

**IMPLEMENTASI MODUL PEMETAAN DAN  
MAPSERVER SISTEM INFORMASI BIS KAMPUS  
UNIVERSITAS INDONESIA**

**SKRIPSI**

Oleh :

**ARIE WAHYU HIDAYAT**  
**04 04 03 013X**



**DEPARTEMEN ELEKTRO  
FAKULTAS TEKNIK UNIVERSITAS INDONESIA  
GENAP 2007/2008**

**IMPLEMENTASI MODUL PEMETAAN DAN  
MAPSERVER SISTEM INFORMASI BIS KAMPUS  
UNIVERSITAS INDONESIA**

**SKRIPSI**

Oleh :

**ARIE WAHYU HIDAYAT  
04 04 03 013X**



**SKRIPSI INI DIAJUKAN UNTUK MELENGKAPI SEBAGIAN  
PERSYARATAN MENJADI SARJANA TEKNIK**

**DEPARTEMEN ELEKTRO  
FAKULTAS TEKNIK UNIVERSITAS INDONESIA  
GENAP 2007/2008**

## **PERNYATAAN KEASLIAN SKRIPSI**

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul :

### **IMPLEMENTASI MODUL PEMETAAN DAN MAPSERVER SISTEM INFORMASI BIS KAMPUS UNIVERSITAS INDONESIA**

yang dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada Program Studi Teknik Elektro Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia, sejauh yang saya ketahui bukan merupakan tiruan atau duplikasi dari skripsi yang sudah dipublikasikan dan atau pernah dipakai untuk mendapatkan gelar kesarjanaan di lingkungan Universitas Indonesia maupun di Perguruan Tinggi atau Instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Depok, 24 Juni 2008

Arie Wahyu Hidayat

NPM 04 04 03 013X

## **PENGESAHAN**

Skripsi dengan judul :

### **IMPLEMENTASI MODUL PEMETAAN DAN MAPSERVER SISTEM INFORMASI BIS KAMPUS UNIVERSITAS INDONESIA**

dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada program studi Teknik Elektro Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia dan disetujui untuk diajukan dalam sidang ujian skripsi.

Depok, 24 Juni 2008

Dosen Pembimbing,

Dr. Ir. Riri Fitri Sari, M.Sc, MM

NIP 132.127.785

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada :

**Dr. Ir. Riri Fitri Sari, M.Sc, MM**

selaku dosen pembimbing yang telah bersedia meluangkan waktu untuk memberi pengarahan, diskusi dan bimbingan serta persetujuan sehingga skripsi ini dapat selesai dengan baik.

Arie Wahyu Hidayat  
NPM 04 04 03 013X  
Departemen Teknik Elektro

Dosen Pembimbing  
Dr. Ir. Riri Fitri Sari, M.Sc, MM

**IMPLEMENTASI MODUL PEMETAAN DAN MAPSERVER SISTEM  
INFORMASI BIS KAMPUS UNIVERSITAS INDONESIA**

**ABSTRAK**

Seorang pengguna jasa bis kampus tentunya sangat ingin mengetahui posisi bis kampus yang akan dinaikinya, sehingga ia dapat melakukan aktifitas lain tanpa perlu membuang waktu lama menunggu bis kampus, atau apabila pengguna dapat mengetahui bahwa bis kampus akan segera tiba di halte fakultas tempat ia berkuliah, maka pengguna dapat segera bergegas mengejar bis tersebut. Salah satu alternatif yang dapat dilakukan adalah menerapkan Sistem Informasi Bis Kampus sehingga pengguna dapat mengetahui posisi bis kampus lewat Internet.

Skripsi ini membahas perancangan Sistem Informasi Bis Kampus modul pemetaan dan MapServer serta analisa data hasil implementasi sistem. Pada modul pemetaan, data waktu kedatangan bis yang diperoleh dari pengamatan waktu kedatangan bis kampus di lapangan, langsung diterapkan pada listing program pemetaan (ArcView), sehingga apabila ada bis kampus di suatu halte, program pemetaan dapat langsung mengatur layer posisi bis. Apabila ada bis, maka layer akan di-*overwrite* dengan koordinat sebenarnya. Sedangkan apabila tidak ada bis, maka layer akan di-*overwrite* dengan koordinat sebarang. Layer hasil *overwrite* ini yang nantinya akan dipanggil oleh modul MapServer untuk menampilkan peta posisi bis kampus yang dapat diakses oleh pengguna lewat Internet.

Data waktu kedatangan bis kampus, terutama di halte Teknik, telah diuji akurasi dan didapat nilai deviasi maksimal sebesar lima menit. Proses *overwrite layer* telah diuji pada tiga komputer dengan spesifikasi yang berbeda. Kemudian dari analisa waktu *overwrite layer* tersebut didapat bahwa semakin cepat prosesor suatu komputer, maka waktu yang diperlukan untuk *overwrite layer* semakin singkat. Pengaruh spesifikasi komputer terhadap waktu respon server terhadap *request* peta posisi bis juga telah diuji di laboratorium Mercator. Dari analisa data waktu respon tersebut didapat bahwa spesifikasi komputer cukup berpengaruh terhadap waktu respon. Tanggapan user terhadap Sistem Informasi Bis Kampus Universitas Indonesia cukup beragam, namun mayoritas user mengatakan bahwa Sistem Informasi Bis Kampus UI berguna, GUI yang ditampilkan cukup bagus, tingkat akurasi cukup tinggi, dan memiliki waktu respon yang cepat. Selain itu pengguna juga berharap agar rancangan ini dapat benar-benar direalisasikan dengan desain yang lebih stabil, data posisi bis yang berubah secara *real time*, serta tampilan yang lebih menarik.

**Kata kunci : bis kampus, MapServer, program pemetaan, *overwrite layer*.**

Arie Wahyu Hidayat  
NPM 04 04 03 013X  
Electrical Engineering Departement

Counsellor  
Dr. Ir. Riri Fitri Sari, M.Sc, MM

**IMPLEMENTATION OF MAPPING AND MAPSERVER MODULE OF  
COLLEGE BUS INFORMATION SYSTEM UNIVERSITY OF INDONESIA**

**ABSTRACT**

Every University bus user would like to know the position of the University bus, so they can cut the waiting time or perform other activities while waiting. They can rush to the nearest stopping point once they know that the bus is about to arrive. Information about University bus position can be obtained by implementing University of Indonesia Bus Information System. With this system, user will be able to know the position of UI bus by accessing a the bus information server from the Internet using their Laptop or PDA's.

This final project paper explains the design of mapping and MapServer module of UI Bus Information System and the analysis of its implementation's data. In mapping module, the data of bus arrival time which is obtained from the field survey is implemented directly into mapping program listing (ArcView). If there is a bus at a stopping point, the mapping program will overwrite the bus layer coordinate with its real coordinate. Meanwhile, if there is no college bus at a stopping point, mapping program will overwrite the bus layer coordinate with a dummy coordinate, which is located outside UI region. The overwritten layer is then called by the MapServer module, so this map of the bus position can be accessed by users from the Internet.

The accuracy of bus arrival time's data, especially at stopping point at the Faculty of Engineering, has been tested and it has been found that the data has maximum deviation of five minutes. The layer overwrite process is also has been tested in three different computers with different specification. The overwrite time's data shows that the faster computer's processor will gives a faster overwrite time. The influence of computer specification against server response time has also been tested at Mercator's laboratory. From the analysis of response time's data, we know that the faster computer's processor will gives a faster response time, although not significant. The opinion from the users about this UI Bus Information System is varied. However, most of them stated that UI Bus Information System is useful, have a moderate graphical user interface, pretty good accuracy, and fast response time. Users also hope that this design could be really implemented with more stable design and better display. It is expected that in the future this system can support real time data (context aware) instead of statistical average of bus arrival time.

**Keywords : college bus, MapServer, mapping program, layer overwrite.**

# DAFTAR ISI

IMPLEMENTASI MODUL PEMETAAN DAN MAPSERVER SISTEM INFORMASI BIS KAMPUS UNIVERSITAS INDONESIA .....	ii
PERNYATAAN KEASLIAN SKRIPSI .....	iii
PERNGESAHAN.....	iv
UCAPAN TERIMA KASIH .....	v
ABSTRAK .....	vi
ABSTRACT .....	vii
DAFTAR ISI .....	viii
DAFTAR GAMBAR.....	x
DAFTAR TABEL .....	xi
DAFTAR GRAFIK .....	xi
DAFTAR ISTILAH.....	xiii
BAB 1 PENDAHULUAN.....	1
1.1. LATAR BELAKANG.....	1
1.2. TUJUAN.....	2
1.3. BATASAN MASALAH .....	2
1.4. SISTEMATIKA PENULISAN .....	2
BAB 2 TEORI DASAR.....	4
2.1. SISTEM INFORMASI GEOGRAFIS .....	4
2.2. MAPSERVER .....	8
2.2.1. Komponen Pembentuk MapServer.....	10
2.2.2. MapFile.....	12
BAB 3 SISTEM INFORMASI BIS KAMPUS MODUL PEMETAAN DAN MAPSERVER.....	15
3.1. INSTALASI TOOLS.....	15
3.2. PERENCANAAN DESAIN.....	16
3.3. PEMBUATAN DATA SPASIAL.....	19
3.4. PEMBUATAN MAPFILE .....	21
3.5. PEMBUATAN PROGRAM PEMETAAN.....	23



3.5. PEMBUATAN HALAMAN WEB .....	25
BAB 4 ANALISA IMPLEMENTASI SISTEM INFORMASI BIS KAMPUS	
UNIVERSITAS INDONESIA .....	28
4.1. PENGUMPULAN DATA DAN ANALISA DATA KEHADIRAN BIS	
KAMPUS .....	28
4.2. PENGUMPULAN DATA DAN ANALISA WAKTU OVERWRITE	
LAYER.....	29
4.3. PENGUMPULAN DATA DAN ANALISA WAKTU AKSES SERVER...	30
4.4. TANGGAPAN USER TERHADAP IMPLEMENTASI SISTEM	
INFORMASI BIS KAMPUS.....	32
4.5. PENGEMBANGAN IMPLEMENTASI SISTEM INFORMASI BIS	
KAMPUS KE DEPAN.....	34
BAB 5 KESIMPULAN .....	36
DAFTAR ACUAN .....	37
DAFTAR PUSTAKA.....	39
LAMPIRAN .....	40

## DAFTAR GAMBAR

Gambar 2.1 Contoh proses registrasi data raster menggunakan MapInfo.....	5
Gambar 2.2 Contoh peta vektor sederhana yang tersusun atas bentuk titik, garis, dan polygon [2].....	6
Gambar 2.3 Penyusun layer data spasial .....	7
Gambar 2.4 Prinsip tumpang tindih pada layer data spasial.....	8
Gambar 2.5 Arsitektur aplikasi web-based GIS dengan MapServer [5] .....	9
Gambar 2.6 Komponen Pembentuk MapServer.....	10
Gambar 3.1 <i>Use Case Diagram</i> dari sistem informasi bis kampus modul pemetaan dan MapServer .....	16
Gambar 3.2 <i>Class Diagram</i> modul pemetaan .....	17
Gambar 3.3 <i>Statechart diagram</i> pada modul pemetaan dan MapServer Sistem Informasi Bis Kampus.....	18
Gambar 3.4 <i>Sequence diagram</i> dari Sistem Deteksi Bis Kampus modul pemetaan dan MapServer .....	19
Gambar 3.5 Hasil akhir penerapan mapfile pada Sistem Informasi Bis Kampus .....	22
Gambar 3.6 Algoritma Program Pemetaan metode statistik kedatangan bis .....	23
Gambar 3.7 Halaman Beranda (Home) pada Sistem Informasi Bis Kampus .....	25
Gambar 3.8 Tampilan dan peta yang dimiliki pada halaman Peta Raster .....	26
Gambar 3.9 Tampilan pada halaman Bantuan.....	27
Gambar 3.10 Tampilan pada halaman Tentang Kami (About Us).....	27
Gambar 3.11 Modul-modul dalam Sistem Informasi Bis Kampus .....	34
Gambar 3.12 Algoritma overwrite layer menggunakan tabel database.....	35

## DAFTAR TABEL

Tabel 3.1 Parameter peta digital pada sistem informasi bis kampus.....	22
Tabel 4.1 Jadwal kedatangan bis kampus di halte Teknik berdasarkan pengamatan di lapangan.....	28
Tabel 4.2 Spesifikasi komputer Hamburg, Cheetah, dan Bening-01 .....	29
Tabel 4.3 Data waktu overwrite layer pada Hamburg, Bening-01, dan Cheetah .....	30
Tabel 4.4 Data waktu respon posisi bis kampus pada Hamburg dan Bening-01 .....	31
Tabel 4.5 Hasil Kuisisioner Sistem Informasi Bis Kampus.....	32
Tabel 4.6 Hasil pengolahan data kuisisioner Sistem Informasi Bis Kampus .....	33

## DAFTAR GRAFIK

Tabel 4.1 Grafik waktu overwrite layer pada delapan menit pertama.....	30
Tabel 4.2 Grafik waktu respon server terhadap request dari client .....	31
Tabel 4.3 Rentang nilai optimis dan optimis responden .....	34

## DAFTAR ISTILAH

ArcView	Suatu tool Sistem Informasi Geografis yang dikembangkan oleh <i>Enviromental System Research Institute (ESRI)</i> .
Bis kampus	Bis khusus yang diediakan oleh Universitas Indonesia untuk melayani keperluan transportasi warga UI di dalam wilayah UI.
<i>GIS tools</i>	Tools/program yang digunakan untuk keperluan Sistem Informasi Geografis, dapat bersifat <i>standalone</i> atau <i>web based</i> .
Halaman <i>web</i>	Halaman <i>web site</i> yang dapat langsung dinikmati oleh user.
<i>Layer</i>	Data citra (raster atau vektor) yang memiliki koordinat geografis.
Linux	Sistem Operasi yang bersifat gratis dan open source dengan berbagai distro.
MapFile	Suatu data teks ASCII yang berekstensi <i>.map</i> dan memiliki parameter-parameter peta digital yang akan dibuat oleh MapServer.
MapServer	Aplikasi <i>freeware</i> dan <i>open source</i> yang memungkinkan kita menampilkan data spasial (peta) di <i>web</i> .
Overwrite layer	Menghapus <i>layer</i> yang sebelumnya kemudian menggantinya dengan <i>layer</i> baru.
Peta digital	Citra peta yang ditampilkan oleh komputer.

