

**SIMULASI DAERAH BANJIR
MENGUNAKAN SISTEM INFORMASI GEOGRAFIS
DI KABUPATEN SRAGEN**

SKRIPSI

Oleh:

**MURDIYANTO
NIM. 04550073**



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
2010**

**SIMULASI DAERAH BANJIR
MENGUNAKAN SISTEM INFORMASI GEOGRAFIS
DI KABUPATEN SRAGEN**

SKRIPSI

Diajukan Kepada:

**Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

Oleh:

**MURDIYANTO
NIM. 04550073**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
2010**

**SURAT PERNYATAAN
ORISINILITAS PENELITIAN**

Saya yang bertanda tangan di bawah ini:

Nama : Murdiyanto
NIM : 04550073
Fakultas/Jurusan : Sains dan Teknologi / Teknik Informatika
Judul Penelitian : SIMULASI DAERAH BANJIR MENGGUNAKAN
SISTEM INFORMASI GEOGRAFIS DI KABUPATEN
SRAGEN

Menyatakan dengan sebenar-benarnya bahwa hasil penelitian saya ini tidak terdapat unsur-unsur penjiplakan karya penelitian atau karya ilmiah yang pernah dilakukan atau dibuat oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata hasil penelitian ini terbukti terdapat unsur-unsur jiplakan, maka saya bersedia untuk mempertanggungjawabkan, serta diproses sesuai peraturan yang berlaku.

Malang, 31 Juli 2010
Yang membuat pernyataan,

Murdiyanto
NIM. 04550073

**SIMULASI DAERAH BANJIR
MENGUNAKAN SISTEM INFORMASI GEOGRAFIS
DI KABUPATEN SRAGEN**

SKRIPSI

Oleh:

**MURDIYANTO
NIM. 04550073**

Telah disetujui oleh:

Dosen Pembimbing I,

Dosen Pembimbing II,

**Syahiduz Zaman, M.Kom
NIP. 197005022005011005**

**M. Ainul Yaqin, M.Kom
NIP. 197610132006041004**

Tanggal,

**Mengetahui,
Ketua Jurusan Teknik Informatika**

**Ririen Kusumawati, M.Kom
NIP. 197203092005012002**

**SIMULASI DAERAH BANJIR
MENGUNAKAN SISTEM INFORMASI GEOGRAFIS
DI KABUPATEN SRAGEN**

SKRIPSI

Oleh

**Murdiyanto
NIM. 04550073**

**Telah Dipertahankan Di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)**

Tanggal,

Susunan Dewan Penguji :

Tanda Tangan

- | | | |
|-------------------------|--|------------|
| 1. Penguji Utama | : Totok Chamidy, M.Kom
NIP. 196912222006041001 | () |
| 2. Ketua | : Zainal Abidin, M. Kom
NIP. 197606132005011001 | () |
| 3. Sekertaris | : Syahiduzzaman, M.Kom
NIP. 197005022005011005 | () |
| 4. Anggota | : M. Ainul Yaqin, M.Kom
NIP. 197610132006041004 | () |

**Mengetahui dan Mengesahkan
Ketua Jurusan Teknik Informatika**

**Ririen Kusumawati, M.Kom
NIP : 197203092005012002**

MOTTO

Jalani peranmu, penuhi takdirmu
dengan kualitas terbaikmu.

Pasrah bukan tidak melakukan,
Pasrah itu sedang melakukan.

Bukan mengkompromi keadaan,
Tapi kualitasmu yang ditingkatkan.

(Murdian_D' Plato)

PERSEMBAHAN

- 1. Ibu dan Bapakku yang telah berjuang untuk bisa mengawal sampai selesai belajarku, engkau pahlawanku.*
- 2. Mbak Asih dan keluarga, maaf jika adikmu ini menyusahkan Orang Tua kita.*
- 3. Keluarga Besar Pondok Pesantren Al-Hikmah Tanon, aku tidak pernah lupa habitatku.*

KATA PENGANTAR

Puji syukur ke hadirat Allah SWT atas limpahan rahmat dan karuni-Nya sehingga penulis dapat menyelesaikan karya ilmiah ini dengan judul **“SIMULASI DAERAH BANJIR MENGGUNAKAN SISTEM INFORMASI GEOGRAFIS DI KABUPATEN SRAGEN”**. Penelitian ini dimaksudkan untuk memenuhi salah satu syarat dalam meraih gelar Sarjana Komputer di Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim (UIN Maliki) Malang.

Sholawat dan salam penulis sampaikan kepada Nabi Muhammad SAW, lewat perjuangannya penulis bisa merasakan nikmatnya iman dan Islam. Alhamdulillah penulis bisa menyelesaikan skripsi ini, meskipun penulis menyadari masih banyak kekurangan. Sejak awal sampai dengan akhir penulisan ini, tidak sedikit bantuan yang penulis terima sehingga dalam kesempatan ini penulis ingin menyampaikan terima kasih kepada Allah SWT yang telah memberikan berkah, rahmat dan hidayah-Nya hingga terselesaikannya skripsi ini:

1. Prof. Dr. H. Imam Suprayogo, selaku rektor UIN Maliki Malang.
2. Prof. Drs. H. Sutiman Bambang S.,SU.,Dsc, selaku dekan Fakultas Sains dan Teknologi UIN Maliki Malang.
3. Ririen Kusumawati, S.Si, M.Kom selaku ketua jurusan Teknik Informatika Fakultas Sains dan Teknologi UIN Maliki Malang, sekaligus sebagai dosen wali.

4. Syahiduz Zaman, M.Kom. selaku dosen pembimbing dalam pengerjaan Skripsi ini yang telah memberikan kritik, saran, kemudahan dan kepercayaan dalam pengerjaan skripsi ini.
5. M. Ainul Yaqin, M. Kom. selaku dosen pembimbing integrasi sains dan Islam, yang telah memberi masukan, saran serta bimbingan dalam proses menyelesaikan skripsi ini.
6. Teman-teman Teknik Informatika angkatan 2004, yang selalu memberi dukungan untuk menyelesaikan skripsi ini, Zaenal, Agung, Andre, Jaki, Ivana, Brutus, Ciplek, Tq6, terutama Danbul.
7. Keluarga besarku Mapala Tursina, yang selalu membuatku menangis untuk memberi ilmu hidup, terutama angkatan X angkatan diklatku (Talkeh, Ngatiri, Lowo, Cengkring, Cemet, Manyun, Ciput, Canteng, Sinden, dan yang mati lainnya). Angkatan-angkatan yang pernah berjuang bareng denganku tahun 2008-2009: Angkatan XI (Paimin, Tamadi, Soak, Kopro, Jablay dan yang mati lainnya), angkatan XII (Kawul, Bedug, Benjo, Tunggeng, Kempren, Peot, Gigis, Bengkek, Keset, dan yang mati lainnya), angkatan XIII adik-adik kebanggaanmu engkau lahir dariku (Bimo, Mengis, Kebox, Gomel, Oon, dan yang mati lainnya), angkatan buncitku, ksatria angkatan XIV ingat, pintar dulu baru nakal (Bunyam, Bagor, Kacing, Sumeh, Kumes, Leak, Kebo, dan yang akan mati lainnya). Dan para pendahulukmu mas Klowor, kang Rengges, kang Beset, cak Tobil. Viva Mapala Tursina! Viva Mapala Tursina! Viva Mapala Tursina.

8. Dan semua pihak yang tidak dapat saya sebutkan satu-persatu, kami ucapkan terimakasih banyak atas bantuan, dan motivasinya.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna, oleh karena itu kritik dan saran yang sifatnya membangun sangat diperlukan untuk memperbaiki mutu penulisan selanjutnya.

Malang, 31 Juli 2010

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERNYATAAN.....	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
HALAMAN MOTTO.....	v
HALAMAN PERSEMBAHAN	vi
KATA PENGANTAR	vii
DAFTAR ISI	x
DAFTAR GAMBAR.....	xiii
DAFTAR TABEL	xv
DAFTAR LAMPIRAN.....	xvi
ABSTRAK	xvii
BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	7
1.3 Batasan Masalah	7
1.4 Tujuan Penelitian	8
1.5 Manfaat Penelitian	8
1.6 Metode Penelitian	9
1.7 Sistematika Penulisan	11
BAB II KAJIAN PUSTAKA	
2.1 Profil Kabupaten Sragen	12
2.1.1 Sejarah Kabupaten Sragen	15
2.2 Sungai Bengawan Solo	18
2.2.1 Bagian Sungai.....	18
2.3 Banjir	19
2.4 Pengertian Simulasi	22
2.5 Sistem Informasi.....	23
2.6 Sistem Informasi Geografis.....	24
2.6.1 Pengertian Sistem Informasi Geografis	28
2.6.2 Subsistem Sistem Informasi Geografis.....	29
2.7 Arc View GIS	30
2.7.2 Project Arcview	31
2.7.2 Views	31
2.7.3 Tables.....	32
2.7.4 Charts	32
2.7.5 Layouts.....	32
2.7.6 Scripts	32
2.8 Web Sistem Informasi Geografis	38

2.9	Mapserver.....	42
2.9.1	Sejarah Perkembangan Mapserver	44
2.9.2	Arsitektur Mapserver	45
2.9.3	MapServer for Windows (MS4W)	48
2.9.4	Mapscript Environment	49
2.9.5	Map File	53
2.10	PHP (Personal Home Page).....	54
2.11	PostgreSQL	56
2.12	PostGIS	63
2.13	PgRouting.....	64
2.13.1	Algoritma Dijkstra	65
2.14	Permodelan Sistem	67
2.14.1	Data Flow Diagram.....	67
2.14.2	Entity Relationship Diagram.....	69
2.14.2.1	Kardinalitas Relasi	70
2.14.2.2	Tahapan pembuatan <i>ERD</i>	71
BAB III ANALISIS DAN PERANCANGAN APLIKASI		
3.1	Analisis Sistem	73
3.1.1	Spesifikasi Aplikasi	73
3.1.2	Spesifikasi Pengguna	74
3.1.3	Lingkungan operasi	74
3.2	Perancangan Sistem	75
3.3	Perancangan Aplikasi WEB	76
3.3.1	Perancangan Modul Aplikasi	76
3.3.1.1	Modul Aplikasi Peta.....	76
3.3.1.2	Modul Aplikasi Web.....	79
3.4	Perancangan Basis Data	79
3.4.1	Konteks diagram.....	80
3.4.2	Data Flow Diagram.....	82
3.4.3	Entity Relationship Diagram (ERD).....	91
3.4.4	Tabel data.....	93
BAB IV HASIL DAN PEMBAHASAN		
4.1	Implementasi	100
4.1.1	Ruanglingkup Perangkat Keras	101
4.1.2	Ruanglingkup Perangkat Lunak	101
4.1.3	Implementasi Database	102
4.1.4	Implementasi Antarmuka	103
4.2	Pembahasan	111
4.2.1	Input Data.....	111
4.2.2	Edit Data	119
4.2.3	Tampilan Peta	125

BAB V	PENUTUP	
5.1	Kesimpulan.....	157
5.2	Saran	158
DAFTAR PUSTAKA	159
LAMPIRAN	161

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Arsitektur WEB SIG.....	40
Gambar 2.2 Perbandingan Thin dan Thick Sistem Pada Sistem Clie Server	41
Gambar 2.3 Arsitektur Peta Dalam Halaman Web.....	46
Gambar 2.4 Arsitektur Mapserver	48
Gambar 2.5 Proses Mapscript Pada Web Server.....	50
Gambar 2.6 Cara Kerja Modul pgRouting.....	64
Gambar 3.1 Alur proses modul visualisasi data Peta.....	77
Gambar 3.2 Alur kerja modul query posisi.....	78
Gambar 3.3 Aliran data secara umum.....	79
Gambar 3.4 Konteks Diagram SIG Banjir	81
Gambar 3.5 DFD Level 1 SIG Banjir	83
Gambar 3.6 DFD Level 2 Proses Login.....	85
Gambar 3.7 DFD Level 2 Dari Proses Informasi	86
Gambar 3.8 DFD Level 2 Dari Proses Administrator.....	88
Gambar 3.9 DFD Level 2 Dari Proses Peta	90
Gambar 3.10 Entity Relationship Diagram SIG Banjir	92
Gambar 4.1 Diagram Konversi Data	103
Gambar 4.2 Halaman Utama Admin	105
Gambar 4.3 Halaman Utama User	106
Gambar 4.4 Halaman Utama Pengunjung.....	109
Gambar 4.5 Form Pencarian.....	111
Gambar 4.6 Form Hubungi Kami.....	112
Gambar 4.7 Form Tambah User	113
Gambar 4.8 Form Pembagian Modul.....	115
Gambar 4.9 Form Tambah Berita.....	116
Gambar 4.10 Form Tambah agenda	117
Gambar 4.11 Form Tambah Pengumuman.....	118
Gambar 4.12 Form Edit User	119
Gambar 4.13 Form Edit Modul	120
Gambar 4.14 Form Edit Agenda.....	121
Gambar 4.15 Form Edit Pengumuman	122
Gambar 4.16 Form Edit Banner	123
Gambar 4.17 Form Edit Data Banjir.....	124
Gambar 4.18 Form Edit Data Desa.....	125
Gambar 4.19 Form Input Kedalaman air dan Pencarian rute.....	126
Gambar 4.20 Tampilan Peta Tidak Banjir	127
Gambar 4.21 Tampilan Peta Banjir Biasa.....	130
Gambar 4.22 Tampilan Peta Banjir Sedang (ketinggian 68 meter).....	131
Gambar 4.23 Tampilan Peta Banjir Tinggi (ketinggian 85 meter).....	132
Gambar 4.24 Form Pencarian Rute Banjir 85 meter	133
Gambar 4.25 Tampilan Hasil Pencarian Rute Banjir 85 meter	135
Gambar 4.26 Tampilan Peta Hasil Pencarian Rute Banjir 85 meter	139

Gambar 4.27 Tampilan Hasil Query Ketinggian Banjir	140
Gambar 4.28 Tampilan Hasil Query Informasi Desa	142
Gambar 4.29 Alur aplikasi	143

DAFTAR TABEL

	Halaman
Table 2.1 Prinsi-prinsip Pengembangan SIG	39
Table 2.2 Perkembangan MapServer.....	44
Tabel 3.1 Simbol dalam DFD.....	80
Tabel 3.2 User	93
Tabel 3.3 Agenda	93
Tabel 3.4 Banner	94
Tabel 3.5 Hubungi	94
Tabel 3.6 Berita	94
Table 3.7 Modul	95
Tabel 3.8 Pengumuman.....	95
Tabel 3.9 Kategori	95
Tabel 3.10 Dijkstra Result.....	96
Tabel 3.11 Jalan	96
Tabel 3.12 Kec/posko	96
Tabel 3.13 Kontur	97
Tabel 3.14 Geometry Column	97
Tabel 3.15 Sungai	97
Tabel 3.16 Desa	98
Tabel 3.17 Banjir	98
Tabel 3.18 Bangunan	99

DAFTAR LAMPIRAN

Lampiran A Proses Instalasi <i>MapServer</i> dan PostgreSQL	161
Lampiran A Tutorial PgRouting (Dijkstra).....	177

ABSTRAK

Murdiyanto. 2010. 04550073. **Simulasi Daerah Banjir Menggunakan Sistem Informasi Geografis di Kabupaten Sragen.**

Pembimbing : (I) Syahiduz Zaman, M.Kom, (II) M. Ainul Yaqin, M.Kom.

Kata Kunci : Banjir, Sistem Informasi Geografis, *Mapserver*, *PostgreSQL*

Pembuatan Web-GIS salah satunya didorong karena penggunaan internet yang sangat luas dimasyarakat dan pemerintah, karena internet maka peta sekarang bisa diakses oleh semua pihak. Oleh karena itu dibuatlah sebuah Web-Gis simulasi banjir. Karena kejadian banjir akhir tahun 2007, membuat masyarakat dan pemerintah kaget dalam hal penanganan, salah satunya diakibatkan karena sedikitnya informasi tentang banjir.

Simulasi merupakan suatu teknik meniru operasi-operasi atau proses-proses yang terjadi dalam suatu sistem dengan bantuan perangkat komputer dan dilandasi oleh beberapa asumsi tertentu sehingga sistem tersebut bisa dipelajari secara ilmiah.

Simulasi banjir ini menggunakan peta kontur dan peta desa sebagai penentu daerah yang terkena banjir, dengan memasukan sebuah inputan maka, akan diketahui prediksi luapan banjir dan dapat juga mencari rute terpendek dalam keadaan banjir. Aplikasi penentu rute terpendek ini dibuat menggunakan fungsi *shortest path Dijkstra* yang dimiliki modul *pgRouting*, yang merupakan fungsi tambahan dari *PostgreSQL/PostGIS* untuk menangani masalah *routing* pada peta geografis. Aplikasi ini juga menggunakan *mapscript* sebagai modul yang digunakan untuk membuat fungsi dan class *MapServer* agar dapat dijalankan di PHP. Sedangkan *MapServer* sendiri adalah sebuah layanan untuk memproses dan menampilkan data spasial yang berasal dari database. Untuk sistem manajemen basisdatanya, digunakan DBMS *PostgreSQL/PostGis* yang sudah mendukung tipe data spasial.

Dengan adanya aplikasi ini, maka dapat diketahui daerah yang terkena banjir, dan dapat juga mengetahui rute jalan yang bisa dilalui dalam keadaan banjir.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Tidak ada orang yang mengharapkan sebuah musibah. Namun, suka tidak suka musibah tetap datang, baik yang bersifat lokal, regional, maupun internasional. Tidak hanya dalam hitungan tahun, bahkan juga hitungan jam. Bencana angin kencang yang mengakibatkan robohnya rumah, tanaman, pepohonan, gardu listrik, dan sebagainya. Bencana banjir yang menenggelamkan banyak infrastruktur, banyak menimbulkan kerusakan dan kerugian. Bencana itu tidak hanya menghancurkan harta benda, namun juga nyawa manusia, meninggal secara langsung maupun tidak langsung karena terkena penyakit akibat dari sebuah bencana.

Musibah memang membuat manusia menderita. Namun dilihat dari sudut mental spiritual, musibah merupakan jalan naik menuju hidup yang lebih baik. Musibah merupakan momentum atau isyarat bagi manusia menuju perubahan. Sebab, musibah memberi peluang kepada manusia untuk mempelajari pengalaman menyedihkan, hingga manusia menemukan kehidupan sejati dan hakikat keberadaan manusia di alam semesta ini. Allah SWT berfirman dalam surat Al Baqarah, ayat 155-156.

وَلَنَبْلُوَنَّكُمْ بِشَيْءٍ مِّنَ الْخَوْفِ وَالْجُوعِ وَنَقْصٍ مِّنَ الْأَمْوَالِ وَالْأَنْفُسِ
 وَالْثَّمَرَاتِ ۗ وَبَشِيرِ الصَّابِرِينَ ﴿٥٥﴾ الَّذِينَ إِذَا أَصَابَتْهُمُ مُصِيبَةٌ قَالُوا إِنَّا لِلَّهِ
 وَإِنَّا إِلَيْهِ رَاجِعُونَ ﴿٥٦﴾

“Sesungguhnya akan Kuberikan kepadamu kecemasan, kelaparan, kekurangan harta dan jiwa serta buah-buahan, tetapi berbahagialah orang yang bersabar,” yakni orang yang ketika menemui musibah berkata: “Sesungguhnya kita kepunyaan Allah dan kepada-Nya kita akan kembali.”

"Mereka itulah yang mendapat keberkatan yang sempurna dan rahmat dari Tuhan mereka, dan mereka itulah orang-orang yang mendapat petunjuk.

Keterangan tafsir Jalalain tentang ayat tersebut adalah (Dan sungguh Kami akan memberimu cobaan berupa sedikit ketakutan) terhadap musuh, (kelaparan) paceklik, (kekurangan harta) disebabkan datangnya malapetaka, (dan jiwa) disebabkan pembunuhan, kematian dan penyakit, (serta buah-buahan) karena bahaya kekeringan, artinya Kami akan menguji kamu, apakah kamu bersabar atau tidak. (Dan sampaikanlah berita gembira kepada orang-orang yang sabar) bahwa mereka akan menerima ganjaran kesabaran itu berupa surga.

Jangan dianggap bahwa semua bencana adalah azab, karena terkadang bencana adalah jembatan bagi orang beriman untuk menambahkan keimannya, dan membawa manusia lebih taat kepada tuhanNya. Dengan bersabar manusia diharapkan mendapatkan ganti yang lebih baik dari Allah SWT. Seperti kisah Ummu Salamah yang diceritakan dikitab tafsir Ibnu Katsir ketika beliau mendapatkan musibah, beliau memohon kepada Allah SWT agar mendapatkan pahala dari musibahnya dan diberi ganti yang lebih baik, dan Allahpun mengabulkanya.

Oleh karena itu, ketika musibah menimpa, selain pasrah (tawakkal) kepada Allah, manusia juga harus berusaha keras agar musibah tersebut tidak terulang kembali. Karenanya, manusiapun perlu melakukan *muhasabah* atau mengevaluasi diri sendiri akan dosa dan kesalahan apa yang telah diperbuat pada masa-masa sebelumnya. Terkadang manusia lupa akan nikmat dan anugerah yang telah diberikan Tuhan kepada manusia, tanah yang subur, udara yang segar, tidak terkecuali air yang bersih, yang semuanya itu berguna bagi kehidupan manusia.

Begitu bergunanya air bagi kehidupan manusia, dengan air manusia dapat digunakan minum, mencuci, mandi, irigasi dan lainnya. Namun, air juga dapat mendatangkan musibah jika air itu terlalu berlebihan, misalnya air sungai terlalu banyak yang akhirnya mengakibatkan terjadi banjir. Seperti musibah banjir yang terjadi pada manusia pada zaman dahulu yang telah diceritakan dalam Al-Qur'an surah Saba' ayat 16:

فَأَعْرَضُوا فَأَرْسَلْنَا عَلَيْهِمْ سَيْلَ الْعَرِمِ وَبَدَّلْنَاهُمْ بِجَنَّتَيْهِمْ جَنَّتَيْنِ ذَوَاتِ أُكُلٍ خَمْطٍ وَأَثَلٍ
وَشَيْءٍ مِّن سِدْرٍ قَلِيلٍ ﴿١٦﴾

“Tetapi mereka berpaling, Maka kami datangkan kepada mereka banjir yang besar dan kami ganti kedua kebun mereka dengan dua kebun yang ditumbuhi (pohon-pohon) yang berbuah pahit, pohon Atsl dan sedikit dari pohon Sidr (Qs. Saba’/34: 16).

Menurut Tafsir Jalalain pada ayat itu menjelaskan “(Tetapi mereka berpaling) tidak mau bersyukur kepada-Nya dan bahkan mereka kafir kepada-Nya (maka Kami datangkan kepada mereka banjir yang besar) lafal *Al 'Arim* adalah bentuk jamak dari lafal *'Urmah* yang artinya adalah bendungan yang menampung air sampai waktu yang dibutuhkan. Maksudnya dam yang membendung

kebutuhan air mereka pecah sehingga menenggelamkan kebun-kebun dan harta benda mereka (dan Kami ganti kedua kebun mereka dengan dua kebun yang ditumbuhi) lafal *Dzawaatai* merupakan bentuk *Tatsniyah* dari lafal *Dzawaatun* yang *Mufrad* (pohon-pohon yang berbuah pahit) yang sangat pahit buahnya lagi tidak enak rasanya; dapat dibaca *Ukuli Khamthin* yaitu dengan di-*mudhaf*-kan, lafal *Ukulin* ini bermakna *Maakuulin* yaitu yang dimakan sebagaimana dapat pula dibaca *Ukulin Khamthin*, lalu dibaca *Ukulin Wa Khamtin* (pohon Atsl dan sedikit pohon Sidr)”.

Adanya musibah ini, menunjukkan betapa lemahnya manusia di hadapan Allah SWT. Sehingga semua hal yang dibanggakan manusia di dunia ini, tidak akan ada artinya, manakala Allah SWT menghendaki kehancurannya.

Kejadian banjir tidak hanya terjadi pada zaman dahulu, banjir juga terjadi disetiap masa, tidak terkecuali pada zaman sekarang ini, seperti halnya banjir yang pernah terjadi di sungai Bengawan Solo, yang merupakan sungai terpanjang di pulau Jawa, panjangnya melewati dua provinsi, yaitu provinsi Jawa Tengah dan provinsi Jawa Timur. Dalam dua provinsi itu terdapat beberapa kota/kabupaten termasuk juga kabupaten Sragen yang dilewati sungai tersebut. Sungai Bengawan Solo dengan panjang ± 600 km mengalir dari Wonogiri Propinsi Jawa Tengah sampai ke Ujung Pangkah, Gresik Propinsi Jawa Timur, merupakan sumber air potensial bagi usaha-usaha Pengembangan Sumber Daya Air (PSDA) untuk memenuhi berbagai keperluan antara lain:

- a. Air Baku, Industri dan Rumah Tangga
- b. Penyediaan air Irigasi

- c. Pariwisata
- d. Pembangkit Listrik Tenaga Air (PLTA)
- e. Sarana Transportasi
- f. Dan lain-lain

Daerah pengaliran sungai Bengawan Solo Hulu secara keseluruhan meliputi wilayah seluas $\pm 6.072 \text{ km}^2$ dan wilayah kabupaten Sragen termasuk didalamnya dengan luas $\pm 941,55 \text{ km}^2$.

Banjir yang hampir selalu terjadi setiap tahunnya di Wilayah Sungai (WS) Bengawan Solo disebabkan oleh kondisi topografi yang kurang baik, dimana sebagian besar wilayahnya berada pada dataran rendah dan alur sungai yang berkelok-kelok (*meandering*). Tidak seperti tipologi sungai di pegunungan yang penuh dengan jeram dan terjal.

Kejadian banjir pada bulan Desember tahun 2007 merupakan kejadian banjir besar sebagaimana tahun 1966 dan tahun 1994 di sungai Bengawan Solo dan anak-anak sungai Bengawan Solo, yang mengakibatkan areal tergenang mencapai $\pm 45.000 \text{ ha}$ meliputi wilayah Surakarta, Sragen, Ngawi, Bojonegoro, Tuban, Lamongan, Gresik. Kejadian banjir itu sangat merugikan warga, mulai dari kerugian material maupun non material. Dan juga berbagai kerusakan fasilitas umum serta kerugian-kerugian yang ditimbulkannya, antara lain rusaknya prasarana pengairan (bendungan, irigasi, tanggul), rusaknya prasarana transportasi umum, rusaknya pemukiman dan pertanian (rumah tinggal, sawah, tambak, dan lain-lain), kegagalan panen, gangguan kesehatan, timbulnya korban jiwa,

pengungsian penduduk, terganggunya pelaksanaan pendidikan, dan pelayanan umum yang lainnya.

Kejadian banjir itu sangat mengagetkan karena sangat jarang terjadi banjir besar, oleh karena itu pemerintah terkesan lambat dalam hal ketanggapan dan penanggulangnya. Akhirnya yang terjadi di lapangan, masyarakat menjadi sangat bingung. Dalam keadaan mencekam, karena kurangnya sebuah informasi, banyak isu yang terjadi di masyarakat. Isu bahwa banjir akan terus bertambah, atau daerah tertentu tidak bisa disuplai bantuan karena daerah tersebut terisolasi, sehingga distribusi bantuan barang dan obat-obatan sangat terganggu.

Sulit dalam memprediksi bencana khususnya bencana banjir di kabupaten Sragen, karena kabupaten Sragen berada di daerah tengah sehingga meskipun di kabupaten Sragen ini hujannya biasa (tidak terlalu deras) namun daerah atas (hulu yang lebih atas) hujan sangat deras dan terus menerus, bisa saja banjir akan terjadi dan berkembang menjadi besar di kabupaten Sragen sebagai air kiriman. Oleh karena itu, untuk mengantisipasi, meminimalisir dan untuk membantu pemerintah dalam tanggap dan penanggulangan banjir dibuatlah sebuah sistem informasi yang bisa membantu dalam penanganan banjir di daerah tengah khususnya kabupaten Sragen. Sebagai manusia memang harus bangkit dan berusaha untuk menanggulangi bencana, karena bencana akan menjadi hikmah jika manusia bisa memahami dan mengambilnya.

Diketahui penggunaan internet sekarang sudah sangat masif di kalangan masyarakat dan pemerintah, maka dari itu dibuatlah sebuah sistem informasi yang bisa memberikan gambaran secara geografis, tentang kejadian banjir di kabupaten

Sragen. Sistem ini diharapkan bisa berguna bagi pemerintah, tapi juga bisa berguna bagi para donatur yang daerahnya tidak terkena bencana untuk membantu. Selain itu sistem ini diharapkan berguna juga buat masyarakatnya untuk memenuhi kebutuhan informasi. Oleh karena itu dibuatlah sebuah simulasi banjir.

Kenapa simulasi banjir, menurut Veriawan (2010) yang mengutip dari Law dan Kelton (1991) “simulasi adalah suatu teknik meniru operasi-operasi atau proses-proses yang terjadi dalam suatu sistem dengan bantuan perangkat komputer dan dilandasi oleh beberapa asumsi tertentu sehingga sistem tersebut bisa dipelajari secara ilmiah”.

Karena sistem atau program ini belum teruji di lapangan, dan juga bukan sebagai *record* sejarah banjir yang terjadi di kabupaten Sragen. Maka sistem ini, diberi judul Simulasi Daerah Banjir Menggunakan Sistem Informasi Geografis di Kabupaten Sragen.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka rumusan masalah dalam penelitian ini adalah: *Bagaimana mensimulasikan daerah banjir dengan sistem informasi geografis di kabupaten Sragen berbasis web?*

1.3 Batasan Masalah

Batasan masalah pada penelitian ini adalah membangun dan membuat website sistem informasi geografis yang dapat menampilkan daerah yang

tergenang banjir berdasarkan kontur tanah dan Tinggi Muka Air (TMA) serta dapat mencari rute jalan terpendek antar kecamatan atau posko. Dan hanya terbatas di Wilayah Sungai Bengawan Solo (WSBS) kabupaten Sragen.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah bagaimana merancang dan membuat *website* Simulasi Daerah Banjir Menggunakan Sistem Informasi Geografis di Kabupaten Sragen sebagai model acuan untuk penanggulangan banjir.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini diharapkan dapat:

1. Membantu masyarakat untuk memperoleh informasi tentang kawasan yang sering terkena banjir dan daerah yang potensi terkena banjir sehingga dapat mempersiapkan diri.
2. Membantu pemerintah untuk memetakan kawasan-kawasan desa yang terkena banjir, sehingga dapat cepat dalam penanggulangannya.
3. Membantu pemerintah maupun donatur mencari rute untuk mendistribusikan bantuan baik barang maupun obat-obatan.
4. Meningkatkan dan mengembangkan kemampuan diri dalam lingkungan yang sesuai dengan disiplin ilmu yang dipelajari.
5. Menambah wawasan, memperdalam pengetahuan, dan menimba pengalaman dalam penelitian ini sehingga dapat menjadi tenaga kerja

yang berguna dan siap untuk menghadapi segala macam tantangan dalam dunia kerja sesungguhnya.

1.6 Metode Penelitian

Penyusunan laporan penelitian ini melalui beberapa metode yang dipakai, yaitu:

a. Persiapan

Yang termasuk dalam persiapan ini adalah diawali dengan penyusunan proposal dan mengurus perijinan dalam penelitian di samping juga belajar penggunaan *software* pendukung.

b. Pengumpulan Data

Dalam pengumpulan data terdiri dari tiga kegiatan, yaitu:

1. Konsultasi langsung dengan pihak yang ahli pada bidang tersebut, dalam hal ini adalah dosen pembimbing, pihak dan orang-orang yang dianggap tahu dan mampu dalam bidang ini.
2. Studi Literatur untuk mendapatkan literatur yang telah ada yang berupa buku dan jurnal.
3. Mencari dan mengambil beberapa data di instansi pemerintah maupun perorangan.

c. Analisis Data

Data yang diperoleh dari pengumpulan data di atas kemudian dianalisis, diolah sesuai kebutuhan program. apabila terjadi kekurangan data dapat dilakukan penambahan, begitu juga sebaliknya.

d. Perancangan Sistem

Setelah data terkumpul dan dianalisis kemudian dilakukan perancangan sistem yang terdiri dari perancangan alur sistem, *database* dan perancangan tampilan aplikasi (*interface*).

e. Evaluasi Sistem

Sebelum menuju proses pembuatan aplikasi terlebih dahulu sistem yang telah disusun di perancangan sistem dievaluasi guna mengetahui kekurangan-kekurangan sementara dari sistem tersebut.

f. Pembuatan Aplikasi

Dalam proses ini dilakukan pembuatan tampilan sistem, pembuatan *database*, dan penyusunan *coding* program.

g. Pengujian Sistem

Dalam pengujian sistem, aplikasi akan diuji dengan cara memasukkan data yang sudah diperoleh. Jika masih terjadi kesalahan maka akan diperiksa kembali mulai dari perancangan sistem sampai didapat hasil yang maksimal. Dalam tahapan ini juga dilakukan *trial and error* jika terjadi kesalahan, untuk menemukan *performa* terbaik.

h. Dokumentasi dan Penyusunan Laporan

Setelah dilakukan pengujian sistem kemudian dilakukan proses dokumentasi dan penyusunan laporan yang bertujuan untuk mempermudah pengembangan selanjutnya.

1.7 Sistematika Penulisan

BAB I Pendahuluan

Bab ini berisi latar belakang, perumusan masalah, tujuan, batasan masalah dan metodologi penelitian skripsi ini.

BAB II Kajian Pustaka

Bab ini menjelaskan konsep dan teori dasar yang mendukung penulisan skripsi ini seperti Sistem Informasi Geografis, PHP, Mapserver, PostgreSQL.

BAB III Analisis Dan Perancangan Aplikasi

Bab ini menjelaskan mengenai analisis dan perancangan Aplikasi Sistem Informasi Geografis Simulasi Banjir di Kabupaten Sragen secara keseluruhan.

BAB IV Hasil Dan Pembahasan

Bab ini berisi pengujian terhadap hasil pengujian dari aplikasi yang telah dibangun.

BAB V Penutup

Bab ini berisi kesimpulan dan saran terhadap seluruh kegiatan skripsi yang telah dilakukan.

BAB II

KAJIAN PUSTAKA

2.1 Profil Kabupaten Sragen

Kabupaten sragen merupakan salah satu kabupaten di propinsi Jawa Tengah. Secara geografis kabupaten Sragen berada di perbatasan antara Jawa Tengah dan Jawa Timur. Batas-batas wilayah kabupaten Sragen:

- a. Sebelah Timur : Kabupaten Ngawi (propinsi Jawa Timur)
- b. Sebelah Barat : Kabupaten Boyolali
- c. Sebelah Selatan : Kabupaten Karanganyar
- d. Sebelah Utara : Kabupaten Grobogan

Luas wilayah Kabupaten Sragen adalah 941,55 km² yang terbagi dalam 20 kecamatan, 8 kelurahan, dan 200 desa. Secara fisiologis, wilayah kabupaten Sragen terbagi atas 40.037,93 Ha (42,52%) lahan basah (sawah) dan 54.117,88 Ha (57,48%) lahan kering. **Kabupaten Sragen terletak pada 7 ° 15 LS dan 7 ° 30 LS 110 ° 45 BT dan 111 ° 10 BT.**

Wilayah kabupaten Sragen berada di dataran rendah dengan ketinggian rata rata 109 Mdpl. Sragen mempunyai iklim tropis dengan suhu harian yang berkisar antara 19-31 ° C. Curah hujan rata-rata di bawah 3000 mm per tahun dengan hari hujan di bawah 150 hari per tahun. Jumlah penduduk Sragen berdasarkan data tahun 2005 sebanyak 865.417 jiwa, terdiri dari 427.253 penduduk laki-laki dan 438.164 penduduk perempuan. Kepadatan penduduk rata-

rata 919 jiwa/km², luas wilayah: 94.155 Ha dengan luas Sawah: 40.129 Ha dan Tanah Kering: 54.026 Ha.

Dan kabupaten Sragen dibagi menjadi dua bagian, yaitu:

- a. Sebelah selatan Bengawan Solo: Luas wilayah 32.760 ha (34,79 %), Tanah sawah 22.027 ha (54,85 %), 9 Kecamatan, 88 Desa & Kelurahan.
- b. Sebelah utara Bengawan Solo: Luas wilayah 61.395 ha (65,21 %), Tanah sawah: 18.102 ha (45,15 %), 11 Kecamatan, 120 Desa.

Bengawan Solo adalah sungai terpanjang di pulau Jawa, dengan mata air dari daerah Wonogiri dan bermuara di daerah Gresik. Sungai ini panjangnya sekitar 548,53 km dan mengalir dua provinsi yaitu Jawa Tengah dan Jawa Timur. Kabupaten yang dilalui adalah Wonogiri, Pacitan, Sukoharjo, Klaten, Solo, Sragen, Ngawi, Blora, Bojonegoro, Tuban, Lamongan, dan Gresik.

Sragen berada di lembah daerah aliran sungai Bengawan Solo yang mengalir ke arah timur. Sebelah utara berupa perbukitan, bagian dari sistem pegunungan Kendeng. Sedangkan di selatan berupa pegunungan, lereng dari gunung Lawu.

Sragen terletak di jalur utama Solo-Surabaya. Kabupaten ini merupakan gerbang utama sebelah timur provinsi Jawa Tengah, yang berbatasan langsung dengan provinsi Jawa Timur. Sragen dilintasi jalur Kereta Api lintas selatan pulau Jawa (Surabaya-Yogyakarta-Jakarta) dengan stasiun terbesarnya Sragen, serta lintas Semarang-Solo dengan stasiun terbesarnya Gemolong.

Kecamatan adalah pembagian wilayah administratif di Indonesia di bawah kabupaten atau kota. Kecamatan terdiri atas desa-desa atau kelurahan-kelurahan.

Kabupaten Sragen terdiri atas 20 kecamatan, yang dibagi lagi atas sejumlah 208 desa dan kelurahan. Pusat pemerintahan berada di kecamatan Sragen.

Dua puluh kecamatan tersebut adalah Gemolong, Ngrampal, Plupuh, Sambirejo, Sambungmacan, Sragen, Sidoharjo, Sukodono, Sumberlawang, Tangen, Tanon, Gesi, Gondang, Jenar, Kalijambe, Karangmalang, Kedawung, Masaran, Miri.

Kabupaten Sragen dipetakan menjadi dua wilayah yaitu wilayah utara Bengawan Solo dan selatan Bengawan Solo, dengan pembagian seperti berikut:

- a. Utara : 11 kecamatan, 116 desa dan 4 kelurahan, dengan potensi:
pertanian, pariwisata, industri dan perdagangan.
- b. *Selatan* : 9 kecamatan, 80 desa dan 8 kelurahan, tanah relatif lebih subur,
dengan potensi: pertanian sawah, perdagangan, industri,
pariwisata.

Keadaan alam di kabupaten Sragen mempunyai relief yang beraneka ragam, ada daerah pegunungan kapur yang membentang dari timur ke barat terletak di sebelah utara bengawan Solo dan dataran rendah yang tersebar di seluruh kabupaten Sragen, dengan jenis tanah: gromusol, alluvial regosol, latosol dan mediteran.

Dan keadaan klimatologi kabupaten Sragen dengan iklim tropis dan temperatur sedang dengan cuah hujan rata-rata di bawah 3.000 mm/tahun dan hari hujan dengan rata-rata dibawah 150 hari/tahun.

Keterangan profil kabupaten Sragen di atas berdasarkan sumber resmi yang diambil dari situs resmi Pemerintah Kabupaten Sragen tahun 2010 tentang Geografi Kabupaten Sragen.

2.1.1 Sejarah Kabupaten Sragen

Hari jadi kabupaten Sragen ditetapkan dengan Perda Nomor: 4 Tahun 1987, yaitu pada hari Selasa Pon, tanggal 27 Mei 1746. Tanggal dan waktu tersebut adalah dari hasil penelitian serta kajian pada fakta sejarah, ketika Pangeran Mangkubumi yang kelak menjadi Sri Sultan Hamengku Buwono yang ke- I menancapkan tonggak pertama melakukan perlawanan terhadap Belanda menuju bangsa yang berdaulat dengan membentuk suatu pemerintahan lokal di desa Pandak, Karangnongko masuk *tlatah* Sukowati sebelah timur. Dengan kronologi dan prosesi sebagai berikut:

Pangeran Mangkubumi adik dari Sunan Pakubuwono II di Mataram sangat membenci Kolonialis Belanda. Apalagi setelah Belanda banyak meng*intervensi* Mataram sebagai pemerintahan yang berdaulat. Oleh karena itu dengan tekad yang menyala bangsawan muda tersebut lolos dari istana dan menyatakan perang dengan Belanda. Dalam sejarah peperangan tersebut, disebut dengan Perang Mangkubumen (1746 - 1757).

