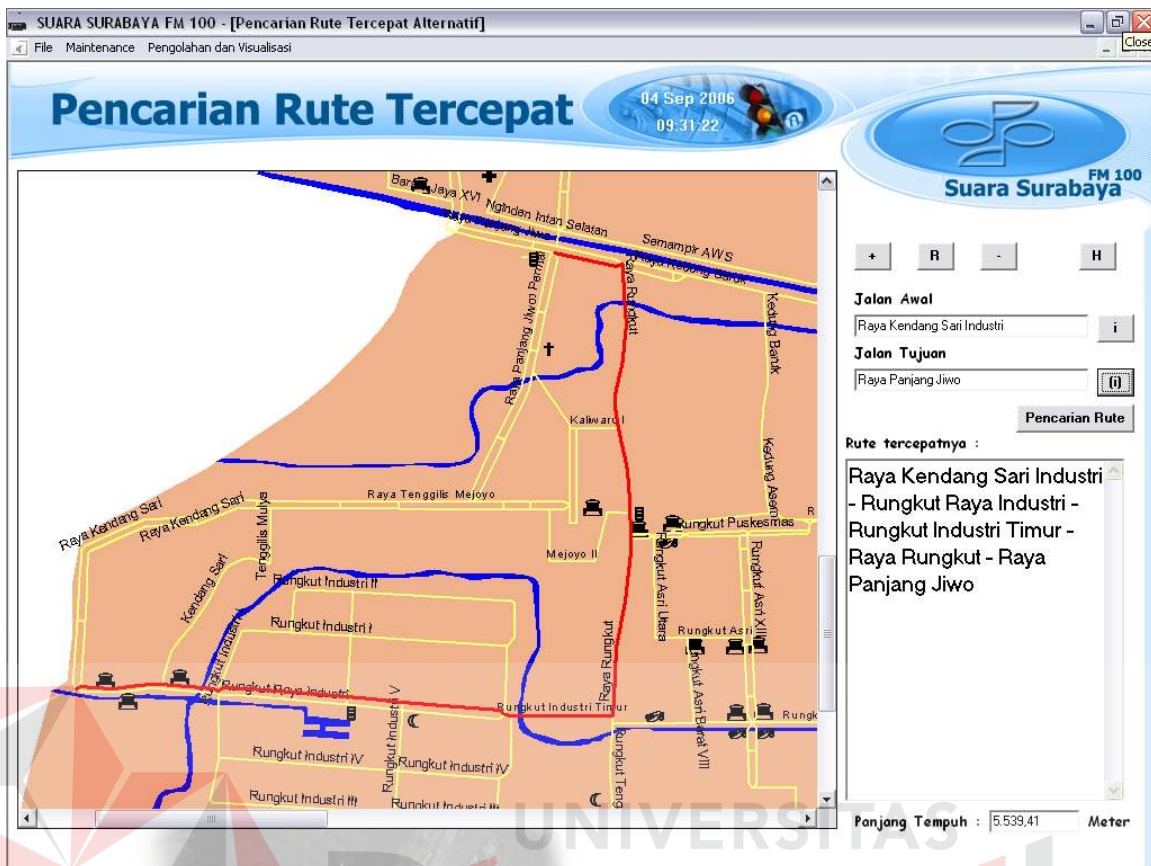
setiap kotak isiannya. setelah itu tekan tombol “Pencarian Rute” untuk menjalankan proses pencarian, *progres bar* akan muncul diawal sampai akhir proses. Hasil rute tercepat akan ditampilkan beserta panjang jalan yang ditempuh.