Program pemetaan	GIS tool yang memiliki fitur pemrograman dan telah diimplementasikan oleh algoritma manipulasi layer.
Raster	Citra peta digital dari foto satelit atau foto udara yang memiliki dimensi keruangan.
<i>Script Avenue</i>	Bahasa pemrograman pada ArcView.
String	Variabel dalam bahasa pemrograman yang berisi karakter
Vektor	Citra titik, garis, atau poligon yang memiliki informasi geografis.
<i>Web browser</i>	<i>Tool/program</i> yang digunakan untuk menjelajahi ( <i>browsing</i> ) Internet.
<i>Web server</i>	Sistem komputer yang menyimpan berbagai informasi yang dapat diakses oleh <i>user</i> lewat Internet.
<i>Wine</i>	<i>Emulator</i> untuk menjalankan program-program Windows di Linux.

# BAB 1

## PENDAHULUAN

### 1.1 LATAR BELAKANG

Transportasi bisa dikatakan sebagai salah satu kebutuhan pokok manusia. Hal ini dikarenakan manusia bukan makhluk stasioner, mereka perlu melakukan perpindahan dari satu tempat ke tempat lain untuk memenuhi kebutuhan mereka.

Berbagai jenis angkutan umum akhirnya diciptakan untuk memenuhi kebutuhan transportasi manusia, baik yang beroperasi di darat, laut ataupun udara. Akan tetapi di balik berkembangnya angkutan umum ini, terdapat satu masalah yang perlu diatasi oleh penyedia jasa maupun Pemerintah, yaitu informasi tentang posisi suatu angkutan umum.

Sebagai contoh, seorang pengguna jasa bis kampus tentunya sangat ingin mengetahui posisi bis kampus yang akan dinaikinya, sehingga ia dapat melakukan aktifitas lain tanpa perlu membuang waktu lama menunggu bis kampus, atau apabila pengguna dapat mengetahui bahwa bis kampus akan segera tiba di halte Fakultas tempat ia berkuliah, maka pengguna dapat segera bergegas mengejar bis tersebut sehingga tidak perlu menunggu bis berikutnya.

Salah satu alternatif yang dapat dilakukan adalah menerapkan sistem informasi kendaraan berbasis *web* pada angkutan umum. Data yang diperoleh dari hasil pelacakan atau statistik kehadiran bis ini kemudian diimplementasikan ke suatu *web server* yang dapat diakses oleh pengguna jasa angkutan umum, sehingga mereka dapat melihat posisi kendaraan tersebut untuk kemudian dijadikan referensi dalam pengambilan keputusan yang efisien dan efektif.

Berbagai *tools* Sistem Informasi Geografis, baik yang berbasis *web* seperti MapServer maupun *standalone* seperti ArcView, memungkinkan kita untuk membuat suatu rancangan Sistem Informasi Bis Kampus yang dapat memperlihatkan posisi bis terakhir via *web* yang terus berubah secara dinamis berdasarkan perubahan database acuan atau waktu kehadiran bis kampus.

## 1.2 TUJUAN

Tujuan pelaksanaan skripsi ini adalah mengimplementasikan sistem informasi bis kampus, terutama modul manipulasi *layer* (pemetaan) dan mempublikasikan peta beserta posisi bis tersebut ke dalam *web* menggunakan MapServer, sehingga pengguna dapat mengetahui posisi bis kampus.

## 1.3 BATASAN MASALAH

Fokus dari skripsi ini adalah membentuk algoritma manipulasi layer data spasial dari data kehadiran bis kampus pada halte-halte di Universitas Indonesia. Selain itu, dalam skripsi ini dijabarkan pula tentang pemanfaatan MapServer sehingga dapat terbentuk suatu peta digital yang dapat digunakan untuk melihat posisi bis kampus.

## 1.4 SISTEMATIKA PENULISAN

Secara sistematis, skripsi ini terdiri dari 5 (lima) bab dengan perincian sebagai berikut :

### BAB I PENDAHULUAN

Bab ini terdiri dari latar belakang, tujuan penulisan, batasan masalah, metode penelitian, dan sistematika penulisan.

### BAB II TEORI DASAR

Pada bab ini dijelaskan secara singkat mengenai Sistem Informasi Geografis dan MapServer.

### BAB III SISTEM INFORMASI BIS KAMPUS MODUL PEMETAAN DAN MAPSERVER

Pada bab ini akan dijelaskan tentang tahapan-tahapan perancangan modul pemetaan dan MapServer pada Sistem Informasi Bis Kampus Universitas Indonesia yang meliputi instalasi tools, perencanaan, membuat data spasial, pembuatan MapFile sehingga dihasilkan suatu peta digital, pembuatan program pemetaan (manipulasi layer data spasial) berdasarkan data kehadiran bis, sehingga menghasilkan suatu peta digital yang dapat memberitahukan posisi bis kampus serta pembuatan halaman *web*.



**BAB IV ANALISA IMPLEMENTASI SISTEM INFORMASI BIS  
KAMPUS**

Bab ini berisi data dan analisa tingkat akurasi sistem serta pengaruh waktu *overwrite* dan waktu akses sistem terhadap spesifikasi komputer. Selain itu pada bab ini akan diberitahukan tanggapan pengguna dalam pemakaian Sistem Informasi Bis Kampus ini.

**BAB V KESIMPULAN**

Bab ini berisi kesimpulan-kesimpulan yang diperoleh dari perancangan serta uji coba yang telah dilakukan.

## **BAB II**

### **TEORI DASAR**

#### **2.1 SISTEM INFORMASI GEOGRAFIS**

Sistem Informasi Geografis (SIG) adalah sistem informasi khusus yang mengelola data yang memiliki informasi spasial (bereferensi keruangan). Sistem Informasi Geografis juga dapat didefinisikan sebagai sistem komputer yang memiliki kemampuan untuk membangun, menyimpan, mengelola dan menampilkan informasi bereferensi geografis. Para praktisi juga memasukkan orang yang membangun dan mengoperasikannya dan data sebagai bagian dari sistem ini [1].

Teknologi Sistem Informasi Geografis dapat digunakan dalam penyelidikan ilmiah, manajemen sumber daya, manajemen aset, perencanaan tata kota, kartografi, kriminologi, sejarah, pemasaran, dan logistik [2]. Sebagai contoh, penerapan SIG memungkinkan perencana untuk secara cepat menghitung waktu tanggap darurat saat terjadi bencana alam. Selain itu, SIG juga dapat digunakan oleh pengguna jasa logistik untuk mengetahui telah sampai dimana barang mereka.

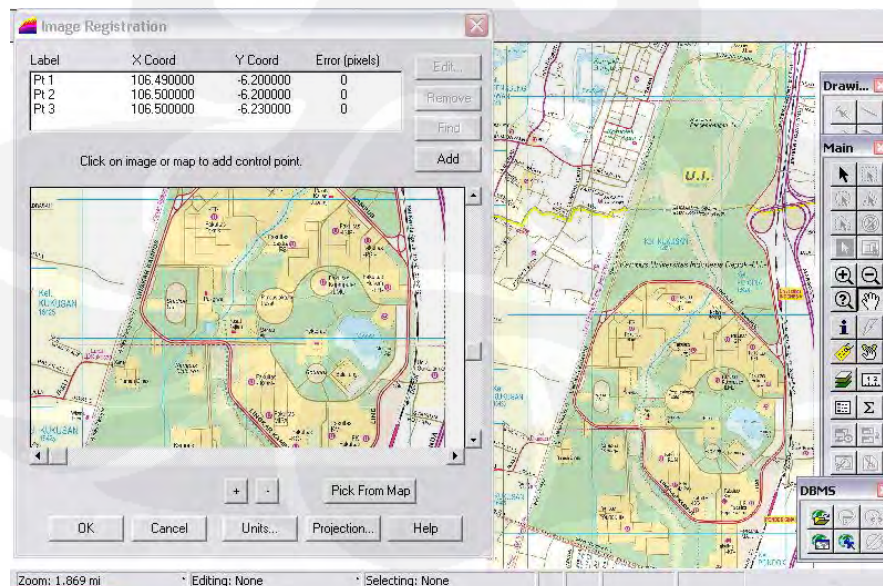
Sistem Informasi Geografis dapat dibagi menjadi dua jenis, yaitu sistem manual (analog) dan sistem otomatis (yang berbasis digital komputer). Perbedaan yang paling mendasar terletak pada cara pengelolaannya. Sistem manual sendiri telah dilakukan sejak jaman prasejarah. Hal ini dibuktikan dengan ditemukannya gambar binatang beserta rute migrasinya yang dibuat sekitar 15.500 tahun lalu pada sebuah gua di Lascaux, Perancis [2]. Sedangkan pada era modern, sistem manual dilakukan dengan cara menggabungkan beberapa data seperti peta, lembar transparansi, foto udara, laporan statistik dan laporan survey lapangan. Kesemua data tersebut dikompilasi dan dianalisis secara manual dengan alat tanpa komputer [3].

Sedangkan Sistem Informasi Geografis otomatis telah menggunakan komputer sebagai sistem pengolah data melalui proses digitasi. Sumber data

digital dapat berupa citra satelit atau foto udara digital serta foto udara yang terdigitasi [3].

Komponen utama Sistem Informasi Geografis dapat dibagi ke dalam empat komponen, yaitu: perangkat keras (*digitizer, scanner, Central Processing Unit (CPU), harddisk, dan lain-lain*), perangkat lunak (*ArcView, Idrisi, ARC/INFO, ILWIS, MapInfo, dan lain-lain*), organisasi (*manajemen*) dan pemakai (*user*). Kombinasi yang benar antara keempat komponen utama ini akan menentukan kesuksesan suatu proyek pengembangan Sistem Informasi Geografis [3].

Dalam Sistem Informasi Geografis, dikenal dua jenis data data spasial, yaitu raster dan vektor [2]. Data raster ialah segala macam data citra digital, seperti foto dari kamera digital atau data hasil *scan* sebuah peta, yang memiliki satuan unit pixel. Jenis data ini, terutama yang diambil dari citra satelit atau foto udara, lebih cenderung disukai karena apa yang pengguna lihat merupakan keadaan sesungguhnya dari lokasi tersebut pada tanggal citra diambil. Namun perlu diingat, tidak sembarang citra digital dari foto udara atau citra satelit dapat dikatakan sebagai data spasial raster. Citra tersebut perlu ditambahkan informasi tambahan (*registrasi*) berupa koordinat bumi yang mewakili suatu pixel. Biasanya diperlukan tiga titik yang berbeda dalam suatu citra digital yang harus dipetakan terhadap koordinat bumi (*lintang selatan/lintang utara dan bujur barat/bujur timur*).



**Gambar 2.1.** Contoh proses registrasi data raster menggunakan MapInfo

Berbeda dengan data raster, data vektor tersusun atas bentuk-bentuk geometri sederhana (garis, titik, dan *polygon*). Bentuk titik biasanya digunakan untuk merepresentasikan suatu lokasi yang tidak luas, seperti sumur atau mobil. Bentuk garis merupakan sekumpulan titik yang membentuk suatu kenampakan memanjang satu dimensi sederhana (garis lurus atau *polyline*) yang biasanya digunakan untuk merepresentasikan informasi linier, seperti rel kereta api, jalan raya, dan sungai. Sedangkan bentuk *polygon* biasanya digunakan untuk merepresentasikan suatu daerah yang cukup luas, seperti danau, hutan, atau luas provinsi. Bentuk *polygon* merupakan bentuk yang paling banyak digunakan dalam data spasial vektor.



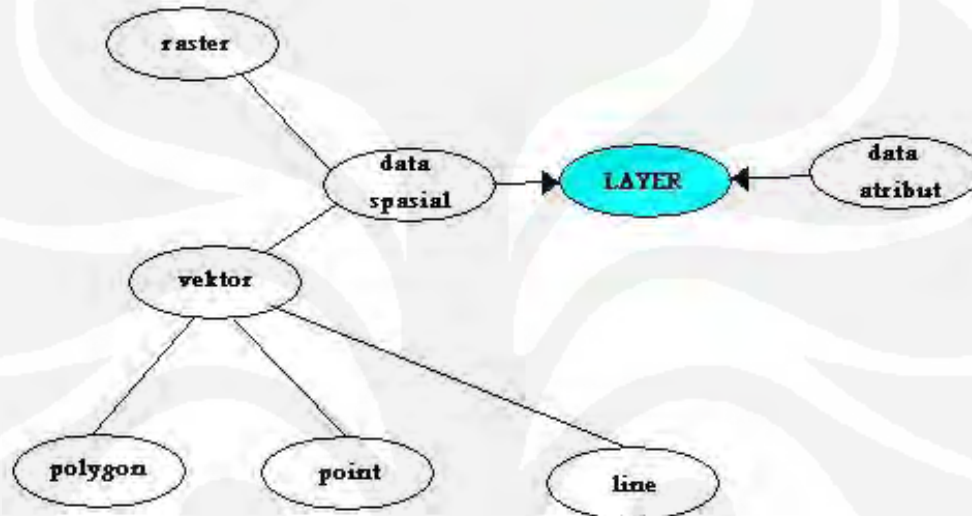
**Gambar 2.2.** Contoh peta vektor sederhana yang tersusun atas bentuk titik, garis, dan polygon [2]

Data vektor tidak perlu dipetakan (registrasi) seperti data raster. Hal ini karena biasanya data vektor langsung dibuat menggunakan suatu program Sistem Informasi Geografis yang secara otomatis memetakan bentuk geometri tersebut.

Selain data spasial, dalam Sistem Informasi geografis juga dikenal data atribut, atau yang lebih dikenal dengan sebutan label. Penyajian data atribut bersifat menempel (*embedded*) dengan data spasial. Dengan kata lain, data atribut tidak dapat ada sendiri tanpa data spasial. Hal ini dikarenakan isi dari data atribut merupakan informasi yang terkait dengan data spasial yang telah ada. Sebagai ilustrasi, suatu data vektor *polygon* hanya dapat menunjukkan pencakupan dari

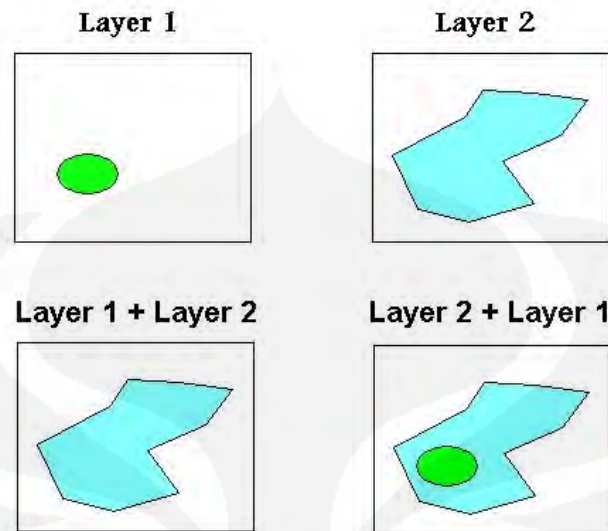
suatu propinsi, oleh karena itu diperlukan data atribut yang dapat memberitahukan informasi tambahan seperti populasi penduduk di propinsi tersebut.

