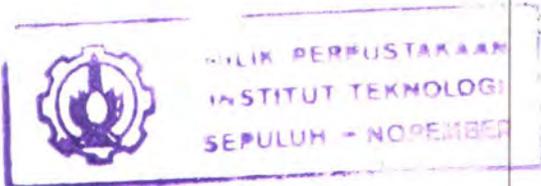


21-655 1114/105



PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK PELACAKAN PENGIRIMAN KOTAK UANG DENGAN MENGGUNAKAN GIS

TUGAS AKHIR



RSIF
005.1
jun
P-1
2004

PERPUSTAKAAN ITS	
Tgl. Terima	20-2-2004
Terima Dari	H/
No. Agenda Prp.	219552

Disusun Oleh :

ACHMAD JUNAIDI

NRP. 5197 100 102

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2004**

PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK PELACAKAN PENGIRIMAN KOTAK UANG DENGAN MENGGUNAKAN GIS

TUGAS AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer

Pada

Jurusan Teknik Informatika

Fakultas Teknologi Informasi

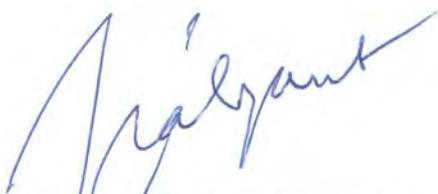
Institut Teknologi Sepuluh Nopember

Surabaya

Mengetahui / Menyetujui,

Dosen Pembimbing I

Dosen Pembimbing II



Ir. Aris Tjahyanto M.Kom

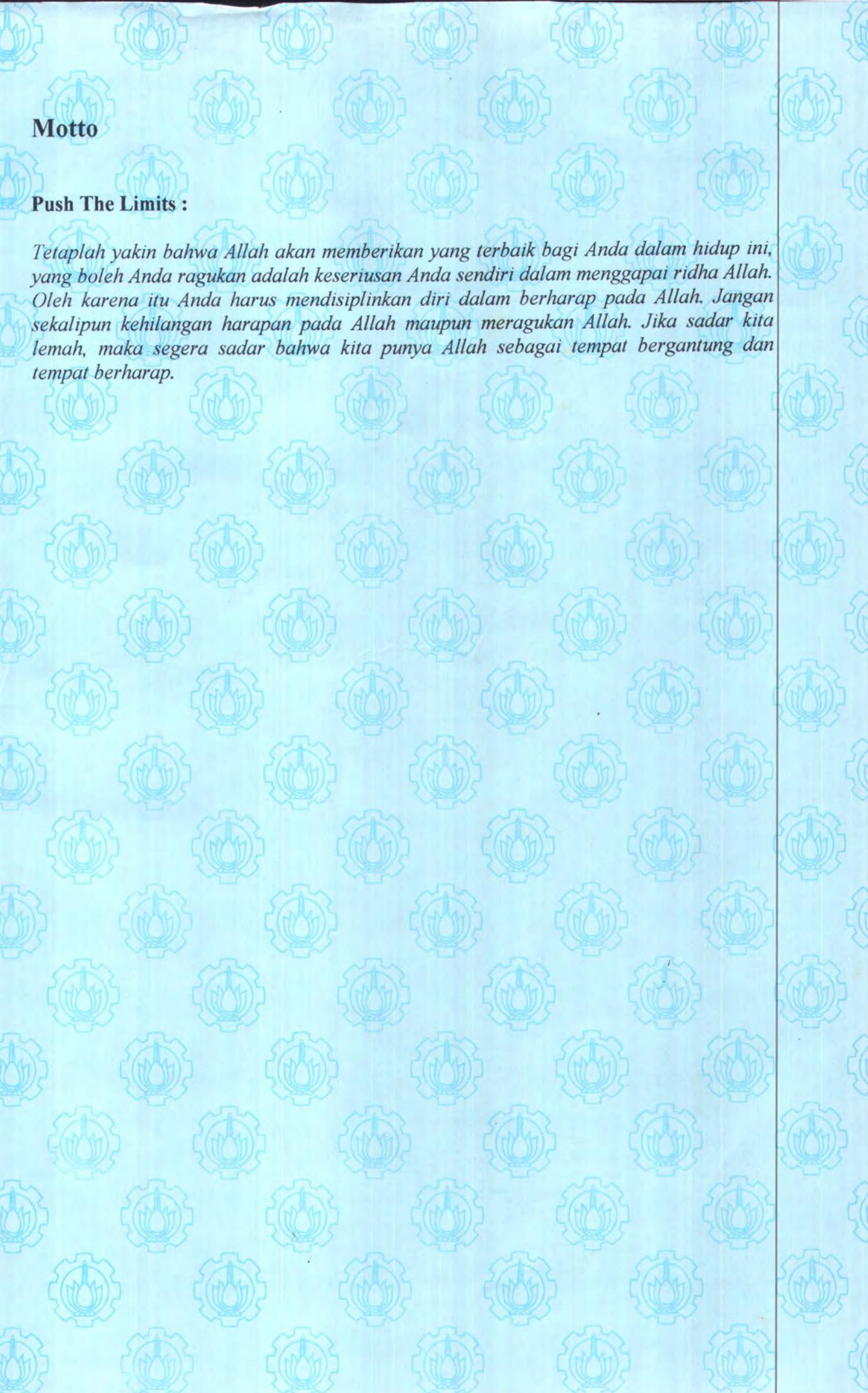
NIP. 131 933 299



Dr. Ir. Muhammad Taufik

NIP. 131 633 395

**SURABAYA
JANUARI, 2004**



Motto

Push The Limits :

Tetaplah yakin bahwa Allah akan memberikan yang terbaik bagi Anda dalam hidup ini, yang boleh Anda ragukan adalah keseriusan Anda sendiri dalam menggapai ridha Allah. Oleh karena itu Anda harus mendisiplinkan diri dalam berharap pada Allah. Jangan sekalipun kehilangan harapan pada Allah maupun meragukan Allah. Jika sadar kita lemah, maka segera sadar bahwa kita punya Allah sebagai tempat bergantung dan tempat berharap.



ABSTRAK

ABSTRAK

Uang adalah alat perantara pertukaran barang dan jasa. Roda perekonomian tidak dapat berjalan tanpa adanya uang. Bank Indonesia sebagai bank sentral, ditunjuk oleh pemerintah untuk mencetak dan mengedarkan uang serta bertugas mengurus dan mengatur peredaran uang di dalam negeri. Saat ini telah digunakan berbagai macam teknologi yang canggih untuk menangani transaksi keuangan di perbankan seperti ATM dan internet banking. Akan tetapi teknologi untuk menangani masalah fisik uang masih sangat minim. Salah satunya dalam hal pengiriman uang.

Selama ini proses administrasi pengiriman yang dilakukan belum terkomputerisasi. Selama perjalanan, kondisi dan posisi pengiriman tidak dapat dimonitor secara realtime. Dibutuhkan sebuah perangkat lunak untuk menyimpan dan mengolah data transaksi pengiriman uang secara elektronik dan mampu melacak kondisi dan posisi uang selama dalam perjalanan.

Dalam tugas akhir ini dibuat aplikasi berbasis web yang menyediakan sarana pencatatan dan pengolahan data proses pengiriman uang mulai dari pencatatan tanggal pelaksanaan, petugas yang bertanggung jawab, dan laporan beserta rekapitulasi pengiriman yang telah dilakukan. Dan yang paling pokok adalah pembuatan sistem pelacakan yang akan memantau posisi, kecepatan, dan status dari pengiriman yang dilakukan. Sistem pelacakan ini memanfaatkan Sistem Informasi Geografis yang ditunjang dengan penggunaan alat AVL.

Dengan sistem ini, data administrasi proses pengiriman uang dapat disimpan dalam satu skema basis data sebagai arsip. Dan dengan pelacakan secara realtime dapat membantu pengambilan tindakan bilamana terjadi gangguan selama proses pengiriman.

Kata kunci: Pengiriman Uang, Pelacakan, Kondisi, Posisi, Kecepatan, AVL, Web, Bank



KATA PENGANTAR

KATA PENGANTAR

Alhamdulillah, segala puji bagi Allah SWT semata atas limpahan rahmat dan hidayah-Nya sehingga memungkinkan penulis untuk menyelesaikan Tugas Akhir yang berjudul **“Perancangan dan Pembuatan Perangkat Lunak Pelacakan Pengiriman Kotak Uang Dengan Menggunakan GIS”**.

Tugas Akhir yang memiliki beban sebesar 4 satuan kredit merupakan tahapan yang harus ditempuh oleh setiap mahasiswa untuk menyelesaikan studi program strata satu (S-1) dan memperoleh gelar Sarjana Komputer pada jurusan Teknik Informatika di Institut Teknologi Sepuluh Nopember Surabaya.

Penulis menyadari bahwa sebagai manusia biasa penulis mempunyai banyak kekurangan dan keterbatasan, namun dengan keterbatasan yang ada penulis tetap berharap semoga Tugas Akhir ini dapat bermanfaat bagi semua pihak.

Tidak ada gading yang tak retak, begitu pula dengan laporan Tugas Akhir ini. Karena itu, dengan kerendahan hati Penulis masih mengharapkan masukan, saran, kritik dari pembaca yang berguna bagi pengembangan aplikasi Tugas Akhir ini.

Surabaya, Januari 2004

Achmad Junaidi

UCAPAN TERIMA KASIH

Dengan tak bosannya mengucapkan syukur Alhamdulillah kepada **Allah SWT**, yang telah memberi terlalu banyak dari yang layak penulis terima, telah penulis curi sekelumit misteri-Mu, dan penulis kembalikan dalam bentuk buku ini. Di kesempatan ini, Penulis hendak menyampaikan rasa penghormatan yang setingginya serta rasa terima kasih kepada pihak-pihak yang telah memberi bantuan baik itu berupa moril maupun material dan langsung maupun tidak langsung kepada:

1. **Ibu (Siti Amanah)** dan **Bapak (Baedowi)** tercinta atas segala dukungan semangat, bimbingan dan kasih sayang.
2. **Bapak dan Ibu Soedjarwo** yang telah menjadi orang tua asuh penulis.
3. **Bapak Ir. Aris Tjahyanto, M.Kom** dan **Dr. Ir. Muhammad Taufik** sebagai dosen pembimbing Tugas Akhir penulis atas segala bimbingan dan arahnya.
4. **Bapak Yudhi Purwananto, M.Kom** selaku Ketua Jurusan Teknik Informatika ITS dan **Bapak Ir. Arif Djunaidy, M.Sc, Ph.D** selaku Dekan Fakultas Teknologi Informasi ITS.
5. **Bapak Dwi Sunaryono, S.Kom** selaku dosen wali penulis.
6. Adik-adikku **Aris, Diana, Saipul, dan Feni** atas dorongan semangat dan kasih sayang yang telah diberikan kepada penulis.
7. **Bapak Faizal Johan Atletiko S.Kom, Bapak Imam Kuswardayan S.Kom, Bapak Wahyu Suadi, M.Kom, Bapak Ir. Khakim Ghozali, Bapak Royyana Muslim I, S.Kom, Ibu Chastine Fatichah S.Kom, Ibu Diana Purwitasari, S.Kom** dan dosen-dosen Jurusan Teknik Informatika ITS

yang telah memberikan ilmu yang sangat besar manfaatnya selama proses belajar mengajar.

8. **Mas Hoedan, Mas Yudi, Pak Narno, Mbak Davi, Pak Mu'in, Pak Mono, Mbak Irna** dan seluruh **staf dan karyawan Jurusan Teknik Informatika ITS** yang telah banyak memberikan kemudahan administrasi dan fasilitas.
9. **Mbak Indah, Mbak Etmi, Mbak Lina, Nur, Nila, Nita, Jun, Donnie, Mas Edi, Mas Yudi, Mas Heri** dan seluruh karyawan PIKTI dengan canda tawanya yang membuat penulis senantiasa fresh.
10. **Bapak Subiantoro** sekeluarga (**Ibu, Mas Adit, Mbak Indah, Lia, Nila**), atas dukungan semangatnya dan nasehat beserta petuah-petuahnyanya kepada penulis.
11. **Bapak Fajar Effendy** selaku nara sumber yang banyak membantu penulis dalam pencarian data.
12. **Kendy, Bhakti, Ronnie, JP** yang selalu mendukung dan tempat bertanya penulis dalam mengerjakan perangkat lunak tugas akhir ini.
13. **Ruli00, Joko00, Indie, Ciwid** sebagai editor yang telah membantu penulis dalam penyusunan buku.
14. Teman bersama TA : **Nissa, Kamal, Dwi, Medi, Kendy, Premi, Luluk, Puput, Yotha, Widhi, Rifqi, Rizki, Ade, Dhanic, Nunung, Kholid, Odi, Ilma, Mbak Lela**
15. Teman-teman angkatan 97 C0D: **Nopin, Nova, Hera, Tita, Adhita, Winnie, Dhani, Pipin, Tante, Guswid, Vita, Zunaita, Yazid, Dewa, Yudi, Qoyim, Damen, Wiro, Guruh, Kooper, KI, Dinglek** dan lainnya yang tidak bisa disebutkan satu persatu.

16. Teman-teman angkatan 98 COE: **Heri W, Didit, Kendy, Cphee, Wong, Leo, Deku, Ereal “Krokodilabist” Cool, Budianto, Deni, Dhane, Reza, Yudha, Mbah Nur, Bhakti, Joko, Rizki Paduka Sang Jenggot Naga, Arif Kakek** dan lainnya yang tidak bisa disebutkan satu persatu.
17. Teman-teman angkatan 99 COF: **MS, Kamal, Ilma, Medi, Nina, Indie, Ervan, Dhion, Ruli, Erzu, Rontog, Anik, Liga, Rifqi, Ical, Firman** dan lainnya yang tidak bisa disebutkan satu persatu.
18. Teman-teman angkatan 2000 C10: **JP, Joko, Jakka, Gunna, Siro, Udin, Unang, DR Senbei, Ruli, Sofi, Dyah, Suci, Wida, Putri, Titin, Ika, Erma, Aini, Reni, Deddy, Maria, Lidya, Irfan, Anis, Diyan, Veni, Kholimi, Pak Dhe, Zimux”Amazing Crocodilust”** dan lainnya yang tidak bisa disebutkan satu persatu.
19. Teman-teman angkatan 2001 C11: **Widya, Phe, Dhewy, Dewi, Dwi, Nina, Siti, Ossy, Hatfi, Sempal, Lufi, Yeni, Fanni, Rudi, Eko, Sigit, Mardi** dan lainnya yang tidak bisa disebutkan satu persatu.
20. Teman-teman angkatan 2002 C12: **Rina, Uwid, Knip, Antie, Huda, Chiman, Victor, Zhain a.k.a phalanx, Hendra** dan lainnya yang tidak bisa disebutkan satu persatu.
21. Kru Laboratorium Pemrograman, **Mas Gayuh, Indie99, Indra99, Rozi00, Joko00, Gunna00, Jaka00, Lintang00, lin00** yang banyak membantu dalam fasilitas.
22. AOE mania: **Yo, Anty_Komitmen, Bejujag, Ghost Prince, ATL, ASDF, Erye, Yudha, Sleeper_N, Al-Capone.**

23. Terakhir, rekan-rekan saya. Banyak di antaranya sudah tersebut, tapi lebih banyak lagi yang belum. Mereka, yang tahu tanpa perlu disebut namanya satu persatu.

Tiada untaian kata yang cukup yang dapat penulis sampaikan sebagai balas atas jasa yang penulis terima melainkan hanya harapan semoga ALLAH SWT membalas semua amal tersebut. Jazakumullah Khairan Katsiran.



DAFTAR ISI

DAFTAR ISI

DAFTAR ISI	i
DAFTAR GAMBAR	iv
DAFTAR TABEL	vi
BAB I PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Tujuan dan Manfaat.....	2
1.3. Perumusan Masalah.....	3
1.4. Pembatasan Masalah	3
1.5. Metodologi Pelaksanaan Tugas Akhir	4
1.6. Sistematika Pembahasan	6
BAB II TEORI PENUNJANG	8
2.1. Uang.....	8
2.1.1. Jenis Uang	8
2.1.2. Fungsi Uang	9
2.2. Bank.....	11
2.2.1. Bank Sentral	11
2.2.2. Bank Pelaksana	13
2.3. Arus Perputaran Uang	13
2.4. Pengiriman uang	14
2.5. Sistem Informasi Geografis.....	16
2.5.1. Pengertian Sistem Informasi Geografis	17
2.5.2. Unsur Utama Sistem Informasi Geografis	18
2.5.2.1. Sumber data.....	18
2.5.2.2. Jenis Data	19
2.5.3. Komponen Informasi Kenampakan Geografis.....	19
2.5.4. Kemampuan Pengolahan Data Grafik dan Non Grafik.....	20
2.6. Peta	21
2.6.1. Elemen Grafik Penyusun Bentuk Dasar Peta.....	22
2.6.2. Sistem Koordinat.....	24
2.7. ASPMAP.....	25
2.7.1. Format Peta	28
2.7.2. Map Object.....	30
2.7.2.1. Layer Object.....	30
2.7.2.2. DynamicLayer Object.....	31
2.7.2.3. DynamicPoint Object.....	31
2.7.2.4. CallOuts Object.....	31
2.7.2.5. Recordset Object.....	31
2.7.2.6. FeatureRenderer Object	31
2.7.2.7. Feature Object	32
2.7.2.8. Symbol Object.....	32
2.7.2.9. FontObject.....	32
2.8. Sistem Pelacakan Secara Umum.....	32
2.8.1. Teknologi Komunikasi.....	32
2.8.2. Global Positioning Primer.....	33

2.8.3.	Komponen-komponen Sistem Pelacakan.....	37
2.8.3.1.	Remote Sensor dan Receiver	37
2.8.3.2.	Alternatif Media Komunikasi	38
2.8.3.3.	Sistem Satelit.....	42
2.8.4.	Automatic Vehicle Location	42
2.8.4.1.	Sistem AVL secara <i>Off-Line</i>	44
2.8.4.2.	Sistem AVL secara <i>On-Line</i>	45
2.8.4.3.	Konversi Koordinat.....	46
BAB III SISTEM PELACAKAN PADA PENGIRIMAN KOTAK UANG..		51
3.1.	Konsep Umum Pelacakan Posisi Pengiriman Kotak Uang.....	51
3.2.	Konsep Umum Pelacakan Kotak Uang Berbasis GIS menggunakan AVL(Automatic Vehicle Location).....	52
3.3.	Perbandingan Metode Pelacakan Posisi	54
3.4.	Konsep Teknis Pelacakan Kotak Uang Berbasis GIS menggunakan AVL (Automatic Vehicle Location).....	57
3.5.	Analisa Kebutuhan Sistem Pelacakan.....	59
3.6.	Skenario Pada Data Kecil.....	67
BAB IV PERANCANGAN PERANGKAT LUNAK.....		74
4.1.	Desain Fitur	74
4.1.1.	Perekaman Data Perjalanan.....	75
4.1.2.	Pemantauan Posisi Terkini	75
4.1.3.	Bersifat Online-Real Time	76
4.1.4.	Analisa Data Perjalanan	77
4.1.5.	Tingkat Keamanan Yang Tinggi.....	79
4.1.6.	Memiliki Sistem Peringatan Terhadap Bahaya.....	80
4.1.7.	Pelaporan Yang Lengkap Dan Terstruktur	81
4.1.8.	Kontrol Sistem Yang Ketat.....	81
4.2.	Perancangan Proses	82
4.2.1.	DAD Level 0 : Proses Pelacakan Mobil Pembawa Kotak Uang.....	84
4.2.2.	DAD Level 1	85
4.2.3.	DAD Level 2 : Proses Tracking.....	89
4.3.	Arsitektur Sistem	90
4.4.	Perancangan Data	92
4.4.1.	Data Masukan.....	92
4.4.2.	Data Proses.....	94
4.4.3.	Data Keluaran.....	95
4.4.4.	Desain Basis Data	96
4.5.	Perancangan Antarmuka	105
BAB V IMPLEMENTASI PERANGKAT LUNAK.....		111
5.1.	Pembuatan Matriks CRUD.....	115
5.2.	Pembuatan dan Pengolahan Peta	118
5.3.	Pembuatan Koneksi Basis Data	120
5.4.	Pembuatan Fasilitas Login	120
5.5.	Pembuatan Fasilitas Administrasi Data Master	123
5.6.	Pembuatan Fasilitas Penambahan Data Transaksi Perjalanan	128
5.7.	Pembuatan Fasilitas Melihat Peta	136

5.8.	Pembuatan Aplikasi Server.....	142
5.9.	Menampilkan posisi terkini dan rekaman perjalanan pada peta ...	147
BAB VI	UJI COBA PERANGKAT LUNAK DAN ANALISA HASIL.....	154
6.1.	Lingkungan Uji Coba	154
6.2.	Skenario Uji Coba.....	154
6.2.1.	Uji Coba Skenario 1	155
6.2.2.	Uji Coba Skenario 2 : Status AVL yang Berubah.....	158
6.2.3.	Uji Coba Skenario 3 : Kontrol Kecepatan.....	162
6.3.	Analisa Hasil Ujicoba.....	166
BAB VII	PENUTUP.....	169
7.1.	Kesimpulan	169
7.2.	Saran.....	169
DAFTAR PUSTAKA.....		171



MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER



DAFTAR GAMBAR

DAFTAR GAMBAR

Gambar 2.1 ..Arus perputaran uang.....	13
Gambar 2.2 Tiga Aspek Sistem Informasi Geografis.....	18
Gambar 2.3 Teknik proyeksi peta dalam kartografi5.....	20
Gambar 2.4Enam elemen grafik penyusun peta.....	23
Gambar 2.10 Object Diagram AspMap 2.0.....	27
Gambar 2.11Komunikasi secara global.....	33
Gambar 2.12 Posisi satelit penyedia layanan GPS.....	34
Gambar 2.13 Penentuan koordinat oleh satelit.....	36
Gambar 2.14 Penentuan koordinat oleh satelit dengan cara lain.....	36
Gambar 2.15 Manajemen armada menggunakan AVL.....	43
Gambar 2.16. Cara Kerja Sistem AVL secara <i>Off-Line</i>	45
Gambar 2.17 Cara Kerja Sistem AVL <i>On-Line</i>	46
Gambar 3.1 AVL dan antena.....	58
Gambar 3.2Sistem kerja dari AVL.....	59
Gambar 4.1 Cara Kerja Sistem AVL On-Line	76
Gambar 4.2. Contoh peringatan yang diberikan	78
Gambar 4.3. Contoh peringatan darurat.....	79
Gambar 4.4. Fasilitas Login	80
Gambar 4.5. Sistem Peringatan	81
Gambar 4.6. Proses Pelacakan.....	83
Gambar 4.7 Diagram DAD Level 0.....	84
Gambar 4.6 Diagram DAD Level 1.....	86
Gambar 4.7Flowcart Proses Analisa.....	88
Gambar 4.7Diagram DAD Level 2 : Proses Tracking.....	89
Gambar 4.11 Arsitektur Sistem Secara Umum.....	91
Gambar 4.12 Contoh tampilan peta masukan	93
Gambar 4.13 Diagram Konsep Data.....	98
Gambar 4.14Diagram Fisik Data.....	100
Gambar 4.12MenuUtama Aplikasi.....	107
Gambar 4.15. Contoh Halaman Muka.....	107
Gambar 4.16. Menu Administrator.....	108
Gambar 4.17. Alur Pengaksesan Halaman Operator.....	108
Gambar 4.18. Alur Pengaksesan Halaman Manajer.....	109
Gambar 4.19.Alur Pengaksesan Halaman Asisten Manajer.....	110
Gambar 5.1. Integrasi bagian-bagian implementasi.....	113
Gambar 5.2. Cara Kerja Aplikasi Server	115
Gambar 5.3 Peta ITS	118
Gambar 5.4 Tabel data spasial	119
Gambar 5.6. Pesan Gagal Login	121
Gambar 5.5. Letak Form Login.....	122
Gambar 5.7. Administrasi Data Master Koper.....	123
Gambar 5.8 Form Tambah Data dan Edit Data	126

Gambar 5.9. Tampilan halaman Penambahan Data Perjalanan	128
Gambar 5.10. Isian Tempat Asal dan Tempat Tujuan.....	129
Gambar 5.11. Memilih Mobil.....	130
Gambar 5.12. Hasil Memilih Kendaraan.....	130
Gambar 5.13 Memasukkan Data Koper.....	132
Gambar 5.14 Menambahkan Data Koper dan Uang.....	133
Gambar 5.15. Fasilitas Melihat Peta	136
Gambar 5.16. Hasil Identifikasi.....	137
Gambar 5.17. Antarmuka Aplikasi Server.....	142
Gambar 5.18. Tampilan Peta Posisi Terkini Mobil yang sedang Melakukan Perjalanan.....	151
Gambar 5.19. Tampilan Peta Rekaman Perjalanan Mobil.....	153
Gambar 6.1. Posisi Kendaraan Waktu Start.....	156
Gambar 6.2. Posisi Kendaraan Waktu di Jl. Taman Alumni.....	157
Gambar 6.3. Posisi kendaraan pada saat telah tiba di Bank BNI.....	157
Gambar 6.4. Posisi Kendaraan Waktu Start.....	159
Gambar 6.5 Kondisi Perjalanan pada titik ke-14	160
Gambar 6.6. Peringatan Status AVL yang berubah.....	161
Gambar 6.7. Perjalanan pada titik ke-9.....	164
Gambar 6.8. Peringatan melebihi kecepatan	164
Gambar 6.9. Peringatan berhenti melebihi waktu yang ditentukan.....	165



DAFTAR TABEL

DAFTAR TABEL

Tabel 2.1 Contoh metode komunikasi.....	39
Tabel 3.1 Perbandingan Kemampuan dan Kekurangan Metode Pelacakan.....	53
Tabel 4.1 Matrik Kondisi Perjalanan.....	77
Tabel 4.2. Spesifikasi Data Masukan.....	93
Tabel 4.3 Tabel Group.....	101
Tabel 4.4 Tabel Group.....	101
Tabel 4.5 Tabel Koper.....	101
Tabel 4.6 Tabel Koper_Jalan.....	102
Tabel 4.7 Tabel Mobil.....	102
Tabel 4.8 Tabel Perjalanan.....	103
Tabel 4.9 Tabel Posisi.....	104
Tabel 4.10 Tabel Status.....	104
Tabel 4.11 Tabel User.....	104
Tabel 4.12 Tabel Posisi_Terkini.....	105
Tabel 4.13 Tabel History_Pelanggaran.....	105
Tabel 5.1 Matriks CRUD.....	117
Tabel 6.1 Rekaman Data Perjalanan dengan Status AVL yang berubah.....	160
Tabel 6.2. Rekaman Data Perjalanan dengan Kecepatan yang Berubah-ubah	163



BAB I
PENDAHULUAN

BAB I

PENDAHULUAN

Dalam bab ini akan dijelaskan mengenai latar belakang, permasalahan yang diangkat sebagai bahasan utama, serta tujuan dan manfaat yang ingin dicapai dari pembuatan Tugas Akhir ini. Selain itu bab ini juga mendefinisikan batasan masalah, metodologi pembuatan Tugas Akhir dan sistematika pembahasan keseluruhan Tugas Akhir.

1.1. Latar Belakang

Perkembangan teknologi informasi yang sangat pesat pada dekade terakhir ini telah menyentuh hampir semua bidang dan sendi kehidupan, termasuk diantaranya adalah penggunaan teknologi informasi yang memanfaatkan sistem informasi geografis. Sistem informasi geografis mempunyai keunggulan pada penyajian hasil analisa yang bisa mendekati kenyataan di lapangan. Hal ini tentu saja harus didukung oleh detil perhitungan untuk setiap kriteria dan keakuratan data. Data yang dipergunakan bisa berupa peta, tabel, diagram bahkan hasil wawancara atau kuisioner dan survey lapangan secara langsung.

Dalam pemenuhan kebutuhan akan data terutama data mengenai posisi bisa dilakukan dengan menggunakan alat GPS (*Global Positioning System*). GPS yang diaplikasikan pada teknologi transportasi sekarang ini banyak digunakan pada sistem navigasi kendaraan, manajemen penerbangan dan juga digunakan pada manajemen pelayanan darurat. Dengan teknologi ini maka user bisa melacak posisi mereka dan juga dapat mengirimkan informasi ke kantor pusat. Dalam

perkembangannya, teknologi ini sudah dikemas dalam sebuah alat yaitu AVL (*Automatic Vehicle Location*).

Alat ini bisa juga diaplikasikan pada sektor perbankan, yang secara spesifik yaitu pada proses pengiriman atau pemindahan uang tunai dengan menggunakan kotak uang. Sayangnya selama ini proses tersebut masih dilakukan secara manual. Meskipun ada laporan secara terinci pada waktu pemberangkatan dan penyerahan ke tujuan, akan tetapi tidak dapat memonitor perjalanan kotak uang tersebut secara *real-time* dan menyimpan data-datanya ke dalam sebuah arsip.

Dengan menggunakan AVL, didukung dengan Sistem Informasi Geografis (SIG) dan GSM maka pelacakan terhadap perjalanan dari kotak uang tersebut dapat dilakukan sehingga dapat dimonitoring dan juga dapat diarsipkan. Dari hal itu, nantinya juga dapat dilakukan pencegahan - pencegahan bilamana terjadi hal-hal yang tidak diinginkan, semisal kotak tersebut dicuri atau dirampok. Bila terjadi gangguan pada alat baik disengaja atau tidak, maka alat tersebut akan mengirimkan sinyal ke kantor pusat. Setelah itu bisa dilakukan tindakan pelacakan selanjutnya.

1.2. Tujuan dan Manfaat

Tujuan dari Tugas Akhir ini adalah membuat perangkat lunak yang bisa melakukan pelacakan terhadap kotak uang yang dikirimkan dengan memanfaatkan alat AVL dan pengolahan datanya menggunakan Sistem Informasi Geografis (SIG) untuk kemudian ditampilkan ke dalam web.

Sedangkan manfaat yang dapat diambil dari tugas akhir ini adalah didapatkannya data-data proses pengiriman uang yang akan menunjang proses bisnis yang dilakukan oleh bank.

1.3. Perumusan Masalah

Permasalahan yang diusahakan penyelesaiannya dalam Tugas Akhir ini meliputi:

1. Bagaimana pembuatan dan pengolahan peta digital beserta data-data spasial yang dibutuhkan untuk sistem pelacakan ini?
2. Bagaimana menampilkan visualisasi data-data pelacakan ke dalam peta digital secara *real-time* ?
3. Bagaimana mengolah dan mentransfer hasil olahan Sistem Informasi Geografis tersebut ke dalam web ?

1.4. Pembatasan Masalah

Batasan permasalahan yang menunjang terselesaikannya Tugas Akhir ini meliputi:

1. Peta yang dipergunakan untuk simulasi adalah peta digital Institut Teknologi Sepuluh Nopember yang terdiri dari jalan beserta gedung yang ada.
2. Sistem yang dibangun menangani pelacakan pengiriman kotak uang beserta perpindahannya. Dalam simulasi ini kendaraan yang mengangkut dianggap mewakili kotak uang.
3. Pemrograman *interface* dilakukan dengan ASP dan pengolahan peta digitalnya menggunakan *MapInfo*.
4. Untuk kondisi darurat, sistem hanya akan memberikan peringatan pada aplikasi web.

5. Sistem pelacakan yang dibuat hanya memberitahukan bahwa terjadi kondisi tertentu. Mengenai bagaimana penanganan kondisi tersebut tidak termasuk dalam ruang lingkup
6. Sistem hanya mencatat transaksi pengiriman saja tidak termasuk pengaturan sumber daya mobil dan koper.

1.5. Metodologi Pelaksanaan Tugas Akhir

Metodologi penelitian yang digunakan dalam pengerjaan Tugas Akhir ini adalah sebagai berikut :

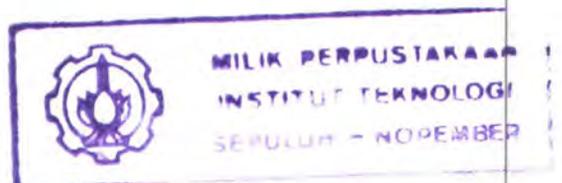
1. Studi Literatur

Tahap ini merupakan tahap pengumpulan informasi yang diperlukan untuk pembuatan perangkat lunak. Informasi yang dibutuhkan diperoleh dengan membaca beberapa literatur yang berhubungan dengan sistem pelacakan, melakukan diskusi dengan pihak yang mengetahui proses bisnis pengiriman kotak uang oleh bank, serta bagaimana mengimplementasi Sistem Informasi Geografi yang berbasis web.

2. Perancangan Perangkat Lunak

▪ Perancangan Data

Tahap ini dilakukan dengan mengumpulkan semua data yang diperlukan. Data yang berhubungan dengan topik di atas didapatkan dari beberapa pihak, yaitu dari Badan Perencanaan ITS, hasil survey lapangan dan wawancara dengan pihak-pihak terkait.



- Perancangan Proses

Tahap untuk menganalisis semua data yang didapatkan dan membuat proses analisis dalam bentuk DFD (*Data Flow Diagram*), struktur ER diagram, dan *Flowchart* diagram.

- Perancangan Antar Muka

Pada tahap ini dilakukan perancangan antar muka yang meliputi perancangan sistem menu, perancangan sistem *wizard* dan perancangan *form-form* dialog berdasarkan studi literatur yang telah dilakukan.

3. Pembuatan Perangkat Lunak

Pada tahap ini dilakukan implementasi dari rancangan yang telah dibuat. Tugas akhir ini dijalankan pada *browser* Internet Explorer 6 dan dibuat dengan menggunakan MapInfo serta ASP untuk implementasinya ke dalam web.

4. Uji Coba, Evaluasi, dan Modifikasi

Evaluasi dan menguji coba perangkat lunak yang telah selesai dikembangkan sangat perlu dilakukan. Apabila ditemukan kesalahan, maka tindakan modifikasi atau perbaikan bagi kesempurnaan perangkat lunak tersebut juga akan dilakukan.

5. Penyusunan Laporan Tugas Akhir

Penulisan buku Tugas Akhir yang merupakan dokumentasi dari pelaksanaan Tugas Akhir. Buku ini disusun secara bertahap sejak memasuki tahap studi literatur hingga tahap uji coba dan evaluasi.

1.6. Sistematika Pembahasan

Sistematika penulisan tugas akhir ini dapat dijelaskan sebagai berikut :

BAB I PENDAHULUAN

Bab ini berisi latar belakang masalah, permasalahan, tujuan dan manfaat, batasan permasalahan, metodologi yang digunakan, dan sistematika penyusunan buku tugas akhir.

BAB II TEORI PENUNJANG

Bab ini membahas konsep dasar dari proses pengiriman kotak uang, Sistem Informasi Geografis (SIG) yang meliputi latar belakang perlunya SIG, pengertian SIG, peta beserta komponen penyusun peta, pemrograman web dengan menggunakan ASP beserta komponen yang digunakan yaitu ASPMAP serta proses konsep umum mengenai pelacakan.

BAB III SISTEM PELACAKAN PADA PENGIRIMAN KOTAK UANG

Pada bab ini dibahas mengenai konsep umum pelacakan pada pengiriman kotak uang, konsep umum pelacakan yang memanfaatkan GIS, perbandingan konsep pelacakan yang ada, konsep teknis pelacakan pengiriman kotak uang yang memanfaatkan GIS, dan sekenario pada data kecil.

BAB IV PERANCANGAN PERANGKAT LUNAK

Bab ini membahas desain dan implementasi perangkat lunak. Desain perangkat lunak meliputi desain data, desain proses dan desain antar muka.

BAB V IMPLEMENTASI PERANGKAT LUNAK

Pada bab ini dibahas mengenai implementasi terhadap desain yang dilakukan pada tahap desain, termasuk didalamnya disertakan script-script yang berperan penting dalam aplikasi.

BAB VI UJI COBA DAN EVALUASI SISTEM

Pada bab ini dibahas mengenai uji coba yang dilakukan terhadap aplikasi yang dibuat. Hasil uji coba kemudian dibandingkan dengan perhitungan manual. Dari hasil perbandingan tersebut dilakukan evaluasi untuk mengetahui kemampuan dari aplikasi yang dibuat.

BAB VII KESIMPULAN DAN SARAN

Berisi kesimpulan yang diambil dari hasil uji coba dan saran untuk pengembangan selanjutnya.



BAB II
TEORI PENUNJANG

BAB II

TEORI PENUNJANG

2.1. Uang

Secara umum, uang didefinisikan sebagai barang yang disetujui oleh masyarakat sebagai alat perantara pertukaran barang dan jasa serta diterima sebagai alat pembayaran utang. Dalam arti yuridis/hukum, uang adalah alat pembayaran yang sah dan harus diterima oleh setiap orang yang memiliki tagihan pada orang lain.

2.1.1. Jenis Uang

Uang yang beredar dan berlaku dalam masyarakat dapat dikelompokkan menjadi dua jenis yaitu uang kartal dan uang giral.

a. Uang Kartal (*Chartal*)

Uang kartal adalah mata uang yang dicetak dan diedarkan oleh pemerintah melalui bank sentral. Dewasa ini, wujud uang kartal ada dua macam, yaitu uang logam dan uang kertas.

1. Uang logam.

Uang logam adalah uang yang terbuat dari logam, baik logam mulia (emas dan perak) maupun logam lain, seperti nikel, tembaga, dan aluminium. Nilai logam sebagai uang ditulis dalam satuan pecahan uang tersebut.

2. Uang kertas.

Uang kertas adalah uang yang terbuat dari kertas dan dicetak oleh bank sentral.

b. Uang Giral (*Demand Deposit*)

Uang giral adalah saldo simpanan di bank dalam bentuk giro/rekening koran yang dapat digunakan sebagai alat pembayaran dengan menggunakan cek atau surat giro. Uang ini merupakan bentuk abstrak atau pengganti dari uang kartal. Contoh uang giral yang saat ini banyak beredar di masyarakat adalah:

1. Cek.

Cek adalah surat perintah kepada bank untuk membayar sejumlah uang kepada pihak tertentu.

2. Giro

Giro adalah mekanisme pembayaran melalui bank dengan memindah bukukan sejumlah uang dari rekening tertentu kepada pihak lain yang disebutkan dalam surat giro.

3. Rekening Koran

Rekening koran adalah pengadministrasian tabungan di bank yang setiap waktu dapat diambil atau ditambah dan dapat dipergunakan sebagai alat pembayaran.

2.1.2. Fungsi Uang

Pada prinsipnya fungsi uang di dalam masyarakat dapat dikelompokkan menjadi dua kategori, yaitu fungsi asli dan fungsi turunan. Fungsi uang tersebut adalah sebagai berikut:

a. Fungsi Asli

Yang dimaksudkan fungsi asli uang adalah fungsi yang hakiki dari uang. Ada dua macam fungsi asli uang, yaitu sebagai

1. Alat perantara tukar menukar

Uang dapat mempermudah dan memperlancar proses penukaran/perdagangan. Jika pertukaran harus dilaksanakan dengan cara barter, setiap pihak yang memiliki kelebihan barang/jasa dan memerlukan yang lain harus saling mencari dan mau saling menukarkan barang/jasa yang mereka milik. Dengan adanya uang, keharusan mencari orang yang saling memiliki kesamaan keinginan tersebut tidak diperlukan lagi. Siapapun yang memiliki uang dan memerlukan barang/jasa tertentu dapat dengan leluasa menukarkannya.

2. Alat pengukur nilai/kesatuan hitung.

Uang dapat digunakan untuk menentukan nilai/harga dari berbagai jenis barang/jasa. Dengan adanya kesatuan hitung yang berupa uang, setiap jenis barang/jasa dapat diukur nilainya dengan menggunakan satuan yang sama sehingga dapat mempermudah mengukur perbandingan satu dengan yang lain.

- b. Fungsi Turunan

Fungsi turunan uang merupakan fungsi sekunder. Ada tiga macam fungsi turunan uang, yaitu sebagai:

1. Alat penyimpan nilai/kekayaan

Orang yang memiliki kekayaan dapat menyimpannya dalam bentuk barang atau uang. Menyimpan kekayaan dalam bentuk barang, banyak faktor yang harus diperhitungkan seperti tempat penyimpanan, keamanan/resiko hilang, dan resiko kerusakan. Untuk menghindari resiko, uang dapat disimpan di bank. Di samping lebih ringan, menyimpan kekayaan dalam bentuk uang lebih fleksibel penggunaannya.

2. Alat pembayaran berjangka

Penjualan kredit dapat meningkatkan daya beli konsumen sehingga mendorong majunya dunia perdagangan khususnya dan perekonomian pada umumnya. Dengan uang, penjual(kreditur) memiliki standar untuk menentukan jumlah kewajiban yang harus dibayar oleh pembeli(debitur).

2.2. Bank

Menurut Undang-Undang Republik Indonesia tentang Perbankan, Nomor 7, Tahun 1992, bank adalah badan usaha yang menghimpun dana dari masyarakat dalam bentuk simpanan dan menyalurkannya kepada masyarakat dalam rangka meningkatkan taraf hidup rakyat banyak.

2.2.1. Bank Sentral

Bank sentral adalah bank yang ditunjuk oleh pemerintah untuk mencetak dan mengedarkan alat pembayaran yang sah. Karena tugasnya mengurus dan mengatur peredaran uang di dalam negeri, bank sentral disebut juga bank sirkulasi. Dalam hal ini kedudukan bank sentral bersifat monopoli, artinya tidak

ada bank lain yang boleh menyainginya. Di tiap-tiap negara, bank sentral memiliki kedudukan dan peranan istimewa dalam perekonomian.

Bank sentral biasanya dipergunakan sebagai alat politik pemerintah dalam bidang moneter untuk mengatur tingkat kestabilan ekonomi. Bank ini merupakan badan resmi yaitu badan milik pemerintah, bukan badan swasta.

Bank sentral yang merupakan pemegang monopoli untuk mencetak dan mengatur peredaran uang giral maupun kartal, memiliki tugas sebagai berikut :

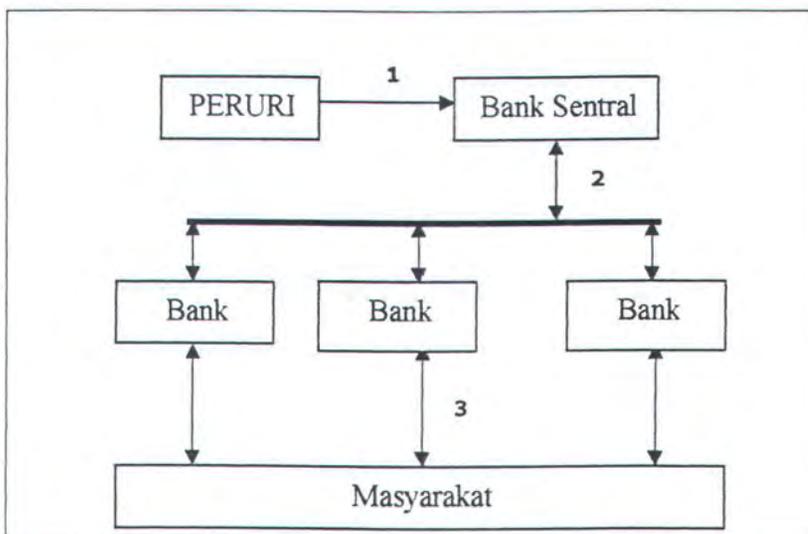
1. Menjaga dan memelihara kestabilan nilai uang baik untuk lalu-lintas pembayaran dalam negeri maupun luar negeri.
2. Menyelenggarakan dan mengatur peredaran uang.
3. Memperluas, memperlancar, dan mengatur lalu-lintas pembayaran uang giral.
4. Menyelenggarakan kliring bank.
5. Menentukan tingkat dan struktur bunga serta menetapkan pembatasan kualitas (jenis) dan kuantitas (jumlah) pemberian kredit oleh bank-bank lain.
6. Memberikan kredit likuiditas kepada bank lain dengan maksud meningkatkan produktifitas dan memberikan kredit tingkat akhir untuk membantu likuiditas pada saat bank tersebut mengalami keadaan kritis.
7. Bertindak sebagai pemegang kas pemerintah dan memberikan jasa perantara kepada pemerintah.
8. Menguasai dan mengurus tata usaha cadangan emas dan devisa milik negara.

2.2.2. Bank Pelaksana

Menurut peraturan Bank Indonesia Nomor 5/20/PBI/2003, Bank Pelaksana adalah bank penerima fasilitas Kredit Likuiditas Bank Indonesia (KLBI) dalam rangka kredit program. Bank Pelaksana ada yang merupakan BUMN milik negara misal: Bank BNI, Bank BRI, dan ada juga yang merupakan milik swasta seperti: Bank BCA, City Bank. Secara garis besar, bank pelaksana merupakan bank yang bertugas untuk melayani transaksi secara langsung dengan masyarakat.