Gambar 4.9 Tampilan Form Pengolahan Pencarian Rute (Untuk Gatekeeper)

4.2.10 Form Pengolahan Pencarian Rute (Untuk Penyiar)

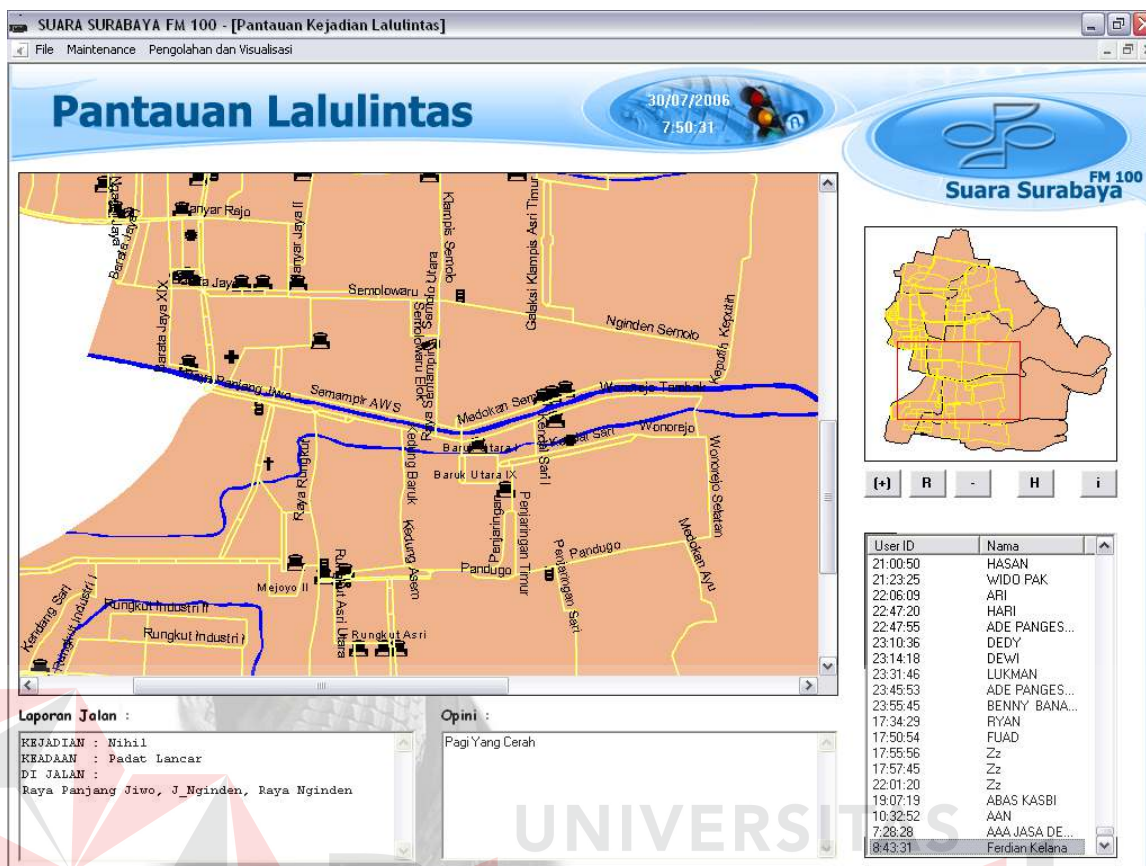
Sama seperti form pada sub bab 4.2.9, Form ini diperuntukkan khusus untuk penyiar, perbedaannya ukuran font yang lebih besar untuk membantu penyiar dalam membaca rute.



Gambar 4.10 Tampilan Form Pengolahan Pencarian Rute (Untuk Penyarir)

4.2.11 Form Visualisasi Pantauan Kejadian Lalulintas

Form ini digunakan oleh user untuk melihat keadaan jalan berdasarkan beberapa laporan yang masuk.