Data spasial dan data atribut tersebut kemudian disatukan menjadi suatu *layer*. *Layer* inilah yang menjadi masukan utama dalam *tools* Sistem Informasi Geografis untuk menampilkan peta digital. Jika disimpulkan, susunan/prinsip dari *layer* dapat dilihat pada Gambar 2.3 di bawah ini.



**Gambar 2.3.** Penyusun layer data spasial

Suatu *layer* dapat ditampilkan dengan *layer* lain dalam suatu peta digital dengan prinsip tumpang tindih (*layer* atas akan menutupi *layer* bawah). Dengan kata lain, apabila terdapat dua *layer* dalam satu koordinat, dimana *layer* pertama berupa *polygon* dengan luas yang lebih kecil daripada *layer* kedua yang juga berupa *polygon*, maka *layer* kedua akan benar-benar menutupi *layer* pertama apabila diletakkan di atas *layer* pertama. Sebaliknya, *layer* pertama akan terlihat sebagai bagian dari *layer* kedua apabila *layer* pertama diletakkan di atas *layer* kedua seperti Gambar 2.4 di bawah.



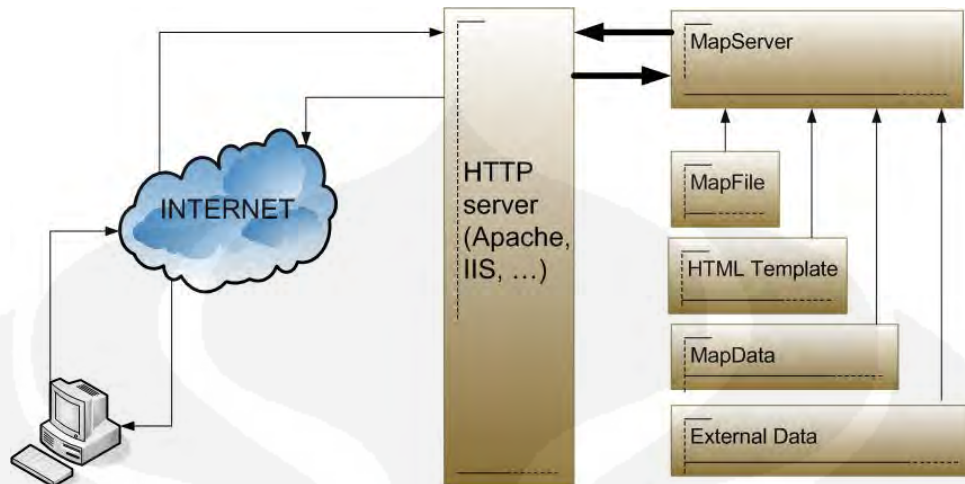
**Gambar 2.4.** Prinsip tumpang tindih pada layer data spasial

## 2.2 MAPSERVER

MapServer merupakan aplikasi *freeware* dan *open source* yang memungkinkan kita menampilkan data spasial (peta) di *web*. Aplikasi ini pertama kali dikembangkan di Universitas Minnesota, Amerika Serikat untuk proyek ForNet (sebuah proyek manajemen sumber daya alam) yang disponsori oleh NASA [4].

Pengembangan MapServer menggunakan berbagai aplikasi *open source* atau *freeware* seperti Shapelib (<http://shapelib.maptools.org>) untuk baca/tulis format data Shapefile, FreeType untuk *me-render* karakter, GDAL/OGR untuk baca/tulis berbagai format data vektor maupun raster, dan Proj4 untuk menangani berbagai proyeksi peta.

Pada dasarnya, MapServer adalah sebuah program CGI (*Common Gateway Interface*). Program tersebut akan dieksekusi di *web server*, dan berdasarkan beberapa parameter tertentu (terutama konfigurasi dalam bentuk file \*.MAP), akan menghasilkan data yang kemudian akan dikirim ke *web browser*, baik dalam bentuk gambar peta ataupun bentuk lain.



**Gambar 2.5.** Arsitektur aplikasi web-based GIS dengan MapServer [5]

Berdasarkan arsitektur MapServer sebagai program CGI pada Gambar 2.5, dapat kita lihat bahwa *browser (client)* akan mengirimkan *request* ke *web server* dalam bentuk *request* yang berhubungan dengan spasial. Kemudian oleh *web server request* ini dikirimkan ke *server* aplikasi (yang telah dibangun dengan menggunakan *script* yang tersedia) dan MapServer (program CGI). Setelah itu MapServer akan membaca *mapfile*, data peta, dan data eksternal untuk membentuk sebuah gambar yang sesuai dengan *request*. Setelah gambar di-*render*, file *image* yang bersangkutan akan dikirim ke *web server* hingga akhirnya sampai ke *client* sesuai dengan format tampilan yang telah diatur dalam *template*[5].

Dalam buku Panduan Menggunakan MapServer yang ditulis oleh Ir. Ruslan Nuryadin [6] diberitahukan beberapa fitur MapServer, yaitu:

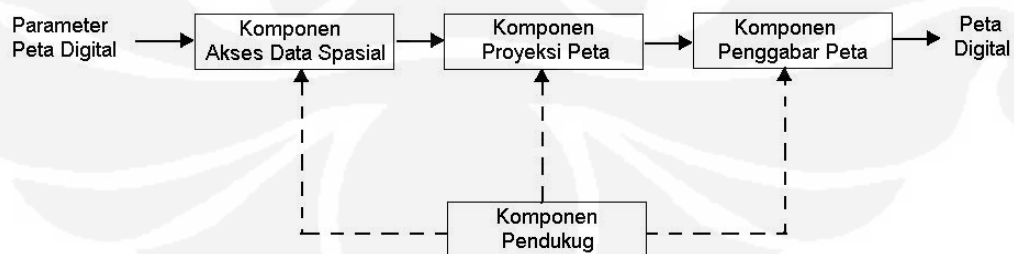
- Menampilkan data spasial dalam format vektor seperti *Shapefile*, *ArcSDE*, *PostGIS* dan berbagai format data vektor lain dengan menggunakan library *OGR*.
- Menampilkan data spasial dalam format raster seperti *TIFF/GeoTIFF*, *EPPL7* dan berbagai format data raster lain dengan menggunakan library *GDAL*.
- Menggunakan *quadtree* dalam *indexing* data spasial, sehingga operasi-operasi spasial dapat dilakukan dengan cepat.
- Dapat dikembangkan (*customizable*) dengan tampilan keluaran yang dapat diatur menggunakan file-file *template*.

- Dapat melakukan seleksi objek berdasar nilai, atau berdasar titik area, atau berdasar sebuah objek spasial tertentu.
- Mendukung *rendering* karakter berupa *font* TrueType.
- Mendukung penggunaan data raster maupun vektor yang dibagi-bagi (*tiled*) menjadi sub bagian yang lebih kecil sehingga proses untuk mengambil dan menampilkan gambar dapat dipercepat.
- Dapat menggambarkan elemen peta (skala grafis, peta indeks dan legenda peta) secara otomatis.
- Dapat menggambarkan peta tematik yang dibangun menggunakan ekspresi logika maupun ekspresi reguler.
- Konfigurasi dapat diatur secara langsung melalui parameter yang ditentukan pada URL.
- Dapat menangani berbagai sistem proyeksi secara langsung.

Saat ini, selain dapat mengakses MapServer sebagai program CGI, kita dapat mengakses MapServer sebagai modul MapScript, melalui berbagai bahasa skrip seperti PHP, Perl, Python atau Java. Akses fungsi-fungsi MapServer melalui skrip akan memudahkan dalam pengembangan karena pengembang dapat memilih bahasa yang paling familiar.

### 2.2.1 Komponen Pembentuk MapServer

Secara umum, komponen pembentuk MapServer dapat dibagi menjadi empat komponen, yaitu komponen untuk akses data spasial, komponen penggambar peta, komponen proyeksi peta, dan komponen pendukung [6] seperti pada Gambar 2.6 di bawah ini.



**Gambar 2.6.** Komponen Pembentuk MapServer



### 2.2.1.1 Komponen Akses Data Spasial

Komponen ini bertugas untuk mengani baca/tulis data spasial, baik yang tersimpan sebagai file maupun tersimpan pada DBMS. Library yang digunakan antara lain Shapelib dan GDAL/OGR.

#### 1) Shapelib

Shapelib merupakan *library* yang ditulis dalam bahasa C untuk keperluan baca/tulis format data shapefile (\*.shp) yang didefinisikan oleh ESRI (*Environmental System Research Institute*). Format shapefile umum digunakan oleh berbagai aplikasi Sistem Informasi Geografis untuk menyimpan data vektor sederhana dengan atribut. Pada MapServer, format data shapefile merupakan format data *default*. Kode shapelib sekarang dikelola oleh Frank Warmerdam, salah seorang kontributor yang banyak berkontribusi pada proyek-proyek aplikasi berbasis peta *open source* [7].

#### 2) GDAL/OGR

*Geographic data Abstraction Library* atau GDAL merupakan library yang berfungsi sebagai penerjemah (*translator*) untuk berbagai format data raster. MapServer menggunakan *library* GDAL untuk mengakses 81 format data raster mulai dari Arc/Info ASCII Grid hingga X11Pixmap [8].

OGR merupakan *library* dengan fungsionalitas yang identik dengan GDAL, hanya saja *library* ini diperuntukan kepada beragam format data vektor. Kode OGR sekarang ini digabung dalam kode library GDAL dan dapat digunakan untuk keperluan akses data vektor mulai dari Arc/Info *Binary Coverage* hingga Informix DataBlade, termasuk ESRI shapefile dan MapInfo *table* yang sangat banyak digunakan [9].

### 2.2.1.2 Komponen Penggambar Peta

MapServer akan mengirimkan tampilan peta berupa gambar. Kita dapat memilih apa format data gambar yang akan digunakan. Di bawah ini akan dijabarkan beberapa komponen yang berperan dalam membentuk gambar peta yang dihasilkan oleh MapServer. *Library* yang digunakan antara lain libpng, libjpeg, GD, dan FreeType [5].

#### 1) libpng

Libpng merupakan *library* yang digunakan untuk baca/tulis gambar dalam format PNG [10].

2) libjpeg

Libjpeg merupakan *library* yang digunakan untuk baca/tulis gambar dalam format JPG/JPEG [11].

3) GD

*Library* GD digunakan oleh MapServer untuk menggambar objek geografis seperti garis, poligon atau bentuk geometris lain. GD juga dapat digunakan untuk menghasilkan gambar dalam format PNG, JPEG, selain menggunakan libpng atau libjpeg secara langsung [12].

4) FreeType

FreeType merupakan *library* yang digunakan MapServer untuk menampilkan tulisan *font* TrueType [13].

### 2.2.1.3 Komponen Proyeksi Peta

*Library* Proj.4 digunakan MapServer untuk menangani sistem proyeksi peta.

Aplikasi ini dikembangkan pertama kali oleh Gerald Evenden [14].

### 2.2.1.4 Komponen Pendukung

1) Zlib

Zlib dibutuhkan oleh *library* GD untuk keperluan kompresi data gambar [15].

2) Regex

*Library* ini digunakan mapServer untuk menangani ekspresi regular.

## 2.2.2 MapFile

Dalam menjalankan fungsionalitasnya, MapServer selalu memerlukan mapfile (suatu file teks ASCII yang berekstensi \*.map) yang mendeskripsikan apa dan dimana sumber datanya, dan bagaimana data tersebut harus ditampilkan. Selain itu, di dalam file teks yang penting ini juga terkandung mekanisme konfigurasi dasar untuk MapServer. Segala sesuatu yang terkait dengan aplikasinya juga didefinisikan di dalam file teks ini, meskipun sebagian besar pilihan juga bisa dirubah melalui variabel CGI. Akan tetapi, selain menggunakan

file \*.map ini, pengguna masih tetap bisa menggunakan MapServer dengan memanfaatkan mapscript.

#### 2.2.2.1 Ketentuan Penulisan Mapfile

Sebelum benar-benar menggunakan MapServer untuk pertama kalinya, tugas pengguna adalah membuat sebuah file teks berekstensi \*.map (mapfile). Mapfile ini memiliki ketentuan-ketentuan sebagai berikut [6].

- Tidak bersifat *case-sensitive*.
- Penulisan string yang berisi beberapa karakter non-alphanumerik (di luar karakter huruf dan angka) atau *keywords* milik MapServer harus diapit oleh tanda petik ("").
- Setiap mapfile dapat digunakan untuk mendefinisikan maksimal 50 *layer* data spasial.
- Penulisan *string* untuk *path* dapat dilakukan secara absolut atau relatif terhadap lokasi (sub-direktori) dimana mapfile yang bersangkutan berada.
- Susunan atau isi mapfile memiliki hirarki struktur dengan objek "MAP" berkedudukan sebagai inti (*root*). Sementara objek-objek yang lain berada di bawah objek ini.
- Penulisan nama atribut (*field*) diapit oleh tanda kurung siku. Sementara nama atribut itu sendiri bersifat *case-sensitive*.

#### 2.2.2.2 Objek Dalam Mapfile

Dalam setiap mapfile, terdapat beberapa definisi objek dengan objek "map" sebagai *root*-nya. Pendefinisian ini selalu diawali oleh *keyword* milik objek itu sendiri, diikuti oleh blok (baris-baris detil definisi objek), dan diakhiri oleh *keyword* "END". Dengan demikian, pendefinisian objek-objek lainnya akan berada dalam blok definisi objek "map" [17].

##### 1) MAP

Objek ini mendefinisikan objek master (objek yang menyimpan semua objek lain yang berada di dalamnya) milik mapfile yang bersangkutan. Selain itu, objek *root* ini juga mendefinisikan objek dan parameter-parameter

peta/aplikasi seperti config, datapattern, debug, status, units, size, extent fontset, imagecolor, layer, legend dan sebagainya.

2) LAYER

Objek ini mendefinisikan layer-layer yang kemudian membentuk tampilan peta dijitalnya. Layer-layer digambarkan sesuai dengan urutan kemunculannya (penulisan) di dalam mapfile terkait. Dengan demikian, layer pertama akan diletakkan di dasar sehingga tampilannya bisa tertutupi oleh tampilan layer berikutnya. Sementara itu, layer terakhir (ditulis paling bawah) akan muncul paling atas dengan menutupi layer-layer di bawahnya.

3) CLASS

Objek ini digunakan untuk mendefinisikan kelas-kelas tematik untuk suatu layer yang ditentukan. Oleh karena itu, setiap layer minimal memiliki satu class.

4) LABEL

Objek ini digunakan untuk mendefinisikan label yang kemudian sering dipakai sebagai anotasi (teks) unsur-unsur spasial. meskipun demikian, label juga bisa digunakan sebagai simbol melalui pemanfaatan font truetype.