2.3. Arus Perputaran Uang

Dengan berputarnya roda perekonomian maka terjadi peredaran uang di masyarakat baik uang kartal maupun uang giral. Perputaran uang ini terjadi di setiap lini baik di masyarakat tingkat bawah, tingkat menengah dan tingkat atas. Melihat wilayah Indonesia yang berkepulauan, perputaran uang yang terjadi juga semakin kompleks. Secara sederhana, arus perputaran uang yang terjadi saat ini dapat digambarkan pada bagan berikut :



Gambar 2.1 Arus perputaran uang

Keterangan :

1. Uang kertas dan uang logam dicetak di Perum Peruri . Uang tersebut kemudian dikirimkan ke bank sentral yaitu BI di seluruh wilayah Indonesia.
2. BI mengedarkan uang baru ke bank-bank pelaksana yang lain seperti BRI, BNI, BCA ataupun Bank Perkreditan Rakyat. Proses yang terjadi adalah timbal balik. Selain arus uang yang berasal dari BI ke bank-bank pelaksana, tiap akhir minggu bank-bank tersebut melakukan penyetoran uang ke BI.
3. Bank bertransaksi dengan masyarakat sehingga uang beredar di masyarakat, baik melalui kredit ataupun untuk modal kerja. Arus yang terjadi juga merupakan timbal balik.

2.4. Pengiriman uang

Dalam peredaran uang, tentunya ada proses perpindahan uang dari pihak yang satu kepada pihak yang lain. Semakin besar nilai uang, resiko yang ditanggung juga semakin besar, khususnya pada pengiriman uang antar bank ataupun antara bank dengan nasabah korporat.

Dalam proses pengiriman, bisa saja timbul kendala. Gangguan yang mungkin timbul selama proses pengiriman bisa disebabkan karena faktor teknis dan bisa juga faktor nonteknis. Beberapa contoh kendala yang mungkin saja terjadi bisa seperti hal berikut: perampokan, pencurian, kemacetan ataupun bencana alam semisal banjir atau tanah longsor. Bila sudah faktor alam sebagai penyebab, sulit untuk menanggulangnya.

Proses pengiriman yang selama ini terjadi yaitu:

1. Pengiriman dari peruri ke bank sentral

Dalam pengiriman uang dari peruri ke bank sentral BI, uang dikemas dalam sebuah peti. Pengangkutan untuk jalan darat dengan menggunakan mobil khusus yaitu *securicor*, terkadang juga menggunakan truk. Bila jalan darat tidak memungkinkan maka digunakan kapal laut atau pesawat terbang.

2. Pengiriman dari bank sentral ke bank pelaksana.

Uang dikirim dari Bank Indonesia ke bank pelaksana untuk diedarkan lebih lanjut.

3. Pengiriman dari bank pelaksana ke nasabah

Pengiriman uang dari bank ke nasabah biasanya merupakan tanggung jawab dari nasabah itu sendiri.

Berikut adalah gambaran teknis secara umum dalam sebuah pengiriman yang biasa dilakukan dengan menggunakan kendaraan truk atau *securicor*.

1. Tahap inisialisasi

Pengirim melakukan kontak dengan pihak yang dituju bahwa akan dilaksanakan pengiriman. Tahap ini biasa dilakukan untuk pengiriman yang tidak terjadwal. Untuk pengiriman yang terjadwal biasanya jarang dilakukan karena sudah otomatis. Pengiriman terjadwal yang dimaksud adalah penyetoran kas dari bank pelaksana ke bank sentral tiap akhir minggu.

2. Tahap pemberangkatan

Pada tahap ini kesiapan perlengkapan dan peralatan diperiksa. Bilamana sudah siap maka dilakukan pemberangkatan. Kurir biasanya disertai dengan pengawalan dari pihak keamanan.

3. Tahap perjalanan

Pada tahap ini kurir harus memberi tahu posisi dan kedudukan terakhir selama perjalanan. Hal-hal yang mencurigakan ataupun kejadian penting selama perjalanan harus dilaporkan ke pihak pengirim. Dalam proses ini uang yang dikirim tidak boleh disentuh oleh siapapun.

4. Tahap serah terima

Setelah sampai di tujuan, maka dilaksanakan proses serah terima. Kurir melapor ke pihak pengirim dan pihak penerima menandatangani surat serah terima.

5. Proses selesai.

2.5. Sistem Informasi Geografis

Pada pertengahan abad ke-20 masyarakat semakin sadar akan pentingnya kebutuhan informasi yang terkait dengan data-data geografis. Pada saat yang sama dampak negatif dari kemajuan teknologi telah menunjukkan perlunya pengaturan sumber daya alam secara benar. Untuk itulah diperlukan suatu sistem yang mampu mengatur dan mengolah data-data geografis sehingga mampu menyajikan suatu informasi yang akurat dan terpadu. Sistem informasi ini disebut sebagai *Sistem Informasi Geografis (SIG)*.

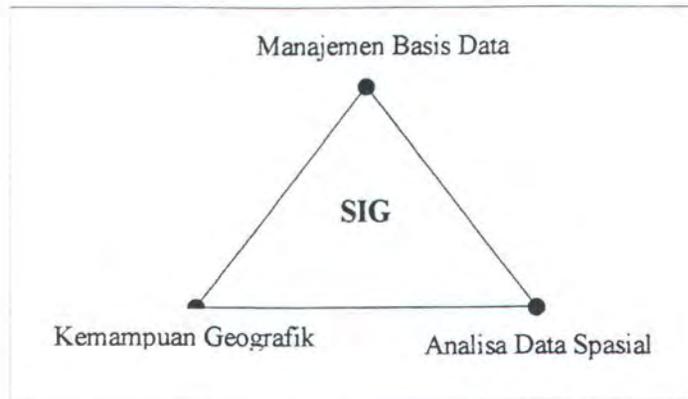
2.5.1. Pengertian Sistem Informasi Geografis

Istilah Sistem Informasi Geografis pertama kali diperkenalkan pada suatu diskusi makalah tahun 1965 di Universitas *Northwestern* yang dibawakan oleh Michael Dacey dan Duane Marble. Dalam makalah tersebut disimpulkan bahwa suatu sistem yang digunakan untuk pemetaan dan pemrosesan informasi spasial termasuk di dalamnya penggambaran teknik, pengolahan dan manajemen *database* geografis, atau lebih kompleks lagi adalah *analisa geografis* dan *modeling-nya*.

Sistem Informasi Geografis mempunyai beberapa batasan. SIG adalah sistem penanganan data keruangan, (*Marble et al*, 1983) . SIG adalah alat yang bermanfaat untuk mengumpulkan, penimbunan, dan pengambilan kembali data-data yang diinginkan, pengubahan, dan penayangan yang berasal dari kenyataan dunia. SIG adalah sistem informasi, referensi internal, otomatisasi dan keruangan.

Menurut Phil Parent (1988), SIG adalah sistem yang memuat data-data spasial yang dapat dianalisa dan dikonversi menjadi informasi untuk tujuan dan aplikasi tertentu, dimana aplikasi khas dari SIG adalah *analisa data* untuk mendapatkan informasi baru.

Sedangkan menurut Francis Hanigan (1988) , Sistem Informasi Geografis merupakan sistem manajemen informasi yang memiliki kemampuan untuk mengumpulkan, menyimpan, dan mengambil informasi berdasarkan lokasi spasial, mengidentifikasikannya, mencari hubungan antar data dalam lingkungannya, serta menganalisa data spasial yang berhubungan sebagai alat bantu pengambilan keputusan.



Gambar 2.2 Tiga Aspek Sistem Informasi Geografis

2.5.2. Unsur Utama Sistem Informasi Geografis

Ada banyak aspek yang perlu diperhatikan untuk membentuk suatu Sistem Informasi Geografis. Aspek-aspek tersebut adalah :

2.5.2.1. Sumber data

Agar dapat berfungsi dan memberikan informasi dari hasil analisisnya SIG memerlukan data masukan yang dapat diperoleh dari tiga sumber , yaitu :

- a. **Data lapangan.** Diperoleh dari pengukuran di lapangan secara langsung. Contohnya pH tanah, salinitas air, curah hujan suatu wilayah.
- b. **Data digital.** Informasi yang telah terekam pada kertas atau film, dan dikonversikan ke dalam bentuk digital. Data lapangan masih diperlukan untuk mengecek kebenaran data digital ini.
- c. **Data citra penginderaan jauh.** Citra penginderaan jauh yang berupa foto udara atau radar harus diinterpretasikan terlebih dahulu sebelum dikonversi ke bentuk digital. Sedangkan citra yang diperoleh dari satelit yang sudah dalam bentuk digital dapat langsung digunakan.

Ketiga sumber data tersebut saling mendukung satu dengan yang lainnya.

2.5.2.2. Jenis Data

Dalam Sistem Informasi geografis terdapat empat jenis data, yaitu :

- a. **Data spasial.** Data yang menunjukkan lokasi dan bentuk permukaan bumi yang terdiri dari titik, garis, dan poligon. Contoh data ini adalah peta.
- b. **Data atribut.** Data yang menggambarkan karakteristik suatu lokasi seperti data sosial, ekonomi, demografi, lingkungan, kesehatan, dan sebagainya.
- c. **Data numerik.** Data berupa angka-angka yang menyatakan jumlah atau kuantitas.
- d. **Data digital.** Data yang berwujud citra/gambar, foto, suara, dan animasi.

2.5.3. Komponen Informasi Kenampakan Geografis

Informasi kenampakan geografi mempunyai empat komponen utama, yaitu: posisi geografi, atribut, hubungan keruangan (*spasial relationships*), dan waktu.

Posisi geografis dapat dinyatakan dalam sistem koordinat Lintang/Bujur (*Latitude/Longitude*) atau sistem UTM (*Universal Transverse Mercator*).

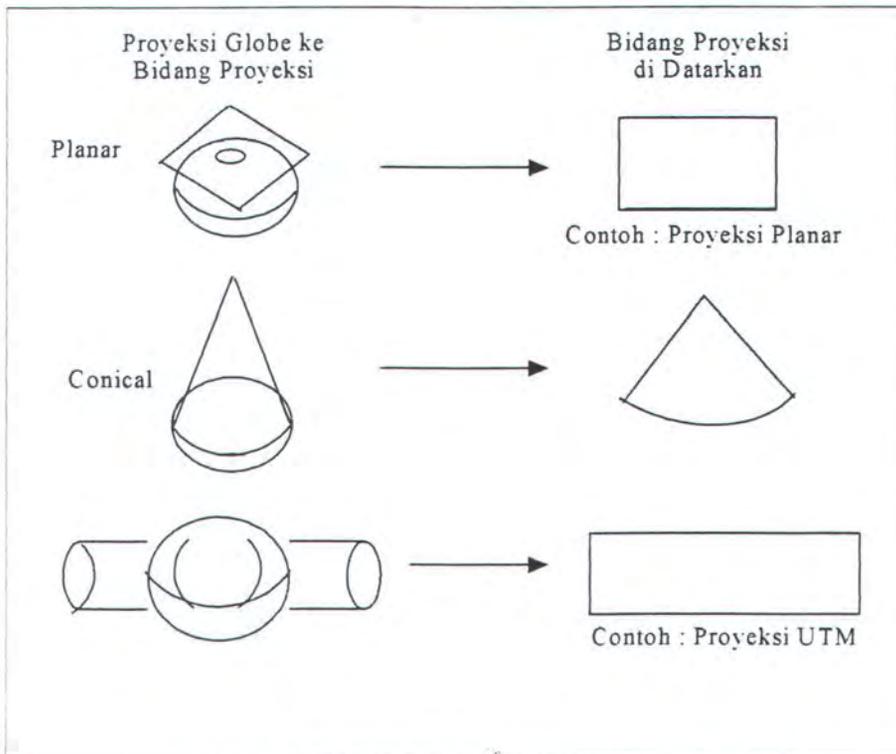
Atribut menjelaskan informasi yang mengarah pada pertanyaan apa (*what it is*). Contoh atribut misalnya: hutan, sawah atau tegalan, kota/daerah, dan sebagainya. Atribut tambahan yang tidak berhubungan dengan posisi geografis disebut sebagai atribut non-keruangan (*non-spatial attribute*).

Waktu merupakan komponen SIG yang perlu juga untuk diperhatikan, karena bisa saja suatu informasi sudah menjadi tidak valid lagi, terutama informasi yang berkaitan dengan waktu.



2.5.4. Kemampuan Pengolahan Data Grafik dan Non Grafik

Dalam sistem informasi geografis telah ada kemantapan keberadaan cabang disiplin ilmu pemetaan (kartografi). Kartografi berasal dari teknik penyajian informasi secara manual. Karena obyek studi geografis merupakan bagian dari bola bumi, maka permasalahan utama dalam kartografi adalah teknik penyajian obyek *spheris* dari bola bumi ke dalam obyek linier pada lembar peta. Secara skematis, beberapa pendekatan standar dalam kartografi dapat disajikan pada gambar 2.3.



Gambar 2.3 Teknik proyeksi peta dalam kartografi5

Sedangkan untuk pengaturan data non grafik, dimungkinkan penyimpanan dan manipulasi data non-grafik secara efisien. Penyimpanan dan pengambilan data non-grafik yang terhubung ke data grafik sering disebut sebagai pemrosesan atribut (*attribute processing*).

2.6. Peta

Peta berasal dari bahasa latin *Mappa* yang berarti kain, yang merupakan representasi dari suatu permukaan bumi dalam ukuran yang diperkecil. Peta adalah gabungan dari beberapa bentuk seperti titik, garis, dan poligon/wilayah yang didefinisikan dari lokasinya dengan memakai acuan sistem koordinat maupun pada atribut non-spasialnya.

Informasi yang terkandung dalam peta (apapun ukuran dan tujuannya) mempunyai beberapa ciri umum, yaitu :

- **Grid dan sistem koordinat**

Sistem *grid* dapat memudahkan penentuan lokasi pada suatu peta. Titik perpotongan antara garis vertikal dan garis horizontal yang sejajar dengan garis-garis penyusun berpotongan pada sebuah titik yang disebut koordinat. Daerah perpotongan dari garis-garis yang berbentuk persegi empat itulah yang disebut *Grid*. Sistem koordinat yang dipakai pada peta adalah sistem koordinat kartesius yang merepresentasikan koordinat suatu titik sebagai pasangan x dan y .

- **Skala**

Merupakan perbandingan jarak pada peta dengan jarak sebenarnya pada permukaan bumi.

- **Jarak**

Sama halnya dengan skala, jarak juga memiliki jenis yang sama dengan skala, yaitu : tepat, proporsional, dapat diperbesar/diperkecil.

- **Batas**

Pada umumnya peta menggambarkan batas sebagai untaian segmen garis atau sering disebut sebagai string.

- **Simbol**

Dipakai untuk merepresentasikan obyek yang tidak dapat direpresentasikan dalam skala.

- **Legenda peta (*map legend*)**

Legenda peta berisi daftar arti dari simbol-simbol penyusun peta, beserta semua aturan yang dipakai pada peta tertentu.

- **Relief dan kontur**

Relief menunjukkan karakteristik bentuk permukaan tanah seperti tinggi dan kemiringan lereng, sedangkan garis kontur digunakan untuk menunjukkan ketinggian pada peta, khususnya pada peta topografi.

- **Isogram / isoline**

Isogram/ isoline adalah garis-garis yang menghubungkan titik-titik pada peta yang bernilai sama.

2.6.1. Elemen Grafik Penyusun Bentuk Dasar Peta

Untuk menyusun bentuk dasar peta digunakan beberapa elemen data gambar, antara lain :

- **Titik (*point*)**

Digunakan untuk menentukan suatu lokasi geometris, didefinisikan sebagai obyek berdimensi nol.

- **Garis (*line*)**

Merupakan kumpulan titik yang diawali dan diakhiri oleh node. Node adalah perpotongan titik dimana dua atau lebih garis bertemu. Garis didefinisikan sebagai obyek berdimensi satu.

- Wilayah (*area*)

Suatu *area* direpresentasikan sebagai poligon, yang terdiri dari garis yang membentuk ruang tertutup. *Area* didefinisikan sebagai obyek berdimensi dua yang kontinyu dan membentuk suatu batas.

- *Pixel*

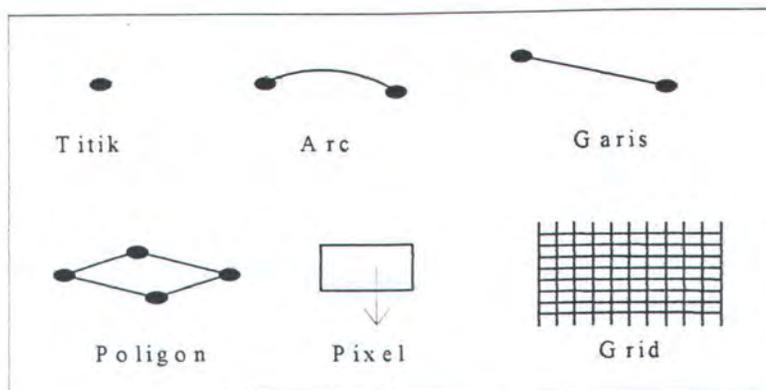
Pixel didefinisikan sebagai elemen gambar berdimensi dua yang merupakan elemen terkecil dari suatu gambar yang tidak dapat dibagi lagi.

- Sel *grid*

Sel *grid* merupakan obyek berdimensi dua yang merepresentasikan sebuah elemen tunggal dari suatu permukaan yang kontinyu.

- Simbol

Merupakan elemen grafik yang merepresentasikan sebuah titik pada suatu peta.



Gambar 2.4 Enam elemen grafik penyusun peta

2.6.2. Sistem Koordinat

Untuk merepresentasikan posisi geografis bumi maupun obyek yang berada pada permukaan bumi digunakan sistem koordinat. Sistem koordinat yang umum digunakan adalah sistem koordinat geografis (bujur dan lintang) dan sistem proyeksi *Universal Transversal Mercator* (UTM).

A. Sistem Koordinat Geografik

Sistem koordinat ini merepresentasikan bumi ke dalam bentuk dua dimensi, lalu membagi bumi dengan garis-garis bujur dan lintang untuk merepresentasikan setiap titik dari permukaan bumi. Satuan yang digunakan adalah derajat (*degree*) dengan urutan derajat, menit, detik. *Equator* merupakan garis lintang nol derajat, sedangkan bujur nol derajat melewati kota Greenwich. Kelebihan sistem koordinat ini adalah dapat merepresentasikan setiap titik di permukaan bumi secara tepat, namun tidak bisa merepresentasikan jarak sebenarnya dari dua titik.

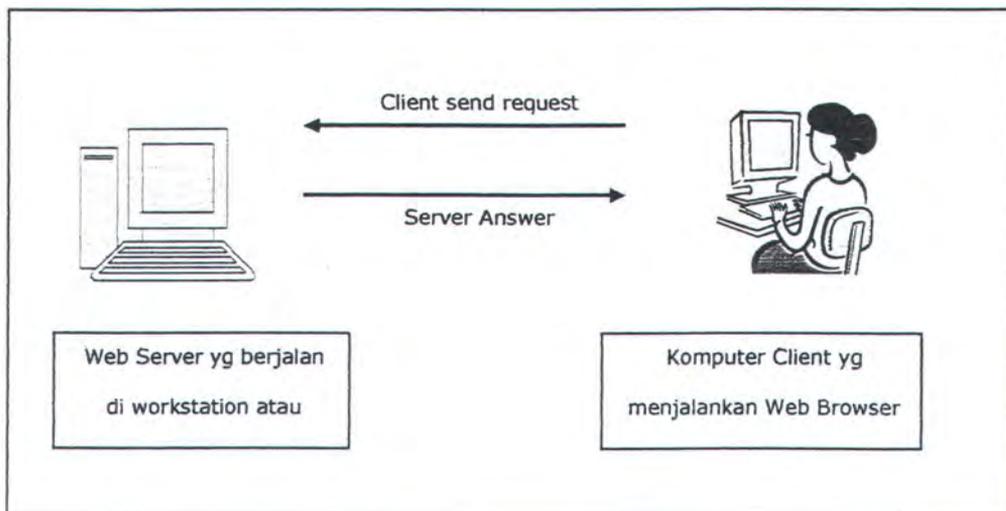
B. Sistem Proyeksi UTM

Sistem ini merepresentasikan bumi dengan cara seperti membungkus bumi dalam pita-pita silinder membujur dan membagi bumi ke dalam 60 zone, dengan setiap zone lebarnya 6 derajat. Penomoran zone mulai dari 1 – 60 (dihitung dari 180 derajat bujur barat), batasan lintang dari 84 derajat lintang utara sampai 80 derajat lintang selatan, dan satuan yang digunakan adalah meter. Kelebihan dari sistem ini adalah bisa merepresentasikan jarak sebenarnya antara dua titik. Sedang

kelemahannya adalah bentuk bumi terutama di bagian kutub menjadi seperti dilebarkan, karena setiap zone dari UTM pasti melewati bagian kutub.

2.7. ASPMAP

Pemrograman ASP (*Active Server Pages*) merupakan program script sebagai alat bantu pembuatan HomePage. Halaman-halaman web akan bergerak secara dinamis dengan adanya program script, sedangkan form-form HTML cenderung statis. Gambaran umum suatu Web adalah dengan pengolahan data dari client yang meminta layanan halaman ke WebServer. Untuk sistem Windows NT, maka harus memakai Web Server yang sudah menunjang ASP, IIS ver 4 atau versi dibawahnya dengan patch yang telah tersedia.



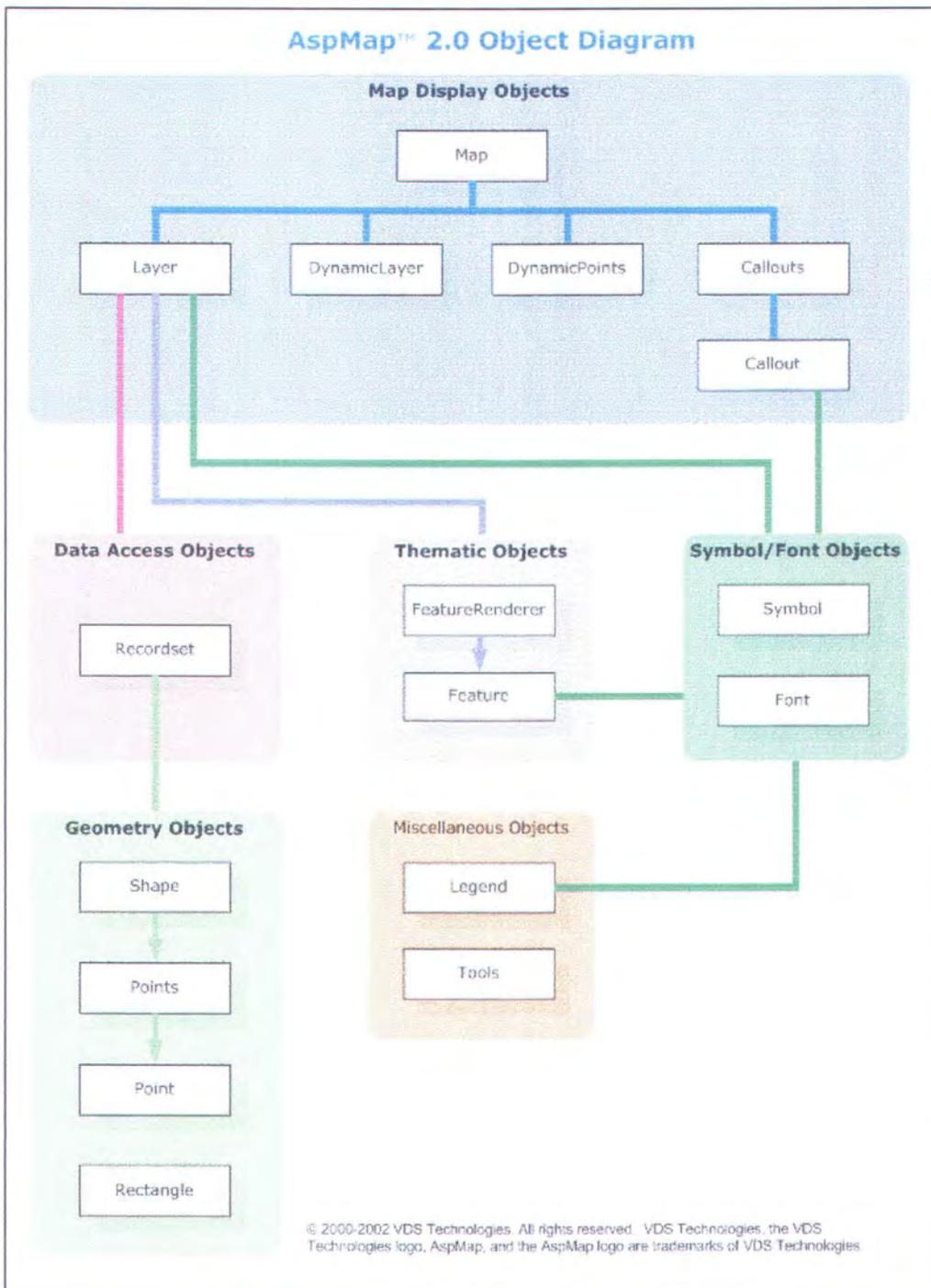
Gambar 2.9 Uraian umum Web Server – Client

AspMap adalah sekumpulan komponen yang berada di server yang berfungsi untuk menampilkan peta ke dalam aplikasi sebuah web. *AspMap* sendiri bisa diakses dengan menggunakan bahasa pemrograman ASP, ASP.NET, Visual Basic, dan bahasa pemrograman lain yang support terhadap COM dan .NET.

Dengan menggunakan komponen ini kita bisa melakukan :

- a. Menampilkan sebuah peta dengan berbagai *layer* semisal kota, jalan, sungai dan *boundry* serta lokasi yang tersimpan dalam *database*.
- b. Menampilkan lokasi dalam peta dari titik koordinat yang tersimpan dalam sebuah *database*.
- c. Pada saat ingin melihat lebih detail kita bisa memperbesar objek peta yang ingin dilihat.
- d. Dapat mengidentifikasi dari fitur yang ditampilkan hanya dengan klik *mouse* pada peta.
- e. Menggambar komponen grafis seperti *point*, garis, elips, persegi dan *polygon*, komentar dan petunjuk.
- f. Memilih fitur-fitur sepanjang garis, didalam kota, area, *polygon*, dan lingkaran.
- g. Dapat memilih fitur-fitur yang berada dalam batas jarak tertentu dari fitur yang lain.
- h. Dapat melakukan operasi *query* pada sekumpulan data yang berhubungan dengan fitur yang telah dipilih.
- i. Dapat melakukan operasi *render* pada sebuah *layer* dengan berdasarkan nilai yang terkandung atau juga berdasarkan ekspresi logika tertentu.
- j. Bisa menambahkan label pada fitur berupa teks yang berasal dari nilai *field*.
- k. Gambar peta yang telah dibuat bisa diekspor ke dalam format gambar yang standard semisal *.BMP, *.JPG, *.P
- l. Dapat menampilkan data *real time* secara dinamis.

Berikut adalah object diagram dari AspMap 2.0



Gambar 2.10 Object Diagram AspMap 2.0



2.7.1. Format Peta

Format data yang bisa digunakan oleh AspMap adalah sebagai berikut :

a. Bentuk vector :

▪ Shapefile (*.shp)

Merupakan format non-topologi sederhana untuk menyimpan lokasi geometris dan informasi atribut dari fitur geografis.

Koordinat pada *shapefile* menggunakan system koordinat Cartesian.

Format shapefile mendefinisikan bentuk geometris dan atribut dari fitur ber-referensi geografis pada beberapa file dengan suatu ekstensi file tertentu yang disimpan pada folder yang sama dalam disk. Ekstensi file tersebut adalah :

- shp – file yang menyimpan fitur geometris
- .shx – file yang menyimpan indeks dari fitur geometris
- .dbf – file dBASE yang menyimpan atribut informasi dari fitur tersebut.

Geometri dari tiap fitur disimpan sebagai suatu bentuk yang terdiri dari kumpulan koordinat *vector*. Atribut dari tiap fitur akan disimpan sebagai *record* pada table dBASE (.dbf) yang akan diasosiasikan dengan *shapefile* (.shp), sehingga akan ada satu *record* pada *table* dBASE untuk tiap fitur pada file utama.

▪ TAB (*.tab)

Format TAB adalah format non-topologi yang sederhana, untuk menyimpan lokasi geometris dan atribut informasi dari fitur geografis.

Koordinat pada file TAB menggunakan system koordinat Cartesian. Format TAB mendefinisikan geometris dan atribut dari fitur geografis pada beberapa file dengan suatu ekstensi file tertentu yang disimpan pada folder yang sama dalam disk. Ekstensi file tersebut adalah :

- .tab – file utama
- .map – file yang menyimpan fitur geometris
- .id – file dBASE yang menyimpan indeks dari fitur geometris
- .dat – file dBASE yang menyimpan atribut informasi fitur
- .ind – indeks field tabel

Geometri dari tiap fitur disimpan sebagai suatu *shape* yang terdiri dari kumpulan koordinat *vector*. Atribut dari tiap fitur disimpan sebagai suatu *record* dari table dBASE (.dat) yang diasosiasikan dengan *shapefile* (.map). Sehingga akan ada satu *record* pada table dBASE untuk tiap fitur pada *map file*.

b. Bentuk Raster :

- TIFF

TIFF mensupport warna *image* hitam-putih, *grayscale*, *pseudocolor* dan warna *image* yang sebenarnya, dimana kesemuanya dapat disimpan dalam format terkompresi atau tidak terkompresi.

Macam TIFF berikut didukung :

- TIFF 6.0 (TIFF *revision* 6.0).
- GeoTIFF (TIFF dengan *header* Geo)

- ECW

Enhanced Compressed Wavelet (ECW) adalah format *image raster* yang terkompresi, didesain untuk efisiensi penyimpanan dan pengambilan *image* digital yang berkualitas tinggi dan berukuran besar.

AspMap menyediakan dukungan pada *image ECW 2.x* dan menggunakan file DLL berikut :

- EFECW.DLL - *AspMap* DLL;
- NCSUTIL.DLL - ECW *runtime* DLL;
- NCSCNET.DLL - ECW *runtime* DLL;
- NCSECW.DLL - ECW *runtime* DLL.

2.7.2. Map Object

Objek *Map* menampilkan koleksi dari *layer – layer*. Objek *Map* dapat digunakan untuk menambah dan menghilangkan *layer*, mengubah tingkat *layer*, memperbesar ataupun memperkecil peta, digunakan untuk mengubah sistem koordinat antara dunia dan gambar, dan sebagainya. Objek *Map* juga menyimpan informasi tentang peta seperti property *BackColor*, *Extent*, dan *FullExtent*.

2.7.2.1. Layer Object

Objek *Layer* merepresentasikan *layer* data geografis pada sebuah peta. Segiempat pembatas dari sebuah *Layer* disimpan dalam property *Extent*. Sebuah *recordset* yang berhubungan dengan *Layer* dapat diambil dengan property *Recordset*. Properti *Renderer* digunakan untuk menyimbolkan fitur dalam sebuah *layer*. Secara umum, objek *Layer* menggambar dengan *Symbol* tunggal.

2.7.2.2. DynamicLayer Object

Objek *DynamicLayer* dapat digunakan untuk menambah atau memanipulasi grafik dan teks khusus pada sebuah peta. Objek *DynamicLayer* berisi nol atau lebih elemen. Sebuah elemen diwakili dengan sebuah bentuk dan sebuah teks *string*.

2.7.2.3. DynamicPoint Object

Objek *DynamicPoint* mewakili satu set titik dalam sebuah ruang. Objek *DynamicPoint* digunakan untuk melacak sebuah kendaraan, pesawat terbang, kapal, atau fenomena geografis. Objek ini berisi nol atau lebih elemen yang diwakili dengan sebuah bentuk dan sebuah teks *string*.

2.7.2.4. CallOuts Object

Sebuah *Callouts* adalah koleksi dari objek gambar *callout*. *Callouts* terdiri dari sebuah *text box* dan segmen garis dengan batas segiempat.

2.7.2.5. Recordset Object

Objek *Recordset* merepresentasikan *record – record* yang berhubungan dengan sebuah *Layer*, atau *record – record* yang dihasilkan dari sebuah *query*.

2.7.2.6. FeatureRenderer Object

Pengubahan *layer* peta menurut nilai/symbol atau pasangan ekspresi/symbol. *FeatureRenderer* mempunyai dua cara pengubahan, yaitu :

1. Mengubah peta menurut nilai spesifik dari sebuah data.
2. Mengubah peta menurut ekspresi logika.

2.7.2.7. Feature Object

Objek *Feature* mewakili level terkecil dari sebuah objek *Layer*, seperti sebuah jalan, sebuah gedung, dan lain sebagainya.

2.7.2.8. Symbol Object

Objek *Symbol* terdiri dari atribut – atribut yang mengatur cara menampilkan fitur. Pemilihan gaya dan atribut dari *Symbol* bergantung pada jenis fitur yang dikerjakan.

2.7.2.9. FontObject

Objek *Font* berisi informasi yang dibutuhkan untuk membentuk teks yang akan ditampilkan pada objek *Layer*.

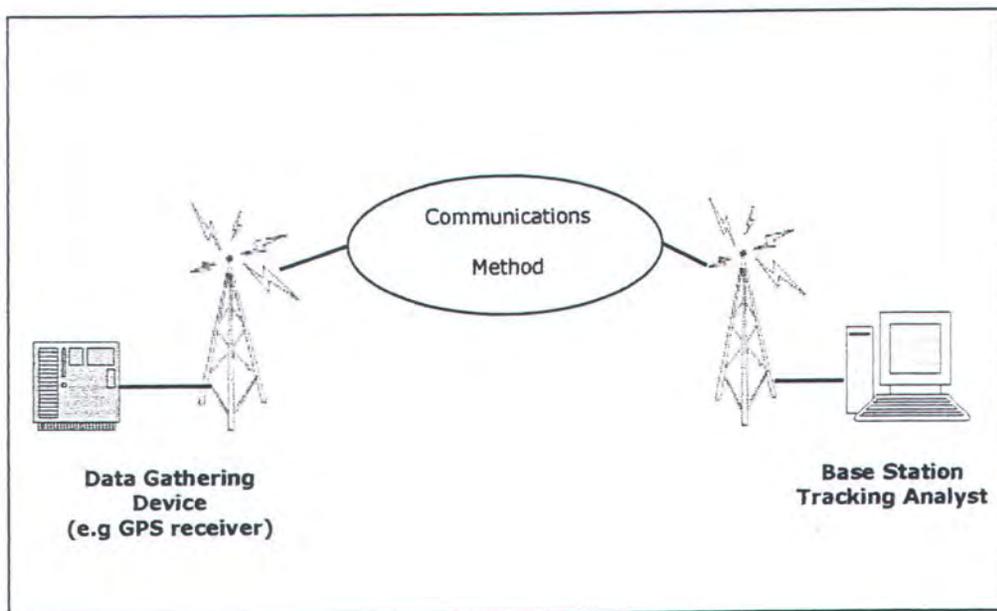
2.8. Sistem Pelacakan Secara Umum

Secara istilah kata, lacak berarti mengikuti, jejak, bekas, membawa. Hal itu berlaku juga bagi istilah pelacakan yang berarti mengikuti sesuatu dengan maksud untuk memonitor dan memastikan bahwa segala sesuatu berjalan dengan baik. Dalam sub bab ini akan dibahas mengenai pelacakan secara umum, teknologi yang dapat digunakan beserta komponen-komponen yang termasuk di dalamnya.

2.8.1. Teknologi Komunikasi

Kemampuan untuk melacak sesuatu secara *real-time* maupun dengan proses update statik, telah terhambat oleh kurang tersedianya teknologi komunikasi dan sensor yang mencukupi. Bagaimanapun, dengan ditemukannya baru-baru ini yakni strategi-strategi komunikasi *multiple layer* yang menggunakan selular tingkat tinggi dan satelit-satelit, memungkinkan terciptanya aplikasi pelacakan

real-time yang meliputi seluruh dunia. Teknologi sensor juga meningkat secara drastis dalam kemampuannya seiring dengan penurunan dalam ukuran. Hal tersebut memperluas jenis-jenis dari aplikasi pelacakan yang mungkin. Meskipun beberapa teknologi pelacakan cenderung mahal, memerlukan perluasan *coverage area* dan peningkatan pemakaian, namun akan cepat dalam menurunkan biaya untuk membangun teknologi ini.

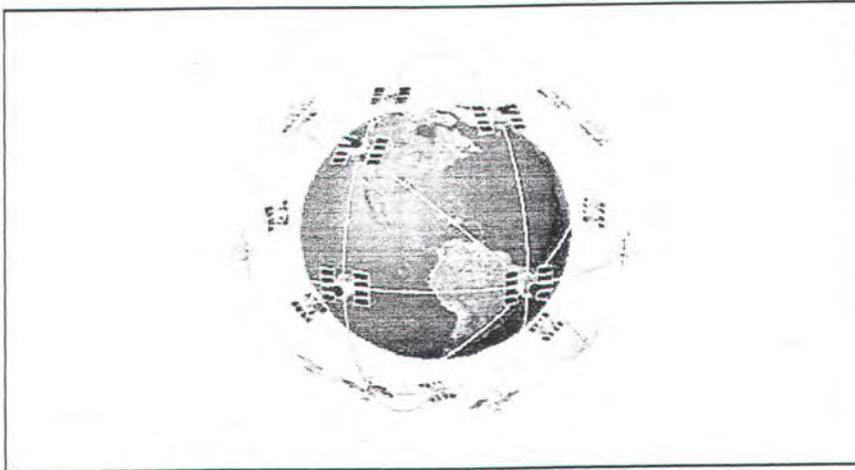


Gambar 2.11 Komunikasi secara global

2.8.2. Global Positioning Primer

Kunci diantara teknologi-teknologi pelacakan yang ada salah satunya adalah *Global Positioning System* (GPS). GPS merupakan suatu bentuk teknologi informasi yang menggunakan sistem perangkat keras dan perangkat lunak, serta informasi yang ditransmisi dari satelit-satelit untuk menyediakan informasi turunan bagi pengguna. Informasi turunan ini (waktu, posisi, dan kecepatan) dapat

dikombinasi dengan sistem lainnya seperti alat komunikasi dan perangkat lunak untuk analisis.



Gambar 2.12 Posisi satelit penyedia layanan GPS

Informasi seperti garis lintang (*latitude*), garis bujur (*longitude*), ketinggian (*altitude*), dan waktu, disediakan GPS *receiver* untuk proses lebih lanjut oleh pengguna. Informasi tersebut dapat diterjemahkan ke bentuk yang lebih mudah untuk dimengerti, seperti posisi relatif terhadap peta yang ada, atau digunakan untuk menyediakan informasi seperti jarak dari titik-titik yang didefinisikan pengguna (misalnya posisi yang secara spesifik diidentifikasi oleh pengguna).

Aplikasi-aplikasi GPS dapat digunakan dalam menjawab tiga jenis pertanyaan, antara lain:

- Mengetahui posisi pemakai GPS itu sendiri

Contohnya: pendaki gunung, pengendara, pilot, dan *surveyor*.

- Mengetahui informasi GPS untuk banyak pengguna

Contohnya: seorang manajer armada truk dapat menggunakan GPS dan telepon selular untuk berkomunikasi dengan pengendara truk,

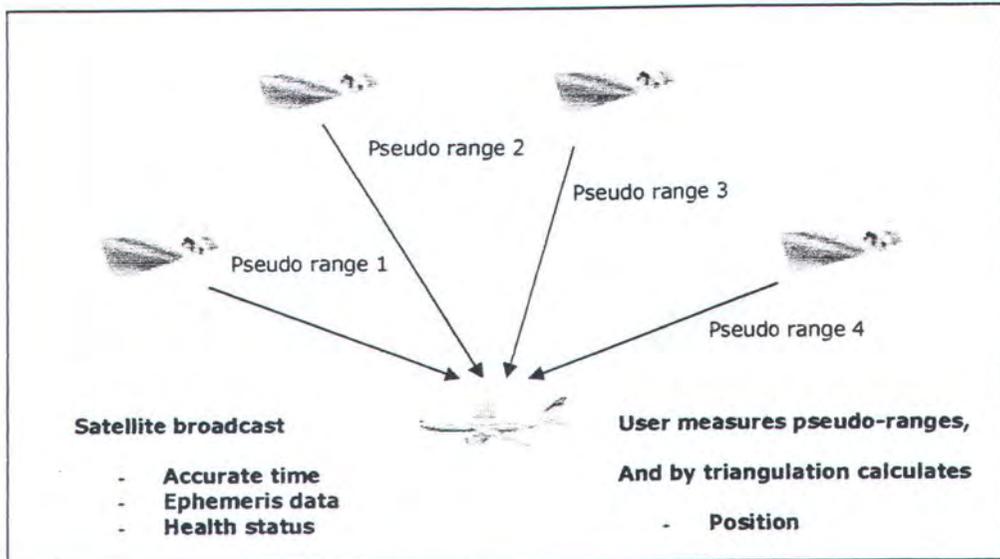
mengatur traffic pesawat terbang internasional dengan ratusan pesawat terbang yang bergerak di pola yang kompleks setiap saat.

- Mengetahui posisi benda tertentu

Contohnya adalah untuk melacak aset, seperti pengiriman cargo, atau mengatur paket-paket data dalam jaringan informasi yang padat.

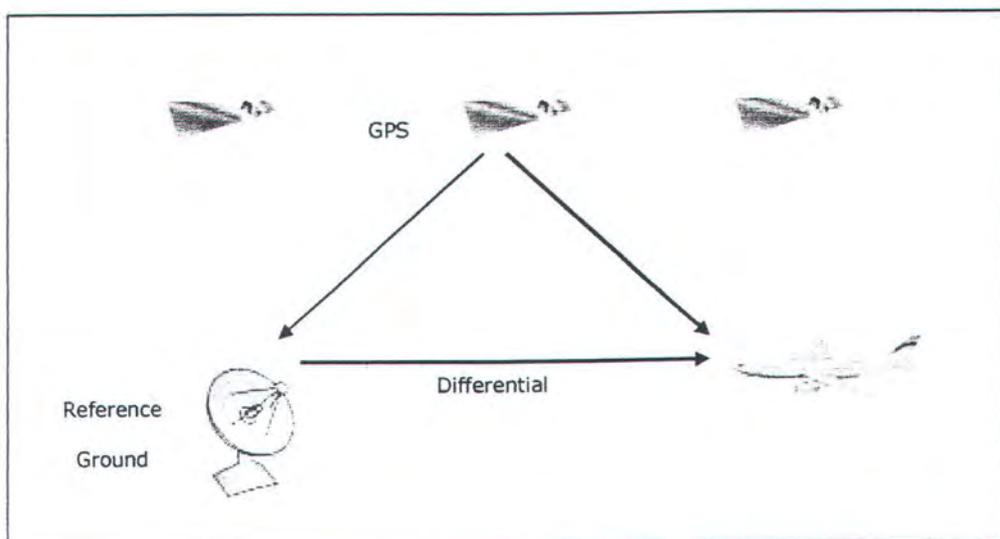
GPS bekerja dengan mengukur seberapa lama GPS menerima sinyal-sinyal radio untuk dicapai bumi dari satelit-satelit GPS tersebut. Sebuah receiver mengerjakan hal tersebut dengan men-*generate* sekumpulan kode-kode yang sama dengan yang sedang ditransmisikan oleh satelit-satelit sistem. *Receiver* kemudian mengkalkulasi waktu tunda antara kode-kodenya dengan kode yang diterima dari satelit GPS dengan menentukan seberapa jauh *receiver* harus merubah kode-kodenya sendiri untuk mencocokkan dengan yang ditransmisikan oleh satelit-satelit. Waktu perjalanan ini kemudian dikalikan dengan kecepatan cahaya untuk menentukan jarak *receiver* terhadap satelit-satelit.

Di dalam teori, sebuah GPS *receiver* dapat menghitung posisi 3 dimensi dengan mengukur jaraknya dari tiga satelit yang berbeda, tapi dalam prakteknya empat buah satelit diperlukan karena terdapat *timing offset* antara jam pada receiver dan jam pada sebuah satelit. Keempat pengukuran memungkinkan komputer *receiver* untuk memecahkan masalah *timing offset* dan menguranginya dari solusi navigasi.



Gambar 2.13 Penentuan koordinat oleh satelit

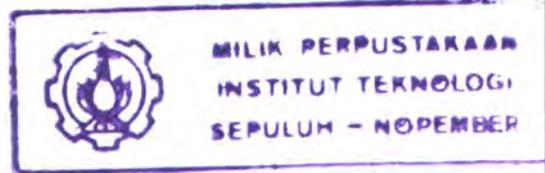
Untuk banyak aplikasi, akurasi yang lebih besar diperlukan daripada kemungkinan dengan hanya GPS saja (misalnya akurasi posisi dari sedikit meter ataupun sentimeter). *Differential* GPS (DGPS) adalah metode pengoperasian GPS yang memungkinkan pengguna untuk mendapatkan akurasi yang tinggi.



Gambar 2.14 Penentuan koordinat oleh satelit dengan cara lain

2.8.3. Komponen-komponen Sistem Pelacakan

Mengintegrasikan berbagai jenis komponen dari aplikasi pelacakan memerlukan pemahaman terhadap komponen-komponen seperti akuisisi data, komunikasi data, dan integrasi data. Bagian berikut akan mendefinisikan komponen solusi pelacakan yang kritis sesuai dengan contoh-contoh dari teknologi spesifik yang ditawarkan.