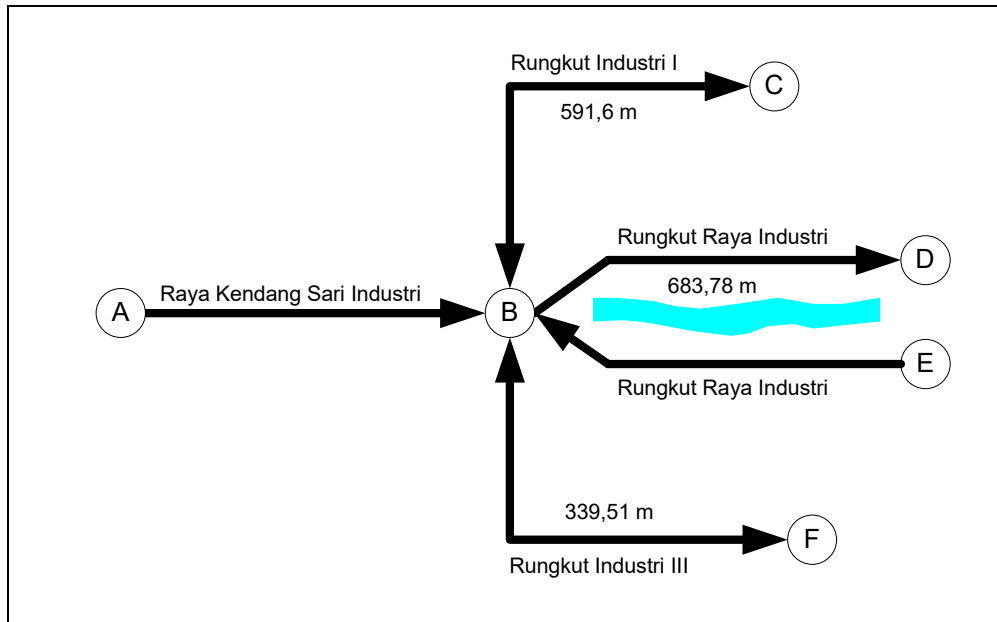
Gambar 4.11 Tampilan Form Visualisasi Pantauan Kejadian

4.3. Evaluasi

Proses evaluasi dari sistem prediksi pengangguran ini terdiri dari perhitungan manual, dan implementasi program

4.3.1 Perhitungan Manual Probabilitas disetiap Next Node

Berikut ini akan dijelaskan perhitungan manual, untuk mendapatkan probabilitas pemilihan *next node* dalam Algoritma *Ant Colony Optimization*. Seperti terlihat pada gambar 4.11, dimana terlihat sketsa jalan yang akan dicari probabilitasnya. Ketika semut jalan dari *node A* ke *node B*, di *node B* semut menemui percabangan (*next node*) jalan yaitu percabangan ke *node C* (Rungkut Industri I), *node D* (Rungkut Raya Industri), dan *node F* (Rungkut Industri III).



Gambar 4.11 Seketsa Salah Satu Persimpangan Jalan

Sebelum menghitung probabilitas *next node*, dicari dulu waktu tempuh (d_{ij}) yang dibutuhkan untuk melintasi jalan dari *node* yang ditempati semut sekarang ke tiap *next node*. Perhitungan waktu tempuh menggunakan rumus, Jarak (Km) / Kecepatan * Bobot kelas jalan. Setelah dilakukan perhitungan ditemukan hasil :

$$d_{bc} = 0,05496$$

$$d_{bd} = 0,01368$$

$$d_{bf} = 0,0204$$

Setelah waktu tempuh diketahui, sekarang mekukan perhitungan probabilitas menggunakan Rumus 2.4.

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{k \in \text{allowed}_k} [\tau_{ik}(t)]^\alpha \cdot [\eta_{ik}]^\beta} & \text{if } j \in \text{allowed}_k \\ 0 & \text{otherwise} \end{cases}$$

Dengan nilai awal feromon (τ) = 1 disetiap ruas jalan yang nantinya akan di-*update* jika ruas jalan tersebut dilewati oleh semut. Sedang nilai $\alpha = 1$ dan $\beta = 5$, diberi nilai tetap sejak awal hingga ditemukan rute yang terbaik. Untuk mempermudah perhitungan, diasumsikan lama tempuh dikali 100. Artinya, jika lama tempuh 0,0204 jam, maka diasumsikan menjadi 2,04 satuan.