5) STYLE

Objek ini digunakan untuk menyimpan parameter-parameter simbol yang dipakai. dengan adanya objek ini, setiap kelas dapat memiliki simbol dengan tipe, ukuran, dan warna tersendiri.

## **BAB 3**

### **SISTEM INFORMASI BIS KAMPUS MODUL PEMETAAN DAN MAPSERVER**

Sistem informasi bis kampus memungkinkan pengguna melihat posisi bis kampus lewat Internet. Hal ini dikarenakan posisi bis telah diketahui melalui data kedatangan bis pada tiap halte. Data ini kemudian diimplementasikan pada *listing* program, sehingga dapat menghasilkan peta posisi bis kampus yang cukup akurat.

Pada bab ini akan dibahas rancangan modul pemetaan dan MapServer pada sistem informasi bis kampus, sehingga didapat peta digital yang menunjukkan posisi bis kampus terakhir. Hal-hal yang harus dilakukan untuk mencapai tujuan itu adalah instalasi *tools*, perencanaan desain, membuat data spasial, pembuatan MapFile, membuat program pemetaan, dan membuat halaman *web*.

#### **3.1 INSTALASI TOOLS**

Rancangan Sistem Informasi Bis Kampus ini dibuat pada Linux Fedora 7. Tools yang diperlukan adalah program pemetaan yang dilengkapi dengan programming *tool*, dan MapServer. Dalam desain kali ini, program pemetaan yang digunakan adalah ArcView GIS 3.3 dan MapServer yang digunakan adalah bundel FGS Linux (fgs-mapserver\_5.0.2-fgs\_1.0.0-linux-i386.bin) yang diunduh melalui situs <http://www.maptools.org/>.

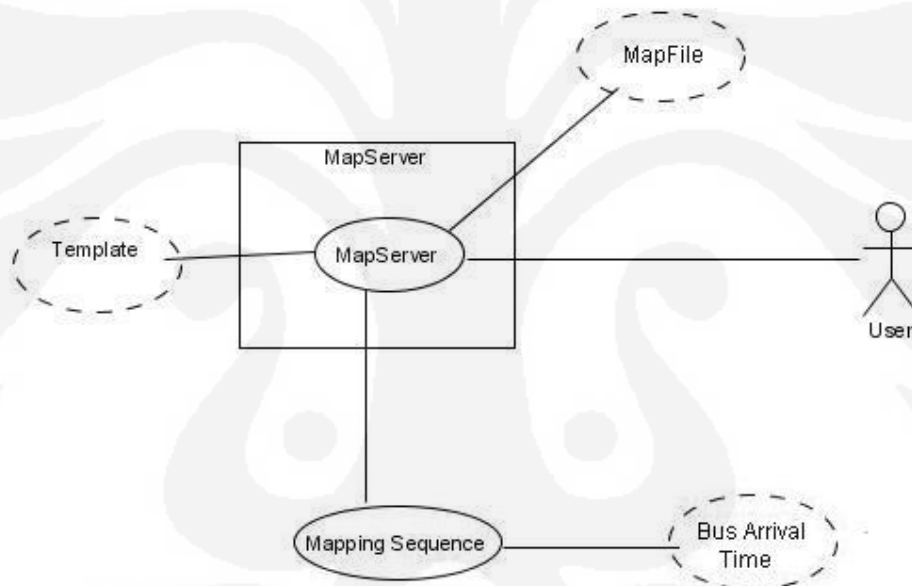
Instalasi bundel FGS dilakukan lewat Terminal. Pada proses instalasi kita dapat menentukan *path* instalasi dan *port* yang digunakan. Salah satu komponen pembentuk FGS Linux adalah Apache-base-2.2 dan akan ikut ter-*install* bersama dengan komponen FGS lainnya. Akan tetapi, instalasi Apache pada FGS tidak akan menghilangkan instalasi Apache yang telah ada sebelumnya. Dengan kata lain, setelah bundel FGS ter-*install*, pengguna dapat memilih ingin menggunakan Apache pada FGS atau Apache yang telah ada sebelumnya.

Selanjutnya sebelum dilakukan instalasi ArcView, maka pengguna harus terlebih dahulu meng-*install* Wine Windows Emulator. Hal ini dikarenakan

ArcView merupakan *mapping tool* untuk *platform* Windows, dan tidak akan dapat berjalan pada platform Linux tanpa suatu *emulator*. Wine akan menciptakan suatu *Virtual Windows Environment*, sehingga pengguna dapat langsung meng-*install* program Windows pada *platform* Linux.

### 3.2 PERENCANAAN DESAIN

Rancangan sistem informasi bis kampus dibuat berdasarkan *tools* yang disediakan. Oleh karena itu tahap perencanaan desain dilakukan setelah instalasi *tools*. Orientasi pemikiran dalam perencanaan kali ini adalah dapat mencapai tujuan menggunakan *tools* yang ada. Berdasarkan orientasi tersebut, dapat dibuat *use case diagram* seperti berikut:

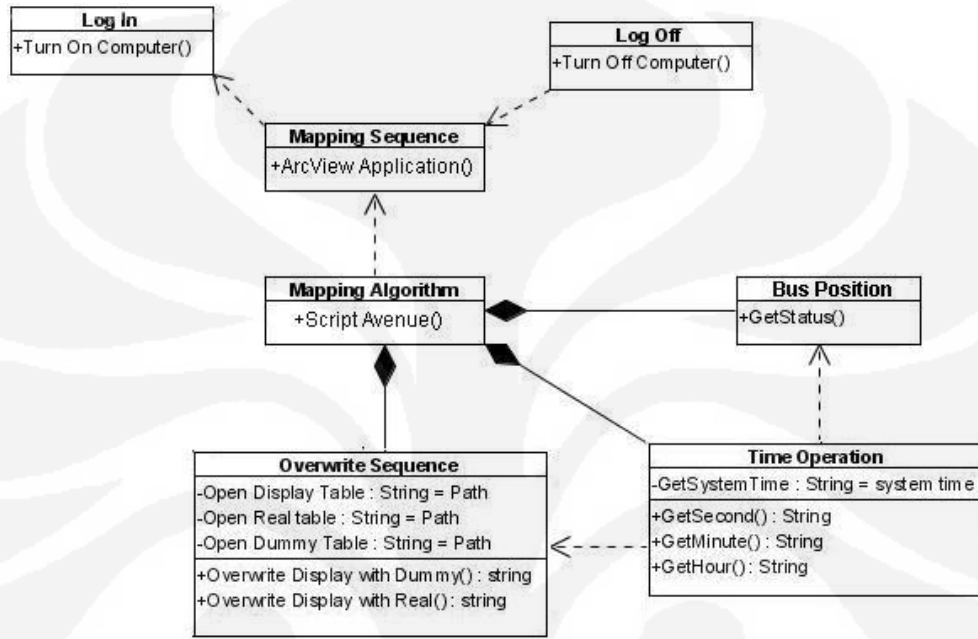


**Gambar 3.1.** *Use Case Diagram* dari sistem informasi bis kampus modul pemetaan dan MapServer

Dari *use case* di atas dapat diketahui bahwa *mapping sequence* dipengaruhi oleh waktu kedatangan bus. Dengan kata lain, waktu kedatangan bus langsung diimplementasikan pada program pemetaan. Program inilah yang mengendalikan keadaan layer posisi bus kampus. Hasil dari manipulasi *layer* tersebut kemudian akan dipanggil oleh MapServer agar dapat dinikmati langsung oleh pengguna. File *template* berfungsi sebagai pengatur tampilan, seperti menampilkan legenda arah

utara, dan skala. Sedangkan MapFile berfungsi sebagai penyimpanan parameter dan pembentuk citra peta.

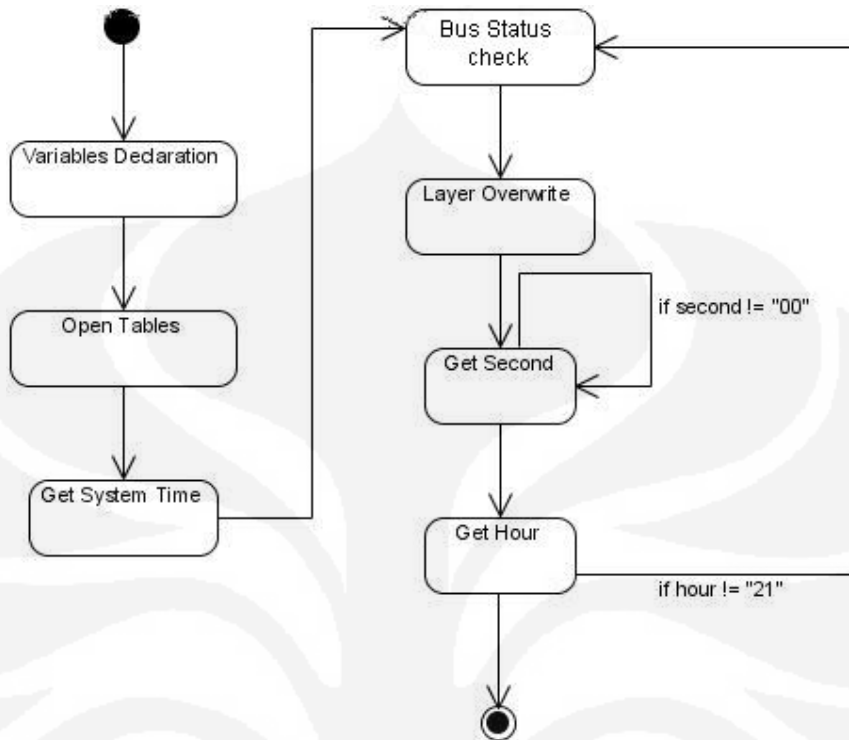
Sedangkan variabel yang diperlukan dalam algoritma pemetaan serta keterkaitan antar variabel tersebut dapat dilihat dalam class diagram berikut:



**Gambar 3.2.** Class Diagram modul pemetaan

Pada *class diagram* di atas dapat dilihat bahwa pada program yang ditulis dalam Script Avenue (fitur pemrograman pada ArcView) terdapat tiga prosedur kunci, yaitu: operasi waktu, operasi *overwrite*, dan pengecekan posisi bus. Semua variabel yang digunakan dalam algoritma pemetaan berupa *string*. Pengendali utama adalah operasi waktu, dimana fungsi ini akan mengambil waktu sistem dengan format hh:mm:ss kemudian memecahnya menjadi tiga bagian dan dimasukkan ke dalam string khusus, yaitu jam (hh), menit (mm), dan detik (ss). Berdasarkan nilai ketiga variabel inilah program pemetaan melakukan manipulasi *layer*.

*Draft* algoritma pada program pemetaan dapat diilustrasikan pada statechart diagram berikut:

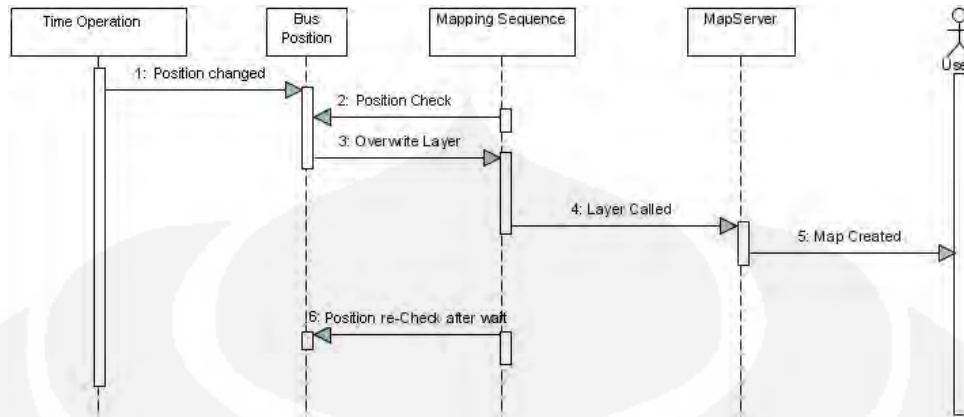


**Gambar 3.3.** *Statechart diagram* pada modul pemetaan dan MapServer Sistem Informasi Bis Kampus

Dari *statechart diagram* di atas dapat dilihat perubahan proses yang terjadi dari program pemetaan. Hal yang pertama dilakukan sejak program dijalankan adalah pendefinisian variabel, kemudian membuka semua tabel (*layer*) data spasial yang diperlukan. Sistem kemudian mengambil waktu sistem untuk kemudian dimasukkan ke dalam *string*. Kemudian sistem akan memeriksa posisi bis untuk kemudian melakukan *overwrite layer* data spasial berdasarkan posisi tersebut. Pengecekan posisi dan *overwrite layer* ini terjadi satu menit sekali dan akan terus berulang hingga pukul 21.00.

Sedangkan prinsip kerja yang lebih jelas dan waktu yang diperlukan untuk tiap proses dapat dilihat pada *sequence diagram* berikut:





**Gambar 3.4.** Sequence diagram dari Sistem Informasi Bis Kampus modul pemetaan dan MapServer

Operasi waktu akan terus aktif agar terus memperbarui nilai string waktu untuk kemudian dijadikan referensi dalam pengubahan posisi bis. Umumnya posisi bis berubah satu halte tiap satu menit. Program pemetaan (*mapping tool*) akan terus aktif. Program pemetaan akan mengecek status pada posisi bis, lalu melakukan *overwrite layer* posisi bis berdasarkan status posisi tersebut. Setelah melakukan *overwrite*, program pemetaan akan melakukan *looping* tanpa proses selama satu menit. *Looping* ini bertujuan untuk mencegah terjadinya *overwrite layer* secara terus menerus. Hal ini dikarenakan *layer* posisi bis tidak dapat dipanggil oleh MapServer saat terjadi proses *overwrite* dan akan meninggalkan pesan kesalahan.

MapServer juga terus aktif agar pengguna dapat mengakses peta digital kapanpun, kecuali saat terjadi *overwrite layer*.

### 3.3 PEMBUATAN DATA SPASIAL

Karena *output* akhir dari desain ini adalah peta digital Universitas Indonesia yang menunjukkan posisi bis, maka tentu saja diperlukan data-data spasial yang telah dipersiapkan sebelumnya. Data-data spasial ini dapat dibuat dengan menggunakan berbagai tools Sistem Informasi Geografis (*GIS tools*) seperti MapInfo atau ArcView. Dalam kasus ini, data spasial yang diperlukan dibuat dengan menggunakan MapInfo.

Layer-layer data spasial yang disiapkan yaitu border, fasilitas, danau, fakultas, politeknik, stasiun, jalan, serta layer-layer halte bis dimana diperlukan dua layer untuk satu halte untuk menunjukkan bis kampus tanda merah atau biru.

- Border

*Layer* border bertipe *polygon* yang menunjukkan batas-batas kampus Universitas Indonesia.

- Danau

*Layer* danau bertipe *polygon* dan berisi informasi tentang danau-danau di lingkungan UI.