Dalam perjalanan perangnya pangeran muda dengan pasukannya dari Keraton bergerak melewati desa-desa Cemara, Tingkir, Wonosari, Karangsari, Ngerang, Butuh, Guyang. Kemudian melanjutkan perjalanan ke Desa Pandak, Karangnongko.

Di desa ini pangeran Mangkubumi membentuk pemerintahan pemberontak. Desa Pandak, Karangnongko dijadikan pusat pemerintahan Projo Sukowati, dan beliau meresmikan namanya menjadi pangeran Sukowati serta mengangkat pula beberapa pejabat Pemerintahan.

Karena secara geografis terletak di tepi jalan lintas tentara kompeni Surakarta–Madiun, pusat pemerintahan tersebut dianggap kurang aman, maka kemudian sejak tahun 1746 dipindahkan ke desa Gebang yang terletak disebelah tenggara desa Pandak Karangnongko.

Sejak itu pangeran Sukowati memperluas daerah kekuasaannya meliputi desa Krikilan, Pakis, Jati, Prampalan, Mojoroto, Celep, Jurangjero, Grompol, Kaliwuluh, Jumbleng, Lajersari dan beberapa desa Lain.

Dengan daerah kekuasaan serta pasukan yang semakin besar pangeran Sukowati terus menerus melakukan perlawanan kepada Kompeni Belanda bahu-membahu dengan saudaranya Raden Mas Said, yang berakhir dengan perjanjian Giyanti pada tahun 1755, yang terkenal dengan perjanjian Palihan Negari, yaitu kasunanan Surakarta dan Kasultanan Yogyakarta, dimana Pangeran Sukowati menjadi Sultan Hamengku Buwono ke-1 dan perjanjian Salatiga tahun 1757, dimana Raden Mas Said ditetapkan menjadi Adipati Mangkunegara I dengan mendapatkan separuh wilayah Kasunanan Surakarta.

Selanjutnya sejak tanggal 12 Oktober 1840 dengan Surat Keputusan Sunan Paku Buwono VII yaitu serat Angger–angger Gunung, daerah yang lokasinya setrategis ditunjuk menjadi Pos Tundan, yaitu tempat untuk menjaga

ketertiban dan keamanan lalu lintas barang dan surat serta perbaikan jalan dan jembatan, termasuk salah satunya adalah Pos Tundan Sragen.

Perkembangan selanjutnya sejak tanggal 5 Juni 1847 oleh Sunan Paku Buwono VIII dengan persetujuan Residen Surakarta Baron de Geer ditambah kekuasaan yaitu melakukan tugas kepolisian dan karenanya disebut Kabupaten Gunung Pulisi Sragen. Kemudian berdasarkan Staatsblaad No 32 Tahun 1854, maka disetiap Kabupaten Gunung Pulisi dibentuk Pengadilan Kabupaten, dimana Bupati Pulisi menjadi Ketua dan dibantu oleh Kliwon, Panewu, Rangga dan Kaum.

Sejak tahun 1869, daerah kabupaten Pulisi Sragen memiliki empat Distrik, yaitu Distrik Sragen, Distrik Grompol, Distrik Sambungmacan dan Distrik Majenang. Selanjutnya sejak Sunan Paku Buwono VIII dan seterusnya diadakan reformasi terus menerus dibidang pemerintahan, di mana pada akhirnya kabupaten Gunung Pulisi Sragen disempurnakan menjadi kabupaten Pangreh Praja. Perubahan ini ditetapkan pada jaman pemerintahan Paku Buwono X, Rijkblaad No. 23 Tahun 1918, di mana kabupaten Pangreh Praja sebagai daerah otonom yang melaksanakan kekuasaan hukum dan pemerintahan.

Dan akhirnya memasuki zaman kemerdekaan pemerintah Republik Indonesia, kabupaten Pangreh Praja Sragen menjadi Pemerintah Daerah Kabupaten Sragen. Penjelasan tentang sejarah kabupaten Sragen di atas bersumber pada situs resmi Pemerintah Kabupaten Sragen tahun 2010 tentang Sejarah Sragen.

2.2 Sungai Bengawan Solo

Sungai adalah tempat-tempat dan wadah-wadah serta jaringan pengaliran air mulai dari mata air sampai muara dengan dibatasi kanan kirinya serta sepanjang pengalirannya oleh garis sempadan.

Bengawan **Solo** adalah sungai terpanjang di Pulau Jawa, Indonesia dengan mata air dari daerah Pergunungan Kidul, Wonogiri dan menemui laut di daerah Gresik. Sungai ini panjangnya sekitar 548.53 km dan mengalir di dua wilayah yaitu Jawa Tengah dan Jawa Timur. Kabupaten yang dilalui adalah Wonogiri, Pacitan, Sukoharjo, Klaten, Solo, Sragen, Ngawi, Blora, Bojonegoro, Tuban, Lamongan, dan Gresik.

2.2.1 Bagian Sungai

Sungai Bengawan Solo merupakan sungai terpanjang di Jawa, maka lingkup daerahnya sangat luas. Seperti sungai pada umumnya sungai Bengawan Solo mempunyai beberapa bagian, mulai dari daerah Hulu, Tengah, maupun Hilir. Seperti keterangan yang diambil dari wikipedia (2009) tentang Bengawan Solo :

a. Daerah Hulu

Daerah ini mayoritasnya meliputi daerah Hulu Kali Tenggar, Hulu Kali Muning, Hulu Waduk Gajah Mungkur serta sebahagian Kabupaten Wonogiri dengan penampang sungai yang berbentuk V. Tanaman pada daerah ini dikuasai oleh tumbuhan akasia. Aktivitas yang banyak dilakukan di daerah ini adalah pertanian, seperti padi dan kacang tanah. Benteng sungai pada daerah ini rata-rata bertebing curam dan tinggi.

Disebabkan banyak digunakan untuk pertanian, daerah sekitar sungai pada bahagian ini banyak mengalami hakisan dan mendapan yang cukup tinggi.

b. Daerah Tengah

Daerah ini meliputi daerah hilir Waduk Gajah Mungkur, sebagian kabupaten Wonogiri, Pacitan, Sukoharjo, Klaten, Solo, Sragen, sebagian kabupaten Ngawi dan sebahagian Tempuran (hilir) Kali Madiun. Selain itu, daerah ini merupakan daerah yang padat penduduknya. Pada umumnya, kegiatan ekonomi di daerah bagian sungai ini lebih tinggi berbanding bagian hulu dan hilir, dan didominasi oleh kegiatan industri. Akibatnya, banyak limbah yang masuk ke sungai dan mencemari tanaman di daerah ini. Aktivitas masyarakat yang paling menonjol di daerah ini adalah pertanian, penggunaan air sebagai keperluan harian, penternakan dan industri.

c. Daerah Hilir

Daerah ini meliputi daerah sebagian Tempuran (hilir) Kali Madiun, sebagian kabupaten Ngawi, Blora, Bojonegoro, Lamongan, Tuban dan berakhir di Desa Ujungpangkah, Gresik.

2.3 Banjir

Banjir adalah suatu keadaan sungai dimana aliran airnya tidak tertampung oleh palung sungai. Sedangkan palung sungai adalah bagian utama dari sungai berupa alur yang berfungsi untuk mengalirkan sejumlah air per satuan waktu (debit) dari bagian hulu kebagian hilir. Besar kecilnya palung sungai dan intensitas hujan di daerah hulu sangat berpengaruh pada frekuensi kejadian banjir.

Banjir merupakan suatu bencana alam, yang banyak merugikan manusia baik materi maupun non materi, bahkan nyawa manusia bisa hilang.

Bencana ada dua kategori, ada yang merupakan ujian sebagai penambah keimanan, ada pula sebagai azab seperti azab yang diberikan oleh Allah SWT kepada umat Nabi Nuh seperti yang dijelaskan dalam potongan (QS. Al-Ankabut[29.14]):

..... فَأَخَذَهُمُ الطُّوفَانُ وَهُمْ ظَالِمُونَ ﴿١٤﴾

..... Maka mereka ditimpa banjir besar, dan mereka adalah orang-orang yang zalim.

Dalam tafsir Jalalain dijelaskan (Maka mereka ditimpa banjir besar) yaitu, air bah yang sangat tinggi sehingga tenggelamlah mereka semuanya (dan mereka adalah orang-orang yang zalim) maksudnya adalah orang-orang yang menyekutukan Allah.

Jadi, dalam suatu kaum jika terjadi sebuah bencana tidak terkecuali banjir, jika umatnya sudah beriman dan taat, namun masih terkena bencana dan musibah dapat disimpulkan bencana itu merupakan sebuah ujian bagi para mukminin untuk meningkatkan keimanannya. Namun jika di suatu daerah itu kaumnya berbuat ingkar, maka musibah dan bencana di daerah itu merupakan azab yang pedih.

Terkadang Allah SWT memberikan ujian dulu kepada hambanya untuk membuktikan keteguhan imannya, seperti yang diterangkan dalam ayat di bawah ini:

أَمْ حَسِبْتُمْ أَنْ تَدْخُلُوا الْجَنَّةَ وَلَمَّا يَأْتِكُمْ مَثَلُ الَّذِينَ خَلَوْا مِنْ قَبْلِكُمْ مَسَّتْهُمُ الْبَأْسَاءُ
وَالضَّرَّاءُ وَزُلْزِلُوا حَتَّى يَقُولَ الرَّسُولُ وَالَّذِينَ ءَامَنُوا مَعَهُ مَتَى نَصْرُ اللَّهِ أَلاَ إِنَّ نَصْرَ اللَّهِ
قَرِيبٌ ﴿٢١٤﴾

“Apakah kamu mengira bahwa kamu akan masuk surga, padahal belum datang kepadamu (cobaan) sebagaimana halnya orang-orang terdahulu sebelum kamu. Mereka ditimpa oleh malapetaka dan kesengsaraan, serta digoncangkan (dengan bermacam-macam cobaan) sehingga berkatalah Rasul dan orang-orang yang beriman bersamanya, ‘Bilakah datangnya nashrullah (pertolongan Allah).’ Ingatlah, sesungguhnya pertolongan Allah itu amat dekat.” (QS. Al-Baqarah: 214).

Menurut tafsir Jalalain ayat ini diturunkan mengenai susah payah yang menimpa kaum muslimin: (Ataukah), maksudnya apakah (kamu mengira bahwa kamu akan masuk surga. padahal belum) maksudnya belum (datang kepadamu seperti) yang datang (kepada orang-orang yang terdahulu sebelum kamu) di antara orang-orang beriman berupa bermacam-macam cobaan, lalu kamu bersabar sebagaimana mereka bersabar? (Mereka ditimpa oleh); kalimat ini menjelaskan perkataan yang sebelumnya (malapetaka), maksudnya kemiskinan yang memuncak, (kesengsaraan) maksudnya penyakit, (dan mereka digoncang) atau dikejutkan oleh bermacam-macam bala, (hingga berkatalah) baris di atas atau di depan artinya telah bersabda (Rasul dan orang-orang yang beriman yang bersamanya) yang menganggap terlambatnya datang bantuan disebabkan memuncaknya kesengsaraan yang menimpa mereka, ("Bilakah) datangnya (pertolongan Allah) yang telah dijanjikan kepada kami?" Lalu mereka mendapat jawaban dari Allah, ("Ingatlah, sesungguhnya pertolongan Allah itu amat dekat") kedatangannya.

Dan menurut Syekh Ahmad Atailah:

Sebenarnya kesusahan dari bencana yang menimpamu, akan menjadi ringan, apabila kalian sudah mengetahui bahwa Allah Ta'ala sedang mengujimu. Sebab Dialah yang mencoba kamu melalui qadar-Nya. Dialah yang telah mengarahkan kamu untuk mengadakan pilihan yang paling baik.

Kesimpulannya, banjir merupakan sebuah bencana. Jika manusia beriman pasti akan menghadapi bencana dengan hati yang sabar dan rasa Optimis yang tinggi untuk segera bangkit menghadapi dan memikirkan cara untuk menanggulangnya.

2.4 Pengertian Simulasi

Menurut Ellyns (2009) menjelaskan tentang definisi simulasi sebagai berikut:

Simulasi dapat diartikan sebagai meniru suatu sistem nyata yang kompleks yang penuh dengan sifat *probabilistik*, tanpa harus mengalami keadaan yang sesungguhnya. Hal ini dapat dilakukan dengan membuat sebuah miniatur yang *representative* dan *valid* dengan tujuan *sampling* dan survei statistik pada sistem nyata yang dapat dilakukan pada tiruan ini. Jadi, simulasi mempelajari atau memprediksi sesuatu yang belum terjadi dengan cara meniru atau membuat model sistem yang dipelajari dan selanjutnya mengadakan eksperimen secara numerik dengan menggunakan komputer.

Dalam simulasi digunakan komputer untuk mempelajari sistem secara numerik, dimana dilakukan pengumpulan data untuk melakukan estimasi statistik untuk mendapatkan karakteristik asli dari sistem. Simulasi merupakan alat yang tepat untuk digunakan terutama jika diharuskan untuk melakukan eksperimen dalam rangka mencari komentar terbaik dari komponen-komponen sistem. Hal ini dikarenakan sangat mahal dan memerlukan waktu yang lama jika eksperimen dicoba secara riil. Dengan melakukan studi simulasi maka dalam waktu singkat

dapat ditentukan keputusan yang tepat serta dengan biaya yang tidak terlalu besar karena semuanya cukup dilakukan dengan komputer.

Ellyns(2009) menjelaskan lagi tentang simulasi sebagai berikut:

Proses simulasi juga berhubungan dengan penyusunan tiruan sistem dengan menggunakan interaksi antar bilangan random yang mengikuti distribusi dari pola data tertentu. Sehingga diperlukan suatu distribusi tertentu untuk mensimulasikan suatu sistem. Model simulasi ada dua macam :

1. Simulasi Analog

Yaitu simulasi yang mempergunakan representasi fisik untuk menjelaskan karakteristik penting dari suatu masalah. Contoh : model hidraulik sistem ekonomi makro.

2. Simulasi Simbolik

Pada dasarnya meniru model matematik yang pemecahannya (dipermudah) dengan menggunakan computer, disebut simulasi computer.

2.5 Sistem Informasi

Sistem informasi dapat didefinisikan sebagai berikut :

- a. Suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.
- b. Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan atau untuk mengendalikan organisasi.
- c. Suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

2.6 Sistem Informasi Geografis

هُوَ الَّذِي جَعَلَ الشَّمْسَ ضِيَاءً وَالْقَمَرَ نُورًا وَقَدَرَهُ مَنَازِلَ لِتَعْلَمُوا عَدَدَ السِّنِينَ وَالْحِسَابَ ۗ مَا خَلَقَ اللَّهُ ذَلِكَ إِلَّا بِالْحَقِّ يُفَصِّلُ الْآيَاتِ لِقَوْمٍ يَعْلَمُونَ ﴿٥﴾

“Dia-lah yang menjadikan matahari bersinar dan bulan bercahaya dan ditetapkan-Nya manzilah-manzilah (tempat-tempat) bagi perjalanan bulan itu, supaya kamu mengetahui bilangan tahun dan perhitungan (waktu). Allah tidak menciptakan yang demikian itu melainkan dengan hak. dia menjelaskan tanda-tanda (kebesaran-Nya) kepada orang-orang yang Mengetahui.” (QS. Yunus, 10 : 5).

Dari kutipan ayat di atas menjelaskan bahwa semua yang diciptakan Allah adalah bermanfaat dan mengandung penuh hikmah. Salah satu ciptaan Allah yang dijelaskan menurut ayat di atas diantaranya adalah matahari dan bulan. Kedua ciptaan Allah tersebut merupakan suatu penanda yang dapat digunakan dalam mengetahui bilangan tahun dan bulan. Pada zaman Yunani kuno para ilmuan telah menggunakan matahari untuk menghitung jarak antara dua buah tempat yang saling berjauhan dengan memanfaatkan matahari. Dengan matahari pula para ilmuan Yunani kuno telah berhasil menghitung diameter bumi, walaupun tingkat keakurasiannya masih kurang.

Pada ayat yang lain Allah juga menjelaskan fungsi dari bintang-bintang yang bersinar di malam hari, dan ayat tersebut adalah:

وَهُوَ الَّذِي جَعَلَ لَكُمُ النُّجُومَ لِتَهْتَدُوا بِهَا فِي ظُلُمَاتِ الْبَرِّ وَالْبَحْرِ ۗ قَدْ فَصَّلْنَا الْآيَاتِ لِقَوْمٍ يَعْلَمُونَ ﴿٦﴾

“Dan dialah yang menjadikan bintang-bintang bagimu, agar kamu menjadikannya petunjuk dalam kegelapan di darat dan di laut.

Sesungguhnya kami Telah menjelaskan tanda-tanda kebesaran (kami) kepada orang-orang yang Mengetahui.”(QS. Al An'am, 6: 97)

Allah SWT telah menciptakan bintang-bintang untuk keperluan manusia yang dengan itu bisa mengambil manfaat dari cahayanya, kemunculannya, dan posisinya di kegelapan malam ketika manusia berada di tengah padang pasir dan lautan sehingga bisa menemukan jalan dalam pengembaraan. Sebagian berada di depan dan sebagian lagi berada di belakang, di kiri dan di kanan. Bintang-bintang itu menjadi petunjuk bagi orang-orang yang bepergian menuju kota-kota, mencari arah Ka`bah, mengenali waktu di malam hari, dan menemukan arah jalan di padang pasir dan lautan.

Kalimat “*agar kamu menjadikannya petunjuk dalam kegelapan*” bukan berarti bintang-bintang diciptakan hanya untuk memberi petunjuk kepada manusia, tetapi Allah telah menciptakan bintang-bintang untuk banyak manfaat lainnya. Apabila seseorang merenung dan memperhatikan bintang-bintang yang kecil dan besar, dimana posisi mereka, orbit, hubungan dan gerakan mereka, dan juga memperhatikan manfaat matahari dan bulan serta bagaimana mereka bermanfaat dalam kehidupan, pertumbuhan, perkembangan hewan, tumbuhan, pernyataan suci ini akan terbukti baginya. Dalam kalimat “*Sesungguhnya kami Telah menjelaskan tanda-tanda kebesaran (kami) kepada orang-orang yang Mengetahui*” mewakili makna bintang. Artinya, Allah SWT telah menunjukkan dalam ayat-ayatNya dan menjelaskan kepada manusia yang memperoleh (ilmu) pengetahuan melalui cara perenungan.

Seiring dengan berjalannya waktu dan kemajuan dalam bidang teknologi navigasi, manusia telah menciptakan bererapa alat navigasi sehingga mereka

tidak lagi bergantung kepada posisi bintang-bintang untuk menentukan arah, diantara adalah peta, kompas, dan GPS. Akan tetapi bukan berarti bintang-bintang tersebut sudah tidak berguna, sebab masih banyak para ilmuwan yang mencoba mempelajari guna mengetahui manfaat lain dari bintang-bintang tersebut. Sehingga dari pengembangan ilmu pengetahuan tersebut muncullah suatu cabang keilmuan yang dikenal dengan sistem informasi geografis (SIG).

SIG itu unsur utamanya adalah geografi yakni penggambaran permukaan bumi, baik berupa gunung-gunung, jalan-jalan, batu-batuan, yang lengkap dengan warna-warna alamnya, hijau, putih, coklat, dan lain-lain. Al-Qur'an juga mendeskripsikan keadaan alam itu didalam Surat Fathir [35: 27]:

وَمِنَ الْجِبَالِ جُدَدٌ بَيْضٌ وَحُمْرٌ مُخْتَلِفٌ أَلْوَانُهَا وَعَرَايِبٌ سُودٌ ﴿٢٧﴾

“Dan di antara gunung-gunung itu ada garis-garis putih dan merah yang beraneka macam warnanya dan ada (pula) yang hitam pekat”.

Tafsir Jalalain menjelaskan (Dan di antara gunung-gunung itu ada garis-garis) *Judadun* adalah bentuk jamak dari lafal *Juddatun*, artinya jalan yang terdapat di gunung dan lainnya (putih, merah) dan kuning (yang beraneka macam warnanya) ada yang tua dan ada yang muda (dan ada -pula yang hitam pekat) *di'athafkan* kepada lafal *Judadun*, artinya ialah batu-batu yang besar yang hitam pekat warnanya. Dikatakan *Aswadu Gharbiibu*, hitam pekat; tetapi sangat sedikit dikatakan *Gharabiibu Aswadu*.

Jadi isyarat tentang keberadaan ilmu geologi dan geografi sudah ada dalam ajaran agama islam, seperti yang telah dijelaskan di atas. Al-Qura'an memberikan gambaran tentang lapisan bumi, warna-warna alam, dan

penggambaran geografis. Dengan perkembangan zaman, dengan sejalan dengan ditemukannya komputer, disiplin ilmu tentang Sistem Informasi Geografis mulai berkembang.

SIG mulai dikenal pada awal 1980-an. Seiring dengan berkembangnya perangkat lunak maupun perangkat keras komputer, SIG berkembang sangat pesat pada era 1990-an. Sistem informasi geografis sebenarnya terdiri dari tiga suku kata, yakni Sistem, Informasi, dan Geografis, maka dari itu pengertian dari tiga unsur kata ini akan sangat membantu dalam memberi gambaran tentang pengertian sistem informasi geografis. Sebenarnya sistem informasi geografis dengan sistem informasi lainnya sama saja dalam alur logikanya, hanya saja sistem ini ada tambahan unsur geografisnya, atau lebih terangnya sistem informasi geografis lebih menekankan pada informasi geografisnya.

Geografis, merupakan bagian dari keruangan (*spasial*). Dan kedua istilah ini sering terjadi pertukaran dalam penggunaannya, sehingga munculah istilah yang baru yakni geospasial. Dan dalam SIG ketiga istilah ini mengandung pengertian yang sama yaitu sama-sama mengandung pengertian suatu persoalan mengenai bumi; permukaan dua atau tiga dimensi.

Kemudian istilah informasi geografis mengandung pengertian informasi mengenai tempat-tempat yang terletak di permukaan bumi. Mengetahui dimana sebuah obyek terletak di permukaan bumi, dan sebuah informasi yang mengikutinya, keterangan-keterangan (atribut) yang ada di permukaan bumi yang menempel pada posisi geografisnya.

Dengan begitu, pengertian sistem informasi geografis dengan merujuk pada pengertian sistem informasi, maka SIG merupakan suatu kesatuan formal yang terdiri dari sumberdaya fisik dan logika yang berkenaan dengan obyek-obyek yang terdapat di permukaan bumi. Jadi, SIG juga merupakan sejenis perangkat lunak yang dapat digunakan untuk pemasukan, penyimpanan, manipulasi, menampilkan, dan keluaran informasi geografis berikut atribut-atributnya.

2.6.1 Pengertian Sistem Informasi Geografis

Banyak pengertian tentang sistem informasi geografis, karena SIG sendiri terus berkembang. Selain itu SIG juga merupakan kajian ilmu yang relatif baru dibandingkan dengan kajian ilmu yang lain. Berikut beberapa definisi sistem informasi geografis, seperti yang dijelaskan oleh Prahasta [2002:54]:

- a. SIG adalah sistem komputer yang digunakan untuk memasukan (*capturing*), menyimpan, memeriksa, mengintegrasikan, memanipulasi, menganalisa, dan menampilkan data-data yang berhubungan dengan posisi-posisi di permukaan bumi (Rice20).
- b. SIG adalah kombinasi perangkat keras dan perangkat lunak komputer yang memungkinkan untuk mengelola (*manage*), menganalisa, memetakan, informasi spasial berikut atribut-atributnya (data deskriptif) dengan akurasi kartografi (basic20).
- c. SIG adalah sistem yang berbasis komputer yang digunakan untuk menyimpan dan memanipulasi informasi-informasi geografi. SIG dirancang untuk mengumpulkan, menyimpan dan menganalisis obyek-obyek dan fenomena dimana lokasi geografi merupakan karakteristik yang penting atau kritis untuk dianalisis. Dengan demikian, SIG merupakan sistem komputer yang memiliki empat kemampuan berikut dalam menangani data yang bereferensi geografi: (a) Masukan, (b) Manajemen data (penyimpanan dan pemanggilan data), (c) Analisis dan manipulasi data, (d) Keluaran. (Aronoft89).

- d. SIG adalah sistem yang terdiri dari perangkat keras, perangkat lunak, data, manusia (*brainware*), organisasi dan lembaga yang digunakan untuk mengumpulkan, menyimpan, menganalisa, dan menyebarkan informasi-informasi mengenai daerah-daerah di permukaan bumi (Chrisman97).
- e. SIG adalah sistem komputer yang digunakan untuk memanipulasi data geografi. Sistem ini diimplementasikan dengan perangkat keras dan perangkat lunak komputer yang berfungsi untuk: (a) Akusisi dan verifikasi data, (b) Kompilasi data, (c) Penyimpanan data, (d) Perubahan dan updating data, (e) Manajemen dan pertukaran data, (f) Manipulasi data, (g) Pemanggilan dan presentasi data, dan (h) Analisis data (Bern92).

2.6.2 Subsistem Sistem Informasi Geografis

Dari definisi di atas bisa ditarik kesimpulan tentang beberapa subsistem sistem informasi geografis:

1. **Data Input:** Subsistem ini bertugas untuk mengumpulkan dan mempersiapkan data spasial dan atribut dari berbagai sumber. Subsistem ini pula yang mempunyai tanggung jawab dalam mengkonversi atau menstransformasikan format-format data-data aslinya kedalam format yang dapat digunakan oleh SIG. Ada dua macam data dasar geografi, yaitu data spasial dan data atribut:
 - a. Data spasial (keruangan), yaitu data yang menunjukkan ruang, lokasi atau tempat-tempat di permukaan bumi. Data spasial berasal dari peta analog, foto udara dan penginderaan jauh dalam bentuk cetak kertas.
 - b. Data atribut (deskriptis), yaitu data yang terdapat pada ruang atau tempat. Atribut menjelaskan suatu informasi. Data atribut diperoleh dari statistik, sensus, catatan lapangan dan tabular (data yang disimpan dalam bentuk tabel) dan lainnya. Data atribut dapat dilihat dari segi kualitas, misalnya kekuatan pohon. Dan dapat dilihat dari segi kuantitas, misalnya jumlah pohon.

2. Data Output: Subsistem ini menampilkan atau menghasilkan keluaran seluruh atau sebagian basisdata baik dalam bentuk softcopy maupun bentuk hardcopy seperti: tabel, grafik, peta, dan lain-lain.
3. Data Management: Subsistem ini mengorganisir baik data spasial maupun data atribut kedalam sebuah basisdata sedemikian rupa sehingga mudah dipanggil, di-update, dan di-edit.
4. Data Manipulation dan Analysis: subsistem ini menentukan informasi-informasi yang dapat dihasilkan oleh SIG. Selain itu, subsistem ini juga melakukan manipulasi dan pemodelan data untuk menghasilkan informasi yang diharapkan. (Prahasta, 2002:56)

2.7 Arc View GIS

Arc View adalah salah satu perangkat lunak GIS yang paling populer dan paling banyak digunakan untuk mengelola data spasial dewasa ini. Software ini dibuat oleh ESRI (*Environmental Systems Research Institute*), perusahaan yang mengembangkan program Arc/Info. Dengan ArcView kita dengan mudah dapat melakukan input data, menampilkan data, mengelola data, menganalisis data, dan membuat peta serta laporan yang berkaitan dengan data spasial bereferensi geografis.

Dari struktur data yang telah dibahas di atas, ArcView lebih memfokuskan perhatian pada struktur data vector. Namun demikian, ArcView juga mempunyai kemampuan untuk menganalisis data berbasis raster (grid dan citra penginderaan jauh).

2.7.1 Project Arcview

Data dalam ArcView diorganisasi dalam suatu proyek. Setiap bekerja dengan ArcView, maka bekerja pada suatu project ArcView. Satu project kosong ditampilkan dengan nama Untitled. Kemudian dapat digunakan project kosong ini atau bisa memakai project yang pernah digunakan dan sudah disimpan, melalui *File-Open Project*. Setiap project memiliki lima komponen, yaitu Views, Tables, Charts, Layouts, dan Scripts. Views digunakan untuk mengelola data grafis, Tables dipakai untuk manajemen data atribut, Charts digunakan untuk mengelola grafik (bukan data grafis), Layouts dipakai untuk membuat komposisi peta untuk dicetak, dan Scripts digunakan untuk membuat modul yang berisikan kumpulan perintah ArcView yang ditulis dalam bahasa pemrograman Avenue.

Project ArcView tidak menyimpan data sebenarnya, seperti data grafis, atribut atau yang lainnya. Project Arcview hanyalah menyimpan referensi lokasi dan bagaimana data ditampilkan. Ini berarti kita dapat menggunakan data ArcView dalam beberapa Project ArcView tanpa adanya duplikasi. File Project ArcView adalah file teks yang dapat dibuka pada sembarang software teks editor atau word processing.

2.7.2 Views

Views pada project ArcView dipakai untuk mengelola data grafis, baik vector maupun grid atau raster. Seluruh pekerjaan yang berkaitan dengan manajemen data grafis dapat dilakukan pada View, mulai dari input data, manipulasi tampilan data grafis, sampai analisis data.

2.7.3 Tables

Tables (tabel) pada project ArcView digunakan untuk manajemen data atribut atau tabular. Membuat tabel baru, menambah field dan record, join antartabel merupakan beberapa pekerjaan yang dapat dilakukan pada *Tables*. Pada struktur data vector, data yang tersimpan dalam tabel saling terkoneksi dengan data grafis pada view. Perubahan data pada tabel akan menyebabkan perubahan data grafis pada View, dan sebaliknya.

2.7.4 Charts

Charts pada project ArcView digunakan untuk mengelola grafik. Data yang digunakan dalam grafik bersumber dari data tabel. Beberapa jenis grafik yang dapat digunakan atau ada pada fasilitas Arcview ini diantaranya grafik batang, kolom, garis, area, lingkaran, dan scatter XY.

2.7.5 Layouts

Fasilitas Layout digunakan untuk membuat layout atau komposisi peta sebelum peta dicetak. Fasilitas ini meliputi penggunaan view untuk layout, pembuatan legenda, skala, arah utara, judul peta, dan asesoris lainnya.

2.7.6 Scripts

Scripts pada project ArcView merupakan kumpulan dari perintah ArcView yang ditulis dalam bahasa Avenue. Bagi pengguna ArcView tingkat advance, penggunaan fasilitas ini akan sangat membantu dalam analisis data, karena kita dapat melakukan *customize* dan kreasi sendiri analisis sesuai dengan kebutuhan.

Disamping itu, proses yang sifatnya berulang-ulang dapat dibuatkan Scripts sehingga pekerjaan menjadi lebih efektif dan efisien.

Untuk lebih jelasnya lagi tentang kemampuan perangkat SIG ArcView ini secara umum dapat dijabarkan sebagai berikut:

1. Pertukaran data: Membaca dan menuliskan data dari dan ke dalam perangkat lunak SIG lainnya:
 - a. ArcView dapat membaca data spasial raster yang ditulis dalam format-format perangkat lunak SIG dan pengidraan jauh: misalnya *JPEG*, *BMP*, *TIFF*, *GeoTIFF*, *BSQ*, *BIL*, *BIP*, *ERDAS (LAN & GIS)*, *ERDAS Image*, *GRID ArcInfo(data grid-cell)*, *raster SUN*, dan sebagainya.
 - b. ArcView dapat membaca data spasial vektor yang dituliskan dalam format-format perangkat lunak SIG lainnya (import): misalnya *ArcInfo (coverage)*, *MapInfo (MIF)*, *AutoCad (DWG dan DXF)*, dan lain sebagainya.
 - c. ArcView dapat menuliskan basisdata spasial vektor (*coverage* dan *shape files*) baik ke dalam format *shape file* sendiri maupun ke dalam perangkat lunak SIG lainnya, misalnya MapInfo.
2. Melakukan analisis statistik dan operasi-operasi matematis.
3. Menampilkan informasi (basisdata) spasial maupun atribut:
 - a. ArcView dapat menampilkan informasi (basisdata dengan format sendiri) baik yang terdapat pada sistem komputer yang bersangkutan maupun yang tersebar di jaringan komputer (network).
 - b. ArcView dapat mengakses dan menampilkan basisdata eksternal

- c. Menampilkan informasi atau data dalam bentuk *View* (tampilan untuk dilayar monitor). *Layout* (tata letak peta siap cetak). *Table* (table data). *Chart* (grafik).
4. Menjawab *query* spasial maupun atribut:
 - a. menghubungkan informasi spasial dengan atribut-atribut yang terdapat (disimpan) di dalam basisdata atribut: (1). Memilih feature(entitas) spasial, muncul informasi spasialnya. (2). Memilih data atribut, muncul representasi spasial dari feature yang dipilih. (3). Memilih data atribut, muncul data-data atribut lainnya yang terdapat di dalam basisdata atribut. (4). Memilih suatu feature spasial, muncul feature spasial lainnya yang terkait.
 - b. Menggunakan SQL sebagai standar untuk melakukan query terhadap basis datanya.
 5. Melakukan fungsi-fungsi dasar SIG.
 6. Membuat peta tematik: menyediakan pustaka simbol dan warna (*feature*) untuk membuat peta tematik.

Menggunakan simbol dan warna untuk merepresentasikan *feature*-nya berdasarkan atribut-atributnya (membuat peta tematik turunan). Misalnya peta suatu wilayah administrasi (contohnya kecamatan) dapat diberi arsiran yang rapat dan warna yang agak gelap untuk mempresentasikan populasi penduduk yang padat. Sementara pada peta tematik lainnya, untuk wilayah administrasi yang sama, dapat diberi (pola)

arsiran yang jarang dan warna agak muda untuk merepresentasikan atribut pendapatan perkapita yang berada di bawah rata-rata.

7. Memodifikasi aplikasi dengan menggunakan bahasa skrip.

Menyediakan bahasa pemrograman sederhana atau skrip (Avenue) untuk mengotomasi pengoperasian rutin dan memodifikasi aplikasi-aplikasi SIG yang dikembangkan dengan menggunakan perangkat lunak ArcView.

8. Melakukan fungsi-fungsi SIG khusus lainnya (dengan menggunakan extension yang dapat ditujukan untuk mendukung penggunaan perangkat lunak SIG ArcView).

- a. *Network analyst*: Modul perangkat lunak yang digunakan untuk melakukan analisis-analisis yang berhubungan dengan jaringan (routing).
- b. *Internet map server (IMS)*: Modul perangkat lunak yang digunakan untuk mempublikasikan peta (basisdata digital) ke jaringan internet (*web-base GIS*) hingga dapat diakses dengan menggunakan program browser
- c. *3D analyst*: Modul perangkat lunak yang digunakan untuk membuat, menganalisa, dan menampilkan data spasial tiga dimensi (permukaan digital).
- d. *Tracking analyst*: Modul perangkat lunak yang digunakan untuk melakukan perekaman tampilan, dan pemantauan data, baik secara langsung (*real time*) maupun tidak langsung (*playback*, atau *replay*)

yang memiliki kecenderungan perubahan posisi geografis dari waktu ke waktu. Contoh yang umum adalah pengamatan terhadap suatu obyek (misalnya kendaraan) bergerak yang dipasang GPS. Dengan alat ini, posisi-posisi obyek yang bergerak dapat direpresentasikan dengan menggunakan simbol dan warna tertentu di atas peta yang menjadi latar belakangnya.

- e. *Image analyst*: modul perangkat lunak yang digunakan untuk melakukan analisis-analisis yang berhubungan dengan citra digital (pengindraan jauh).

ArcView dalam operasinya menggunakan, membaca dan mengolah data dalam format Shapefile, selain itu ArcView juga dapat memanggil data-data dengan format BSQ, BIL, BIP, JPEG, TIFF, BMP, GeoTIFF atau data grid yang berasal dari ARC/INFO serta banyak lagi data-data lainnya. Setiap data spasial yang dipanggil akan tampak sebagai sebuah Theme dan gabungan dari theme-theme ini akan tampil dalam sebuah view. ArcView mengorganisasikan komponen-komponen programnya (view, theme, table, chart, layout dan script) dalam sebuah project. Project merupakan suatu unit organisasi tertinggi di dalam ArcView.

Salah satu kelebihan dari ArcView adalah kemampuannya berhubungan dan bekerja dengan bantuan extensions. Extensions (dalam konteks perangkat lunak SIG ArcView) merupakan suatu perangkat lunak yang bersifat “plug-in” dan dapat diaktifkan ketika penggunaannya memerlukan kemampuan fungsionalitas tambahan. Extensions bekerja atau berperan sebagai perangkat lunak yang dapat

dibuat sendiri, telah ada atau dimasukkan (di-instal) ke dalam perangkat lunak ArcView untuk memperluas kemampuan-kemampuan kerja dari ArcView itu sendiri. Contoh-contoh extensions ini seperti *Spasial Analyst, Edit Tools v3.1, Geoprocessing, JPGE (JFIF) Image Support, Legend Tool, Projection Utility Wizard, Register and Transform Tool dan XTools Extensions*.

Shapefile ESRI terdiri dari beberapa file: file utama, file indeks, dan sebuah table Dbase. File utama merupakan *direct-access*, file dengan panjang record yang bervariasi dimana setiap record-nya mendeskripsikan sebuah shape(feature) dengan sebuah list (daftar) verteks-verteksnya. Pada file indeks, setiap record mengandung offset record file utama yang bersesuaian dari awal ke file utama. Table Dbase berisi atribut-atribut feature, satu record per feature. Relasi one to one antara feature (geometri) dengan atributnya didasarkan pada nomor recordnya. Record atribut, urutannya, harus sama sebagaimana di dalam file utama.

Sesuai dengan konvensi penamaannya, file utama, file indeks, dan file Dbase memiliki nama depan (suffix) yang sama, tetapi nama-nama belakangnya (suffix atau extension) berbeda. Nama-nama belakangnya berturut-turut adalah “SHP” (file utama), “SHX” (file indeks), dan “DBF” (file table atribut).

Shapefile ESRI dapat dibuat atau dihasilkan dengan menggunakan empat cara berikut:

- a. Ekspor: format data spasial ini dapat dihasilkan dari proses ekspor perangkat lunak SIG lainnya, misalnya dengan menggunakan ArcInfo, SDE13, MapInfo.

- b. Dijitasi : shapefile dapat secara langsung dibuat melalui proses dijitasi
- c. Semi pemrograman: shapefile dapat secara langsung dibuat, dibaca, atau dituliskan dengan menggunakan salah satu bahasa (semi) pemrograman skrip/makro yang dimiliki oleh beberapa perangkat SIG serumpun (misalnya Avenue-nya ArcView, MapObjects, ARC Macro Language/AML ArcInfo, Simple macro Language/SML PC ArcInfo)
- d. Bahasa pemrograman: dengan memahami spesifikasi teknisnya, shapefile dapat secara langsung dibuat, dibaca, atau dituliskan dengan menggunakan salah satu bahasa pemrograman yang ada (misalnya C/C++, pascal/Delphi, Basic).

Keuntungan-keuntungan jika bekerja dengan menggunakan data spasial shapefile ArcView adalah sebagai berikut:

- a. Proses pengambilan (draw) atau penggambaran kembali (redraw) dari *feature* petanya dapat dilakukan dengan relative cepat-setidaknya lebih cepat dari proses penggambaran coverage milik ArcInfo
- b. Informasi atribut dan geometrinya dapat di edit
- c. Dapat dikonversikan ke dalam format-format data spasial lainnya
- d. Memungkinkan untuk proses on-screen digitizing.