$$P_{bc} = \frac{[\tau_{bc}]^\alpha [\eta_{bc}]^\beta}{[\tau_{bc}]^\alpha [\eta_{bc}]^\beta + [\tau_{bd}]^\alpha [\eta_{bd}]^\beta + [\tau_{bf}]^\alpha [\eta_{bf}]^\beta}$$

$$\begin{aligned} P_{bc} &= \frac{[1]^1 [1/5,496]^5}{[1]^1 [1/5,496]^5 + [1]^1 [1/1,368]^5 + [1]^1 [1/2,04]^5} \\ &= 0.0019942 / 0.2390209 = 0.0083432 \end{aligned}$$

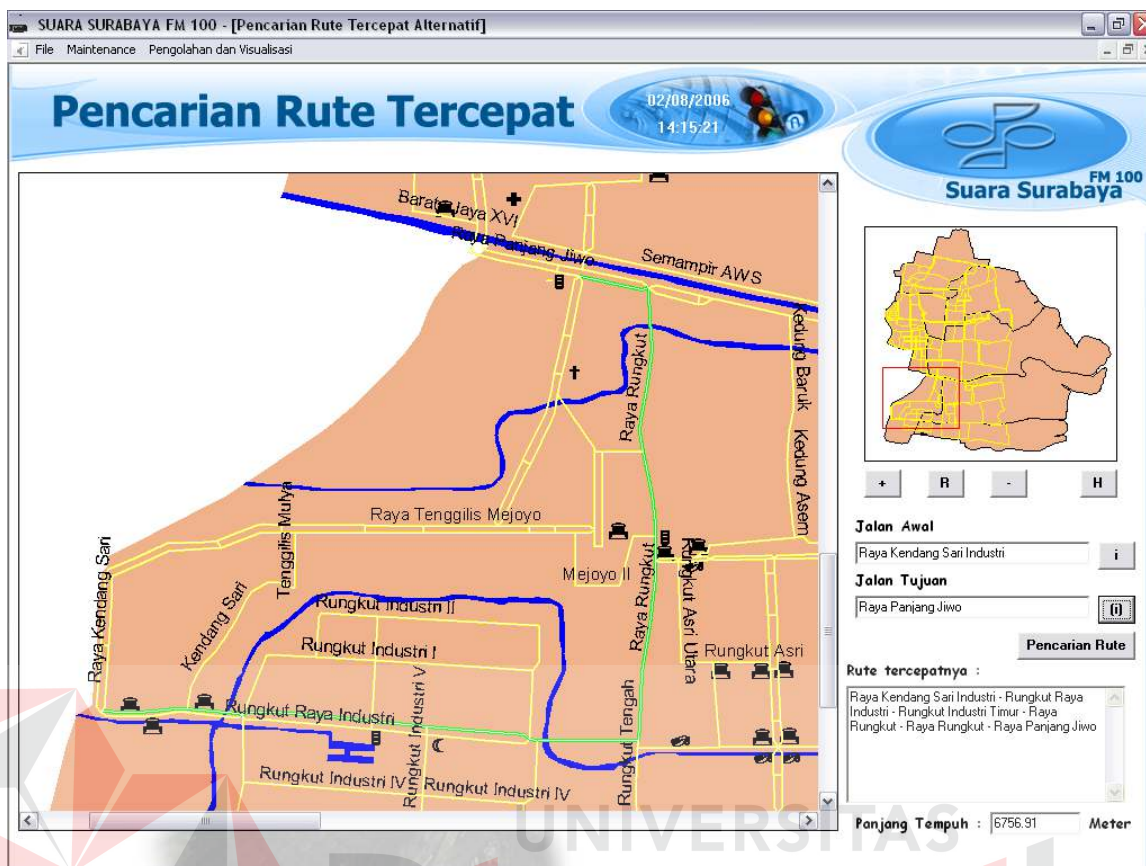
$$\begin{aligned} P_{bd} &= \frac{[1]^1 [1/1,368]^5}{[1]^1 [1/5,496]^5 + [1]^1 [1/1,368]^5 + [1]^1 [1/2,04]^5} \\ &= 0.2087226 / 0.2390209 = 0.87324 \end{aligned}$$

$$\begin{aligned} P_{bf} &= \frac{[1]^1 [1/2,04]^5}{[1]^1 [1/5,496]^5 + [1]^1 [1/1,368]^5 + [1]^1 [1/2,04]^5} \\ &= 0.0283041 / 0.2390209 = 0.1184168 \end{aligned}$$

Dari perhitungan nilai probabilitas diatas, diketahui probabilitas ke *node* d lebih besar dari *node* yang lain, maka peluang semut untuk memilih Rungkut Raya Industri adalah besar. Perhitungan probabilitas ini dilakukan hanya jika *Next Node* lebih dari 1, jika *Next Node* hanya satu, maka *node* itu akan langsung dipilih.