- Fasilitas

*Layer* fasilitas bertipe *polygon* yang menunjukkan lokasi fasilitas-fasilitas di lingkungan kampus UI Depok seperti Asrama Mahasiswa, Pusat Studi Jepang (PSJ), Masjid Ukhuwah Islamiyah, Perpustakaan, Pusat Kegiatan Mahasiswa (Pusgiwa), Stadion, Rektorat, Balairung, Rotunda, Lapangan Hockey, Lapangan Tennis, Gymnasium, Balai Sidang, dan Gedung Kesejahteraan dan Fasilitas Mahasiswa (GKFM).

- Fakultas

*Layer* fakultas bertipe *polygon* yang menunjukkan fakultas-fakultas yang terdapat di kampus UI Depok, yaitu Fakultas Hukum (FH), Psikologi (FPsi), Ilmu Sosial dan Ilmu Politik (FISIP), Ilmu Budaya (FIB), Ekonomi (FE), Teknik (FT), Ilmu Komputer (Fasilkom), Kesehatan Masyarakat (FKM), Matematika dan Ilmu Pengetahuan Alam (FMIPA), dan Ilmu keperawatan (FIK).

- Jalan

*Layer* jalan bertipe *polyline* dan memberikan informasi tentang jalan-jalan di lingkungan kampus UI Depok yang dilalui oleh bis kampus.

- Politeknik

*Layer* Politeknik merupakan *polygon* dan menunjukkan lokasi kampus Politeknik Negeri Jakarta (PNJ).

- Stasiun

*Layer* stasiun bertipe *polygon* dan menunjukkan stasiun-stasiun kereta yang terdekat dari kampus UI Depok (stasiun Universitas Indonesia dan Pondok Cina).

- Posisi bis kampus

*Layer* posisi bis kampus merupakan layer-layer tipe titik yang tiap titiknya menunjukkan halte bis untuk satu rute bis kampus (merah/biru). Dengan kata lain, satu layer merupakan satu titik yang mewakili satu halte pada satu jalur bis (contoh: teknik biru).

### 3.4 PEMBUATAN MAPFILE

Setelah semua data spasial yang diperlukan telah siap, maka langkah kita selanjutnya adalah menentukan parameter peta digital yang akan dibuat oleh MapServer. Parameter peta ini yang nantinya akan dijadikan acuan utama oleh MapServer dalam menampilkan peta digital pada *web*.

Parameter peta dapat dibuat dengan dua cara, membuat mapfile (\*.map) atau dituangkan dalam MapScript, dimana MapScript merupakan suatu interface pemrograman pada MapServer yang ditulis dalam bahasa PHP dan memungkinkan kita mengembangkan peta digital interaktif.

*File* PHP/MapScript selain berfungsi sebagai penyedia parameter peta, dapat pula bertindak sebagai *template web*. *Template* berfungsi untuk menampilkan legenda, arah utara, dan skala peta. Selain itu, dalam *file* MapScript ini juga diberikan perintah yang memungkinkan peta yang dihasilkan memiliki fitur-fitur peta interaktif seperti perbesar, perkecil, fit image, dan menggeser peta.

Namun karena tujuan utama rancangan kali ini adalah menampilkan peta UI dengan posisi bis kampus, maka penggunaan MapScript dinilai kurang penting. Selain itu, implementasi MapScript juga lebih kompleks bila dibandingkan dengan MapFile biasa. Oleh karena itu dalam implelementasi, parameter peta dituangkan dalam mapfile tanpa *template*. Sehingga keluaran akhir yang dihasilkan hanya berupa gambar peta statis yang berisi informasi tentang posisi bis kampus seperti pada Gambar 3.5 berikut.



**Gambar 3.5.** Hasil akhir penerapan mapfile pada Sistem Informasi Bis Kampus

Detail parameter tiap *layer* yang diterapkan dalam mapfile sistem informasi bis kampus dapat dilihat pada Tabel 3.1 di bawah ini.

**Tabel 3.1.** Parameter peta digital pada sistem informasi bis kampus

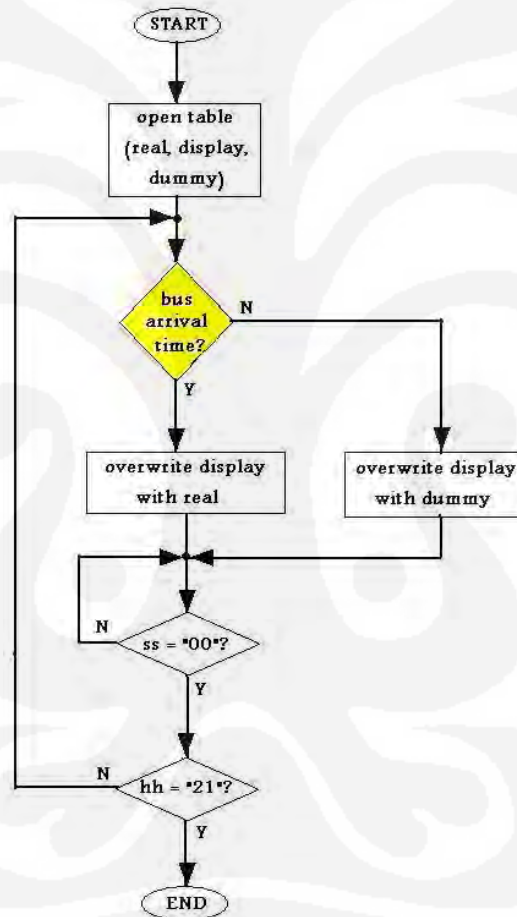
No.	Nama	Type	Data	Label	Warna (R,G,B)
1	Border	polygon	border_poly_region.shp	tidak	170,255,170
2	Fasilitas	polygon	fasilitas.tab	ya	0,255,255
3	Danau	polygon	danau.tab	tidak	0,128,255
4	Fakultas	polygon	fakultas.tab	ya	255,255,255
5	Politeknik	polygon	politeknik.tab	tidak	255,255,150
6	Stasiun	polygon	stasiun.tab	ya	128,128,0
7	Jalan	line	jalan_bikun_polyline.shp	tidak	0,0,0
8	halte merah <sup>1</sup>	point	halte_merah_disp.tab	tidak	255.0.0
9	halte biru <sup>2</sup>	point	halte_biru_disp.tab	tidak	0,0,255

<sup>1</sup> Halte merah terdiri atas layer-layer halte dengan jalur bis label merah (dari FH sampai FPsi) yang terdiri atas layer asrama\_red, komando\_red, stasiun\_red, FH\_red, masjid\_red, pocin\_red, fkm\_red, fmipa\_red, poltek\_red, stadion\_red, ft\_red, fe\_red, fib\_red, fisip\_red, fpsj\_red.

<sup>2</sup> Halte biru terdiri atas layer-layer halte dengan jalur bis label biru (dari FPsi sampai FH) dengan susunan layer-layer yang sama dengan halte merah, hanya saja kata "red" diubah menjadi "blue" (contoh: FH\_red menjadi FH\_blue).

### 3.5 PEMBUATAN PROGRAM PEMETAAN

Tujuan dari program pemetaan ini adalah mengatur tampilan layer posisi bus pada MapServer. Pada rancangan kali ini, tampilan layer posisi bus terakhir dimanipulasi menggunakan metode *overwrite* berdasarkan masukan yang diterima. Masukan dapat berupa data kedatangan bus kampus atau tabel *database* (*context-aware*). Pada rancangan kali ini, masukan yang digunakan adalah data kedatangan bus kampus dengan algoritma seperti Gambar 3.6 berikut ini.



**Gambar 3.6.** Algoritma Program Pemetaan metode statistik kedatangan bus

Pada algoritma ini, setelah program dimulai, program akan langsung membuka tabel-tabel posisi bus kampus yang akan ditampilkan (*display*), yang sebenarnya (*real*), juga membuka tabel koordinat sembarang yang terletak di luar wilayah Kampus UI Depok (*dummy*). Kemudian program akan langsung memeriksa pada saat itu bus kampus, baik label merah maupun biru, berada posisi

mana. Apabila terdapat bis kampus pada suatu halte, tabel posisi bis kampus yang akan ditampilkan (*display*) akan di-*overwrite* dengan koordinat sebenarnya bis kampus di halte tersebut (*real*). Namun apabila tidak ada bis kampus di suatu halte, maka tabel posisi bis kampus yang akan ditampilkan (*display*) akan di-*overwrite* dengan sebarang yang terletak di luar wilayah UI (*dummy*).

Selain *overwrite*, fungsi lain yang memegang peranan penting pada rancangan kali ini adalah fungsi waktu. Fungsi waktu berfungsi untuk mengambil waktu sistem untuk kemudian dimasukkan ke dalam string dengan format hh:mm:ss. Dimana hh berarti jam (00 sampai 24), mm berarti menit (00 sampai 59) dan ss berarti detik (00 sampai 59).

Pada algoritma, setelah proses *overwrite* selesai, program akan melihat apakah string "ss" bernilai "00" atau bukan, dan akan terus berlanjut hingga "ss" bernilai "00". Dengan kata lain, proses *overwrite layer* posisi bis hanya bisa terjadi satu menit sekali. *Looping* ini bertujuan untuk mengurangi ketergantungan desain akan spesifikasi komputer, karena *layer* posisi bis tidak dapat digunakan oleh MapServer apabila sedang melakukan proses *overwrite*, sehingga sistem akan memberikan pesan kesalahan dan pengguna tidak dapat melihat peta digital untuk mengetahui posisi bis.

Setelah *string* "ss" bernilai "00", sistem akan kembali memeriksa apakah *string* "hh" bernilai "21" atau tidak. Jika kondisi belum terpenuhi, maka program akan kembali melihat kondisi memeriksa waktu kedatangan bis dan melakukan *overwrite layer*. Namun apabila sudah bernilai "21", berarti waktu telah menunjukkan pukul 21.00 dimana sudah tidak ada bis kampus yang beroperasi pada jam itu, sehingga program dapat mengakhiri *sequence*-nya.

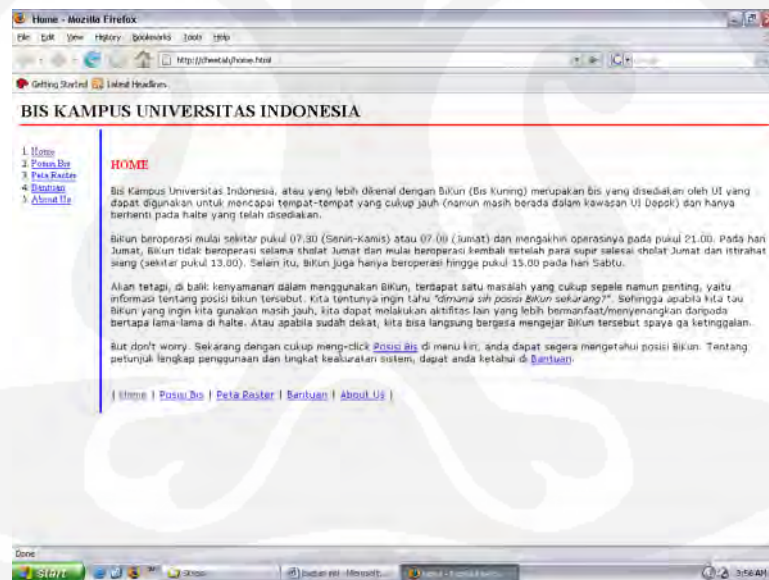
Penentuan waktu kedatangan bis dilakukan dengan cara mencatat waktu kedatangan bis di halte Teknik dan mencatat waktu rata-rata yang diperlukan bis kampus untuk pindah dari satu halte ke halte selanjutnya. Hal ini dilakukan karena para supir bis kampus beroperasi tidak menggunakan jadwal, tetapi berdasarkan perasaan dan pengalaman mereka selama menjadi supir bis kampus. Misal, banyak mahasiswa yang akan menuju ke fakultasnya masing-masing pada pukul 8 hingga 9. Sehingga pada rentang tersebut akan dioperasikan bis kampus dengan jumlah yang lebih banyak dibandingkan pada rentang pukul 11 hingga 12, dimana

pada rentang waktu tersebut jumlah mahasiswa sedang sibuk dengan kegiatan perkuliahan mereka, sehingga jumlah pengguna bis kampus menjadi sedikit.

### 3.6 PEMBUATAN HALAMAN WEB

Pengguna tentunya akan merasa kesulitan atau terganggu apabila harus mengetik *link* yang panjang pada *web browser* mereka untuk dapat melihat peta posisi bis kampus. Selain itu, pengguna juga memerlukan pengarahannya singkat agar dapat memahami informasi yang terkandung dalam peta posisi bis kampus. Oleh karena itu diperlukan suatu halaman *web* yang informatif dan *user friendly*.

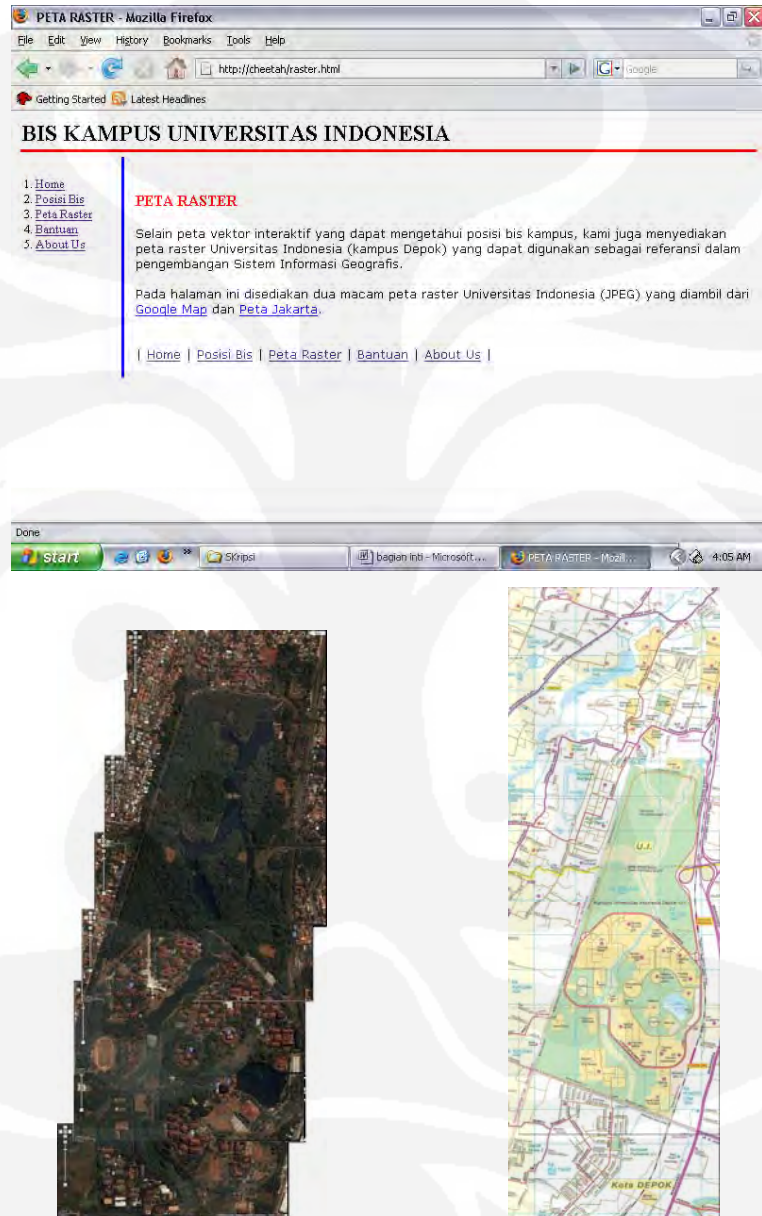
Halaman *web* pada Sistem Informasi Bis Kampus tersusun atas lima halaman, yaitu Beranda, Posisi Bis, Peta Raster, Bantuan, dan Tentang Kami. Halaman Beranda (Home) berisi tentang deskripsi singkat bis kampus Universitas Indonesia dan latar belakang Sistem Informasi Bis Kampus. Tampilan halaman Home pada Sistem Informasi Bis Kampus dapat dilihat pada Gambar 3.7 berikut ini.



