

**RANCANG BANGUN APLIKASI SISTEM INFORMASI
GEOGRAFIS BERBASIS WEB UNTUK PEMETAAN PONDOK
PESANTREN DI JOMBANG**

SKRIPSI



Disusun Oleh :

AHMAD HANAFI

09.18.902

**JURUSAN TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2014**

LEMBAR PERSETUJUAN DAN PENGESAHAN

RANCANG BANGUN APLIKASI SISTEM INFORMASI
GEOGRAFIS BERBASIS WEB UNTUK PEMETAAN PONDOK
PESANTREN DI JOMBANG


SKRIPSI

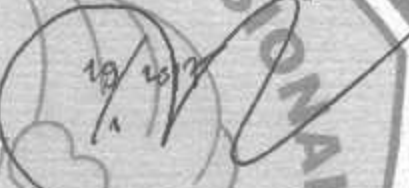
Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik Informatika Strata Satu (S-1)

Disusun Oleh :
AHMAD HANAFI
NIM : 09.18.902

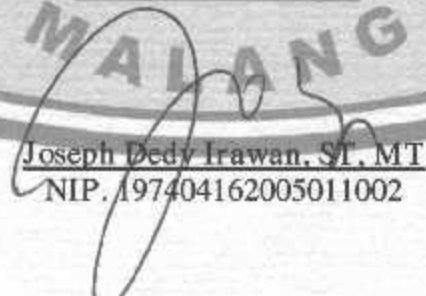
Dosen Pembimbing I

Dosen Pembimbing II


Dr. Aryuanto Soetedjo, ST, MT
NIP.P. 1030800417


Nurlaily Vandyansyah, ST

Diperiksa dan Disetujui
Mengetahui
Ketua Prodi Teknik Informatika S-1


Joseph Dedy Irawan, ST, MT
NIP. 197404162005011002

PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2014



LEMBAR KEASLIAN

PROGRAM STUDI TEKNIK INFORMATIKA S1 FAKULTAS TEKNOLOGI INDUSTRI INSTITUT TEKNOLOGI NASIONAL MALANG

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertandatangan di bawah ini :

Nama : AHMAD HANAFI

NIM : 09.18.902

Program Studi : Teknik Informatika S-1

Fakultas : Teknologi Industri

Menyatakan bahwa skripsi yang berjudul :

**” RANCANG BANGUN APLIKASI SISTEM INFORMASI
GEOGRAFIS BERBASIS WEB UNTUK PEMETAAN PONDOK
PESANTREN DI JOMBANG”**

Adalah skripsi saya sendiri, bukan duplikat, dan seluruh sumber yang dikutip maupun dirujuk telah saya nyatakan dengan sebenarnya.

Malang, Agustus 2014

Yang membuat pernyataan



AHMAD HANAFI

**RANCANG BANGUN APLIKASI SISTEM INFORMASI GEOGRAFIS
BERBASIS WEB UNTUK PEMETAAN PONDOK PESANTREN DI JOMBANG**

AHMAD HANAFI (09.18.902)

**Teknik Informatika S-1, Institut Teknologi Nasional Malang
Email : aficool19@yahoo.com**

Dosen Pembimbing : I. Dr. Aryunto Soetedjo, ST, MT

II. Nurlaily Vendyansyah, ST.

Abstrak

Sistem Informasi Geografis Pondok Pesantren di Jombang ini merupakan suatu sistem yang dirancang dan dibuat dimana akan dihasilkan suatu produk informatif yang bermanfaat bagi masyarakat umum, khususnya santriwan dan santriwati dalam mendapatkan informasi tentang lokasi pesantren di Jombang.

Sistem Informasi Geografis ini dibuat dengan menggunakan Arcview sebagai software untuk membuat peta digital, menggunakan *database postgresSQL*, *ms4w* sebagai *mapserver* dengan *chameleon* sebagai *framework* menggunakan bahasa pemrograman *php* dan *html*. Dalam proses penambahan informasi hanya bisa dilakukan oleh admin yang sudah teregistrasi dalam sistem. Informasi yang diberikan dalam Sistem Informasi Geografis Pondok pesantren Jombang ini berupa informasi pondok pesantren yang dikelompokkan berdasarkan jenisnya menjadi tiga bagian, yaitu pemetaan pondok pesantren putra, pondok pesantren putri, dan pondok pesantren putra dan putri

Pengujian yang dilakukan adalah pengujian fungsi dengan menggunakan browser *MozillaFirefox* dan *Opera*, didapatkan 100% berjalan sesuai fungsinya. Pengujian user dilakukan kepada 10 responden dan didapatkan hasil 100% menyatakan bahwa aplikasi dapat berfungsi sebagai penunjang informasi lokasi pondok pesantren di Jombang.