4.3.2 Uji Coba Program

Untuk menguji validasi sistem yang dibuat, berikut ini akan digambarkan hasil eksekusi program :



Gambar 4.12 Hasil Pencarian dengan Algoritma *Ant Colony Optimization*

Pada gambar diatas dapat dijelaskan, setelah tombol cari ditekan, sistem melakukan proses pencarian rute tercepat. Pada proses pencarian rute terpendek tersebut didapat hasil yang akan dijabarkan sebagai berikut :

a. Input nama jalan

Jalan Asal : Raya Kendang Sari Industri

Jalan Tujuan : Raya Panjang Jiwo

b. Hasil Pencarian Rute tercepat

Rute jalan : Raya Kendang Sari Industri - Rungkut Raya Industri - Rungkut Industri Timur - Raya Rungkut - Raya Rungkut - Raya Panjang Jiwo

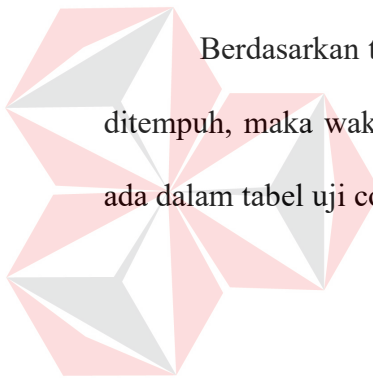
Jarak yang ditempuh : 6.756,91 meter

4.3.3 Kinerja Sistem

Pada sub bab ini akan dilakukan uji coba pencarian rute terpendek untuk mengetahui waktu proses yang dibutuhkan, dimana keadaan semua jalan adalah lancar.

Uji coba untuk mengetahui lama waktu memproses dapat dilihat dalam tabel 4.1.

Berdasarkan tabel 4.1. dapat disimpulkan bahwa semakin panjang rute jalan yang ditempuh, maka waktu yang dibutuhkan semakin lama. Jika diambil waktu proses yang ada dalam tabel uji coba, rata-rata waktu proses adalah satu menit empat puluh detik.



UNIVERSITAS
Dinamika

Table 4.1 Uji Coba Waktu Proses Pencarian Rute Tercepat

No	Jalan Asal	Jalan Tujuan	Rute Perjalanan	Panjang Rute(m)	Lama Proses
1.	Raya Panjang Jiwo	Pucang Anom	Raya Panjang Jiwo - J_Nginden - Raya Nginden - Raya Manyar - Ngagel Jaya Utara - Ngagel Jaya - Pucang Anom Timur - Pucang Anom	4.248,62	29 detik
2.	Rungkut Madya	Raya Dharma Wangsa	Rungkut Madya - Rungkut Asri XIII - Rungkut Puskesmas - Raya Rungkut - Raya Panjang Jiwo - J_Nginden - Raya Nginden - Raya Manyar - Ngagel Jaya Utara - Ngagel Jaya - Pucang Anom Timur - Raya Dharma Wangsa - Raya Dharma Wangsa	9443,22	2 menit 10 detik
3.	Raya Kendang Sari Industri	Raya Panjang Jiwo	Raya Kendang Sari Industri - Rungkut Raya Industri - Rungkut Industri Timur - Raya Rungkut - Raya Panjang Jiwo	5.539,41	35 detik
4.	Raya Mulyorejo	Sulawesi	Raya Mulyorejo - Lingkar Raya Mulyorejo - Raya ITS - Bundaran ITS - Raya Kertajaya Indah - Raya Kertajaya Indah - Manyar Kertoarjo - Kertajaya - Sulawesi	7.584,47	1 menit 13 detik
5.	Wiguna Timur	Kenjeran	Wiguna Timur - Medokan Sawah - Medokan Ayu - Rungkut Asri Timur XV - Penjaringan Sari - Pandugo - Penjaringan Timur - Kendal Sari - Baruk Utara XVI - Baruk Utara IX - Raya Baruk Utara - J_MERR II C - Medokan Semampir - Raya Semampir - Semolo Waru Tengah I - Semolo Utara - Klampis Semolo - Klampis Harapan - Arif Rahman Hakim - Manyar Kertoadi - Raya Kertajaya Indah - Raya ITS - Lingkar Raya Mulyorejo - Raya Mulyorejo - Tempurejo - Kenjeran	14.589,59	3 menit 57 detik