2.8.3.1. Remote Sensor dan Receiver

Kemajuan teknologi bidang *remote sensor* dan *GPS receiver* berkembang secara eksponensial dalam beberapa tahun ini. Sensor sekarang mampu memperoleh data pada banyak segi biometrik (suhu tubuh, detak jantung, tekanan darah), sumber daya alam (pertanian, perhutanan, geologi), perbedaan temporal (pertumbuhan penduduk, gunung es, ombak laut, pelayaran dan perikanan), bencana alam (banjir, kebakaran, luapan minyak), dan transportasi (jalan, sinyal lalu lintas, mesin, peralatan). Dari bermacam-macam sifat aplikasi ini sesuai dengan nuansa yang berkaitan, keadaan karakteristik sensor meliputi:

- *Modularitas dan interoperability*
- *Wireless transducers*
- *Onboard intelligence*
- Fleksibilitas melalui kemampuan untuk deprogram ulang
- *Transducer-independent architecture*
- Kemampuan untuk memerintah dari jauh

Kunci penggerak teknologi yang mendukung pencapaian karakteristik-karakteristik ini dan *dictate* kemampuan dari sensor generasi selanjutnya adalah:

- Sumber tenaga yang tahan lama, khususnya baterai
- Mikroprosesor yang hemat energi
- Sistem komunikasi yang canggih
- Mampu untuk digunakan di medan yang berat

Contoh-contoh dari sensor komersial yang tersedia termasuk pelacakan penangkapan binatang, gelang kaki untuk personel, sensor jalan, dan tanda pengenal *cargo*. Bagaimanapun, perlu dicatat bahwa kebanyakan sensor/ *receiver* ini memiliki *encoded* data string yang harus didekode dengan *base station receiver* yang sesuai.

2.8.3.2. Alternatif Media Komunikasi

Untuk kebanyakan aplikasi pelacakan, empat jenis media komunikasi dapat mendukung kebutuhan komunikasi antara sensor dan *base station*, yaitu *wireline (fixed-to-fixed)*, *wide area wireless (fixed-to-mobile)*, *dedicated short-range communications (fixed-to-mobile)*, dan *vehicle-to-vehicle (mobile-to-mobile)*. Tiap-tiap aplikasi dapat memiliki kebutuhan komunikasi yang berbeda yang cocok dengan kebutuhan khusus untuk frekuensi, jarak, ukuran data, dan banyak lagi. Sebagai contoh, beberapa metode komunikasi yang lebih populer dipresentasikan di bawah ini untuk area aplikasi berikut:

Tipe Aplikasi	Media Komunikasi
911	Proprietary Wireless
Automatic Vehicle Location	CDPD, Radio, Satellite
Remote Sensing	Wireline, RF
Animal Tracking	UHF
Maritime	INMARSAT

Tabel 2.1 Contoh metode komunikasi

A. Wireline

Terdapat beberapa pilihan teknologi wireline untuk kebutuhan komunikasi *fixed-to-fixed*. Contohnya, aplikasi pengaturan lalu lintas dapat digunakan dengan menyewa atau membeli *twisted wire pairs*, *coaxial cable*, atau *fiber optics* untuk mendapatkan informasi dan untuk memonitor sensor pengawas kontrol lalu lintas, sinyal lalu lintas, tanda-anda pesan yang berubah-ubah, dan lain-lain. Dalam aplikasi lainnya, akan lebih menguntungkan untuk menggunakan *terrestrial microwave links*, *spread spectrum radio*, atau *area radio network* untuk menyediakan komunikasi antara pusat manajemen dan *remote controllers*. Meskipun ini merupakan teknologi komunikasi *wireless*, namun digunakan untuk menyediakan komunikasi *fixed-to-fixed* pada contoh tersebut; *communications in the examples cited*; sebagai akibatnya sering disebut sebagai media komunikasi *wireline*.

Pilihan jaringan Wireline termasuk *private networks*, *public shared networks*, atau gabungan dari keduanya. Teknologi Private network meliputi Ethernet, *Fiber Distributed Data Interface* (FDDI), *Synchronous Optical Network* (SONET), dan *Asynchronous Transfer Mode* (ATM). Teknologi Public shared network meliputi *leased analog lines*, *leased digital lines*, *frame relay*, *Integrated Services Digital Network* (ISDN), *metropolitan Ethernet*, *Internet*, dan *Switched Multimegabit Data Service* (SMDS).

B. Wireless

1. Wide Area Wireless

Terdapat dua kategori komunikasi *wireless* berdasarkan *range* dan area dari *coverage-wide area* dan *short range*. Komunikasi *Wide area wireless* (*fixed-to-mobile*) cocok untuk servis dan aplikasi dimana informasi disebarkan ke pengguna yang tidak dilokasikan dekat dengan sumber transmisi dan yang memerlukan *seamless coverage*. Komunikasi *Wide area wireless* lebih jauh dibedakan pada apakah searah atau dua arah. Sebagai contoh dari transmisi broadcast searah adalah laporan lalu lintas yang diterima sinyal radio AM atau FM. Sebuah *mobile traveler* yang memesan dan menerima informasi lalu lintas terkini dari sebuah *information service provider* adalah sebuah contoh dari komunikasi dua arah.

Teknologi *Two-way wide area wireless* meliputi *Global System for Mobile Communications* (GSM), *Special Mobile Radio* (SMR), *Enhanced Special Mobile Radio* (ESMR), *Personal Communications System* (PCS), ARDIS, RAM, *Geotek*,

220 MHz, *Metricom*, *Tetherless Access Ltd.* (TAL), *two-way paging*, dan *Cellular Digital Packet Data* (CDPD). CDPD, merupakan salah satu teknologi yang populer, yang memungkinkan transmisi data TCP/IP sepanjang sistem selular analog dan memberi keuntungan dari spektrum selular yang tak berguna.

Teknologi ***One-way broadcast communication*** meliputi AM *subcarrier*, FM *subcarrier*, dan *Highway Advisory Radio* (HAR). Sistem FM *subcarrier* meliputi *Subcarrier Traffic Information Channel* (STIC), *Data Radio Channel* (DARC), *High Speed FM Subcarrier Data System* (HSDS), RBDS, ALERT, dan SCA. Persoalan terkait meliputi terbatasnya coverage dari sistem yang tersedia saat ini, perangkat keras yang dimiliki dan antarmuka dari *High Data Rate Systems* (HSDS, STIC, dan DARC), dan *high data rate capacity limitations*.

2. Short-Range Wireless

Komunikasi *Short-range wireless* berkaitan dengan transfer informasi dari *localized interest*. Terdapat dua jenis komunikasi short-range wireless, ***vehicle-to-vehicle*** dan ***Dedicated Short-range Communications (DSRC)***. Komunikasi *Vehicle-to-vehicle (mobile-to-mobile) short-range wireless* diperlukan untuk mendukung Automated Highway System (AHS) dan kemungkinan besar implementasi untuk menghindari terjadinya koalisi pada persimpangan. Aplikasi yang sesuai untuk DSRC (fixed-to-mobile) meliputi *toll collection*, *parking fee collection*, *roadside safety inspections*, *credential checks*, *invehicle signing*, *intersection collision avoidance*, dan komunikasi AHS terpilih (contohnya: *safety checks*, *access authorization*, dan *system status updates*). Teknologi aplikatif

lainnya meliputi *radio frequency* (RF) dan *Infrared* (IR) *short-range wireless beacon tag communications* untuk kebutuhan DSRC.

2.8.3.3. Sistem Satelit

Komunikasi Satelit adalah garis pandang dari angkasa, dan menyediakan perlindungan yang baik untuk di luar ruangan. Tetapi, komunikasi satelit menyediakan layanan yang kecil atau bahkan tidak ada sama sekali untuk lokasi di terowongan atau di hutan yang lebat.

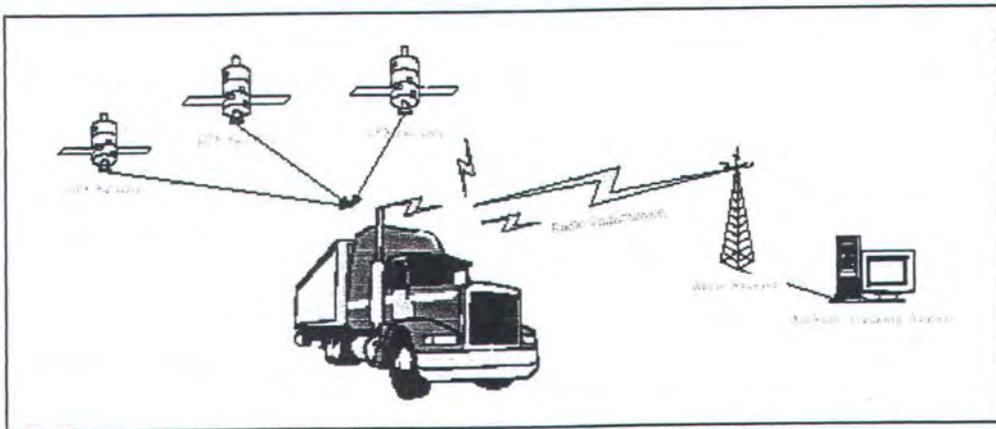
Ada suatu kumpulan komunikasi satelit yang sesuai bagi aplikasi untuk mengikuti jalan. Kumpulan ini termasuk variasi dari *Little* (data saja) dan *Big* (suara dan data) "*low-earth-orbit*" (LEO) *system*, sama seperti sistem *medium-earth-orbit* (MEO) dan *geosynchronous orbit* (GEO). Beberapa dari sistem ini belum disebar. Bagaimanapun sistem-sistem tersebut direncanakan untuk digunakan dalam beberapa tahun mendatang. Sistem-sistem tersebut termasuk ORBCOMM, STARSYS, VITASAT, MSAT, Constellation, GLOBALSTAR, IRIDIUM, TELEDESIC, Ellipso, Odyssey, Skycell, VSAT, dan OmniTRACS.

2.8.4. Automatic Vehicle Location

Sistem *Automatic Vehicle Location* (AVL) yang menggunakan GPS, menunjukkan lokasi dari kendaraan dalam sebuah jaringan lalu lintas atau armada. Aplikasi dengan menggunakan AVL ini dapat digunakan untuk pelacakan mobil atau truk, unit kargo, kapal laut, atau pesawat udara yang bergerak pada area yang ditentukan. Dalam penggunaan AVL, semua kendaraan dalam armada harus dilengkapi dengan *mobile radio transceiver* atau *modem*, *GPS receiver*, dan

antena GPS. Sedangkan di kantor pusat harus ada komputer yang dilengkapi dengan antenna penerima dan juga peta digital sehingga dapat menunjukkan lokasi dari pergerakan kendaraan yang dilacak.

Beberapa organisasi mengimplementasikan sistem AVL untuk memperoleh manfaat seperti kecepatan waktu respon, peningkatan keamanan personel, dan efektifitas dari sumber daya yang dapat menghemat waktu dan biaya. Saat ini AVL digunakan di berbagai bidang termasuk manajemen armada, panggilan darurat, keamanan kendaraan, penunjuk arah, sistem navigasi.



Gambar 2.15 Manajemen armada menggunakan AVL

Sebagai contoh adalah dalam manajemen armada. Dengan teknologi AVL, perusahaan dapat mengetahui dimana keberadaan truk mereka setiap waktu; berkomunikasi dengan sopirnya, atau untuk mengetahui kondisi mesin, catu daya, ataupun suhu mesin. Selain menunjukkan informasi lokasi, sensor dapat juga menunjukkan kualitas dari sopir seperti mengemudi terlalu cepat, terlalu lambat, terlalu banyak berhenti atau kurang berhenti. Dengan sistem AVL ini dapat membantu perusahaan dalam meningkatkan waktu respon kepada *customer*,

asistensi navigasi, penanganan pencurian kendaraan, dan lain sebagainya. Aplikasi manajemen armada ini membutuhkan unit remote yang diletakkan pada kendaraan dan *software* monitoring pada kantor pusat. *Remote* unit yang dimaksud adalah prosesor *onboard*, *GPS receiver* dan *radio modem* yang dapat mengumpulkan informasi secara terus menerus dan kemudian menyimpan data tentang kendaraan seperti posisi (koordinat *latitude/longitude*), kecepatan, dan arah.

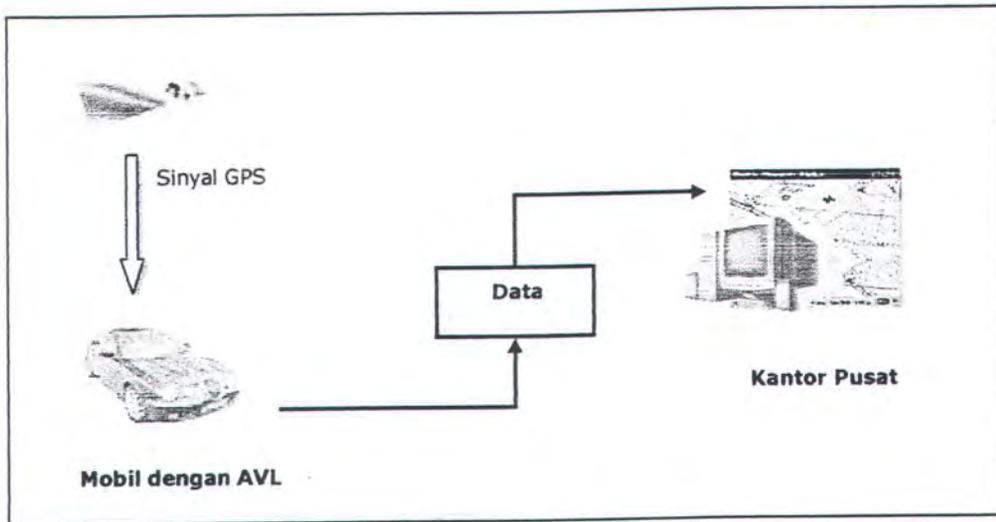
Secara garis besar jenis AVL dapat diklasifikasikan menjadi 2 macam yaitu:

1. Sistem AVL secara *Off-Line*
2. Sistem AVL secara *On-Line*

Berikut akan dijelaskan lebih detail tentang kedua jenis AVL tersebut.

2.8.4.1. Sistem AVL secara *Off-Line*

Pada jenis sistem AVL ini, proses penelusuran (*tracking*) dilakukan secara *off-line*. Hasil penelusuran data dapat disimpan di dalam unit AVL dan dapat di-*download* kembali melalui RS232 dengan menghubungkan unit ke sebuah PC. Pada sistem ini lebih sederhana sehingga dapat lebih menghemat biaya misalnya untuk gaji bulanan operator jaringan dan biaya komunikasi. Jika posisi geografis dari tiap pemberhentian kendaraan tidak diperlukan, maka sistem dapat dijalankan tanpa *GPS receiver*. Keseluruhan jarak tempuh dan waktu dihitung dengan antarmuka odometer dan jam internal yang bersifat *real-time*. Cara kerja Sistem AVL secara *off-line* ditunjukkan pada gambar 2.16.



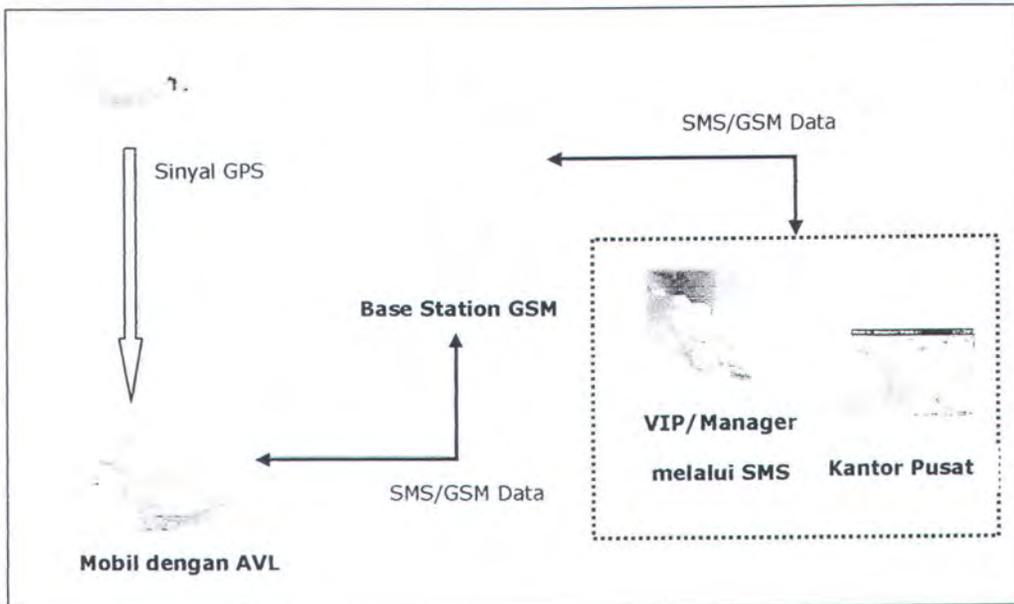
Gambar 2.16. Cara Kerja Sistem AVL secara *Off-Line*

2.8.4.2. Sistem AVL secara *On-Line*

Sedangkan pada jenis sistem AVL ini, proses penelusuran (*tracking*) dilakukan secara *on-line*, yakni operator pusat dapat mengetahui informasi seperti posisi, kecepatan, jarak yang telah ditempuh, dan arah kendaraan sewaktu-waktu. Pada sistem ini tiap kendaraan dilengkapi dengan unit AVL yang dapat menerima sinyal dari satelit GPS. Kemudian GPS *receiver* menentukan lokasi, kecepatan dan tujuan dari kendaraan pada saat ini. Data ini dapat disimpan atau dapat ditransmisikan secara langsung ke *operating centre*. Cara kerja Sistem AVL secara *on-line* ditunjukkan pada gambar 2.17.

Keuntungan utama dari sistem AVL *On-Line* adalah dapat menelusuri posisi secara *real-time* karena pengiriman informasi data posisi melalui SMS dari AVL *On-Line* dapat diatur berdasarkan tiap satuan waktu atau jarak. Kantor pusat juga dapat meminta informasi tersebut sewaktu-waktu, yakni melalui SMS yang dikirimkan dari nomor kartu telepon selular pada modem di kantor pusat ke

nomor kartu telepon selular pada AVL, baru kemudian AVL tersebut mengirimkan informasi data posisi melalui SMS ke kantor pusat.



Gambar 2.17 Cara Kerja Sistem AVL *On-Line*

Dengan pemanfaatan sistem AVL *On-Line* akan memungkinkan seorang manajer mengetahui posisi unit mobil perusahaannya dengan cara mengirimkan SMS ke kantor pusat. Selanjutnya kantor pusat menelusuri posisinya terlebih dahulu, baru kemudian mengirimkan SMS yang berisi informasi posisi unit mobil perusahaan ke telepon selular manajer.

2.8.4.3. Konversi Koordinat

Data-data koordinat yang dikirimkan oleh AVL dalam bentuk sistem koordinat geografis yang berupa *longitude* dan *latitude*. Sedangkan peta menggunakan proyeksi UTM. Karakteristik dari kedua sistem koordinat tersebut mempunyai banyak perbedaan. Sistem koordinat geografis merepresentasikan bumi ke dalam bentuk dua dimensi, lalu membagi bumi dengan garis-garis bujur

dan lintang untuk merepresentasikan setiap titik dari permukaan bumi. Satuan yang digunakan adalah derajat (*degree*) dengan urutan derajat, menit, detik. *Equator* merupakan garis lintang nol derajat dan bujur nol derajat yang melewati kota Greenwich. Kelebihan sistem ini adalah dapat merepresentasikan setiap titik di permukaan bumi secara tepat, namun tidak bisa merepresentasikan jarak sebenarnya dari dua titik.

Sedangkan pada proyeksi UTM mempunyai karakteristik-karakteristik sebagai berikut :

1. Seluruh wilayah permukaan bumi (ellipsoida) dibagi dalam 60 wilayah (zone) yang masing-masingnya dibatasi oleh 2 meridian selebar 6° .
2. Masing – masing zone mempunyai sistem koordinat tersendiri yaitu titik potong meridian sentralnya dengan garis equator. Titik nol ini disebut dengan titik nol sejati (*true origin*)
3. Zone-zone proyeksi UTM diberi nomor mulai zone 1 dengan 180° BB sampai 174° BB. Terus ke timur sampai zone 60 antara 174° BT sampai dengan 180° BT.
4. Batas-batas lintang dalam sistem UTM adalah 80° lintang selatan dan 84° lintang utara, dimana untuk daerah ini dibagi dalam jalur-jalur selebar 8° lintang. Sedangkan pembagian dimulai dari 80° lintang selatan ke arah utara.
5. Jalur-jalur dengan lebar 8° lintang dan 6° bujur diberi tanda dengan huruf-huruf. Dimulai dengan huruf C untuk jalur antara 80° dan 72° lintang selatan, berturut-turut sampai dengan huruf X, dengan catatan bahwa huruf I dan D tidak digunakan.
6. Angka perbesaran sebesar 0,9996 diterapkan sepanjang meridian tengah (sentral) = $k_0 = 0,9996$

7. Dalam grid metrik angka meridian sentral diberi koordinat fiktif sebesar 500.000 meter. Untuk menghindari angka negatif, bagi koordinat-koordinat titik – titik di sebelah selatan equator, maka equator diberi koordinat 10.000.000 meter, sedangkan bagi koordinat – koordinat titik – titik utara equator, maka equator diberi koordinat 0.

Dengan demikian perlu adanya konversi data-data koordinat dari AVL yang berupa koordinat geografis menjadi UTM.

Dalam penerapan sistem UTM bagi peta dasar nasional, seluruh wilayah Indonesia terbagi atas 9 wilayah (zone), yang masing-masing mempunyai lebar 6° bujur, mulai dari meridian 90° bujur timur sampai dengan 144° bujur timur dengan batas sampai paralel 10° lintang utara dan 15° lintang selatan dengan 4 satuan daerah yaitu L, M, N, dan P

Langkah pertama yang dilakukan adalah mengkonversi terlebih dahulu data-data longitude dan latitude ke dalam format derajat, menit, detik karena pada AVL masih berbentuk desimal. Kemudian dapat digunakan rumus yang sederhana (metode iterasi juga ada) sebagai berikut:

$$X = [N + H] \cos \phi \cdot \cos \lambda \quad (2)$$

$$Y = [N + H] \cos \phi \cdot \sin \lambda \quad (3)$$

Dimana:

ϕ = latitude

λ = longitude

h = ketinggian di atas bidang elips (meter)

N = Prime Vertical of Curvature (meters):

a 2

$$N = \frac{a^2}{[a^2 \cos^2 \phi + b^2 \sin^2 \phi]^{1/2}} \quad (4)$$

$$[a^2 \cos^2 \phi + b^2 \sin^2 \phi]^{1/2}$$

a = 6378137 (WGS84), sumbu semi-mayor

b = 6356752,3142 (WGS84), sumbu semi-minor

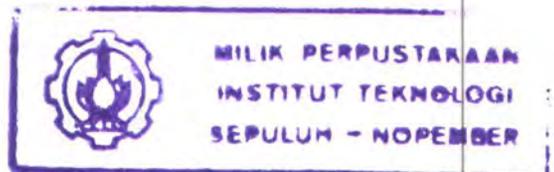
Sedangkan perhitungan yang menggunakan metode iterasi menggunakan rumus berikut :

Untuk transformasi ke koordinat UTM, harga perbesaran pada meridian tengah ditetapkan sebesar $k_0 = 0,9996$

$$X_{UTM} = k_0 \cdot X_{TM}$$

$$Y_{UTM} = k_0 \cdot Y_{TM}$$

lengkapnya :



$$X_{UTM} = \left[k_0 \cdot \frac{N}{\rho} \cos \phi \right] \Delta\lambda + \left[\frac{k_0 \cdot N}{6 \rho^3} \cos^3 \phi (1 - t^2 + \eta^2) \right] \Delta\lambda^3 +$$

$$\left[\frac{k_0 \cdot N}{120 \rho^5} \cos^5 \phi (1 - 18t^2 + t^4 + 14\eta^2 - 58t^2\eta^2 - 64\eta^4 t^2 + 4\eta^6 - 24\eta^6 t^2) \right] \Delta\lambda^5 +$$

$$\frac{k_0 \cdot N}{5040 \rho^7} \cos^7 \phi (61 - 479 t^2 + 179 t^4 - t^6 + 331 \eta^2 - 3298 \eta^2 t^2 +$$

$$1771 \eta^2 t^4 + 715 \eta^4 - \dots \dots \dots) \Delta\lambda^7 \quad (5)$$

$$Y_{UTM} =$$

$$\left[k_0 G \right] + \left[\frac{k_0 N}{2 \rho^2} \sin \phi \cos \phi \right] \Delta\lambda^2 + \left[\frac{k_0 N}{24 \rho^4} \sin \phi \cos^3 \phi (5 - t^2 + 9\eta^2 + 4\eta^4) \right] \Delta\lambda^4$$

$$\left[\frac{k_0 N}{-720 \rho^6} \sin \phi \cos^5 \phi (61 - 58t^2 + t^4 + 270\eta^2 - 330t^2\eta^2 + 445\eta^4 -$$

$$680\eta^4 t^2 + 324\eta^6 - 600\eta^6 t^2 + \dots \dots \dots) \Delta\lambda^6 \right] \quad (6)$$

Antara meridian tengah (500.000 mE) dengan garis grid 320 mE(sebelah meridian tengah) dan 680 mE(timur meridian tengah) terjadi reduksi skala. Faktor

skala pada daerah ini mempunyai harga dari 0,99960 sampai 1,00000. Di luar batas grid 320.000mE dan 680.000 mE dan 680.000 mE faktor skala lebih besar daripada 1,00000. Berarti jarak-jarak pada peta tergambar lebih besar daripada permukaan bumi . Untuk daerah dekat dengan tepi zone, sekitar 300.000 m sebelah barat atau timur meridian tengah untuk jarak 1000 meter akan tergambar $1,00070 \times 100 \text{ m} = 1000,70 \text{ meter}$. Berarti mengalami perbesaran jarak 70 cm per 1000 meter.

BAB III



SISTEM PELACAKAN PADA PENGIRIMAN KOTAK UANG

BAB III

SISTEM PELACAKAN PADA PENGIRIMAN KOTAK UANG

3.1. Konsep Umum Pelacakan Posisi Pengiriman Kotak Uang

Informasi mengenai posisi kotak uang pada waktu pengiriman sangat diperlukan. Oleh karena itu pihak bank berkepentingan untuk memonitor posisi-posisi kotak uang yang sedang dikirimkan. Hal ini disebabkan besarnya resiko dalam sebuah pengiriman. Dengan selalu memonitor posisi-posisi terakhir maka pihak pengirim akan lebih mudah dalam mengatasi kendala yang mungkin terjadi selama perjalanan dan dapat membantu pihak yang berwajib bila terjadi tindak kriminal pada proses perjalanan.

Selama ini pelacakan dilakukan melalui komunikasi dengan menggunakan radio komunikasi. Pada proses pemberangkatan, kurir dilengkapi dengan radio komunikasi dan surat-surat berita acara proses pengiriman.

Dalam menangani kendala yang mungkin terjadi dalam sebuah pengiriman misalnya terjadi perampokan maka kurir harus melaporkan ke pihak pengirim. Selama ini tidak ada langkah pelacakan selanjutnya, hanya dilihat apakah penyebab dari kesalahan tersebut. Apabila kesalahan manusia dalam hal ini kurir, maka dia akan dikenakan sanksi pidana. Mengenai nasib dari uang yang dikirimkan, kecil kemungkinannya bisa diperoleh kembali. Dengan demikian jelas bahwa kekurangan dari metode yang digunakan selama ini adalah:

1. Tidak mampu mengetahui posisi kotak uang secara pasti di bumi. Posisi tersebut hanya berdasarkan informasi yang disampaikan oleh kurir. Kebenaran dari informasi tersebut masih diragukan.
2. Tidak mampu mendukung perekaman perjalanan pengiriman kotak uang, yakni posisi-posisi yang pernah ditempuh selama perjalanan pengiriman. Rekaman perjalanan dapat dipakai untuk analisis dan evaluasi lebih lanjut untuk pelaksanaan selanjutnya.
3. Tidak dapat mengetahui kecepatan kendaraan pengangkut sehingga sulit dalam mengontrol kecepatannya.

3.2. Konsep Umum Pelacakan Kotak Uang Berbasis GIS menggunakan AVL(Automatic Vehicle Location)

Informasi penting yang dibutuhkan dalam pelacakan posisi kotak uang untuk menangani kendala-kendala yang mungkin terjadi pada perjalanan kotak uang, antara lain adalah posisi terkini kotak uang serta kecepatan yang sedang ditempuh oleh kotak uang. Beberapa permasalahan yang telah dijelaskan sebelumnya akan dapat ditangani dengan lebih efisien apabila dapat diketahui secara cepat dimana posisi dan kecepatan kotak uang terkini.

Pada sistem pelacakan kotak uang berbasis GIS ini terdapat peta digital yang melingkupi wilayah pengiriman yang dilaksanakan. Di situ terdapat data-data lengkap mengenai daerah pengiriman seperti data jalan, data bangunan, data area, dan lain-lain. Di samping itu digunakan juga alat AVL yang dapat memberikan data mengenai lokasi titik terkini pada proses pengiriman disamping data kecepatan beserta status dari kendaraan yang digunakan dalam sebuah pengiriman.

Dengan memanfaatkan peta digital dan AVL maka pihak pengirim dapat dengan mudah memantau perjalanan. Peta disajikan dalam bentuk web yang nantinya untuk mempermudah pengaksesan program pada klien. Data-data yang dikirimkan oleh AVL senantiasa diupdate ke dalam database dan disajikan secara realtime ke dalam peta digital. Hal ini juga akan membantu dalam masalah pengarsipan karena disimpan secara rapi dalam sebuah database. Bilamana data perjalanan yang lampau ingin dilihat kembali maka dapat dengan mudah dilakukan. Dari perjalanan yang telah dilakukan bisa senantiasa dievaluasi untuk perbaikan pada proses selanjutnya.

Dalam sebuah perjalanan memang tidak semuanya berjalan dengan lancar. Adakalanya terdapat kendala-kendala. Dengan system pelacakan berbasis GIS menggunakan AVL ini dapat mengatasi beberapa kendala yang muncul.

Pada status alat dapat diketahui bilamana terjadi gangguan selama perjalanan. Informasi ini senantiasa dikirimkan ke kantor pusat. Bila status = 0 maka kondisi aman, bila status berubah maka ada gangguan yang terjadi semisal perampokan. Bilamana terjadi perampokan tentunya mobil dibuka dengan paksa sehingga sensor alat tersebut segera mengidentifikasi adanya gangguan. Atau bisa juga dari kurir menekan tombol darurat. Dengan adanya status ini maka tindakan pencegahan atau penanganan kasus bisa segera dilakukan dengan cepat.

Untuk sensor kecepatan, data kecepatan dari mobil akan selalu dipantau. Apakah kecepatannya melebihi ambang batas yang telah ditentukan. Apabila melebihi batas maka pihak pusat sebagai pemantau bisa langsung memperingatkan driver untuk mengurangi kecepatan, atau bisa juga sebaliknya.

Perlu diwaspadai juga bahwa kendaraan tidak boleh berhenti terlalu lama di sebuah titik tanpa alasan yang jelas, misal ketika ada lampu merah. Kendaraan tidak diperkenankan untuk berhenti terlalu lama di lampu merah. Lampu merah biasanya berkisar antara 1 menit sampai 3 menit. Bila kendaraan berhenti lebih dari 5 menit pada titik yang sama maka kurir harus segera diberi peringatan dan menanyakan kendala apa yang terjadi. Bila memang terjadi kemacetan maka nantinya pihak pusat bisa memberikan pertimbangan untuk menempuh jalur alternatif.

Dalam setiap pengiriman terutama untuk pengiriman rutin biasanya sudah ditentukan jalur yang harus dilalui. Akan tetapi memang tidak dapat dipungkiri akan adanya kendala yang mengharuskan kendaraan menempuh jalur lain. Dengan mengecek posisi dari kendaraan maka dapat diketahui bilamana jalur yang dituju telah melenceng dari jalur yang telah ditetapkan. Bila hal ini terjadi pihak pusat dapat segera memberikan peringatan.

3.3. Perbandingan Metode Pelacakan Posisi

Selain metode pelacakan posisi kotak uang yang selama ini diterapkan, sebenarnya masih terdapat metode pelacakan lainnya. Metode yang lainnya antara lain dengan menggunakan sinyal radio serta dengan menggunakan satelit, yakni GPS (*Global Positioning System*).

Metode pelacakan yang pertama adalah dengan cara mengukur jarak dengan menggunakan sinyal radio. Konsepnya adalah dengan mengukur waktu yang diperlukan untuk sinyal khusus dari radio di stasiun pemancar ke alat yang dirancang khusus untuk menerima sinyal tersebut. Kemudian mengalikan waktu

perjalanan sinyal dengan kecepatan sinyal (300,000,000 meter/detik) adalah sama dengan jarak dari pemancar ke penerima.

Ada juga beberapa alat dan metode navigasi yang mengandalkan stasiun bumi untuk menentukan posisi. Cara yang paling mudah dan sering digunakan sampai sekarang untuk mengetahui posisi sebuah radio pemancar, maka radio pemancar tersebut akan memancarkan sinyal (transmit) dan meminta bantuan minimal 3 buah radio penerima (receiver) untuk mengarahkan antenanya ke arah sinyal yang datang. Kemudian arah antena penerima digambar pada peta sebagai garis lurus dan titik perpotongan tiga garis tersebut adalah posisi radio pemancar.

Pengukuran yang akurat dari waktu perjalanan sinyal sangat penting jika sepersejuta detik kesalahan dalam pengukuran waktu perjalanan sama dengan 300 meter kesalahan dalam jarak.

Untuk mengatasi keterbatasan metode sebelumnya, maka para ahli beralih ke satelit yang didasarkan pada sistem radio navigasi yang sudah ditingkatkan kemampuannya dan diletakkan pada orbit yang tinggi untuk dapat menjangkau wilayah yang lebih luas.

Kelebihan metode dengan menggunakan satelit yakni dapat digunakan setiap saat, tanpa bergantung pada waktu dan cuaca serta mampu mendapatkan koordinat posisi di bumi. Karena karakteristik ini maka penggunaan satelit dapat meningkatkan efisiensi dan fleksibilitas dari pelaksanaan survei dan pemetaan, sehingga lebih cepat dan hemat biaya operasional.

Untuk mendapatkan keakuratan posisi yang paling tinggi maka diperlukan metode pelacakan yang paling baik agar berguna dalam analisis penanganan

terhadap kendala dan hambatan yang mungkin terjadi dalam operasional perjalanan kotak uang.

Tabel 3.1. berikut ini adalah perbandingan kemampuan dan kekurangan dari metode pelacakan dengan menggunakan telepon, sinyal radio, dan GPS (*Global Positioning System*).

Metode Pelacakan	Telepon	Sinyal Radio	GPS
Kemampuan			
- Keakuratan dalam menentukan posisi kotak uang di atas bumi	Kurang	Sedang	Baik
- Mendukung perekaman posisi-posisi yang pernah dilintasi dalam perjalanan kotak uang	Kurang	Kurang	Baik
- Mendapatkan informasi kecepatan kotak uang	Kurang	Kurang	Baik

Tabel 3.1 Perbandingan Kemampuan dan Kekurangan Metode Pelacakan

Selain daripada kemampuan-kemampuan yang telah disebutkan, AVL (*Automatic Vehicle Location*) memiliki kelemahan yakni penggunaan AVL harus dilakukan di ruang terbuka yang terbebas dari halangan, karena antena GPS yang dimiliki AVL harus dapat selalu menangkap sinyal GPS untuk mengetahui koordinat di bumi. Oleh karena itu, antena GPS tersebut harus diletakkan di ruang terbuka.

3.4. Konsep Teknis Pelacakan Kotak Uang Berbasis GIS menggunakan AVL (Automatic Vehicle Location)

Program pelacakan kotak uang berbasis GIS yang akan dibuat ini menggunakan AVL (*Automatic Vehicle Location*) *On-Line* yang berbasis pada GPS (*Global Positioning System*) dan GSM (*Global System of Mobile Communication*). AVL dipasang di setiap kendaraan sehingga informasi seperti posisi, kecepatan, dan arah kotak uang dapat diketahui secara *real time*. Informasi-informasi tersebut kemudian dikirim melalui SMS (*Short Message Service*) dengan menggunakan teknologi GSM. Apabila sinyal GSM mengalami hambatan semisal berada di luar *service area* sehingga tidak bisa ditangkap, maka pada saat itu data-data dari AVL disimpan. Data-data tersebut dapat di-*download* dengan menggunakan interface RS232. Kantor pusat dapat melihat posisi kotak uang pada sebuah peta digital. Informasi dari SMS yang diterima oleh perangkat lunak AVL, selanjutnya diparser untuk disimpan ke database yang kemudian akan ditampilkan di peta digital.

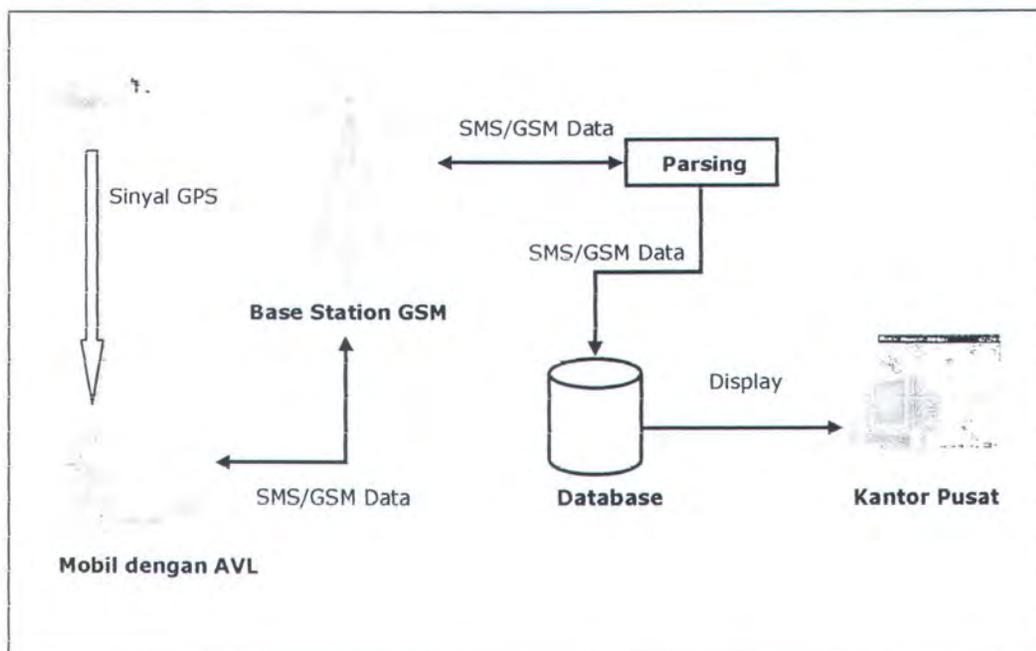
Dalam pemanfaatan alat AVL secara *on-line*, tiap kendaraan dilengkapi dengan komponen sebuah *radio transceiver*, GPS *receiver*, dan sebuah antena GPS, sedangkan Kantor Pusat harus memiliki sebuah komputer yang dilengkapi dengan receiver SMS.

Gambar 3.1 AVL dan antena

SMS selanjutnya diterima oleh perangkat lunak AVL kemudian diparser untuk disimpan ke database, yakni data-data koordinat x (longitude), koordinat y (latitude), kecepatan, arah, dan waktu saat itu. Setelah tersimpan didalam database, maka data-data posisi kotak uang baru dapat ditampilkan di peta yang akan di-*autorefresh* setiap 2 menit. Mengenai berapa lama waktu setiap kali refresh bisa diset pada AVLnya. AVL yang terpasang pada kendaraan dilengkapi dengan nomor ponsel, begitu pula pada modem penerima SMS yang terdapat di kantor pusat sehingga nomor ponsel AVL yang terpasang pada kendaraan dapat mengirim SMS ke nomor ponsel modem di kantor pusat. Operator di kantor pusat dapat melihat lokasi setiap kotak uang yang sedang melakukan perjalanan serta pergerakannya pada sebuah peta terdigitasi, sehingga dapat dimonitor dan dianalisis apabila terdapat hambatan teknis pada perjalanan kotak uang.

Dalam Tugas Akhir ini, digunakan data-data posisi yang diperoleh dari AVL *off-line* serta data simulasi. Data masukan yang berupa SMS, disimpan didalam database bawaan perangkat lunak AVL. Kemudian terdapat *tool* untuk proses *update*-an di server yakni melakukan penyimpanan data-data dari AVL ke tabel rekaman perjalanan dan *update* tabel posisi kotak uang terkini. Selanjutnya

data-data dari tabel posisi kotak uang terkini ditampilkan ke peta dengan menggunakan ASPMap. Untuk proses analisis dilakukan di server dan disesuaikan dengan kebutuhan *client*.



Gambar 3.2 Sistem kerja dari AVL

3.5. Analisa Kebutuhan Sistem Pelacakan

Dari proses pelacakan pada pengiriman uang yang dilakukan selama ini masih mengandung kelemahan-kelemahan seperti disebutkan pada sub bab 3.1 diatas. Di satu sisi teknologi pelacakan telah berkembang sedemikian pesatnya. Dengan semakin dibutuhkannya tingkat keamanan yang lebih dan administrasi yang lebih baik maka sistem pelacakan harus bisa mengatasi berbagai kelemahan dengan didukung teknologi terkini. Dari hasil wawancara dengan Bapak Fajar Effendy, selaku Asisten Manajer Bank Indonesia wilayah Kediri pada tanggal 3

Mei 2003 dan tanggal 24 Mei 2003, bisa disimpulkan bahwa sistem pelacakan yang dibutuhkan melingkupi kriteria berikut :

1. Memiliki Perekaman Data Perjalanan

Salah satu kelemahan yang ada selama ini adalah tidak adanya perekaman terhadap data perjalanan. Kalaupun ada, hanyalah bersifat sederhana dan manual. Ataupun hanya berupa data yang menunjukkan bahwa pada saat tertentu terdapat proses pengiriman uang.

Dengan tidak adanya rekaman data perjalanan, tidak dapat diketahui apakah proses perjalanan berlangsung baik ataukah sebaliknya. Secara logika, bila sebuah proses sudah berjalan dengan baik maka tinggal diteruskan dan dipertahankan atau lebih ditingkatkan lagi. Tetapi bila proses berjalan buruk maka dilakukan evaluasi untuk memperbaiki proses selanjutnya. Sebuah evaluasi bisa dilakukan bila didukung dengan data-data yang real.

Untuk itu diperlukan data rekaman perjalanan yang melingkupi :

- a. Data Transaksi : data yang menyimpan proses pengiriman meliputi waktu dan tempat baik asal maupun tujuan pengiriman. Selain itu juga atribut-atribut yang melengkapi semisal data kendaraan yang digunakan, pegawai yang bertanggung jawab dan juga berapa jumlah uang yang dikirim.
- b. Data Posisi : merupakan data tentang posisi selama perjalanan pada rentang waktu tertentu

2. Dapat Mengetahui Posisi Terkini Perjalanan

Kelemahan yang lain pada proses pelacakan selama ini adalah tidak diketahuinya posisi terkini secara akurat. Karena menggunakan komunikasi radio atau Handy Talkie maka informasi posisi sangat tergantung pada orang yang melakukan perjalanan. Informasinya pun juga sangat terbatas. Selain itu, tidak dapat mencegah bilamana petugas pengiriman melaporkan posisi palsu. Maka diharapkan dapat diperoleh informasi yang akurat yang sesuai dengan aturan yang baku. Salah satunya dengan menggunakan sistem posisi koordinat bumi berupa garis bujur dan garis lintang yang berlaku secara internasional.

3. Bersifat Online-Real Time

Semakin cepat data terkini bisa diperoleh maka akan menghemat biaya dan waktu serta meningkatkan keamanan selama proses perjalanan. Waktu yang dibutuhkan untuk mengambil tindakan bilamana terjadi hal-hal yang tidak diinginkan juga bisa cepat.

Bersifat online berarti bahwa sistem dapat diakses dari berbagai tempat. Sebagai contoh, data di kantor pusat bisa diakses dari kantor cabang. Tentunya dengan hak akses tertentu dan user yang tidak berhak tidak dapat mengaksesnya.

Bersifat real time berarti bahwa data yang didapat adalah data terkini. Jeda waktu kejadian dan perekaman data sangatlah kecil.

4. Analisa Data Perjalanan

Data-data selama perjalanan, selain direkam juga diolah sebagai bahan prediksi terhadap hal-hal yang membahayakan. Setiap kali data dikirim maka setiap itu juga dianalisa. Beberapa komponen data yang menjadi bahan analisa adalah :

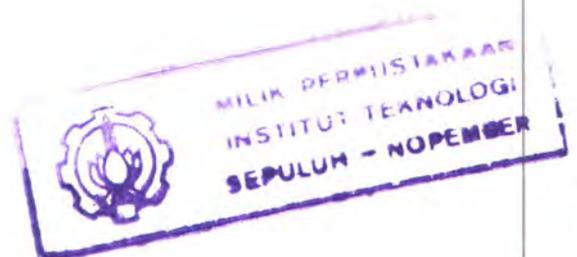
a. Data Status :

Data status menunjukkan status dari AVL. Secara normal dia akan mengirimkan sinyal bahwa status=0. Hal ini berarti tidak ada gangguan pada AVL tersebut.

Pemasangan AVL pada kendaraan tidak hanya berdiri sendiri tetapi AVL mempunyai input output yang bisa disambungkan ke sistem kendaraan. Bisa disambungkan ke sistem penguncian pada pintu dan juga pada sistem mesin. Adanya gangguan menunjukkan, ada perlakuan tidak wajar terhadap AVL itu sendiri dan juga terhadap kendaraan yang dipakai.