**Gambar 3.7.** Halaman Beranda (Home) pada Sistem Informasi Bis Kampus

Halaman kedua pada Sistem Informasi Bis Kampus adalah halaman posisi bis. Pada halaman ini pengguna sudah langsung bisa melihat peta posisi bis kampus. Tampilan pada halaman Posisi Bis sama seperti pada Gambar 3.5 yang telah diperlihatkan sebelumnya.

Halaman ketiga adalah halaman Peta Raster. Pada halaman ini pengguna dapat melihat citra peta UI yang diambil dari Google Map dan program Flash Peta Jakarta. Tampilan dan peta yang dimiliki pada halaman Peta Raster dapat dilihat pada Gambar 3.8 di bawah ini.



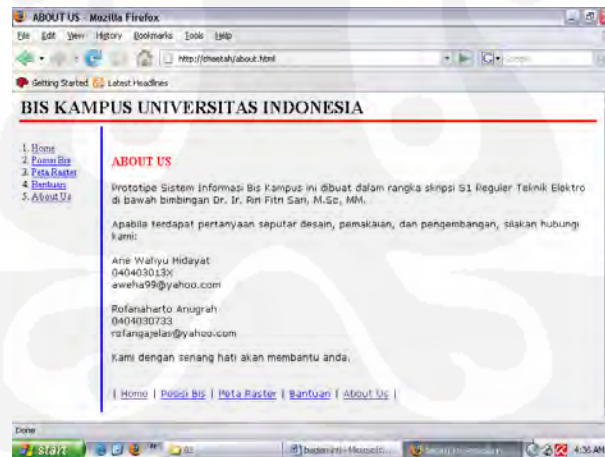
**Gambar 3.8.** Tampilan dan peta yang dimiliki pada halaman Peta Raster  
Halaman keempat adalah halaman bantuan. Pada halaman ini terdapat penjelasan singkat serta keterangan-keterangan tentang Sistem Informasi Bis Kampus. Tampilan pada halaman Bantuan dapat dilihat pada Gambar 3.9. di bawah ini.





**Gambar 3.9.** Tampilan pada halaman Bantuan

Halaman kelima adalah halaman Tentang Kami (About Us). Pada halaman ini pengguna dapat mengetahui identitas dan *e-mail* penyusun. Sehingga apabila ada permasalahan atau pertanyaan seputar desain, pemakaian atau pengembangan kedepan, pengguna dapat menghubungi penyusun untuk mendapatkan informasi serta dokumentasi tentang Sistem Informasi Bis Kampus. Tampilan pada halaman About Us dapat dilihat pada gambar 3.10 di bawah ini.



**Gambar 3.10.** Tampilan pada halaman Tentang kami (About Us)

## BAB 4

### ANALISA IMPLEMENTASI SISTEM INFORMASI BIS KAMPUS UNIVERSITAS INDONESIA

#### 4.1 PENGUMPULAN DATA DAN ANALISA DATA KEHADIRAN BIS KAMPUS

Salah satu kenyataan yang cukup mengejutkan tentang bis kampus Universitas Indonesia adalah para supir bis beroperasi tidak berdasarkan jadwal. Melainkan berdasarkan perasaan dan pengalaman selama menjadi supir bis kampus. Sebagai ilustrasi, bis kampus akan dioperasikan lebih banyak pada pukul 8 pagi dibandingkan pada pukul 12 siang. Hal ini dikarenakan berdasarkan pengalaman para supir, pada pukul 8 pagi jumlah pengguna bis kampus jauh lebih banyak dibandingkan pada pukul 12. Hal ini dapat dipahami karena umumnya kuliah mahasiswa UI dimulai pada pukul 8 pagi.

Kenyataan ini tentu saja memberikan masalah dalam menentukan waktu kedatangan bis kampus. Oleh karena itu dalam rancangan kali ini untuk menentukan posisi bis kampus dilakukan dengan cara mencatat waktu kedatangan bis di salah satu halte, yaitu halte Teknik. Berdasarkan hasil pencatatan waktu kedatangan bis kampus tersebut, didapat jadwal kedatangan bis kampus di halte Teknik seperti Tabel 4.1 berikut ini.

**Tabel 4.1.** Jadwal kedatanagn bis kampus di halte Teknik berdasarkan pengamatan di lapangan

MERAH	JAM	BIRU
00, 16, 17, 18, 19, 20, 37, 39, 43, 53, 58	8	00, 08, 14, 16, 20, 22, 23, 32, 37, 42, 50
04, 12, 17, 25, 30, 41, 53	9	00, 03, 06, 13, 20, 31, 34, 42, 50, 53
02, 13, 27, 39, 58	10	34, 53, 57
10, 29, 42, 45	11	26, 35, 56
21, 29, 40,	12	21, 39, 46
29, 33, 44, 49	13	16, 20, 41, 44
15, 24, 40, 45, 54	14	13, 17, 39, 57
03, 07, 14, 31, 58	15	01, 13, 30, 43
06, 25, 34, 46	16	10, 22, 35
09, 19, 34, 39, 7, 50	17	02, 11, 14, 26, 42

Pengambilan data untuk pukul 8, 13, 15, dan 17 dilakukan pada tanggal 14 Mei 2008. Pengambilan data untuk pukul 9 dan 10 dilakukan pada tanggal 15 Mei 2008. Pengambilan data untuk pukul 11, 12, dan 14 dilakukan pada tanggal 19 Mei 2008. Sedangkan pengambilan data untuk pukul 16 dilakukan pada tanggal 21 Mei 2008. Pengambilan data sengaja dilakukan pada hari selain hari Jumat dan Sabtu. Hal ini dikarenakan hari-hari tersebut merupakan hari-hari spesial, dimana bis kampus beroperasi tidak seperti biasanya.

Penulis telah menguji secara acak tingkat akurasi jadwal tersebut. Dan dari hasil pengujian didapat hasil yang cukup mengagumkan, yaitu tingkat kesalahan jadwal maksimal sebesar 5 menit. Dari sini dapat kita ketahui bahwa perasaan dan pengalaman para supir bis kampus UI telah membentuk suatu jadwal operasi sendiri bagi mereka yang terus mereka laksanakan secara rutin.

#### 4.2 PENGUMPULAN DATA DAN ANALISA WAKTU OVERWRITE LAYER

Pengambilan data waktu *overwrite layer* dilakukan secara manual menggunakan *stopwatch* pada tiga komputer dengan spesifikasi berbeda, yaitu Hamburg (salah satu komputer Mercator), Cheetah (komputer pribadi penulis), dan Bening-01 (laptop) dengan spesifikasi masing-masing komputer seperti pada Tabel 4.2 di bawah ini.

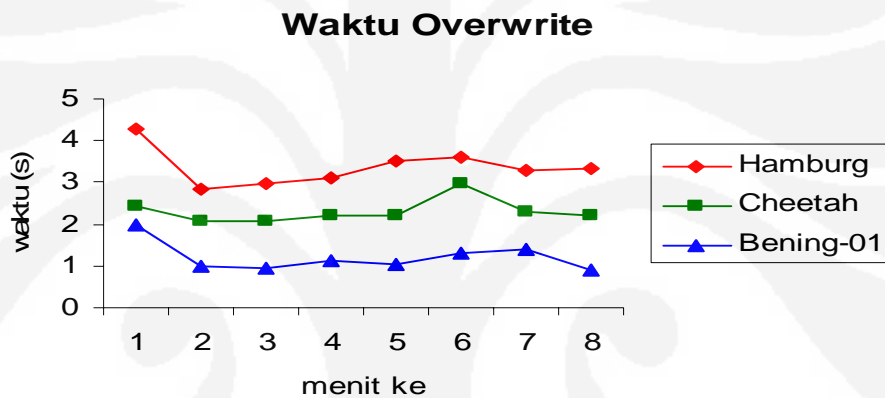
**Tabel 4.2.** Spesifikasi komputer Hamburg, Cheetah, dan Bening-01

Komputer	Processor	Memory
Hamburg	AMD Athlon 1.3 GHz	512 MB DDR RAM
Cheetah	Intel Pentium 4 1.7 GHz	512 MB DDR RAM
Bening-01	AMD Turion 2.2 GHz	512 MB DDR2 RAM

Data waktu *overwrite layer* yang diambil pada masing-masing komputer adalah waktu *overwrite layer* yang terjadi dalam delapan menit pertama, dan jumlah layer yang di-*overwrite* oleh ketiga komputer ini adalah sama, yaitu 30 layer posisi bis kampus. Data serta grafik yang diperoleh dari pengambilan data tersebut dapat dilihat pada Tabel 4.3 dan Grafik 4.1 di bawah ini.

**Tabel 4.3.** Data waktu *overwrite layer* pada Hamburg, Bening-01, dan Cheetah

menit ke	Hamburg (detik)	Bening-01 (detik)	Cheetah (detik)
1	4.26	1.96	2.43
2	2.83	0.98	2.07
3	2.96	0.94	2.08
4	3.11	1.12	2.22
5	3.52	1.04	2.21
6	3.6	1.31	2.96
7	3.31	1.38	2.28
8	3.33	0.91	2.2



**Grafik 4.1.** Grafik waktu *overwrite layer* pada delapan menit pertama

Dari Tabel 4.3 dan Grafik 4.1 di atas dapat kita ketahui bahwa semakin cepat CPU komputer, maka waktu *overwrite layer* yang dibutuhkan akan semakin cepat. Hal ini dibuktikan dengan kenyataan bahwa komputer yang memiliki waktu *overwrite* tercepat adalah Bening-01 dengan *clock* CPU 2.2 GHz, disusul oleh Cheetah (1.7 GHz) dan Hamburg (1.3 GHz).

Komputer dengan waktu *overwrite* yang cepat sangat dibutuhkan dalam Sistem Informasi Bis Kampus. Hal ini dikarenakan *layer* posisi bis kampus tidak dapat dipanggil oleh MapServer dan akan menimbulkan pesan kesalahan selama proses *overwrite* untuk mengganti posisi bis kampus.

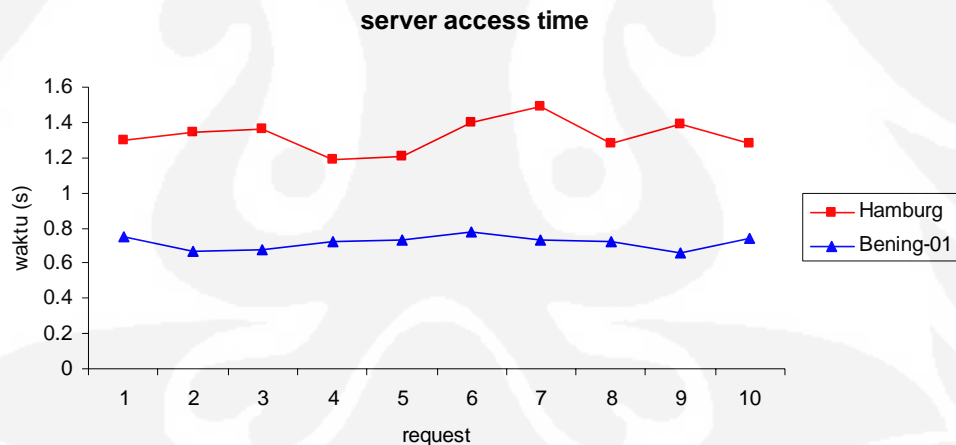
#### 4.3 PENGUMPULAN DATA DAN ANALISA WAKTU AKSES SERVER

Pengumpulan data waktu akses server Sistem Informasi Bis Kampus dilakukan secara manual menggunakan *stopwatch* pada tingkat Intranet (LAN) di

laboratorium Mercator. Komputer yang dijadikan server kali ini adalah Hamburg dan Bening-01. Sedangkan komputer yang bertindak sebagai *client* dan mengirimkan *request* posisi bis ke *server* adalah Oberhausen (salah satu komputer Mercator). *Data* serta grafik yang diperoleh dari pengambilan data tersebut dapat dilihat pada Tabel 4.4 dan Grafik 4.2 di bawah ini.

**Tabel 4.4.** Data waktu respon posisi bis kampus pada Hamburg dan Bening-01

Request	Hamburg	Bening-01
1	1.3	0.75
2	1.34	0.67
3	1.36	0.68
4	1.19	0.72
5	1.21	0.73
6	1.4	0.78
7	1.49	0.73
8	1.28	0.72
9	1.39	0.66
10	1.28	0.74



**Grafik 4.2.** Grafik waktu respon server terhadap request dari client

Dari data dan grafik di atas kembali dapat kita ketahui bahwa spesifikasi komputer cukup berpengaruh dalam waktu akses, walaupun tidak signifikan. Perbedaan yang dihasilkan sekitar satu detik. Perbedaan yang kecil serta waktu akses yang cepat ini dikarenakan *server* diakses melalui LAN. Apabila Sistem Informasi Bis Kampus benar-benar diterapkan dan dapat diakses lewat Internet,

maka spesifikasi jaringan menjadi jauh lebih berpengaruh dibandingkan spesifikasi komputer *server* dalam memberikan respon yang cepat.

#### **4.4 TANGGAPAN USER TERHADAP IMPLEMENTASI SISTEM INFORMASI BIS KAMPUS**

Untuk mengetahui tanggapan *user* terhadap implementasi Sistem Informasi Bis Kampus dilakukan dengan cara meminta *user* untuk mengisi suatu form kuisioner yang telah disediakan. Pada kuisioner tersebut akan dapat diketahui tanggapan user terhadap tampilan, akurasi, peta posisi bis, kinerja, dan kegunaan Sistem Informasi Bis Kampus. Selain itu, pada akhir kuisioner ini pengguna akan diminta untuk memberikan komentar, saran, dan harapan terhadap prototipe Sistem Informasi Bis Kampus Universitas Indonesia ini.