(Nuarsa, 2005:17)

2.8 Web Sistem Informasi Geografis

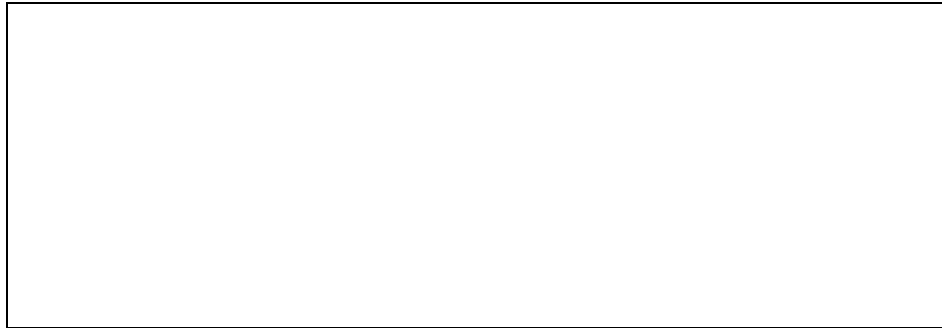
Sistem *Informasi Geografis* merupakan sistem yang dirancang untuk bekerja dengan data yang terreferensi secara spasial atau koordinat-koordinat geografi. SIG memiliki kemampuan untuk melakukan pengolahan data dan

melakukan operasi-operasi tertentu dengan menampilkan dan menganalisa data. Aplikasi SIG saat ini tumbuh tidak hanya secara jumlah aplikasi namun juga bertambah dari jenis keragaman aplikasinya. Pengembangan aplikasi SIG kedepannya mengarah kepada aplikasi berbasis Web yang dikenal dengan *Web GIS*. Hal ini disebabkan karena pengembangan aplikasi di lingkungan jaringan telah menunjukkan potensi yang besar dalam kaitannya dengan *geo informasi*. Sebagai contoh adalah adanya peta online sebuah kota dimana pengguna dapat dengan mudah mencari lokasi yang diinginkan secara online melalui jaringan intranet/internet tanpa mengenal batas geografi penggunanya. Secara umum Sistem Informasi Geografis dikembangkan berdasarkan pada prinsip *input*/masukan data, manajemen, analisis dan representasi data. Di lingkungan web prinsip-prinsip tersebut di gambarkan dan diimplementasikan seperti pada table berikut:

Table 2.1 Prinsi-prinsip Pengembangan SIG

GIS Prinsip	Pengembangan Web
Data Input	Client
Manajemen Data	DBMS dengan komponen spasial
Analisis Data	GIS Library di Server
Representasi Data	Client/server

Untuk dapat melakukan komunikasi dengan komponen yang berbeda-beda di lingkungan web maka dibutuhkan sebuah *web server*. Karena standar dari geo data berbeda-beda dan sangat spesifik maka pengembangan arsitektur sistem mengikuti arsitektur '*Client Server*'.

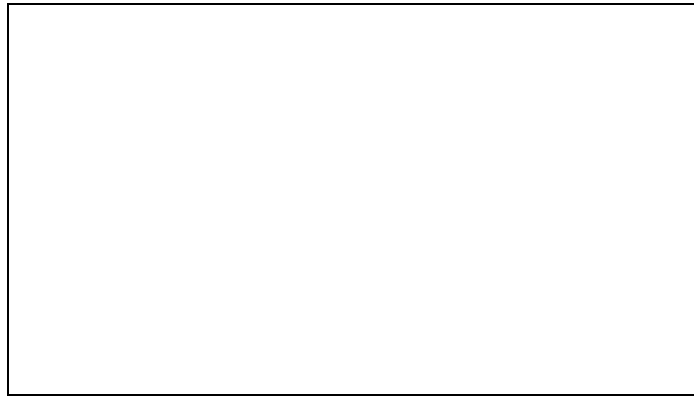


(Sumber: Denny Charter, Ilmu Komputer.com)

Gambar 2.1 Arsitektur WEB SIG

Gambar di atas menunjukkan arsitektur minimum sebuah sistem Web-GIS. Aplikasi berada di sisi client yang berkomunikasi dengan Server sebagai penyedia data melalui web Protokol seperti HTTP (*Hyper Text Transfer Protocol*). Aplikasi seperti ini bisa dikembangkan dengan *web browser* (Mozilla Firefox, Opera, Internet Explorer, dll). Untuk menampilkan dan berinteraksi dengan data GIS, sebuah browser membutuhkan *Pug-In* atau *Java Applet* atau bahkan keduanya. Web Server bertanggung jawab terhadap proses permintaan dari client dan mengirimkan tanggapan terhadap respon tersebut. Dalam arsitektur web, sebuah web server juga mengatur komunikasi dengan server side GIS Komponen. Server side GIS Komponen bertanggung jawab terhadap koneksi kepada database spasial seperti menterjemahkan *query* kedalam SQL dan membuat representasi yang diteruskan ke server. Dalam kenyataannya Side Server GIS Komponen berupa software libraries yang menawarkan layanan khusus untuk analisis spasial pada data. Selain komponen hal lain yang juga sangat penting adalah aspek fungsional yang terletak di sisi client atau di server. Gambar berikut dua

pendekatan yang menunjukkan kemungkinan distribusi fungsional pada system client/server berdasarkan konsep *pipeline visualization*.



(Sumber: Denny Charter, Ilmu Komputer.com)

Gambar 2.2 Perbandingan *Thin* dan *Thick* Sistem Pada Sistem *Client Server*

Pendekatan-1: Thin Client : Memfokuskan diri pada sisi server. Hampir semua proses dan analisa data dilakukan berdasarkan request disisi server. Data hasil pemrosesan dikirimkan ke client dalam format HTML, yang didalamnya terdapat file gambar sehingga dapat dilihat dengan browser. Pada pendekatan ini interaksi pengguna terbatas dan tidak fleksibel.

Pendekatan-2: Thick/Fat Client : Pemrosesan data dilakukan disisi client, data dikirim dari server ke client dalam bentuk data vector yang disederhanakan. Pemrosesan dan penggambaran kembali dilakukan disisi client. Cara ini menjadikan user dapat berinteraksi lebih interaktif dan fleksibel.

Sementara internet merupakan kumpulan jaringan komputer yang tersebar secara geografis dan terhubung dalam satu alat komunikasi dan protokol yang sama sehingga memungkinkan untuk saling berhubungan. Web atau lebih dikenal dengan world wide web (www) merupakan aplikasi jaringan yang mendukung

terlaksananya HTTP (hypertext transfer protocol) dalam suatu jaringan internet. Sehingga dapat disimpulkan bahwa internet tidak sama dengan web. Internet merujuk kepada infrastruktur jaringan, sedangkan web merujuk kepada salah satu aplikasi yang berjalan dalam jaringan internet. Oleh karena itu, Internet GIS tidak sama dengan web GIS (Peng, Z. and M. Tsou, 2003). Internet GIS merujuk kepada penggunaan internet untuk melakukan pertukaran data, melakukan analisis spasial, dan menyajikan hasil analisis. Sedangkan web GIS menggunakan aplikasi web dalam melakukan operasi GIS.

2.9 Mapserver

Mapserver merupakan aplikasi *freeware* dan *open source* yang memungkinkan kita menampilkan data spasial (peta) pada halaman web. Aplikasi ini pertama kali dikembangkan di Universitas Minesotta, Amerika Serikat untuk proyek *ForNet* (sebuah project untuk manajemen sumber daya alam) yang didukung oleh NASA dilanjutkan dengan dikembangkan project *TerraSIP* untuk manajemen data lahan.

Pengembangan MapServer menggunakan berbagai aplikasi open source atau freeware seperti Shapelib yang digunakan untuk baca tulis format data Shapefile, FreeType untuk merender karakter, GDAL/OGR untuk baca atau tulis format data vektor maupun raster, dan Proj.4 untuk mengganti beragam proyeksi peta.

Pada bentuk paling dasar MapSever merupakan sebuah program CGI(Common Gateway Interface). Program tersebut akan dieksekusi di web server, dan berdasarkan parameter tertentu terutama konfigurasi dalam bentuk file

*.MAP akan menghasilkan data yang kemudian akan dikirim ke web browser, baik dalam bentuk peta atau dalam bentuk lain.

Fitur-fitur dari MapServer diantaranya adalah:

- a. Menampilkan data spasial dalam format vektor seperti Shapefile(ESRI), ArcSDE(ESRI), PostGIS dan berbagai format fektor lain dengan menggunakan library OGR.
- b. Menampilkan data spasial dalam bentuk format raster seperti TIFF/GeoTIFF, APPL7 dan berbagai format data raster lain dengan menggunakan library GDAL
- c. Menggunakan *quadtree* dalam *indexing* data spasial, sehingga operasi-operasi spasial dapat dilakukan dengan cepat.
- d. Dapat dikembangkan dengan tampilan keluaran yang dapat diatur dengan menggunakan file tamplate.
- e. Dapat melakukan seleksi obyek berdasarkan nilai, titik, area, atau berdasarkan sebuah obyek spasial tertentu.
- f. Mendukung rendering karakter berupa font *TrueType*.
- g. Mendukung penggunaan data raster maupun vektor yang dibagi-bagi menjadi sub bagian yang lebih kecil sehingga proses untuk mengambil dan menampilkan gambar dapat dipercepat.
- h. Dapat menggambarkan peta tematik yang dibangun menggunakan ekspresi logik atau ekspresi reguler.
- i. Dapat menampilkan label dari obyek spasial, dengan label dapat diatur sedemikian rupa sehingga tidak salin tumpang tindih.

- j. Konfigurasi dapat diatur secara *on the fly* melalui parameter yang ditentukan pada URL.
- k. Dapat menangani beragam sistem proyeksi secara *on the fly*.

Saat ini, selain dapat mengakses MapServer sebagai program CGI, dapat di akses MapServer sebagai modul MapScript, melalui berbagai bahasa skrip: *PHP*, Perl, Python atau Java. Sehingga akses fungsi-fungsi MapServer melalui skrip akan lebih memudahkan pengembangan aplikasi. Pengembang dapat memilih bahasa yang paling familiar.

2.9.1 Sejarah Perkembangan Mapserver

Sampai mencapai status seperti sekarang, MapServer berevolusi melalui berbagai tahapan. Di bawah ini tahun-tahun penting perkembangan MapServer:

Table 2.2 Perkembangan MapServer

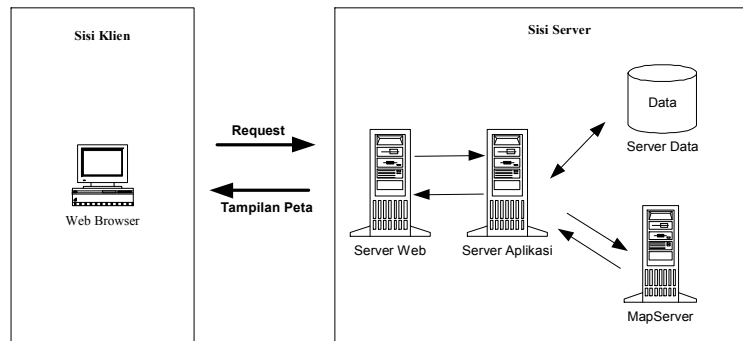
Tahun	Perkembangan
1994	Xerox Parc Map Viewer dibuat. Aplikasi ini merupakan aplikasi web pertama yang menampilkan data spasial secara interaktif. Kode program aplikasi ini sendiri bersifat tertutup, tetapi keberadaannya memunculkan ide untuk pengembangan MapServer selanjutnya.
1995	BW CAW Interactive Mapping dibuat. Aplikasi ini digunakan untuk sistem pengambilan keputusan berbasis GIS, dengan komponen pemetaan di web.
1996	Project ForNet dilaksanakan. Project ini dilaksanakan oleh Universitas Minesotta dan Minesotta DNR, atas dana dari NASA.
1997	MapServer versi 1.0 dirilis. Ditulis dalam bahas C, dan merupakan produk dari project ForNet di tahun sebelumnya. Versi ini menggunakan <i>library</i> GD (http://www.boutell.com/gd) untuk penggambaran peta, dan <i>Shapelib</i> untuk akses data spasial dalam format <i>Shapefile</i> .

Tahun	Perkembangan
1998	MapServer versi 2.0 dirilis. Versi ini merupakan versi pertama yang dishare di luar project <i>ForNet</i> , tepatnya <i>Australia's Environmental Resource Information Network (ERIN)</i> . Pada versi ini ditambahkan dukungan sistem proyeksi dengan menggunakan <i>library Proj.4</i> yang dikembangkan USGS (<i>United States Geological Survey</i>).
2000	Project TerraSIP dilaksanakan, sekaligus MapServer versi 3.0 dirilis. Versi ini merupakan versi pertama yang dirilis ke publik sebagai program <i>open source</i> . Versi ini sudah mendukung data raster dalam format TIFF menggunakan <i>library libTIFF</i> , dan karakter <i>TrueType</i> sudah didukung menggunakan <i>library FreeType</i> .
2001	Mapscript versi 1.0 dirilis, bersamaan dengan rilis MapServer versi 3.2. Aplikasi tersebut dibangun menggunakan <i>tool SWIG (Simplified Wrapper and Interface Generator, http://www.swig.org)</i> . Beberapa pengembang dari Kanada mengembangkan versi pertama dengan bahasa pemrograman skrip <i>PHP (http://www.php.net)</i> .
2002	MapServer versi 3.5 dirilis, dengan dukungan terhadap data spasial berbasis DBMS, baik PostgreSQL PostGIS maupun SDE (<i>Spatial Database Engine</i>) dari ESRI. MapServer versi 3.6 juga dirilis tahun ini dengan dukungan terhadap spesifikasi WMS (<i>Web Mapping Services</i>) yang dikeluarkan oleh OGC (<i>Open Geospatial Consortium</i>).
2003	<i>MapServer User Meeting (MUM)</i> yang pertama dilaksanakan. MapServer versi 4.0 dirilis. Versi ini mendukung keluaran dalam 24 bit warna (<i>true color</i>), juga memungkinkan keluaran dalam format <i>PDF</i> dan <i>SWF</i> .
2004	MapServer versi 4.4 dirilis. Versi ini mengikuti spesifikasi WMS (<i>Web Mapping Services</i>) yang dikeluarkan oleh OGC (<i>Open Geospatial Consortium</i>).

(Nuryadin, 2005:5)

2.9.2 Arsitektur Mapserver

Bentuk umum arsitektur aplikasi berbasis peta dalam halaman web dapat jelaskan sesuai dengan gambar 2.3.



(Sumber: *Yomi Agung Susanto*)
 Gambar 2.3 Arsitektur Peta Dalam Halaman Web

Pada gambar 2.3, interaksi antara klien dengan server berdasar skenario request dan respon. Web browser di sisi klien mengirim request ke server web. Karena server web tidak memiliki kemampuan pemrosesan peta, maka request yang berkaitan dengan pemrosesan peta akan diteruskan oleh server web ke server aplikasi dan MapServer. Hasil pemrosesan akan dikembalikan lagi melalui server web, yang terbungkus dalam bentuk file HTML atau applet. Arsitektur aplikasi pemetaan di web dibagi menjadi dua pendekatan, di antaranya sebagai berikut:

a. Pendekatan Thin Client

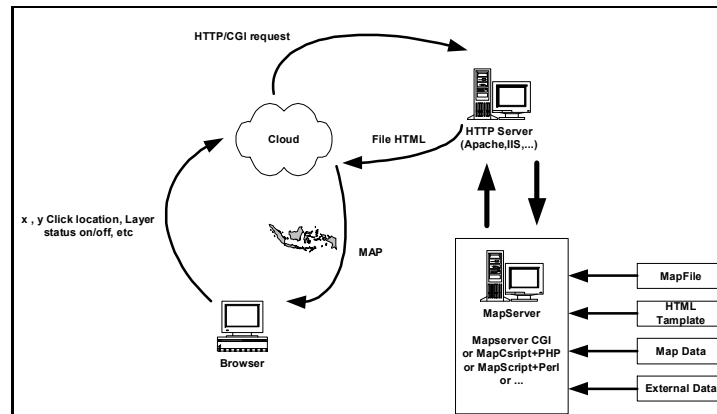
Pendekatan ini memfokuskan diri pada sisi server. Hampir semua proses dan analisi data dilakukan berdasarkan request di sisi server. Data hasil pemrosesan kemudian dikirim ke klien dalam format standard HTML, yang didalamnya terdapat file gambar dalam format standard misalnya GIF, PNG, JPG, sehingga dapat dilihat dengan menggunakan sembarang web browser. Kelemahan utama pendekatan ini menyangkut keterbatasan opsi interaksi dengan user yang kurang fleksibel.

b. Pendekatan Thick Client

Pada pendekatan ini, pemrosesan data dilakukan di sisi klien menggunakan beberapa kontrol seperti kontrol ActiveX atau applet. Kontrol ActiveX atau applet akan dijalankan di klien untuk memungkinkan web browser menangani format data yang tidak dapat ditangani oleh web browser dengan kemampuan standart. Dengan adanya pemrosesan di klien, maka transfer data antara klien dengan web server akan berkurang. Tidak seperti pada pendekatan thin client, data dikirim ke klien dalam bentuk data vektor yang disederhanakan. Pemrosesan dan penggambaran kembali akan dilakukan di sisi klien. Dengan cara demikian, pengembang aplikasi dapat lebih fleksibel dalam menentukan prosedur interaksi aplikasi dengan user. Kelemahan dari pendekatan ini, harus ada tambahan aplikasi yang dipasang di komputer klien (Kontrol ActiveX atau Applet).

Mapserver menggunakan pendekatan Thin Client. Sehingga pemrosesan dilakukan di sisi server. Informasi peta dikirim ke web browser di sisi klien dalam bentuk file gambar (JPG, PNG, GIF, atau TIFF). Pada saat ini kelemahan pendekatan thin client sudah dapat diatasi dengan adanya *Framework* aplikasi seperti *Chameleon CartoWeb*, *Kmap*.

Aplikasi yang dibangun dengan menggunakan MapServer memiliki arsitektur sebagai berikut:



Gambar 2.4 Arsitektur Mapserver

Pada sistem aplikasi, browser (client) mengirimkan request ke web server dalam bentuk request yang terkait dengan data spasial (lokasi [X,Y]) click kursor, status [on/off] layer yang akan dimunculkan, dan lain sebagainya. Kemudian oleh web server request tersebut dikirim ke server dan MapServer. Kemudian MapServer akan membaca mapfile, dan data peta. Untuk membentuk sebuah gambar yang sesuai dengan request dari client. Setelah gambar tersebut di render, file gambar yang bersangkutan akan dikirim ke web server kembali dan diteruskan ke browser client sesuai dengan format template tampilannya. Dilihat dari tempat pengolahan data arsitektur tersebut cenderung bersifat *thin-client*, sebab prosedur-prosedur yang terkait dengan pengolahan data dan analisis proses dilakukan di server, sementara browsernya hanya menerima hasil request dalam bentuk file HTML.

2.9.3 MapServer for Windows (MS4W)

MapServer for Windows (MS4W) adalah suatu perangkat lunak yang sangat memudahkan para pengguna di dalam meng-*install* (atau melakukan *set-*

up) Mapserver (UMN atau Cheetah) pada platform sistem operasi Ms. Windows. Tujuan utama pembuatan paket ini adalah untuk memudahkan semua (tingkatan) pengguna, secepatnya (terhindar dari segala detail yang rumit), di dalam mempersiapkan lingkungan kerja yang diperlukan oleh MapServer di lingkungan Ms. Windows. Selain itu, paket ini juga merupakan suatu cara atau lingkungan yang sangat baik untuk memaketkan dan kemudian mendistribusikan aplikasi-aplikasi MapServer kepada pihak manapun (Prahasta, 2007:41).

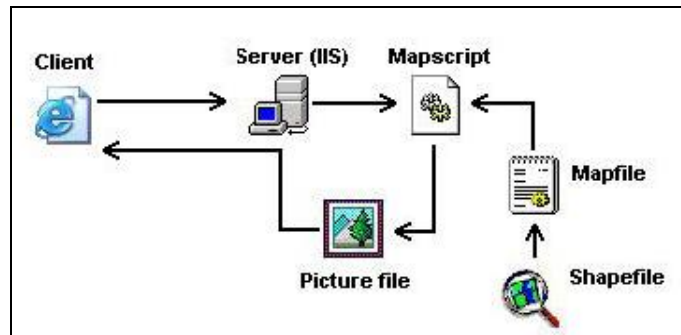
Dalam MS4W ada beberapa Isi paket, Paket dasar MS4W akan melakukan instalasi lingkungan *server web (pre-configured)* bersama dengan beberapa komponen seperti berikut:

1. Server HTTP Apache
2. PHP
3. MapServer CGI
4. PHP/Mapscript
5. Program Utility (pustaka) GDAL & OGR
6. Program utility MapServer

2.9.4 Mapscript Environment

Php/Mapscript, atau sering disebut Mapscript saja, adalah sebuah modul yang digunakan untuk membuat fungsi-fungsi dan class-class MapServer agar dapat dijalankan di *PHP3/PHP4*. Sedangkan MapServer sendiri adalah sebuah service untuk memproses dan menampilkan data spatial, lengkap dengan proyeksi-proyeksi dan koordinatnya. MapServer dikembangkan oleh universitas Minnesota (<http://mapserver.gis.umn.edu>) dan merupakan freeware sehingga

dapat dipakai oleh semua orang secara gratis. MapScript dapat berjalan di sistem operasi windows maupun linux. Dan dapat dijalankan pada beberapa Web Server. Secara garis besar, proses yang akan dilakukan oleh mapsript dapat dilihat pada gambar 2.6 dibawah.



(Sumber: Prasetyo, Daniel Hary. 2004)

Gambar 2.5 Proses Mapsript Pada Web Server

Proses diawali dengan request dari client ke server. Web Server akan menjalankan fungsi-fungsi yang ada di library Mapsript. Data spatial yang akan digunakan berupa data bertipe shapefile, yang merupakan file spatial standar dari ESRI. Dengan perantara MapFile, sebagai pengatur setting dari data yang akan ditampilkan, MapScript akan membaca data spatial di shapefile ini, memprosesnya sesuai request dari client, kemudian menyimpannya kedalam bentuk file gambar (GIF, JPG, atau PNG). File gambar ini kemudian akan di load ke client dalam bentuk object Image HTML. Karena bentuk peta yang ditampilkan merupakan file gambar maka kerja client tidak berat, terutama jika dibandingkan dengan cara lain yang menggunakan activeX.

MapScript digunakan sebagai interface untuk class-class yang terdapat di MapServer. Secara garis besar, peta tersusun dari beberapa layer. Layer tersusun

dari bentuk-bentuk yang polygon, garis, atau titik yang disebut dengan Shape. Class-class yang terdapat di MapServer melingkupi manipulasi untuk Peta, Layer, dan Shape ini. class-class didalam MapServer yang sering digunakan untuk mengembangkan Web GIS antara lain adalah :

a. MapObj Class

Class untuk berhubungan dengan MapFile, atau secara tidak langsung berhubungan dengan data peta. Class ini juga sebagai class utama di MapScript untuk memproses dan menyimpan data peta ke file gambar. Beberapa fungsi dan class penting yang terdapat pada class ini, antara lain:

1. Mempunyai kumpulan class layerObj yang menyusun peta dan fungsi untuk mengatur urutan peta.
2. Fungsi untuk menggambar peta, disimpan ke dalam class ImageObj.
3. Fungsi untuk menggambar legend, dan disimpan kedalam class ImageObj
4. Fungsi untuk menggambar scalebar, dan disimpan kedalam class ImageObj
5. Fungsi SetExtent untuk menentukan Extent dari Peta.
6. Fungsi ZoomPoint, ZoomRectangle, ZoomScale untuk melakukan pembesaran (zoom in) atau pengecilan (Zoom out peta).
7. Fungsi untuk mengeset proyeksi peta

8. Fungsi QueryByPoint, QueryByRect, QueryByShape, QueryByFeature untuk mencari object di peta yang ada di posisi tertentu dengan Rectangle, Shape, dan dengan peta lain.

b. LayerObj Class

Class yang mewakili satu layer yang menyusun peta. Fungsi-fungsi yang penting pada class ini antara lain:

1. Fungsi GetShape untuk mengambil sebuah shape yang ada pada layer. Disimpan dalam shapeObj class.
2. Fungsi AddFeature, untuk menambah layer dengan sebuah shape baru.

c. ImageObj Class

Adalah class yang menyimpan gambar hasil pengolahan peta.

d. LabelObj Class

Class yang mengatur label-label yang akan tampil di peta

e. WebObj Class

Class yang berhubungan dengan Alamat-alamat internet

f. PointObj, LineObj, RectObj Class, ShapeObj Class

Berturut-turut adalah Class yang digunakan untuk manipulasi Titik, Garis, Kotak, dan Polygon

g. ScaleBar Class

Class untuk menggambar Scalebar.

h. LegendObj Class

Class untuk menggambar legenda.

2.9.5 Map File

Mapscript memerlukan sebuah file text yang berisi setting layer-layer peta yang akan ditampilkan. File ini disebut Mapfile dengan extensi.map. Susunan dari Mapfile dengan dua layer peta adalah seperti berikut :

```
1 NAME Canada
2 SIZE 400 400
3 STATUS ON
4 SYMBOLSET ./symbols/symbols.sym
5 EXTENT -140 45 -53 80
6 UNITS METERS
7 SHAPEPATH "C:/Inetpub/wwwroot/Mapscript/peta/"
8 WEB
9 IMAGEPATH "C:\Inetpub\wwwroot\Mapscript\tmp\"
10 IMAGEURL "C:\Inetpub\wwwroot\Mapscript\tmp\"
11 END
12 LAYER
13 NAME Province
14 TYPE POLYGON
15 STATUS ON
16 DATA Province
17 CLASS
18 COLOR 20 150 10
19 OUTLINECOLOR 220 220 220
20 END
21 END
22 LAYER
23 NAME roads
24 TYPE LINE
25 STATUS ON
26 DATA Roads
27 CLASS
28 COLOR 250 20 20
29 END
30 END
31 END
```

Pada baris 1, Name adalah judul dari susunan peta yang akan ditampilkan. Baris ke 2, Size adalah ukuran dari file gambar yang akan dihasilkan. Baris ke 5, Extent adalah bingkai sebagai batas view peta. Berturut-turut angka-angka disampingnya adalah batas kiri, batas bawah, batas kanan, dan batas atas. Angka-angka ini disesuaikan dengan posisi dimana peta berada, dengan mengikuti koordinat yang digunakan oleh peta tersebut. Angka-angka ini dapat dilihat saat

kita mengeser-geser mouse di peta ini ketika dilihat dengan program Arcview. Baris berikutnya, Units adalah satuan panjang yang digunakan. ShapePath adalah direktori dimana data peta (shapefile) berada. Bagian berikutnya menyimpan alamat dimana file gambar yang dihasilkan akan disimpan, dan alamat URLnya. Bagian berikutnya adalah susunan dari Layer-layer peta yang akan ditampilkan. Layer yang ditulis lebih awal adalah layer yang lebih bawah. Pada baris 13 dan 23, Name adalah judul. Berikutnya Type adalah tipe dari shapefile. Ada tiga tipe yaitu Polygon, Line, dan Point. Data, pada baris 16 dan 26, adalah nama file petanya. Pada baris 18 dan 28, Color adalah warna yang akan ditampilkan. Dengan angka-angka dibelakangnya adalah susunan RGB (red, green, blue). Disini juga merupakan tempat untuk menentukan tebal garis atau besarnya suatu titik. Dan juga bentuk arsiran pada garis maupun Polygon. Untuk menambah layer-layer yang lain, dilakukan dengan mengulangi setting di bagian ini.

2.10 PHP (Personal Home Page)

PHP singkatan dari *Personal Home Page Tools*, adalah sebuah bahasa *scripting* yang dibundel dengan HTML, yang dijalankan di sisi server. Sebagian besar perintahnya berasal dari bahasa C, Java dan Perl dengan beberapa tambahan fungsi khusus PHP. Bahasa ini memungkinkan para pembuat aplikasi web menyajikan halaman HTML dinamis dan interaktif dengan cepat dan mudah, yang dihasilkan server. PHP juga dimaksudkan untuk mengganti teknologi lama seperti CGI (Common Gateway Interface).

PHP bisa berinteraksi dengan hampir semua teknologi web yang sudah ada. Developer bisa menulis sebuah program PHP yang mengeksekusi suatu

program CGI di server web lain. Fleksibilitas ini sangat bermanfaat bagi pemilik situs-situs web yang besar dan sibuk, karena pemilik masih bisa mempergunakan aplikasi-aplikasi yang sudah terlanjur dibuat di masa lalu dengan CGI, ISAP, atau dengan script seperti Perl, Awk atau Python selama proses imigrasi ke aplikasi baru yang dibuat dengan PHP. Ini mempermudah dan memperhalus peralihan antara teknologi lama dan teknologi baru.

PHP merupakan bahasa *interpreter* yang hampir mirip dengan bahasa C dan *perl* yang memiliki kesederhanaan dalam perintah. *PHP* dapat digunakan bersama dalam *WML* sehingga pembangunan situs *webside* dapat dilakukan dengan cepat dan mudah. *PHP* dapat digunakan untuk memperbarui *database*, membuat *database*, dan mengerjakan perhitungan matematika.

Kelebihan dari *PHP* adalah mampu membuat aplikasi *web* dengan koneksi *database* yang cukup banyak, seperti *Adabas D*, *Dbase*, *Empress*, *FilePro (read only)*, *Hyperwave*, *IBM DB2*, *Informix*, *Ingres*, *InterBase*, *FrontBase*, *MSQL*, *Direct MS-SQL*, *MySQL*, *ODBC*, *Oracle (OCI7 and OCI8)*, *Ovrimos*, *PostgreSQL*, *SQLite*, *Solid*, *Sybase*, *Velocis*, *Unix dbm*.

Selain kehandalan dalam *database PHP* juga bersifat *Cross-platform* yaitu dapat mendukung berbagai jenis sistem operasi misalnya semua varian *Linux*, semua turunan *Unix*, termasuk *HP/UX*, *Solaris* dan *BSD*, tanpa terkecuali untuk jenis *Microsoft Windows*, *Mac OS X*, *RISC OS*. Selain itu jenis *webserver* yang didukung, selain *Apache*, adalah *Microsoft Internet Server*, *Personal Webserver*, *Netscape and Planet Server*, *Orielly Webside Pro Server*, *Caudium*, *Xitami*, *OmniHTTPd*.

2.11 PostgreSQL

هَذَا كِتَابُنَا يَنْطِقُ عَلَيْكُمْ بِالْحَقِّ إِنَّا كُنَّا نَسْتَنْسِخُ مَا كُنْتُمْ تَعْمَلُونَ ﴿٢٩﴾

"Inilah Kitab (catatan) kami yang menuturkan kepadamu dengan benar. Sesungguhnya kami Telah menyuruh mencatat apa yang Telah kamu kerjakan". (QS. Al- Jaatsiyah:29)

Kitab Jalalain menafsirkan ayat diatas dengan: (Inilah kitab catatan Kami) yakni kitab catatan malaikat pencatat amal perbuatan manusia (yang menuturkan terhadap kalian dengan benar. Sesungguhnya Kami telah menyuruh mencatat) menulis dan mengarsipkan (apa yang telah kalian kerjakan).

Ayat di atas menyatakan bahwa Al-Quran telah mengenalkan kepada umat manusia tentang ide pencatatan, dan pencatatan yang dikenalkan Al-Qur'an lebih hebat, karena pencatatannya meliputi data yang besar dan dalam penyimpanan yang lama. Pencatatan yang dilakukan malaikat kepada amal perbuatan manusia tentang baik-buruk sejak lahir hingga mati. Coba bayangkan betapa besar dan banyaknya data yang disimpan.

Sejak manusia mengenal tulisan, manusia sudah mulai mencatat, hanya saja media pencatatan masih sangat sederhana. Bahkan pada zaman Nabi Muhammad Al-Qur'an hanya ditulis di pelepah kurma, tulang, dan batu-batuan. Al-Qur'an belum dibukukan, baru ketika masa sahabat Khulafaur Rosidin baru ada pembukuan, mengingat kebutuhan pendokumentasian sangat penting.

Pada era sekarang pendokumentasian sudah lumayan maju, meskipun tidak sehebat metodenya Allah SWT yang dilakukan oleh malaikat. Manusia sudah mampu menciptakan model pendokumentasian dalam skala yang besar, dan lebih aman dari bahaya pelapukan. Metode itu dikenal dengan istilah Databases.

Ada banyak sekali model-model dari database tersebut diantaranya adalah PostgreSQL.

PostgreSQL atau postgres adalah server *Object Relational-Database Management System* (ORDBMS) open source. Postgres lahir dan dikembangkan oleh Universitas of California, Berkeley(1977-1985). Postgres berawal dari kode Ingres, yang kemudian dikembangkan menjadi database server relational yang berhasil dikomersilkan oleh *Relational Technologies/Ingres Corporation*.

Object-relational database tersebut dinamakan Postgres. Kemudian postgres diambil oleh illustra Information Technologies, untuk dikembangkan sebagai produk database komersial. Kemudian diambil alih oleh Informix dan diintegrasikan ke dalam Informix Universal Server.

Pada tahun 1986 – 1994, postgres telah banyak digunakan untuk penelitian dan produksi aplikasi. Postgres juga digunakan sebagai bahan belajar diberbagai Universitas dan berhenti pada Postgres versi 4.2.

Pada tahun 1994-1995, dua orang mahasiswa lulusan Berkeley, yaitu Jolly Chen dan Andrew Yu, menambahkan kemampuan SQL pada Postgres dan diberi nama Postgres95 dengan menggunakan kode ANSI C. perubahan internal dilakukan pada Postgres95, untuk meningkatkan kinerja dan kemudahan dalam pengembangan sehingga Postgres95 dapat berjalan lebih cepat dibanding dengan postgres versi 4.2.

Penambahan yang dilakukan pada Postgres95 di antaranya adalah:

- a. Bahasa query PostQUEL diganti menjadi SQL(implementasi pada server).
subQuery belum didukung namun dapat dilakukan dengan menggunakan

fungsi-fungsi SQL yang didefinisikan. Dimasukkan perintah klausa GROUP BY pada SQL.

- b. Tersedianya program psql untuk interaktif query dengan SQL. Program psql ini menggunakan GNU readline.
- c. Interface object besar ditinggalkan. Mekanisme yang digunakan untuk menyimpan object besar yaitu menggunakan Inversion.
- d. Penghapusan system instance-level.
- e. Tutorial singkat penggunaan fitur SQL, disertakan dengan contoh programnya.
- f. GNU make digunakan untuk proses build. Postgres95 bisa dikompilasikan dengan GCC.

Awal tahun 1996, dibentuk sebuah tim yang tergabung dalam sebuah komunitas yang dipersiapkan untuk menangani lonjakan permintaan yang besar, bagi sebuah database server yang open source. Sehingga dengan adanya tim tersebut PostgreSQL dikembangkan lagi dengan mengevaluasi kesalahan-kesalahan yang ditemukan. Dan pada tahun 1996 nama database server Postgres95 dihapus dan diganti menjadi PostgreSQL. Release PostgreSQL dilakukan setiap tiga hingga lima bulan, dimana tiap periode dilakukan pengembangan selama dua hingga tiga bulan.

Beberapa prestasi yang telah diperoleh oleh PostgreSQL, diantaranya adalah terpilih sebagai Best Database Management System oleh Linux World Editor's pada tahun 1999 dan sebagai Best Database oleh Linux Journal Editor's Choice Award pada tahun 2000.

PostgreSQL menawarkan fitur-fitur yang dimiliki oleh DBMS komersial di antaranya:

- a. Dukungan tipe data yang banyak digunakan pada database komersial, seperti floating point, integer, character string, money, date/time dan tipe data binary.
- b. Dukungan tipe data yang semakin beragam, seperti tipe data untuk geometri (seperti Point, Polygon, Circle, dan Line), tipe data jaringan(TCP/IP) untuk menyimpan data pada IP4, IP6 dan Mac Address(Lnet,cidr,maddr).
- c. Didukung tipe data array dan tipe data komposit serta konsep tipe data Object Identifiers (OIDs), yang digunakan PostgreSQL sebagai primary key pada beberapa table. Pada PostgreSQL versi 8.x, OIDs digunakan sebagai default pada pembuatan table pada database.
- d. Dukungan penyimpanan binary large object(gambar, suara, video). Kinerja PostgreSQL sebagai server database object-relational semakin memudahkan user untuk mengimplementasikan system aplikasi yang dibuat
- e. Dukungan fitur Foreign Key dan referential integrity, membuat PostgreSQL banyak dipilih sebagai database server dalam pengembangan aplikasi.
- f. PostgreSQL telah mengimplementasikan tipe join SQL99: inner join, left, right, full outer join, natural join, yang mempermudah proses query.

- g. Dukungan fitur view dan trigger menyederhanakan proses query yang kompleks pada sisi server. Fungsi trigger bisa ditulis dalam bahasa C, Procedural Language.
- h. Dukungan fungsi Full-text indexing.
- i. Tersedianya fungsi ODBC.
- j. Fitur rule yang dapat digunakan untuk memanipulasi data pada operasi SQL yang dilakukan, serta fungsi yang berkenaan dengan keamanan data, seperti fungsi hash cryptographic(MD5, SHA1).
- k. Dukungan standar regular expression (full POSIX) dan case-insensitive regular expression matching.
- l. Dukungan dari bahasa pemrograman pada sisi server, seperti: C, SQL, PL/pgSQL, Tcl, PERL, Python, dan ruby.
- m. Temporary table yang digunakan sebagai tabel temporer, dimana akan dihapus pada saat koneksi database berakhir.
- n. Dukungan fitur schema yang memungkinkan:
 - 1. Banyak user menggunakan database yang sama tanpa mengganggu yang lainnya.
 - 2. Untuk memudahkan mengorganisasikan database dalam satu program tertentu.
 - 3. PostgreSQL 8.0 adalah PostgreSQL pertama yang bisa dioperasikan pada Microsoft Windows sebagai server, dimana PostgreSQL berjalan sebagai salah satu service windows server. Release

PosgreSQL ini bisa dijalankan pada form Windows XP, Windows 2000 maupun Windows 2003.

PosgreSQL memberikan fitur dan kinerja yang lengkap untuk Administrasi sever, seperti:

1. Model keamanan untuk setiap user dan group pada obyek database
2. Akses yang bisa dibatasi ke sever berdasar kan host, nomor IP, user name, dan database tertentu.
3. Dukungan tools dalam proses backup semua ojek database bukan hanya tabel saja tetapi termasuk juga fungsi, triger, privillages, tipe, data custom.
4. Operasi backup data dan kebalikannya (restore) data dilakukan oleh user database yang mempunyai privileges superuser.
5. Dukungan backup data yang menggunakan database cluster dengan WAL archive.
6. Dukungan kerberos untuk proses authentication.
7. Koneksi database yang terenkripsi dengan menggunakan SSL dan SSH.
8. Dukungan koneksi lewat TCP/IP atau local unix dimain socket.
9. Dukungan database replikasi dengan menggunakan aplikasi open source DB MIRROR.
10. Dukungan Tabelaespace yang memungkinkan administrator server memilih file sistem dan media penyimpanan untuk tabel tertentu dalam database.