Perlakuan tidak wajar pada AVL semisal AVL dipindah paksa dari tempat pemasangan atau ada kerusakan komponen. Perlakuan tidak wajar pada kendaraan semisal pintu kendaraan dibuka paksa atau sedang mengalami kecelakaan. Kesimpulannya, bila status AVL berubah maka perlu mendapatkan perhatian yang serius mengenai penyebab perubahan status tersebut.

b. Data Kecepatan :



Selama dalam perjalanan, kendaraan melaju dengan kecepatan tertentu. Kecepatan = 0 hanya terjadi pada saat akan berangkat dan pada saat telah sampai di tujuan.. Kontrol kecepatan juga termasuk hal yang penting dilakukan. Apakah kendaraan melaju dengan kecepatan di bawah standar atautkah melaju dengan kecepatan di atas standar. Bilamana melaju terlalu kencang bisa berakibat terjadinya kecelakaan yang membahayakan proses pengiriman. Dalam hal ini pihak pusat seharusnya bisa memantau dan segera mengingatkan pengemudi yang menjalankan kendaraan terlalu kencang.

Yang lebih harus diwaspadai adalah bilamana kendaraan berhenti di tengah perjalanan. Hal tersebut rawan terjadinya perampokan atau penodongan. Ada beberapa alasan yang bisa membuat kendaraan untuk berhenti di tengah perjalanan yaitu : lampu lalu lintas, macet, mogok, ban bocor dan lain sebagainya. Sebagai toleransi, untuk beberapa kasus tertentu semisal berhenti di lampu merah, maka waktu yang dibutuhkan tidak boleh melebihi ambang batas. Bilamana melebihi maka akan diberikan warning di kantor pusat untuk investigasi lebih lanjut.

c. Data Posisi :

Dimanakah posisi terkini merupakan informasi yang paling penting untuk diketahui karena hal itu menunjukkan bagaimana jalannya sebuah proses pengiriman. Apakah proses pengiriman berjalan lancar atautkah ada hambatan-hambatan.

Dalam setiap proses pengiriman dengan tujuan tertentu, jalur yang harus dilalui sudah ditentukan sedemikian rupa, yang merupakan jalur terbaik. Dari data posisi yang didapat bisa diketahui apakah kendaraan melewati jalur yang telah ditentukan atau tidak. Apabila ternyata menyimpang dari jalur yang ditentukan, seberapa jauhkah penyimpangannya. Hal ini dapat diketahui dengan membandingkan posisi terakhir dengan jarak terdekat dari jalur yang ditentukan dengan menggunakan rumus jarak terdekat sebagai berikut :

$$d = \frac{|ax + bx + c|}{\sqrt{x^2 + y^2}} \dots\dots\dots(1)$$

Dari rumus tersebut diketahui berapakah jarak penyimpangan yang dilakukan. Apakah sudah melewati batas jarak toleransi atau belum.

Dari posisi ini juga, apabila mobil dirampok atau dilarikan, maka posisinya akan tetap terpantau sehingga akan membantu penanganan yang dilakukan oleh pihak yang berwajib.

5. Tingkat Keamanan Yang Tinggi

Sistem pelacakan yang dibangun haruslah memiliki tingkat keamanan yang tinggi. Tidak mudah disusupi oleh pihak lain yang tidak berkepentingan. Data-data terjaga dengan aman.

6. Memiliki Sistem Peringatan Terhadap Bahaya

Pada saat terjadi gangguan atau bahaya yang berpotensi menimpa pada proses pengiriman, pihak kurir maupun pihak pusat biasanya terlambat untuk mengetahui. Sebelum terjadinya keadaan bahaya tentunya ada kejadian-kejadian yang mendahului. Dan hal tersebut semestinya sudah dapat diprediksi sejak awal. Pada sistem yang dibuat, diharapkan mampu memberikan peringatan terhadap situasi yang berbahaya sesuai yang dengan kriteria-kriteria yang telah ditentukan. Kita bisa mengambil contoh dengan mengambil data kecepatan perjalanan. Secara normal, kendaraan hanya akan berhenti atau kecepatan = 0 pada saat akan berangkat dan pada saat sampai di tujuan. Bilamana berhenti di tengah perjalanan, semisal di lampu lalu lintas, maka masih bisa ditoleransi asalkan waktu berhenti tidaklah melebihi batas. Bilamana melebihi batas maka akan dikirimkan sinyal peringatan, sehingga proses yang menyebabkan berhenti tersebut segera diketahui dan tidak berlanjut ke hal-hal yang lebih gawat.

Macam-macam bahaya atau kendala dapat diklasifikasikan sebagai berikut:

- a. Bahaya tingkat ringan: yang termasuk dalam kategori bahaya yang ringan adalah bahaya yang tidak begitu berpotensi mengganggu jalannya proses pengiriman secara signifikan seperti keterlambatan jadwal pemberangkatan, kendala teknis pada kendaraan.

- b. Bahaya tingkat sedang: yang termasuk dalam kategori bahaya tingkat sedang adalah kemacetan walau hal itu sangat tergantung kepada jenis kemacetan yang terjadi.
- c. Bahaya tingkat berat: yang termasuk dalam kategori bahaya tingkat berat yaitu penodongan, pencurian, perampokan terhadap uang maupun kendaraan yang digunakan dalam proses pengiriman. Selain itu, adanya gangguan alam seperti terjadinya bencana alam banjir atau tanah longsor juga termasuk dalam kategori ini.

7. Pelaporan Yang Lengkap Dan Terstruktur

Pihak eksekutif atau pimpinan memerlukan pelaporan atas kegiatan pengiriman untuk perencanaan ke depan. Laporan yang disajikan, selain akurat sesuai dengan data dilapangan, juga harus fleksibel. Yang dimaksud fleksibel adalah data disajikan sesuai dengan kebutuhan. Misal untuk laporan perjalanan bisa disajikan dalam laporan per minggu, bisa juga dikelompokkan per kasus, dan lain sebagainya.

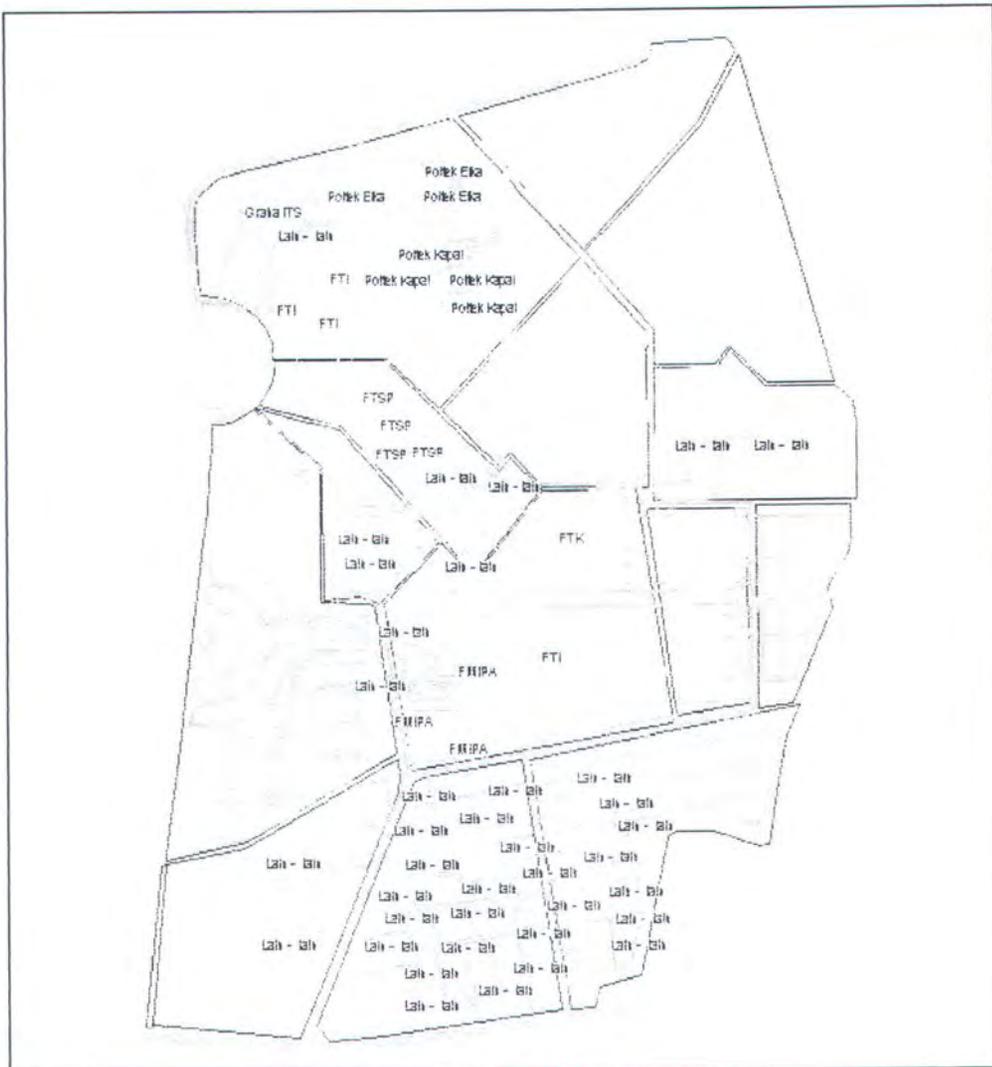
8. Kontrol Sistem Yang Ketat

Dalam setiap aktifitas, pihak pusat atau pimpinan adalah yang paling bertanggung jawab terhadap kelancaran sebuah proses. Pihak pusat seharusnya bisa mengontrol secara penuh proses yang terjadi. Dibutuhkan semacam alat yang bisa mengontrol jarak jauh sehingga bila terjadi

gangguan pada kendaraan, pihak pusat masih memiliki hak kontrol terhadap kendaraan dengan jarak jauh.

3.6. Skenario Pada Data Kecil

Dalam bagian berikut akan dibahas mengenai bagaimana sistem pelacakan bekerja disertai dengan contohnya. Pada contoh ini akan digunakan simulasi dengan menggunakan peta ITS.



Gambar 3.3 Peta ITS

Pada simulasi ini ditentukan bahwa proses pengiriman dilakukan dengan kondisi sebagai berikut:

- i. Start : T. Informatika
- ii. Tujuan : Bank BNI Cabang ITS

Sedangkan jalur yang sudah ditentukan dan yang harus dilewati adalah sebagai berikut :

T.Informatika → Jl. Raya ITS → Bundaran ITS → Jl. Taman Alumni → Jl. T.Mesin → Jl. T. Elektro → Bank BNI

Dengan menggunakan kendaraan yang digunakan sudah dilengkapi dengan AVL, maka proses pengiriman dijalankan Berikut prosedur pengiriman yang dilakukan:

4. Inisialisasi

Mengacu bahwa semua transaksi pada proses pengiriman uang direkam, maka dilakukan pencatatan. Data-data yang perlu dimasukkan adalah sebagai berikut:

- a. Data kendaraan: data kendaraan pengangkut meliputi plat nomor kendaraan. Dari informasi ini akan didapatkan juga nama kurir/driver yang bertugas. Selain itu nomer identitas dari AVL juga ikut tercantum karena atribut AVL ikut ke dalam database kendaraan.
- b. Koper : data dan jumlah koper yang dikirim. Masing-masing koper mempunyai ID masing-masing. Dari data koper tersebut juga dirinci berapa isi dari masing-masing koper sehingga diketahui jumlah total uang yang dikirim.
- c. Tujuan : tujuan pengiriman

- d. Waktu : data tanggal beserta jam pelaksanaan dari proses pengiriman. Hal ini akan terisi secara otomatis.

Disamping data-data transaksi yang dimasukkan, pihak pengirim yakni Informatika menghubungi pihak tujuan yakni bank BNI bahwa akan dilakukan pengiriman sehingga kedua belah pihak bisa mempersiapkan diri.

5. Pemberangkatan

Pada tahap ini dilakukan pengecekan terhadap persiapan pemberangkatan. Apakah semua data sudah lengkap dimasukkan? Kondisi kendaraan bagus? Dan AVL-nya dipastikan dalam kondisi menyala. Kotak uang tidak boleh diutak-atik lagi. Bilamana persiapan dirasa sudah matang maka rombongan diberangkatkan.

6. Perjalanan

Pada saat perjalanan senantiasa dilakukan pengecekan terhadap beberapa kriteria berikut :

a. Status AVL:

Setiap data yang dikirimkan oleh AVL ke pusat, akan selalu terdapat informasi mengenai status dari AVL tersebut. Selama AVL dalam kondisi menyala dan status = 0 maka kondisinya dalam keadaan baik. Apabila kondisinya ternyata berubah, maka pihak pusat akan segera diperingatkan dan secepatnya harus menghubungi kurir dan menanyakan sebab berubahnya status pada AVL.

b. Kecepatan Kendaraan:

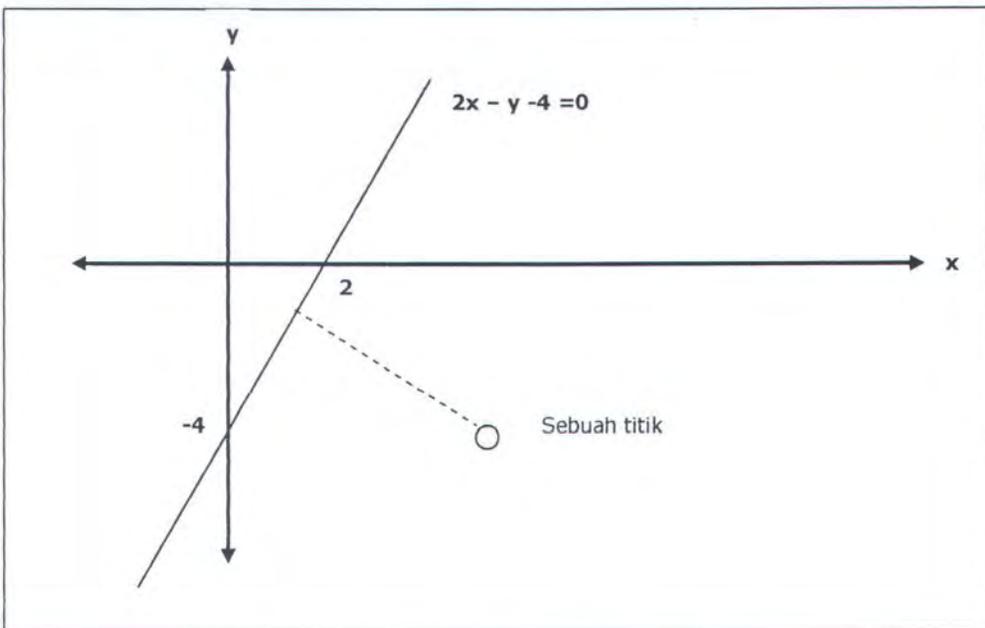
Kecepatan rata-rata selama perjalanan biasanya berkisar antara 60 km/jam sampai dengan 100 km/jam. Apabila melebihi dari 100 km/jam maka pengemudi harus diperingatkan untuk mengurangi kecepatan.

Dan dengan menggunakan rumus jarak terdekat dapat dihitung sebagai berikut :

Kita ambil contoh bahwa jalur yang seharusnya disederhanakan menjadi persamaan garis lurus yang sesuai dengan persamaan berikut :

$$2x - y - 4 = 0$$

Bila digambarkan dalam koordinat kartesian sebagai berikut :



Gambar 3.5 Persamaan Garis Lurus dan Jarak Terdekat

Dari titik ditarik garis lurus yang mempunyai sudut 90^o dengan persamaan garis lurus, maka akan didapatkan jarak terdekat. Misal koordinat titik tersebut adalah (6,-4), maka berdasarkan rumus

Persamaan garis lurus $2x - y - 4 = 0$

$$d = \frac{|ax + by + c|}{\sqrt{a^2 + b^2}}, \text{ dimana}$$



d = jarak terdekat

a = koefisien dari x , dimana $a = 2$

b = koefisien dari y , dimana $b = 1$

c = konstanta, dimana $c = -4$

sedangkan titik $(x, y) = (6, -4)$

$$d = \frac{|ax + by + c|}{\sqrt{x^2 + y^2}}$$

$$d = \frac{|2 \cdot 6 + 1 \cdot (-4) + (-4)|}{\sqrt{6^2 + (-4)^2}}$$

$$d = \frac{|12 - 4 - 4|}{\sqrt{36 + 16}}$$

$$d = \frac{|4|}{\sqrt{52}}$$

$$d = \frac{4}{\sqrt{52}}$$

Bila digunakan satuan meter, maka jarak penyimpangan adalah sebesar

$$d = \frac{4}{\sqrt{52}} \text{ m atau sekitar } 0,56 \text{ m}$$

Dari jarak tersebut, bisa disimpulkan apakah penyimpangan melebihi ambang batas toleransi atau tidak.

7. Serah terima

Perjalanan telah sampai di tujuan. Kurir menyerahkan berita acara serah terima kepada pihak BNI. Kemudian dilakukan pengecekan terhadap kotak uang yang dikirim. Bila dianggap sukses maka kurir melaporkan ke Informatika bahwa

pengiriman sukses. Dengan begitu status pengiriman diset sukses dan kemudian diarsipkan.

8. Selesai

Dengan telah selesainya sebuah pengiriman, maka tidak serta merta berhenti hanya sampai tahap itu saja. Pihak eksekutif sebagai penanggung jawab mengevaluasi hasil pengiriman yang telah dilakukan. Hal ini bertujuan untuk lebih meningkatkan kinerja pada saat yang akan datang. Tidak selamanya pengiriman berjalan lancar. Untuk itu harus dipersiapkan sedini mungkin untuk hal-hal yang tidak terduga.



BAB IV
PERANCANGAN PERANGKAT LUNAK

BAB IV

PERANCANGAN PERANGKAT LUNAK

Bab ini menjelaskan tentang tahapan proses perancangan perangkat lunak mulai dari penggambaran aplikasi secara umum sampai perancangan data, perancangan proses dan perancangan antar muka dengan pengguna.

4.1. Desain Fitur

Sebelum membahas perancangan proses maupun perancangan data, mengacu pada analisa kebutuhan sistem yang dibahas pada bab 3 dengan poin-poin berikut:

- a. Memiliki Perekaman Data Perjalanan
- b. Dapat Mengetahui Posisi Terkini Perjalanan
- c. Bersifat Online-Real Time
- d. Analisa Data Perjalanan
- e. Tingkat Keamanan Yang Tinggi
- f. Memiliki Sistem Peringatan Terhadap Bahaya
- g. Pelaporan Yang Lengkap Dan Terstruktur
- h. Kontrol Sistem Yang Ketat

maka akan dijelaskan fitur-fitur yang akan dibuat untuk memenuhi dari kebutuhan tersebut. Perancangan fitur sendiri akan dapat menjelaskan secara global gambaran tentang perangkat lunak yang akan dibuat.

4.1.1. Perekaman Data Perjalanan

Untuk bisa merekam data-data perjalanan, maka dibuat database yang bisa menampung segala jenis transaksi. Macam-macam data itu sendiri dapat dibagi menjadi 2 kelompok besar, yaitu:

- a. Data Master : berisi data-data acuan yang tidak sering berubah. Contoh yang merupakan data master adalah data master koper. Data master koper berisi tentang spesifikasi sebuah koper.
- b. Data Transaksi : berisi data-data yang merupakan hasil pengolahan transaksi. Isi dari tabel transaksi akan sangat beragam tergantung dari jenis transaksi yang dilakukan. Contoh data transaksi adalah data perjalanan.

Skema database yang dibuat adalah bersifat relasional yaitu ada relasi yang menghubungkan antar tabel yang dibuat. Hal ini akan lebih memudahkan dalam perancangan dan juga menghindari adanya redundansi data sesuai dengan kaidah-kaidah pembuatan database.

Dalam implementasinya nanti akan digunakan MS SQL Server 2000 dengan pertimbangan bahwa tool tersebut mudah untuk digunakan dan mudah untuk perawatan.

4.1.2. Pemantauan Posisi Terkini

Posisi terkini dari kendaraan haruslah bisa dipantau. Dengan menggunakan AVL yang online, maka kebutuhan ini akan dapat terpenuhi karena AVL akan selalu mengirimkan data posisi ke kantor pusat sesuai dengan rentang waktu yang telah ditentukan. Rentang waktu inipun bersifat fleksibel tergantung

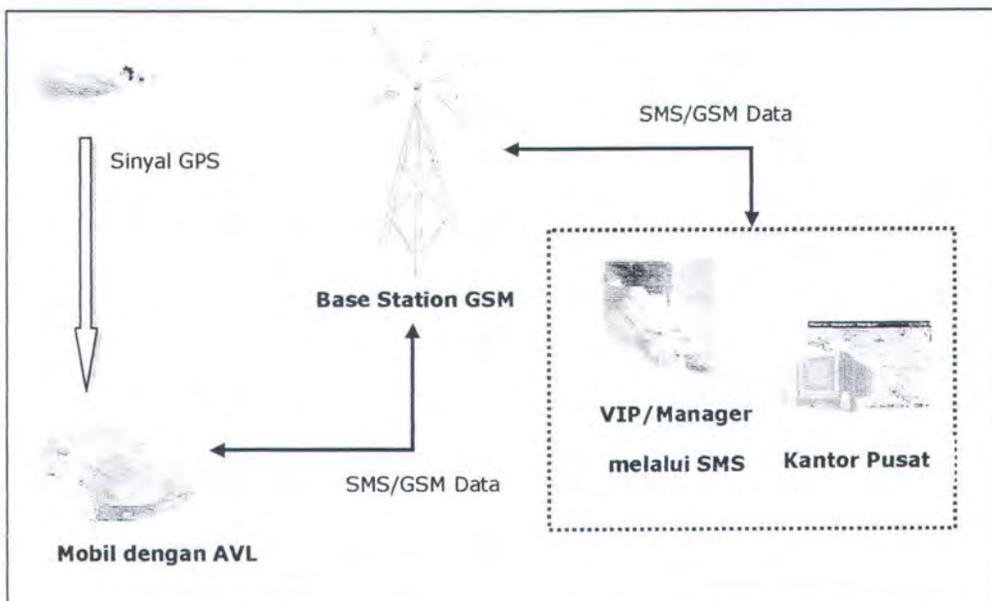
dari pengguna dalam menentukan rentang waktu yang diinginkan sesuai kebutuhan.

Dengan menempatkan komputer yang disertai dengan receiver dari AVL maka kantor pusat akan bisa mendapatkan data posisi terkini.

4.1.3. Bersifat Online-Real Time

Bersifat online berarti data diperoleh pada saat itu juga. Selisih waktu perolehan data terkini menjadi faktor penentu yang sangat penting. Untuk memenuhi kebutuhan ini maka digunakan komunikasi yang dapat mengatasi kendala ruang dan waktu. Sebagai alternatif dipilih media SMS yang menggunakan teknologi GSM. Dengan jaringan yang luas dan meliputi wilayah yang luas serta akses data yang cepat, media SMS merupakan solusi terbaik.

Dalam tugas akhir ini nanti akan disimulasikan bagaimana menerima data yang berasal dari sms dan kemudian menyimpannya ke dalam database.



Gambar 4.1 Cara Kerja Sistem AVL On-Line

4.1.4. Analisa Data Perjalanan

Dalam sebuah proses pengiriman, yang menentukan adalah hasil akhirnya. Apakah pengiriman berjalan sukses atau tidak. Tetapi bagaimanakah sebuah perjalanan dikatakan sukses.

Berangkat dari kondisi kendaraan bisa ada keadaan dimana kendaraan bergerak, dan ada keadaan dimana kendaraan berhenti. Sedangkan situasi yang terjadi bisa dipilah menjadi aman, tidak aman, dan darurat. Hal ini bisa digambarkan dalam matrik berikut ini:

	Aman	Tidak Aman	Darurat
Diam	√	√	√
Bergerak	√	√	√

Tabel 4.1 Matrik Kondisi Perjalanan

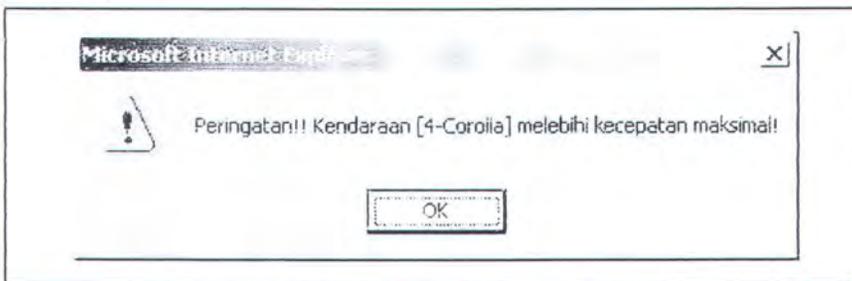
Dalam uraian lebih detail, berbagai macam kondisi perjalanan bisa dijelaskan sebagai berikut:

1. Diam, aman : bilamana kendaraan berhenti pada saat akan diberangkatkan dan juga pada saat telah sampai di tujuan.
2. Diam, tidak aman : bilamana kendaraan berhenti ditengah perjalanan. Kemungkinan penyebabnya adalah berhenti pada saat lampu merah atau terjebak kemacetan
3. Diam, darurat : kendaraan berhenti ditengah perjalanan dalam waktu yang lama melebihi dari 5 menit dan tidak ada konfirmasi dari pengendara kendaraan. Hal ini bisa juga disebabkan dengan adanya bencana alam atau telah terjadi perampokan dan tindak pidana pada rombongan.

4. Bergerak, aman: bilamana kendaraan bergerak dalam kecepatan yang normal selama dalam perjalanan.
5. Bergerak, tidak aman: bilamana kendaraan bergerak melebihi kecepatan yang ditentukan.
6. Bergerak, darurat: bilamana kendaraan bergerak tetapi status AVL telah berubah.

Dengan melihat kondisi yang terjadi, reaksi yang diberikan oleh sistem juga berbeda-beda. Berikut macam-macam reaksi yang diberikan:

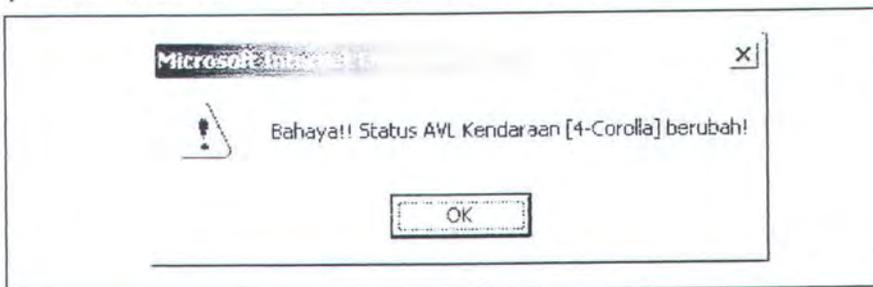
1. Kondisi aman : tidak ada reaksi
2. Kondisi tidak aman : pada tahap ini reaksi yang diberikan berupa peringatan. Pada layar monitor akan muncul peringatan mengenai keadaan yang terjadi.



Gambar 4.2. Contoh peringatan yang diberikan

Diharapkan dengan pemberian peringatan ini maka operator maupun pihak manajer yang mengetahui akan segera menghubungi kurir yang sedang melakukan perjalanan dan kemudian mengkonfirmasi kendala apa yang mengganggu perjalanan. Pihak kantor pusat bisa memberikan saran sebagai jalan keluar untuk mengatasi kendala yang terjadi. Tetapi bilamana tingkat gangguan meningkat menjadi bahaya atau darurat maka bisa diambil langkah-langkah penanganan yang efektif.

3. Kondisi darurat: pada tahap ini reaksi yang diberikan adalah dengan memberitahukan peringatan bahaya mengenai apa yang telah terjadi. Tidak cuma hanya sekedar pada tingkat peringatan tetapi sudah memberitahukan bahwa sudah memasuki level darurat.

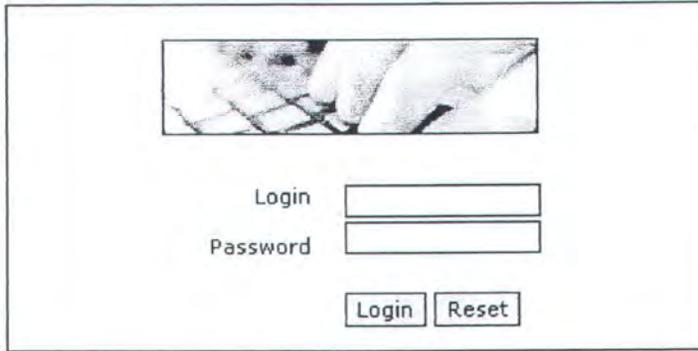


Gambar 4.3. Contoh peringatan darurat

Dengan pemberitahuan peringatan bahaya, maka pihak kantor pusat diharapkan segera merespon keadaan ini termasuk segera menghubungi dan berkoordinasi dengan pihak yang berwajib. Alangkah lebih baik bila ternyata kurir yang sedang dalam perjalanan tersebut masih bisa dihubungi dan ditanya mengenai kendala yang terjadi. Hal tersebut dapat menambah kelengkapan informasi mengenai situasi yang terjadi sehingga langkah-langkah penanganan yang diambil juga sesuai.

4.1.5. Tingkat Keamanan Yang Tinggi

Selain proses yang bagus, sistem harus memiliki tingkat keamanan yang bagus pula. Untuk mendukung sistem keamanan ini, maka pengguna diharuskan login untuk mengakses menu-menu dalam sistem. Hak masing-masing user nantinya tergantung dari level hak akses yang dimilikinya.



Gambar 4.4. Fasilitas Login

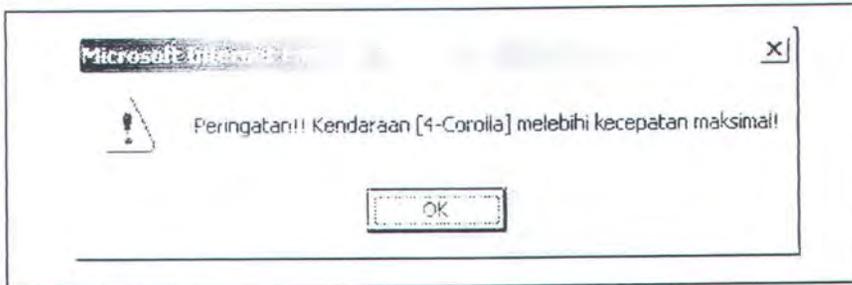
Hak akses user dibuat berdasarkan kewenangan yang dimiliki. Berikut jenis level beserta hak yang dimiliki :

- a. Administrator: merupakan level yang paling tinggi. Bertugas untuk mengelola data terutama untuk data-data master. Bisa melakukan operasi tambah, edit, hapus pada semua tingkat dan semua menu.
- b. Manager: manajer bertugas untuk mengawasi pengoperasian sistem pelacakan ini. Kebanyakan menu dari manager ini berupa laporan-laporan. Manajer tidak dapat merubah data transaksi.
- c. Asisten Manager: asisten manager bertugas mengawasi operator secara langsung. Hanya setingkat level asisten manajer ke atas yang bisa memberikan status akhir perjalanan. Asisten manajer tidak dapat merubah data transaksi
- d. Operator: operator bertugas untuk memasukkan data transaksi. Selain memasukkan data transaksi, operator juga bisa membuat laporan-laporan.

4.1.6. Memiliki Sistem Peringatan Terhadap Bahaya

Selama proses pengiriman berjalan, data-data senantiasa dikirimkan ke kantor pusat. Dari hasil analisa data yang dikirimkan, apabila terjadi keanehan

atau tidak sesuai dengan kriteria, maka sistem akan langsung memberikan peringatan.



Gambar 4.5. Sistem Peringatan

Dengan kemampuan memberikan peringatan, pihak pusat akan bisa cepat bereaksi terhadap hal-hal yang tidak diinginkan.

4.1.7. Pelaporan Yang Lengkap Dan Terstruktur

Untuk memenuhi kebutuhan pelaporan ini maka dirancang laporan-laporan apa saja yang bisa dibuat. Berikut beberapa macam laporan yang memungkinkan untuk dikerjakan :

- a. Laporan pengiriman : berupa daftar pengiriman yang telah dilakukan
- b. Laporan pengiriman berdasarkan status: setiap selesai pengiriman, maka akan dimasukkan status ke dalamnya. Disini akan dibuat list pengiriman berdasarkan status yang dipilih. Misal: pengiriman kapan saja yang mengalami gangguan atau tidak sukses.

4.1.8. Kontrol Sistem Yang Ketat

Dalam menjalankan fungsi pengawasan, pihak kantor pusat menghendaki adanya interaksi langsung terhadap sistem. Pada saat pengiriman, meskipun kendaraan pengangkut sudah jauh dari kantor pusat, tetapi kantor pusat masih memiliki akses kontrol terhadap kendaraan tersebut.

Bila dalam pelaporan data dari AVL ke kantor pusat menggunakan media SMS maka media kontrol dari kantor pusat ke kendaraan juga menggunakan media SMS.

Dalam tugas akhir ini, khusus untuk kontrol via SMA tidak diimplementasikan mengingat batasan dari tugas akhir.

4.2. Perancangan Proses

Desain proses digunakan untuk mengetahui proses apa saja yang berlangsung pada sistem. Untuk desain proses dari perangkat lunak ini digunakan Diagram Aliran Data (DAD). Pembuatan DAD tersebut menggunakan *Tool Power Designer* versi 6.0.

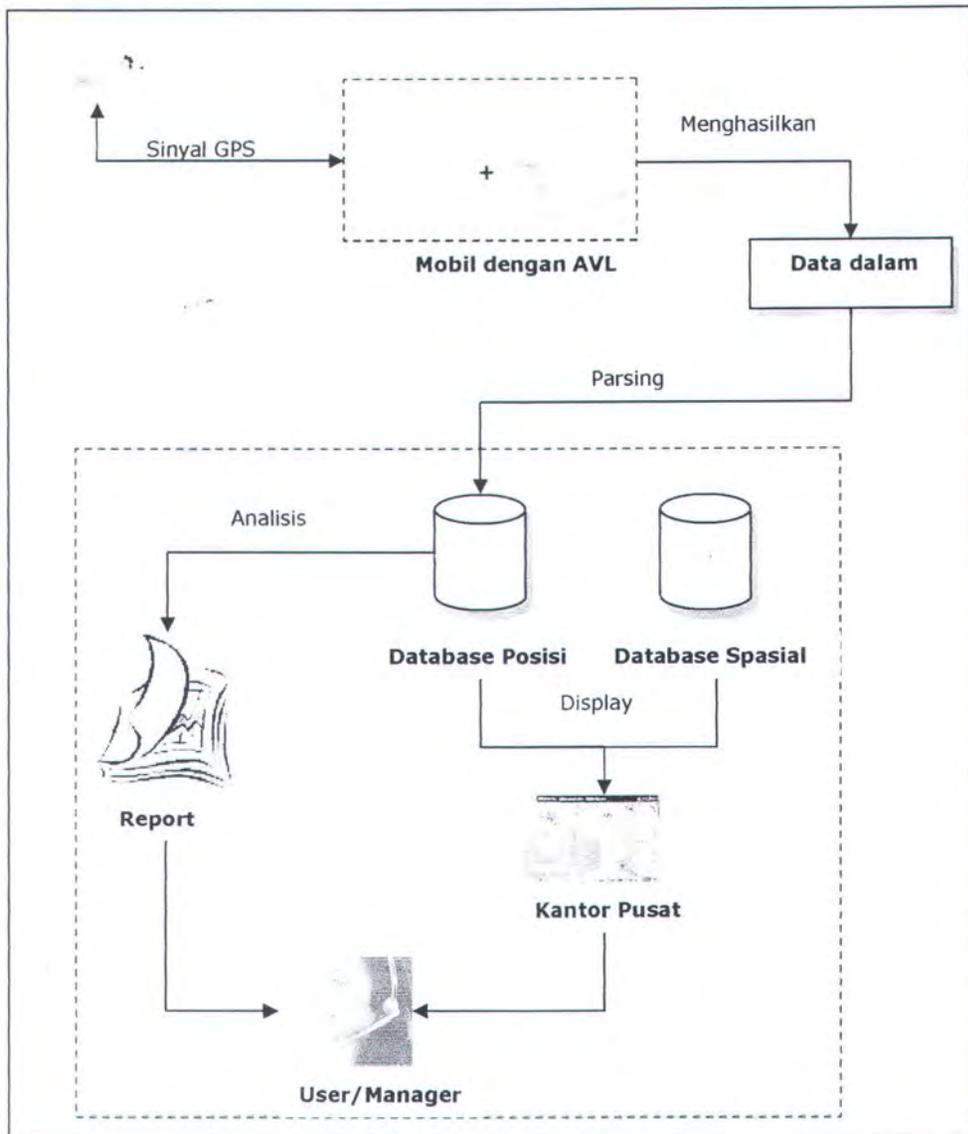
DAD akan menunjukkan hubungan antara proses satu dengan yang lain, entitas-entitas atau orang-orang, serta data masukan dan data keluaran yang terlibat dalam proses-proses tersebut.

Sebelum dijelaskan lebih detail mengenai pembuatan Diagram Alir Data, maka akan dibahas terlebih dahulu mengenai cara kerja sistem yang akan dibuat. Dengan adanya gambaran mengenai proses secara global akan lebih mudah untuk mendefinisikan yang lebih detail.

Dalam sistem akan digunakan alat AVL. AVL ini akan mengirimkan data-data dalam bentuk SMS yang kemudian diterima oleh pusat. Data-data tersebut kemudian diparsing dan dimasukkan ke dalam database posisi.

Selain database posisi, juga ada database spasial yang menyimpan informasi peta area pengiriman. Kedua macam data yaitu data posisi dan data spasial dipadukan kemudian ditampilkan sebagai peta digital yang dinamis dan berbasis web.

Berikut adalah gambaran kerja dari sistem secara umum :



Gambar 4.6. Proses Pelacakan

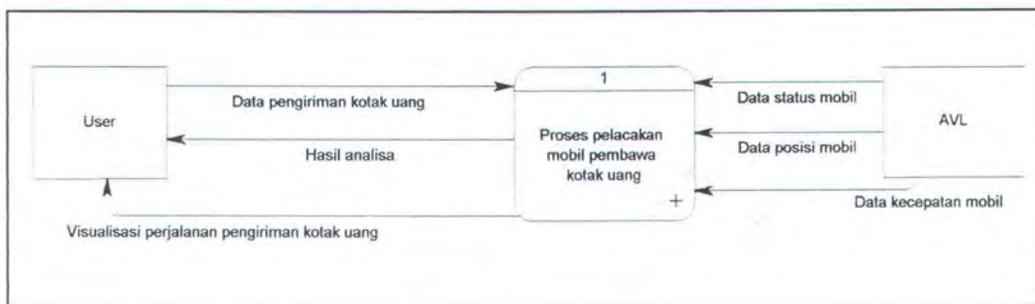
Dengan tampilan antar muka yang berbasis web akan lebih memudahkan untuk user dalam mengaksesnya. Selain ditampilkan ke dalam peta digital, data posisi juga akan dianalisis dan kemudian disajikan ke dalam report untuk bahan

evaluasi sehingga untuk pelayanan dapat lebih ditingkatkan untuk masa yang akan datang.

4.2.1. DAD Level 0 : Proses Pelacakan Mobil Pembawa Kotak Uang

DAD dimulai dari level 0 (nol), yang biasanya disebut sebagai *System Context Diagram*. DAD level 0 menggambarkan satu proses global yang meliputi seluruh proses dalam sistem. Sistem perangkat lunak yang dibuat adalah untuk melacak kotak uang. Secara ideal dengan sistem semacam ini maka tiap kotak uang akan dipasang pelacak. Akan tetapi hal ini menyebabkan kesulitan sehingga untuk lebih memudahkan maka mobil sebagai pembawa kotak uang dianggap cukup mewakili dari keberadaan kotak uang tersebut. Dengan demikian yang menjadi fokus pelacakan secara teknis adalah mobil yang digunakan untuk mengangkut kotak uang.

DAD Level 0 dapat digambarkan pada gambar berikut :



Gambar 4.7 Diagram DAD Level 0

Pada DFD level 0 ini ada beberapa entitas yang terlibat. Entitas tersebut merupakan entitas induk yang merupakan komponen utama dalam sistem. Berikut beberapa entitas:

1. User : user merupakan entitas yang berperan untuk memasukkan data transaksional perjalanan. User menerima output berupa posisi dari kotak uang yang sedang dikirim dan juga hasil analisa selama perjalanan yang menunjukkan kondisi aman tidaknya kotak uang yang dikirim.
2. AVL: merupakan entitas yang memberikan inputan berupa data posisi, data kecepatan, dan data status dari kotak uang yang dikirim. AVL ini hanya memberikan input saja tidak menerima output.
3. Administrator: Administrator merupakan entitas yang terpisah dari keseluruhan proses sistem. Keterlibatan administrator adalah sebagai pengaturan hak pengguna lainnya dan sebagai penyalin data (*back up*).

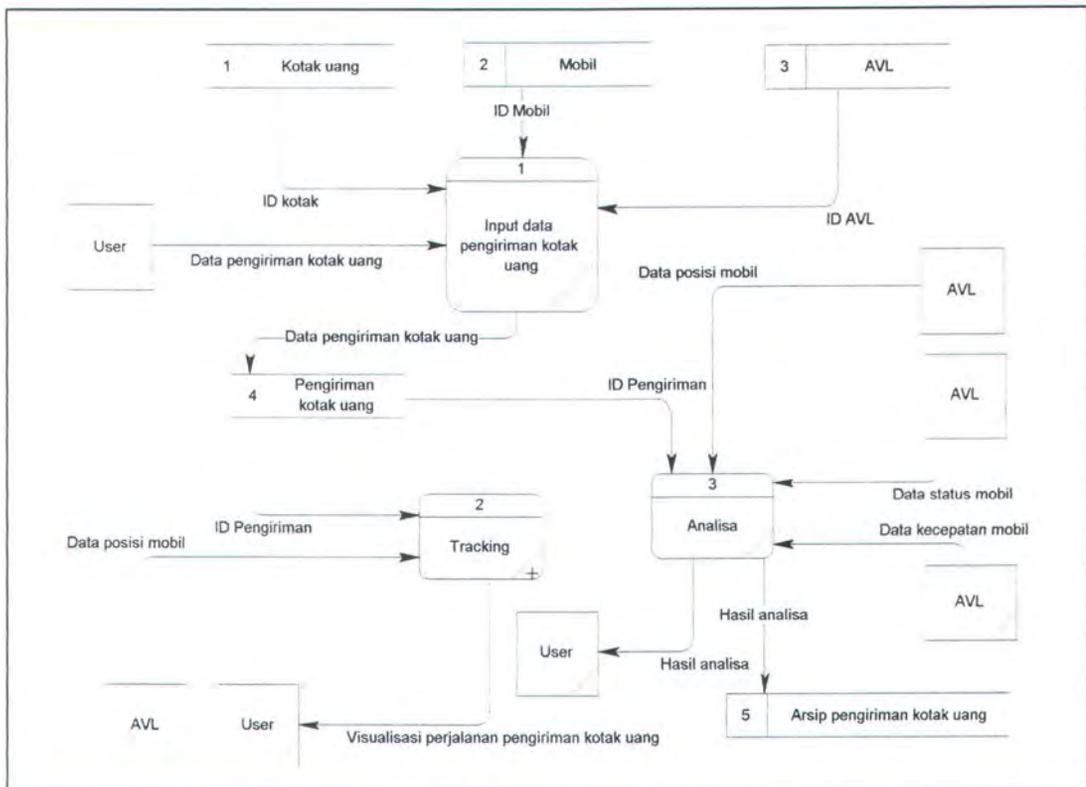
4.2.2. DAD Level 1

Langkah selanjutnya adalah pemecahan (dekomposisi) sistem menjadi proses – proses yang merupakan fitur – fitur utama pada aplikasi. Pada DAD Level 1 terdapat 3 proses yaitu proses input data perjalanan, tracking, dan analisis. Pada proses input data perjalanan, user memasukkan semua data yang terlibat dalam sebuah perjalanan yaitu data kotak uang, data mobil, data AVL beserta data transaksional yang lain yang berupa waktu pengiriman, jumlah uang yang dikirim dan lain sebagainya. Data-data tersebut kemudian akan disimpan ke dalam database.

Proses tracking merupakan proses dimana data perjalanan dari database dan data-data yang diperoleh ditampilkan ke dalam sebuah peta digital.

Proses analisa merupakan proses yang memeriksa data-data yang diterima antara lain data posisi, data kecepatan, dan data status yang merupakan parameter dari keberhasilan dalam sebuah proses pengiriman.

Proses – proses tersebut digambarkan pada DAD level 1 seperti tampak pada Gambar berikut :



Gambar 4.6 Diagram DAD Level 1

Pada proses input data pengiriman kotak uang adalah proses yang dilakukan secara manual dengan memasukkan data-data transaksi pengiriman. Dengan demikian untuk proses tersebut sudah merupakan proses yang atomic dan tidak bisa dipecah lagi.