Seluruh responden terdiri atas mahasiswa S1 Teknik Elektro Universitas Indonesia, baik ekstensi dan reguler. Pengisian kuisioner dilakukan secara spontan di laboratorium Mercator.

Dari kuisioner yang diberikan dapat kita ketahui bahwa dari 20 *user*, 10 orang mengatakan sangat setuju bahwa Sistem Informasi Bis Kampus berguna, 8 orang mengatakan setuju, dan 2 orang mengatakan cukup berguna.

Responden juga diminta untuk memberikan pendapat mereka tentang tampilan (GUI) pada Sistem Informasi Bis Kampus ini. Dari 20 orang responden, 2 orang mengatakan tampilan yang dimiliki sangat bagus, 8 orang mengatakan bagus, 4 orang mengatakan cukup bagus (biasa), dan 2 orang mengatakan kurang bagus.

Selain itu, lewat kuisioner ini *user* juga diminta untuk memberi tanggapan tentang tingkat akurasi posisi bis yang ditampilkan dalam peta. Hasilnya, 12 orang mengatakan akurat, 6 orang mengatakan cukup akurat, dan satu orang mengatakan kurang akurat.

Untuk kinerja, 4 orang responden mengatakan waktu yang diperlukan *server* untuk menanggapi *request* peta posisi bis sangat cepat, sedang 16 orang sisanya mengatakan cepat. Rangkuman dari hasil kuisioner Sistem Informasi Bis Kampus Universitas Indonesia dapat dilihat dalam Tabel 4.5 berikut ini.

**Tabel 4.5.** Hasil Kuisioner Sistem Informasi Bis Kampus

No.	Parameter	Jumlah Responden terhadap Nilai				
		1	2	3	4	5
1	Sistem Informasi Bis Kampus berguna, karena dapat menampilkan peta posisi Bis Kampus lewat Internet	0	0	2	8	10
2	Tampilan (GUI) pada web site Sistem Informasi Bis Kampus ini menarik	0	6	4	8	2
3	Tampilan peta pada Sistem Informasi Bis Kampus menarik	0	2	6	8	4
4	Posisi Bis Kampus yang ditampilkan akurat	0	2	6	12	0
5	Waktu yang diperlukan untuk mengakses peta tidak lama (cepat)	0	0	0	16	4

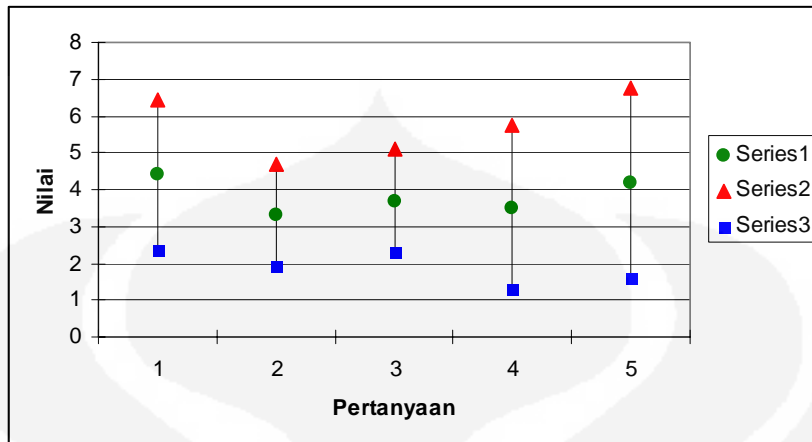
Pada Tabel 4.5 di atas, nilai 5 berarti sangat setuju, nilai 4 berarti setuju, nilai 3 berarti cukup, nilai 2 berarti tidak setuju, dan nilai 1 berarti sangat tidak setuju. Berdasarkan data kuisioner tersebut kita dapat mengetahui nilai optimis terhadap parameter dengan menggunakan formula:

$$\text{average} \pm 1.96 \left( \frac{\text{stdev}}{\sqrt{20}} \right)$$

Pada formula di atas, *average* berarti rata-rata data yang dicari dengan cara membagi bobot nilai dengan jumlah responden. Sedangkan *stdev* berarti standar deviasi data untuk satu parameter (pertanyaan) pada kuisioner. Dengan menerapkan formula tersebut kita dapat mengetahui nilai optimis dan pesimis untuk tiap parameter dari data responden seperti Tabel 4.6 dan Grafik 4.3 berikut ini.

**Tabel 4.6.** Hasil pengolahan data kuisioner Sistem Informasi Bis Kampus

Jumlah Responden terhadap Nilai					StDev	rata-rata	optimis	pesimis
1	2	3	4	5				
0	0	2	8	10	4.690416	4.4	6.455665	2.344335
0	6	4	8	2	3.162278	3.3	4.685929	1.914071
0	2	6	8	4	3.162278	3.7	5.085929	2.314071
0	2	6	12	0	5.09902	3.5	5.734744	1.265256
0	0	0	13	7	5.87367	4.2	6.774249	1.625751



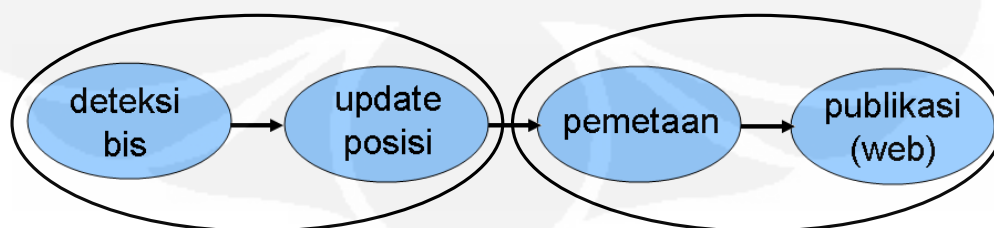
**Grafik 4.3** Rentang nilai optimis dan optimis responden

Sedangkan tentang *server* mana yang lebih cepat antara Bening-01 dan Hamburg, 8 dari 10 *user* berpendapat sama saja, sedangkan 2 orang sisanya berpendapat Bening-01 lebih cepat dibandingkan Hamburg. Respon yang cepat dan waktu akses antara Hamburg dan Bening-01 yang hampir sama dikarenakan *user* mengakses *server* melalui LAN, dimana jarak antara *server* dan *client* hanya beberapa meter.

Pada akhir kuisisioner dapat kita ketahui bahwa *user* ingin Sistem Informasi Bis Kampus ini dapat benar-benar direalisasikan dengan rancangan yang lebih stabil, tampilan yang lebih menarik, dapat diakses lewat PDA dan *Handphone*, serta posisi bis yang berubah secara *real time*.

#### 4.5 PENGEMBANGAN IMPLEMENTASI SISTEM INFORMASI BIS KAMPUS KE DEPAN

Perancangan Sistem Informasi Bis Kampus Universitas Indonesia pada awalnya terdiri atas empat modul, yaitu modul deteksi bis, *update* posisi, pemetaan, dan publikasi (MapServer) seperti pada Gambar 3.11 berikut ini.



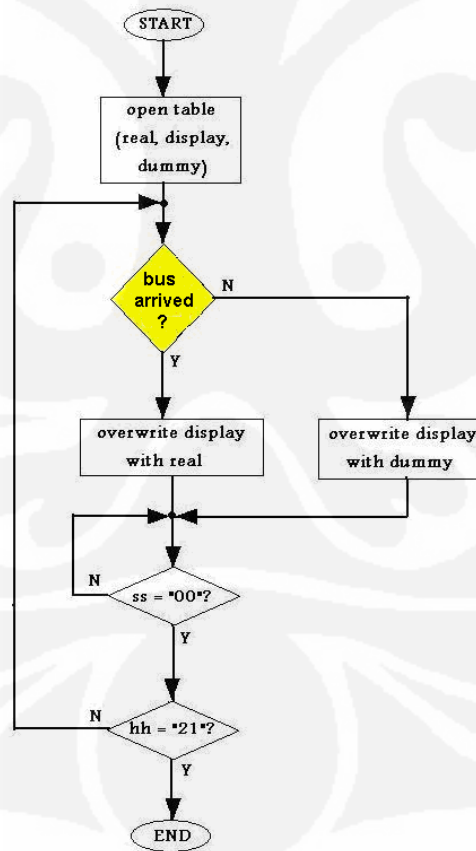
**Gambar 3.11** Modul-modul dalam Sistem Informasi Bis Kampus



Modul deteksi bus dan update posisi dikerjakan secara terpisah oleh rekan penulis. Karena beberapa hal modul ini tidak dapat langsung diterapkan pada Sistem Informasi Bus Kampus. Sehingga pada implementasi kali ini posisi bus yang seharusnya di-update secara *real time* diganti dengan waktu kedatangan bus kampus.

Untuk mengkomunikasikan antara modul *update* posisi dengan modul pemetaan digunakan perintah DDE (*Dynamic data Exchange*) yang dimiliki oleh ArcView. Perintah ini memungkinkan ArcView berkomunikasi dengan aplikasi lain yang sedang aktif di *memory*.

Posisi bus yang telah di-update dalam suatu aplikasi database akan diambil oleh ArcView menggunakan perintah DDE sehingga memungkinkan pengguna untuk mengetahui posisi bus kampus terakhir secara hampir *real time*. Berdasarkan tabel database inilah proses *overwrite layer* posisi bus kampus dilakukan seperti yang dituangkan dalam algoritma pada Gambar 3.12 berikut ini.



**Gambar 3.12** Algoritma *overwrite layer* menggunakan tabel *database*

## BAB 5

### KESIMPULAN

1. Sistem Informasi Bis Kampus memungkinkan para pengguna bis kampus melihat posisi bis kampus terakhir lewat Internet.
2. Langkah-langkah yang perlu dilakukan dalam perancangan modul pemetaan dan MapServer pada Sistem Informasi Bis Kampus adalah instalasi MapServer (FGS Linux) dan ArcView (menggunakan Wine), perencanaan desain (UML), membuat data spasial, pembuatan MapFile, membuat program pemetaan, dan membuat halaman *web*.
3. Terdapat tiga komponen dalam algoritma pemetaan, yaitu: operasi waktu, operasi *overwrite*, dan data posisi bis, dimana operasi *overwrite* bekerja berdasarkan operasi waktu dan data posisi bis.
4. Penentuan data posisi bis dilakukan dengan cara mencatat waktu kedatangan bis kampus di satu halte dan memperkirakan waktu perpindahan bis. Metode ini menghasilkan *error* maksimal lima menit
5. Semakin cepat prosesor komputer *server*, maka waktu yang diperlukan untuk *overwrite layer* posisi bis akan semakin singkat dan kemungkinan menghasilkan pesan kesalahan akan semakin kecil.
6. Spesifikasi komputer *server* cukup berpengaruh terhadap lamanya respon yang harus diberikan terhadap *request* peta posisi bis walaupun tidak signifikan.
7. Pada kuisisioner evaluasi Sistem Informasi Bis Kampus diketahui bahwa pengguna berpendapat Sistem Informasi Bis Kampus berguna, memiliki tampilan yang biasa, tingkat akurasi cukup tinggi, memiliki waktu respon yang cepat. Selain itu pengguna juga berharap agar rancangan ini dapat benar-benar direalisasikan dengan desain yang lebih stabil dan tampilan yang lebih menarik.

## DAFTAR ACUAN

- [1] "Sistem Informasi Geografis",  
[http://id.wikipedia.org/wiki/Sistem\\_informasi\\_geografis](http://id.wikipedia.org/wiki/Sistem_informasi_geografis)  
\_\_\_\_\_, terakhir diakses 27 April 2008
- [2] "*Geographic information system*",  
[http://en.wikipedia.org/wiki/Geographic\\_information\\_system](http://en.wikipedia.org/wiki/Geographic_information_system), \_\_\_\_\_,  
terakhir diakses 27 April 2008
- [3] "Sistem Informasi Geografi (SIG)/*Geographic Information System*",  
<http://mbojo.wordpress.com/2007/04/08/sistem-informasi-geografi-sig/>,  
La An, terakhir diakses 20 April 2008
- [4] "MapServer", <http://en.wikipedia.org/wiki/Mapserver>, \_\_\_\_\_, terakhir  
diakses 27 April 2008
- [5] "Membangun Aplikasi *Web-based GIS* dengan MapServer", Eddy Prahasta,  
Penerbit Informatika, Bandung , 2007.
- [6] "Panduan Menggunakan MapServer", Ir. Ruslan Nuryadin, Penerbit  
Informatika, Bandung , 2005.
- [7] "*Shapefile C Library V1.2*", <http://shapelib.maptools.org/>, \_\_\_\_\_, terakhir  
diakses 18 Maret 2008
- [8] "*GDAL Raster Formats*", [http://www.gdal.org/formats\\_list.html](http://www.gdal.org/formats_list.html), \_\_\_\_\_,  
terakhir diakses 18 Maret 2008
- [9] "*OGR vEctor Formats*", [http://www.gdal.org/ogr/ogr\\_formats.html](http://www.gdal.org/ogr/ogr_formats.html), \_\_\_\_\_,  
terakhir diakses 18 Maret 2008

- [10] "OGR vEctor Formats", [http://www.gdal.org/ogr/ogr\\_formats.html](http://www.gdal.org/ogr/ogr_formats.html), Greg Roelofs, terakhir diakses 18 Maret 2008
- [11] "IJG", <http://www.ijg.org/>, \_\_\_\_\_, terakhir diakses 18 Maret 2008
- [12] "GD Graphics Library", <http://www.boutell.com/gd/>, Boutell.com, terakhir diakses 18 Maret 2008
- [13] "The FreeType Project", <http://freetype.sourceforge.net/index2.html>, David Turner, terakhir diakses 18 Maret 2008
- [14] "PROJ.4 - Cartographic Projections Library", <http://proj.maptools.org/>, Frank Warmerdam, terakhir diakses 18 Maret 2008
- [15] "zlib: A Massively Spiffy Yet Delicately Unobtrusive Compression Library", <http://www.zlib.net/>, Jean Gailly and Mark Adler, terakhir diakses 18 Maret 2008
- [16] "Mapfile Reference", <http://mapserver.gis.umn.edu/docs/reference/mapfile/referencemanual-all-pages>, University of Minnesota, terakhir diakses 18 Maret 2008

## DAFTAR PUSTAKA

Eddy Prahasta, *Belajar dan Memahami MapInfo* (Bandung: Penerbit Informatika, 2006)

Eddy Prahasta, *Aplikasi Pemrograman MapInfo* (Bandung: Penerbit Informatika, 2005)