PostgreSQL berjalan di banyak platform sistem operasi, sehingga database server ini banyak digunakan dalam pengembangan aplikasi, diantaranya:

1. *FreeBSD (x86,Alpha).*
2. *BSD/OS (x86,Sparc).*
3. *OpenBSD (x86,Sparc).*
4. *NetBSD (x86, Alpa, ARM, m68k, Power PC, Sparc, VAX).*
5. *Digital Unix*
6. *Linux (x86, Alpa, ARM, MIPS, Power PC, Sparc. s/390).*
7. *SCO OponServer.*
8. *SCO UnixWare.*
9. *SunOS 4*
10. *Sun Solaris (x86, Sparc).*
11. *Compaq Tru64 Unix.*
12. *AIX.*
13. *BeOS.*
14. *Windows (XP, WIN 2000, WIN 2003)*

Dukungan dari database pemrograman berupa library yang bisa digunakan untuk koneksi ke sever database PostgreSQL, menyebabkan pengguna database PostgreSQL berasal dari berbagai latarbelakan penguasaan bahasa pemrograman, diantaranya:

1. *Python (lewat PyGreSql/PoPy)*
2. *JDBC (Java Database Connectivity)*
3. *PHP Librar php-pgsql*

4. *Tcl*
5. *Standard SQL embeded C*
6. *Perl (Standard DBI/DBD)*
7. *Native C dan C++ API*
8. *ODBC*

2.12 PostGIS

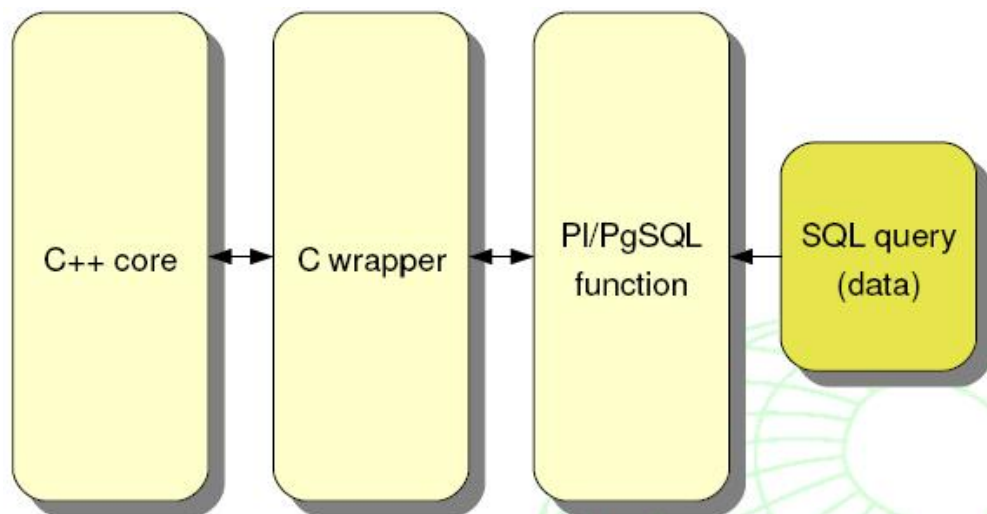
PostGIS adalah sebuah perangkat lunak tambahan (*geo-spatial extension*) untuk database PostgreSQL yang berfungsi untuk menyimpan obyek-obyek sistem informasi geografis (data spasial) (Anonymous3. 2007). PostGIS dikembangkan sebagai sebuah teknologi database spasial *open source* dibawah lisensi *GNU general public licence*. fungsi PostGIS pada database PostgreSQL adalah sebagai *spatially enables* (menambahkan tabel khusus untuk penyimpanan data-data sapsial) untuk menyimpan obyek-obyek sistem informasi geografis (data spasial) dalam database PostgreSQL. selain menambahkan tabel data spasial, PostGIS juga akan menambahkan tabel geometry (*geom_table*) untuk menyimpan informasi dan atribut geometri data spasial, sehingga dengan adanya informasi geometri peta tersebut bisa dilakukan analisis dan pemrosesan data sig menggunakan *query sql*.

Dalam PostGIS juga sudah terintegrasi dua konverter yaitu *shp2pgsql* dan *pgsql2shp*. fungsi *shp2pgsql* adalah untuk mengkonversi data shapefile (format yang dikembangkan ESRI untuk menyimpan informasi–informasi atribut dan geometri data spasial) ke format databse (format.sql) sehingga memungkinkan database postgres untuk membuat kolom geometri yang digunakan oleh 52

PHPMapscript untuk menampilkan fitur geometri, sedangkan *pgsql2shp* berfungsi mengkonversi database (format.sql) ke format shapefile.

2.13 PgRouting

PgRouting adalah sebuah *tools open source* yang menyediakan fungsionalitas *routing* pada database PostgreSQL (Anonymous4.. 2007). PgRouting dapat digunakan untuk menyelesaikan masalah pencarian jalur terpendek (*Shortest Path*) dan juga *Travelling Salesman Problem* (TSP) (Patrushev, 2007). Saat ini fungsionalitas *routing* bawaan yang disediakan oleh modul pgRouting adalah fungsionalitas routing dengan menggunakan algoritma dijkstra (*shortest_path_dijkstra*), algoritma A* (*shortest_path_astar*), algoritma *Shooting Star* dan fungsi untuk menangani masalah *Travelling Salesman Problem* (TSP). Berikut ini adalah gambar cara kerja fungsi pada modul pgRouting:



Gambar 2.6 Cara Kerja Modul pgRouting

(sumber: (<http://pgrouting.postlbs.org/wiki/>))

2.13.1 Algoritma Dijkstra

Graf merupakan suatu cabang ilmu yang memiliki banyak terapan. Banyak sekali struktur yang bisa direpresentasikan dengan graf, dan banyak masalah yang bisa diselesaikan dengan bantuan graf. Sering kali graf digunakan untuk merepresentasikan suatu jaringan. Misalkan jaringan jalan raya dimodelkan graf dengan kota sebagai simpul (vertex/node) dan jalan yang menghubungkan setiap kotanya sebagai sisi (edge) yang bobotnya (weight) adalah panjang dari jalan tersebut. Dalam beberapa model persoalan dimungkinkan bahwa bobot dari suatu sisi bernilai negatif. Misalkan simpul merepresentasikan bandara, sisi merepresentasikan penerbangan yang memungkinkan, dan bobot dari setiap sisi adalah biaya yang dikeluarkan dalam penerbangan tersebut. Untuk suatu kasus dimana seseorang akan dibayar untuk menempuh rute tertentu oleh suatu biro penerbangan, maka bobotnya akan bernilai negatif. Lintasan terpendek merupakan salah satu dari masalah yang dapat diselesaikan dengan graf.

Jika diberikan sebuah graf berbobot, masalah lintasan terpendek adalah bagaimana kita mencari sebuah jalur pada graf yang meminimalkan jumlah bobot sisi pembentuk jalur tersebut.

Terdapat beberapa macam persoalan lintasan terpendek antara lain:

- a. Lintasan terpendek antara dua buah simpul tertentu (a pair shortest path).
- b. Lintasan terpendek antara semua pasangan simpul (all pairs shortest path).

- c. Lintasan terpendek dari simpul tertentu ke semua simpul yang lain (single-source shortest path).
- d. Lintasan terpendek antara dua buah simpul yang melalui beberapa simpul tertentu (intermediate shortest path).

Beberapa algoritma yang digunakan untuk menyelesaikan persoalan ini adalah algoritma Dijkstra, algoritma Bellman-Ford, dan algoritma Floyd-Warshall.

Algoritma Dijkstra, dinamai menurut penemunya, Edsger Dijkstra, adalah algoritma dengan prinsip greedy yang memecahkan masalah lintasan terpendek untuk sebuah graf berarah dengan bobot sisi yang tidak negatif. Misalnya, bila simpul dari sebuah graph melambangkan kota-kota dan bobot tiap simpul melambangkan jarak antara kota-kota tersebut, algoritma Dijkstra dapat digunakan untuk menemukan jarak terpendek antara dua kota. Input algoritma ini adalah sebuah graf berarah dan berbobot, G dan sebuah source vertex s dalam G . V adalah himpunan semua simpul dalam graph G . Setiap sisi dari graph ini adalah pasangan vertices (u,v) yang melambangkan hubungan dari vertex u ke vertex v . Himpunan semua edge disebut E . Weights dari edges dihitung dengan fungsi $w: E \rightarrow [0, \infty)$; jadi $w(u,v)$ adalah jarak non-negatif dari vertex u ke vertex v . Cost dari sebuah edge dapat dianggap sebagai jarak antara dua vertex, yaitu jumlah jarak semua edge dalam path tersebut. Untuk sepasang vertex s dan t dalam V , algoritma ini menghitung jarak terpendek dari s ke t .

Dijkstra Algorithms (Shortest Path Algorithms) adalah algoritma untuk menemukan jarak terpendek dari suatu vertex ke vertex yang lainnya pada suatu

graph yang berbobot, dimana jarak antar vertex adalah bobot dari tiap edge pada graph tersebut. Algoritma dijkstra mencari jarak terpendek untuk tiap vertex dari suatu graph yang berbobot. Algoritma dijkstra mencari jarak terpendek dari node asal ke vertex terdekatnya, kemudian ke vertex kedua, dan seterusnya. Secara umum, sebelum dilakukan initerasi, algoritma sudah mengidentifikasi jarak terdekat dari $i-1$ vertex terdekatnya. Selama seluruh edge berbobot tertentu yang (positif), maka vertex terdekat berikutnya dari node asal dapat ditemukan selama vertex berdekatan dengan vertex T_i . Kumpulan vertex yang berdekatan dengan vertex di T_i dapat dikatakan sebagai “fringe vertices”. Vertex inilah yang merupakan kandidat dari algoritma dijkstra untuk memilih vertex berikutnya dari node asal. Cara kerja algoritma Dijkstra memakai strategi greedy, dimana pada setiap langkah dipilih sisi dengan bobot terkecil yang menghubungkan sebuah simpul yang sudah terpilih dengan simpul lain yang belum terpilih.

2.14 Permodelan Sistem

Model analisis harus dapat mencapai tiga sasaran utama, yaitu menggambarkan apa yang dibutuhkan oleh pelanggan, membangun dasar bagi pembuatan desain perangkat lunak, membatasi serangkaian persyaratan yang dapat divalidasi begitu perangkat lunak dibangun.

2.14.1 Data Flow Diagram

Diagram aliran data merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil. Salah satu keuntungan menggunakan diagram aliran data adalah memudahkan pemakai atau *user* yang

kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan.

Beberapa jenis diagram pada Data Flow Diagram, diantaranya :

a) *Diagram Konteks*

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh *input* ke sistem atau *output* dari sistem. Ia akan memberi gambaran tentang keseluruhan sistem. Sistem akan dibatasi oleh *boundary* (dapat digambarkan dengan garis putus). Dalam *diagram konteks* hanya ada satu proses. Tidak boleh ada *store* dalam diagram konteks.

b) *Diagram Nol/Zero (Overview Diagram)*

Diagram nol adalah diagram yang menggambarkan proses dari *dataflow diagram*. Diagram nol memberikan pandangan secara menyeluruh mengenai sistem yang ditangani, menunjukkan tentang fungsi-fungsi utama atau proses yang ada, aliran data, dan eksternal entity. Pada level ini sudah dimungkinkan adanya/ digambarkannya data store yang digunakan. Untuk proses yang tidak dirinci lagi pada level selanjutnya, simbol ‘*’ atau ‘P’ (functional primitive) dapat ditambahkan pada akhir nomor proses. Keseimbangan *input* dan *output* (*balancing*) antara diagram 0 dengan diagram konteks harus terpelihara.

c) *Diagram Rinci (Level Diagram)*

Diagram rinci adalah diagram yang menguraikan proses apa yang ada dalam diagram zero atau diagram level di atasnya. Dalam satu level sebaiknya

tidak terdapat lebih dari 7 buah proses dan maksimal 9 proses, bila lebih maka harus dilakukan dekomposisi.

2.14.2 Entity Relationship Diagram (ERD)

ERD adalah model konseptual yang mendeskripsikan hubungan antara penyimpanan (dalam *DFD*). *ERD* digunakan untuk memodelkan struktur data dan hubungan antar data. Dengan *ERD*, model dapat diuji dengan mengabaikan proses yang dilakukan.

Diagram relasi entitas (entity relationship diagram-ERD) seperti ditunjukkan oleh namanya, berhubungan dengan data didalam entitas dan hubungan antar entitas. Ketika pengguna dan spesialis informasi mulai untuk berkomunikasi tentang kebutuhan data untuk suatu sistem informasi, mereka akan berbicara mengenai pengumpulan *field-field* data yang saling berhubungan daripada *field-field* data individu. Kumpulan konseptual *field-field* data yang saling berhubungan ini disebut entitas (*entities*). Meskipun secara intuitif akan langsung tertarik untuk menganggap entitas sebagai tabel-tabel, kita tidak dapat melakukannya. Tabel merupakan hasil dari pemecahan entitas menjadi unit-unit berukuran lebih kecil yang mengikuti aturan-aturan struktur basis data. Suatu entitas dapat berubah menjadi satu tabel, namun sering kali suatu entitas dipecah menjadi beberapa tabel. ERD adalah tingkat konseptualisasi data yang lebih tinggi daripada tabel.

ERD juga mengungkapkan entitas-entitas mana yang sebaiknya secara konseptual dihubungkan dengan entitas yang lain. Hubungan antar entitas tidak ditentukan oleh *field-field* data yang sama dalam masing-masing entitas, karena

selama tahap awal pengembangan sistem ini ketika ERD pertama kali dikonseptualisasikan, *field-field* data yang sebenarnya tidaklah diketahui. Akan tetapi, pengguna dan para profesional sistem informasi dapat mengonseptualisasikan bagaimana *record-record* didalam entitas dapat berhubungan dengan *record-record* di entitas-entitas yang lain.

Entitas-entitas di dalam ERD akan memiliki nama, sama halnya seperti tabel yang memiliki nama. Relasi juga akan menghubungkan entitas-entitas sama halnya seperti garis-garis yang menghubungkan tabel-tabel melalui *field-field* yang sama diantara tabel. Relasi ERD akan menunjukkan jika satu *record* dalam satu entitas akan berhubungan dengan satu atau lebih *record* di entitas yang lain.

2.14.2.1 Kardinalitas Relasi

Dalam *ERD* hubungan (relasi) dapat terdiri dari sejumlah entitas yang disebut dengan derajat relasi. Derajat relasi maksimum disebut dengan kardinalitas sedangkan derajat minimum disebut dengan modalitas. Jadi kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas lain. Kardinalitas relasi yang terjadi diantara dua himpunan entitas (misalnya A dan B) dapat berupa :

- a. Satu ke satu (*one to one/ 1-1*)

Setiap entitas pada himpunan entitas A dapat berelasi dengan paling banyak satu entitas pada himpunan entitas B, demikian juga sebaliknya.

- b. Satu ke banyak (*one to many/ 1- N*)

Setiap entitas pada himpunan entitas A dapat berelasi dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya.

- c. Banyak ke banyak (*many to many*/ N –N)

Setiap entitas pada himpunan entitas A dapat berelasi dengan banyak entitas pada himpunan entitas B, demikian juga sebaliknya.

2.14.2.2 Tahapan Pembuatan ERD

Diagram *ER* dibuat secara bertahap, ada dua kelompok pentahapan yang biasa ditempuh didalam pembuatan diagram *ER*, yaitu :

- a. Tahap pembuatan diagram ER awal (preliminary design)
- b. Tahap optimasi diagram ER (final design)

Tujuan tahap pertama adalah untuk mendapatkan sebuah rancangan basis data minimal yang dapat mengakomodasi kebutuhan penyimpanan data terhadap sistem yang sedang ditinjau. Tahap awal ini umumnya mengabaikan anomali-anomali (proses pada basis data yang memberikan efek samping yang tidak diharapkan) yang memang ada sebagai suatu fakta. Anomali-anomali tersebut biasanya baru dipertimbangkan pada tahap kedua.

Tahap kedua mempertimbangkan anomali-anomali dan juga memperhatikan aspek-aspek efisiensi, performansi dan fleksibilitas. Tiga hal tersebut seringkali dapat saling bertolak belakang. Karena itu, tahap kedua ini ditempuh dengan melakukan koreksi terhadap tahap pertama. Bentuk koreksi yang terjadi dapat berupa pendekomposisian himpunan entitas, penggabungan himpunan entitas, pengubahan derajat relasi, penambahan relasi baru atau perubahan (penambahan dan pengurangan) atribut-atribut untuk masing-masing

entitas dan relasi. Langkah-langkah teknis yang dapat dilakukan untuk mendapatkan *ERD* awal adalah :

- a. Mengidentifikasi dan menetapkan seluruh himpunan entitas yang akan terlibat.
- b. Menentukan atribut-atribut key (kunci) dari masing-masing himpunan entitas.
- c. Mengidentifikasi dan menetapkan seluruh himpunan relasi diantara himpunan entitas-himpunan entitas yang ada beserta foreign-keynya (kunci asing/kunci tamu).
- d. Menentukan derajat/kardinalitas relasi untuk setiap himpunan relasi.
- e. Melengkapi himpunan entitas dan himpunan relasi dengan atribut dekriptif (atribut yang bukan kunci)

BAB III

ANALISIS DAN PERANCANGAN APLIKASI

3.1 Analisis Sistem

Tujuan dari analisis sistem adalah untuk mengidentifikasi kebutuhan sistem, kebutuhan perangkat keras (*hardware*), maupun kebutuhan perangkat lunak (*software*) sebagai dasar kebutuhan untuk perancangan sistem. Analisis ini akan mengidentifikasi permasalahan dan kekurangan yang ada pada sistem yang akan dibangun meliputi analisis sistem spesifikasi aplikasi, spesifikasi pengguna, dan lingkungan operasi.

3.1.1 Spesifikasi Aplikasi

Spesifikasi yang akan dijelaskan adalah aplikasi *web-gis*, dan yang akan dibangun dengan kemampuan sebagai berikut:

1. Menampilkan profil lembaga/individu, informasi/berita, agenda, pengumuman.
2. Menampung surat masuk/kontak kami, melalui email.
3. Menampilkan desa terkena banjir, berdasarkan hubungan ketinggian desa dan ketinggian air.
4. Menampilkan informasi tentang dampak banjir.
5. Menampilkan jalur terpendek antar kecamatan/posko di kabupaten Sragen.
6. Memiliki kemampuan navigasi peta seperti *zoom in*, *zoom out*, *zoom all*, *query*, dan legenda.

3.1.2 Spesifikasi Pengguna

Aplikasi ini ditujukan untuk digunakan oleh masyarakat, atau instansi terkait seperti Bappeda atau Dinas Pekerjaan Umum kabupaten Sragen sebagai media informasi tentang masalah banjir. Dan juga donor/donator yang peduli akan permasalahan bencana alam.

3.1.3 Lingkungan Operasi

Untuk membangun aplikasi web sesuai dengan spesifikasi kebutuhan, dibutuhkan lingkungan operasi sebagai berikut:

- a. Sistem operasi Windows XP.

Sistem operasi ini dipilih karena sudah banyak dikenal dan sangat *familiar* oleh para pengguna, sehingga mudah dalam pengoperasiannya.

- b. *Web server*

Web server digunakan untuk memberikan layanan web sehingga dapat diakses oleh pengguna baik yang berada pada jaringan lokal maupun pada jaringan internet. Web server yang digunakan adalah Apache. Perangkat lunak ini digunakan karena bersifat gratis dan memiliki kemampuan kerja yang tinggi dibanding dengan perangkat lunak web server lainnya.

- c. Map Server

Map Server digunakan sebagai perangkat lunak visualisasi peta digital. Alasan dipilih perangkat lunak ini adalah:

1. Bersifat gratis dan *open source*.
2. Mendukung format peta digital SIG yang banyak digunakan yaitu SHP.

3. Selain format SHP, Map Server juga mendukung format data yang berasal dari basis data PostgreSQL/PostGIS sehingga memungkinkan peta digital disimpan dalam basisdata dan terintegrasi dengan data-data atribut.
4. Memiliki API dalam bahasa pemrograman PHP.
5. Banyak tutorialnya.

d. PostgreSQL

PostgreSQL adalah perangkat lunak DBMS sedangkan PostGIS adalah perangkat lunak tambahan (*third party*) yang berfungsi untuk menambah kemampuan dari PostgreSQL dalam menangani basis data spasial. Alasan dipilihnya perangkat lunak ini adalah karena bersifat gratis, dan DBMS yang terdapat PostGIS yang digunakan untuk menyimpan data yang bersifat spasial (keruangan) dan merupakan DBMS yang direkomendasikan bagi pengguna Map Server.

e. *Interpreter* PHP

- f. PHP adalah bahasa pemrograman *server side* yang bekerja pada sisi server yang berfungsi untuk menangani *request* dari pengguna dan berkomunikasi dengan Map Server serta DBMS PostgreSQL melalui API yang disediakan. Perangkat lunak ini digunakan karena bersifat gratis dan *open source*.

3.2 Perancangan Sistem

Tahap ini ditujukan untuk mencapai sebuah bentuk yang baik dan optimal pada aplikasi yang akan dibangun, berdasarkan pada analisis sistem seperti yang

telah dijelaskan sebelumnya. Pembangunan aplikasi ini dengan berbagai pertimbangan faktor-faktor permasalahan dan kebutuhan sistem. Upaya yang dilakukan adalah dengan berusaha mencari kombinasi penggunaan teknologi, perangkat keras (*hardware*), dan perangkat lunak (*software*) yang tepat sehingga diperoleh hasil yang optimal dan mudah untuk diimplementasikan.

3.3 Perancangan Aplikasi Web

Bagian ini akan membahas tentang proses perancangan Aplikasi web dengan mempertimbangkan berbagai kebutuhan atau beberapa spesifikasi yang telah ditetapkan ditahap sebelumnya, yakni tahap analisis system. Dalam proses ini rancangan meliputi modul aplikasi, rancangan basisdata dan rancangan antarmuka (*interface*).

3.3.1 Perancangan Modul Aplikasi

Dalam perancangan modul aplikasi ini terbagi menjadi dua bagian. Yaitu modul aplikasi peta dan modul aplikasi web.

3.1.1.1 Modul Aplikasi Peta

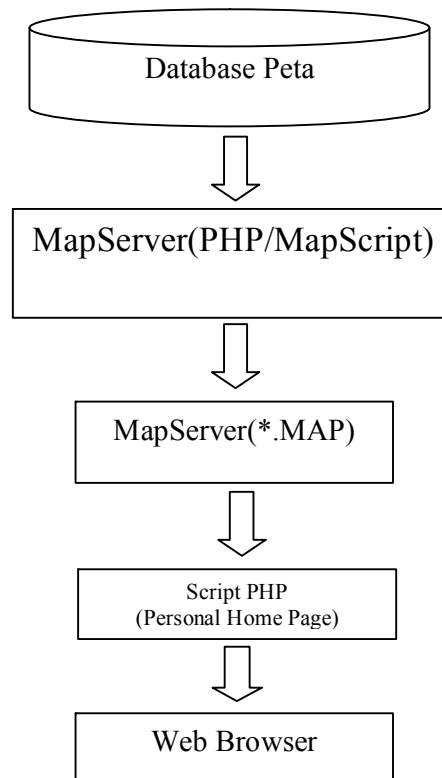
Modul aplikasi dibagi menjadi dua buah modul yaitu modul visualisasi data, modul *query* peta.

1. Modul Visualisasi Data

Modul ini berfungsi untuk memvisualisasikan peta digital dari kontur kabupaten, jalan, administrasi desa, sungai, kecamatan beserta atribut-atribut peta seperti peta referensi, legenda, *scalebar*, inputan tinggi air, dan

inputan pencarian rute. Modul ini dibangun dengan menggunakan PHP/MapScript dan script php. Dengan MapScript dapat dibuat objek peta dengan format *.MAP yang kemudian dapat *render* menjadi format PNG. Sumber data untuk semua peta adalah tabel basis data PostgreSQL/PostGIS, data spasial maupun atribut informasi petanya. Data tersebut merupakan hasil konversi dari data SHP ke SQL dengan menggunakan aplikasi shp2sql data loader yang sudah terdapat di aplikasi PostgreSQL. Alur proses modul ini dapat dilihat pada gambar di bawah ini:

Alur proses modul ini dapat dilihat pada gambar 3.1:



Gambar 3.1 Alur proses modul visualisasi data Peta

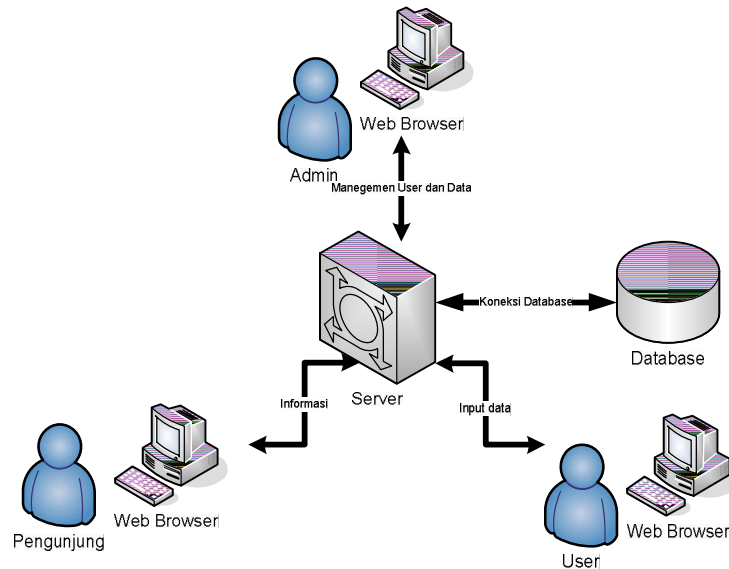
2. Modul *Query* Peta

Fungsi dari modul ini adalah untuk menangani proses *query* yang artinya adalah proses untuk menampilkan informasi yang terkandung oleh data spasial (peta) yang dilakukan pengguna saat mengklik salah satu titik yang ada pada *interface* peta. Hasil dari klik (*query*) ini adalah informasi yang melekat pada objek spasial, yang pada aplikasi ini adalah administrasi desa. Aplikasi ini akan menampilkan informasi yang melekat pada layer administrasi desa sesuai posisi klik user. Dan modul ini dibangun dengan Script PHP/MapScript yang dibantu oleh bahasa pemrograman PHP. Alur kerja ini dapat dilihat pada gambar.

Gambar 3.2 Alur kerja modul *query* posisi

3.1.1.2 Modul Aplikasi Web

Modul ini berfungsi sebagai acuan aliran data secara umum dimana terdapat tiga entitas yang terdiri dari pengunjung, user, dan admin, dengan menggunakan database PostgreSQL.



Gambar 3.3 Aliran data secara umum

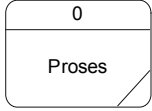



3.4 Perancangan Basis Data

Aplikasi yang akan dibangun membutuhkan basisdata untuk menyimpan data-data spasial beserta data atribut dan data informasi yang terkait dengan aplikasi ini. Agar basisdata yang dibangun dapat diimplementasikan dengan baik, maka terlebih dahulu dilakukan proses perancangan basisdata. Untuk melakukan proses desain secara umum digunakan DFD (Data Flow Diagram). yang menjelaskan tentang fungsi-fungsi dan alur kerja yang terdapat dalam sistem informasi tersebut secara logika. Untuk mendesain relasi antar entitas dan tabel digunakan ERD (*Entity Relationship Diagram*).

Data flow diagram, menjelaskan kepada user bagaimana nantinya fungsi-fungsi di sistem informasi secara logika akan bekerja. Data flow diagram akan menginterpretasikan *Logical Model* dari suatu sistem.

Beberapa simbol yang digunakan dalam DFD antara lain:

Tabel 3.1 Simbol dalam DFD

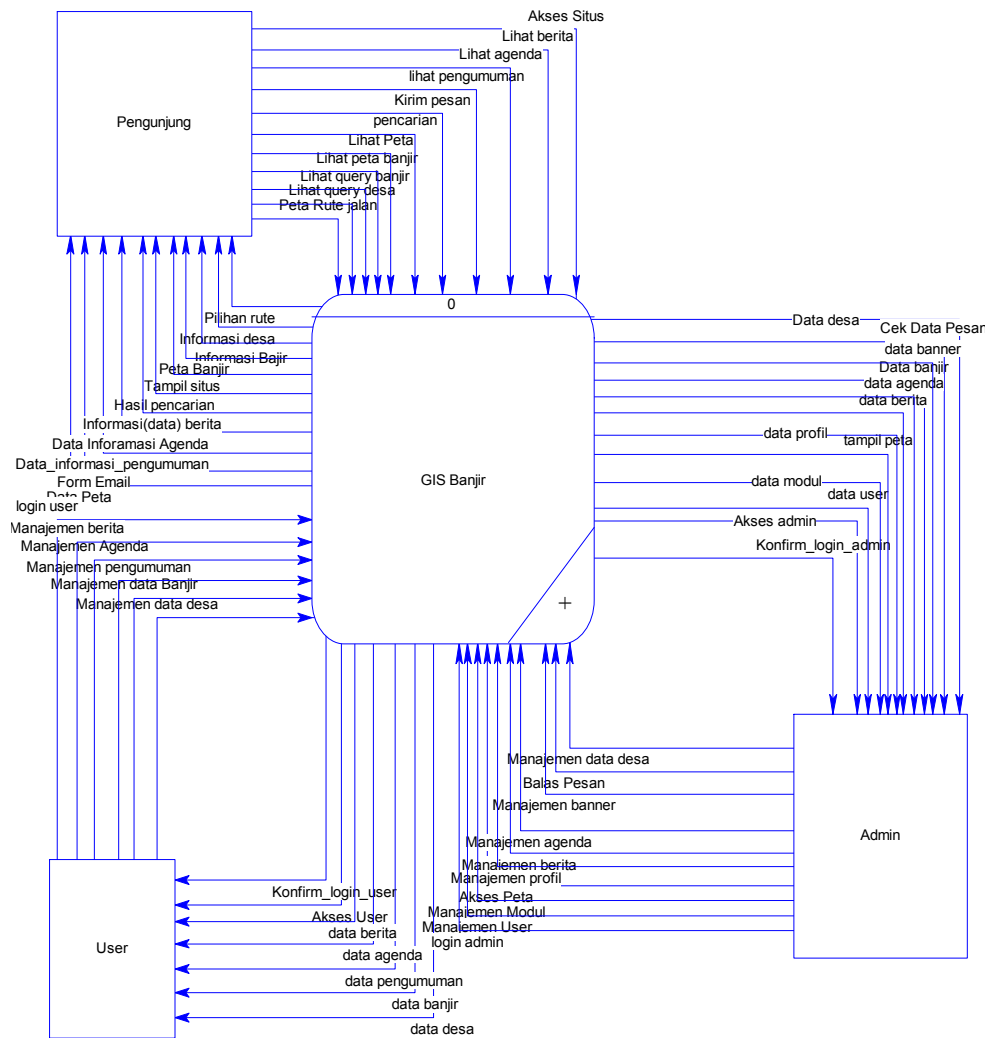
Simbol	Nama Simbol	Keterangan
	Proses (<i>Process</i>)	Merupakan kegiatan atau pekerjaan yang dilakukan oleh orang atau mesin komputer, dimana aliran data masuk, ditransformasikan ke aliran data keluar.
	Penyimpanan Data (<i>Data Store</i>)	Komponen yang berfungsi untuk menyimpan data/file adalah fungsi yang mentransformasikan data secara umum.
	Sumber data / tujuan data (<i>Entity</i>)	Merupakan sumber atau tujuan dari arus data yang dapat digambarkan secara fisik, seseorang atau sekelompok orang atau system lain.
	Aliran Data (<i>Flowline</i>)	Disimbolkan dengan anak panah, dimana arus data mengalir diantara proses, simpangan data, kesatuan luar, kesatuan ruang.

3.4.1. Konteks Diagram

Konteks diagram (*Diagram konteks*) adalah diagram yang terdiri dari suatu proses yang menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh *input* ke sistem atau *output* dari sistem. Ia akan memberi gambaran tentang keseluruhan sistem.

Dalam *diagram konteks* hanya ada satu proses. Tidak boleh ada *store* dalam diagram konteks.

Konteks Diagram SIG Banjir dapat dilihat pada gambar 3.4. berikut ini:



Gambar 3.4 Konteks Diagram SIG Banjir

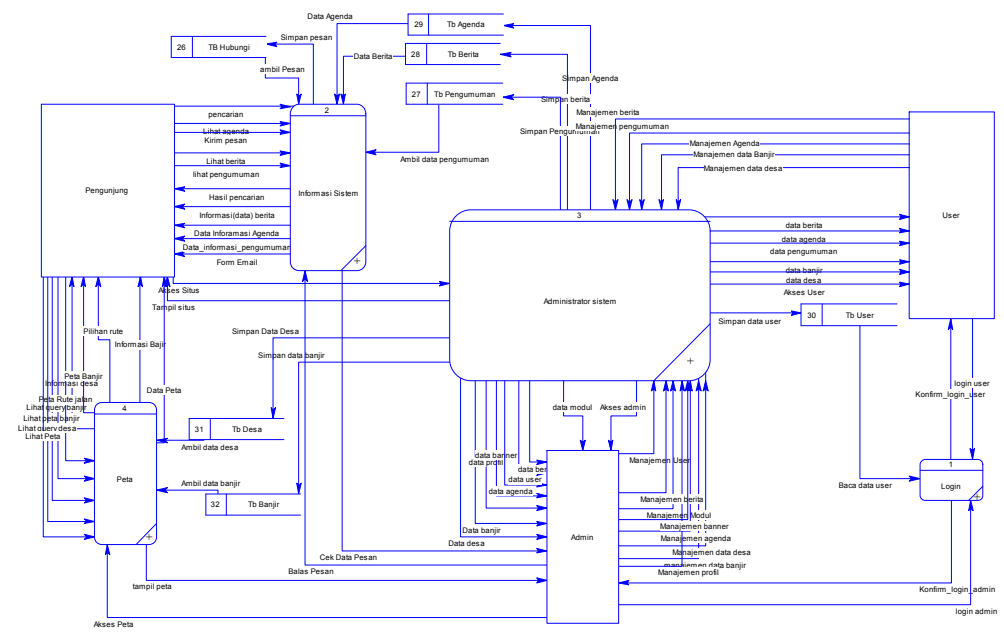
Pada Konteks Diagram di atas dapat dilihat bahwa proses yang terjadi dalam aplikasi SIG Banjir Kabupaten Sragen melibatkan tiga sumber atau tujuan data yaitu Pengunjung, User, dan Admin.

Bagian admin bertugas diantaranya input user dan password, input modul, input berita, edit desa, dan edit informasi banjir. Sementara tugas user hampir sama dengan tugas admin akan tetapi hak akses dari user hanya sebatas input berita, input agenda, dan input pengumuman.

Sementara pengunjung dalam aplikasi ini dapat memperoleh informasi berita, agenda, pengumuman, serta bisa mengirim pesan ke pengelola situs yang ditampilkan, dengan memilih menu yang sudah disediakan.

3.4.2. *Data Flow Diagram (DFD)*

DFD terperinci adalah penjabaran dari konteks diagram. Bentuk DFD terperinci level 1 dari SIG Banjir adalah sebagai berikut:

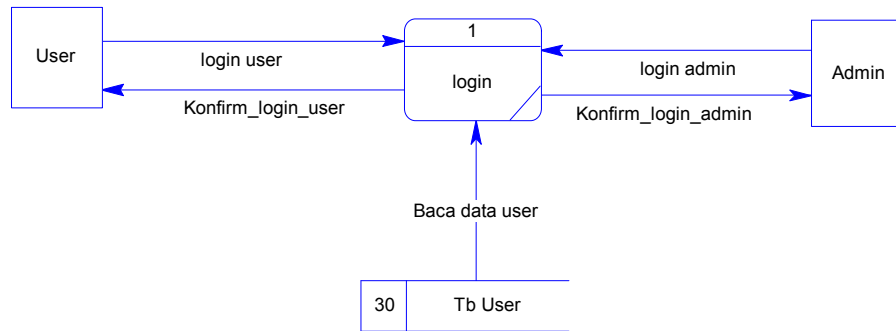


Gambar 3.5 DFD Level 1 SIG Banjir

Dalam level 1 SIG Banjir terdapat empat proses yang dilakukan oleh admin dan pengguna.

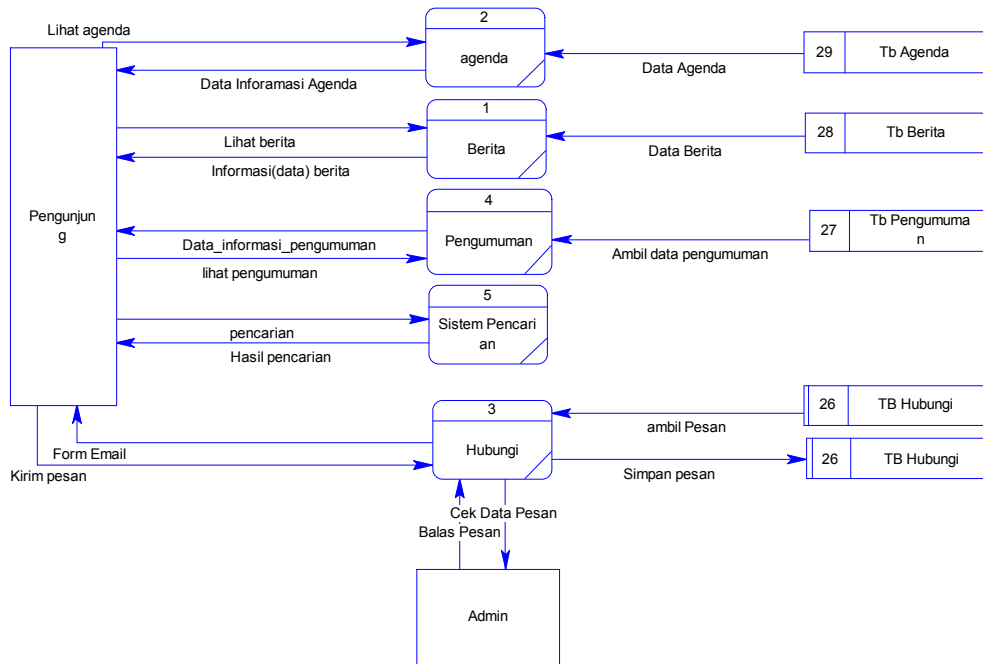
- a. Proses 1 yaitu login user. Proses ini digunakan untuk mengecek login user untuk mendapatkan hak akses untuk mengelola *content* database. Setiap kali user melakukan login, data yang dimasukkan akan dicocokkan dengan data yang ada dalam database. Apabila data tersebut cocok dengan database, maka user dapat akses untuk mengelola *content* database. Namun, apabila tidak cocok, maka sistem akan menolak user tersebut untuk masuk kedalam sistem.
- b. Proses 2 yaitu informasi merupakan proses yang menangani semua informasi yang akan ditampilkan dalam website. Informasi tersebut diambil dari database. Diantaranya adalah berita dan pengumuman, dll.
- c. Proses 3 yaitu administrator sistem yang menangani manajemen data yang akan ditampilkan dalam website. Data-data disimpan dalam database yang ada.
- d. Proses 4 yaitu peta merupakan proses yang menangani semua yang berhubungan dengan peta yang ditampilkan dalam website. Namun admin dapat melakukan penambahan data peta sesuai keperluan. Dan pengguna juga dapat memperoleh informasi jalur terpendek, prediksi banjir dengan memasukkan inputan yang dibutuhkan.

Untuk lebih jelasnya, Subproses ini selanjutnya akan dijabarkan dalam DFD level 2.



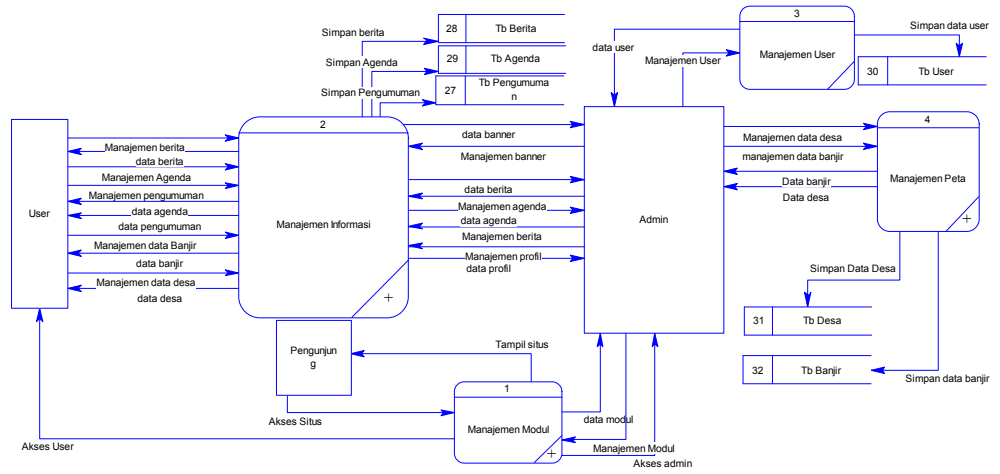
Gambar 3.6 DFD Level 2 Proses Login

Pada DFD level 2 proses Login ini menjelaskan tentang proses login, baik login oleh user maupun login sebagai admin. User dan admin hanya dibedakan pada hak akses pengelolaan situs saja. Masing-masing ketika login, data inputan akan di cocokan dengan store yang sudah ada. Jika cocok maka diberi hak untuk akses.