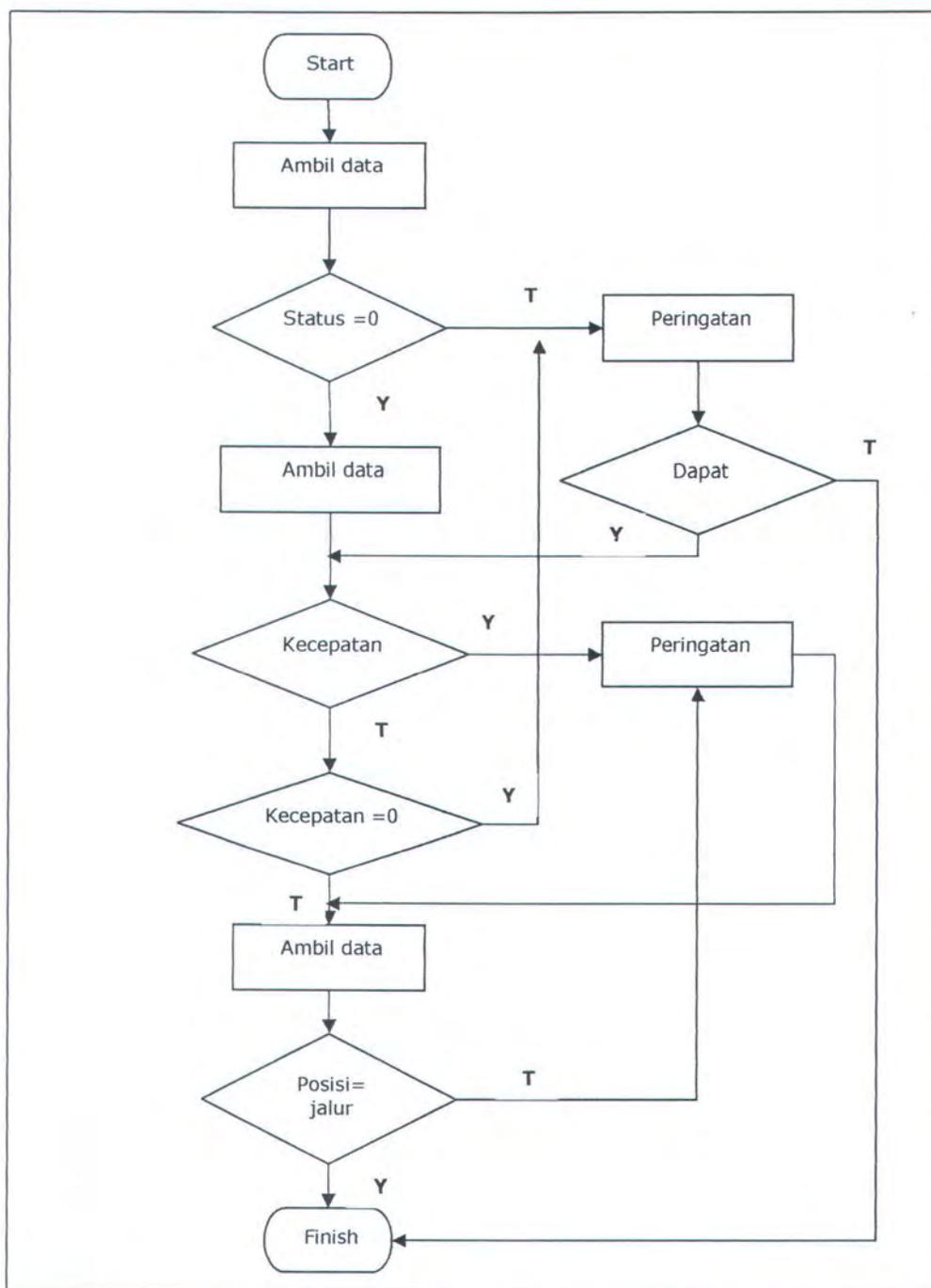
Untuk proses yang kedua yaitu proses tracking masih dipecah lagi menjadi proses-proses yang lebih detail. Dengan demikian pada proses tracking ini selanjutnya akan didekomposisi pada DAD level 2.

Pada proses ketiga yaitu proses analisa merupakan proses yang sudah atomic. Proses analisa tidak akan didekomposisi menjadi proses-proses lagi. Pada proses analisa yang dilakukan adalah :

1. Ambil id pengiriman sebagai kunci untuk menunjukkan bahwa analisa dilakukan pada pengiriman yang mana..
2. Ambil data status.
3. Cek data status, bila status=0 proses dilanjutkan, bila status=1 maka tampilkan peringatan.
4. Ambil data kecepatan.
5. Cek data kecepatan, bila kecepatan > 100 km/jam atau kecepatan = 0 selama > 3 menit maka tampilkan peringatan.
6. Ambil data posisi.
7. Cek data posisi, bila menyimpang maka tampilkan peringatan.
8. Selesai.

Peringatan yang diberikan ada 2 hal. Hanya peringatan saja dan peringatan bahaya. Peringatan biasa diberikan untuk mencegah kondisi berlanjut ke tingkat yang lebih parah. Sedangkan untuk taraf bahaya berpotensi besar untuk menyebabkan kegagalan proses pengiriman. Maka disini perlu dilihat apakah penyebab peringatan bahaya bisa diselesaikan, bila dapat diselesaikan maka

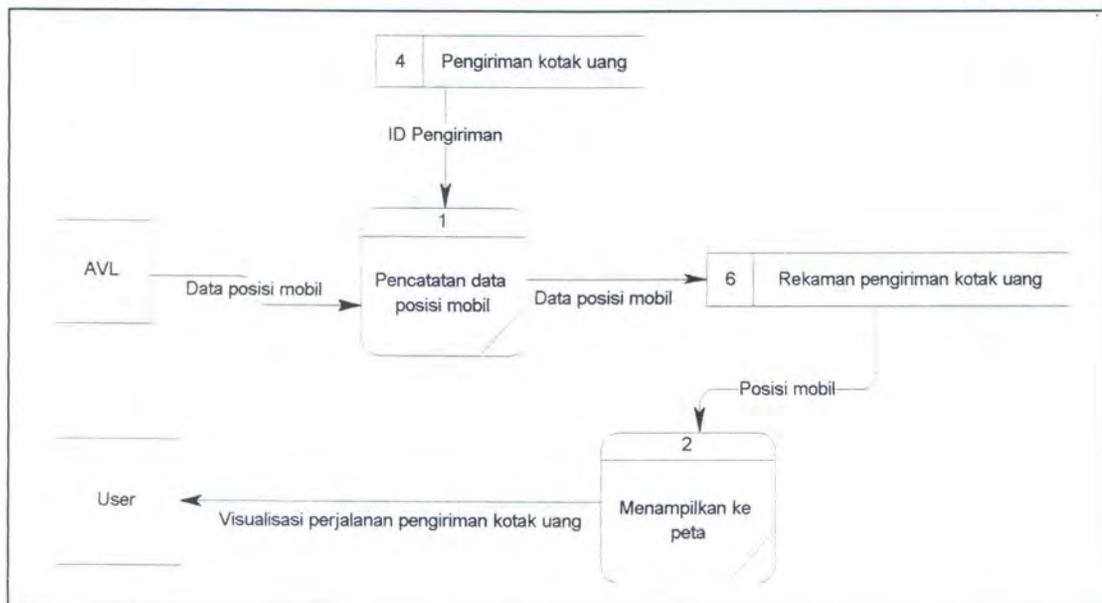
proses bisa dilanjutkan, tetapi bila tidak dapat diselesaikan maka proses dianggap gagal dan proses berhenti.



Gambar 4.7 Flowcart Proses Analisa

4.2.3. DAD Level 2 : Proses Tracking

Proses tracking kemudian di dekomposisi kembali menjadi proses pencatatan data posisi mobil, dan menampilkan ke peta. Penggambaran dekomposisi ini dibuat pada DAD Level 2 Proses Tracking seperti tampak pada gambar berikut:



Gambar 4.7 Diagram DAD Level 2 : Proses Tracking

Pada proses tracking ini, data yang diterima dari AVL disimpan dulu ke dalam database dan digabungkan dengan data-data pengiriman kotak uang yang dimasukkan sebelumnya. Hubungan antara data perjalanan dan rekaman pengiriman kotak uang adalah master-detail. Proses pencatatan yang dilakukan hanyalah memasukkan data yang ada ke dalam database. Dengan demikian proses pencatatan data posisi mobil merupakan proses yang atomik dan sederhana sehingga tidak perlu dijelaskan lagi

Setelah data-data tersebut disimpan, maka akan ditampilkan ke dalam peta digital yang berbasis web sehingga user dapat mengetahui posisi dari kotak uang. Proses menampilkan koordinat posisi mobil dapat dijelaskan dengan pseudo-code berikut ini:

1. Ambil id_perjalanan, id_posisi, posisi_x, posisi_y dari data rekaman pengiriman kotak uang. Id_perjalanan untuk memilih perjalanan mana yang akan ditampilkan. Id_posisi menunjukkan urutan titik yang ada.
2. Simpan posisi_x dan posisi_y dari id_posisi = 1 ke dalam variabel.
3. Buat layer posisi pada peta
4. Gambar titik x dan titik y ke dalam peta
5. Ulangi sampai dengan id_posisi=maks

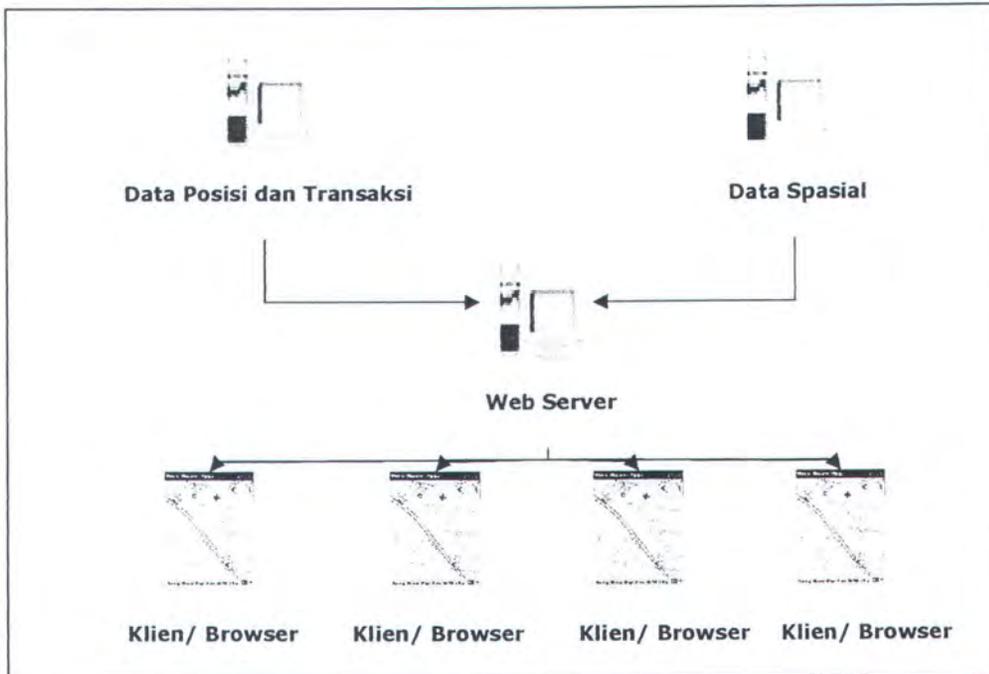
Proses menampilkan ini akan terus diulangi sampai id_posisi menunjukkan tingkat yang paling tinggi.

4.3. Arsitektur Sistem

Pada perangkat lunak sistem pelacakan pengiriman kotak uang menggunakan 2 server data yang terbagi menjadi data spasial dan data nonspasial beserta data-data transaksi. Selain server data dibutuhkan juga server web yang akan menangani akses sistem via web.

Memang secara teori, server tersebut letaknya terpisah-pisah. Akan tetapi dalam prakteknya bisa diletakkan dalam 1 komputer saja asalkan spesifikasinya bisa memenuhi untuk menjalankan ketiga servis tersebut. Untuk server data-data transaksional dan data-data nonspasial digunakan RDBMS MS SQL server 2000. Alasan penggunaan RDBMS ini karena mudah digunakan sehingga tidak menyulitkan end user dalam melakukan maintenance terhadap data-data yang ada.

Dalam pembuatan perangkat lunak pelacakan pengiriman kotak uang menggunakan arsitektur sebagai berikut :



Gambar 4.11 Arsitektur Sistem Secara Umum

Dari gambar 4.11 bisa diberikan penjelasan sebagai berikut:

1. Data Posisi dan Transaksi

Database ini menyimpan data-data posisi selama perjalanan. Semua masukan yang dihasilkan oleh AVL disimpan disini. Selain data posisi, juga digunakan data transaksi. Yang termasuk data transaksi adalah data-data master seperti master_koper, master_mobil dan data transaksi pengiriman uang. Untuk database ini digunakan MS SQL Server 2000 Personal Edition yang di letakkan di Lp-46.

2. Data Spasial

Data spasial merupakan data yang dibutuhkan sebuah peta. Di dalamnya tersimpan informasi mengenai komponen-komponen yang ada didalamnya,

semisal informasi jalan, informasi ketinggian, informasi kondisi lahan. Dalam pembangunannya digunakan software MapInfo 7 Professional.

3. Web Server

Untuk bisa diakses ke dalam web, maka dibutuhkan web server. Untuk itu digunakan MS IIS 5.0 dalam platform MS Windows 2000. Bahasa pemrogram yang digunakan adalah ASP.

4. Klien/Browser

Klien berguna untuk mengakses menu-menu yang disediakan. Klien yang dimaksud disini adalah browser untuk mengakses halaman-halaman web. Browser yang biasa digunakan adalah Internet Explorer 6.0

4.4. Perancangan Data

Perancangan data bertujuan untuk menentukan kebutuhan perangkat lunak yang akan dibuat. Bentuk data apa saja yang diperlukan dan data yang bagaimana yang harus dibuat, sehingga data tersebut dapat digunakan secara optimal digunakan pada saat menjalankan perangkat lunak.

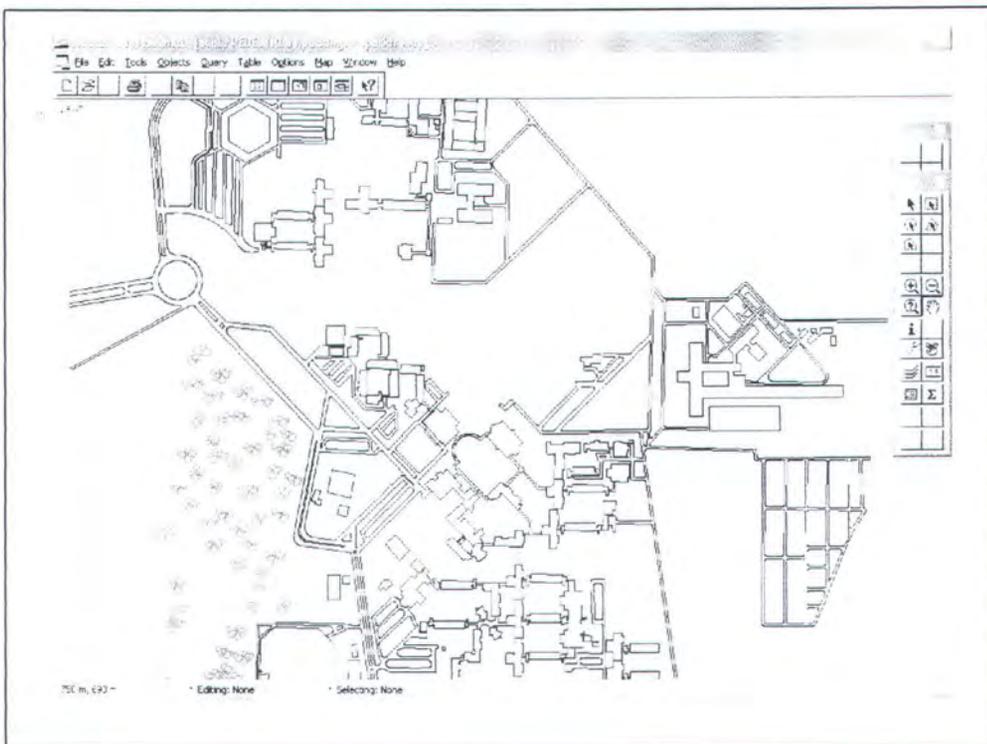
Data yang digunakan untuk implementasi perangkat lunak ini dibagi menjadi tiga bagian utama, yaitu data masukan, data yang digunakan selama proses serta data keluaran.

4.4.1. Data Masukan

Di dalam perancangan ini, perangkat lunak yang dipergunakan berupa peta digital yang telah diproyeksikan ke dalam koordinat UTM (Universal Transverse Mercator). Sedangkan *layer-layer* yang dipakai antara lain adalah jalan, gedung,

data-data posisi kendaraan yang didapatkan dari AVL, serta peta lingkungan ITS sebagai peta dasarnya.

Contoh tampilan peta dapat dilihat pada gambar 4.12, sedangkan spesifikasi dari masing-masing peta tersebut dapat dilihat pada tabel 4.2.



Gambar 4.12 Contoh tampilan peta masukan

Data Peta	Tipe Fitur	Format Data
jalan	Garis	Vector MapInfo
gedung	Poligon	Vector MapInfo

Tabel 4.2. Spesifikasi Data Masukan

Layer-layer yang ditampilkan pada peta, antara lain jalan di ITS , gedung , peta wilayah, serta koordinat posisi kendaraan yang diterima dari AVL. Layer-

layer tersebut dipilih untuk ditampilkan pada peta yakni untuk memetakan posisi-posisi kendaraan terkini, sehingga pengguna dapat lebih mudah melihat posisi kendaraan sedang berada.

Untuk field dari data-data posisi kendaraan yang akan diterima dari AVL, antara lain koordinat x (longitude), koordinat y (latitude), kecepatan, arah, jarak yang telah ditempuh, dan waktu penerimaan SMS

4.4.2. Data Proses

Data proses adalah data yang digunakan selama proses berlangsung. Ada beberapa macam data yang dipakai untuk data proses, yaitu:

- **Atribut Koordinat X dan Y dari posisi kendaraan**

Atribut koordinat X dan Y akan diproses untuk menampilkan posisi kendaraan yang dipantau pada peta digital.

- **Atribut Status dari AVL**

Atribut status dari AVL berfungsi untuk mengetahui kondisi dari AVL selama dalam perjalanan. Ketika digunakan, AVL harus dapat dipastikan dalam kondisi yang baik. Karena AVL biasanya disambungkan ke dalam sistem kendaraan, maka adanya gangguan pada AVL bisa berarti gangguan pada kendaraan juga.

- **Atribut Kecepatan Kendaraan**

Atribut kecepatan kendaraan diproses untuk memberikan peringatan bahwa kendaraan telah melampaui batas kecepatan maksimum dan telah melaju di bawah kecepatan minimum.

4.4.3. Data Keluaran

Data keluaran adalah data yang dihasilkan dari proses-proses yang terdapat dalam sistem. Ada dua macam data yang dihasilkan dari proses-proses tersebut, antara lain:

- **Peta posisi kendaraan**

Data keluaran berupa peta yang menampilkan posisi kendaraan selama proses perjalanan.

- **Hasil Analisis**

Terdapat tiga jenis analisis dalam sistem ini, yaitu:

- a) Kondisi status AVL

Data keluaran dari hasil analisis yaitu peringatan yang akan ditampilkan di web apabila terdapat status AVL yang berubah atau mati dalam perjalanan..

- b) Pengontrolan kecepatan agar tidak melampaui batas kecepatan maksimum yang diijinkan

Data keluaran dari proses hasil analisis ini yaitu berupa peringatan yang akan ditampilkan di web apabila terdapat kendaraan yang melampaui batas kecepatan yang diperbolehkan karena hal ini sangat berbahaya bagi keselamatan perjalanan.

- c) Perhitungan posisi relatif pada jalur yang ditentukan.

Data keluaran dari proses hasil analisis ini adalah berupa peringatan yang akan ditampilkan di web apabila terdapat kendaraan yang telah keluar dari jalur yang ditentukan.

4.4.4. Desain Basis Data

Pada perancangan data ini dibuat diagram yang menggambarkan entitas – entitas data yang terlibat beserta relasi yang terjadi antara entitas tersebut. Entitas data disini merupakan representasi dari tabel pada basisdata yang nantinya akan dibuat. Berdasarkan dari analisa kebutuhan tentang data-data yang dijadikan sebagai masukan, data-data yang diproses, dan data-data hasil yang dikeluarkan, maka dibutuhkan tabel-tabel sebagai berikut.

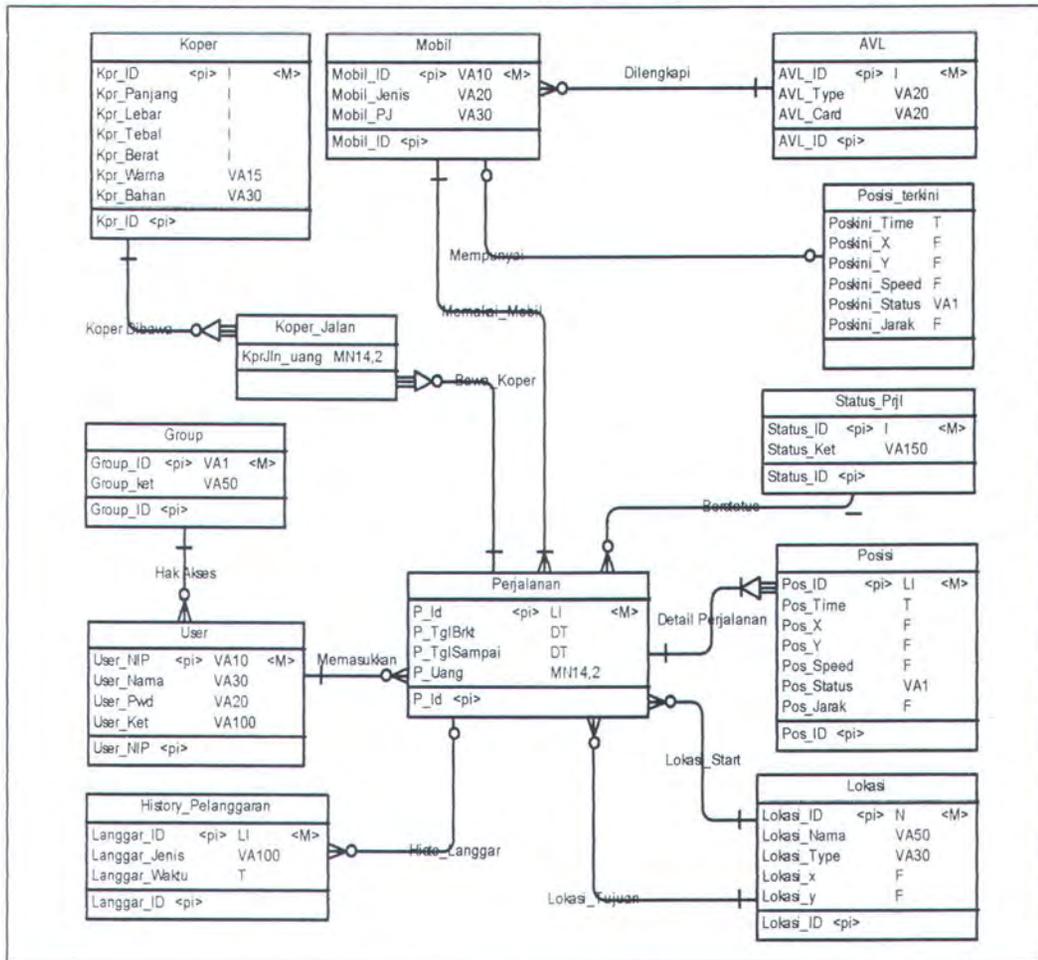
1. Tabel AVL: merupakan table master yang berfungsi untuk menyimpan informasi mengenai spesifikasi dari alat AVL yang digunakan beserta nomer kartu SIM yang ditancapkan di dalamnya.
2. Tabel Group: merupakan tabel master yang berfungsi untuk menyimpan kategori group dari user yang akan menentukan hak akses yang dimiliki oleh user yang bersangkutan.
3. Tabel Koper: merupakan tabel master yang berfungsi untuk menyimpan informasi mengenai detail fisik koper yang digunakan dalam pengiriman.
4. Tabel Koper_Jalan: merupakan tabel traksaksional yang menyimpan informasi mengenai jenis koper, mobil yang digunakan dalam proses pengiriman uang. Di samping itu juga menyimpan informasi detail dari kapasitas uang masing-masing koper.
5. Tabel Mobil: merupakan tabel master yang berfungsi untuk menyimpan informasi mengenai jenis mobil yang digunakan dalam proses pengiriman beserta orang yang bertanggung jawab terhadap mobil tersebut.

6. Tabel Perjalanan: merupakan tabel transaksional yang berfungsi untuk menyimpan setiap data perjalanan pengiriman. Informasi ini mencakup tanggal pemberangkatan sampai tanggal sampai. Kemudian start pemberangkatan dan tujuan yang ingin ditempuh. Dari sini juga nantinya akan dapat diketahui status dari perjalanan sebagai kesimpulan akhirnya. Apakah pengiriman berjalan sukses atau ada hambatan. Hal ini bisa dilihat dari field status_id. Bilamana pada tabel koper_jalan terdapat jumlah uang yang dibawa pada masing-masing koper maka pada tabel ini terdapat jumlah keseluruhan uang yang dikirim yang tersimpan dalam field p_uang.
7. Tabel Posisi: merupakan tabel transaksional yang berfungsi untuk menyimpan posisi yang terbaru selama dalam proses perjalanan. Database akan terupdate secara otomatis berdasarkan data yang dikirim oleh alat AVL. Data yang nantinya digunakan untuk proses analisis adalah data tentang posisi, kecepatan dan status. Status yang dimaksud dalam tabel ini adalah untuk menyimpan kondisi dari alat, apakah dalam keadaan baik ataukah ada yang berusaha membuka atau merusak dengan sengaja. Data ini berguna untuk mengecek keadaan darurat.
8. Tabel Status: merupakan tabel master yang berfungsi untuk menyimpan macam-macam status perjalanan.
9. Tabel User: merupakan tabel master yang berfungsi untuk menyimpan data-data user yang menggunakan sistem ini.
10. Tabel Posisi_Terkini: merupakan tabel transaksional yang berfungsi untuk menyimpan posisi yang terbaru selama dalam proses perjalanan

11. Tabel History_Pelanggaran: merupakan tabel yang berfungsi untuk merekam pelanggaran yang telah dilakukan selama dalam perjalanan.

12. Tabel Lokasi: merupakan tabel master yang berfungsi untuk menyimpan posisi yang biasa dijadikan tempat asal dan tempat tujuan.

Secara konsep, tabel-tabel tersebut relasinya dapat digambarkan sebagai berikut:

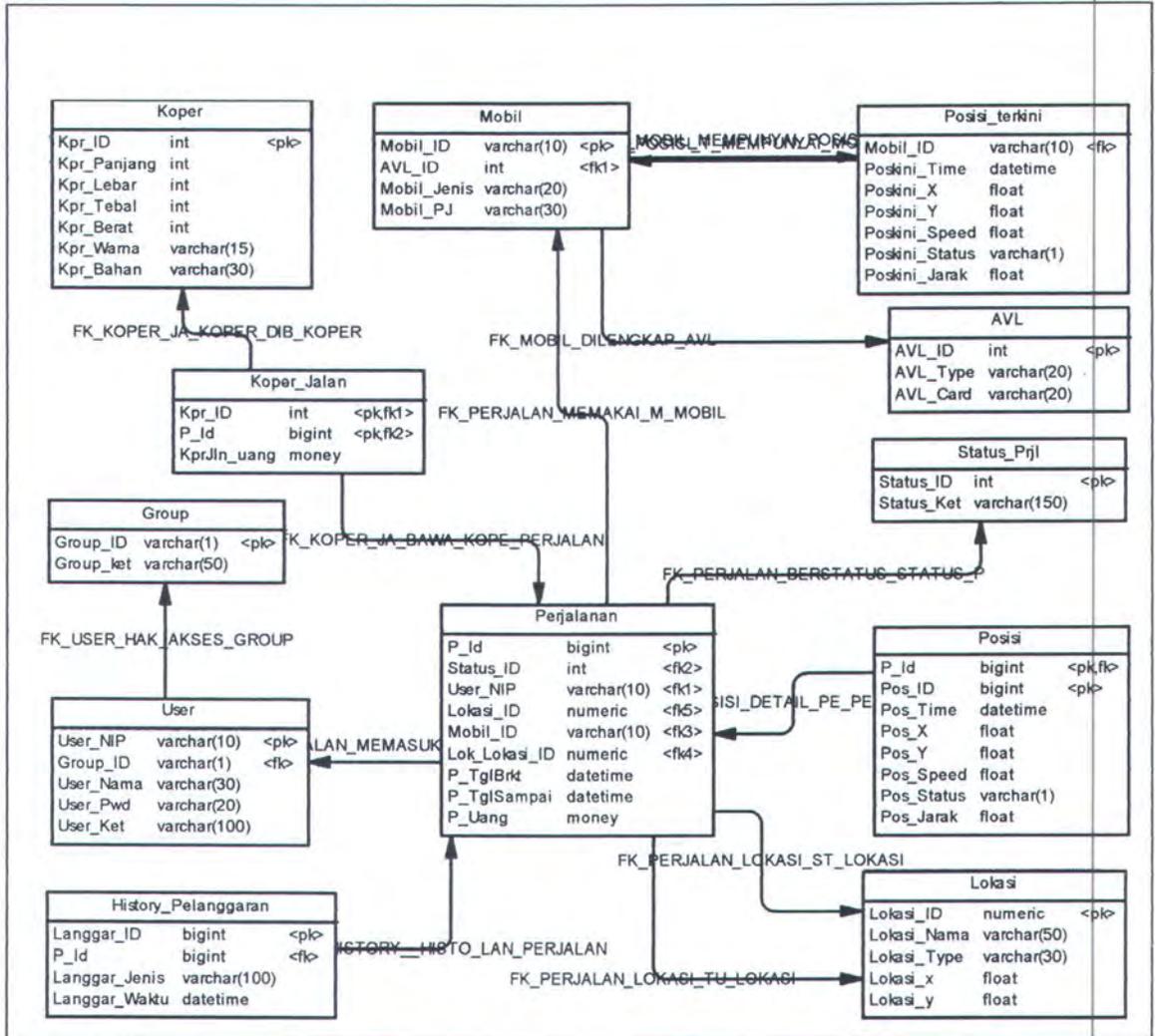


Gambar 4.13 Diagram Konsep Data

Entitas – entitas yang terdapat di dalam Diagram tersebut adalah :

7. Entitas Koper (Kpr_ID, Kpr_Panjang, Kpr_Lebar, Kpr_Tebal, Kpr_Berat, Kpr_Warna, Kpr_Bahan).
8. Entitas Mobil (Mobil_ID, Mobil_Jenis, Mobil_Pj)
9. Entitas AVL (AVL_ID, AVL_Type, AVL_Card)
10. Entitas Group (Group_ID, Group_Ket)
11. Entitas User (User_NIP, User_Nama, User_Pwd, User_ket)
12. Entitas Koper_Jalan (KprJln_Uang)
13. Entitas Status_Prjl (Status_ID, Status_Ket)
14. Entitas Perjalanan (P_ID, P_Start, P_Tujuan, P_TglBrkt, P_TglSampai, P_Uang)
15. Entitas Posisi (Pos_ID, Pos_Time, Pos_X, Pos_Y, Pos_Speed, Pos_Status, Pos_Jarak)
16. Entitas Posisi_Terkini (Poskini_Time, Poskini_X, Poskini_Y, Poskini_Speed, Poskini_Status, Poskini_Jarak)
17. Entitas History_Pelanggaran (Langgar_ID, Langgar_Jenis, Langgar_Waktu).

Rancangan konseptual tersebut kemudian dikonversi sehingga menghasilkan struktur fisik yang menggambarkan detail lengkap penyimpanan dan alur pengaksesan data.



Gambar 4.14 Diagram Fisik Data

Berdasarkan rancangan tersebut, maka dibentuklah tabel – tabel pada basisdata yang mempunyai bentukan seperti di bawah ini. Sebelumnya, perlu diketahui bahwa PK adalah *Primary Key*, FK adalah *Foreign Key*. Diagram dari rancangan struktur fisik tersebut digambarkan pada gambar berikut :

1. Tabel AVL



Tabel AVL merupakan table master yang berfungsi untuk menyimpan informasi mengenai spesifikasi dari alat AVL yang digunakan beserta nomer kartu SIM yang ditancapkan di dalamnya.

No	Name	Tipe data	Pengisi data	Keterangan
1	AVL_ID	Integer	Auto	Nomor seri dari AVL
2	AVL_Type	Varchar(20)	Admin	Jenis dari AVL berupa online dan offline
3	AVL_Card	Varchar(20)	Admin	Nomor kartu SIM yang dipasang pada AVL

Tabel 4.3 Tabel Group

2. Tabel Group

Tabel group merupakan tabel master yang berfungsi untuk menyimpan kategori group dari user yang akan menentukan hak akses yang dimiliki oleh user

No	Name	Tipe data	Pengisi data	Keterangan
1	Group_ID	Varchar(1)	Auto	ID dari group
2	Group_ket	Varchar(50)	Admin	Keterangan dari masing-masing group

Tabel 4.4 Tabel Group

3. Tabel Koper

Tabel koper merupakan tabel master yang berfungsi untuk menyimpan informasi mengenai detail fisik koper yang digunakan dalam pengiriman.

No	Name	Tipe data	Pengisi data	Keterangan
1	Kpr_ID	Integer	Auto	ID Koper
2	Kpr_Panjang	Integer	Admin	Panjang Koper
3	Kpr_Lebar	Integer	Admin	Lebar Koper
4	Kpr_Tebal	Integer	Admin	Tebal Koper
5	Kpr_Berat	Integer	Admin	Berat Koper
6	Kpr_Warna	Varchar(50)	Admin	Warna Koper
7	Kpr_Bahan	Varchar(50)	Admin	Bahan Koper

Tabel 4.5 Tabel Koper

4. Tabel Koper_Jalan

Tabel koper_jalan merupakan tabel transaksional yang menyimpan informasi mengenai jenis koper, mobil yang digunakan dalam proses pengiriman uang. Di samping itu juga menyimpan informasi detail dari kapasitas uang masing-masing koper.

No	Name	Tipe data	Pengisi data	Keterangan
1	Kpr_ID	Integer	Operator	ID Koper
2	Mobil_ID	Varchar(10)	Operator	ID Mobil
3	P_Id	Integer	Operator	ID Pengiriman
4	KprJln_uang	Money	Operator	Jumlah uang pada masing-masing koper

Tabel 4.6 Tabel Koper_Jalan

5. Tabel Mobil

Tabel mobil merupakan tabel master yang berfungsi untuk menyimpan informasi mengenai jenis mobil yang digunakan dalam proses pengiriman beserta orang yang bertanggung jawab terhadap mobil tersebut.

No	Name	Tipe data	Pengisi data	Keterangan
1	Mobil_ID	Varchar(10)	Auto	ID Mobil
2	AVL_ID	Integer	Admin	ID AVL
3	Mobil_Jenis	Varchar(20)	Admin	Jenis Mobil
4	Mobil_PJ	Varchar(30)	Admin	Orang yang bertanggung jawab pada mobil tersebut /pengemudi

Tabel 4.7 Tabel Mobil

6. Tabel Perjalanan

Tabel perjalanan merupakan tabel transaksional yang berfungsi untuk menyimpan setiap data perjalanan pengiriman. Informasi ini mencakup tanggal pemberangkatan sampai tanggal sampai. Kemudian start pemberangkatan dan tujuan yang ingin ditempuh. Dari sini juga nantinya akan dapat diketahui status dari perjalanan sebagai kesimpulan akhirnya. Apakah pengiriman berjalan sukses

atau ada hambatan. Hal ini bisa dilihat dari field `status_id`. Bilamana pada tabel `koper_jalan` terdapat jumlah uang yang dibawa pada masing-masing koper maka pada tabel ini terdapat jumlah keseluruhan uang yang dikirim yang tersimpan dalam field `p_uang`.

No	Name	Tipe data	Pengisi data	Keterangan
1	P_Id	Bigint	Auto	ID Pengiriman
2	User_NIP	Varchar(10)	Auto	NIP user yang memasukkan data
3	Status_ID	Integer	Operator	Status pengiriman, apakah sudah selesai atau belum
4	P_Start	Varchar(50)	Operator	Tempat pemberangkatan awal koper
5	P_Tujuan	Varchar(50)	Operator	Lokasi tujuan pengiriman koper
6	P_TglBrkt	Datetime	Auto	Tanggal mulai pemberangkatan
7	P_TglSampai	Datetime	Operator	Tanggal pengiriman telah sampai tujuan
8	P_Uang	Money	Auto	Jumlah uang total yang dibawa dalam pengiriman

Tabel 4.8 Tabel Perjalanan

7. Tabel Posisi

Tabel posisi merupakan tabel transaksional yang berfungsi untuk menyimpan posisi yang terbaru selama dalam proses perjalanan. Database akan terupdate secara otomatis berdasarkan data yang dikirim oleh alat AVL. Data yang nantinya digunakan untuk proses analisis adalah data tentang posisi, kecepatan dan status. Status yang dimaksud dalam tabel ini adalah untuk menyimpan kondisi dari alat, apakah dalam keadaan baik atukah ada yang berusaha membuka atau merusak dengan sengaja. Data ini berguna untuk mengecek keadaan darurat.

No	Name	Tipe data	Pengisi data	Keterangan
1	P_Id	Bigint	Auto	ID pengiriman
2	Pos_ID	Bigint	Auto	ID dari posisi koordinat koper
3	Pos_Time	Datetime	Auto	Waktu yang menunjukkan posisi sekarang
4	Pos_X	Float	Auto	Posisi koordinat X yang sekarang
5	Pos_Y	Float	Auto	Posisi koordinat Y yang sekarang
6	Pos_Speed	Float	Auto	Kecepatan koper sekarang ini
7	Pos_Status	Varchar(1)	Auto	Status sekarang ini
8	Pos_Jarak	Float	Auto	Jarak yang telah ditempuh

Tabel 4.9 Tabel Posisi

8. Tabel Status

Tabel status merupakan tabel master yang berfungsi untuk menyimpan macam-macam status perjalanan. Sebagai gambaran, status perjalanan bias dibagi menjadi 3 yaitu: sampai dan sukses, sampai ada hambatan, gagal.

No	Name	Tipe data	Pengisi data	Keterangan
1	Status_ID	Integer	Auto	ID Status
2	Status_Ket	Varchar(150)	Admin	Keterangan status pengiriman

Tabel 4.10 Tabel Status

9. Tabel User

Tabel user merupakan tabel master yang berfungsi untuk menyimpan data-data user yang menggunakan sistem ini. ID dari user ini akan disimpan untuk menjaga keamanan.

No	Name	Tipe data	Pengisi data	Keterangan
1	User_NIP	Varchar(10)	Admin	NIP dari user
2	Group_ID	Varchar(1)	Admin	ID Group user
3	User_Nama	Varchar(30)	Admin	Nama user
4	User_Pwd	Varchar(20)	Admin	Password user
5	User_Ket	Varchar(100)	Admin	Keterangan user

Tabel 4.11 Tabel User

10. Tabel Posisi_Terkini

Tabel Posisi_Terkini merupakan tabel transaksional yang berfungsi untuk menyimpan posisi yang terbaru selama dalam proses perjalanan

No	Name	Tipe data	Pengisi data	Keterangan
1	Poskini_ID	Bigint	Auto	ID dari posisi koordinat koper
2	Poskini_Time	Datetime	Auto	Waktu yang menunjukkan posisi sekarang
3	Poskini_X	Float	Auto	Posisi koordinat X yang sekarang
4	Poskini_Y	Float	Auto	Posisi koordinat Y yang sekarang
5	Poskini_Speed	Float	Auto	Kecepatan koper sekarang ini
6	Poskini_Status	Varchar(1)	Auto	Status sekarang ini
7	Poskini_Jarak	Float	Auto	Jarak yang telah ditempuh

Tabel 4.12 Tabel Posisi_Terkini

11. Tabel History_Pelanggaran

Tabel History_Pelanggaran merupakan tabel yang berfungsi untuk merekam pelanggaran yang telah dilakukan selama dalam perjalanan.

No	Name	Tipe data	Pengisi data	Keterangan
1	P_Id	Bigint	Auto	ID Perjalanan
2	Langgar_ID	Bigint	Admin	ID Pelanggaran
3	Langgar_Jenis	Varchar(100)	Admin	Jenis Pelanggaran
4	Langgar_Waktu	Datetime	Admin	Waktu Pelanggaran

Tabel 4.13 Tabel History_Pelanggaran

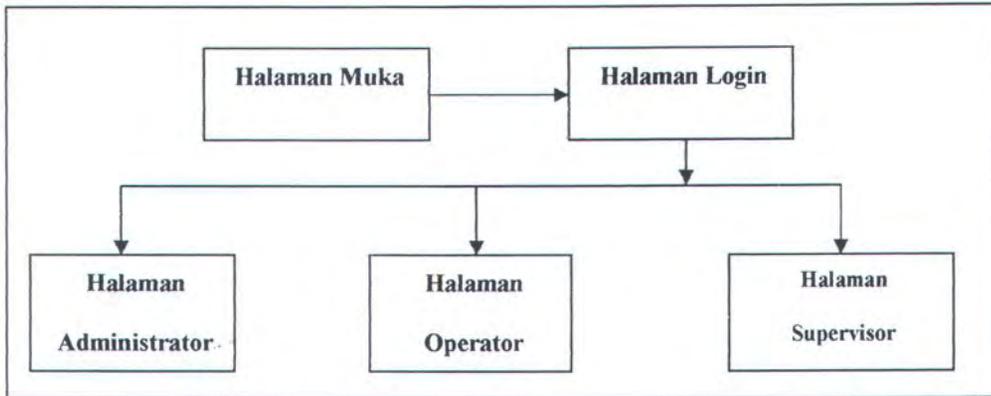
4.5. Perancangan Antarmuka

Perancangan antar muka dibutuhkan untuk mempermudah penggambaran pengaksesan data pada sistem yang akan dibuat. Sesuai dengan kebutuhan dari aplikasi, terdapat empat tipe halaman yang akan dibuat, yaitu :

1. Informasi tertulis. Halaman ini digunakan untuk menampilkan informasi dari data yang telah dimasukkan oleh pengguna dalam bentuk tulisan. Data ditampilkan dalam bentuk *tabular* (tabel dengan satu *record* ditampilkan memanjang ke samping kanan) dan *columnar* (data ditampilkan secara rinci). Selain itu data juga dapat ditampilkan dalam bentuk format laporan yang sesungguhnya untuk dapat langsung dicetak.
2. Informasi Peta. Halaman ini berfungsi untuk menampilkan informasi dalam bentuk peta. Peta yang ditampilkan adalah peta ITS yang dilengkapi dengan proses pelacakan.
3. Tambah Data. Halaman ini digunakan untuk mengisi data baru.
4. Ubah Data. Halaman ini digunakan untuk mengubah data yang telah ada sebelumnya.