BAB V

PENUTUP

5.1. Kesimpulan

Dari hasil studi pustaka, analisa, desain, pengembangan, implementasi serta proses pencarian rute tercepat menggunakan algoritma *Ant Colony Optimization* dapat disimpulkan:

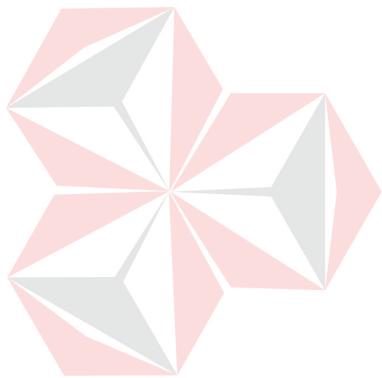
1. Aplikasi ini dapat memberikan info rute tercepat yang ditampilkan dalam bentuk peta beserta informasi jalan-jalan yang dilalui dengan jarak tempuh yang dilalui, sehingga rute yang ditemukan dapat mudah disampaikan oleh penyiar radio SS kepada pendengar radio.
2. Lamanya proses tergantung pada jauh dekatnya rute yang ditempuh. Semakin jauh jalan berangkat dengan jalan tujuan, maka semakin lama proses yang dibutuhkan untuk mendapatkan solusi, begitu juga sebaliknya.

5.2 Saran

Untuk lebih meningkatkan kemampuan dari aplikasi pengolahan dan visualisasi data pantauan kejadian lalulintas yang dibuat oleh penulis, penulis memberikan saran sebagai berikut :

1. Aplikasi ini dapat ditambahkan aplikasi peramalan untuk meramalkan daerah-daerah mana yang akan terjadi kemacetan, dari laporan-laporan yang masuk beberapa bulan atau beberapa tahun terakhir. Sehingga pencarian rute akan lebih *real* lagi.
2. Akan lebih berguna lagi jika aplikasi ini dibuat versi mobile, agar para pengendara bisa melihat kejadian-kejadian yang terjadi di jalan lewat internet,

dan langsung dapat melihat rute mana yang dapat diambil, mulai dari rumah atau kantor sampai tempat tujuan.



UNIVERSITAS
Dinamika

DAFTAR PUSTAKA

Doerner K, Hartl F.R, Reimann M, 2000, *Cooperative Ant Colonies for Optimizing Resource Allocation in Transportation*, 16 November 2000, Departement of Management Science, University of Vienna, Austria.

Doerner K, Hartl F.R, Reimann M, 2000, *Ant Colony Optimization applied to The Pickup and Delivery Problem*, 9 November 2000, Departement of Management Science, University of Vienna, Austria.

Dorigo M, Maniezzo V, Colomi A, 1996, *The Ant System: Optimization by a colony of cooperating agents*, IRIDIA, Université Libre de Bruxelles, Avenue Franklin Roosevelt.

Peraturan Pemerintah Republik Indonesia Nomor 43 Tahun 1993
Tentang Prasarana Dan Lalu Lintas Jalan. 1993.
Jakarta.

PT. Karya Pembina Swajaya, 2004, Surabaya & Perkembangannya (Edisi Terbaru),
Surabaya.



UNIVERSITAS
Dinamika