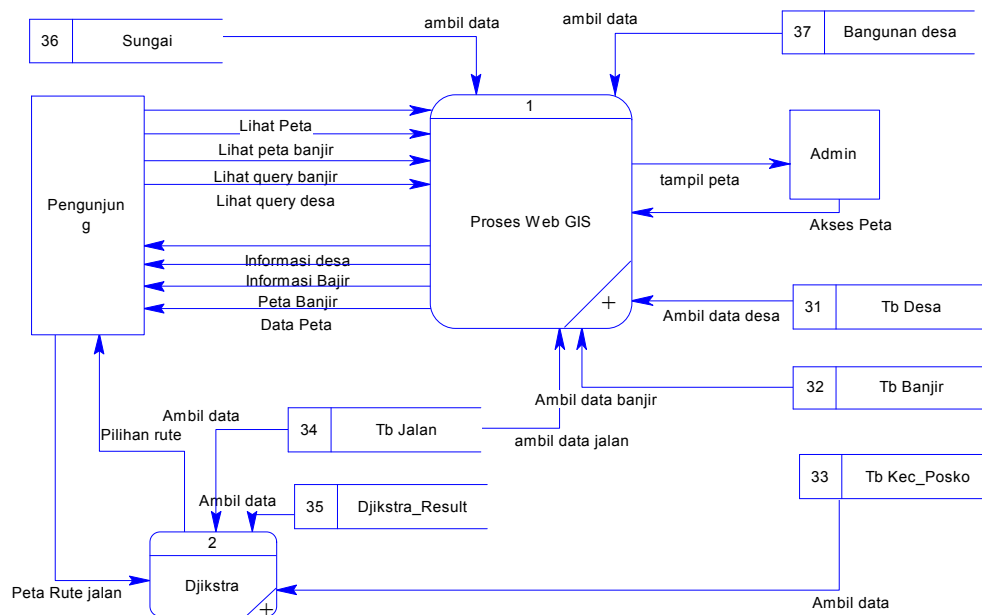
Gambar 3.7 DFD Level 2 Proses Informasi

Pada DFD level 2 proses Informasi ini menjelaskan tentang proses sebuah informasi. Pengunjung berhak melihat berita, pengumuman, agenda, sementara ketika pengunjung memilih salah satu menu itu, maka sistem akan memproses dan mengambil store di databases yang telah disediakan. Sementara untuk menu hubungi, pengunjung bisa mengirim pesan yang kemudian akan disimpan di databases. Kemudian setelah tersimpan admin akan menanggapi pesan dari pengunjung dengan mengambil data yang sudah tersimpan di databases.



Gambar 3.8DFD Level 2 Proses Administrator

Pada DFD level 2 proses Administrator dibagi menjadi empat subsistem manajemen informasi, manajemen modul, manajemen peta, manajemen user. Subsistem Manajemen informasi menjelaskan admin dan user dapat memanajemen beberapa menu, mulai dari berita, agenda, pengumuman, dll. Dan subsistem manajemen modul digunakan untuk mengatur menu apa yang dapat diakses oleh pengunjung dan user. Kemudian manajemen peta menjelaskan tentang bagaimana meng-update data banjir dan data informasi desa yang nantinya akan digunakan dalam proses query. Subsistem selanjutnya, manajemen user digunakan untuk proses pengurangan dan penambahan user oleh admin.



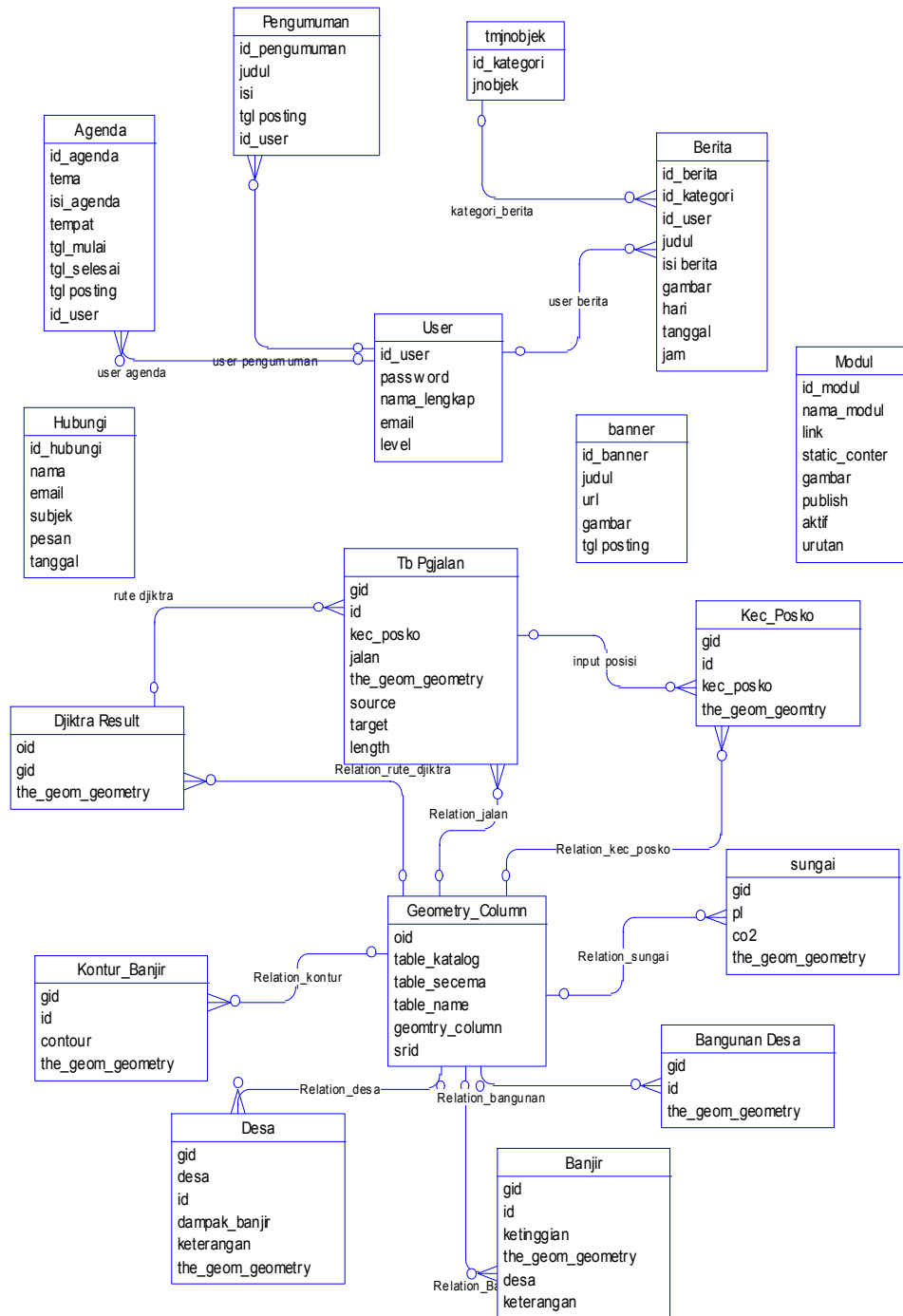
Gambar 3.9 DFD Level 2 Proses Peta

Pada DFD level 2 proses peta SIG Banjir ini merupakan pengembangan dari proses peta yang terdapat pada DFD level 1. Disini terlihat bahwa dalam proses peta, admin dapat mengakes peta (meng-update data desa dan banjir). Sedangkan pengguna dapat melihat peta jalan, dan jalur terpendek (dengan memilih jalan asal dan tujuan untuk mendapatkan jalur terpendek), lihat peta banjir, lihat query desa dan banjir, dll.

3.4.3. *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) atau bisa disebut diagram E-R merupakan model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. Diagram E-R merupakan sebuah diagram yang menggambarkan hubungan/relasi antar *Entity*, diagram E-R lebih menekankan pada struktur dan hubungan antar data, berbeda dengan DFD yang merupakan model jaringan fungsi yang akan dilaksanakan oleh sistem.

Entity Relationship Diagram SIG Banjir dapat dilihat pada gambar berikut ini:



Gambar 3.10 Entity Relationship Diagram SIG Banjir

3.4.4. Tabel data

Tabel-tabel database yang akan dikelola dalam aplikasi ini dibuat melalui tahapan perancangan database yang dianalisis menggunakan *Data Flow Diagram* dan *Entity Relationship Diagram*. Aplikasi database yang digunakan dalam skripsi ini adalah *PostgreSQL*, file databasenya “sragen”. Berikut ini nama-nama table yang digunakan beserta *field-field* yang terdapat pada masing-masing tabel.

1. Table User

Tabel 3.2 User

Name	Data type	Not Null?	Primary key?
id_user	character varying(50)	Yes	Yes
password	character varying(50)	Yes	No
nama_lengkap	character varying(100)	Yes	No
email	character varying(100)	Yes	No
level	character varying(50)	Yes	No

Berfungsi sebagai alat penyimpanan data inputan user yang dilakukan oleh admin, dengan form inputan akan dijelaskan di BAB berikutnya.

2. Tabel Agenda

Tabel 3.3 Agenda

Name	Data type	Not Null?	Primary key?
id_agenda	integer	Yes	Yes
tema	character varying(100)	Yes	No
isi_agenda	text	Yes	No
tempat	character varying(100)	Yes	No
tgl_mulai	date	Yes	No
tgl_selesai	date	Yes	No
tgl_posting	date	Yes	No
id_user	character varying(50)	Yes	No

Berfungsi sebagai alat penyimpanan data inputan agenda yang dilakukan oleh admin, dengan form inputan agenda akan dijelaskan di BAB berikutnya.

3. Tabel Banner

Tabel 3.4 Banner

Name	Data type	Not Null?	Primary key?
id_banner	integer	Yes	Yes
judul	character varying(100)	Yes	No
url	character varying(100)	Yes	No
gambar	character varying(100)	Yes	No
tgl_posting	date	Yes	No

Digunakan untuk menyimpan data yang berhubungan dengan media promosi (banner). Dan fasilitas ini hanya bisa diakses oleh admin.

4. Tabel Hubungi

Tabel 3.5 Hubungi

Name	Data type	Not Null?	Primary key?
id_hubungi	integer	Yes	Yes
nama	character varying(50)	Yes	No
email	character varying(100)	Yes	No
subjek	character varying(100)	Yes	No
pesan	text	Yes	No
tanggal	date	Yes	No

Dan tabel ini berguna untuk tempat penyimpanan pesan yang dikirim oleh pengunjung ke pengelola yang disini diartikan sebagai admin.

5. Tabel Berita

Tabel 3.6 Berita

Name	Data type	Not Null?	Primary key?
id_berita	integer	Yes	Yes
id_kategori	integer	Yes	No
id_user	character varying(50)	Yes	No
judul	text	Yes	No
isi_berita	text	Yes	No
gambar	character varying(100)	Yes	No
hari	character varying(20)	Yes	No
tanggal	date	Yes	No
jam	time without time zone	Yes	No
counter	integer	Yes	No
direktori	character varying(100)	Yes	No

Tabel berita berguna untuk menyimpan data-data yang berupa text dan gambar, dan fasilitas ini digunakan oleh admin.

6. Tabel Modul

Tabel 3.7 Modul

Name	Data type	Not Null?	Primary key?
id_modul	integer	Yes	Yes
nama_modul	character varying(50)	Yes	No
link	character varying(100)	Yes	No
static_content	text	No	No
gambar	character varying(100)	No	No
publish	character varying(10)	Yes	No
status	character varying(10)	Yes	No
aktif	character varying(10)	Yes	No
urutan	integer	Yes	No

Modul digunakan untuk mengelola menu-menu yang tampil di halaman pengunjung maupun user. Dan tabel ini berfungsi untuk menyimpan data-datanya, agar bisa dikelola oleh admin.

7. Tabel Pengumuman

Table 3.8 Pengumuman

Name	Data type	Not Null?	Primary key?
id_pengumuman	integer	Yes	Yes
judul	character varying(100)	Yes	No
isi	text	Yes	No
tanggal	date	Yes	No
tgl_posting	date	Yes	No
id_user	character varying(50)	Yes	No

Digunakan untuk menyimpan data pengumuman oleh admin seperti halnya tabel agenda dan berita.

8. Tabel Kategori

Table 3.9 Kategori

Name	Data type	Not Null?	Primary key?
id_kategori	integer	Yes	Yes
jnobjek	character varying(25)	Yes	No

Dan tabel ini mempunyai relasi dengan tabel berita, karena setiap pengisian berita diperintahkan untuk memilih salah satu kategori yang disediakan.

9. Tabel Dijkstra Result

Table 3.10 Dijkstra Result

Name	Data type	Not Null?	Primary key?
gid	integer	Yes	Yes
The_geom	geometry	NO	No

Tabel ini sangat penting, karena aplikasi pencarian rute terpendek dengan algoritma Dijkstra tidak akan bisa berjalan.

10. Tabel Jalan

Table 3.11 Jalan

Name	Data type	Not Null?	Primary key?
gid	integer	Yes	Yes
Id	integer	Yes	No
kec_posko	character varying(50)	Yes	No
jalan	character varying(50)	Yes	No
the_geom	geometry	No	No
source	integer	Yes	No
target	integer	Yes	No
length	Double precision	Yes	No

Tabel ini berisikan tentang data jalan, yang nantinya akan ditampilkan di peta, dan juga tabel ini sebagai penunjang dari aplikasi pencarian rute terpendek.

11. Tabel Kec/Posko

Table 3.12 Kec/posko

Name	Data type	Not Null?	Primary key?
gid	integer	Yes	Yes
Id	integer	Yes	No
kec_posko	character varying(50)	Yes	No
the_geom	geometry	No	No

Dalam nyatanya tabel ini berisi tentang data spasial, yang dikodekan oleh PostGis dengan tipe data point.

12. Tabel Kontur

Table 3.13 Kontur

Name	Data type	Not Null?	Primary key?
gid	integer	Yes	Yes
ld	integer	Yes	No
countur	Double precision	Yes	No
the_geom	geometry	No	No

Tabel ini berisi tentang data spasial bertipe line (garis), yang merupakan turunan dari data ketinggian.

13. Tabel Geometry Column

Table 3.14 Geometry Column

Name	Data type	Not Null?	Primary key?
f_table_catalog	character varying(256)	Yes	Yes
f_table_schema	character varying(256)	Yes	No
f_table_name	character varying(256)	Yes	No
f_geometry_column	character varying(256)	Yes	No
coord_dimension	Integer	Yes	No
srid	Integer	Yes	No
type	character varying(30)	Yes	No

Tabel ini merupakan tabel index dari semua data spasial, mulai yang bertipe point, line, dan juga polygon. Dengan relasi the geomnya.

14. Tabel Sungai

Table 3.15 Sungai

Name	Data type	Not Null?	Primary key?
gid	integer	Yes	Yes
pl	character varying(16)	Yes	No
col2	integer	Yes	No
the_geom	geometry	No	No

Berfungsi sebagai tempat penyimpanan data spasial sungai bertipe polygon.

15. Tabel Desa

Table 3.16 Desa

Name	Data type	Not Null?	Primary key?
gid	integer	Yes	Yes
count	integer	Yes	No
desa	character varying(22)	Yes	No
kecamatan	character varying(16)	Yes	No
kabupaten	character varying(13)	Yes	No
propinsi	character varying(19)	Yes	No
the_geom	geometry	No	No
dampak_banjir	text	No	No
keterangan	text	No	No
id	integer	Yes	No

Tabel ini mempunyai dua fungsi, fungsi pertama seperti halnya tabel peta yang lainnya, berisi tentang data spasial bertipe polygon. Fungsi yang kedua sebagai tempat penyimpanan bagi aktifitas inputan yang dilakukan oleh admin dan user yang nantinya berguna bagi proses query.

16. Tabel Banjir

Table 3.17 Banjir

Name	Data type	Not Null?	Primary key?
gid	integer	Yes	Yes
id	integer	Yes	No
ketinggian	character varying(16)	Yes	No
the_geom	geometry	No	No
desa	text	No	No
keterangan	text	No	No

Fungsinya tidak jauh berbeda dengan fungsi tabel desa, berfungsi sebagai penyimpan data spasial dan berfungsi sebagai tempat penyimpanan inputan yang dilakukan oleh admin.

17. Tabel Bangunan

Table 3.18 Bangunan

Name	Data type	Not Null?	Primary key?
gid	integer	Yes	Yes
Id	integer	Yes	No
keterangan	character varying(25)	Yes	No
the_geom	geometry	No	No

Berfungsi untuk menyimpan data spasial bangunan sragen bertipe line.

BAB IV

HASIL DAN PEMBAHASAN

4.1. Implementasi

Proses pembangunan komponen-komponen pokok sebuah sistem informasi yang sudah didesain perlu dibuat sebuah implementasi. Karena implementasi digunakan sebagai tolak ukur/pengujian dan analisa hasil dari program yang telah dibuat. Implementasi sistem juga merupakan sebuah proses pembuatan dan penerapan sistem secara utuh baik dari sisi perangkat keras maupun perangkat lunaknya. Dan juga tujuan dari pengujian ini adalah untuk mengetahui apakah aplikasi yang telah dibuat sesuai dengan perancangannya. Selain itu juga untuk mengetahui detail jalannya aplikasi serta kesalahan yang ada untuk pengembangan dan perbaikan lebih lanjut. Sementara itu, pada tahap ini juga dilakukan langkah persiapan sumber daya manusia dari yang menjalankan system tersebut disamping perangkat keras dan perangkat lunaknya. Pada bab ini akan dibahas juga hal-hal yang berkaitan dengan Web Sistem Informasi Geografis Simulasi Banjir.

Adapun implementasi yang akan dijelaskan disini meliputi lingkungan perangkat keras dan lingkungan perangkat lunak.

4.1.1 Ruang Lingkup Perangkat Keras

Dalam pengembangan SIG Simulasi Banjir di Kabupaten Sragen ini menggunakan perangkat keras Laptop Compaq Presario CQ40 dengan spesifikasi sebagai berikut:

1. Prosesor AMD Turion(tm) X2 Dual-Core Mobile RM-75, 2.2 GHz
2. RAM 1790 MB
3. Hardisk Dengan Kapasitas 320 GB
4. Monitor 14.1" High-definition widescreen display with BrightView technology dengan resolusi 1280 x 800 pixels.
5. Keypad
6. Mouse USB

4.1.2 Ruang Lingkup Perangkat Lunak

Perangkat lunak yang digunakan antara lain :

1. Sistem Operasi Windows XP
2. MapSever 4.4.2
3. PHP 4.3.11
4. Apache 2.0.54
5. PostgreSQL 8.2.5
6. ArcView 3.2
7. PgRouting-1.02_pg-8.2.
8. Macromedia Dreamweaver MX

4.1.3 Implementasi Database

Pada bab sebelumnya telah dijelaskan rancangan tabel dan relasi data yang terdiri dari beberapa tabel. Jenis kolom dan fungsinya sudah di jelaskan pada bab III. Tabel-tabel tersebut beserta relasinya diimplementasikan dengan menggunakan postgresQL versi 8.2.

Data yang berhubungan dengan pemetaan merupakan hasil konversi dari data tipe SHP yang berasal dari Arc View ke tag SQL dengan bantuan aplikasi shp2pgsql. Data peta dari ArcView tersebut minimal terdiri dari tiga data peta yaitu *.shp, *.shx, dan *.dbf. Sehingga data dari ArcView akan tersimpan dalam bentuk taq SQL dan semua data peta yang beripe poin, line dan polygon akan di rubah kedalam format WKB(Well Known Binary). Proser konversi data melalui *command prompt* dengan cara masuk terlebih dahulu ke folder “bin” pada PosgreSQL yang ada di program file dan sebelum di konversi copy terlebih dahulu file peta dari *ArcView* ke folder bin, perintah selengkapnya sebagai berikut.

```
C:\Program Files\PostgreSQL\8.2\bin>_shp2pgsql.exe Peta tmPeta > Peta.sql
```

Penjelasan perintah:

```
>_shp2pgsql.exe Peta tmPeta > Peta.sql
```

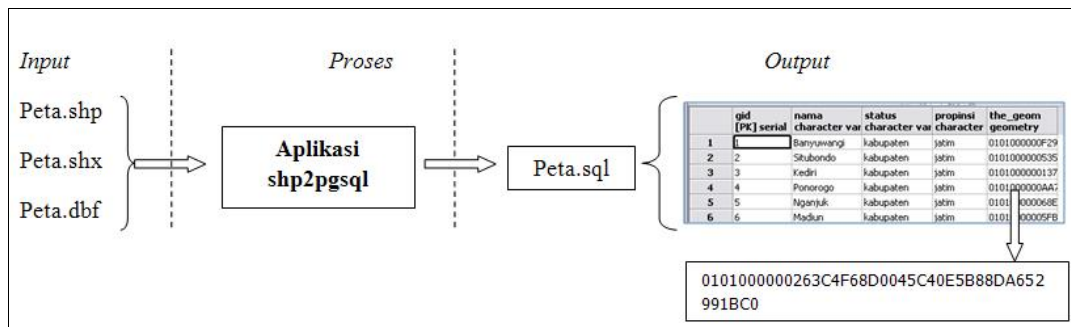
shp2pgsql.exe : Aplikasi PgSql yang digunakan untuk mengkonfersi file *.shp → sql, yang letaknya berada di “*PosgreSQL\8.2\bin* >”

Peta : nama file *.shp pada perintah diatas data *.shp dan pendukungnya (*.dbf, *.shx) sudah berada pada folde \bin > sehingga tidak perlu menuliskan nama filenya. Dan jika

file *.shp berada di lain folder maka alamatnya harus ditulis secara lengkap: C:\Jatim\Peta.shp .

tmPeta : nama tabel yang akan dibuat di database

Peta.sql : nama file hasil konfersian yang ber ekstensi .sql. secara default akan tersimpan di folder \bin >. Jika ingin menyimpan di tempat yang lain maka alamat penyimpanan harus ditlis secara lengkap, sehingga menjadi :
C:\Jatim\Peta.sql



Gambar 4.1 Diagram Konversi Data

4.1.4 Implementasi Antarmuka

Implementasi antar muka pada aplikasi ini dibagi menjadi 3 bagian yaitu antar muka admin, user, dan pengunjung.

a. Halaman Utama Admin

Halaman ini hanya bisa diakses oleh administrator dengan cara harus login, karena administrator mempunyai kewenangan atas manajemen

web dengan memanfaatkan menu-menu yang telah disediakan. Menu-menu yang disediakan dalam halaman ini antara lain:

1. Manajemen User

Di sini admin dapat menambah mengedit dan menghapus user.

2. Manajemen Modul

Dihalaman ini admin dapat mengatur menu apa saja yang akan ditampilkan dan mengatur hak akses di setiap menu.

3. Profil lembaga

Pada halaman ini disediakan form untuk mengelola profil lembaga yang bersangkutan.

4. Berita

Di halaman ini admin dapat melihat semua berita yang di masukkan oleh user dan dapat menambah, mengedit, serta menghapus semua berita.

5. Agenda dan pengumuman

Di halaman ini admin dapat melihat semua agenda atau berita di setiap objek wisata yang di masukkan oleh user dan dapat menambah, mengedit, serta menghapus semua berita.

6. Banner

Halaman banner digunakan untuk menampilkan iklan atau sponsor yang berbentuk gambar yang memberi sponsor atau iklan.

7. Kontak

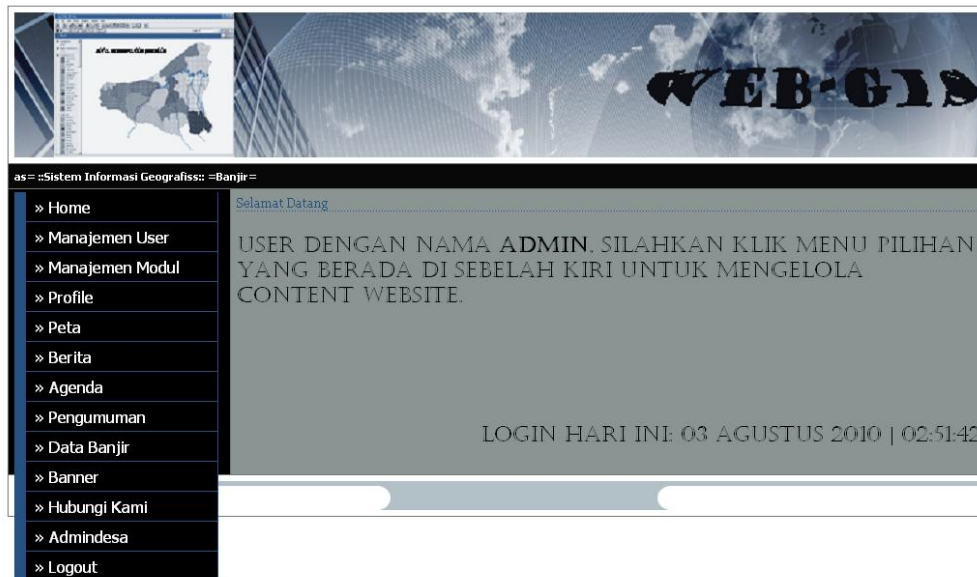
Halaman ini digunakan untuk menerima dan membalas pertanyaan atau saran dari user dan pengunjung.

8. Data Banjir

Bagian ini digunakan untuk mengupdate data-data yang berhubungan dengan banjir.

9. Data Desa

Halaman ini digunakan untuk mengupdate data (informasi yang berkaitan dengan desa), bisa dampak banjir bagi desa tertentu, yang nantinya bisa dilihat di halaman pengunjung.



Gambar 4.2 Halaman Utama Admin

b. Halaman Utama User

Dihalaman ini menu yang disediakan antara lain:

- 1) Berita

Di dalam ini admin dapat melihat semua berita yang di masukkan oleh user dan dapat menambah, mengedit, serta menghapus semua berita.

2) Agenda dan pengumuman

Di dalam ini admin dapat melihat semua agenda atau pengumuman di setiap objek wisata yang di masukkan oleh user dan dapat menambah, mengedit, serta menghapus semua agenda dan pengumuman.

3) Banner

Halaman banner digunakan untuk menampilkan iklan atau sponsor yang berbentuk yang memberi sponsor atau iklan.

4) Kontak

Halaman ini digunakan untuk menerima dan membalas pertanyaan atau saran dari user dan pengunjung.

5) Data Banjir

Bagian ini digunakan untuk mengupdate data-data yang berhubungan dengan banjir.

6) Data Desa

Halaman ini digunakan untuk mengupdate data (informasi yang berkaitan dengan desa), bisa dampak banjir bagi desa tertentu, yang nantinya bisa dilihat di halaman pengunjung.



Gambar 4.3 Halaman Utama User

Kode program dari interface di atas sebagai berikut:

```

<?php
session_start();
if (empty($_SESSION[namauser]) AND empty($_SESSION[passuser])){
    echo "<link href='../config/adminstyle.css' rel='stylesheet'
type='text/css'>
    <center>Untuk mengakses modul, Anda harus login <br>";
    echo "<a href=../index.php><b>LOGIN</b></a></center>";
    }
else{
    ?>

<html>
<head>
<title>"<!--Flood-->":</title>
<link href="config/style.css" rel="stylesheet" type="text/css">
<link href=../GaphLib/adminstyle.css" rel="stylesheet"
type="text/css" />
<meta http-equiv="Content-Type" content="text/html; charset=iso-
8859-1">
</head>

<body>
<div align="center">

    <table width="999" border="1" align="center" cellpadding="0"

```

```

cellspacing="0">
  <tr>
    <td colspan="3"></td>
  </tr>
  <tr bgcolor="#080808">
    <td colspan="2" bordercolor="#FFFFFF"> <marquee>
      <strong><font color="#FFFFFF">=Murdianimas=      ::Sistem
Informasi Geografiss::
      =Banjir=</font></strong> </marquee></td>
    </tr>
  <tr>
    <td width="220" height="25" align="center" valign="top"
bordercolor="#FFFFFF" bgcolor="#233333">
      <p>
        <div id="menu">
          <ul>
            <li><a href=?module=home>&#187; Home</a></li>
            <?php include "menu.php"; ?>
            <li><a href=logout.php>&#187; Logout</a></li>
          </ul>
          <p>&nbsp;</p>
        </div></p>
      </td>
      <td width="777" valign="top" bordercolor="#FFFFFF"
bgcolor="#999999"> <div id="content">
        <?php include "content.php"; ?>
      </div></td>
    </tr>
  <tr>
    <td colspan="2" bordercolor="#FFFFFF" bgcolor="#FFFFFF"></td>
  </tr>
</table>
</div>
</body>
</html>
<?
}
?>

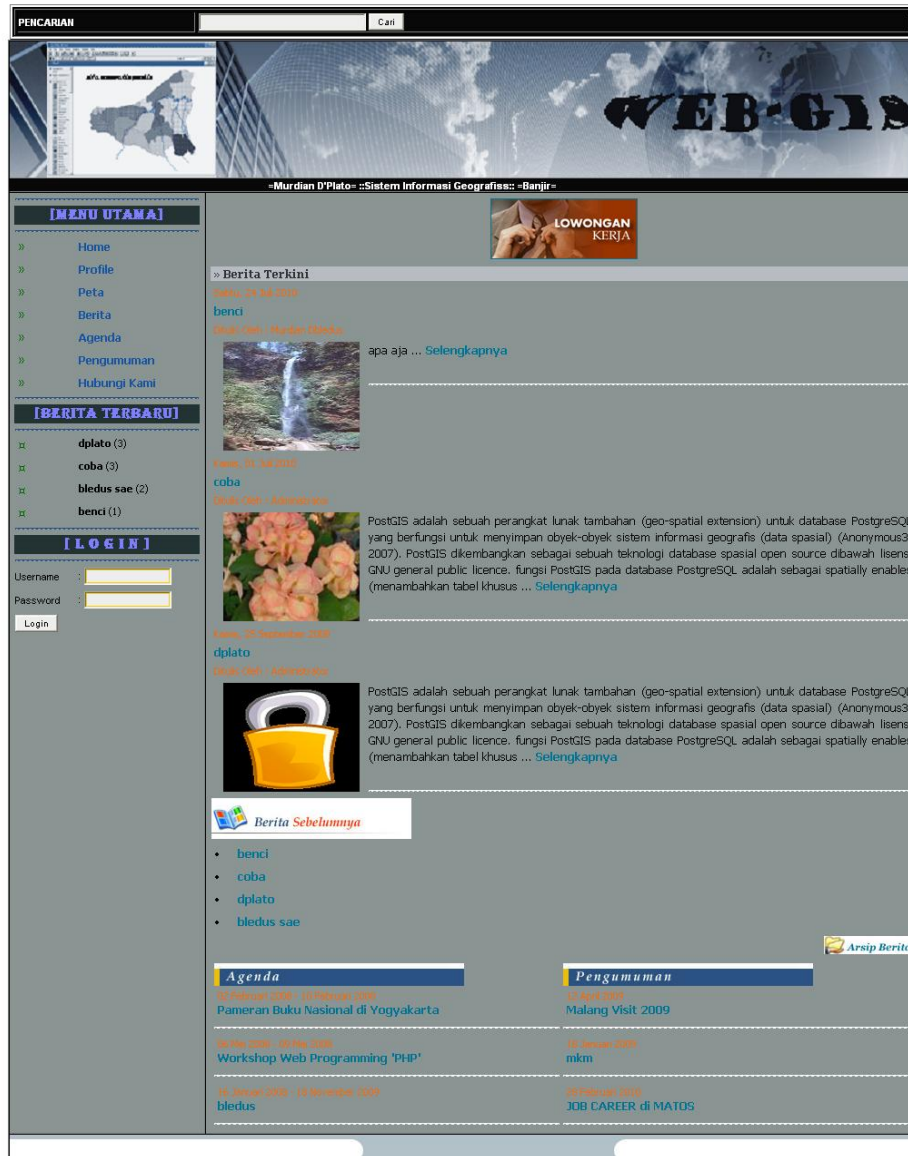
```

c. Halaman Utama Pengunjung

Di halaman ini user hanya dapat mengakses menu-menu yang telah disediakan oleh admin, menu menu tersebut antara lain:

1. Home
2. Profile
3. Peta

4. Berita
5. Agenda
6. Pengumuman
7. Hubungi Kami



Gambar 4.4 Halaman Utama Pengunjung

Kode program dari menu utama yang diambil dari file layout.php:

```

<html>
<head>
<title>"<!--Flood-->":</title>
<link href="config/style.css" rel="stylesheet"
type="text/css">
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
</head>
<style type="text/css">
P
{
font-family: Arial, Helvetica, sans-serif;
font-size: 8pt;
color: #666666;
}

TD
{
font-family: Arial, Helvetica, sans-serif;
font-size: 8pt;
}

.forminputsmall {
text-decoration: none;
font-family: Arial, Helvetica, sans-serif;
font-size: 6pt;
font-weight : normal;
}

.tinytext {
font-size: 6pt;
font-family: Arial, Helvetica, sans-serif;
}
</style>
<body>
<table width="999" border="1" align="center" >
<tr bgcolor="#080808">
<td>
<?php
include"huhcar.php";
?>
</td>
</tr>
</table>
<table width="999" border="1" align="center" cellpadding="0"
cellspacing="0">
<tr>
<td colspan="3"></td>
</tr>
<tr bgcolor="#080808">
<td colspan="2" bordercolor="#FFFFFF">
<marquee>
<strong><font color="#FFFFFF">=Murdian D'Plato=
::Sistem Informasi Geografiss::
=Banjir=</font></strong>

```



```

        </marquee></td>
    </tr>
    <tr>
        <td width="214" valign="top" height="25"
bordercolor="#FFFFFF" bgcolor="#8A9593">
        <p>
            <? include "kiri.php"; ?>
        </p></td>
        <td width="783" valign="top" bordercolor="#FFFFFF"
bgcolor="#8A9593">
            <? include "kanan.php"; ?>
        </td>
    </tr>
    <tr>
        <td colspan="2" bordercolor="#FFFFFF"
bgcolor="#FFFFFF"></td>
    </tr>
</table>
<div align="center"></div>
</body>
</html>

```

4.2 Pembahasan

Pada bagian ini akan dijelaskan fungsi-fungsi dari form yang ada di sistem beserta kode program proses.

4.2.1 Input Data

a. Form Pencarian

Form pencarian berada pada halaman muka(home), hanya bisa mencari kata di berita, dengan cara mengetikan kata yang akan dicari kemudian klik cari atau tekan enter.



Gambar 4.5 Form Pencarian

Dengan kode prog ...

```
// Bagian Hasil Pencarian
elseif ($_GET[module]=='hasilcari'){
    echo "
|  |
| --- |
|  |

";

    $cari = pg_query("SELECT * FROM berita WHERE isi_berita LIKE '%$_POST[kata]%'");
    $jumlah = pg_num_rows($cari);

    if ($jumlah > 0){
        echo "
| <a href=?module=detailberita&id=$r[id_berita]>$r[judul]</a></li>";         }         echo " </ul></td></tr>";     }     else{         echo " |  |  | | --- | --- | | | | |

```

b. Form Hubungi kami

Form ini merupakan fasilitas pengunjung, digunakan pengunjung memberikan pesan kepada pengelola. Dengan cara mengisi semua form yang telah disediakan.

Gambar 4.6 Form Hubungi Kami

Dengan potongan kode program proses pengiriman hasil inputan ke database yang nantinya akan digunakan oleh admin.

```

elseif ($_GET[module]=='kirimemail'){
    pg_query("INSERT INTO hubungi(nama,
                                email,
                                subjek,
                                pesan,
                                tanggal)
    VALUES ('$_POST[nama]',
            '$_POST[email]',
            '$_POST[subjek]',
            '$_POST[pesan]',
            '$tgl_sekarang')");

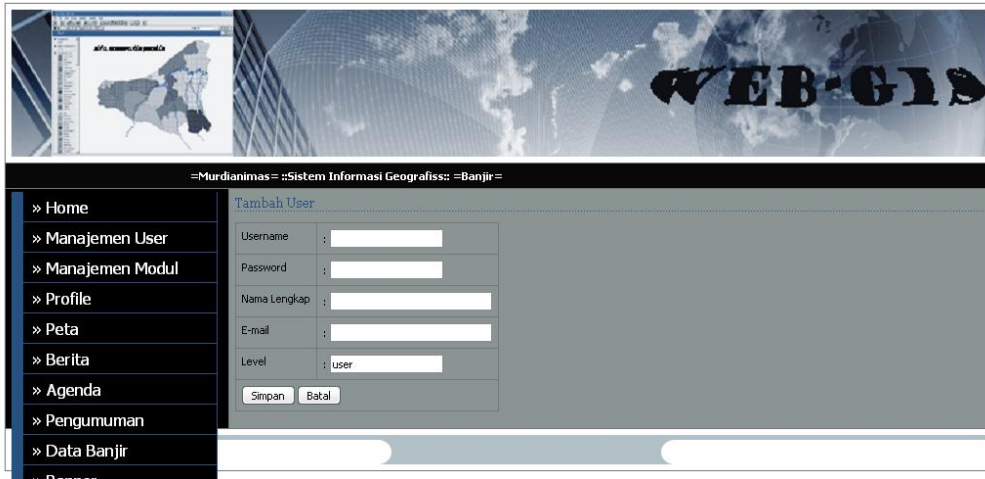
    echo "<tr><td class=judul head>&#187; Status Email</td></tr>";
}

```

```
<tr><td class=isi>Email telah sukses terkirim dan segera akan kami balas</td></tr>
<tr><td class=kembali><br>
[ <a href=javascript:history.go(-1)>Kembali</a>
]</td></tr>";
}
```

c. Form Tambah User

Dalam form ini administrator dapat menambahkan user yang akan diberi hak akses untuk dapat login dan input data



Gambar 4.7 Form Tambah User

Kode program dari proses di atas adalah:

```

// Input user
elseif ($module=='user' AND $act=='input'){
    $pass=md5($_POST[password]);
    pg_query("INSERT INTO user1(id_user,
                                password,
                                nama_lengkap,
                                email,
                                level)
            VALUES('$_POST[id_user]',
                    '$pass',
                    '$_POST[nama_lengkap]',
                    '$_POST[email]',
                    '$_POST[level]')");
    header('location:media.php?module=user');
}

```

d. Form Tambah modul

Pada form ini administrator dapat menambah dan menampilkan menu yang dapat di akses oleh user dan pengunjung. Penjelasan dari form tambah modul adalah sebagai berikut:

1. Nama modul merupakan nama atau label dari menu yang akan ditampilkan.
2. Link textfield ini digunakan untuk memasukkan alamat link yang akan dituju.
3. Publish, apabila dipilih radio button Y maka link akan di tampilkan di menu pengunjung dan user jika N maka sebaliknya.
4. Aktif, menu akan aktif bila radio button Y di pilih, jika N maka sebaliknya.
5. Status, hal ini memberiketerangan apakah menu dapat diakses oleh user apa tidak.



Gambar 4.8 Form Tambah Modul

Dengan kode program input modul sebagai berikut:

```
// Input modul
elseif ($module=='modul' AND $act=='input'){
    pg_query("INSERT INTO modul (nama_modul,
                                link,
                                publish,
                                aktif,
                                status,
                                urutan)
VALUES ('$_POST[nama_modul]',
                                '$_POST[link]',
                                '$_POST[publish]',
                                '$_POST[aktif]',
                                '$_POST[status]',
                                '$_POST[urutan]')");
    header('location:media.php?module='.$module);
}
```

e. Form Tambah berita

Form ini merupakan fasilitasnya pengelola Website alias admin. Cara penggunaanya bisa diketik secara langsung ataupun bisa copy/paste. Dan pada bagian gambar pilih chose kemudian ambil gambar yang sudah disediakan di folder gambar.



Gambar 4.9 Form Tambah Berita

Dengan kode program input berita sebagai berikut:

```
// Input berita
elseif ($module=='berita' AND $act=='input'){
    $lokasi_file = $_FILES['fupload']['tmp_name'];
    $nama_file = $_FILES['fupload']['name'];

    // Apabila ada gambar yang diupload
    if (!empty($lokasi_file)){
        move_uploaded_file($lokasi_file,"foto_berita/$nama_file");
        pg_query("INSERT INTO berita(judul,
                                id_kategori,
                                isi_berita,
                                id_user,
                                jam,
                                tanggal,
                                hari,
                                gambar)
                VALUES ('$_POST[judul]',
                        '$_POST[kategori]',
                        '$_POST[isi_berita]',
                        '$_SESSION[namauser]',
                        '$jam_sekarang',
                        '$tgl_sekarang',
                        '$hari_ini',
                        '$nama_file')");
    }
}
```

```
else{
    pg_query("INSERT INTO berita(judul,
                                id_kategori,
                                isi_berita,
                                id_user,
                                jam,
                                tanggal,
                                hari)
            VALUES ('$_POST[judul]',
                    '$_POST[kategori]',
                    '$_POST[isi_berita]',
                    '$_SESSION[namauser]',
                    '$jam_sekarang',
                    '$tgl_sekarang',
                    '$hari_ini')");
}
header('location:media.php?module='.$_module);
}
```

f. Form Tambah agenda

Merupakan fasilitas yang digunakan oleh admin, yang gunanya untuk mengupdate agenda yang nanti akan bisa dilihat di sisi pengunjung.