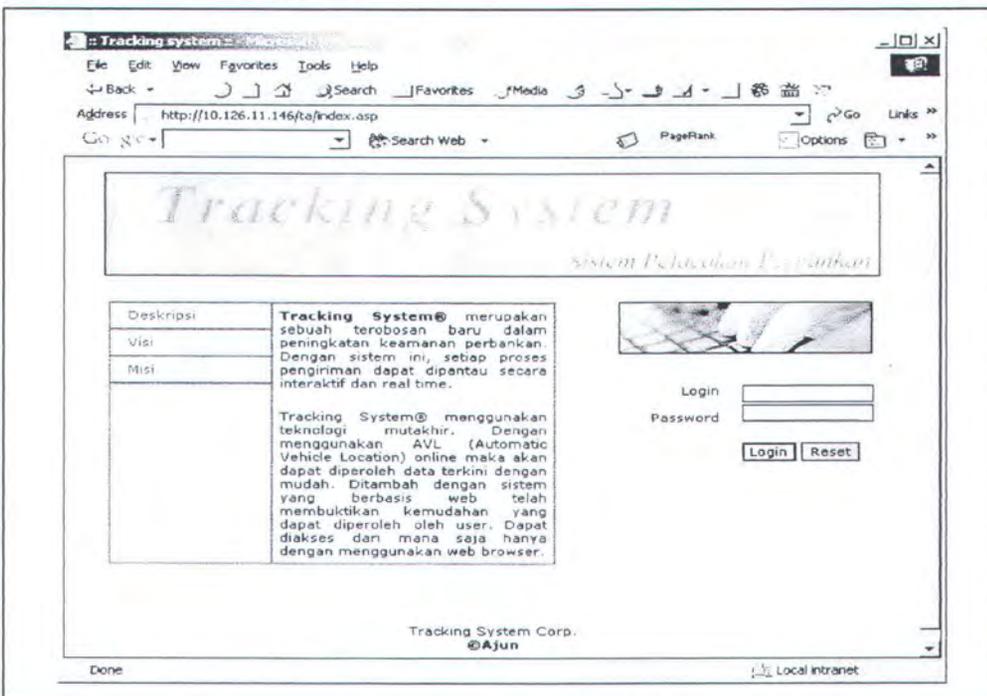
Seperti yang telah dijelaskan sebelumnya, pengguna dari aplikasi ini terdiri dari empat *role*, yaitu Administrator, Operator dan Manajer dan Asisten Manajer. Administrator mempunyai hak akses menyeluruh terhadap data dalam aplikasi termasuk mengatur hak pengguna lainnya. Operator memiliki hak akses mengubah dan menambah data transaksional semisal memasukkan data perjalanan. Manajer memiliki hak akses untuk melihat rekapitulasi perjalanan pengiriman. Ia tidak dapat menambah atau mengubah data. Sedangkan Asisten Manajer dapat mengupdate status dari perjalanan/

Secara umum, gambaran navigasi atau menu utama aplikasi adalah seperti gambar berikut ini



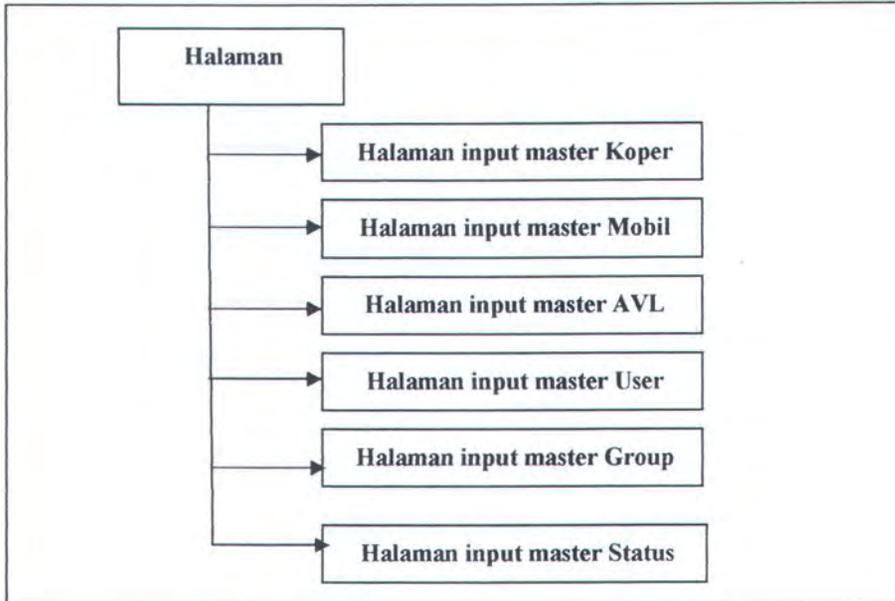
Gambar 4.12 Menu Utama Aplikasi

Halaman Muka dapat diakses oleh semua pengguna. Pada halaman ini akan ditampilkan informasi umum tentang proses pelacakan. Halaman Login merupakan halaman bagi pengguna untuk mengakses menu-menu khusus yang sesuai dengan hak akses yang dimiliki.



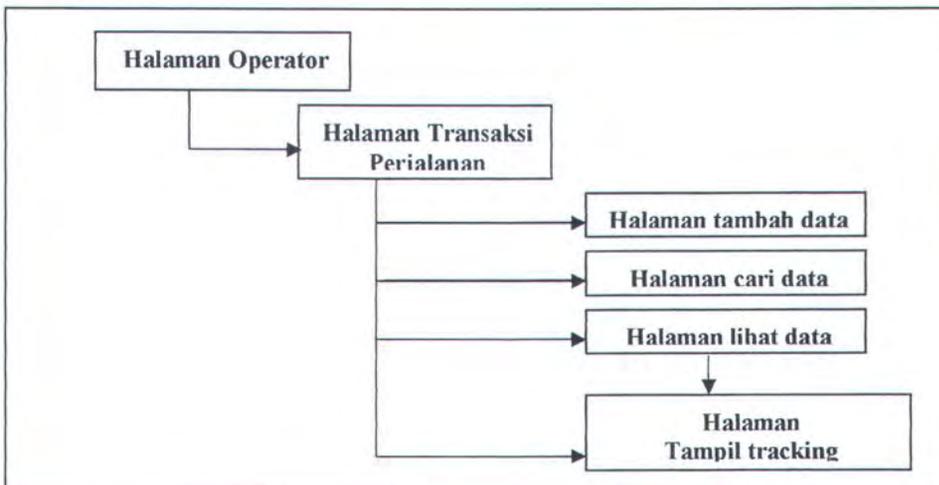
Gambar 4.15. Contoh Halaman Muka

Halaman Administrator hanya dapat diakses oleh pengguna dengan *role* Administrator. Ditampilkan daftar form inputan untuk tabel master.



Gambar 4.16. Menu Administrator

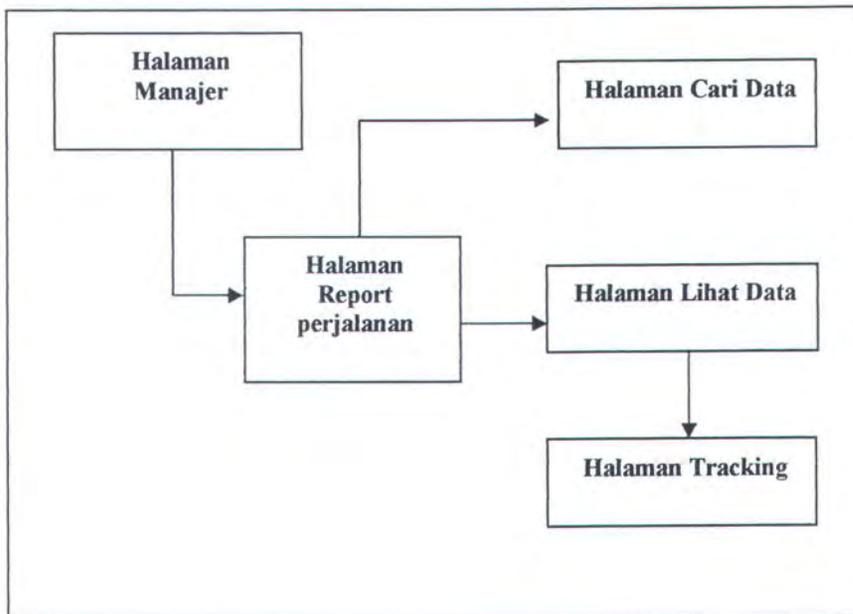
Halaman Operator hanya dapat diakses oleh pengguna dengan *role* Operator. Gambaran alur pengaksesan Halaman Petugas Dinkes dapat dilihat pada gambar berikut :



Gambar 4.17. Alur Pengaksesan Halaman Operator

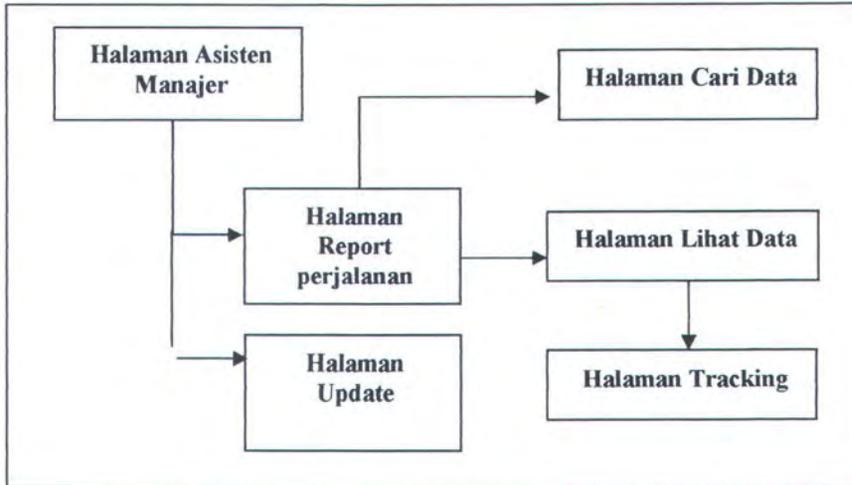
Pada alur diatas, untuk Halaman Transaksi Perjalanan, operator dapat memasukkan data yang kemudian disimpan ke dalam database. Sedangkan untuk membuat laporan, operator bisa menggunakan halaman lihat data untuk bisa melihat data secara lengkap, maupun dengan melakukan *query*. Selain itu, pada halaman lihat data diberikan fasilitas untuk melihat data dalam bentuk laporan – laporan yang diperlukan untuk kemudian dicetak.

Halaman Manajer hanya dapat diakses oleh pengguna dengan *role* Manajer memiliki alur pengaksesan sebagai berikut:



Gambar 4.18. Alur Pengaksesan Halaman Manajer.

Halaman Asisten Manajer hanya dapat diakses oleh pengguna dengan *role* Manajer memiliki alur pengaksesan sebagai berikut:



Gambar 4.19. Alur Pengaksesan Halaman Asisten Manajer.

Pada halaman update perjalanan untuk asisten manajer ini terdapat menu untuk update status dari perjalanan. Halaman ini berfungsi untuk memberikan status pada perjalanan, apakah statusnya sukses atau terdapat kendala.



BAB V
IMPLEMENTASI PERANGKAT LUNAK

MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

BAB V

IMPLEMENTASI PERANGKAT LUNAK

Dengan mengacu pada perancangan sistem pada bab 4, kini dilakukan pembuatan aplikasi berdasarkan rancangan tersebut. Pembuatan perangkat lunak atau aplikasi berbasis web ini dibagi menjadi 4, yaitu :

1. Pembuatan dan pengolahan peta

Dasar dari perangkat lunak ini adalah tersedianya peta digital yang benar dan data-datanya lengkap. Pembuatan dan pengolahan peta menggunakan software MapInfo Professional v.7.

2. Pembuatan database transaksional dan data nonspasial.

Selain data-data spasial yang sudah terkandung dalam peta, maka dibutuhkan juga database untuk menampung data-data transaksional dan data-data non spasial. Untuk database ini diimplementasikan dengan menggunakan MS SQL Server 2000.

3. Pembuatan aplikasi server untuk simulasi online.

Dalam perancangan disebutkan bahwa AVL yang digunakan menggunakan AVL online yang dapat mengirimkan data secara real time. Namun terdapat kendala yang menyebabkan tidak diperolehnya AVL yang online dan akhirnya pelaksanaan di lapangan hanya menggunakan AVL yang offline. Untuk itu dibuat aplikasi server yang akan mensimulasikan penerimaan data secara online. Dalam pembuatan aplikasi server untuk mengolah data ini diimplementasikan dengan menggunakan pemrograman Visual Basic 6, antara

lain untuk memparser data dari database perangkat lunak AVL ke database yang diperlukan untuk proses pelacakan pengiriman kotak uang.

4. Pembuatan aplikasi pelacakan posisi berbasis web

Dalam pembuatan aplikasi pelacakan posisi berbasis web ini diimplementasikan dengan menggunakan pemrograman script ASP dan juga menggunakan komponen untuk mempermudah dalam pemrograman yaitu ASPMap.

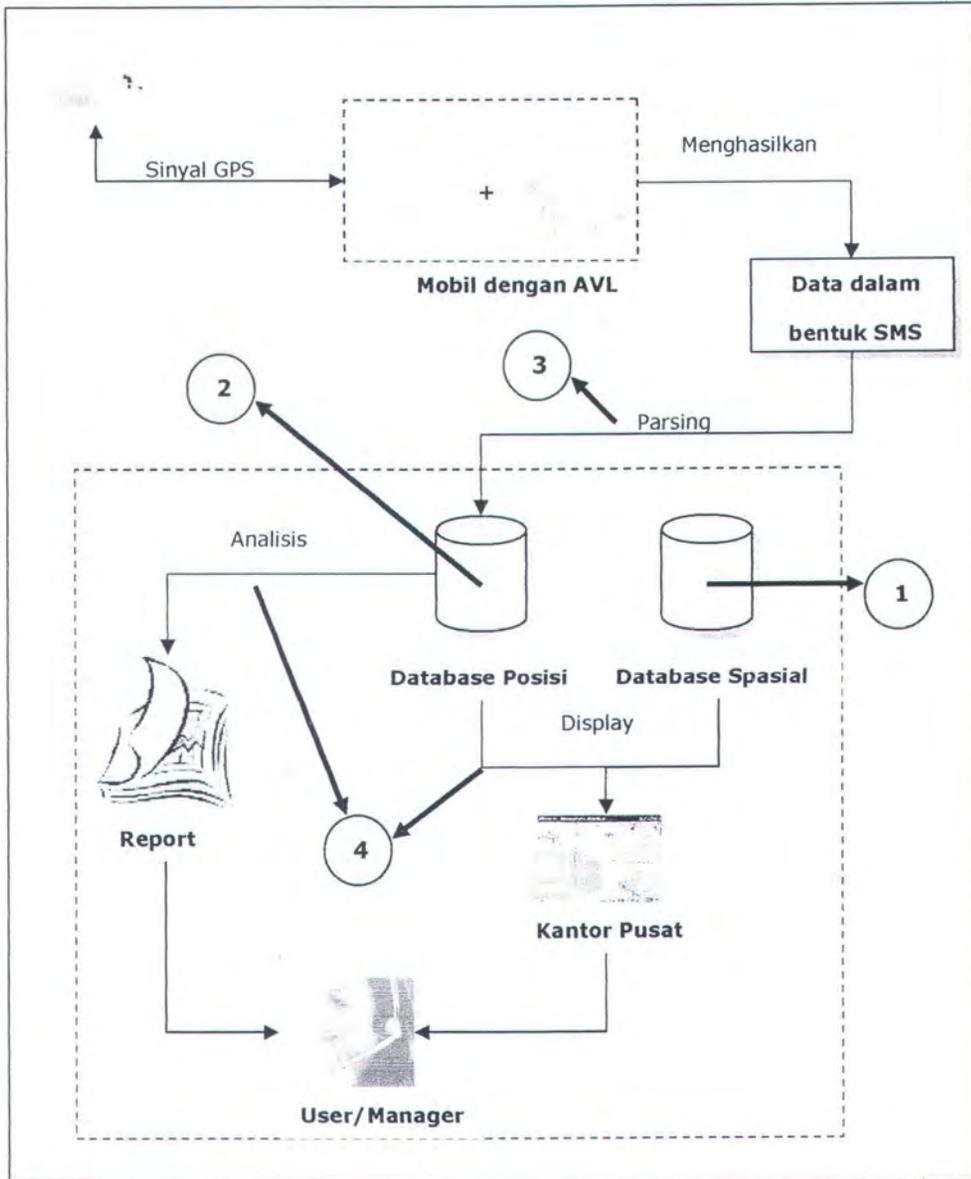
Keempat hal yang disebutkan di atas saling berkaitan satu sama lain. Sesuai dengan tujuan Tugas Akhir ini yaitu membuat sebuah perangkat lunak pelacakan pengiriman kotak uang dengan menggunakan GIS maka pada point yang pertama yaitu tersedianya peta yang sesuai, menjadi landasan yang utama. Peta sebagai salah satu komponen dari Sistem Informasi Geografis memegang peranan penting untuk memberikan informasi lokasi yang diinginkan selama dalam proses pelacakan.

Pada point kedua yaitu pembuatan database transaksi dan data nonspasial mendukung sistem pelacakan ini. Pembuatan database transaksi berguna untuk merekam semua kegiatan transaksi yang telah digunakan. Dengan begitu akan terdapat arsip perjalanan sebagai bukti pelaksanaan kegiatan pengiriman yang dapat dipertanggungjawabkan.

Dengan adanya data spasial dan data nonspasial beserta data transaksi, ketiga hal tersebut kemudian dikombinasikan dan ditampilkan ke dalam web dengan pembuatan aplikasi pelacakan berbasis web. Di dalam aplikasi berbasis web

tersebut peta ditampilkan secara interaktif kemudian titik-titik posisi yang di dapat dari AVL ditampilkan secara periodik.

Berikut gambaran dari ke-4 hal dalam sistem secara keseluruhan:

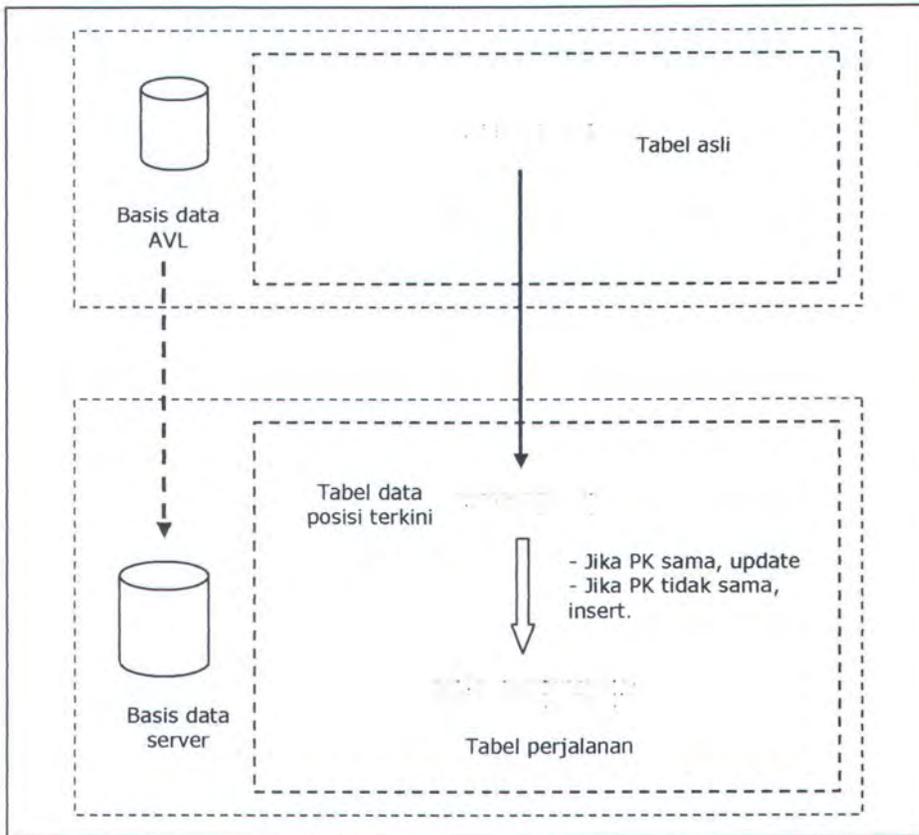


Gambar 5.1. Integrasi bagian-bagian implementasi

Secara khusus pada bagian ke-3 yaitu pembuatan aplikasi server akan dijelaskan lebih lanjut. Pembuatan aplikasi ini bertujuan untuk mensimulasikan penerimaan data dari AVL secara online. Hal ini didasari karena seharusnya sistem memakai AVL yang online tetapi pada kenyataan di lapangan, karena ketidaktersediaan AVL yang online maka terpaksa digunakan AVL jenis yang offline.

Dengan menggunakan AVL offline, data hanya bisa di dapatkan ketika data yang tersimpan dalam AVL didownload. Hasil dari download tersebut berupa data-data posisi, arah, kecepatan dalam bentuk Microsoft Access 2000, sedangkan untuk database server menggunakan Microsoft SQL Server 2000. Aplikasi Server yang akan melakukan simulasi data-data yang diambil dari database AVL Offline, yakni untuk disimpan ke tabel rekaman perjalanan dan posisi terkini perjalanan.

Cara kerja dari aplikasi server tersebut adalah memindahkan data-data posisi dari database AVL dalam MS Access 2000 ke dalam database SQL Server 2000. Dalam data posisi dalam database AVL, masing-masing terdapat waktu yang menerangkan kapan hal kendaraan pada posisi tersebut. Dengan mengacu kepada waktu, maka tiap selang waktu tertentu yaitu dalam rentang 10 detik maka aplikasi server ini akan memindahkan data-data pada 2 tabel di database server. Tabel yang pertama adalah tabel data posisi terkini. Dalam tabel tersebut akan dicek, apabila pada awal ternyata kosong, maka akan dilakukan operasi insert. Tetapi bilamana ternyata sudah ada data dengan id kendaraan dan id perjalanan yang sama maka akan dilakukan operasi update.



Gambar 5.2. Cara Kerja Aplikasi Server

Data-data pada tabel posisi terkini hanya menyimpan satu titik saja yaitu titik terkini. Selanjutnya data-data tersebut dipindahkan ke table perjalanan yang merupakan history dari semua perjalanan yang pernah dilakukan. Data-data ini akan terus disimpan dan digunakan sebagai arsip.

5.1. Pembuatan Matriks CRUD

Aplikasi yang dibuat tidak hanya digunakan oleh 1 macam pengguna saja tetapi bisa bermacam-macam orang dengan kebutuhan yang berbeda pula. Hal tersebut bila dilihat secara horizontal. Bilamana dilihat secara vertikal, masing-masing pengguna memiliki jabatan dan tanggung jawab yang berbeda.

Dengan memperhatikan hal tersebut maka perlu diadakan pemilahan-pemilahan batasan kerja dari masing-masing kelompok sehingga tugas dan wewenangnya dalam sistem tidak saling tumpang tindih. Hal ini nantinya juga berimbas pada keamanan dalam sistem itu sendiri, dimana seharusnya hanya orang yang berhak yang bisa menambah atau merubahnya.

Hubungan antara menu-menu yang akan dibuat dengan hak akses dari masing-masing grup dapat digambarkan ke dalam sebuah matriks yang biasa disebut matriks CRUD. Dengan demikian akan didapatkan gambaran global dari menu-menu yang disediakan dan siapa saja yang bisa mengaksesnya. Perlu dijelaskan kembali bahwa kategori grup adalah sebagai berikut:

1. Administrator: administrator bertanggung jawab terhadap keseluruhan sistem. Tugas yang utama adalah manajemen tabel-tabel master. Hak yang dimiliki merupakan tingkatan tertinggi.
2. Manajer: manajer adalah orang yang tertinggi dalam organisasi. Dia yang paling bertanggung jawab apabila sesuatu menimpa proses pengiriman. Tugas dan wewenangnya adalah memonitor segala kegiatan yang dilakukan dalam sistem tetapi dia tidak bisa merubah data-data yang telah dimasukkan.
3. Asisten Manager: asisten manajer merupakan orang kedua setelah manajer. Asisten bertugas mengawasi secara langsung kegiatan di lapangan termasuk yang menentukan dan memasukkan status dari sebuah proses pengiriman.

4. Operator: operator adalah orang yang bertugas untuk memasukkan data-data transaksi ke dalam database. Operator hanya bertanggung jawab dalam pemasukan data transaksi dan pembuatan laporan-laporan.

Berikut adalah gambaran dari matriks CRUD yang merupakan hubungan antara menu yang dibuat dan hak akses dari user:

Menu	Administrator	Manajer	Ass. Manajer	Operator
1. Administrasi Data Master				
a. Data Master Koper	CUD			
b. Data Master Lokasi	CUD			
c. Data Master Group	CUD			
d. Data Master AVL	CUD			
e. Data Master Status	CUD			
f. Data Master Mobil	CUD			
g. Data Master User	CUD			
2. Transaksi Perjalanan		R	RU	C
3. Status Perjalanan		RU	U	R
4. Laporan				
a. Laporan Data Perjalanan		R	C	C
b. Laporan Data Perjalanan per Status		R	C	C
5. Lihat Peta		R	R	R
6. Lihat Posisi Terkini		R	R	R

Tabel 5.1 Matriks CRUD

Keterangan :

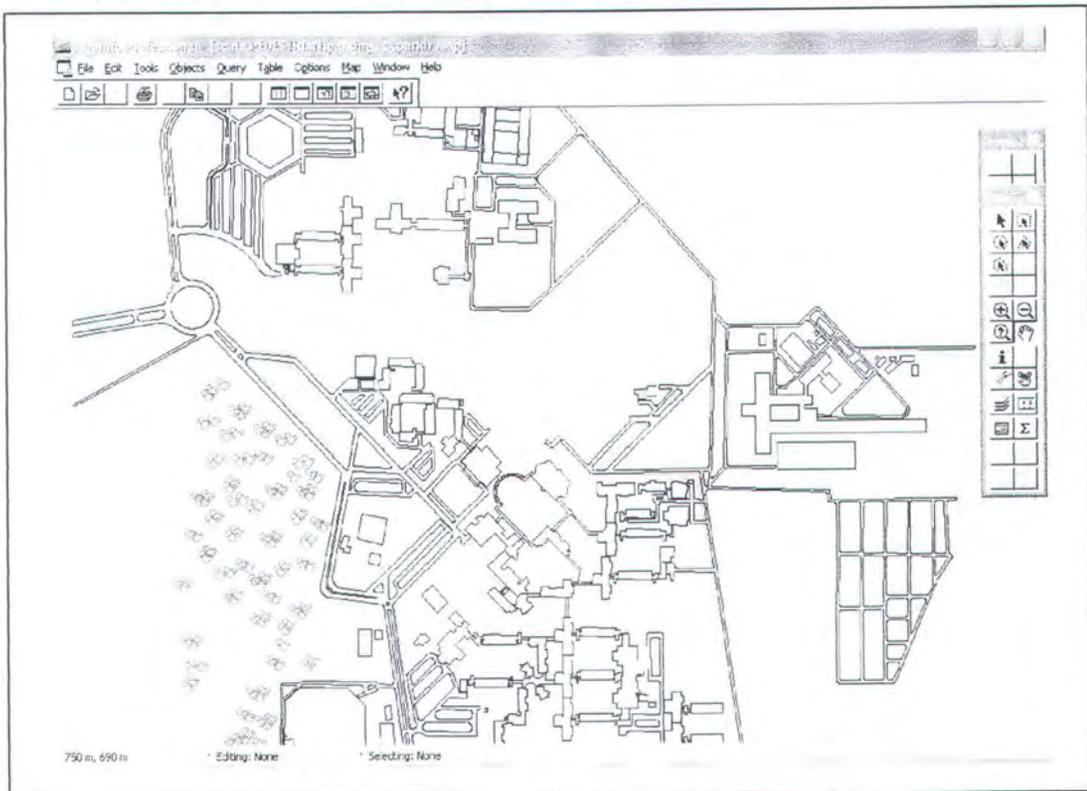
- C = create : user berhak untuk menambahkan item baru
- R = read : user hanya berhak membaca atau melihat.
- U = update : user berhak mengubah item yang telah ada tapi tidak dapat membuat item yang baru

- D = delete : user berhak menghapus item yang sudah ada.

5.2. Pembuatan dan Pengolahan Peta

Dalam pembuatan aplikasi ini dibutuhkan peta digital yang mempunyai data yang lengkap. Terutama data mengenai jalan dan gedung. Dalam simulasi kali ini menggunakan peta ITS. Untuk pengolahan peta tersedia berbagai macam pilihan software seperti : ArcView, MapInfo, Map Point.

Dalam implementasi ini digunakan software MapInfo Profesional v.7. Sumber peta ITS yang digunakan berasal dari Badan Perencanaan ITS. Berikut adalah tampilan peta dengan menggunakan MapInfo :



Gambar 5.3 Peta ITS

Hal-hal yang harus diperhatikan :

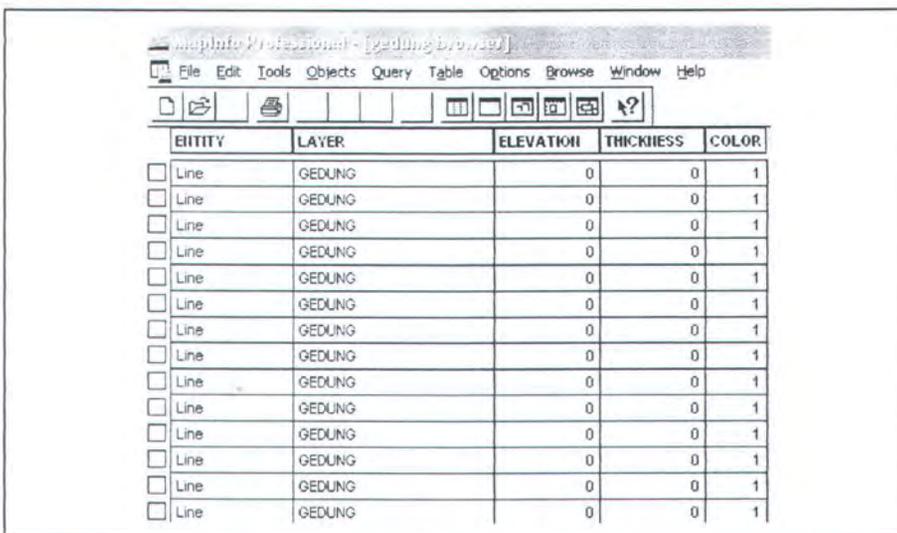
1. Mengeset proyeksi peta

Dalam aplikasi tracking nantinya dibutuhkan koordinat X, Y. Untuk mendapatkan koordinat bisa bermacam-macam tetapi jenis proyeksi yang digunakan sangat mempengaruhi akan data posisi.

Dalam implementasi ini dipakai proyeksi UTM

2. Melengkapi database

Peta digital mempunyai database. Pada MapInfo, database tersebut disimpan pada file yang berekstensi *.dat. Berikut adalah salah satu contoh bentuk database :



ENTITY	LAYER	ELEVATION	THICKNESS	COLOR
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1
<input type="checkbox"/> Line	GEDUNG	0	0	1

Gambar 5.4 Tabel data spasial

Agar bisa tampil dengan baik, maka data harus dilengkapi sehingga bila nanti ditampilkan ke dalam web, hasilnya pun bisa lebih lengkap. Penambahan data bisa dilakukan dengan klik pada peta atau dengan cara langsung mengisi pada tabel seperti di atas.

5.3. Pembuatan Koneksi Basis Data

Koneksi ke basis data dibutuhkan setiap kali aplikasi ingin mengelola data, baik menyimpan maupun mengambil data ke dan dari basis data. Untuk itu, pada setiap halaman web yang membutuhkan akses ke basis data dengan meng-include-kan sebuah file yang bernama conn.asp. File tersebut berisikan kode untuk melakukan koneksi dengan basis data.

Prosedur koneksi ke basis data adalah sebagai berikut:

1. Mendefinisikan Driver koneksi ke SQL Server dengan variable strConn ke Database:tracking yang berada pada Database Server di LP-46
`strConn <-- "DRIVER<--{SQL Server};SERVER<--lp-46; DATABASE<--tracking; UID<--sa;PWD<--"`
2. Set object koneksi dengan menggunakan ADODB Connection
`Set oConn <-- Server.CreateObject("ADODB.Connection")`
3. Buka koneksi ke database server
`oConn.Open strConn`
4. Mendefinisikan dan set Object ADODB Recordset
`Set oRec <-- Server.CreateObject("ADODB.Recordset")`

Script tersebut melakukan membuat koneksi yang menghubungkan dengan MS SQL Server 2000 pada database 'Tracking' dengan user ID= sa. ID 'sa' merupakan tingkatan tertinggi dalam database tersebut.

5.4. Pembuatan Fasilitas Login

Untuk bisa mengakses menu-menu yang telah dibuat, maka user harus login terlebih dahulu. Login merupakan representasi dari level hak akses dari user yang bersangkutan seperti diterangkan pada subbab di atas. Hal ini diperlihatkan pada script pada file login_auth.asp berikut ini.

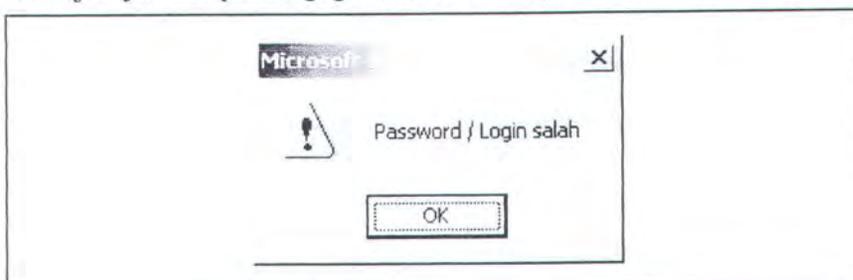
```
<%
- Clear Buffer Cache untuk menampung hasil session user dan set ke True value
  Response.Buffer <-- True
- Definiskan variabel untuk koneksi, object recordset dan string query
  Dim strConn, objConn, objRec, strQuery
- Masukkan file yang sebagai referensi untuk koneksi aplikasi dengan database
  <!--#include file<--"include/conn.asp"-->
- Set Object koneksi ke database
  Set objConn <-- Server.CreateObject("ADODB.Connection")
- Buka koneksi dengan database
```

```

objConn.Open strConn
- Tampung nilai dari form untuk nama login dan password dari user
  user <--- Request.form("login2")
  passwd <--- Request.form("passwd")
- Definiskan SQL string untuk melakukan querying user dan password
  strQuery <--- "select * from [user] where user_nama<---"& user &" and user_pwd<---
  "& passwd &"
- Eksekusi string query
  oRec.Open strQuery, oConn
- Check hasil query apakah berisi null value atau tidak
  If (oRec.BOF <--- True And oRec.EOF <--- True) Then
[+] Jika hasil querying adalah null maka window akan di-direct ke halaman utama
[+] Jika hasil querying ternyata terdapat record maka akan dilakuakn pencatatan
  session login dari user dan profil dari user yang diperlukan untuk otorisasi pada
  halaman selanjutnya.
  session("login")<---"true
  session("group")<---oRec("group_ID")
  session("userID")<---oRec("User_NIP")
  session("nama")<---oRec("user_nama")
  Response.Write "Password Match"
  group<---oRec("group_ID")
- Spesifikasikan hak akses halaman user berdasarkan group dari user
  Select Case (group)
  Case "1" :
    Response.redirect "admin_page.asp"
  Case "2" :
    Response.redirect "manager_page.asp"
  Case "3" :
    Response.redirect "asisten_page.asp"
  Case "4" :
    Response.redirect "operator_page.asp"end select
  end if
- Tutup session koneksi dengan database
  oRec.Close
  oConn.Close
  Set oRec <--- Nothing
  Set oConn <--- Nothing
%>

```

Yang dilakukan oleh script tersebut adalah melakukan koneksi ke tabel user dan mengambil nama user beserta passwordnya. Kemudian dicocokkan dengan user dan password yang diisikan oleh pengguna. Bila cocok maka akan diteruskan ke menu selanjutnya. Tetapi bila gagal akan muncul pesan salah seperti berikut:



Gambar 5.6. Pesan Gagal Login

Setelah muncul pesan gagal login, bila di klik OK maka akan kembali ke halaman login awal untuk mengulangi proses login. Hal ini merupakan proses yang tidak terbatas artinya bila gagal akan kembali lagi ke halaman login berapa kalipun hal itu dicoba. Tetapi bila login yang dimasukkan sudah benar maka akan langsung ke halaman selanjutnya.

Login dapat dilakukan pada halaman utama aplikasi, seperti ditunjukkan pada gambar 5.5.

Deskripsi	Tracking System® merupakan sebuah terobosan baru dalam peningkatan keamanan perbankan. Dengan sistem ini, setiap proses pengiriman dapat dipantau secara interaktif dan real time.
Visi	
Misi	

Tracking System® menggunakan teknologi mutakhir. Dengan menggunakan AVL (Automatic Vehicle Location) online maka akan dapat diperoleh data terkini dengan mudah. Ditambah dengan sistem yang berbasis web telah membuktikan kemudahan yang dapat diperoleh oleh user. Dapat diakses dari mana saja hanya dengan menggunakan web browser.

Tracking System Corp.
©Ajun

Gambar 5.5. Letak Form Login

Pemeriksaan sah atau tidaknya login pengguna dilakukan pada *file Login_auth.asp*. Dari file *login_auth.asp* diketahui bahwa pengguna akan masuk ke halaman sesuai dengan hak akses yang dimiliki. Untuk Administrator maka akan langsung masuk ke halaman khusus administrator. Begitu juga dengan pengguna yang memiliki hak akses lain. Masing-masing memiliki halaman dan

fungsi sendiri-sendiri. Hal ini merupakan salah satu bentuk pengamanan sistem untuk mencegah salah masuk dan mencegah kesalahan yang mungkin dilakukan oleh pengguna baik sengaja maupun tidak.

5.5. Pembuatan Fasilitas Administrasi Data Master

Data master merupakan kumpulan data-data yang dijadikan acuan. Data master cenderung untuk bersifat stabil. Pembuatan data master bertujuan untuk memudah user dalam aplikasi sehari-hari dan untuk mempercepat pemasukan data serta menghindari adanya kesalahan disebabkan salah tulis. User tinggal memilih data yang diinginkan yang sebelumnya sudah tersimpan dalam data master.

Menu-menu yang disediakan untuk mengolah data master berikut ini :Data master koper, Data master lokasi, Data master group, Data master AVL, Data master status, Data master mobil, Data master user.

Untuk lebih jelasnya akan diulas pada pengolahan data master koper. Berikut adalah gambaran dari halaman administrasi data master koper pada file master_koper.asp:

Data Master Koper							
ID	Panjang	Lebar	Tebal	Berat	Warna	Bahan	
<input type="checkbox"/>	1	40	40	40	0	kuning	hitam <input type="button" value="Edit"/>
<input type="checkbox"/>	2	12	22	10	0	hitam	plastik <input type="button" value="Edit"/>
[1] []							
<input type="button" value="Add"/> <input type="button" value="Delete"/>							

Gambar 5.7. Administrasi Data Master Koper

Pertama kali yang dilakukan adalah pengecekan hak akses user sehingga bila langsung ditulis di URL-nya tanpa login terlebih dahulu dapat dicegah

```

- Chek session login group user dan redirect ke halaman utama jika session telah habis
  If Session("group") <> "1" Then
    Response.Redirect "index.asp?cause=notlogin"
  End If

```

Kemudian ditampilkan list dari data yang sudah ada dalam database.

```

1. Definisikan beberapa variabel untuk menampung object koneksi, recordset, dan string
   query
   Dim strConn, objConn, objRec, strQuery
2. Set dan aktifkan object koneksi, serta recordset ke database
   Set objConn <-- Server.CreateObject("ADODB.Connection")
   objConn.Open strConn
   Set objRec <--Server.CreateObject("ADODB.Recordset")
3. Tampung nilai variable dari object request
   pg <-- Request.QueryString("pg")
4. Definisikan dan eksekusi string query untuk melakukan perhitungan terhadap jumlah koper
   dari database
   strQuery <-- "SELECT COUNT(Kpr_ID) As Jum FROM Koper"
   objRec.Open strQuery, objConn
5. Tampung nilai hasil query dari tabel koper
   jum <-- objRec("Jum")
6. Tampilkan seluruh record data hasil kueri ke web dalam bentuk tampilan tabel dan juga
   atur posisi perhalaman yang akan ditampilkan
   idx <-- Request.QueryString("idx")
   If(IsEmpty(idx) Or idx <-- "") Then
     idx <-- 0
   Else
     idx <-- CInt(idx)
   End If
   'Banyaknya record yang ditampilkan dalam satu halaman
   maxrecordperpage <-- 2
   'Banyaknya link halaman yang ditampilkan dalam satu halaman
   maxlinkperpage <-- 2
   maxlink <-- jum \ maxrecordperpage
   If(jum Mod maxrecordperpage > 0) Then maxlink <-- maxlink + 1
   strQuery <-- "SELECT * FROM Koper"
   Set objRec <-- objConn.Execute(strQuery)

   If Not ((objRec.EOF <-- True And objRec.BOF <-- True) And idx ><-- maxlink) Then
     Response.Write "<p><div align<--center class<--putih size<--3><strong>Data Master
     Koper</strong></div></p>"

     counter <-- 0
     script_form <-- ""
     arr_id <-- ""
     objRec.Move (idx * maxrecordperpage)
     Do While Not (objRec.EOF OR counter <-- maxrecordperpage)
       Response.Write "<tr align<--center id<--"" & objRec("Kpr_id") & "">"
       Response.Write "<td>"

       kpr_id <-- objRec("Kpr_ID")
       kpr_panjang <-- objRec("Kpr_panjang")
       kpr_lebar <-- objRec("Kpr_lebar")
       kpr_tebal <-- objRec("Kpr_tebal")
       kpr_berat <-- objRec("Kpr_berat")
       kpr_warna <-- objRec("Kpr_warna")
       kpr_bahan <-- objRec("Kpr_bahan")

```

```



```

Macam-macam operasi yang bisa dilakukan pada data master adalah tambah data, edit data, dan hapus data. Prinsipnya pada tambah data dan edit data hampir sama hanya saja pada tambah data menambahkan data baru sedangkan edit data mengubah data yang telah ada.

Tambah Data Master Koper		Edit Data Master Koper	
ID	: <input type="text"/>	ID	: 1
Panjang	: <input type="text"/>	Panjang	: 40
Lebar	: <input type="text"/>	Lebar	: 40
Tebal	: <input type="text"/>	Tebal	: 40
Netto	: <input type="text"/>	Netto	: 0
Warna	: <input type="text"/>	Warna	: kuning
Bahan	: <input type="text"/>	Bahan	: hitam
<input type="button" value="Save"/> <input type="button" value="cancel"/>		<input type="button" value="Save"/> <input type="button" value="cancel"/>	

Form Tambah Data

Form Edit Data

Gambar 5.8 Form Tambah Data dan Edit Data

Selanjutnya hasil inputan form tersebut akan diolah oleh script pada file koperhandle.asp. Untuk perintah edit digunakan script berikut:

- Lakukan pengecekan pada tombol edit, jika user menekan tombol edit maka tampilkan variabel dari form yang telah diisi oleh user


```

      If cmd <-- "edit" Then
        id <-- Request.Form("idkoper")
        panjang <-- Request.Form("panjangkoper")
        lebar <-- Request.Form("lebarkoper")
        tebal <-- Request.Form("tebalkoper")
        berat <-- Request.Form("nettokoper")
        warna <-- Request.Form("warnakoper")
        bahan <-- Request.Form("bahankoper")
      
```
- Mendefinisikan string variable untuk melakukan update data pada tabel Koper dengan input dari user dan kemudian dilakukan pemrosesan atau eksekusi dari string SQL ke database


```

      strQuery <-- "UPDATE Koper SET Kpr_panjang <-- '" & panjang & "',Kpr_lebar <-- '" &
      lebar
      ",Kpr_tebal <-- '" & tebal & "',Kpr_berat <-- '" & berat & ',Kpr_warna <-- '" & warna &
      "',Kpr_bahan <-- '" & bahan & "' WHERE Kpr_id <-- '" & id & "'"
      oConn.Execute(strQuery)
      
```
- Tutup session koneksi setelah proses update data selesai dilaksanakan


```

      oConn.Close
      Set oConn <-- Nothing
      Set oRec <-- Nothing
      
```
- Redirect ke halaman master_koper.asp setelah seluruh proses update data koper selesai dilaksanakan.


```

      Response.Redirect "master_koper.asp"
      
```



Pada perintah edit, hasil isian ditampung semua dalam variabel. Setelah itu baru diupdate ke dalam tabel master koper pada data yang mempunyai id yang sama.

Sedangkan untuk penambahan data menggunakan script berikut ini:

```

5. Jika tombol yang ditekan oleh user adalah add, maka akan dilakukan proses Insert data baru
pada tabel Koper.
  ElseIf cmd <-- "add" Then
[-] Tampung beberapa variable dari form insert data koper
  id <-- Request.Form("idkoper")
  panjang <-- Request.Form("panjangkoper")
  lebar <-- Request.Form("lebarkoper")
  tebal <-- Request.Form("tebalkoper")
  berat <-- Request.Form("beratkoper")
  warna <-- Request.Form("warnakoper")
  bahan <-- Request.Form("bahankoper")
[-] Mendefisikan string variable untuk melakukan insert data baru pada tabel Koper dengan
input dari user dan kemudian dilakukan pemrosesan atau eksekusi dari string SQL ke
database
  strQuery <-- "INSERT Koper(Kpr_id,Kpr_panjang,Kpr_lebar, Kpr_tebal, Kpr_berat,
Kpr_warna," & "Kpr_bahan) VALUES('" & id & "','" & panjang & "','" & lebar & "','" &
tebal & "','" & berat & "','" & warna & "','" & bahan & "')"
[-] Tutup session koneksi dengan database setelah proses insert data selesai dilaksanakan
  oConn.Execute(strQuery)
  oConn.Close
  Set oConn <-- Nothing
  Set oRec <-- Nothing
[-] Redirect ke halaman master_koper.asp setelah seluruh proses insert.
  Response.Redirect "master_koper.asp"

```

Pada penambahan data terlihat bahwa hasil isian form ditampung ke dalam variable kemudian baru ditambahkan ke dalam tabel master koper. Untuk perintah delete menggunakan script berikut ini:

```

6. Jika tombol yang ditekan user adalah tombol delete data, maka akan dilaksanakan proses
penghapusan data pada tabel koper.
  ElseIf cmd <-- "del" Then
[-] Tampung id koper yang dipilih oleh user dari form dan lakukan proses delete.
  For i <-- 1 To Request.Form("koper").Count
    strQuery <-- "DELETE koper WHERE Kpr_id <-- '" & Request.Form("koper")(i) & "'"
    oConn.Execute(strQuery)
  Next
[-] Tutup koneksi dengan database setelah proses delete selesai dilaksanakan.
  oConn.Close
  Set oConn <-- Nothing
[-] Redirect halaman asp ke halaman master_koper.asp setelah proses delete data selesai
dilaksanakan
  Response.Redirect "master_koper.asp"

```

Pada perintah hapus, digunakan id dari data yang diinginkan. Sebelumnya di cek dulu berapa data yang telah ditandai. Kemudian dengan perintah yang berulang, data yang id-nya dipilih akan dihapuskan dari tabel master koper.