Eddy Prahasta, *Tutorial ArcView* (Bandung: Penerbit Informatika, 2007)

Eddy Prahasta, *Sistem Informasi Geografis: ArcView Lanjut Pemrograman Bahasa Script Avenue* (Bandung: Penerbit Informatika, 2004)

Iwan Sofana, *Mudah Membangun Server dengan Fedora Core* (Bandung: Penerbit Informatika, 2007)

Bill Kropla, *Beginning MapServer: Open Source GIS Development* (New York: Apress, 2005)

## LAMPIRAN

```
# Mapfile Sistem Informasi Bis Kampus
MAP
  NAME UI
  IMAGETYPE JPEG
  EXTENT 106.492 -6.2256 106.5 -6.2078
  STATUS ON
  UNITS DD
  SIZE 440 800
  IMAGECOLOR 245 245 245 # abu-abu
  SHAPEPATH "D:\buku_mapserver\ogr\UI
    DISPLAY\DISP! bikun poly\shp\"
  FONTSET
    "d:\buku_mapserver\font\font.dat"
  SYMBOLSET
    "d:\buku_mapserver\simbol\simbol.
    sym"
#
# Mulai pendefinisian layers & kelas
#
LAYER # 01
  NAME Border
  STATUS default
  TYPE POLYGON
  DATA border_poly_region
#
  CLASS
  STYLE
    COLOR 170 255 170
  END # style
  END # class
#
  END # Akhir definisi layer
#
LAYER # 02
  NAME Fasilitas
  STATUS default
  TYPE POLYGON
  CONNECTIONTYPE ogr
  CONNECTION Fasilitas.tab
  LABELITEM Fasilitas
#
  CLASS
  NAME Fasilitas
  STYLE
    COLOR 0 255 255
  END # style
  LABEL
  FONT arial
  TYPE truetype
  SIZE 9
  POSITION cc
  FORCE true
  COLOR 0 0 0
  END # label
  END # class
#
  END # Akhir definisi layer
#
LAYER # 03
  NAME Danau
  STATUS default
  TYPE POLYGON
  CONNECTIONTYPE ogr
  CONNECTION danau.tab
#
  CLASS
  NAME danau
  STYLE
    COLOR 22 150 255
  END # style
  END # class
#
  END # Akhir definisi layer
#
LAYER # 04
  NAME Fakultas
  STATUS default
  TYPE POLYGON
  CONNECTIONTYPE ogr
  CONNECTION Fakultas.tab
  LABELITEM fakultas
#
  CLASS
  NAME Fasilitas
  STYLE
    COLOR 255 255 255
  END # style
  LABEL
  FONT arial
  TYPE truetype
  SIZE 9
  POSITION cc
  FORCE true
  COLOR 0 0 0
  END # label
  END # class
#
  END # Akhir definisi layer
#
LAYER # 05
  NAME Politeknik
  STATUS default
  TYPE POLYGON
  CONNECTIONTYPE ogr
  CONNECTION politeknik.tab
#
  END # label
  END # class
#
  END # Akhir definisi layer
#
```

```

CLASS
  NAME Politeknik
  STYLE
    COLOR 255 255 150
  END # style
END # class
#
END # Akhir definisi layer
#
LAYER # 06
  NAME Stasiun
  STATUS default
  TYPE POLYGON
  CONNECTIONTYPE ogr
  CONNECTION stasiun.tab
  LABELITEM stasiun
#
CLASS
  NAME Stasiun
  STYLE
    COLOR 128 128 0
  END # style
  LABEL
    FONT arial
    TYPE truetype
    SIZE 9
    POSITION cr
    FORCE true
    COLOR 0 0 0
  END # label
END # class
#
END # Akhir definisi layer
#
LAYER # 07
  NAME Jalan
  STATUS default
  TYPE line
  DATA jalan_bikun_polyline.shp
#
CLASS
  STYLE
    COLOR 0 0 0
  END # style
END # class
#
END # Akhir definisi layer
#
LAYER # 08
  NAME FPsi_blue #layer posisi bis
  STATUS on
  TYPE polygon
  DATA fps_i_blue_disp_region.shp
#
CLASS
  NAME BiKun_Biru
  STYLE
    COLOR 0 0 255
  END # style
END # class

END # style
END # class
#
END # Akhir definisi layer
#
SCALEBAR
  STATUS EMBED
  LABEL
    SIZE Medium
    COLOR 0 0 0
    OUTLINECOLOR 255 255 255
  END # label
  STYLE 0
  INTERVALS 4
  SIZE 400 5
  COLOR 0 0 0
  BACKGROUNDCOLOR 255 255 255
  OUTLINECOLOR 0 0 0
  UNITS METERS
  POSITION LL
  TRANSPARENT TRUE
END # Scalebar
#
LEGEND # BLOK LEGENDA
  IMAGECOLOR 245 245 245 # abu-abu
  OUTLINECOLOR 0 0 0 # abu-abu
  POSITION UL # legenda di kiri atas
  KEYSIZE 20 13
  KEYSPECING 10 5
  POSTLABELCACHE TRUE
  STATUS EMBED
  TRANSPARENT OFF
  LABEL # LABEL LEGENDA
  TYPE TRUETYPE
  FONT ARIAL
  COLOR 0 0 0
  SIZE 10
  ANTIALIAS TRUE
  END # LABEL
END # AKHIR BLOK LEGENDA
#
END # Akhir mapfile)

```

```

'Script program ArcView-nya
'BLUE DISPLAY
asrama_blue_disp =
"d:\buku_mapserver\ogr\UI
DISPLAY\DISP! bikun
poly\shp\asrama_blue_disp_region.shp".AsFile
Name
'file posisi bikun biru (asli)
'BLUE REAL
asrama_blue_real =
"d:\buku_mapserver\ogr\UI REAL\ASLI!
bikun
poly\shp\asrama_blue_asli_region.shp".AsFile
Name
'file posisi bikun biru (ditampilkan)
'RED DISPLAY
asrama_red_disp =
"d:\buku_mapserver\ogr\UI
DISPLAY\DISP! bikun
poly\shp\asrama_red_disp_region.shp".AsFile
Name
'posisi bikun merah (asli)
'RED REAL
asrama_red_real =
"d:\buku_mapserver\ogr\UI REAL\ASLI!
bikun
poly\shp\asrama_red_asli_region.shp".AsFile
Name
'posisi bikun merah (ditampilkan)
,
dummy = "d:\buku_mapserver\ogr\UI
DISPLAY\DISP! bikun
poly\shp\dummy_rectangle.shp".AsFile
Name
,
jam = Date.Now
jam.SetFormat( "hhhh" )
objJam = jam.AsString
,
While (objJam <> "21") 'jam 21
,
menit = Date.Now
detik = Date.Now
menit.SetFormat( "m" )
detik.SetFormat( "s" )
objMenit = menit.AsString
objDetik = detik.AsString
,
if (objJam = "15") then 'jam 15
,
if (objMenit = "00") then
'jangan lupa apus jam sebelumnya
File.Copy (fe_blue_real,fe_blue_disp)
File.Copy
(fmipa_blue_real,fmipa_blue_disp)
File.Copy (dummy,fkm_red_disp)
File.Copy (fmipa_red_real,fmipa_red_disp)
File.Copy (dummy,stasiun_red_disp)
File.Copy (dummy,fe_red_real,fe_red_disp)
File.Copy (dummy,asrama_red_disp)
File.Copy (ui_red_real,ui_red_disp)
File.Copy (dummy,fe_blue_disp)
File.Copy (dummy,ft_blue_disp)
File.Copy (dummy,fkm_blue_disp)
File.Copy
(stadion_blue_real,stadion_blue_disp)
File.Copy
(pocin_blue_real,pocin_blue_disp)
File.Copy (dummy,poltek_red_disp)
File.Copy
(stadion_red_real,stadion_red_disp)
File.Copy (dummy,masjid_red_disp)
File.Copy (pocin_red_real,pocin_red_disp)
ElseIf (objMenit = "03") then
File.Copy (dummy,stadion_blue_disp)
File.Copy (dummy,pocin_blue_disp)
File.Copy
(poltek_blue_real,poltek_blue_disp)
File.Copy
(masjid_blue_real,masjid_blue_disp)
File.Copy (dummy,stadion_red_disp)
File.Copy (ft_red_real,ft_red_disp)
File.Copy (dummy,pocin_red_disp)
File.Copy (fkm_red_real,fkm_red_disp)
ElseIf (objMenit = "04") then
File.Copy (dummy,poltek_blue_disp)
File.Copy (dummy,masjid_blue_disp)
File.Copy
(fmipa_blue_real,fmipa_blue_disp)
File.Copy (fh_blue_real,fh_blue_disp)
File.Copy (dummy,ft_red_disp)
File.Copy (fe_red_real,fe_red_disp)
File.Copy (dummy,fkm_red_disp)
File.Copy (fmipa_red_real,fmipa_red_disp)
File.Copy
(asrama_red_real,asrama_red_disp)
ElseIf (objMenit = "05") then
File.Copy (dummy,fmipa_blue_disp)
File.Copy (dummy,fh_blue_disp)
File.Copy (fkm_blue_real,fkm_blue_disp)
File.Copy (dummy,fe_red_disp)
File.Copy (fib_red_real,fib_red_disp)
File.Copy (dummy,fmipa_red_disp)
File.Copy
(poltek_red_real,poltek_red_disp)
File.Copy (dummy,asrama_red_disp)
File.Copy (ui_red_real,ui_red_disp)
File.Copy (fh_red_real,fh_red_disp)
ElseIf (objMenit = "01") then
File.Copy (dummy,fe_blue_disp)
File.Copy (dummy,fmipa_blue_disp)
File.Copy (ft_blue_real,ft_blue_disp)
File.Copy (fkm_blue_real,fkm_blue_disp)
File.Copy (dummy,fmipa_red_disp)
File.Copy
(poltek_red_real,poltek_red_disp)
File.Copy (dummy,fh_red_disp)
File.Copy
(masjid_red_real,masjid_red_disp)
ElseIf (objMenit = "02") then
File.Copy (dummy,ft_blue_disp)
File.Copy (dummy,fkm_blue_disp)
File.Copy
(stadion_blue_real,stadion_blue_disp)
File.Copy
(pocin_blue_real,pocin_blue_disp)
File.Copy (dummy,poltek_red_disp)
File.Copy
(stadion_red_real,stadion_red_disp)
File.Copy (dummy,masjid_red_disp)
File.Copy (pocin_red_real,pocin_red_disp)
ElseIf (objMenit = "03") then
File.Copy (dummy,stadion_blue_disp)
File.Copy (dummy,pocin_blue_disp)
File.Copy
(poltek_blue_real,poltek_blue_disp)
File.Copy
(masjid_blue_real,masjid_blue_disp)
File.Copy (dummy,stadion_red_disp)
File.Copy (ft_red_real,ft_red_disp)
File.Copy (dummy,pocin_red_disp)
File.Copy (fkm_red_real,fkm_red_disp)
ElseIf (objMenit = "04") then
File.Copy (dummy,poltek_blue_disp)
File.Copy (dummy,masjid_blue_disp)
File.Copy
(fmipa_blue_real,fmipa_blue_disp)
File.Copy (fh_blue_real,fh_blue_disp)
File.Copy (dummy,ft_red_disp)
File.Copy (fe_red_real,fe_red_disp)
File.Copy (dummy,fkm_red_disp)
File.Copy (fmipa_red_real,fmipa_red_disp)
File.Copy
(asrama_red_real,asrama_red_disp)
ElseIf (objMenit = "05") then
File.Copy (dummy,fmipa_blue_disp)
File.Copy (dummy,fh_blue_disp)
File.Copy (fkm_blue_real,fkm_blue_disp)
File.Copy (dummy,fe_red_disp)
File.Copy (fib_red_real,fib_red_disp)
File.Copy (dummy,fmipa_red_disp)
File.Copy
(poltek_red_real,poltek_red_disp)
File.Copy (dummy,asrama_red_disp)
File.Copy (ui_red_real,ui_red_disp)

```



```

ElseIf (objMenit = "06") then
File.Copy (dummy,fkm_blue_disp)
File.Copy
(pocin_blue_real,pocin_blue_disp)
File.Copy
(asrama_blue_real,asrama_blue_disp)
File.Copy (dummy,fib_red_disp)
File.Copy (fisip_red_real,fisip_red_disp)
File.Copy (dummy,poltek_red_disp)
File.Copy
(stadion_red_real,stadion_red_disp)
File.Copy (dummy,ui_red_disp)
File.Copy
(stasiun_red_real,stasiun_red_disp)
ElseIf (objMenit = "07") then
File.Copy (dummy,pocin_blue_disp)
File.Copy (dummy,asrama_blue_disp)
File.Copy
(masjid_blue_real,masjid_blue_disp)
File.Copy (ui_blue_real,ui_blue_disp)
File.Copy (dummy,fisip_red_disp)
File.Copy (fpsi_red_real,fpsi_red_disp)
File.Copy (dummy,stadion_red_disp)
File.Copy (ft_red_real,ft_red_disp)
File.Copy (dummy,stasiun_red_disp)
File.Copy (fh_red_real,fh_red_disp)
ElseIf (objMenit = "08") then
File.Copy (dummy,masjid_blue_disp)
File.Copy (dummy,ui_blue_disp)
File.Copy (fh_blue_real,fh_blue_disp)
File.Copy
(stasiun_blue_real,stasiun_blue_disp)
File.Copy (dummy,fps_i_red_disp)
File.Copy (dummy,ft_red_disp)
File.Copy (fe_red_real,fe_red_disp)
File.Copy (dummy,fh_red_disp)
File.Copy
(masjid_red_real,masjid_red_disp)
ElseIf (objMenit = "09") then
File.Copy (dummy,fh_blue_disp)
File.Copy (dummy,stasiun_blue_disp)
File.Copy (fpsi_blue_real,fps_i_blue_disp)
File.Copy (dummy,fe_red_disp)
File.Copy (fib_red_real,fib_red_disp)
File.Copy (dummy,masjid_red_disp)
File.Copy (pocin_red_real,pocin_red_disp)
ElseIf (objMenit = "10") then
File.Copy (dummy,fps_i_blue_disp)
File.Copy (fisip_blue_real,fisip_blue_disp)
File.Copy (dummy,fib_red_disp)
File.Copy (fisip_red_real,fisip_red_disp)
File.Copy (dummy,pocin_red_disp)
File.Copy (fkm_red_real,fkm_red_disp)
' dst untuk semua jam
End
,
End 'End If jam 15
,
While (objDetik <> "59") 'satu menit
detik = Date.Now
detik.SetFormat( "s" )
objDetik = detik.AsString
End 'End while detik 59
,
jam = Date.Now
jam.SetFormat( "hhhh" )
objJam = jam.AsString
,
End 'End while jam 21

```