Gambar 4.10 Form Tambah Agenda

Adapun kode program untuk input agenda adalah sebagai berikut:

```
// Input agenda
elseif ($module=='agenda' AND $act=='input'){

    $mulai=sprintf("%02d%02d%02d",$_POST[thn_mulai],$_POST[bln_m
ulai],$_POST[tgl_mulai]);

    $selesai=sprintf("%02d%02d%02d",$_POST[thn_selesai],$_POST[b
ln_selesai],$_POST[tgl_selesai]);

    pg_query("INSERT INTO agenda (tema,
                                isi_agenda,
                                tempat,
                                tgl_mulai,
                                tgl_selesai,
                                tgl_posting,
                                id_user)

VALUES ('$_POST[tema]',
                                '$_POST[isi_agenda]',
                                '$_POST[tempat]',
                                '$mulai',
                                '$selesai',
                                '$tgl_sekarang',
                                '$_SESSION[namauser]')");

    header ('location:media.php?module='.$module);
}
```

g. Form Tambah pengumuman

Merupakan fasilitas yang digunakan oleh admin, yang gunanya untuk mengupdate agenda yang nanti akan bisa dilihat di sisi pengunjung.



Gambar 4.11 Form Tambah Pengumuman

Dengan kode program sebagai berikut:

```
// Input pengumuman
elseif ($module=='pengumuman' AND $act=='input'){

    $tanggal=sprintf("%02d%02d%02d",$_POST[thn],$_POST[bln],$_POST[tgl]);

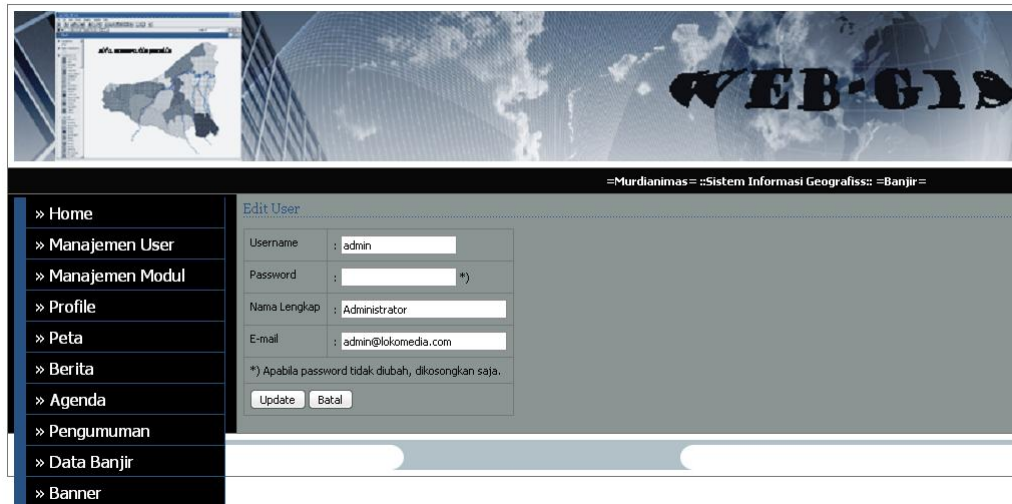
    pg_query("INSERT INTO pengumuman(judul,
                                     isi,
                                     tanggal,
                                     tgl_posting,
                                     id_user)

    VALUES ('$_POST[judul]',
            '$_POST[isi_pengumuman]',
            '$tanggal',
            '$tgl_sekarang',
            '$_SESSION[namauser]')");
    header('location:media.php?module='.$module);
}
```

4.2.2 Edit Data

a. Form Edit User

Form ini digunakan oleh admin untuk mengubah data user, dan form ini hanya bisa di akses oleh administrator.



Gambar 4.12 Form Edit User

Dengan kode program sebagai berikut:

```
// Update user
elseif ($module=='user' AND $act=='update'){
    // Apabila password tidak diubah
    if (empty($_POST[password])) {
        pg_query("UPDATE user1 SET id_user      = '$_POST[id_user]',
                                nama_lengkap  =
'$_POST[nama_lengkap]',
                                email        = '$_POST[email]'
                                WHERE id_user  = '$_POST[id]'");
    }
    // Apabila password diubah
    else{
        $pass=md5($_POST[password]);
        pg_query("UPDATE user1 SET id_user      = '$_POST[id_user]',
                                password      = '$pass',
                                nama_lengkap  =
'$_POST[nama_lengkap]',
                                email        = '$_POST[email]'
                                WHERE id_user  = '$_POST[id]'");
    }
    header('location:media.php?module='.$module);
}
```

b. Form Edit Modul

Form ini digunakan oleh admin jika ingin mengedit modul:

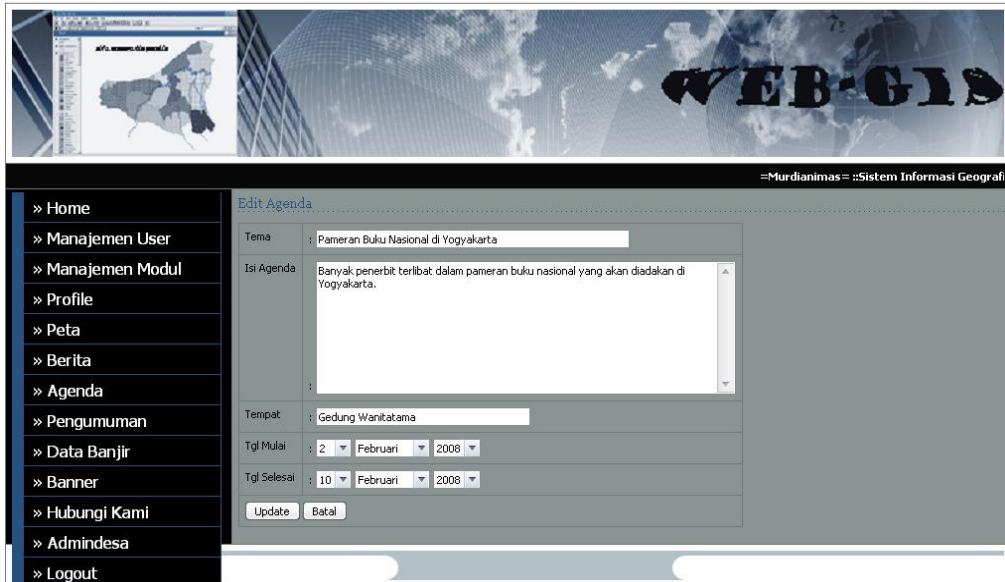
Gambar 4.13 Form Edit Modul

Dengan kode program sebagai berikut:

```
// Update modul
elseif ($module=='modul' AND $act=='update'){
    pg_query("UPDATE      modul      SET      nama_modul      =
'$_POST[nama_modul]',
                                link      = '$_POST[link]',
                                publish    =
'$_POST[publish]',
                                aktif     =
'$_POST[aktif]',
                                status    =
'$_POST[status]',
                                urutan    =
'$_POST[urutan]'
                                WHERE id_modul = '$_POST[id]');
    header('location:media.php?module='.$module);
}
```

c. Form Edit Agenda

Form ini tidak berbeda dengan Form-form yang lain digunakan oleh admin jika ingin mengubah/mengedit Agenda.



Gambar 4.14 Form Edit Agenda

Dengan kode program sebagai berikut:

```
// Update agenda
elseif ($module=='agenda' AND $act=='update'){

$mulai=sprintf("%02d%02d%02d",$_POST[thn_mulai],$_POST[bln_mulai],
$_POST[tgl_mulai]);

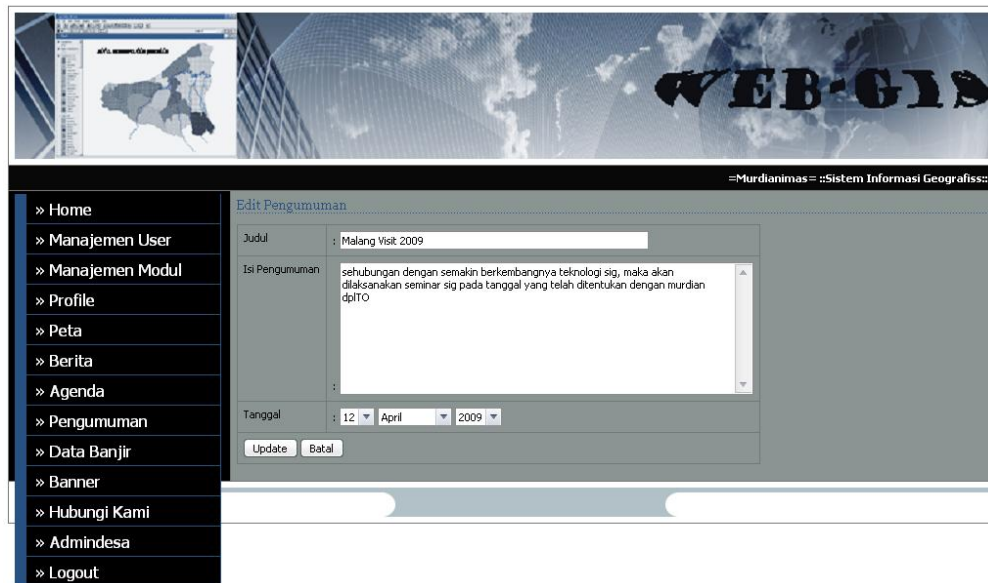
$selesai=sprintf("%02d%02d%02d",$_POST[thn_selesai],$_POST[bln_sel
esai],$_POST[tgl_selesai]);

    pg_query("UPDATE agenda SET tema = '$_POST[tema]',
                                                    isi_agenda
                                                    =
'$_POST[isi_agenda]',
                                                    tgl_mulai    = '$mulai',
                                                    tgl_selesai = '$selesai',
                                                    tempat      = '$_POST[tempat]'
                                                    WHERE id_agenda = '$_POST[id]'");

    header('location:media.php?module='.$module);
}
}
```

d. Form Edit Pengumuman

Fasilitas untuk mengedit pengumuman yang akan tampil di halaman depan.



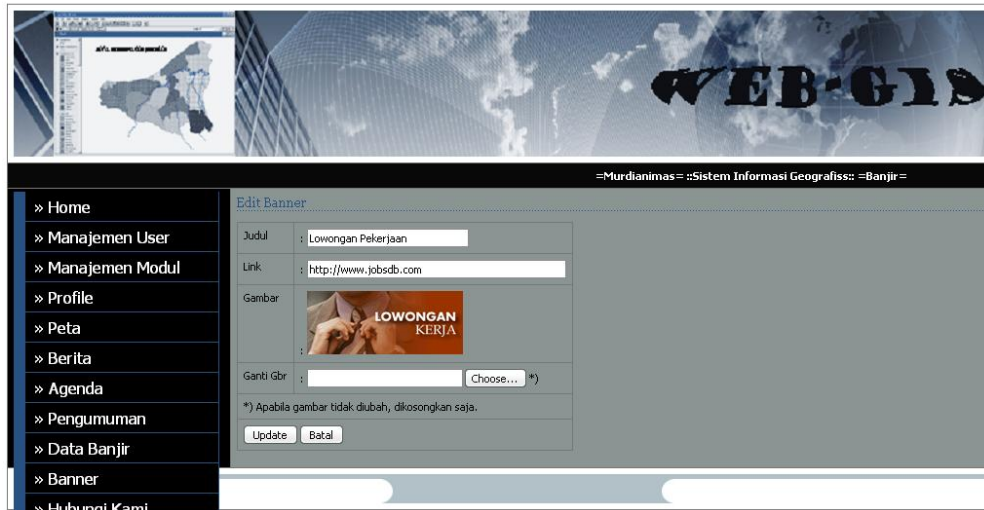
Gambar 4.15 Form Edit Pengumuman

Kode program update pengumuman:

```
elseif ($module=='pengumuman' AND $act=='update'){  
  
    $tanggal=sprintf ("%02d%02d%02d",$_POST[thn],$_POST[bln],$_POST[tgl]);  
  
    pg_query("UPDATE pengumuman SET judul           =  
    '$_POST[judul]',                               =  
    '$_POST[isi_pengumuman]',                     =  
    '$tanggal'                                     =  
    WHERE id_pengumuman =  
    '$_POST[id]'");  
    header ('location:media.php?module='.$module);  
}
```

e. Form Edit Banner

Digunakan untuk mengedit banner, yaitu sebuah gambar untuk media promosi.



Gambar 4.16 Form Edit Banner

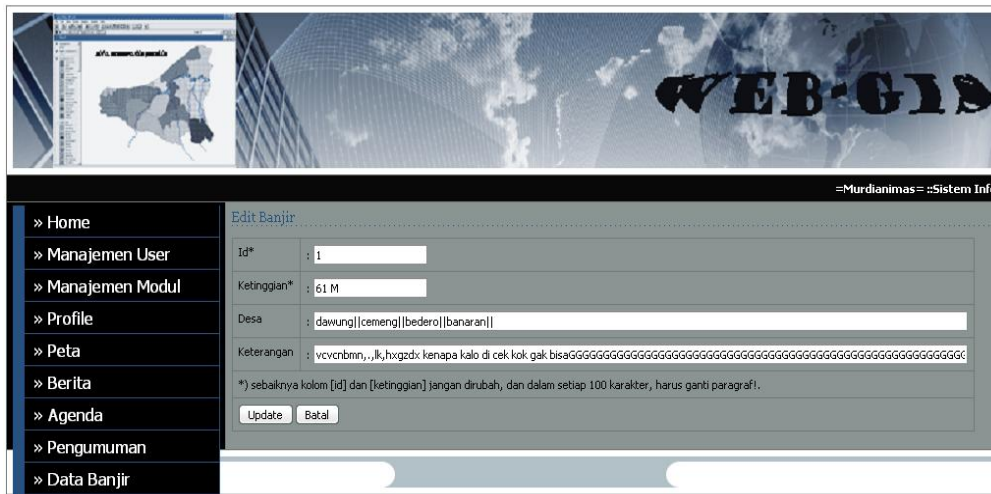
Dengan kode program sebagai berikut:

```
// Update banner
elseif ($module=='banner' AND $act=='update'){
    $lokasi_file = $_FILES['fupload']['tmp_name'];
    $nama_file = $_FILES['fupload']['name'];

    // Apabila gambar tidak diganti
    if (empty($lokasi_file)){
        pg_query("UPDATE banner SET judul = '$_POST[judul]',
            url = '$_POST[link]'
            WHERE id_banner = '$_POST[id]'");
    }
    else{
        move_uploaded_file($lokasi_file,"foto_berita/$nama_file");
        pg_query("UPDATE banner SET judul = '$_POST[judul]',
            url = '$_POST[link]',
            gambar = '$nama_file'
            WHERE id_banner = '$_POST[id]'");
    }
    header('location:media.php?module='.$module);
}
```

f. Edit Data Banjir

Dan fasilitas ini digunakan untuk mengedit Data-data tentang banjir, yang nantinya keterangan ini (atribut) digunakan dalam aplikasi spasialnya dipeta.



Gambar 4.17 Form Edit Data Banjir

Dengan skripnya sebagai berikut:

```
elseif ($module=='banjir' AND $act=='update'){
    pg_query("UPDATE bnjrtombol SET desa = '$_POST[desa]',
                                   keterangan
                                   =
                                   $_POST[keterangan]'
                                   WHERE gid = '$_POST[id]');
    header('location:media.php?module=.'.$module);
}
```

g. Edit Data Desa

Tidak jauh berbeda dengan fasilitas di atasnya, form ini digunakan untuk mengedit data desa yang nantinya akan digunakan di aplikasi spasialnya.

The screenshot shows a web application interface for editing village data. The page has a header with "WEB-GIS" and a navigation menu on the left. The main content area is titled "Edit Desa" and contains a form with the following fields and values:

Id*	: 1
Desa*	: Keden
Kecamatan	: Kali Jambe
Kabupaten	: Sragen
Provinsi	: Jawa Tengah
Dmpk_bnjr	: ridlo ilah
Keterangan	: 30 o 40 .40

At the bottom of the form, there are "Update" and "Batal" buttons. A note below the form states: "(*) seballmya, jika tidak tahu jangan dirubah, dan dalam setiap 100 karakter, harus ganti paragraf".

Gambar 4.18 Form Edit Data Desa

Dengan skript sederhana sebagai berikut:

```
//edit admindesa

elseif ($module=='admindesa' AND $act=='update'){
    pg_query("UPDATE pgadmindesa SET
        dampak_banjir = '$_POST[dampak_banjir]',

        keterangan      = '$_POST[keterangan]'
        WHERE gid      = '$_POST[id]'");
    header('location:media.php?module='.$module);
}
```

4.2.3 Tampilan Peta

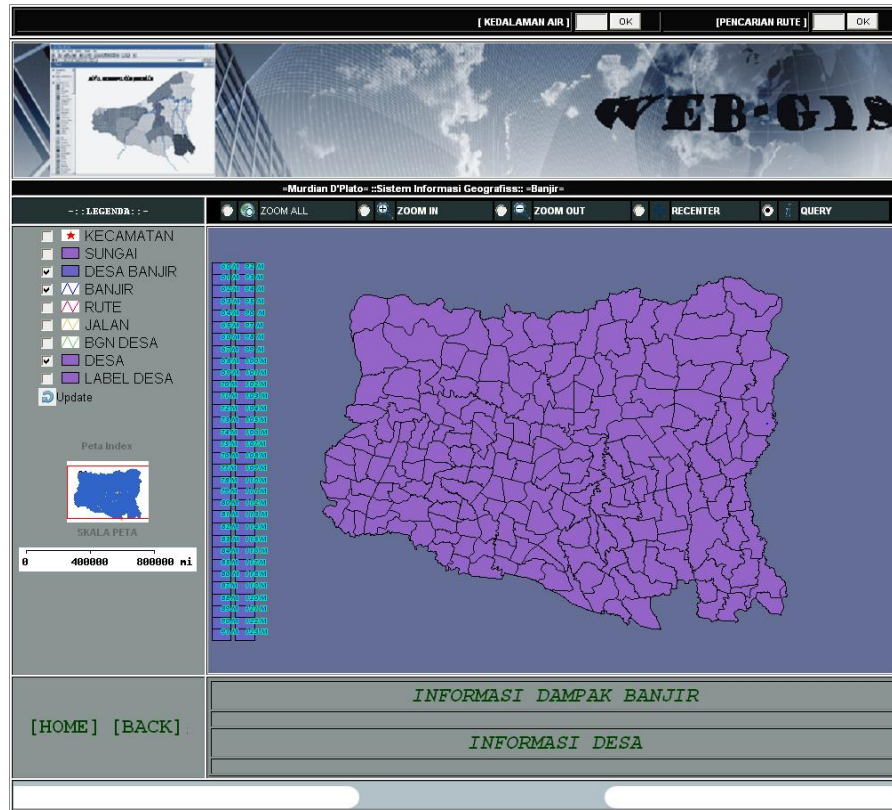
Sebelum menampilkan peta, pengunjung bisa memilih ketinggian banjir dan mencari rute jalan dalam keadaan banjir dengan mengisi form yang ada di atas halaman bagian kanan. Misal, pengunjung ingin mengetahui banjir dengan ketinggian air 85 meter, dan mencari rutanya dengan cara mengisi angka tersebut pada form yang telah disediakan.



Gambar 4.19 Form Input Kedalaman air dan Pencarian rute

Kemudian pada subbab ini akan ditampilkan peta, masing-masing tampilan yang mewakili keadaan banjir yang biasa, sedang, dan keadaan yang tinggi dan peta sragen tidak posisi banjir. Katergori biasa, sedang, dan tinggi tidak mewakili kesimpulan hasil analisis penelitian, hanya saja untuk mempermudah dalam hal penulisan laporan saja.

a. Tampilan Peta Tidak Banjir



Gambar 4.20 Tampilan Peta Tidak Banjir

Dengan skript interfacenya sebagai berikut:

```

<html>
<head>
<title>"<--Flood-->":</title>
<link href="etc/style.css" rel="stylesheet" type="text/css">
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1"></head>

<style type="text/css">

    P
    {
        font-family: Arial, Helvetica, sans-serif;
        font-size: 8pt;
        color: #666666;
    }

    TD
    {
        font-family: Arial, Helvetica, sans-serif;

```

```

        font-size: 8pt;
    }

    .forminputsmall {
        text-decoration: none;
        font-family: Arial, Helvetica, sans-serif;
        font-size: 6pt;
        font-weight : normal;
    }

    .tinytext {
        font-size: 6pt;
        font-family: Arial, Helvetica, sans-serif;
    }
}
</style>
<body>
<table width="999" border="1" align="center" >
  <tr bgcolor="#080808">
    <td>
      <?php
        include"huh2.php";
      ?>
    </td>
  </tr>
</table>
<form name="MAP" method="GET" >
  <table width="999" border="1" align="center" >
    <tr>
      <td colspan="3"></td>
    </tr>
    <tr>
      <td colspan="3" bordercolor="#FFFFFF"
bgcolor="#080808"> <marquee>
        <strong><font color="#FFFFFF">=Murdian D'Plato=
::Sistem Informasi Geografiss::
        =Banjir=</font></strong> </marquee></td>
    </tr>
    <tr>
      <td bgcolor="#233333" width="23%"><div
align="center"><strong><font color="#FFFFFF" face="Courier
New, Courier, mono">-::LEGENDA::-</font></strong></div></td>
      <td width="77%" bgcolor="#000000">
        <?
          include"zomzoman.php";
        ?>
      </td>
    </tr>
    <tr>
      <td align="center" valign="top" bgcolor="#8A9593">
        <?php DrawLegend(); ?>
        <p>&nbsp;</p>
        <table width="75%" border="0">
          <tr>
            <td align="center" valign="top"> <p><b>Peta
Index</b></p>

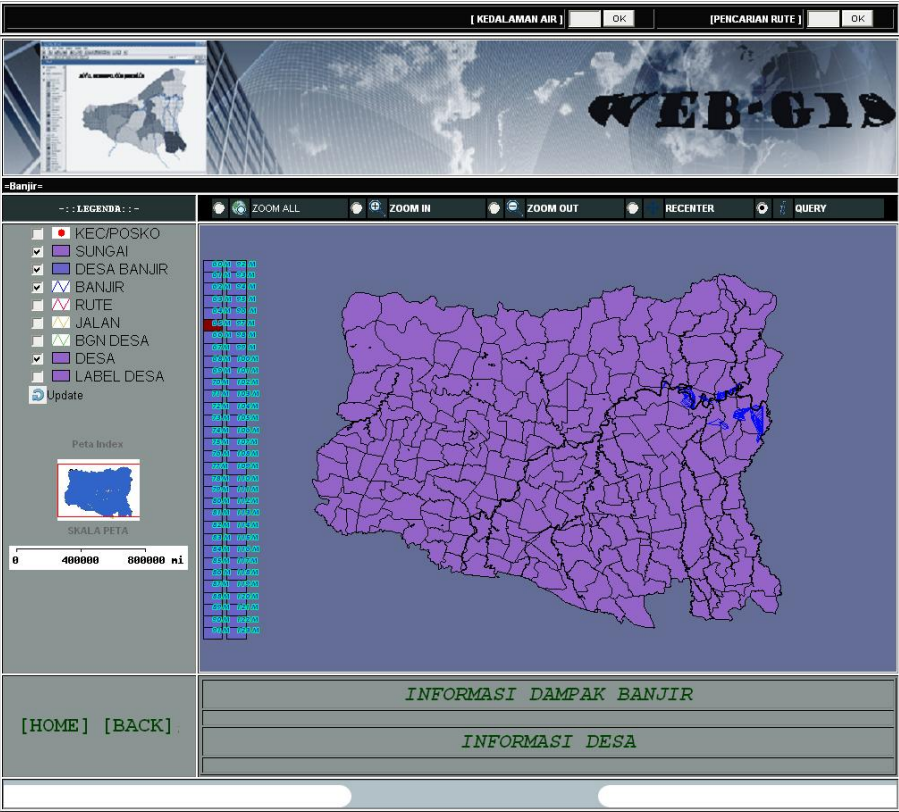
```



```
</tr>
</table>
</form>
</body>
</html>
```

b. Peta Keadaan Banjir Biasa

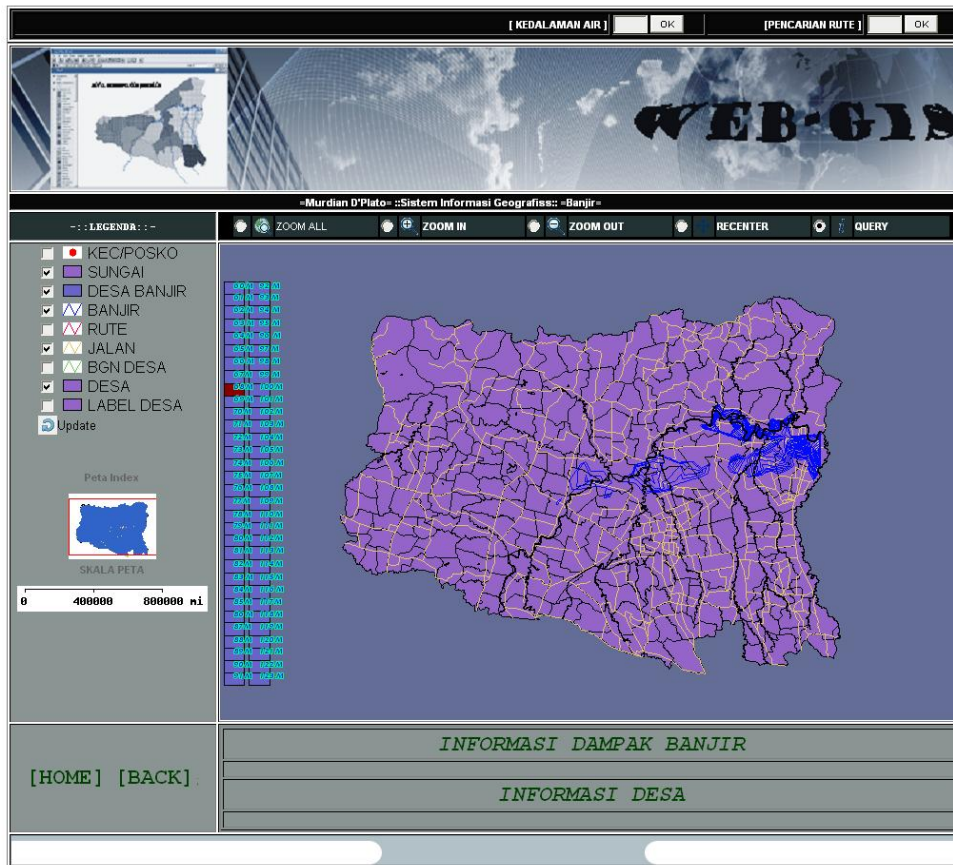
Maksud dari keadaan banjir biasa yaitu, masalah transportasi dan distribusi kedesa-desa tidak terganggu, jadi semua jalan bisa diakses dengan baik.



Gambar 4.21 Tampilan Peta Banjir Biasa (Ketinggian 65 meter)

c. Peta Keadaan Banjir Sedang

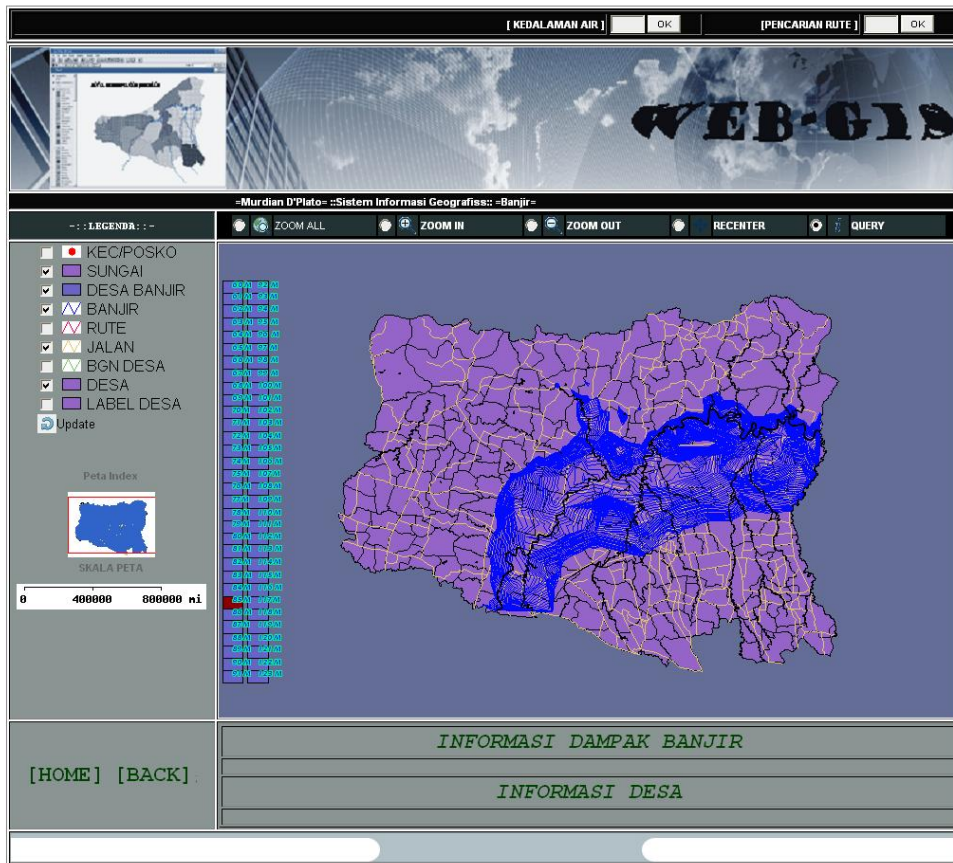
Maksudnya dalam banjir ini sudah banyak desa yang terkena banjir, namun transportasi antara utara sungai dan selatan sungai masih bisa diakses, karena jembatan masih belum terkena banjir.



Gambar 4.22 Tampilan Peta Banjir Sedang (Ketinggian 68 meter)

d. Peta Keadaan Banjir Tinggi

Sementara dalam keadaan banjir yang tinggi, maksudnya adalah dimana keadaan sungai sudah memisahkan akses .jalan antara utara sungai dengan selatan sungai. Jadi daerah yang ada di utara sungai tidak bisa berhubungan dengan daerah selatan sungai.



Gambar 4.23 Tampilan Peta Banjir Tinggi (Ketinggian 85 meter)

e. Pencarian Rute Jalan Terdekat

1) Input lokasi

Tahap awal yang dilakukan pengunjung yang ingin mencari rute terpendek Pariwisata Kota Malang adalah dengan melakukan input lokasi awal dan lokasi tujuan, berikut ini adalah form input data awal pencarian rute terpendek.



Gambar 4.24 Form Pencarian Rute Banjir 85 meter

Dengan kode program sebagai berikut:

```

<?php

include ("../config/koneksi.php");
?>

<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
</head>

<body>
<table width="1000" border="1" align="center" >
  <tr bgcolor="#080808">
    <td>&nbsp; </td>
  </tr>
</table>
  <table width="999" border="1" align="center" >
    <tr>
      <td colspan="2"></td>
    </tr>
    <tr>
      <td colspan="2" bordercolor="#FFFFFF"
bgcolor="#080808"> <marquee>
        <strong><font color="#FFFFFF">=Murdian
D'Plato= ::Sistem Informasi Geografiss::
=Banjir=</font></strong> </marquee></td>
    </tr>
    <tr>
      <td align="center" bgcolor="#233333"> <div
align="center"><strong><font color="#FFFFFF"
face="Courier New, Courier, mono">-::SEARCH::-

```



```

        </tr>
    </table>
    <p>
        <input        type="submit"        name="action"
value="<?php echo "Find Route" ?>">
        <input        name="button"        type=button
onclick=self.history.back() value=Batal>
    </p>
    </form>
</td>
</tr>
<tr>
    <td                align="center"        valign="top"
bgcolor="#8A9593"></td>
</tr>
</table>

</body>
</html>

```

2) Hasil pencarian rute

Setelah melakukan input lokasi awal dan lokasi tujuan kemudian menekan tombol “find route”, maka system akan menampilkan form hasil pencarian rute. Adapun tampilan hasil pencarian rute terpendek menggunakan fungsi *Dijkstra* ditunjukkan seperti pada gambar berikut ini:

=Murdian D'Plato::Sistem Informasi Geografis::=Banjir=
- :: F I N D : -
- :: HASIL PENCARIAN RUTE JALAN [POSKO/KECAMATAN] :-

LOKASI AWAL	Kec. Gemolong
LOKASI TUJUAN	Kec. Sambirejo
TRAYEK	Jln Ds Tenggak-Taraman Jln Ds Gemolong(6) Jln Ds Karangasem(2)-peleman-Genengduwur Jln Ds Taraman(2) Jln Ds Taraman(3)-Singopadu Jln Ds Jetak(3)-Sidoarjo Jln Ds Gawan(4) Jln Ds Sine(14) Jln Ds Sine(13) Jln Ds Sine(16) Jln Ds Sine(11)Guworejo Jln Ds Guworejo(8) Jln Ds Guworejo(6) Jln Ds Wonokerso(6)-Guworejo Jln Ds Wonorejo Jln Ds Wonokerso(8) Jln Ds Wonokerso(10) Jln Ds Wonokerso(5)-kedawung Jln Kedawung(2)/Bendungan(2) Jln Ds Kedawung(3) Jln Ds Bendungan(2) Jln Ds Sambirejo(11) Jln Ds Jetak(4)/Sidoarjo(2) Jln Ds Gabungan Jln Ds Jetak(6)-Sidoarjo(5) Jln Ds Jetak(5)/Sidoarjo(4) Jln Ds Bendungan/kedawung Jln Ds Bendungan(9) Jln Ds Gawan(2)-Jono-Gabungan(3) Jln Ds Slojo/karangasem Jln Ds Ketro-Slojo(2) Jln Ds Guworejo(3) Jln Ds Guworejo(4) Jln Ds Jetak(7)/Sidoarjo(6)
JARAK TEMPUH	± 36.5752832871825 KM

Silahkan Lihat Rute perjalanan anda pada [[Peta](#)] [[Kembali](#)]

Gambar 4.25 Tampilan Hasil Pencarian Rute Banjir 85 meter

Gambar diatas menunjukkan hasil pencarian rute terpendek dengan input lokasi awal dari Kecamatan Gemolong dengan input lokasi tujuan Kecamatan Sambirejo, dan hasil yang ditampilkan dari pencarian rute terpendek tersebut berupa jarak tempuh antara kedua obyek wisata sejauh ± 36.5752832871825 Km dengan nama-nama jalan (trayek) yang bisa dilewati dari Kec. Gemolong menuju Kec. Sambirejo adalah seperti yang ditampilkan pada gambar diatas. Kemudian untuk melihat hasil pencarian pada peta, dapat dilakukan dengan mengklik *link* “peta”. Berikut ini adalah skrip program untuk mencari rute terpendek Kabupaten Sragen dengan menggunakan algoritma Djikstra pada modul *PgRouting*:

```
<?php
```



```

$_POST[trgID]);");
pg_query("SELECT      sum(length(the_geom))/1000      AS
km_roads FROM dijsktra_result;");

?>
    <p>&nbsp;</p>
    <table width="58%" border="1" cellspacing="1"
cellpadding="3">
        <tr>
            <td><font color="#FFFFFF"><strong>LOKASI
AWAL</strong></font></td>
            <td bgcolor="#CCCCCC">
                <?
                $buka=pg_query("SELECT source, kec_posko from
pgjalan where source='$_POST[srcID]");
                while ($s=pg_fetch_array($buka)){
                echo " $s[kec_posko]&nbsp;";
                }
            </td>
        </tr>
        <tr>
            <td width="34%"><font
color="#FFFFFF"><strong>LOKASI
TUJUAN</strong></font></td>
            <td width="66%" bgcolor="#CCCCCC">
                <?
                $buka=pg_query("SELECT source, kec_posko from pgjalan
where source='$_POST[trgID]");
                while ($s=pg_fetch_array($buka)){
                echo "&nbsp;$s[kec_posko]&nbsp;";
                }
                ?>
            </td>
        </tr>
        <tr>
            <td align="top"><font
color="#FFFFFF"><strong>TRAYEK</strong></font></td>
            <td bgcolor="#CCCCCC">
                <? $stampil=pg_query("select jalan from
pgjalan,dijsktra_result where pgjalan.the_geom =
dijsktra_result.the_geom;");
                while($s=pg_fetch_array($stampil)){
                echo "$s[jalan] | ";
                } echo"<br/>";
                ?>
            </td>
        </tr>
        <tr>
            <td align="top"><font
color="#FFFFFF"><strong>JARAK
TEMPUH</strong></font></td>
            <td bgcolor="#CCCCCC">
                <?
                $buka=pg_query("SELECT
sum(length(the_geom))/1000      AS      km roads      FROM

```

```

dijsktra_result");
                while ($s=pg_fetch_array($buka)){
                    echo
"&plusmn;&nbsp;$s[km_roads] KM <br>";
                }
            ?>
        </td>
    </tr>
</table>
</form>
<form id="result">
<?

        echo"Silahkan Lihat Rute perjalanan anda pada
        [
        <a href=http://localhost/Dplato/NIMAS/move.php>Peta</a>
        ]&nbsp;[ <a href=javascript:history.go(-1)>Kembali</a>
        ]";

    ?>

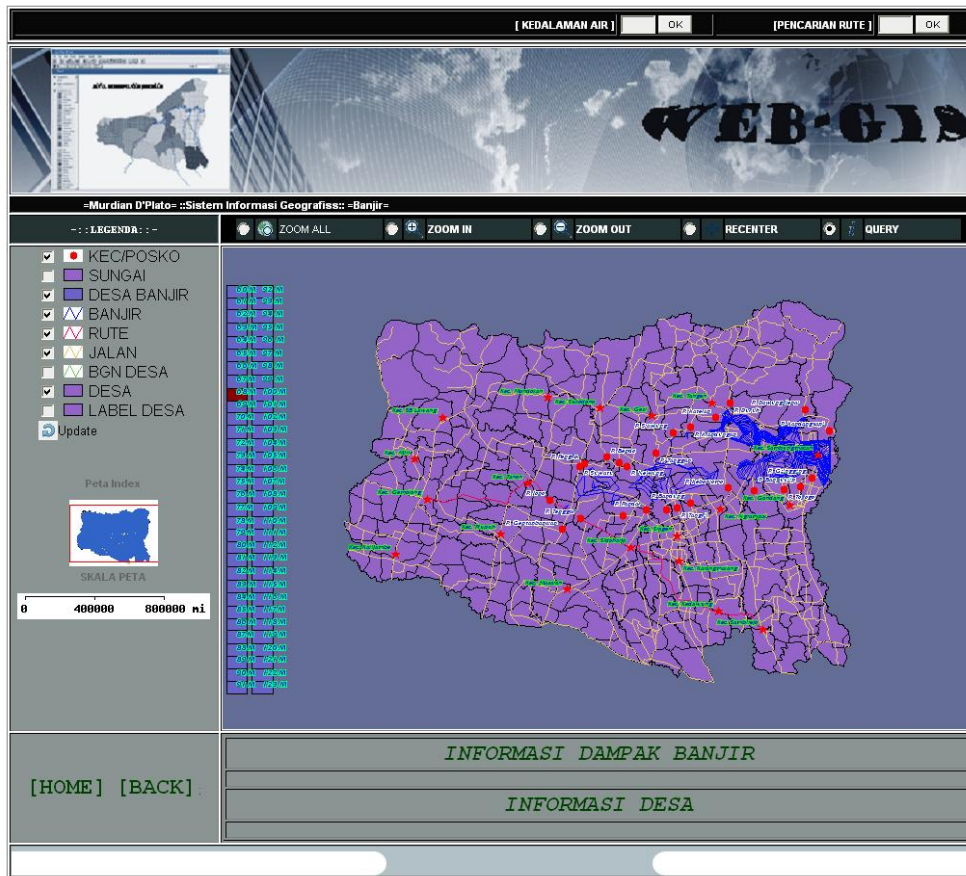
</form></td>
</tr>
<tr>
<td
        align="center"        valign="top"
bgcolor="#8A9593"></td>
</tr>
</table>

</body>
</html>

```

3) Preview peta hasil pencarian rute

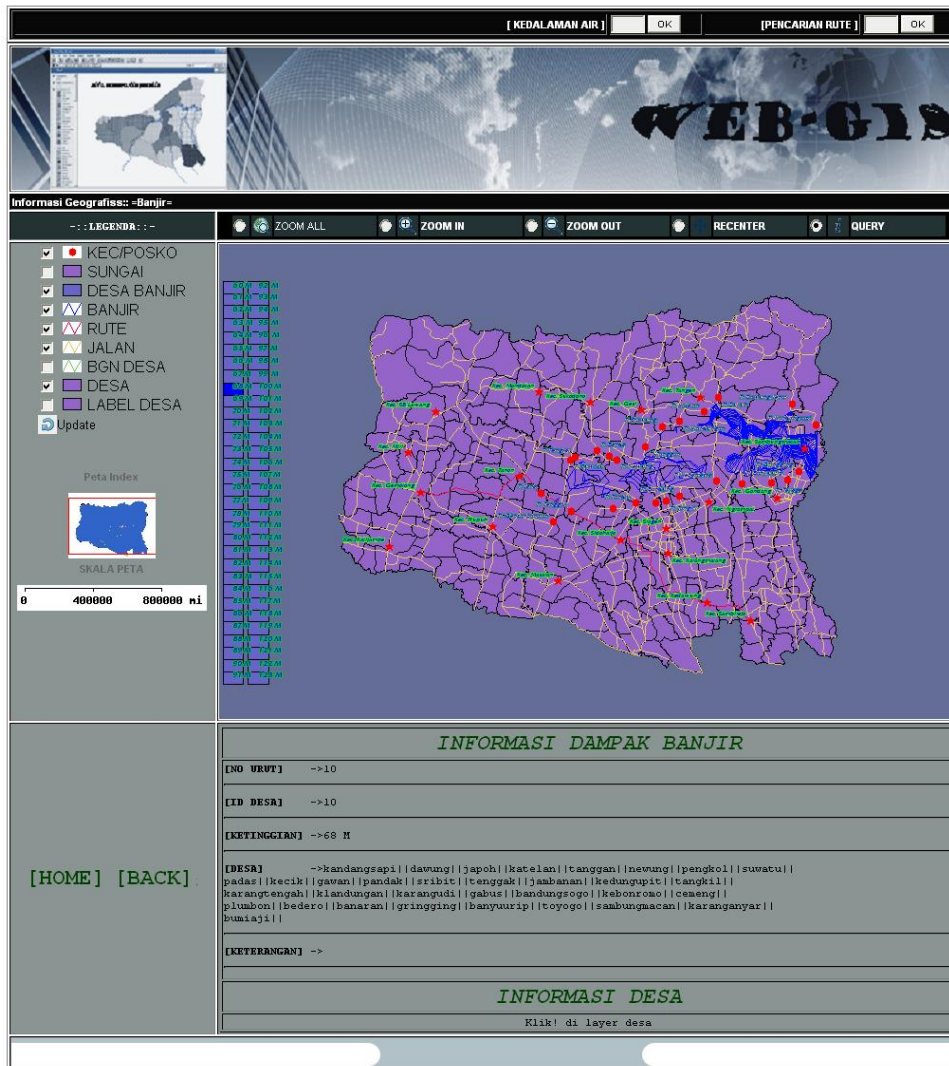
Gambar berikut ini merupakan *preview* hasil pencarian rute terpendek dengan lokasi awal dari Kec. Gemolong dan lokasi tujuan Kec. Sambirejo, rute terpendek antara kedua lokasi digambarkan dengan garis dengan warna merah yang menghubungkan kecamatan (gambar berwarna merah).