5.6. Pembuatan Fasilitas Penambahan Data Transaksi Perjalanan

Halaman web penambahan data transaksi perjalanan dibuat sesuai dengan fungsinya, yaitu mempermudah pengguna berinteraksi dengan sistem. Penambahan data perjalanan dapat dilakukan dengan mengisikan data pada kotak teks, memilih dari kotak *combo*. Penambahan data perjalanan dilakukan sebelum proses pemberangkatan dilakukan dari tempat asal, dimana operator nantinya akan memasukkan tempat asal, tempat tujuan, kendaraan yang digunakan beserta detail koper pada masing-masing kendaraan. Pada tiap-tiap koper nantinya juga melekat atribut jumlah uang yang berada di dalamnya. Contoh tampilan halaman penambahan data ditunjukkan pada gambar 5.9.

The screenshot shows a web application window titled "Tambah Data Perjalanan". The form contains the following fields:

- Tanggal : 1/13/2004 7:22:04 AM
- Asal : [Pilih salah satu ▼]
- Tujuan : [Pilih salah satu ▼]
- Total Uang :

Below the form, there is a table with three columns: "Mobil", "Koper", and "Jumlah Uang". A "Tambah Mobil" button is located below the table. At the bottom of the form, there are "Save" and "cancel" buttons.

Gambar 5.9. Tampilan halaman Penambahan Data Perjalanan

Pada pengisian data perjalanan, tanggal pengiriman akan terisi secara otomatis mengacu ke tanggal hari ini dan jam sekarang. Untuk pengisian mengenai tempat asal pemberangkatan dan tempat tujuan, dapat memilih dari pilihan yang ada. Pilihan tersebut berasal dari tabel data master lokasi. Untuk menghindari kesalahan, maka digunakan javascript sehingga untuk tempat yang sudah diisikan pada item tempat asal, tidak akan keluar pada pilihan tempat tujuan.

Tambah Data Perjalanan

Tanggal : 1/13/2004 7:22:04 AM

Asal : BNI Kampus ▾

Tujuan : Pilih salah satu ▾

Total Uang :

Pilih salah satu
 Informatika
 Perkapalan
 Graha
 Rektorat

Gambar 5.10. Isian Tempat Asal dan Tempat Tujuan

Berikut untuk menangani pengecekan pengisian tempat asal dan tempat tujuan.

Pseudocodenya adalah:

- a. Ambil lokasi_id dan lokasi_nama dari tabel master lokasi
- b. Inisialisasi perjalanan(asal) yaitu mengecek lokasi_id yang telah dipilih
- c. Jalankan fungsi changeTujuan yang akan menghilangkan lokasi_id yang sudah dipilih dari list

Mendefinisikan fungsi changeTujuan() untuk mengambil id_lokasi dari tempat asal yang dipilih

```
function changeTujuan(){
  initPerjalanan(tujuan);
  var id <-- asal.options[asal.selectedIndex].value;
  for(i <-- tujuan.options.length-1; i ><-- 1; i--){
    if(id <--<-- tujuan.options[i].value){
      tujuan.options.remove(i);
    }
  }
}
```

d. Inisialisasi perjalanan(tujuan)

Setelah mengisikan tempat asal dan tempat tujuan pengiriman, maka kini mengisi kendaraan yang akan dipergunakan. Untuk memilih kendaraan juga sudah dipersiapkan pilihan yang diperoleh dari database

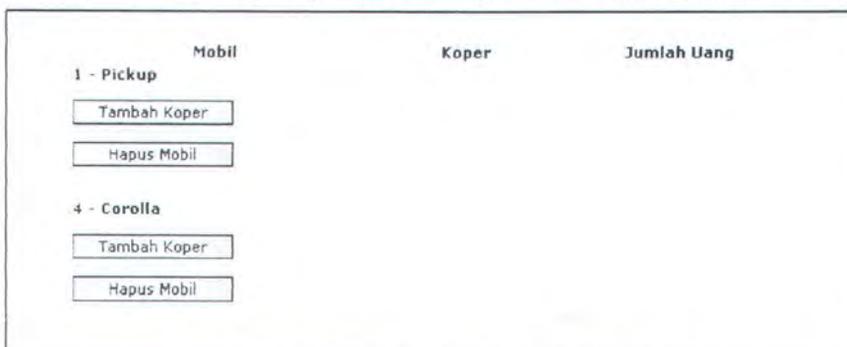


Gambar 5.11. Memilih Mobil

Pilihan mobil tersebut didapatkan dengan script berikut :

```
- isi nilai combobox pada index 0 dengan menu pilihan tipe mobil
  <option value<-- "" >Pilih salah satu </option>
- Isi nilai dari combo pada indeks berikutnya dengan data tipe mobil yang ada pada database
<%
Do While Not (objRec.EOF)
  mobil_jenis <-- objRec("mobil_jenis")
  mobil_id <-- objRec("mobil_id")
%>
<option value<--<%<--mobil_id %> class<--"teks"><%<-- mobil_id %> - <%<-- mobil_jenis
%>
```

Kemudian setelah di klik tombol pilih mobil, maka mobil yang telah dipilih akan tercatat dan kemudian tinggal memasukkan koper beserta uang di masing-masing koper. Bisa jadi dalam sebuah pengiriman digunakan 2 kendaraan atau lebih.



Gambar 5.12. Hasil Memilih Kendaraan

Pada pemilihan mobil ini, juga dilakukan pengecekan agar mobil tidak dimasukkan dalam daftar sebanyak 2 kali. Berikut script yang menangani penambahan mobil ini. Berikut pseudocode dari fungsi tambahMobil:

```

- Mendefinisikan fungsi tambahMobil()
function tambahMobil(){
    var idx <-- oMobil.selectedIndex;
    if(idx <--<-- 0)return;
    var opt <-- oMobil.options[idx];
    var oRow <-- oTable.insertRow();
    oRow.vAlign <-- "top";
    oRow.Align <-- "center";
    oRow.id <-- "row_" + opt.value;
    var oCell <-- oRow.insertCell();

- Buat sel baru dan masukkan mobil_id dan mobil_ket dilengkapi dengan tombol "hapus mobil"
  dan "tambah koper"
    oCell.innerHTML <-- "<b> " + opt.text + "</b><br /><br />+
    <button onclick<--\" tambahKoper (" +
    oTable.rows.length + ", 'row_" +
    opt.value + ")\" class<--\"login\">Tambah Koper</button><br>+ <br>"+
    "<button onclick<--\"hapusMobil(" + oTable.rows.length + ", 'row_" +
    opt.value + ")\"
    class<--\"login\">&nbsp;&nbsp;&nbsp;Hapus Mobil&nbsp;&nbsp;&nbsp;</button>";
    oRow <-- oTable.insertRow();
    oCell <-- oRow.insertCell();
    oCell.colSpan <-- 3;
    oCell.innerHTML <-- "<hr width<--'100%' />";

- Mobil_id yang telah dipilih dihapus dari list opsi pilih mobil
    oMobil.options.remove(idx);
    oMobil.selectedIndex <-- 0;
}

```

Pada fungsi tambah mobil ini menggunakan inner html. Inner HTML adalah kode-kode dalam bahasa html yang dimasukkan ke dalam javascript. Jadi boleh dikata bahwa inner html adalah html yang digenerate oleh javascript. Terlihat pada proses tambah mobil, maka waktu memasukkan data mobil maka javascript tersebut akan membuat sel baru pada tabel yang diinginkan. Dengan demikian penambahan dan pengurangan sel pada tabel bisa dilakukan secara dinamis tergantung kebutuhan yaitu seberapa banyak data yang disisipkan.

Proses selanjutnya adalah memasukkan data koper. Untuk memasukkan data koper ini digunakan file addkoper.asp.

Gambar 5.13 Memasukkan Data Koper

Dalam memasukkan data koper ini dapat memilih koper yang didapatkan dari data tabel master koper. Untuk menampilkannya digunakan script berikut :

1. Mendefinisikan dan melakukan eksekusi terhadap variabel select pada tabel koper

```
strQuery <-- "SELECT * FROM [koper]"
Set objRec <-- objConn.Execute(strQuery)
```
2. Check data hasil query apakah kosong atau tidak

```
If Not (objRec.EOF <-- True And objRec.BOF <-- True) Then
```
3. Isi Combobox dengan data hasil query

```
<option value<-- "" >Pilih salah satu </option>
<%
  Do While Not (objRec.EOF)
    koper_id <-- objRec("kpr_id")
  %>
  <option class<--"teks" value<--<%<--koper_id %> ><%<-- koper_id %></option>
<%
  objRec.Movenext
Loop
```
4. Tutup recordset dan koneksi ke database

```
objRec.Close
END IF
%>
```
5. Mendefinisikan fungsi doReturn() untuk melakukan parsing variable ke object form.

```
<script language<--"javascript"><!--
var uang <-- document.getElementById("jumlahuang");

var koper <-- document.getElementById("koperselct");
function doReturn(){
  var obj <-- new Object();
  var idx <-- koper.selectedIndex;
  var opt <-- koper.options[idx];
  obj.koper <-- opt.value;
  obj.uang <-- uang.value;
  window.returnValue <-- obj;
  koper.options.remove(idx);
  window.close();
}
//-->
</script>
```

Pada `addkoper.asp` ini sebenarnya merupakan file asp biasa. Tetapi karena dia akan muncul karena dipanggil secara 'showmodal' oleh file `addtransaksi.asp` maka perlu trik khusus. Pemanggilan secara ini adalah dengan memanggil file yang dimaksud, tetapi halaman yang lama tidak akan bisa diakses sebelum halaman yang baru dipanggil itu selesai dikerjakan dan ditutup.

Metode pengembalian nilai juga berbeda, tidak bisa hanya dengan mengeset tujuan form pada perintah seperti biasa tetapi dengan menggunakan javascript untuk menampung nilai-nilai yang dimasukkan dan dikembalikan ke halaman sebelumnya.

Mobil	Koper	Jumlah Uang	
1 - Pickup	1	Rp 50.000.000,00	<input type="button" value="Hapus"/>
<input type="button" value="Tambah Koper"/>			
<input type="button" value="Hapus Mobil"/>	2	Rp 6.000.000,00	<input type="button" value="Hapus"/>
4 - Corolla	3	Rp 8.500.000,00	<input type="button" value="Hapus"/>
<input type="button" value="Tambah Koper"/>			
<input type="button" value="Hapus Mobil"/>			

Gambar 5.14 Menambahkan Data Koper dan Uang

Begitu koper dan jumlah uang pada masing-masing koper sudah dimasukkan, maka akan menuju ke file `addtransaksi.asp` kembali. Format uang yang dimasukkan akan otomatis ditampilkan dengan format penulisan rupiah seperti biasa, contoh Rp 232.000,00

Dalam menambahkan koper dan uang menggunakan script berikut:

```
1. Mendefinisikan fungsi tambahKoper() untuk menambahkan object form
function tambahKoper(idx,rowid){
var obj <-- showModalDialog("addkoper.asp", "",
"dialogWidth:500px;dialogHeight:300px;");
var oRow <-- document.getElementById(rowid);
if(!oRow)return;
if(oRow.cells[1] && oRow.cells[2]){
```

```

        oRow.cells[0].rowSpan + <-- 1;
        oRow <-- oTable.insertRow(idx);
    }
    oCell <-- oRow.insertCell();
    oCell.align <-- "center";
    oCell.id <-- "cell1_" + oRow.id + "_" + obj.koper;
    oCell.innerHTML <-- obj.koper + "<br />";
    oCell <-- oRow.insertCell();
    oCell.id <-- "cell2_" + oRow.id + "_" + obj.koper;
    oCell.align <-- "right";
    oCell.innerHTML <-- "Rp <input type=<--\"text\" class=<--\"teks\" value=<--\"\" +
    formatMoney(obj.uang) + "\" /><br />";
    oCell.id <-- "cell3_" + oRow.id + "_" + obj.koper;
    oCell <-- oRow.insertCell();
    oCell.align <-- "center";
    oCell.innerHTML <-- "<button onclick=<--\"hapusKoper(\" +
    obj.koper + "\", \" + oRow.id + "\");\" class=<--\"login\">Hapus</button>";

```

Proses yang dilakukan oleh fungsi tambahKoper:

- b. Tampilkan halaman addkoper.asp untuk pengisian nilai koper dan nilai uang.
- c. Simpan nilai koper ke dalam obj.koper dan nilai uang ke dalam obj.uang
- d. Buat sel baru untuk menampung kedua nilai tersebut dilengkapi dengan tombol "hapusKoper"
- e. Panggil fungsi formatMoney untuk menuliskan format nilai uang yang dimasukkan

Dalam script penambahan koper di atas juga menggunakan inner html. Javascript yang menangani penambahan sel pada tabel bilamana ada penambahan data koper baru pada mobil. Nilai koper yang dimasukkan didapatkan dari nilai yang dikirimkan oleh file addkoper.asp melalui objek obj.koper. Sedangkan nilai uang pada masing-masing koper diperoleh dari objek obj.uang.

Dan untuk memformat tampilan uang menggunakan script berikut:

```

- Mendefinisikan fungsi formatMoney() untuk melakukan formatting field dalam bentuk format money
function formatMoney(money){
    var strmoney <-- money;
    var ret <-- ",00";
    while(strmoney.length > 3){
        var leng <-- strmoney.length;

```

```

    var tmp <-- strmoney.substring(leng-3);
    strmoney <-- strmoney.substring(0, leng-3);
    ret <-- "." + tmp + ret;
}
return (strmoney + ret);

```

Proses format bentuk uang dilakukan dengan membuat variabel yang untuk mengisi 2 angka di belakang tanda ‘,’ seperti ‘,00’. Selanjutnya di cek panjang nilai uang yang dimasukkan. Dari panjang yang diperoleh kemudian dibagi 3 dan tiap 3 angka disisipkan titik ‘.’.

Sedangkan untuk menghapus koper yang telah terlanjur dimasukkan menggunakan script berikut:

```

- Mendefinisikan fungsi hapusKoper() untuk melakukan penghapusan object form.
function hapusKoper(koperid, rowid){
    var oRow <-- document.getElementById(rowid);
    var cell_1 <-- "cell1_" + rowid + "_" + koperid;
    var cell_2 <-- "cell2_" + rowid + "_" + koperid;
    var cell_3 <-- "cell3_" + rowid + "_" + koperid;
    var match <-- 0;
    if(!oRow.cells)return;
    alert(oRow.cells.length);
    for(i <-- oRow.cells.length-1; i ><-- 0; i--){
        if(oRow.cells[i].id <--<-- cell_3){
            oRow.deleteCell(i);
            oRow.deleteCell(i-1);
            oRow.deleteCell(i-2);
            oRow.rowSpan <-- 1;
            break;
        }
    }
}

```

Setiap kali proses memasukkan koper maka tiap itu pula pada tabel akan ditambahkan 3 sel baru yang berisi id koper, jumlah uang, tombol hapus. Maka untuk membatalkan nilai koper yang terlanjur diisikan, pertama-tama harus diketahui dulu baris mana yang akan dihapus.

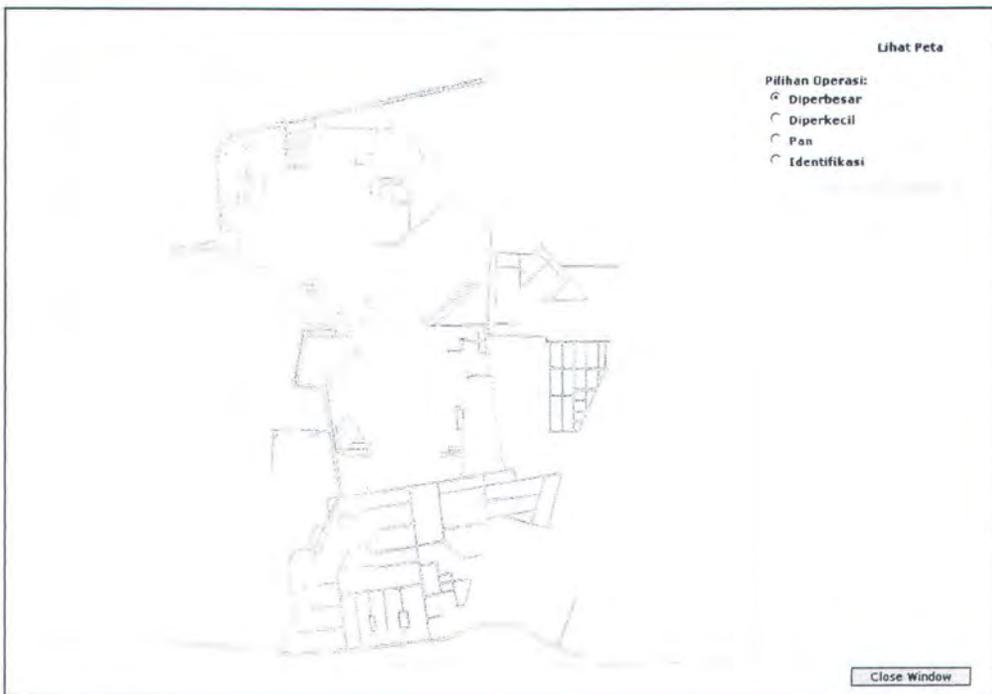
Tiap sel dalam sebuah tabel memiliki id sendiri-sendiri. Saat tombol ‘hapus koper’ di tekan maka parameter id akan di dapatkan. Selanjutnya adalah berusaha mendapatkan id ketiga sel yang diinginkan berdasarkan id yang sebelumnya diperoleh. Setelah mendapatkan id ketiga sel tersebut maka akan dengan mudah dilakukan penghapusan sel.

5.7. Pembuatan Fasilitas Melihat Peta

Pembuatan fasilitas ini bertujuan untuk memberikan gambaran bagi user mengenai area yang akan dilewati selama perjalanan. Dengan mengetahui medan yang akan ditempuh akan dapat membantu pengemudi untuk melewati jalan yang seharusnya dilewati.

Dalam melihat peta ini, diberikan fasilitas untuk melihat secara lebih detail dengan penggunaan operasi zooming, baik untuk pembesaran maupun pengecilan. Untuk memperbesar penampakan peta, maka dipilih opsi diperbesar, kemudian tinggal di klik di peta tepat dimana bagian yang ingin ditampilkan lebih besar.

Berikut adalah halaman untuk melihat peta:



Gambar 5.15. Fasilitas Melihat Peta

Selain memperbesar atau memperkecil penampakan, ditambahkan fasilitas untuk menampilkan deskripsi mengenai objek yang dipilih. Bila memilih opsi

identifikasi, kemudian memilih objek tertentu, maka akan tampil data-data mengenai objek tersebut.

Dan berikut ini adalah contoh hasil identifikasi pada sebuah objek yang dipilih:

Hasil Identifikasi (Nama Layer : jalan)	
Nama Field	Data
ENTITY	Line
LAYER	JALAN
ELEVATION	0
THICKNESS	0
COLOR	7

Gambar 5.16. Hasil Identifikasi

Fasilitas untuk melihat peta ini didapatkan dengan mengakses file `mapviewer.asp`. Berikut adalah fungsi dan prosedur yang digunakan:

1. Prosedur `LoadMapLayer` : merupakan prosedur yang bertugas untuk membuat layer-layer baru yang ingin ditampilkan beserta propertinya
2. Prosedur `strCommand` : merupakan kumpulan-kumpulan perintah untuk operasi pada peta, semisal `zoom in`, `zoom out`

```
' Boolean variables to keep tracking of what operation the user requested
Dim bZoomSelected
Dim bPanSelected
Dim bZoomOutSelected
Dim bIdentifySelected

bZoomSelected <-- false
bPanSelected <-- false
bZoomOutSelected <-- false
bIdentifySelected <-- false
```

Pada baris kode di atas dilakukan deklarasi terhadap variabel-variabel yang digunakan. Hal ini berfungsi agar variabel yang digunakan dapat dikenali oleh program

```

▪ Mendefinisikan variable Map
  Dim Map
▪ Set variabel definisi Map dengan object Map dari ASPMap
  Set Map <-- Server.CreateObject("AspMap.Map")
▪ Setting properti dari Map Object
  Map.BackColor <-- RGB(250, 255, 255)
  Map.Width <-- 700
  Map.Height <-- 600
▪ Load layer dari object Map
  !*****
  ' load map layers
  !*****
  LoadMapLayers Map

```

Langkah kedua yang dilakukan adalah membuat objek peta berdasarkan fungsi yang dibuat oleh komponen Aspmap. Langkah ini merupakan langkah yang harus dilakukan setiap kali akan membuat sebuah objek peta. Setelah membuat objek adalah memanggil prosedur pendefinisian layer-layer pada peta.

Berikut adalah kode program untuk menentukan parameter-parameter yang diminta dalam pembuatan map.

```

strTmp <-- Request("Left")
If Len(strTmp) > 0 Then
  Extent.Left <-- CDbI(strTmp)
  Extent.Bottom <-- CDbI(Request("Bottom"))
  Extent.Right <-- CDbI(Request("Right"))
  Extent.Top <-- CDbI(Request("Top"))
  Set Map.Extent <-- Extent
End If

```

Pada baris di atas adalah pembuatan objek rectangle yang merupakan objek persegi empat sebagai pembatas pada peta. Disini dideklarasikan mengenai batas titik atas, titik bawah, titik kiri, dan titik kanan.

Berikut adalah pendefinisian perintah-perintah yang bisa dilekatkan pada objek peta:

```

Select Case strCommand
  Case "MAP":
    bZoomSelected <-- True
    Map.ZoomFull
    strCommand <-- "ZoomIn"

  Case "ZOOMIN":
    bZoomSelected <-- True

    Set point <-- Map.ToMapPoint(CLng(Request("Click.X")),
CLng(Request("Click.Y")))
    Map.ZoomIn point

  Case "PAN":
    bPanSelected <-- True

    Set point <-- Map.ToMapPoint(CLng(Request("Click.X")),
CLng(Request("Click.Y")))
    Map.CenterAt point

  Case "ZOOMOUT":
    bZoomOutSelected <-- True

    Set point <-- Map.ToMapPoint(CLng(Request("Click.X")),
CLng(Request("Click.Y")))
    Map.ZoomOut point

  Case "ID":
    bIdentifySelected <-- True

    Set point <-- Map.ToMapPoint(CLng(Request("Click.X")),
CLng(Request("Click.Y")))
    layerIndex <-- Identify(Map, point, IdentifyRS)
    If layerIndex ><-- 0 Then
      LayerName <-- Map(layerIndex).Name
      CalloutName <-- IdentifyRS(Map(layerIndex).LabelField)
    End If

  Case "IMAGE":
    if Request("CalloutX").Count > 0 then
      Set callout <-- Map.Callouts.Add
      callout.x <-- CDbl(Request("CalloutX"))
      callout.y <-- CDbl(Request("CalloutY"))
      callout.Text <-- Request("CalloutName")
    end if

    ' send the map image to the client
    Map.ImageFormat <-- 1 ' JPEG format
    Map.JPEGCompression <-- 75 ' JPEG compression level
    Response.BinaryWrite Map.Image
    Response.End
End Select

```

Perintah-perintah yang dibuat prosedurnya adalah perintah operasi pembesaran, pengecilan, pan, dan identifikasi terhadap peta. Pada operasi pembesaran maka

akan memberikan nilai yang selanjutnya akan diberikan pada komponen Aspmap yang nantinya akan mengubah parameter dari objek persegi empat yang tadi dibuat. Batas titik atas, titik bawah, titik kiri, dan titik kanan akan berubah. Hal ini juga berlaku pada proses pengecilan. Sedangkan pada proses identifikasi, yang ditangkap adalah layer peta yang dipilih kemudian nilai layer yang dipilih dikembalikan ke dalam sistem database pada peta. Dengan diketahui nilai objek pada layer terpilih maka tinggal menampilkan data spasial di dalamnya.

Berikut adalah kode program untuk menggenerate peta:

```

*****
' Generate map image URL
*****
Dim MapImageURL
MapImageURL <-- "mapviewer.asp?Command<--IMAGE&Left<--" & CStr(Map.Extent.Left) &
"&Right<--" & CStr(Map.Extent.Right) & "&Top<--" & CStr(Map.Extent.Top) & "&Bottom<--" &
CStr(Map.Extent.Bottom)
if (bIdentifySelected <-- true) and (Not IdentifyRS Is Nothing) then
    MapImageURL <-- MapImageURL & "&CalloutX<--" & CStr(point.x) & "&CalloutY<--" &
CStr(point.y) & "&CalloutName<--" & CalloutName
end if

```

Peta yang ditampilkan ke dalam web merupakan hasil generate dari data spasial pada peta. Program pengolahan peta menggunakan MapInfo yang berekstensi *.tab. Demi keamanan data, maka gambar peta yang ditampilkan pada web sudah digenerate menjadi file gambar yang berekstensi *.jpg.

Prosedur di atas bertugas untuk memanggil proses generate oleh komponen Aspmap agar data-data gambar peta dirubah menjadi gambar biasa.

Berikut adalah kode-kode program dalam bentuk html untuk menampilkan ke dalam web browser:

1. If Not IdentifyRS Is Nothing Then
2. Response.Write "<CENTER><hr>Hasil Identifikasi (Nama Layer : "& LayerName & "

"
3. IdentifyRS.MoveFirst
4. if (Not IdentifyRS.EOF) Then
5. Response.Write "<TABLE BORDER<--1 cellpadding<--"1">"

```

6. Response.Write "<TR align<--"center"><TD bgcolor<--"#84ABDB" width<--"100"><B>Nama Field</B></TD><TD bgcolor<--"#84ABDB" width<--"100">
Data </TD></TR>"
7. for i <-- 0 to IdentifyRS.FieldCount - 1
8. Response.Write "<TR><TD><B>" & IdentifyRS.FieldName(i) & "</B></TD><TD
align<--"center">" & IdentifyRS(i) & "</TD></TR>"
9. Response.Write "</TABLE></CENTER>"
10. Response.Write "<br><hr>"
11. <A HREF<--"mapviewer.asp?Command<--MAP"><b>Kembali ke ukuran normal

```

Kode program di atas merupakan kode program html untuk menampilkan ke dalam web. Yang ditampilkan antara lain table, radio button, dan tombol-tombol perintah yang bila di klik akan menjalankan prosedur-prosedur perintah yang telah di deklarasikan.

Berikut kode program untuk membuat sebuah layer pada peta:

- a. Definiskan folder dimana file-file peta disimpan yaitu di direktori /maps
- b. Tambahkan layer yang diinginkan
- c. Deskripsikan property dari layer yang telah ditambahkan tersebut

```

Sub LoadMapLayers(Map)
Dim Layer, MapDir
1. MapDir <-- "Maps/"
2. AddMapLayer Map, Server.MapPath(MapDir + "jalan.tab")
3. Set Layer <-- Map.Layer("jalan")
4. Layer.ShowLabels <-- False
5. Layer.LabelField <-- "Layer"
6. Layer.LabelFont.Size <-- 10
7. Layer.LabelFont.Outline <-- True
8. Layer.Symbol.FillColor <-- RGB(0, 64, 128)
9. Layer.Symbol.LineColor <-- Layer.Symbol.FillColor
10. Layer.LabelStyle <-- mcPolygonCenter
11. Set Layer <-- Nothing

```

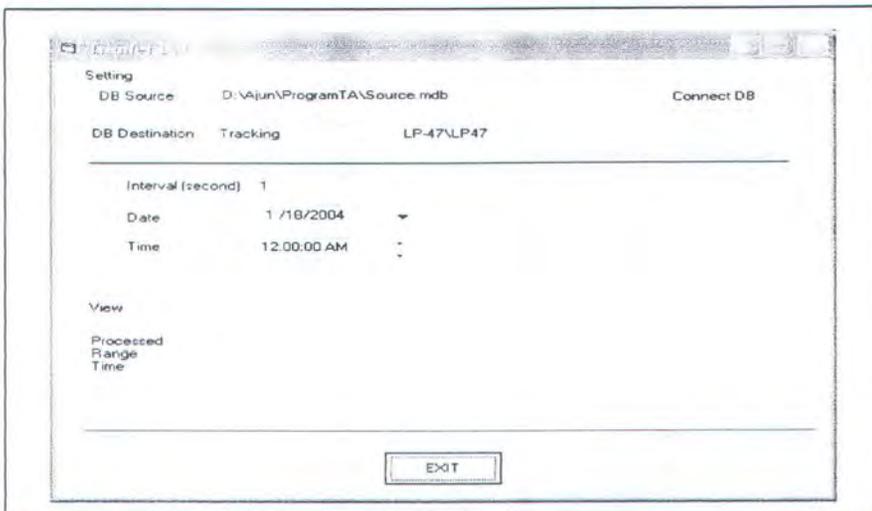
Pertama-tama dalam pembuatan prosedur pembuatan layer sebuah peta adalah mendeklarasikan direktori tempat kita menyimpan file-file peta yaitu direktori dimana kita menyimpan file-file yang berekstensi *.tab. Kemudian satu persatu kita buat layer dengan perintah addmaplayer yang mengacu pada satu file *.tab. Pada baris diatas, salah satu layer adalah layer jalan yang mengacu ke file jalan.tab. Sesudah mendeklarasikan sebuah layer, maka tiap-tiap layer



ditambahkan property sesuai dengan yang dibutuhkan. Properti tersebut antara lain warna, label, dan tipe gambar hasil generate yang diinginkan.

5.8. Pembuatan Aplikasi Server

Proses-proses yang dilakukan di server yakni dilakukan pengecekan tiap 10 detik, dimana waktu mulai pengecekan adalah pada saat tombol mulai ditekan. Proses pengecekan akan berhenti pada saat tombol berhenti ditekan. Antarmuka dari aplikasi server ditunjukkan pada gambar 5.7.



Gambar 5.17. Antarmuka Aplikasi Server

Proses pengambilan data dari database perangkat lunak AVL (*Automatic Vehicle Location*) dilakukan tiap 10 detik, dimana data yang diambil diantara waktu terakhir pengecekan sampai dengan waktu saat ini.

Data yang diperoleh dari database perangkat lunak AVL adalah ID AVL, bukan kode perjalanan. Oleh karena itu, diperlukan pengecekan pada tabel perjalanan, dimana dicari perjalanan pada saat hari ini yang sedang menggunakan ID AVL yang didapat dari database perangkat lunak AVL. Apabila datanya ada,

maka kita dapat mengambil informasi kode perjalanan, nomor mobil, tempat asal kini, dan tempat tujuan kini.

Prosedur yang digunakan :

7. GetSourceConnection : untuk membuat koneksi ke database asal
8. DisconnectSource : untuk memutuskan koneksi ke database asal
9. GetDestConnection : untuk membuat koneksi ke database tujuan
10. DisconnectDest : untuk memutuskan koneksi ke database tujuan
11. ClearAllData : untuk menghapus semua data yang ada pada tabel posisi_terkini
12. GetSourceData : untuk mengambil data dari database asal
13. UpdateDestData : untuk memasukkan data yang telah diambil ke dalam tabel tujuan yaitu tabel posisi_terkini

Prosedur GetSourceConnection

```
Sub GetSourceConnection(SourceDBLocation As String)
    Dim StrConn As String
1. Set Conn <-- New ADODB.Connection
2. StrConn <-- "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & SourceDBLocation & ";" &
    "Persist Security Info=False"
3. Conn.Open StrConn
4. ErrorHandler:
5. Err.Raise Err.Number, Err.Source, Err.Description
```

Prosedur DisconnectSource

```
Public Sub DisconnectSource()
1. On Error GoTo ErrorHandler
2. Conn.Close
3. Set Conn <-- Nothing
4. Exit Sub
5. ErrorHandler:
6. Set Conn <-- Nothing
7. Err.Raise Err.Number, Err.Source, Err.Description
```

Prosedur GetDestConnection

```
Public Sub GetDestConnection(DBName As String, DestDBLocation As String)
1. On Error GoTo ErrorHandler
2. Dim StrConn As String
3. Set Conn2 <-- New ADODB.Connection
4. StrConn <-- "Provider=SQLOLEDB.1;Persist Security Info=False;User ID=sa;Initial
   Catalog=" & DBName & ";Data Source=" & DestDBLocation
5. Conn2.Open StrConn
6. Exit Sub
7. ErrorHandler:
8. Err.Raise Err.Number, Err.Source, Err.Description
```

Prosedur DisconnectDest

```
Public Sub DisconnectDest()
1. On Error GoTo ErrorHandler
2. Conn2.Close
3. Set Conn2 <-- Nothing
4. Exit Sub
5. ErrorHandler:
6. Set Conn2 <-- Nothing
7. Err.Raise Err.Number, Err.Source, Err.Description
```

Prosedur ClearAllData

```
Public Sub ClearAllData()
1. On Error GoTo ErrorHandler
2. Conn2.Execute "DELETE FROM POSISI_TERKINI"
3. Exit Sub
4. ErrorHandler:
5. Err.Raise Err.Number, Err.Source, Err.Description
```

Prosedur GetSourceData

```
Public Sub GetSourceData(StartDateTime As Date, EndDateTIme As Date)
1. On Error GoTo ErrorHandler
2. 'tabel TES udah di acak2 u/ tes, tabel ALL direname jadi tabel DATA
3. Query <-- "SELECT UnitID, Longitude, Latitude, Speed, Date, ReportID, Distance FROM TES "
   & _
4. "WHERE Date BETWEEN #" & StartDateTIme & "# AND #" & EndDateTIme & "#"
5. 'ambil data sesuai range
6. ' Query <-- "SELECT UnitID, Longitude, Latitude, Speed, Date, Distance FROM DATA " & _
7. "WHERE Date BETWEEN #" & StartDateTIme & "# AND #" & EndDateTIme & "#"
8. If (RS.State And adStateOpen) <-- adStateOpen Then RS.Close
9. RS.Open Query, Conn, adOpenStatic, adLockOptimistic
```

Prosedur UpdateDestData

```
Public Sub UpdateDestData()
1. On Error GoTo ErrorHandler
2. Dim rs2 As ADODB.Recordset
3. Set rs2 <-- New ADODB.Recordset
4. Do While Not RS.EOF
5. 'cari dulu udah ada ato belum
6. Query <-- "SELECT Mobil_ID, POSKINI_TIME FROM POSISI_TERKINI WHERE Mobil_ID<--" &
   RS!UnitID & ""
7. If (rs2.State And adStateOpen) <-- adStateOpen Then rs2.Close
```

```

8. rs2.Open Query, Conn2, adOpenStatic, adLockOptimistic
9. If rs2.EOF Then
10. 'kalo belum ditambahin
11. Query <-- "INSERT INTO POSISI_TERKINI (Mobil_ID, POSKINI_X, POSKINI_Y,
    POSKINI_SPEED, " & _
12. "POSKINI_JARAK, POSKINI_TIME, POSKINI_STATUS) VALUES (" & RS!UnitID & "," & _
13. RS!Longitude & "," & RS!Latitude & "," & RS!Speed & "," & RS!Distance & "," & RS!Date &
    "," & RS!ReportID & "*)"
14. Conn2.Execute Query
15. Else
16. 'cek tgl lebih baru yang mana dg yg udah ada pd ID yg sesuai
17. If RS!Date > rs2!POSKINI_TIME Then
18. Query <-- "UPDATE POSISI_TERKINI SET POSKINI_X<--" & RS!Longitude & ", POSKINI_Y<--
    -" & RS!Latitude & _
19. ", POSKINI_SPEED<--" & RS!Speed & ", POSKINI_JARAK<--" & RS!Distance & ",
    POSKINI_TIME<--" & _
20. RS!Date & ", POSKINI_STATUS<--" & RS!ReportID & " WHERE Mobil_ID<--" & RS!UnitID &
    """"
21. Conn2.Execute Query
22. End If
23. End If
24. RS.MoveNext
25. Loop
26. Set rs2 <-- Nothing

```

Setelah mendapatkan informasi kode perjalanan, nomor kendaraan, tempat asal kini, dan tempat tujuan kini dari tabel perjalanan, kemudian data-data yang diperoleh dari database perangkat lunak AVL disimpan di tabel rekaman perjalanan dan posisi terkini perjalanan dengan kode perjalanan yang didapatkan dari tabel perjalanan. Untuk melakukan penyimpanan pada tabel rekaman perjalanan, maka dilakukan penambahan (*increment*) ID rekaman dari data terakhir. Sedangkan untuk menyimpan pada tabel posisi terkini perjalanan, maka perlu dilakukan pengecekan pada tabel posisi terkini perjalanan apakah sudah terdapat posisi terkini dengan kode perjalanan yang didapatkan dari tabel perjalanan. Apabila sudah ada, maka dilakukan proses *update*. Sedangkan bila belum ada, maka dilakukan penambahan data posisi terkini yang baru. Fungsi penyimpanan pada tabel rekaman perjalanan dan tabel posisi terkini perjalanan adalah sebagai berikut:

Trigger Untuk Rekor Baru

```

CREATE TRIGGER REKORD_ANYAR ON [dbo].[POSISI_TERKINI]
1. FOR INSERT, UPDATE
2. AS
3. DECLARE @newMob_Id varchar(10),
           @newp_Id Bigint,
           @max_pos_Id Bigint,
           @ins_pos_time datetime,
           @ins_pos_x float,
           @ins_pos_speed float,
           @ins_pos_y float,
           @ins_pos_status varchar(1),
           @ins_pos_jarak float,
           @ins_pos_speed_langgar float

4. SELECT @newMob_Id <-- (SELECT MOBIL_ID FROM INSERTED )
5. SELECT @newp_Id <-- (SELECT P_ID FROM PERJALANAN WHERE MOBIL_ID <--
   @newMob_Id And status_id<--10)
6. SELECT @ins_pos_speed_langgar <-- (SELECT POSKINI_SPEED FROM INSERTED)

7. if exists (SELECT * FROM POSISI WHERE P_ID <-- @newp_Id)
   begin
8. SELECT @max_pos_Id <-- (SELECT MAX(POS_ID) FROM POSISI WHERE P_ID <--
   @newp_Id)
9.     SELECT @ins_pos_x<--ins.poskini_x,
           @ins_pos_y<--ins.poskini_y,
           @ins_pos_time<--ins.poskini_time,
           @ins_pos_speed<--ins.poskini_speed,
           @ins_pos_status<--ins.poskini_status,
           @ins_pos_jarak<--ins.poskini_jarak
           from inserted ins

10. INSERT INTO POSISI
   values(@newp_Id,@max_pos_Id+1,@ins_pos_time,@ins_pos_x,@ins_pos_y,@ins_pos_spee
   d,@ins_pos_status,@ins_pos_jarak)
   if(@ins_pos_speed > 100)
   begin
11. INSERT INTO HISTORY_PELANGGARAN (P_ID,LANGGAR_WAKTU,Pelanggaran_ID)
   values(@newp_id,@ins_pos_time,'1')
   end
   if(@ins_pos_status <-- 1)
   begin
12. INSERT INTO HISTORY_PELANGGARAN (P_ID,LANGGAR_WAKTU,Pelanggaran_ID)
   values(@newp_id,@ins_pos_time,'3')
   end
   begin
13. SELECT @ins_pos_x<--ins.poskini_x,
           @ins_pos_y<--ins.poskini_y,
           @ins_pos_time<--ins.poskini_time,
           @ins_pos_speed<--ins.poskini_speed,
           @ins_pos_status<--ins.poskini_status,
           @ins_pos_jarak<--ins.poskini_jarak
           from inserted ins

14. INSERT INTO POSISI
   values(@newp_Id,1,@ins_pos_time,@ins_pos_x,@ins_pos_y,@ins_pos_speed,@ins_pos_sta
   tus,@ins_pos_jarak)
   if(@ins_pos_speed > 100)
   begin
15. INSERT INTO HISTORY_PELANGGARAN (P_ID,LANGGAR_WAKTU,Pelanggaran_ID)
   values(@newp_id,@ins_pos_time,'1')
   end
   if(@ins_pos_status <-- 1)
   begin
16. INSERT INTO HISTORY_PELANGGARAN (P_ID,LANGGAR_WAKTU,Pelanggaran_ID)
   values(@newp_id,@ins_pos_time,'3')

```

Kode program di atas bertugas untuk memasukkan data-data pada tabel posisi_terkini ke dalam tabel posisi yang merupakan history perekaman data.

5.9. Menampilkan posisi terkini dan rekaman perjalanan pada peta

Untuk menampilkan posisi terkini dan rekaman perjalanan pada peta di web, digunakan fungsi-fungsi yang telah disediakan ASPMap untuk mentransfer data olahan Sistem Informasi Geografis pada peta berbasis web. Prosedur berikut ini digunakan untuk menampilkan peta di web:

Deklarasi Variabel

```
' Boolean variables to keep tracking of what operation the user requested
1. Dim bZoomSelected
2. Dim bPanSelected
3. Dim bZoomOutSelected
4. Dim bIdentifySelected
5. bZoomSelected <-- false
6. bPanSelected <-- false
7. bZoomOutSelected <-- false
8. bIdentifySelected <-- false
9. Dim IdentifyRS, LayerName, point, CalloutName
10. Set IdentifyRS <-- Nothing
11. Set point <-- Nothing
12. LayerName <-- ""
13. CalloutName <-- ""
```

Prosedur pembuatan objek peta

```
Dim Map
1. Set Map <-- Server.CreateObject("AspMap.Map")
2. Map.BackColor <-- RGB(100, 159, 223)
3. Map.Width <-- 800
4. Map.Height <-- 600
```

Pemanggilan prosedur pembuatan layer

```
1. LoadMapLayers Map
2. AddTrack Map, AirCrafftCoord
```

Definisi Perintah

```
1. Dim strCommand
2. strCommand <-- UCase(Request("Command"))
3. If Len(strCommand) <-- 0 Then
4. strCommand <-- "MAP"
5. Select Case strCommand
6. Case "MAP":
    bZoomSelected <-- True
    Map.ZoomFull
    strCommand <-- "ZoomIn"
```

```

7. Case "ZOOMIN":
    bZoomSelected <-- True
    ' simulate the current coordinate of aircraft
    if CInt(Request("CurrPos")) < CInt(Request("MaxPos"))+1 then
        Set ClickCoord <-- Server.CreateObject("AspMap.Point")
        Set ClickCoord <-- GetAirCraftCoordinate(CInt(Request("CurrPos")))
        Response.Write "Current Coord of Pos " & CInt(Request("CurrPos")) &
" <--" & ClickCoord.x & ", " & ClickCoord.y
        CurrentPosition<--CInt(Request("CurrPos"))
        Set point <-- Map.ToMapPoint(ClickCoord.x,ClickCoord.y)
        Map.ZoomIn point
    end if
8. Case "PAN":
    bPanSelected <-- True
    Set point <-- Map.ToMapPoint(CLng(Request("Click.X")),
CLng(Request("Click.Y")))
    Map.CenterAt point
9. Case "ZOOMOUT":
    bZoomOutSelected <-- True
    if CInt(Request("CurrPos")) < CInt(Request("MaxPos"))+1 then
        Set ClickCoord <-- Server.CreateObject("AspMap.Point")
        Set ClickCoord <-- GetAirCraftCoordinate(CInt(Request("CurrPos")))
        'Response.Write "Current Pos<--" & CInt(Request("CurrPos"))
        CurrentPosition<--CInt(Request("CurrPos"))
        Set point <-- Map.ToMapPoint(CLng(ClickCoord.x),CLng(ClickCoord.y))
        Map.ZoomOut point
    end if
10. Case "ID":
    bIdentifySelected <-- True
    Set point <-- Map.ToMapPoint(CLng(Request("Click.X")),
CLng(Request("Click.Y")))
    layerIndex <-- Identify(Map, point, IdentifyRS)
    If layerIndex ><-- 0 Then
        LayerName <-- Map(layerIndex).Name
        CalloutName <-- IdentifyRS(Map(layerIndex).LabelField)
    End If
11. Case "IMAGE":
    if Request("CalloutX").Count > 0 then
        Set callout <-- Map.Callouts.Add
        callout.x <-- Cdbl(Request("CalloutX"))
        callout.y <-- Cdbl(Request("CalloutY"))
        callout.Text <-- Request("CalloutName")
    end if
    Map.ImageFormat <-- 1 ' JPEG format
    Map.JPEGCompression <-- 75 ' JPEG compression level
    Response.BinaryWrite Map.Image
    Response.End
End Select

```

Generate Image

```
Dim MapImageURL
```

1. MapImageURL <-- "viewer.asp?Command<--IMAGE&Left<--" & CStr(Map.Extent.Left) & "&Right<--" & CStr(Map.Extent.Right) & "&Top<--" & CStr(Map.Extent.Top) & "&Bottom<--" & CStr(Map.Extent.Bottom) & "&PosID<--" & CStr(CurrentPosition)
2. if (bIdentifySelected <-- true) and (Not IdentifyRS Is Nothing) then
3. MapImageURL <-- MapImageURL & "&CalloutX<--" & CStr(point.x) & "&CalloutY<--" & CStr(point.y) & "&CalloutName<--" & CalloutName

Prosedur penampilan informasi identifikasi

```

1. If Not IdentifyRS Is Nothing Then
2. Response.Write "<CENTER><b>Identify Results (" & LayerName & ")</b>:<br>"
3. PrintRecordset IdentifyRS, "#9F9FDF"
4. Response.Write "</CENTER>"

```

Kode program untuk menampilkan data posisi terkini hampir sama persis dengan kode program untuk melihat peta. Yang menjadi perbedaannya adalah pada penampilan posisi terkini, program mengambil titik-titik dari dalam database kemudian membuat layer dan seterusnya menggambar titik-titik tersebut.