Gambar 4.26 Tampilan Peta Hasil Pencarian Rute Banjir 85 meter

f. Query Ketinggian Banjir

Maksud dari query ketinggian banjir adalah jika pengunjung memilih checkbox query dan kemudian meklik dipeta ketinggian dibagian kanan peta maka akan terlihat dengan ketinggian sekiaan maka desa mana saja yang terkena dampak banjir, dengan berapa banyak korban.



Gambar 4.27 Tampilan Hasil Query Ketinggian banjir

Dengan potongan kode program sebagai berikut:

```
function DrawPointQueryResults()
{
    if (!$GLOBALS["gShowQueryResults"]) {
        echo "&nbsp;";
    } else {
        $nResults = 0;
        echo "<PRE>";
        //echo "Propinsi terpilih:\n";
        //echo "-----\n";
        $oLayer = $GLOBALS["goMap"]->getLayerByName("Banjir");
        $nLayerResults = $oLayer->getNumResults();
        if ($nLayerResults > 0) {
```

```

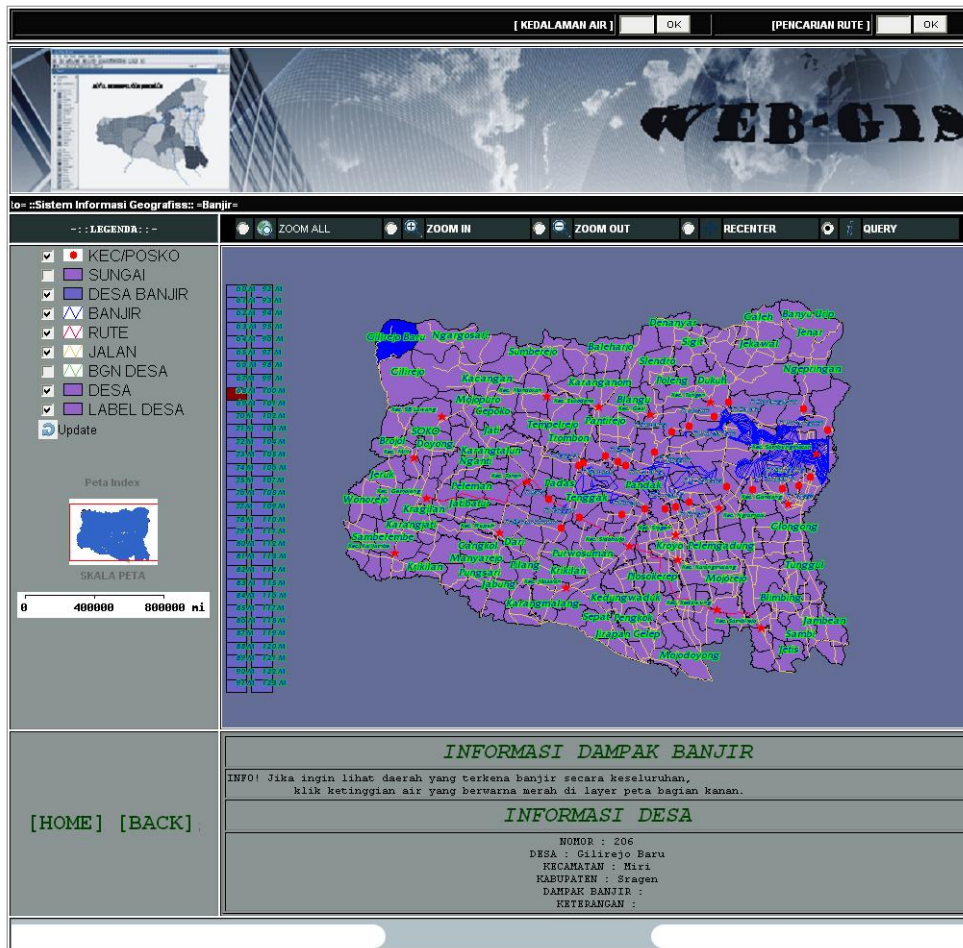
        $oLayer->open();
        $aField      =      explode(";",      $oLayer-
>getMetaData("RESULT_FIELDS"));
        $aDesc      =      explode(";",      $oLayer-
>getMetaData("DESC_FIELDS"));
        for ($i=0; $i<$nLayerResults; $i++) {
            $oRes = $oLayer->getResult($i);
            $oShape = $oLayer->getShape($oRes->tileindex, $oRes-
>shapeindex);
            for ($j=0; $j<count($aField); $j++) {
                echo $aDesc[$j]."->";
                echo $oShape->values[$aField[$j]]."<hr>";
            }
            # DrawShapeQueryResults($oShape);
            $oShape->free();
            $nResults++;
        }
        $oLayer->close();
    }

    if ($nResults == 0) {
        echo "INFO! Jika ingin lihat daerah yang terkena
banjir secara keseluruhan,
        klik ketinggian air yang berwarna merah di layer
peta bagian kanan.";
    }
    echo "</PRE>";
}
}

```

g. Query Data Desa

Query ini tidak jauh berbeda dengan query ketinggian, hanya saja letak kliknya yang berbeda. Pada query letak kliknya langsung dibagian desa yuang ingin dipilih.



Gambar 4.28 Tampilan Hasil Query Informasi Desa

Dengan potongan program sebagai berikut:

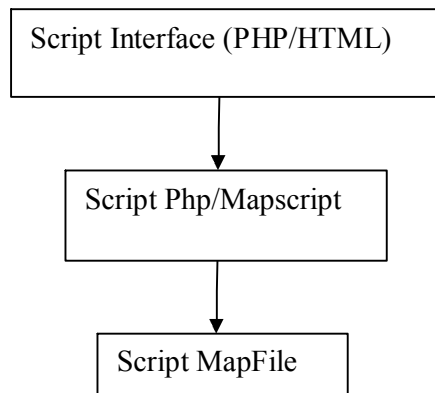
```
function DrawPointQueryResults1 ()
{
  if (!$GLOBALS["gShowQueryResults"]) {
    echo "&nbsp;";
  } else {
    $nResults = 0;
    echo "<PRE>";
    //echo "INFORMASI DESA:\n";
    //echo "-----\n";
    $oLayer = $GLOBALS["goMap"]->getLayerByName("ADMIN
DESA");
    $nLayerResults = $oLayer->getNumResults();
    if ($nLayerResults > 0) {
      $oLayer->open();
      $aField = explode(";", $oLayer-
>getMetaData("RESULT_FIELDS"));
```

```
        $aDesc      =      explode(";", $oLayer->getMetaData("DESC_FIELDS"));
        for ($i=0; $i<$nLayerResults; $i++) {
            $oRes = $oLayer->getResult($i);
            $oShape = $oLayer->getShape($oRes->tileindex,
$oRes->shapeindex);
            for ($j=0; $j<count($aField); $j++) {
                echo $aDesc[$j]." : ";
                echo $oShape->values[$aField[$j]]."\n";
            }
            #DrawShapeQueryResults($oShape);
            $oShape->free();
            $nResults++;
        }
        $oLayer->close();
    }

    if ($nResults == 0) {
        echo "Klik! di layer desa";
    }
    echo "</PRE>";
}
}
```

h. Sistem kerja aplikasi

Dalam aplikasi peta ini memiliki proses kurang lebih sebagai berikut:



Gambar 4.29 Alur aplikasi

Maksudnya adalah Skript interface awal akan meload Php/mapsript dan Php/mapsript kemudian meload kode program MapFile dengan kode program php/mapsript dan Mapfile sebagai berikut:

1) File Php/Mapsript

```
<!-- file: mapsript07.php -->

<?php

Main();

function Main()
{
    $GLOBALS["goMap"] =
ms_newMapObj("./_MAP/map60.map" );

    //batas koordinat seluruh peta, untuk tool 'zoom
all'
    $GLOBALS["gfMinX"] = (float)$GLOBALS["goMap"]-
>extent->minx;
    $GLOBALS["gfMinY"] = (float)$GLOBALS["goMap"]-
>extent->miny;
    $GLOBALS["gfMaxX"] = (float)$GLOBALS["goMap"]-
>extent->maxx;
    $GLOBALS["gfMaxY"] = (float)$GLOBALS["goMap"]-
>extent->maxy;

    //set nilai $aVars dengan nilai parameter URL
    if (sizeof($_POST) > 0) {
        $aVars = $_POST;
    } else {
        if (sizeof($_GET) > 0) {
            $aVars = $_GET;
        } else {
            $aVars = array();
        }
    }

    //tool navigasi default: zoom in
    $GLOBALS["gszCurrentTool"] = "QUERY";
    $GLOBALS["gShowQueryResults"] = FALSE;

    //proses parameter URL
    ProcessURLArray( $aVars );
}

function DrawMap()
{
```

```

if ($GLOBALS["gShowQueryResults"])
    $img = $GLOBALS["goMap"]->drawQuery();
else
    $img = $GLOBALS["goMap"]->draw();

$url = $img->saveWebImage();

$width = $GLOBALS["goMap"]->width;
$height = $GLOBALS["goMap"]->height;

echo "<INPUT TYPE=image SRC=\".$url.\" BORDER=0
WIDTH=\"".
    $width.\"\" HEIGHT=\"\".$height.\"\"
NAME=MAINMAP>\n";

echo "<INPUT TYPE=HIDDEN NAME=MINX VALUE=\"".
    $GLOBALS["goMap"]->extent->minx.\"\">\n";
echo "<INPUT TYPE=HIDDEN NAME=MINY VALUE=\"".
    $GLOBALS["goMap"]->extent->miny.\"\">\n";
echo "<INPUT TYPE=HIDDEN NAME=MAXX VALUE=\"".
    $GLOBALS["goMap"]->extent->maxx.\"\">\n";
echo "<INPUT TYPE=HIDDEN NAME=MAXY VALUE=\"".
    $GLOBALS["goMap"]->extent->maxy.\"\">\n";
}

function DrawScaleBar()
{
    $img = $GLOBALS["goMap"]->drawScaleBar();
    $url = $img->saveWebImage();

    echo"<IMG SRC=$url BORDER=0>\n";
}

function DrawLegend()
{
    echo "<table cellpadding=0 cellspacing=0>";
    echo "<tr bgcolor=#E2EFF5>\n";
    echo "<td></td>\n";
    echo "<td></td>\n";
    echo "</tr>\n";
    echo $GLOBALS["goMap"]->processLegendTemplate(
array() );
    echo "<tr>\n";
    echo "<td><input type='image' ".
        " src='./images/icon_update.png' ".
        " width='20' height='20'></td>\n";
    echo "<td colspan=2>".
        "<font face='Arial,Helvetica,sans-serif' ".
size='2'>".
        "Update</td>\n";
    echo "</tr>\n";
    echo "</table>";
}

```

```

function DrawPointQueryResults()
{
  if (!$GLOBALS["gShowQueryResults"]) {
    echo "&nbsp;";
  } else {
    $nResults = 0;
    echo "<PRE>";
    //echo "Propinsi terpilih:\n";
    //echo "-----\n";
    $oLayer = $GLOBALS["goMap"]->getLayerByName("Banjir");
    $nLayerResults = $oLayer->getNumResults();
    if ($nLayerResults > 0) {
      $oLayer->open();
      $aField = explode(";", $oLayer->getMetaData("RESULT_FIELDS"));
      $aDesc = explode(";", $oLayer->getMetaData("DESC_FIELDS"));
      for ($i=0; $i<$nLayerResults; $i++) {
        $oRes = $oLayer->getResult($i);
        $oShape = $oLayer->getShape($oRes->tileindex, $oRes->shapeindex);
        for ($j=0; $j<count($aField); $j++) {
          echo $aDesc[$j]."->";
          echo $oShape->values[$aField[$j]]."<hr>";
        }
        # DrawShapeQueryResults($oShape);
        $oShape->free();
        $nResults++;
      }
      $oLayer->close();
    }

    if ($nResults == 0) {
      echo "INFO! Jika ingin lihat daerah yang
      terkena banjir secara keseluruhan,
      klik ketinggian air yang berwarna merah di
      layer peta bagian kanan.";
    }
    echo "</PRE>";
  }
}

function DrawPointQueryResults1()
{
  if (!$GLOBALS["gShowQueryResults"]) {
    echo "&nbsp;";
  } else {
    $nResults = 0;
    echo "<PRE>";
    //echo "INFORMASI DESA:\n";
    //echo "-----\n";
    $oLayer = $GLOBALS["goMap"]->getLayerByName("ADMIN DESA");
    $nLayerResults = $oLayer->getNumResults();

```

```

        if ($nLayerResults > 0) {
            $oLayer->open();
            $aField = explode(";", $oLayer->getMetaData("RESULT_FIELDS"));
            $aDesc = explode(";", $oLayer->getMetaData("DESC_FIELDS"));
            for ($i=0; $i<$nLayerResults; $i++) {
                $oRes = $oLayer->getResult($i);
                $oShape = $oLayer->getShape($oRes->tileindex, $oRes->shapeindex);
                for ($j=0; $j<count($aField); $j++) {
                    echo $aDesc[$j]." : ";
                    echo $oShape->values[$aField[$j]]."\n";
                }
                #DrawShapeQueryResults($oShape);
                $oShape->free();
                $nResults++;
            }
            $oLayer->close();
        }

        if ($nResults == 0) {
            echo "Klik! di layer desa";
        }
        echo "</PRE>";
    }
}

function DrawKeyMap()
{
    $img = $GLOBALS["goMap"]->drawreferencemap();
    $url = $img->saveWebImage();

    echo "<INPUT TYPE=image SRC=$url BORDER=1
NAME=KEYMAP>\n";
}

function ProcessURLArray( $aVars)
{
    //simpan tool navigasi yang sedang aktif
    $GLOBALS["gszCurrentTool"] =
(isset($aVars["CMD"])) ?
    $aVars["CMD"] : "QUERY";

    //set batas koordinat peta
    $oExt = $GLOBALS["goMap"];
    $fMinX = isset($aVars["MINX"]) ? $aVars["MINX"] :
$oExt->extent->minx;
    $fMinY = isset($aVars["MINY"]) ? $aVars["MINY"] :
$oExt->extent->miny;;
    $fMaxX = isset($aVars["MAXX"]) ? $aVars["MAXX"] :
$oExt->extent->maxx;;
    $fMaxY = isset($aVars["MAXY"]) ? $aVars["MAXY"] :
$oExt->extent->maxy;;
}

```



```

    $GLOBALS["goMap"]->setextent( $fMinX, $fMinY,
    $fMaxX, $fMaxY );

    //lebar dan tinggi gambar peta
    $fW = $GLOBALS["goMap"]->width;
    $fH = $GLOBALS["goMap"]->height;

    if (isset($_GET["legendlayername"]))
    {
        for( $i=0; $i<$GLOBALS["goMap"]->numlayers;
    $i++ )
        {
            $oLayer = $GLOBALS["goMap"]->
    >getLayer($i);
            if (in_array( $oLayer->name,
    $_GET["legendlayername"] ))
                $oLayer->set( "status", MS_ON );
            else
                $oLayer->set( "status", MS_OFF );
        }
    }

    //lakukan perubahan skala, sesuai tool navigasi
    terpilih
    if (isset($aVars["CMD"]) && isset
    ($aVars["MAINMAP_x"])) {
        //titik tempat user meng-klik pada lokasi peta
        $nX = isset($aVars["MAINMAP_x"]) ?
        intval($aVars["MAINMAP_x"]) : $fW/2.0;
        $nY = isset($aVars["MAINMAP_y"]) ?
        intval($aVars["MAINMAP_y"]) : $fW/2.0;

        if (isset($aVars["MAINMAP_x"]) &&
    isset($aVars["MAINMAP_y"])) {
            $oPixelPos = ms_newpointobj();
            $oPixelPos->setxy($nX, $nY);

            $oGeoExt = ms_newrectobj();
            $oGeoExt->setextent($fMinX, $fMinY, $fMaxX,
    $fMaxY);

            //ubah skala peta, dengan method zoompoint
            atau setextent
            if ($aVars["CMD"] == "ZOOM_IN") {
                $GLOBALS["goMap"]->zoompoint(2, $oPixelPos,
                $fW, $fH, $oGeoExt);
            } else if ($aVars["CMD"] == "ZOOM_OUT") {
                $GLOBALS["goMap"]->zoompoint(-2,
    $oPixelPos,
                $fW, $fH, $oGeoExt);
            } else if ($aVars["CMD"] == "RECENTER") {
                $GLOBALS["goMap"]->zoompoint(1, $oPixelPos,
                $fW, $fH, $oGeoExt);
            } else if ($aVars["CMD"] == "ZOOM_ALL") {
                $GLOBALS["goMap"]->
    >setextent($GLOBALS["gfMinX"],

```

```

        $GLOBALS["gfMinY"], $GLOBALS["gfMaxX"],
        $GLOBALS["gfMaxY"]);
    } else if ($aVars["CMD"] == "QUERY")
    {
        $nGeoX = Pix2Geo($nX, 0, $fW, $fMinX,
        $fMaxX, 0);
        $nGeoY = Pix2Geo($nY, 0, $fH, $fMinY,
        $fMaxY, 1);

        $oGeo = ms_newPointObj();
        $oGeo->setXY($nGeoX, $nGeoY);

        // Simbol '@' digunakan supaya tidak
        muncul pesan peringatan
        // ketika objek tidak ditemukan
        @$GLOBALS["goMap"]->queryByPoint($oGeo,
        MS_SINGLE, -1);

        $GLOBALS["gShowQueryResults"] = TRUE;
    }
} else if (isset($aVars["KEYMAP_x"])
&& isset($aVars["KEYMAP_y"])) {

    $oRefExt = $GLOBALS["goMap"]->reference-
>extent;

    $nX = intval($aVars["KEYMAP_x"]);
    $nY = intval($aVars["KEYMAP_y"]);

    $fWidthPix = doubleval($GLOBALS["goMap"]-
>reference->width);
    $fHeightPix = doubleval($GLOBALS["goMap"]-
>reference->height);

    $nGeoX = Pix2Geo($nX, 0, $fWidthPix, $oRefExt-
>minx, $oRefExt->maxx, 0);
    $nGeoY = Pix2Geo($nY, 0, $fHeightPix, $oRefExt-
>miny, $oRefExt->maxy, 1);

    $fDeltaX = ($fMaxX - $fMinX) / 2.0;
    $fDeltaY = ($fMaxY - $fMinY) / 2.0;

    $GLOBALS["goMap"]->setextent($nGeoX - $fDeltaX,
    $nGeoY - $fDeltaY,
    $nGeoX +
    $fDeltaX, $nGeoY + $fDeltaY);
}

}

function Pix2Geo($nPixPos, $fPixMin, $fPixMax,
$fGeoMin, $fGeoMax,
                $bInversePix)
{

```

```

    $fDeltaPix = ($bInversePix) ? $fPixMax -
    $nPixPos : $nPixPos - $fPixMin;

    $fDeltaGeo = $fDeltaPix * ($fGeoMax - $fGeoMin)
    /
    ($fPixMax - $fPixMin);

    return $fGeoMin + $fDeltaGeo;
}

function IsCurrentTool( $szTool )
{
    return (strcasecmp($GLOBALS["gszCurrentTool"],
    $szTool) == 0);
}

```

?>

2) File .Map (Mapfile)

```

MAP

    NAME "Indonesia"
    STATUS ON
    EXTENT 464280.04 9165688.39 527419.11 9200465.53
    IMAGETYPE PNG
    SIZE 780 500
    # SHAPEPATH "./dataku"
    IMAGECOLOR 100 109 150

    SYMBOLSET "./etc/symbols.sym"
    FONTSET "./etc/fonts.txt"

    UNITS dd

    WEB
        IMAGEPATH "/tmp/ms_tmp/"
        IMAGEURL "/ms_tmp/"
        LOG "/ms_tmp/err.log"

```

```

END

LEGEND
IMAGECOLOR 255 255 255
KEYSIZE 18 12
KEYSPACING 5 5
LABEL
SIZE MEDIUM
TYPE BITMAP
BUFFER 0
COLOR 0 0 89
FORCE FALSE
MINDISTANCE -1
MINFEATURESIZE -1
OFFSET 0 0
PARTIALS TRUE
END
POSITION LL
STATUS ON
TEMPLATE "legend.html"
END

QUERYMAP
COLOR 0 0 255
SIZE -1 -1
STATUS OFF
STYLE HILITE
END

SCALEBAR
STYLE 1
INTERVALS 2
UNITS miles
END
REFERENCE
COLOR -1 -1 -1
EXTENT 464280.04 9165688.39 527419.11
9200465.53
IMAGE "sragen.png"
OUTLINECOLOR 255 0 0
SIZE 120 80
STATUS ON
END

LAYER

NAME "LABELDESA"
GROUP "Polygon"
STATUS OFF
TYPE POLYGON
TRANSPARENCY 1
DATA "the_geom FROM labeldesa USING UNIQUE gid"
CONNECTIONTYPE POSTGIS
CONNECTION "user=postgres dbname=sragen
host=localhost port=5432 password=root"

```

```

CLASSITEM "gid"
LABELITEM "desa"

METADATA
  "DESCRIPTION"      "LABEL DESA"
END

CLASS
  NAME "LABELDESA"
  STYLE
  COLOR 405 100 200
  OUTLINECOLOR 1 1 1
  SIZE 5
END
  LABEL
  FONT fritqat-italic
  TYPE truetype
  SIZE 7
  POSITION UC
  PARTIALS FALSE
  OUTLINECOLOR 000 255 000
  COLOR 0 0 255
END
END

END

LAYER

  NAME "ADMIN DESA"
  GROUP "Polygon"
  STATUS ON
  TYPE POLYGON
  DATA "the_geom FROM pgadmin desa USING UNIQUE
gid"
  CONNECTIONTYPE POSTGIS
  CONNECTION "user=postgres dbname=sragen
host=localhost port=5432 password=root"

  CLASSITEM "gid"
  LABELITEM "desa"

  METADATA
    "DESCRIPTION"      "DESA"
    # "GROUP"          "polygon"
    # "LAYER"          "admin desa"
    "RESULT_FIELDS"
"count; desa; kecamatan; kabupaten; dampak_banjir; keter
angan"
    "DESC_FIELDS"
"NOMOR; DESA; KECAMATAN; KABUPATEN; DAMPAK          BANJIR;
KETERANGAN"
  END

```

```

TEMPLATE "admindesa.html"
TOLERANCE 0
TOLERANCEUNITS PIXELS

CLASS
  NAME "ADMIN DESA"
  STYLE
    COLOR 405 100 200
    OUTLINECOLOR 1 1 1
    SIZE 5
  END
END

END

LAYER

  NAME "BANGUNAN DESA"
  GROUP "line"
  STATUS OFF
  TYPE LINE
  DATA "the_geom FROM pgtgl_new USING UNIQUE
gid"
  CONNECTIONTYPE POSTGIS
  CONNECTION "user=postgres dbname=sragen
host=localhost port=5432 password=root"

  METADATA
    "DESCRIPTION" "BGN DESA"
  END

  CLASS
    Name "BANGUNAN DESA"
    COLOR 625 200 100
    OUTLINECOLOR 0 0 0
  END

END

LAYER

  NAME "JALAN"
  GROUP "line"
  STATUS OFF
  TYPE LINE
  DATA "the_geom FROM pgjalan USING UNIQUE gid"
  CONNECTIONTYPE POSTGIS
  CONNECTION "user=postgres dbname=sragen
host=localhost port=5432 password=root"

  METADATA
    "DESCRIPTION" "JALAN"
  END

  CLASS

```

```

        NAME "JALAN"
        COLOR 225 200 100
        OUTLINECOLOR 0 0 0
    END

END

LAYER
    NAME "Dijkstra"
    GROUP "line"
    STATUS OFF
    TYPE LINE

    CONNECTIONTYPE POSTGIS
    CONNECTION "user=postgres dbname=sragen
host=localhost port=5432 password=root"
    DATA "the_geom from dijkstra_result using unique
gid"

    CLASSITEM "gid"
    #LABELITEM "oid"

    METADATA
        "DESCRIPTION" "ROUTE"
        "RESULT_FIELDS" "oid"
        "GROUP" "line"
        "LAYER" "dijkstra_result"
    END

    CLASS
        NAME "dijkstra"
        LABEL
            TYPE TRUETYPE
            FONT "sans-italic"
            SIZE 8
            MINSIZE 8
            MAXSIZE 9
            POSITION AUTO
            COLOR 0 0 0
            PARTIALS FALSE
        END
        STYLE
            SYMBOL "circle"
            COLOR 255 0 335
            ANTIALIAS TRUE
        END
    END
END

LAYER

    NAME "banjirkontur"
    GROUP "line"
    STATUS ON

```

```

TYPE LINE
  DATA "the_geom FROM pgkontur05mtr USING
UNIQUE gid"
  CONNECTIONTYPE POSTGIS
  CONNECTION "user=postgres dbname=sragen
host=localhost port=5432 password=root"

METADATA
  "DESCRIPTION" "BANJIR"
  "RESULT_FIELDS" "id;contour"
  "DESC_FIELDS" " NO; DESA"
END
  TOLERANCE 0
  TOLERANCEUNITS PIXELS

CLASS
  NAME "kontur"
  EXPRESSION ([contour] >= 0 AND [contour]
<60.5)
  COLOR 259 779 255
  OUTLINECOLOR 1 1 1
  END

END

LAYER

  NAME "Banjir"
  GROUP "Polygon"
  STATUS ON
  TYPE POLYGON
  DATA "the_geom FROM bnjrtombol USING UNIQUE
gid"
  CONNECTIONTYPE POSTGIS
  CONNECTION "user=postgres dbname=sragen
host=localhost port=5432 password=root"

  CLASSITEM "gid"
  LABELITEM "ketinggian"

  METADATA
  "DESCRIPTION" "DESA BANJIR"
  "RESULT_FIELDS"
"gid;id;ketinggian;desa;keterangan"
  "DESC_FIELDS" "<strong>[NO URUT]
</strong>;<strong>[ID DESA]
</strong>;<strong>[KETINGGIAN]</strong>
;<strong>[DESA]
</strong>;<strong>[KETERANGAN] </strong>"
  END

  TEMPLATE "desabanjir.php"
  TOLERANCE 0
  TOLERANCEUNITS PIXELS

```



```

        CLASS
            NAME "Banjir"
            COLOR 105 100 200
            OUTLINECOLOR 1 1 1

            LABEL
            FONT fritqat-italic
            TYPE truetype
            SIZE 5
            POSITION UR
            PARTIALS FALSE
            OUTLINECOLOR 000 255 155
            COLOR 0 0 255
        END
    END

END

LAYER

    NAME "SUNGAI"
    GROUP "Polygon"
    STATUS OFF
    TYPE POLYGON
    DATA "the_geom FROM pgsungai_polygon_region
USING UNIQUE gid"
    CONNECTIONTYPE POSTGIS
    CONNECTION "user=postgres dbname=sragen
host=localhost port=5432 password=root"

    CLASSITEM "gid"
    LABELITEM "pl"

    METADATA
    "DESCRIPTION" "SUNGAI"
    END

    CLASS
    NAME "SUNGAI"
    COLOR 405 100 200
    OUTLINECOLOR 1 1 1
    END

END

LAYER

    NAME "KEC"
    TYPE POINT
    GROUP "point"
    STATUS OFF
    TYPE POINT
    DATA "the_geom FROM kantor_kec USING UNIQUE
gid"

```

```
CONNECTIONTYPE POSTGIS
CONNECTION "user=postgres dbname=sragen
host=localhost port=5432 password=root"
```

```
CLASSITEM "gid"
LABELITEM "kecamatan"
```

```
METADATA
```

```
"DESCRIPTION" "KECAMATAN"
"GROUP" "polygon"
"LAYER" "admin desa"
"RESULT_FIELDS" "kecamatan"
"DESC_FIELDS" "KECAMATAN"
```

```
END
```

```
TOLERANCE 0
TOLERANCEUNITS PIXELS
```

```
CLASS
```

```
NAME "kec"
SYMBOL 2
SIZE 8
COLOR 255 0 0
```

```
LABEL
```

```
FONT fritgat-italic
TYPE truetype
SIZE 7
POSITION AUTO
PARTIALS FALSE
OUTLINECOLOR 000 255 000
COLOR 0 0 255
```

```
END
```

```
END
```

```
END
```

```
END
```

BAB V PENUTUP

5.1 Kesimpulan

Dengan dibuatnya aplikasi ini maka dapat disimpulkan:

1. Aplikasi ini dapat mensimulasikan bahwa genangan air lebih banyak terjadi pada desa-desa yang berada di selatan sungai Bengawan Solo daripada yang berada di utara sungai, karena desa-desa di selatan sungai kontur tanahnya rata-rata lebih rendah dibandingkan desa-desa utara sungai.
2. Aplikasi ini membuktikan bahwa tampilan peta digital dalam bentuk database akan mudah diolah daripada peta digital yang dalam bentuk gambar digital biasa, terlebih pada update data/informasi atributnya.
3. Kedetailan dan keakuratan aplikasi ini dalam mensimulasikan banjir, dipengaruhi oleh kedetailan data kontur, semakin detail semakin akurat.
4. Aplikasi ini hanya dapat mencari rute terpendek di jalan yang tidak terkena banjir, sehingga tidak dapat menemukan rute terpendek di jalan yang terkena banjir.
5. Akurasi dalam proses pembuatan atau dijitasi jalan akan mempengaruhi akurasi hasil perutean dalam aplikasi ini, semakin akurat data jalan, maka semakin akurat pula hasil perutean yang akan ditampilkan.

5.2 Saran

1. Perlu diperhatikan dalam penggunaan versi MapServernya, karena terkadang program tidak jalan dikarenakan salah menginstall versi MS4Wnya berbeda.
2. Jika ingin memindah aplikasi PgRouting dari komputer satu ke komputer lainnya perlu dicermati dalam hal settingnya.
3. Perlu ketelitian dalam hal digitasi khususnya pada digitasi jalan.
4. Perlu dilakukan penelitian dan kegiatan yang mendalam untuk pembuatan aplikasi sejenis, demi mendapatkan keakuratan data.

DAFTAR PUSTAKA

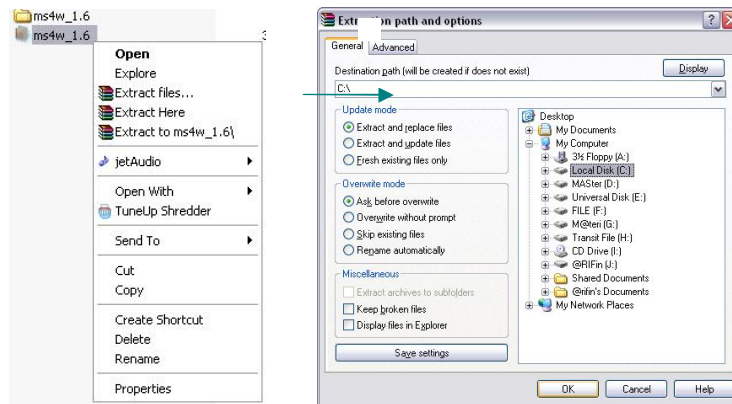
- Abdullah bin Muhammad. 2006. *Tafsir Ibnu Katsir*. Jakarta: Pustaka Imam Syafi'i
- Al-Mahally, M. Ibnu A. & As-Suyuthi, Jalaludin. Tanpa tahun. *Tafsir Jalalain*.
Terjemah oleh Dani Hidayat. 2008. Tasikmalaya: Pustaka Hidayah.
- Adam, A. Lala. 2004. *PHP&PostgreSQL*. Yogyakarta: ANDI
- Atailah, Syekh Ahmad. 1995. *Mutu Manikam dari kitab Al Hikam*. Surabaya:
Mutiara Ilmu.
- Anonymous. 2009. *Sungai Bengawan Solo*. (Online),
http://wapedia.mobi/ms/Sungai_Bengawan_Solo, diakses 27/06/2010)
- Anonymous. 2009. *Sejarah Sragen*, (Online), (
<http://www.sragenkab.go.id/home.php?menu=1>, diakses 27/juni/2010)
- Anonymous. 2009. *Geografi Kabupaten Sragen*, (Online), (
<http://www.sragenkab.go.id/home.php?menu=2>, diakses 27/juni/2010)
- Anonymous. 2007. *Shortest Path Search In Real Road Networks With pgRouting*.
(<http://pgrouting.postlbs.org/wiki/>, diakses pada 31 Juni 2010)
- Ellyns.. 2009. *Definisi Simulasi*(<http://ellyns.wordpress.com/2009/08/28/definisi-simulasi-2/>), (Online), diakses 26 juni 2010
- Hakim lukman, Musalini uus. 2004. *Cara Cerdas Menguasai Layout, Desain dan Aplikasi Web*. Jakarta: Gramedia.
- Hakim, lukman. 2008. *Membongkar Trik Rahasia Para Master PHP*. Yogyakarta:
Lokomedia.
- Nuarsa, I Wayan. 2005. *Menganalisa Data Spasial dengan ArcView 3.3 untuk Pemula*. Jakarta: PT Elex Media Komputindo.
- Nuryadin, Ruslan. 2005. *Panduan MapServer*. Bandung: Informatika.
- Prahasta, Eddy. 2002. *Tutorial ArcView*. Bandung: Informatika.

- Prahasta, Eddy. 2006. *Membangun Aplikasi Web-based GIS Dengan MapSever*. Bandung: Informatika.
- Prahasta, Eddy. 2002. *Konsep-konsep Dasar Sistem Informasi Geografis*. Bandung: Informatika.
- Prahasta, Eddy. 2004. *SIG: Dukungan Tools dan Plug-ins(Extension) Dalam Pengembangan Berbagai Aplikasi*. Bandung: Informatika.
- DPU DIRJEN SDA. 2008. *Pedoman Siaga Banjir Wilayah Sungai Bengawan Solo Musim Hujan 2008/2009*. Surakarta: BBWS Sungai Bengawan Solo.
- Sub Dinas Pengairan. 2004. *Pedoman Siaga Banjir dilingkungan Sub Dinas Pengairan Dinas Pekerjaan Umum Kab. Sragen*. Sragen: Pemkab Sragen DPU.
- Veriawan, Herindra. 2010. *Sistem, Model dan Simulasi*, (Online), <http://indraawan.blog.uns.ac.id/files/2010/04/dasar-teori.pdf>, diakses 26 Juni 2010

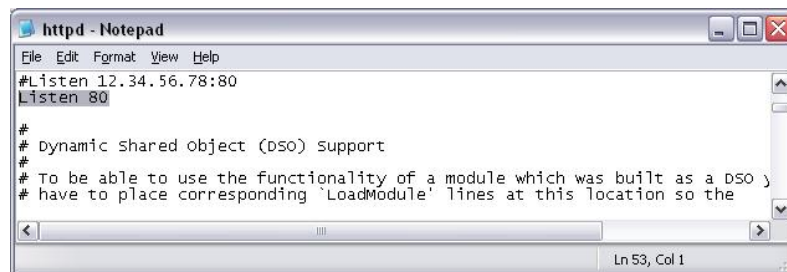
LAMPIRAN A
Proses Instalasi Mapserver,
PostgreSQL.

Instalasi MapServer

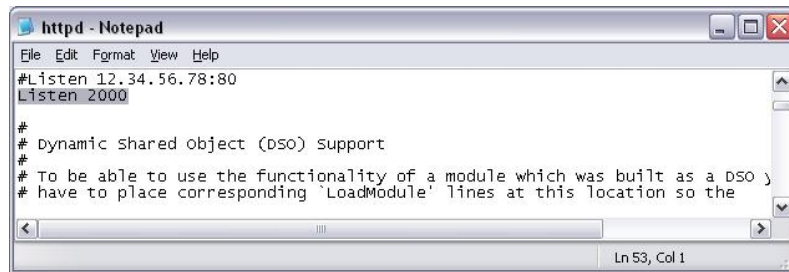
1. Siapkan aplikasi MapSever [ms4w_1.6.zip](http://www.maptools.org/ms4w/index.phtml?page=downloads.html) aplikasi freeware ini dapat di download di <http://www.maptools.org/ms4w/index.phtml?page=downloads.html>
2. Setelah didownload ekstrak file tersebut ke root direktori hardisk misal C:/ atau di D:/



3. Apabila sudah ada aplikasi web server yang lainnya maka settinglah terlebih dahulu file **httpd.conf** yang berada di **C:\ms4w\Apache\conf\httpd.conf**. Perubahan dilakukan pada baris **"Listen 80"** menjadi **"Listen 2000"**, baris ini merupakan konfigurasi port mana yang akan digunakan dalam instalasi Mapserver, default Potr adalah 80 dan dapat diganti dengan port yang nilainya lebih besar dari 1024 supaya tidak terjadi benturan diantara mapserver yang ada.



Menjadi



```
File Edit Format View Help
#Listen 12.34.56.78:80
Listen 2000

#
# Dynamic shared object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding 'LoadModule' lines at this location so the
```

4. Kemudian eksekusi *apache-install.bat* untuk menginstal service Apache

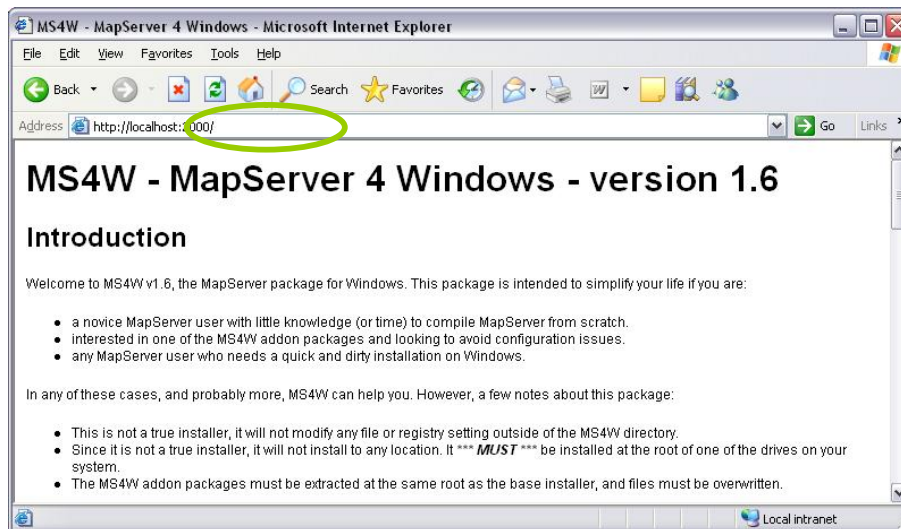


5. Untuk mengecek apakah mapserver sudah terinstal dengan baik atau belum maka bukalah web browser dan ketikkan alamat:

<http://localhost:2000/>

Port Instalasi Mapserver

6. Apabila instalasi telah berhasil maka di web browser akan ada tampilan sebagai berikut:



7. Kemudian restart Apache dengan mengeksekusi *apache-restart.bat*. (Yang ada di direktori “C:\ms4w”)



Kemudian buka kembali <http://localhost:2000/> lewat web browser

8. Copy atau ketikkan “extension=php_mapscript_4.10.0.dll” pada file “php.ini” yang terletak di direktori “C:\ms4w\Apache\cgi-bin” tepatnya dibawah “;windows extension” seperti gambar dibawah ini :



9. Untuk mengecek keberhasilan dari instalasi Mapscript, buat file **MsInfo.php** pada folder D:\ms4w\Apache\htdocs yang berisi :

```
<?php
echo ms_getVersion();
?>
```

Atau

```
<?php
dl("php_mapscript_4.10.0.dll");
echo ms_getVersion();
?>
```

- 🔍 Periksa apakah file “php_mapscript_4.10.0.dll” sama dengan nama file yang ada di direktori C:\ms4w\Apache\php\extensions, jika tidak maka samakan terlebih dahulu.

10. Setelah itu buka <http://localhost:2000/MsInfo.php> jika telah berhasil maka di halaman Webbrowser akan ada tampilan seperti berikut ini:



11. Jika langkah ke elapan belum dilaksanakan maka akan muncul pesan seperti tampilan dibawah ini jika langkah ke sepuluh dijalankan :



🐛 Jika semuanya telah sesuai maka instalasi MapServer telah berhasil.

PostgreSQL

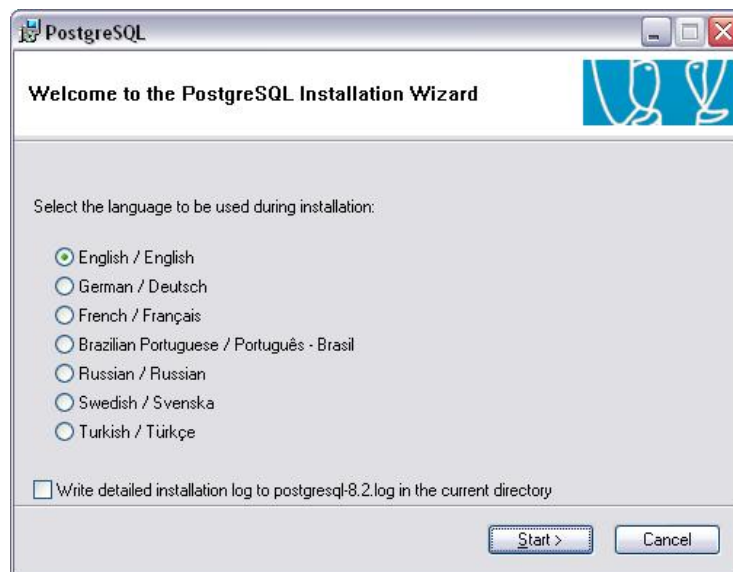
Instalasi PostgreSQL

- Siapkan file postgresql-8.2.5-1.zip
- Ekstraklah file tersebut di sembarang direktori, setelah berhasil akan ada file installer dari PostgreSQL seperti gambar dibawah ini:



Gambar dari Icon Instaler PosgreSQL

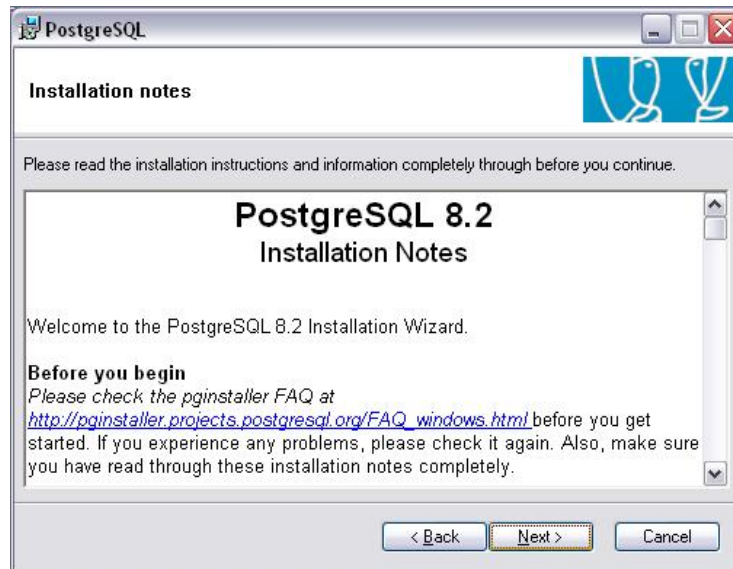
- Klik ganda icon untuk menjalankan aplikasi. sehingga akan tampil window untuk memilih bahasa
- Instalasi tahap 1



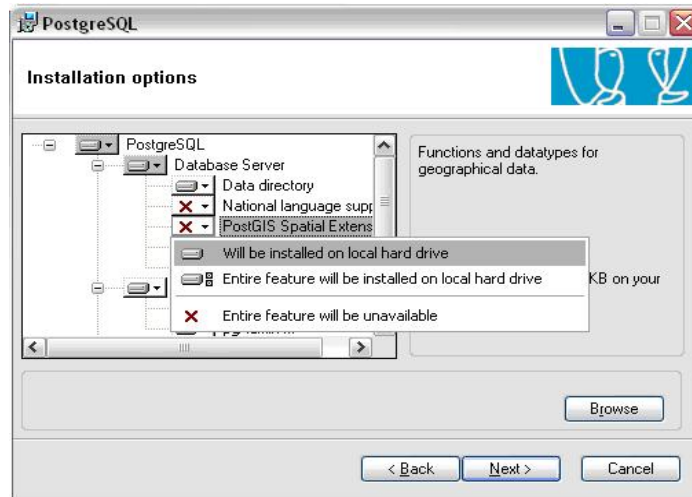
- Instalasi tahap 2



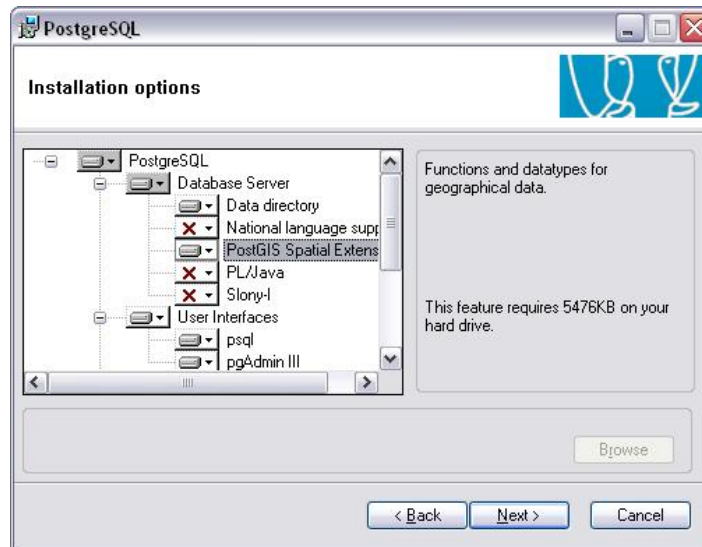
- Instalasi tahap 3



- Instalasi tahap 4, Aktifkanlah PosGIS Spatian Extension

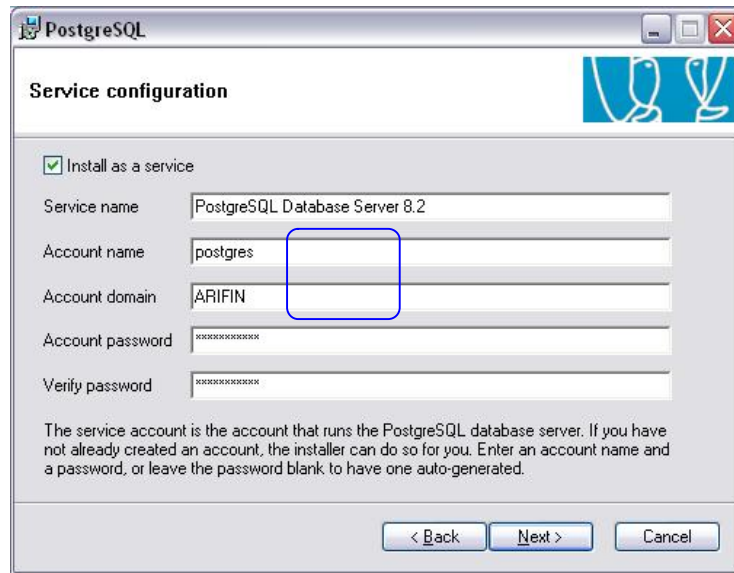


Tampilan sebelum diaktifkannya PostGIS Spatian Extension

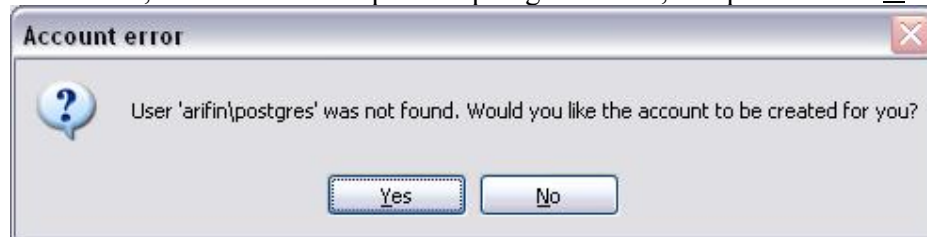


Tampilan setelah diaktifkannya PostGIS Spatian Extension

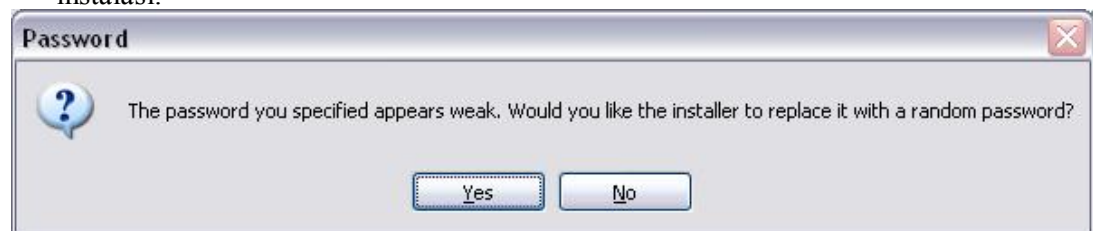
- Instalasi Tahap 5, pada tahap ini text field yang masi kosong adalah password saja, text field pada Account domain field ini akan diisi secara otomatis dengan nama komputer tempat kita menginstal dan tidak perlu dirubah, dan jika dirubah maka pada waktu instalasi berjalan akan ada pesan kesalahan. Semua isian yang kita masukkan akan tesimpan di direkroti *C:\Documents and Settings*, sehingga kita tidak bisa menngunakan nama da domain yang telah kita gunakan sebelumnya jika terjadi kesalahan pada waktu instalasi berjalan.



- Setelah semua field terisi maka tekan tombol Next> untuk melanjutkan instalasi, dan akan muncul pesan seperti gambar x.x, dan pilih tombol YES



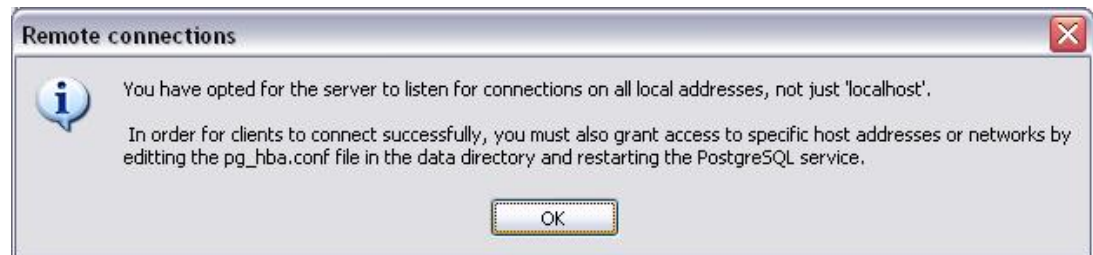
- Setelah itu biasanya muncul pesan lagi yang menyatakan bahwa password yang kita gunakan terlalu lemah, dan pilih tombol No untuk melanjutkan instalasi.



- Jika pada pesan diatas ditekan tombol YES maka akan muncul pilihan untuk menggunakan password acak.



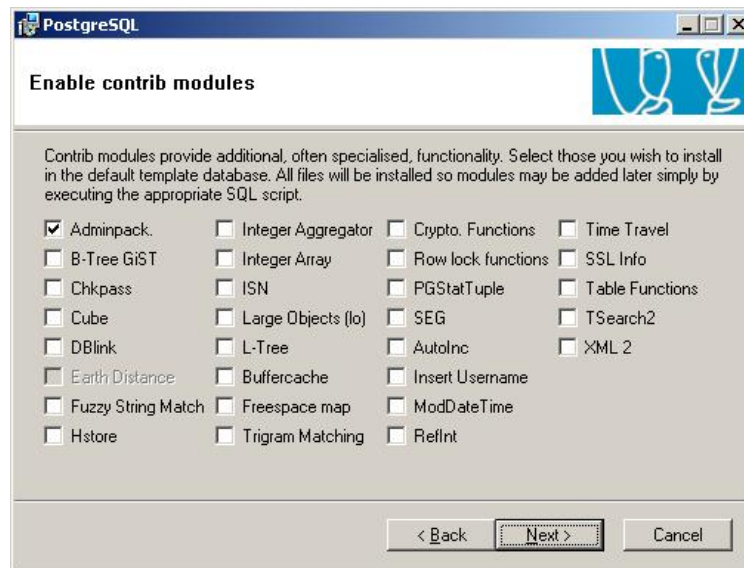
- Instalasi tahap 6



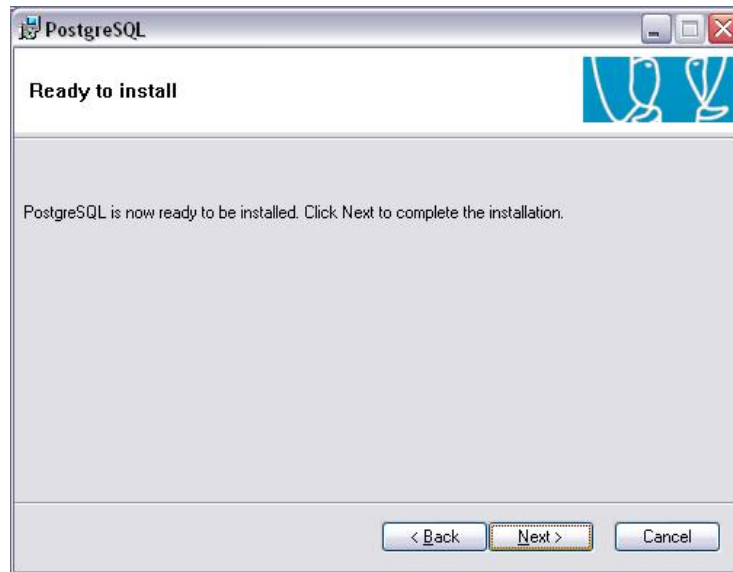
- Instalasi tahap 7



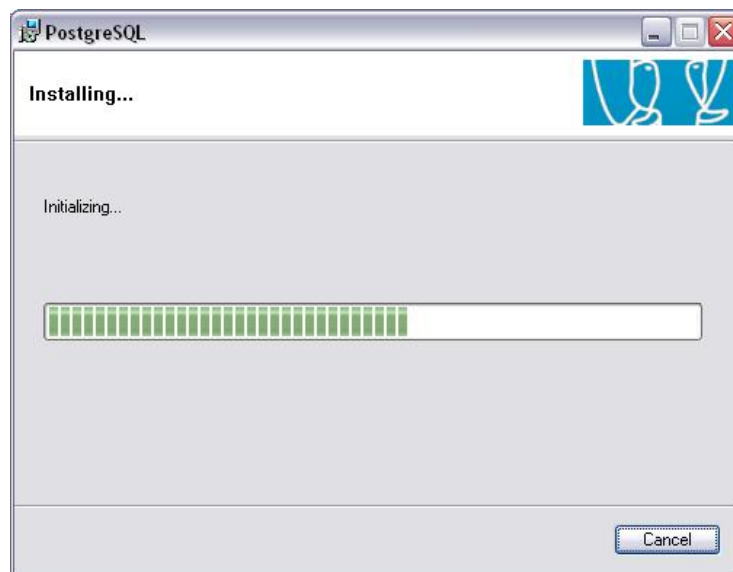
- Instalasi tahap 8



- Instalasi tahap 9



- Instalasi tahap 10, proses instalasi.



Exten dan Projection di file MAP

Nilai EXTENT berfungsi untuk menentukan titik pojok atas-bawah-kiri-kanan atau longitudes dan latitudes dalam satuan decimal degree format, bukan

Degree-Minutes-Seconds) sebuah peta, kita dapat mendapatkan nilai extent pada sebuah SHP dengan menggunakan program OGR.

OGR adalah sebuah library dan command-tool berbasis C++ Open Source, OGR menyediakan fungsi membacanya juga bisa menulis segala macam file berformat vektor, termasuk ESRI Shapefiles, Mapinfo mid/mif dan TAB. OGR juga bagian dari GDAL library.

Pada paket MS4W terdapat program “**ogrinfo.exe**” yang terletak di direktori C:\ms4w\tools\gdal-ogr.

Hal-hal yang perlu disiapkan sebelum aplikasi tersebut dijalankan adalah:

1. Copy file-file berikut ini:

- ✓ gdal13.dll
- ✓ geotiff.dll
- ✓ libecwj2.dll
- ✓ libpq.dll
- ✓ libtiff.dll
- ✓ lti_dsdk_cdll.dll
- ✓ lti_dsdk_dll.dll
- ✓ xerces-c_2_7.dll

yang berada di direktori “C:\ms4w\Apache\cgi-bin” ke direktori “C:\ms4w\tools\gdal-ogr”

2. Eksekusi atau jalankan “setenv.bat” yang berada di direktori “C:\ms4w\setenv.bat”



3. Setelah langkah ke dua dijalankan maka restartlah aplikasi mapserver.

Setelah langkah-langkah diatas dijalankan kemudian bukalah aplikasi CMD atau DOS prompt dan masuk ke direktori dimana aplikasi ogrinfo berada yaitu berada di direktori

“C:\ms4w\tools\gdal-ogr”

pada command prompt.

```
C:\ms4w\tools\gdal-ogr>ogrinfo.exe C:\JAtim\sujatim.shp sujatim -summary
```



Tampilan ketika baris perintah diatas dijalankan lewat command prompt:

```
C:\WINDOWS\system32\cmd.exe

C:\ms4w\tools\gdal-ogr>ogrinfo.exe C:\JAtim\sujatim.shp sujatim -summary
INFO: Open of 'C:\JAtim\sujatim.shp'
using driver 'ESRI Shapefile' successful.

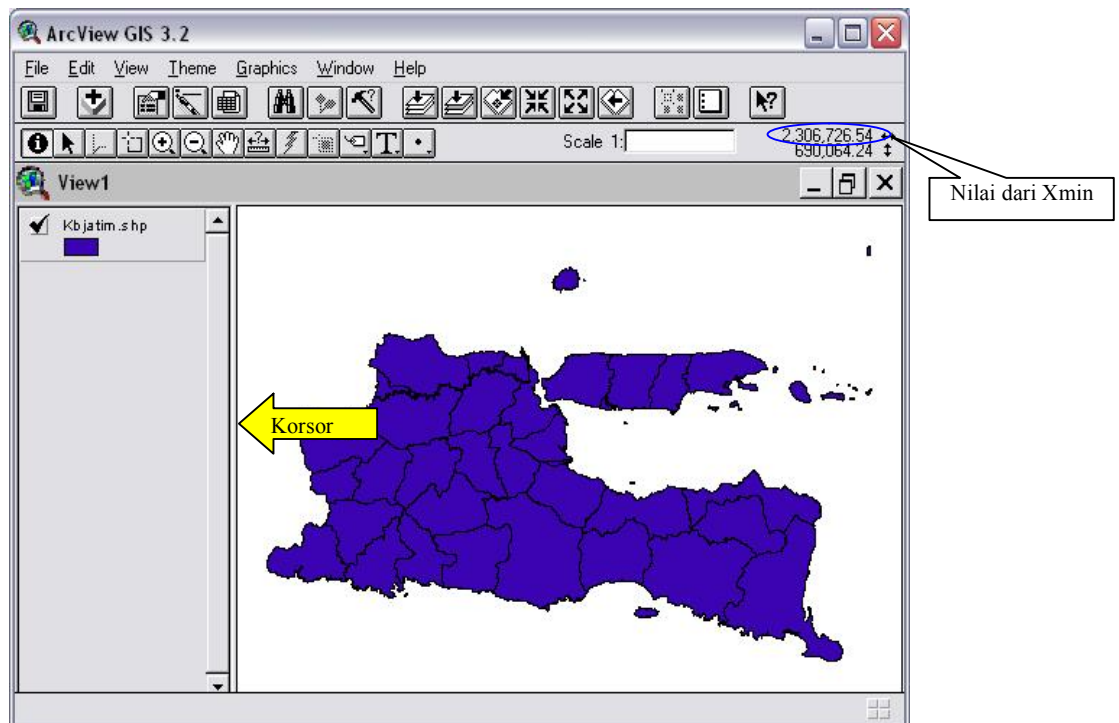
Layer name: sujatim
Geometry: Line String
Feature Count: 1259
Extent: (2327033.500000, 523458.375000) - (2755915.250000, 807341.562500)
Layer SRS WKT:
<unknown>
FNODE_ : Real (11.0)
TNODE_ : Real (11.0)
LPOLY_ : Real (11.0)
RPOLY_ : Real (11.0)
LENGTH: Real (13.6)
SUJATIM_ : Real (11.0)
SUJATIM_ID: Real (11.0)
NAMA_SNG: String (20.0)
ID_SNG: Integer (6.0)

C:\ms4w\tools\gdal-ogr>
```

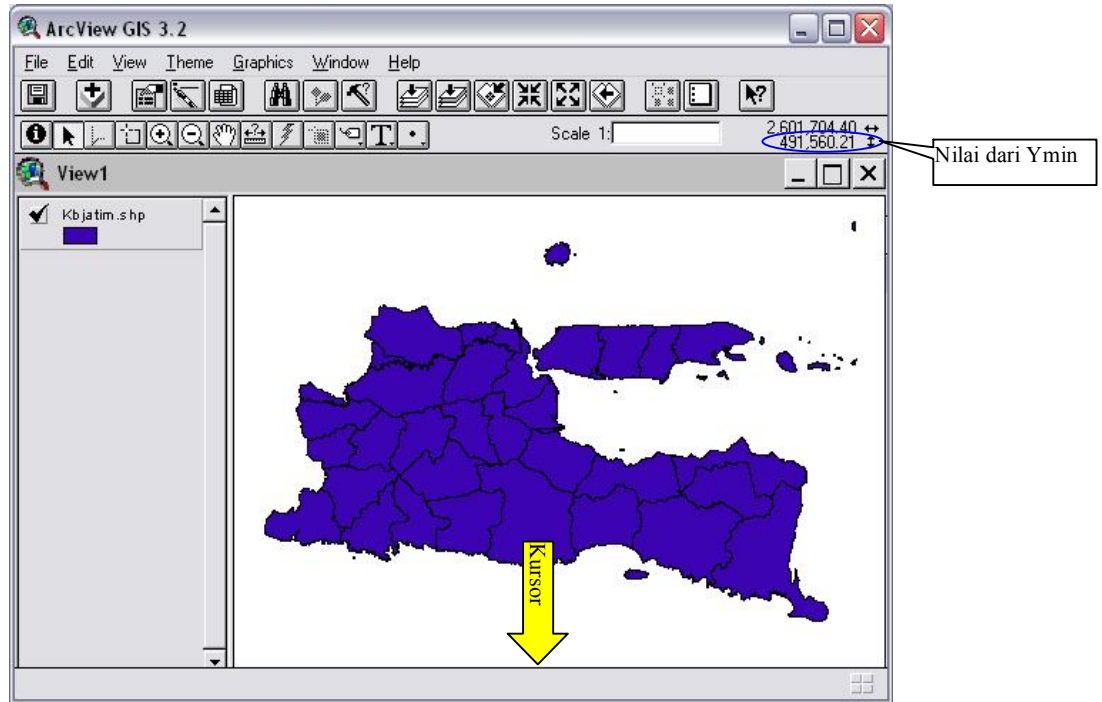
10-06-2008

Cara lain untuk mengetahui batas exten dati fileSHP yaitu dengan cara:

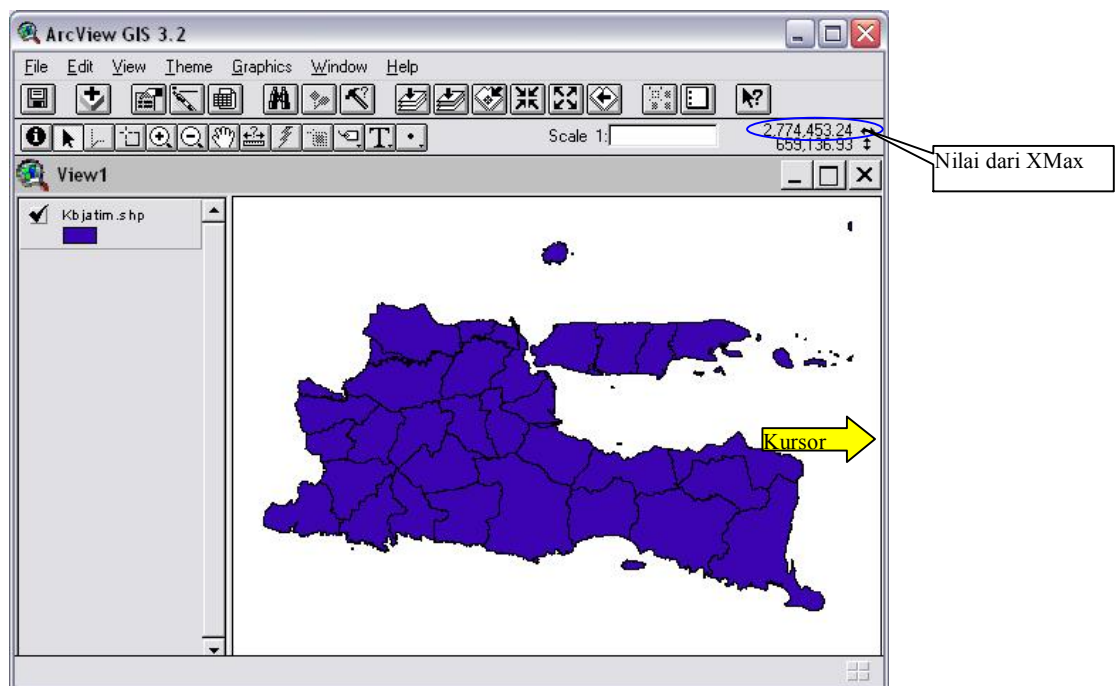
1. Buka aplikasi *.SHP dengan menggunakan ArcView
2. Arahkan kursor pada tepi paling kiri dari area peta
3. Catat nilai yang tertera pada baris pertama di pojok kanan atas sebagai nilai Xmin.



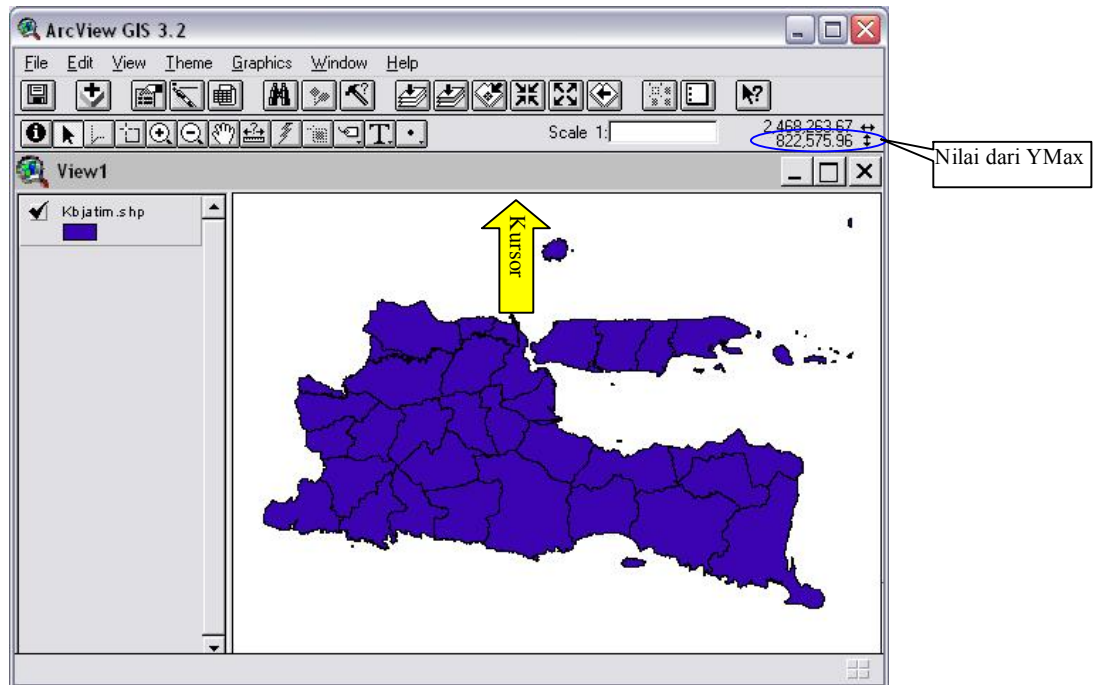
- Arahkan kursor pada tepi paling bawah pada area peta untuk menentukan nilai Ymin



- Arahkan kursor pada tepi paling kanan pada area peta untuk menentukan nilai Xmax



6. Arahkan kursor pada tepi paling atas pada area peta untuk menentukan nilai Ymax



Cara yang terakhir merupakan cara yang manual tapi pasti

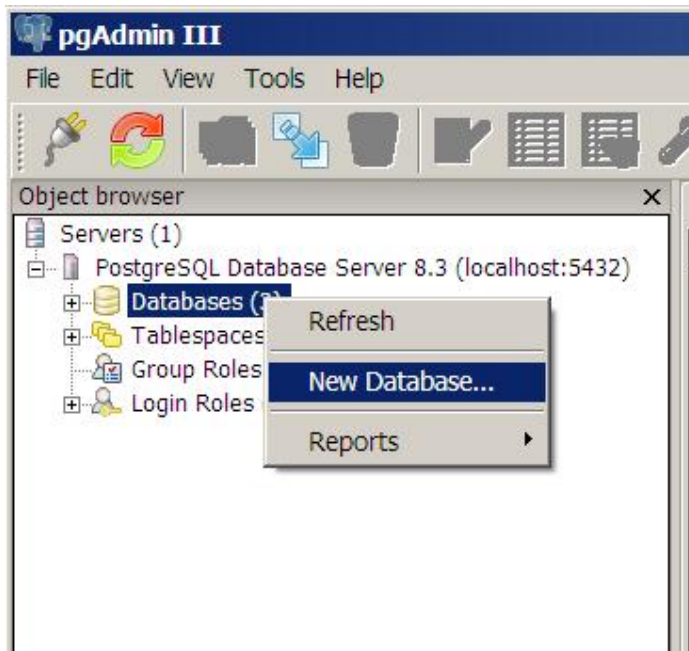
LAMPIRAN B
Tutorial PgRouting
(Dijkstra)

Sumber: http://www.utdallas.edu/~ama054000/rt_tutorial.html

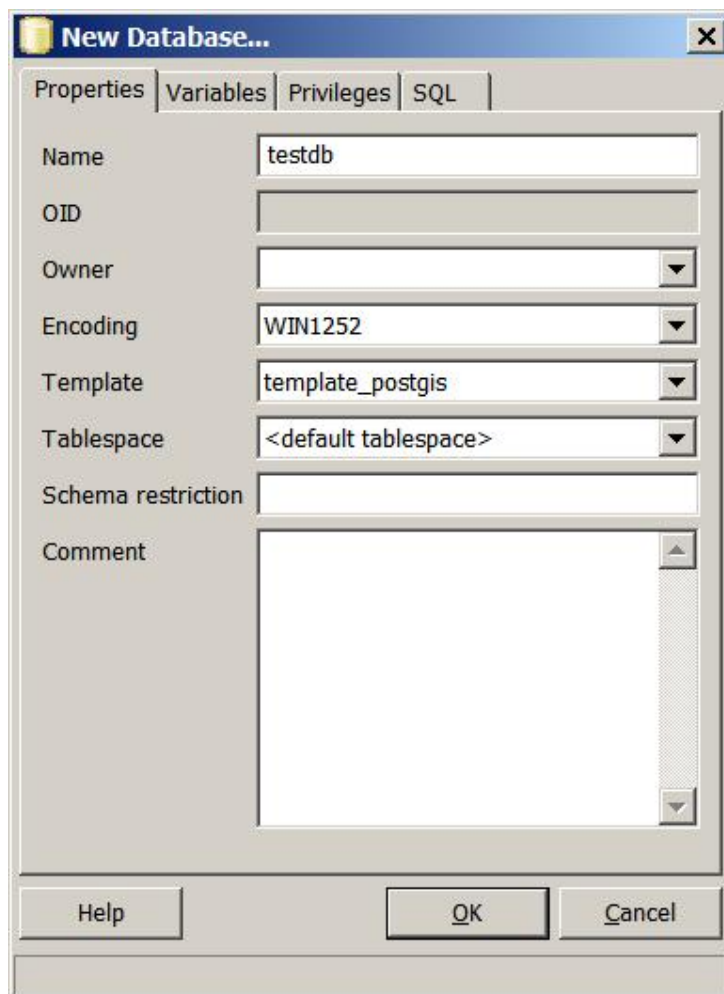
pgRouting Tutorial

Step 1: Create a database called "testdb" using pgAdmin III and add PostGIS functionality via the default template.

- Open pgAdmin III from the Windows Start Menu ("Start->Programs->PostgreSQL 8.3->pgAdmin III").
- Connect to your database by double clicking it in the object browser. You may need to enter password information.
- In pgAdmin III, right click on "Databases" in the table and click "New Database...".



- Name the database "testdb" and for the template, select "template_postgis".
- Click "OK".



Step 2: Add the core pgRouting functionality to the newly created database.

- In pgAdmin III, select the newly created "testdb" database in the object browser.
- Look at the top toolbar in pgAdmin III. There is a SQL query tool. Click on this tool to open it, or click "Tools->Query Tool" from the application menu.
- In the SQL query tool window, click "File->Open" and select "C:\Program Files\PostgreSQL\8.3\share\contrib\routing_core.sql" (the directory structure may differ depending on where you installed your software).

- To execute this query, click the “play” button or navigate the application menu by clicking “Query->Execute”.
- Repeat the same process for “C:\Program Files\PostgreSQL\8.3\share\contrib\routing_core_writers.sql”.
- Now the routing functionality is available to “testdb”.

Step 3: Convert the crude polyline shapefile (edges.shp) to the same crude PL/pgSQL file so that it can be imported to the database.

- From the Windows command prompt, make sure that “C:\Program Files\PostgreSQL\8.3\bin>” is displayed as your current directory location. An alternate option would be to add that directory to the system path.
- Enter this command, replacing variables where necessary:

```
shp2pgsql -s 2276 -i -I "C:\edges.shp" edges >
"C:\edges.sql"
```

- The successful execution of this command should output this to the screen:

```
Shapefile type: Arc
Postgis type: MULTILINESTRING[2]
```

- NOTE: In the above command, “-s 2276” sets the spatial reference (SRID) for the dataset, which happens to be “NAD83 / Texas North Central (ftUS)”. How do we find the SRID for our preferred projection?
 - In pgAdmin III, look at the top toolbar. There is a SQL query tool.
 - Make sure that a PostGIS-enabled database is selected (i.e. “template_postgis” or “testdb”), and click on the SQL query tool to bring up the query window (or click “Tools->Query Tool”).
 - Enter the following code in the SQL query tool to find all spatial reference systems that have “Texas” in the WKT (Well-Known Text) definition:

```
SELECT *
FROM spatial_ref_sys
WHERE srttext LIKE ('%Texas%');
```

- To execute this query, click the “play” button or navigate the application menu by clicking “Query->Execute”.
- This SQL query will return a list of spatial references with SRIDs. Take your pick.

Step 4: Import the crude SQL file to the database as a new table using the SQL query tool.

- In the SQL Query tool window, click “File->Open” and navigate to your newly created SQL file (“C:\edges.sql”) and click “Open”.
- Click the “play” button (or “Query->Execute”) to create your new table in the database.
- You may have to refresh the view to see the “edges” table in the “testdb” database.

Step 5: Prepare the new table for Dijkstra by adding source, target, and length columns. In this example “length” will be the cost of the edges.

- In the SQL query tool, enter the following code and execute:

```
ALTER TABLE edges ADD COLUMN source integer;
ALTER TABLE edges ADD COLUMN target integer;
ALTER TABLE edges ADD COLUMN length double
precision;
```

Step 6: Create the network topology in the “edges” table. Also populate the “length” field which is to be the edge cost in the network topology.

- In the SQL query tool, enter the following code and execute:

```
SELECT          assign_vertex_id('edges',          0.001,
'the_geom', 'gid');
UPDATE edges SET length = length(the_geom);
```

Step 7: Create indices for source, target, and geometry columns in the “edges” table.

- In the SQL query tool, enter the following code and execute:

```
CREATE INDEX source_idx ON edges(source);
CREATE INDEX target_idx ON edges(target);
```

```
CREATE INDEX geom_idx ON edges USING GIST(the_geom
GIST_GEOMETRY_OPS);
```

Step 8: Perform the routing operation and store the results in a “dijkstra_result” table.

- In the SQL query tool, enter the following code and execute:

```
DROP TABLE IF EXISTS dijsktra_result;
CREATE TABLE dijsktra_result(gid int4) with oids;
SELECT      AddGeometryColumn('dijsktra_result',
'the_geom', '2276', 'MULTILINESTRING', 2);
INSERT INTO dijsktra_result(the_geom)
SELECT the_geom FROM dijsktra_sp('edges',
52, 35);
```

- Now the results can be viewed in uDig (or any other GIS software that can view PostGIS layers) by connecting to the database and adding the “dijkstra_result” table.
- Here is an example of the results viewed in uDig (start node: 52, end node: 35):