Prosedur berikut ini digunakan untuk mengambil layer-layer yang akan ditampilkan pada peta:

```

Sub LoadMapLayers(Map)
Dim Layer, MapDir
1. MapDir <-- "Maps/"
2. AddMapLayer Map, Server.MapPath(MapDir + "jalan.tab")
3. Set Layer <-- Map.Layer("jalan")
4. Layer.ShowLabels <-- False
5. Layer.LabelField <-- "Layer"
6. Layer.LabelFont.Size <-- 10
7. Layer.LabelFont.Outline <-- True
8. Layer.Symbol.FillColor <-- RGB(0, 64, 128)
9. Layer.Symbol.LineColor <-- Layer.Symbol.FillColor
10. Layer.LabelStyle <-- mcPolygonCenter
11. Response.ContentType <-- "image/jpeg"
12. Response.BinaryWrite map.Image
13. Set Layer <-- Nothing

```

Prosedur berikut ini digunakan untuk mengambil data-data posisi terkini kendaraan yang sedang melakukan perjalanan:

Prosedur AddTrack

```

Sub AddTrack(Map, AirCraftCoord)
Dim Track, Points, Layer
1. Set Track <-- Server.CreateObject("AspMap.DynamicPoints")
2. Track.AddPoint AirCraftCoord.x, AirCraftCoord.y, "", ""
3. Track.Type <-- mcLineLayer ' lines
4. if Map.AddLayer(Track) then
5. Set Layer <-- Map(0)
6. Layer.Symbol.LineStyle <-- 3
7. Layer.Symbol.LineColor <-- RGB(0,0,127)
8. Set Points <-- Server.CreateObject("AspMap.DynamicPoints")
9. Points.AddPoint AirCraftCoord.x, AirCraftCoord.y, "", "aircraft"
10. Points.Type <-- mcPointLayer ' points

```

```

11. if Map.AddLayer(Points) then
    Set Layer <-- Map(0)
    Layer.LabelField <-- "LABEL"
    Layer.ShowLabels <-- true
    Layer.LabelFont.Size <-- 13
    Layer.LabelFont.Bold <-- true
    Layer.LabelFont.Italic <-- true
    Layer.LabelFont.Color <-- RGB(0, 0, 255)
    Layer.LabelFont.Outline <-- true
    Dim Renderer
    Set Renderer <-- Layer.Renderer
    Renderer.Field <-- "TAG"
    Set feature <-- renderer.Add
    feature.Value <-- "aircraft"
    feature.Symbol.PointStyle <-- mcBitmapPoint ' bitmap
    feature.Symbol.Bitmap <-- Server.MapPath("image/car.bmp")
    feature.Symbol.Size <-- 13
    feature.Symbol.TransparentColor <-- RGB(255,255,255) ' white

```

Prosedur Callout

Dim Callout

```

1. Set callout <-- Map.Callouts.Add
2. callout.x <-- AirCraftCoord.x
3. callout.y <-- AirCraftCoord.y
4. callout.Text <-- "x<--" & AirCraftCoord.x & ", y<--" & AirCraftCoord.y
5. callout.Font.Size <-- 13
End Sub

```

Fungsi GetAirCraftCoordinate

Function GetAirCraftCoordinate(PosID)

Dim Point

Dim strConn

Dim oRs

```

1. strConn <-- "DRIVER<--{SQL Server};SERVER<--10.126.11.146;DATABASE<--
tracking;UID<--sa;PWD<--"
2. Set oRs <-- Server.CreateObject("ADODB.Recordset")
3. oRs.CursorLocation <-- adUseClient ' ADO bookmarks support
4. oRs.Open "SELECT descrip, lat_y as y, long_x as x From airports where id<--"& PosID,
strConn, adOpenKeyset, adLockReadOnly, adCmdText
5. oRs.MoveFirst
6. Set Point <-- Server.CreateObject("AspMap.Point")
7. Point.x <-- oRs("x")
8. Point.y <-- oRs("y")
9. Set GetAirCraftCoordinate <-- Point

```

Untuk mengambil data-data posisi terkini, yang pertama dilakukan adalah melakukan koneksi ke database untuk mengambil data-data pada tabel posisi terkini. Data-data tersebut kemudian disimpan ke dalam sebuah variabel.

Hal yang kedua yaitu membuat layer yang nantinya akan dipergunakan untuk menggambar titik-titik yang sudah diperoleh. Setelah deklarasi layer, selanjutnya

titik yang sudah didapatkan tadi digambar ke dalam peta. Proses ini terus diulang-ulang sampai data pada posisi terkini tidak berubah lagi. Agar lebih interaktif pada titik tersebut diberi gambar mobil. Dengan demikian pada peta terlihat gambar mobil yang bergerak sesuai dengan lintasan yang mengacu pada data-data dalam tabel posisi terkini.

Tampilan Peta posisi terkini kendaraan yang sedang melakukan perjalanan ditunjukkan pada gambar 5.18.



Gambar 5.18. Tampilan Peta Posisi Terkini Mobil yang sedang Melakukan Perjalanan

Dalam peta terlihat gambar mobil yang sedang melakukan perjalanan. Dan disitu ditunjukkan juga berapa posisi terkini yang sedang dijalani.

Prosedur berikut ini digunakan untuk mengambil data-data rekaman perjalanan:

Prosedur LoadMapLayer

```

LoadMapLayers Map
Dim strConn
Dim oRs
1. strConn <-- "Provider<--Microsoft.Jet.OLEDB.4.0;Data Source<--" & _
2. Server.MapPath("airports.mdb")
3. Set oRs <-- Server.CreateObject("ADODB.Recordset")
4. oRs.CursorLocation <-- adUseClient ' ADO bookmarks support
5. oRs.Open "SELECT id, descrip, lat_y as y, long_x as x From airports", strConn,
adOpenKeyset, adLockReadOnly, adCmdText
6. if not Map.AddLayer(oRs) then
7. Response.Write "Can't add database layer"
8. Response.End
9. Map(0).Symbol.FillColor <-- RGB(255,0,0)

```

Inisialisasi Track

```

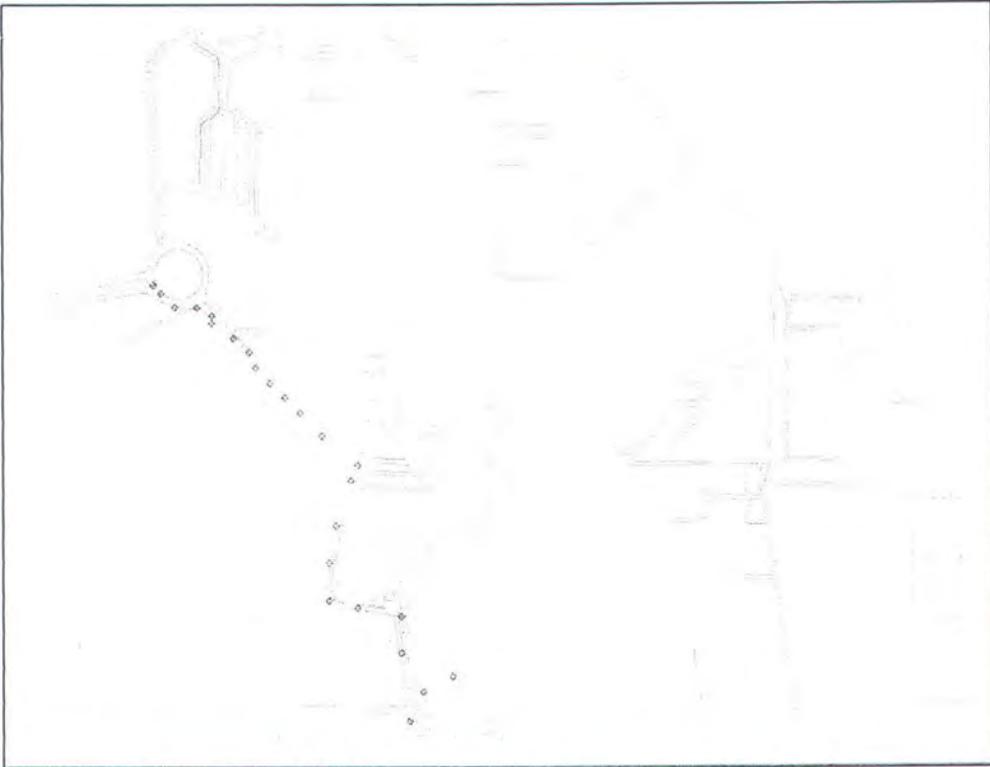
Dim Track
1. Set Track <-- Server.CreateObject("AspMap.DynamicPoints")
2. Track.AddPoint 250, 500, "", "pesawat"
3. Track.Type <-- mcLineLayer ' lines
4. if Map.AddLayer(Track) then
    Set Layer <-- Map(0)
    Layer.Symbol.LineStyle <-- 3
    Layer.Symbol.LineColor <-- RGB(127,0,127)
    Dim Renderer
    Set Renderer <-- Layer.Renderer
    ' aircraft symbol
    Set feature <-- renderer.Add
    feature.Value <-- "pesawat"
    feature.Symbol.PointStyle <-- mcBitmapPoint ' bitmap
    feature.Symbol.Bitmap <-- Server.MapPath("image/car.bmp")
    feature.Symbol.Size <-- 32
    feature.Symbol.TransparentColor <-- RGB(255,255,255) ' white

```

Untuk menampilkan arsip perekaman data dalam sebuah peta, maka data yang diambil merupakan data yang berasal dari tabel posisi. Titik-titik yang diperoleh digambarkan semuanya ke dalam peta. Dan kemudian antar titik tersebut digambar garis sehingga lintasan yang dilalui terlihat jelas.

Pada proses melihat arsip perjalanan juga dilengkapi dengan perintah-perintah pembesaran, pengecilan, identifikasi yang memudahkan pengguna dalam menjalankan aplikasi.

Tampilan peta rekaman perjalanan ditunjukkan pada gambar 5.19.



Gambar 5.19. Tampilan Peta Rekaman Perjalanan Mobil

Di dalam peta terlihat titik-titik yang merupakan bekas lintasan yang telah dilewati selama dalam perjalanan. Gambar di atas diambil pada lintasan yang telah diproses pada pembesaran sebesar 1 kali.

BAB VI



UJI COBA PERANGKAT LUNAK DAN ANALISA HASIL

BAB VI

UJI COBA PERANGKAT LUNAK DAN ANALISA HASIL

Di bagian sebelumnya telah dijelaskan mengenai spesifikasi dari sistem pelacakan yang telah dibuat. Dalam subbab ini akan dibahas mengenai ujicoba terhadap sistem yang telah dibuat. Program yang telah selesai dibuat akan diuji dengan berbagai macam kondisi.

6.1. Lingkungan Uji Coba

Uji coba terhadap sistem yang telah dibuat dilakukan dengan menggunakan lingkungan dengan spesifikasi sebagai berikut:

Spesifikasi sistem:

- Microsoft Windows XP Profesional
- Microsoft Access XP
- Microsoft SQL Server 2000
- Visual Basic 6
- ASP dan ASPMap

Spesifikasi hardware:

- Pentium II 450 Mhz
- RAM 384 MB
- 4,2 GB Hardisk

6.2. Skenario Uji Coba

Pada uji coba ini diberikan beberapa skenario yang ditujukan untuk mengetahui fungsionalitas dari perangkat lunak yang dibuat. Uji coba dilaksanakan untuk menguji kesesuaian perangkat lunak dengan desain fitur yang

ingin dibuat. Pada uji coba ini akan dipilih fitur-fitur yang utama yang antara lain visualisasi posisi terkini, sifat online-real time, sistem peringatan bahaya, analisa data perjalanan, dan kontrol sistem yang ketat.

Visualisasi posisi terkini dan sifat online-real time diuji dengan skenario menampilkan posisi terkini ke dalam peta yang berbasis web. Analisa data dan peringatan bahaya diuji dengan skenario perubahan status AVL pada saat dalam perjalan. Sedangkan kontrol sistem yang ketat diuji dengan skenario pengontrolan kecepatan kendaraan. Berikut 3 skenario yang diujikan, yaitu:

6. Pelacakan posisi kendaraan terkini
7. Pelacakan posisi dengan tiba-tiba status AVL=1
8. Kontrol kecepatan pada saat melewati jalur tertentu

6.2.1. Uji Coba Skenario 1

Pada hari Senin tanggal 12 Januari 2003 akan dilaksanakan pengiriman uang sebesar 13 juta rupiah. Pengiriman dilakukan dengan menggunakan kendaraan minibus. Uang tersebut terbagi dalam 3 koper, masing-masing:

- a. Koper A berisi 5 juta
- b. Koper B berisi 5 juta
- c. Koper C berisi 3 juta.

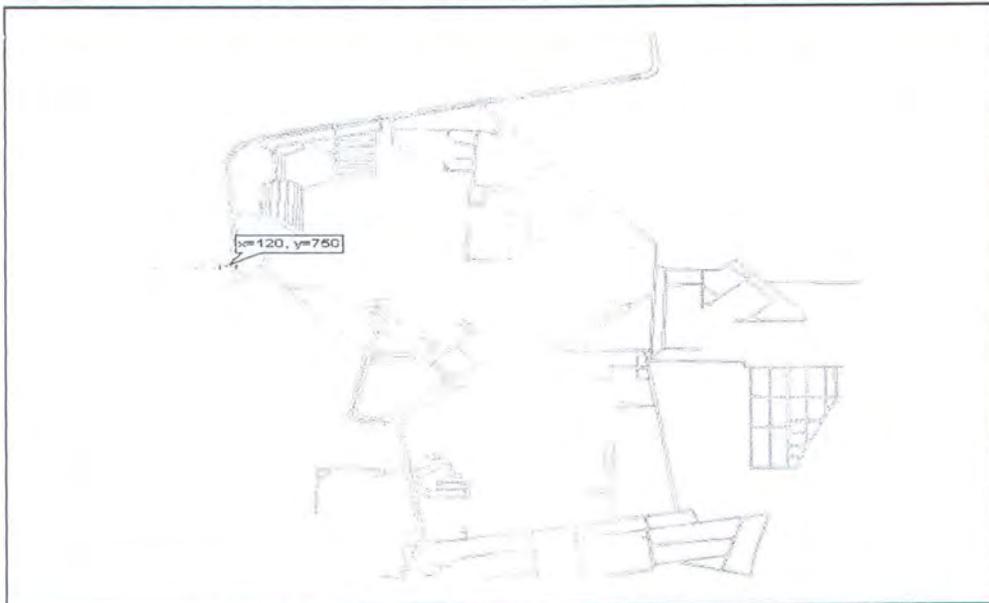
Posisi kantor pusat yang merupakan adalah di daerah bundaran ITS dengan posisi koordinat(120,750). Tujuan pengiriman adalah Kantor Bank BNI Cabang ITS dengan posisi koordinat (530,230). Jalur pengiriman yang sudah ditentukan adalah sebagai berikut :

Bundaran ITS → Jl. Taman Alumni → Jl. T. Elektro → Jl. T. Mesin → Kantor Bank BNI 46 Cabang ITS.

Ujicoba yang pertama dilakukan untuk mengetahui, apakah benar bahwa perangkat lunak yang telah dibuat mampu menampilkan posisi secara realtime ke dalam web. Dan perlu juga dicermati apakah titik-titik yang ditampilkan sesuai dengan koordinat dari database.

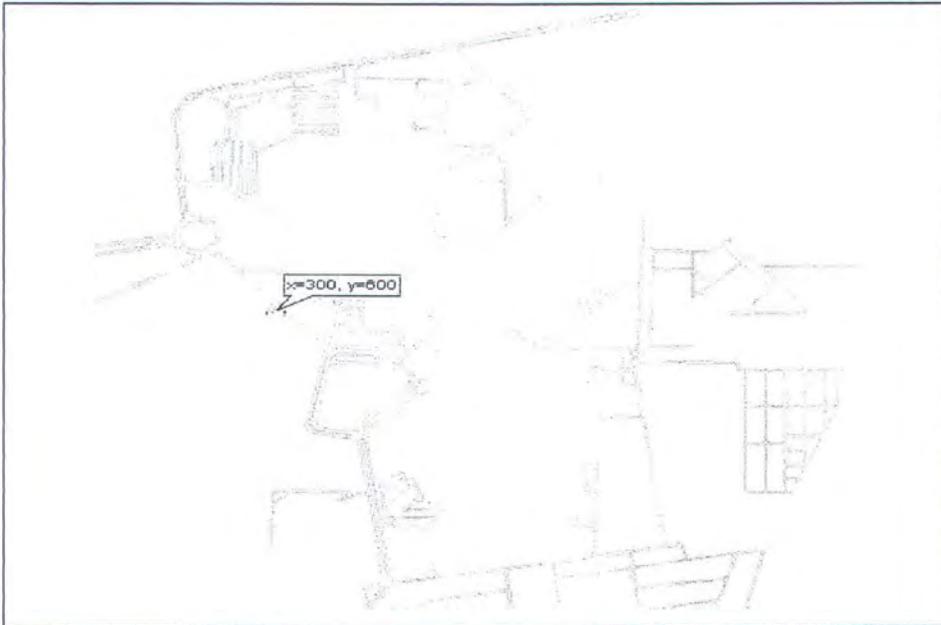
Pertama-tama yang dilakukan adalah memasukkan data-data transaksi ke dalam database. Kapan perjalanan itu dilakukan, tempat asal dan tempat tujuan pengiriman, mobil yang dipakai, koper yang dibawa beserta jumlah uang di dalamnya. Hal ini bisa disebut sebagai proses persiapan pemberangkatan.

Setelah proses persiapan pemberangkatan telah selesai dilakukan, maka AVL yang sebelumnya dalam kondisi off dirubah menjadi kondisi on. Posisi kendaraan pada saat setelah berangkat dari kantor pusat ditunjukkan pada gambar 5.20, dimana AVL telah dinyalakan dan mulai mengirimkan data.



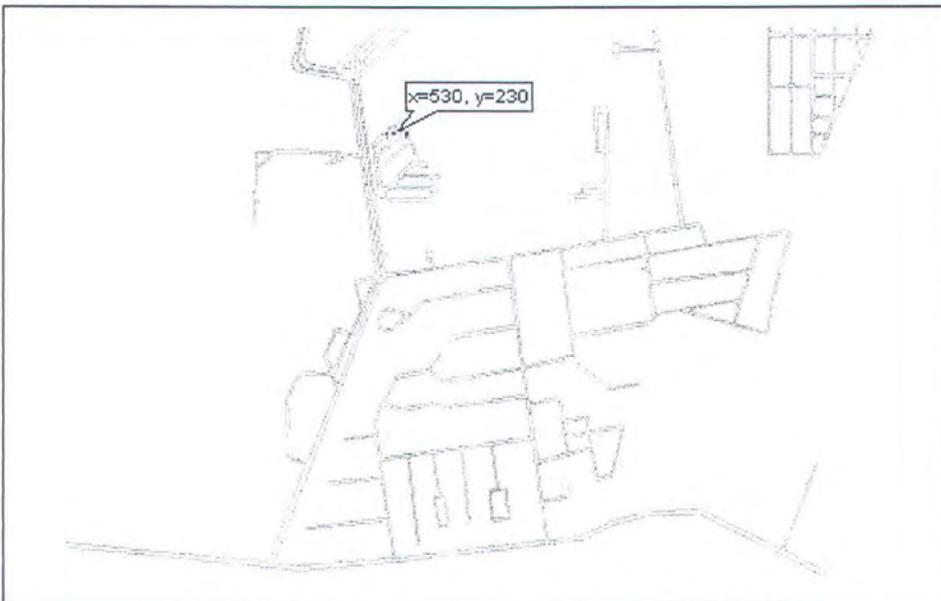
Gambar 6.1. Posisi Kendaraan Waktu Start

Posisi kendaraan pada saat berada di Jl. Taman Alumni ditunjukkan pada gambar 5.21.



Gambar 6.2. Posisi Kendaraan Waktu di Jl. Taman Alumni

Posisi kendaraan pada saat telah tiba di kantor Bank BNI 46 Cabang ITS ditunjukkan pada gambar 5.22.



Gambar 6.3. Posisi kendaraan pada saat telah tiba di Bank BNI

Dari skenario pertama dapat dilihat bahwa perangkat lunak dapat menampilkan posisi terkini dari kendaraan. Keterangan mengenai posisi terkini juga ditampilkan langsung ke dalam web sehingga dapat dengan mudah diketahui berapakah koordinat dari mobil yang sekarang ini.

Dengan mampu menampilkan posisi terkini, berarti perangkat lunak yang dibuat juga telah membuktikan bersifat online-real time.

6.2.2. Uji Coba Skenario 2 : Status AVL yang Berubah.

Pada hari Selasa tanggal 13 Januari 2003 akan dilaksanakan pengiriman uang sebesar 13 juta rupiah. Pengiriman dilakukan dengan menggunakan kendaraan [4-Corolla]. Uang tersebut terbagi dalam 3 koper, masing-masing:

- a. Koper A berisi 5 juta
- b. Koper B berisi 5 juta
- c. Koper C berisi 3 juta.

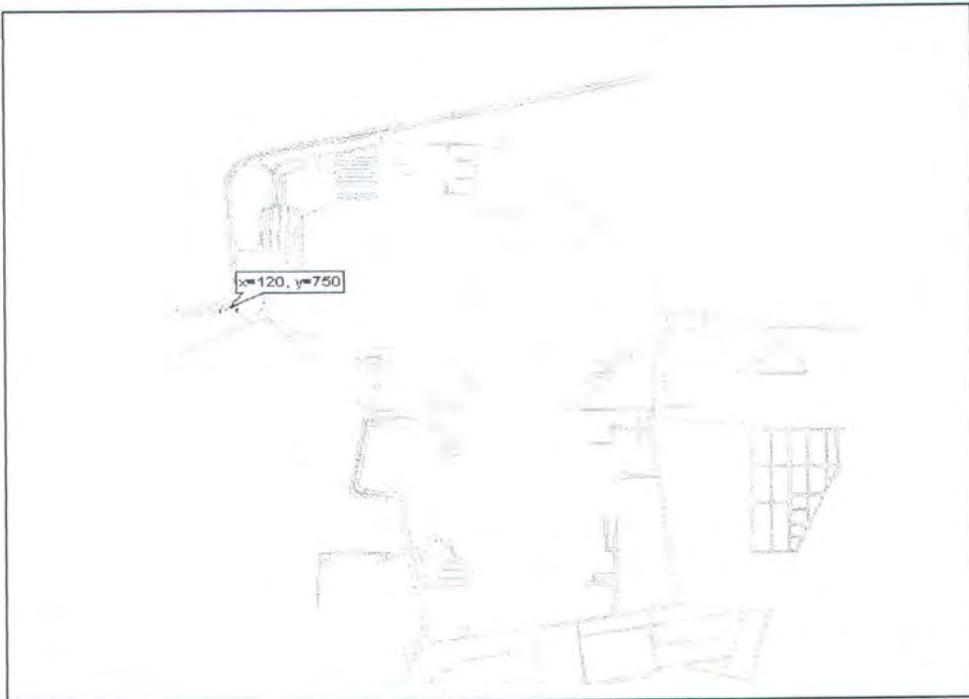
Posisi kantor pusat yang merupakan adalah di daerah bundaran ITS dengan posisi koordinat(120,750). Tujuan pengiriman adalah Kantor Bank BNI Cabang ITS dengan posisi koordinat (530,230). Jalur pengiriman yang sudah ditentukan adalah sebagai berikut :

Bundaran ITS → Jl. Taman Alumni → Jl. T. Elektro → Jl. T. Mesin → Kantor Bank BNI 46 Cabang ITS.

Semula perjalanan berjalan dengan lancar sebagaimana pada kasus yang pertama. Akan tetapi pada saat kendaraan telah mulai memasuki Jl. T. Elektro sesuatu telah terjadi.

Pada ujicoba yang kedua ini untuk melihat bahwa perangkat lunak yang dibuat mampu memberikan peringatan terhadap kejadian-kejadian yang telah mengganggu perjalanan pengiriman. Pada ujicoba yang kedua diambil kasus berubahnya status AVL pada saat mobil sedang berjalan.

Pada bab sebelumnya telah dibahas bahwa salah satu kriteria yang membahayakan proses perjalanan adalah berubahnya status AVL ketika sedang berjalan. Hal ini masuk kategori 'bergerak darurat'. Berikut jalannya ujicoba dengan kasus kedua.



Gambar 6.4. Posisi Kendaraan Waktu Start

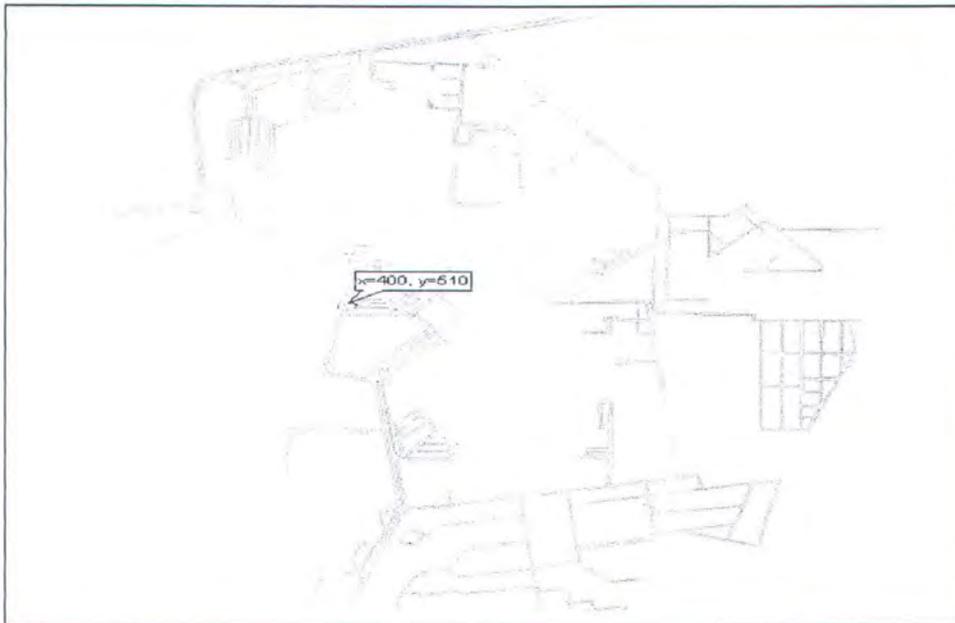
Pada saat awal memang tidak terjadi gejala yang mencurigakan. Mobil melaju seperti biasanya. Ternyata ketika memasuki titik ke 15, status AVL tiba-tiba berubah menjadi 1. Hal ini menandakan bahwa telah terjadi sesuatu terhadap

kendaraan. Berikut petikan rekaman perjalanan sampai dengan titik ke-15 yang telah mengalami perubahan status AVL.

ID	LONG X	LAT Y	Status AVL
1	120	750	0
2	130	740	0
3	150	720	0
4	180	720	0
5	200	710	0
6	200	700	0
7	230	680	0
8	250	660	0
9	260	640	0
10	280	620	0
11	300	600	0
12	320	580	0
13	350	550	0
14	400	510	0
15	390	490	1
16	370	430	1
17	360	380	1

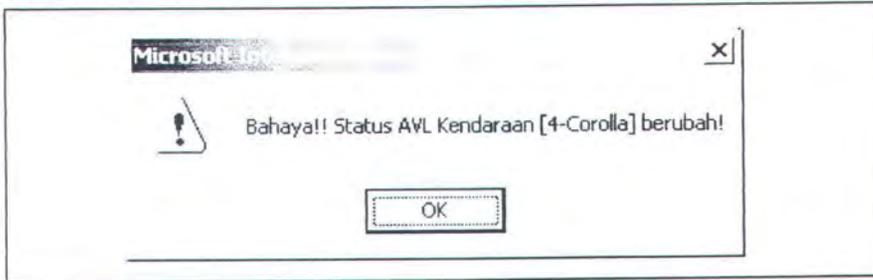
Tabel 6.1 Rekaman Data Perjalanan dengan Status AVL yang berubah

Mobil tetap berjalan dengan normal bahkan sampai pada titik ke-14 seperti ditunjukkan pada gambar 5.24.



Gambar 6.5 Kondisi Perjalanan pada titik ke-14

Pada saat titik ke-15, karena terjadi perubahan status AVL maka muncul peringatan sebagai berikut:



Gambar 6.6. Peringatan Status AVL yang berubah

Perubahan status AVL merupakan suatu indikasi yang masuk dalam kategori bahaya. Bila AVL mati maka data perjalanan tidak dapat direkam. Atau bila mengirimkan sinyal perubahan status berarti sesuatu telah terjadi.

AVL biasanya disambungkan dengan sistem pada mobil semisal sistem mesin dan pintu mobil. Bila ada gangguan semisal pintu mobil dibuka paksa atau ada kerusakan mesin maka AVL akan langsung merespon. Pintu mobil dibuka paksa bisa disebabkan karena terjadi tindak pidana semisal perampokan atau penodongan. Dengan demikian peringatan bahaya memang menjadi hal yang seharusnya.

Dari hasil studi kasus di atas telah terbukti, ketika status AVL telah berubah dari 0 menjadi 1 maka pada saat itu muncul peringatan bahaya. Dari sini bisa disimpulkan bahwa perangkat lunak yang dibuat telah mampu memberikan peringatan terhadap bahaya.

6.2.3. Uji Coba Skenario 3 : Kontrol Kecepatan

Pada hari Rabu tanggal 14 Januari 2003 akan dilaksanakan pengiriman uang sebesar 13 juta rupiah. Pengiriman dilakukan dengan menggunakan kendaraan [4-Corolla]. Uang tersebut terbagi dalam 3 koper, masing-masing:

- a. Koper A berisi 5 juta
- b. Koper B berisi 5 juta
- c. Koper C berisi 3 juta.

Posisi kantor pusat yang merupakan adalah di daerah bundaran ITS dengan posisi koordinat(120,750). Tujuan pengiriman adalah Kantor Bank BNI Cabang ITS dengan posisi koordinat (530,230). Jalur pengiriman yang sudah ditentukan adalah sebagai berikut :

Bundaran ITS → Jl. Taman Alumni → Jl. T. Elektro → Jl. T. Mesin → Kantor Bank BNI 46 Cabang ITS.

Kontrol kecepatan berfungsi untuk memantau kecepatan dari kendaraan. Pengontrolan yang dilakukan tidak hanya bila kendaraan telah melebihi kecepatan yang telah ditentukan, tetapi juga bila kendaraan berhenti di tengah perjalanan tanpa suatu alasan yang jelas. Walaupun kadangkala berhenti di tengah jalan karena lampu merah, tetapi waktu yang dibutuhkan tidak boleh melebihi waktu yang telah ditentukan yakni 3 menit. Bilamana berhenti melebihi waktu tersebut maka akan muncul peringatan.

Pada ujicoba yang ketiga ini bertujuan untuk mengetahui bahwa pihak pusat sebagai penanggung jawab proses pengiriman masih mempunyai daya kontrol yang kuat walau mobil masih dalam perjalanan. Kriteria yang dijadikan

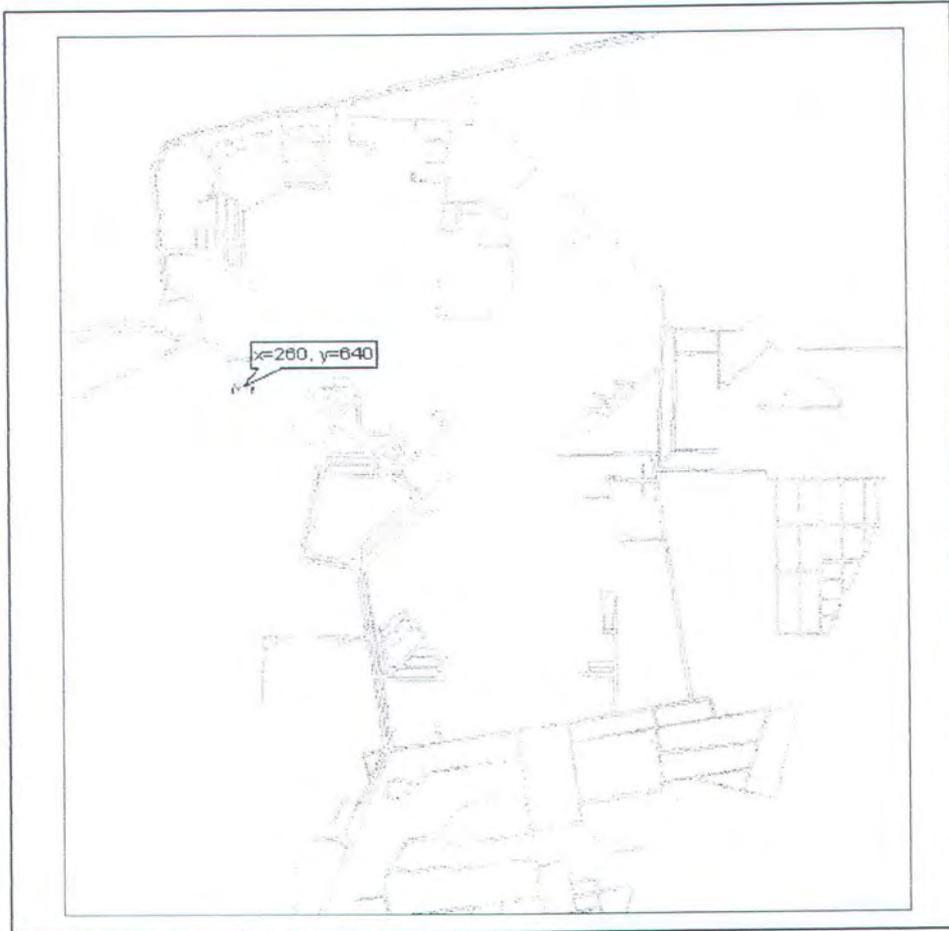
contoh adalah kecepatan kendaraan. Bila pihak pusat memang memiliki daya kontrol maka dengan mengetahui kecepatan dari kendaraan akan dapat mengingatkan pengemudi bila melaju dengan kecepatan yang tidak sesuai. Apakah itu melebihi batas atau justru kurang dari kriteria yang ditentukan.

Berikut adalah cuplikan tabel rekaman data perjalanan yang menunjukkan kontrol kecepatan.

ID	LONG X	LAT Y	Kecepatan	Status AVL
1	120	750	0	0
2	130	740	30	0
3	150	720	50	0
4	180	720	60	0
5	200	710	60	0
6	200	700	75	0
7	230	680	80	0
8	250	660	90	0
9	260	640	100	0
10	280	620	105	0
11	300	600	110	0
12	320	580	95	0
13	350	550	70	0
14	400	510	50	0
15	390	490	20	0
16	370	490	0	0
17	370	490	0	0
18	370	490	0	0
19	370	490	0	0
20	360	380	40	0

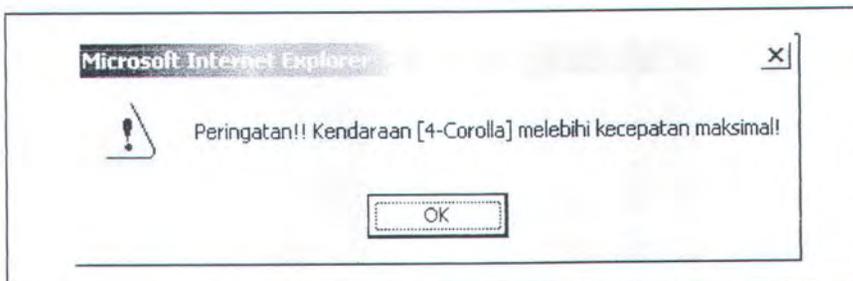
Tabel 6.2. Rekaman Data Perjalanan dengan Kecepatan yang Berubah-ubah

Sejak awal kendaraan telah melaju. Semakin lama kecepatannya semakin bertambah. Hal ini bisa dilihat pada tabel di atas. Sampai akhirnya pada titik ke-9 kecepatan telah mencapai 100 km/jam. Pada posisi dengan kecepatan sama atau kurang dari 100 km/jam tidak akan mendapatkan peringatan. Berikut gambar peta pada posisi titik ke-9 seperti ditunjukkan oleh gambar 5.26.



Gambar 6.7. Perjalanan pada titik ke-9

Kemudian kendaraan tetap melaju, bahkan menambahkan kecepatannya. Saat diketahui kecepatan melebihi 100 km/jam maka sistem akan langsung memberikan peringatan seperti berikut:

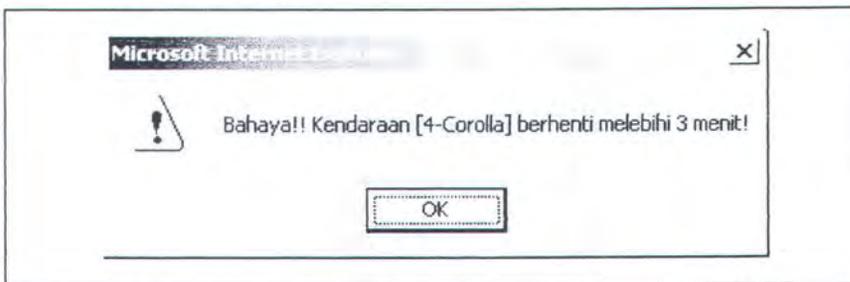


Gambar 6.8. Peringatan melebihi kecepatan

Pada saat kecepatan melebihi batas maksimal, maka yang diberikan hanyalah sebatas peringatan saja. Hal ini beda bilamana yang terjadi adalah berhenti melebihi batas yang telah ditentukan.

Pada tabel dapat kita lihat ketika kendaraan memasuki titik ke-16. Kendaraan tiba-tiba berhenti. Untuk kecepatan=0 yang pertama, hal ini masih bisa dimaklumi. Tetapi untuk mencegah hal-hal yang tidak diinginkan, pada tahap ini pihak pusat bisa langsung mengontak kurir.

Kendaraan ternyata tetap pada kecepatan=0. Hal ini terlihat pada titik ke-17. Hal ini pun masih bisa dimaklumi. Tetapi ternyata pada titik ke-18, kecepatan kendaraan tetap 0 dan posisinya tidak berubah. Maka hal ini termasuk dalam kategori bahaya dan sistem akan langsung memberikan peringatan bahaya. Bukan hanya sekedar peringatan saja.



Gambar 6.9. Peringatan berhenti melebihi waktu yang ditentukan

Dari ujicoba yang ketiga telah terbukti bahwa ketika kecepatan kendaraan telah melebihi batas yang ditentukan akan langsung muncul peringatan. Dengan masukan ini maka pihak pusat dapat langsung segera menegur pengemudi untuk mengurangi kecepatannya. Hal tersebut juga berlaku bilamana kecepatan pengemudi di bawah standar yang ditentukan.

Untuk kasus berhenti di tengah jalan memang sudah termasuk kategori yang bahaya mengingat kendaraan yang mengangkut uang dan berhenti di tengah

jalan dengan waktu yang lama akan memancing tindak pidana. Tindak pidana yang bisa terjadi adalah perampokan. Atau bahkan bisa dari pengemudi itu sendiri yang telah berbuat curang dan kemudian mengurangi jumlah uang yang dikirim untuk dialihkan ke tempat lain selama kendaraan berhenti di tengah perjalanan. Hal-hal semacam ini harus dihindari salah satunya dengan pemberian peringatan.

Pada ujicoba yang ketiga ini terbukti bahwa perangkat lunak yang dibuat mampu memberikan peringatan terhadap hal-hal yang bisa memancing bahaya dan membuktikan bahwa meskipun masih dalam perjalanan, pihak pusat masih memiliki kontrol yang kuat terhadap kendaraan yang dipakai selama proses pengiriman.

6.3. Analisa Hasil Ujicoba

Ujicoba dilakukan untuk menguji perangkat lunak yang sudah dibuat, apakah sudah sesuai dengan fitur yang telah dirancang atau tidak. Studi kasus pada ujicoba disesuaikan agar bisa menjawab pertanyaan-pertanyaan tersebut.

Pada ujicoba yang pertama yaitu menampilkan posisi terkini kendaraan telah sukses dilaksanakan. Posisi dari mobil dapat tampil di peta dan bergerak secara dinamik berdasarkan titik-titik koordinat yang telah tersimpan ke dalam database.

Pada ujicoba yang kedua, dimana sistem pelacakan diuji dengan data status AVL yang berubah pada saat dalam perjalanan. Sistem pelacakan telah terbukti mampu menampilkan peringatan bahaya berkenaan dengan perubahan status tersebut.

Pada ujicoba yang ketiga, dimana fokus yang diuji adalah kecepatan dari kendaraan. Hasil yang diperoleh, sistem mengeluarkan peringatan pada saat

kendaraan melaju melebihi batas maksimal. Tidak hanya itu, sistem pelacakan juga mengeluarkan peringatan bahaya pada saat kendaraan berhenti di tengah jalan.

Mengacu pada hasil ketiga uji coba tersebut, terbukti bahwa fitur-fitur yang telah dibuat memang yang diperlukan. Kombinasi faktor visualisasi posisi, analisa data perjalanan, pemberian peringatan, kontrol pusat yang ketat dan faktor keamanan data merupakan solusi untuk meningkatkan keamanan pada proses pengiriman uang. Manfaat kombinasi bisa dilihat dari contoh berikut. Ketika kendaraan dibajak oleh perampok dan dilarikan, posisi dari kendaraan tersebut masih di deteksi. Hal ini cukup membantu pihak yang berwajib dalam meringkus pelaku pidana tersebut. Secara logika, perampok dalam melarikan kendaraan pasti melaju dengan kecepatan yang tinggi. Dengan demikian bisa dipastikan laju kendaraan akan melebihi batas. Hal ini sudah memberikan input kepada pihak pusat tentang apa yang terjadi. Ditambah lagi bahwa bila perampok tersebut dalam merebut kendaraan telah membuka pintu secara paksa, maka AVL akan berubah status karena sistem AVL telah disambungkan dengan sistem pintu pada mobil. Hal ini juga telah memberikan informasi ke pihak pusat bahwa telah terjadi pelanggaran prosedur.

Dari contoh perampokan diatas, ternyata pihak pusat dengan gampang mengetahui dan kemudian melacaknya dengan bantuan fitur-fitur yang telah dibuat. Bahkan tanda-tanda telah terjadinya pelanggaran prosedur tidak hanya diperoleh dari 1 masukan saja tetapi dari berbagai fitur seperti yang telah dijelaskan.

Kesimpulan akhir, bahwa dengan adanya perangkat lunak sistem pelacakan kotak uang dengan menggunakan GIS ini dapat membantu proses pengiriman kotak uang yang selama ini selalu dilakukan dengan proses yang masih konvensional yaitu masih mengandalkan sumber daya manusia saja tanpa memanfaatkan teknologi yang ada.





BAB VII

PENUTUP

BAB VII

PENUTUP

7.1. Kesimpulan

Berdasarkan analisa hasil uji coba perangkat lunak maka dibuat kesimpulan sebagai berikut :

1. Adanya perangkat lunak dapat membantu kantor pusat dalam mengambil keputusan yang tepat pada saat terjadi kendala selama dalam proses pengiriman. (meliputi pemantauan status AVL, kontrol kecepatan, penanganan hambatan teknis).
2. Dengan pemanfaatan data yang diperoleh dari AVL (Automatic Vehicle Location), aplikasi dapat menampilkan pada peta posisi terkini dari kendaraan yang sedang melakukan perjalanan.
3. Aplikasi dapat mengolah data dari AVL kemudian menyimpannya sebagai data hasil perjalanan yang pernah ditempuh untuk dianalisa lebih lanjut oleh pihak pusat.
4. Aplikasi dapat membantu proses pengiriman yang selama ini dilakukan secara konvensional dengan memanfaatkan teknologi komunikasi terkini.

7.2. Saran

Berdasarkan hasil evaluasi yang dilakukan terhadap sistem, ada beberapa saran yang perlu dipertimbangkan dalam pengembangan aplikasi ini, yaitu :

1. Untuk pemberian peringatan, bisa dilanjutkan dengan mengirimkan peringatan ke ponsel pihak manajer yang bertanggung jawab. Jadi tidak hanya sekedar

peringatan di web. Hal ini dapat lebih mempercepat waktu respon karena manajer dapat menerima pesan peringatan di manapun dan kapanpun berada.

2. Media komunikasi yang digunakan melalui media sms memang terbukti lebih murah. Tetapi dengan ditemukannya teknologi GPRS maka bisa diambil alternatif menggunakan media GPRS tersebut. Dengan pertimbangan dapat berkoneksi secara terus menerus dengan biaya minimal karena hanya teks yang dikirimkan.



DAFTAR PUSTAKA

DAFTAR PUSTAKA

1. [DI-1995]Department of Infrastructure "Report on global applications of GPS based transport technologies and potential use in Australia". Energy and Resources;1995.
2. [ESR-1999]ESRI. "ArcView Tracking Analyst: Complete Tracking Solutions". An ESRI White Paper. Environmental Systems Research Institute, Inc. May 1999.
3. [VT-2003]Vehicle tracking, GSM GPS modem / module; Available from: <http://www.gsm-mode.de/GSM>. Accessed September 10, 2003.
4. [TNL-2003]Trimble Navigation Limited : "How GPS Work".2003; Available from: http://www.trimble.com/gps/nfsections/howworks/aa_hwl.htm. Accessed December 2, 2003.
5. [UTX-2003]Utexas."GPS Overview".2003 ; Available from: <http://www.utexas.edu/depts/grg/gcraft/notes/gps/gps.html>. Accessed September 23, 2003
6. [BI-2003]BI."Peraturan Bank Indonesia".2003;Available from: <http://www.bi.go.id>. Accessed Oktober 12, 2003.

... Sunday ...
...
...
...
...

...