Kata kunci : Sistem Informasi Geografis, Pondok Pesantren, Jombang, *Arcview*, *Postgresql*, *Mapserver*, *Chameleon*, Pemrograman *Php*.

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT atas karunia, rahmat dan hidayahNya, sehingga penulis dapat menyelesaikan penelitian yang berjudul **“RANCANG BANGUN APLIKASI SISTEM INFORMASI GEOGRAFIS BERBASIS WEB UNTUK PEMETAAN PONDOK PESANTREN DI JOMBANG”**.

Skripsi ini dapat terselesaikan tidak terlepas dari dukungan berbagai pihak. Penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Bapak Ir. Soeparno Djiwo, MT selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. H. Anang Subardi, MT selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Bapak Joseph Dedy Irawan, ST, MT selaku Kepala Jurusan Teknik Informatika S-1 Institut Teknologi Nasional Malang.
4. Bapak Dr. Aryuanto Soetedjo, ST, MT selaku Dosen Pembimbing I yang telah memberikan saran dan bimbingannya dalam penyusunan laporan ini.
5. Ibu Nurlaily Vendyansyah, ST selaku Dosen Pembimbing II yang telah memberikan saran dan bimbingannya dalam penyusunan laporan ini.
6. Bapak dan Ibu Dosen yang telah mengajar penulis selama studi di Institut Teknologi Nasional Malang.
7. Rekan-rekan Teknik Informatika serta berbagai pihak yang turut membantu dalam penyelesaian laporan ini.
8. Untuk Orang Tua dan Keluarga yang telah mendukung saya sampai sekarang.

Semoga apa yang telah disajikan dapat memberikan manfaat dan pengetahuan bagi para pembaca. Segala kritik dan saran yang bersifat membangun, diterima dengan senang hati sebagai tambahan ilmu pengetahuan.

Malang, Agustus 2014

Penulis

DAFTAR ISI

| | |
|---|----------|
| LEMBAR PERSETUJUAN | |
| ABSTRAKSI | |
| KATA PENGANTAR | i |
| DAFTAR ISI | iii |
| DAFTAR GAMBAR | vi |
| DAFTAR TABEL | x |
| | |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 1 |
| 1.3 Batasan Masalah..... | 2 |
| 1.3 Tujuan Penelitian | 2 |
| 1.5 Metodologi..... | 2 |
| 1.6 Sistematika Penulisan | 3 |
| | |
| BAB II DASAR TEORI | 5 |
| 2.1 Sistem Informasi Geografis | 5 |
| 2.1.1 Pengertian Sistem Informasi Geografis | 5 |
| 2.1.2 Macam-macam Data dalam SIG | 5 |
| 2.1.3 Software SIG..... | 6 |
| 2.2 ARCVIEW | 6 |
| 2.3 BASIS DATA | 7 |
| 2.3.1 Data dan Informasi | 9 |

| | | |
|----------------|--|-----------|
| 2.3.2 | Pengertian Basis Data | 9 |
| 2.3.3 | Entity Relationship Diagram (ERD) | 10 |
| 2.3.4 | Database Management System (DBMS) | 12 |
| 2.3.5 | PostgreSQL | 13 |
| 2.3.6 | PostGIS | 13 |
| 2.4 | Data Flow Diagram (DFD) | 13 |
| 2.5 | Flowchart | 15 |
| 2.6 | MapServer | 16 |
| 2.6.1 | Pengenalan MapServer | 16 |
| 2.6.2 | Arsitektur MapServer | 17 |
| 2.6.3 | Chameleon | 18 |
| BAB III | ANALISA DAN PERANCANGAN SISTEM..... | 20 |
| 3.1. | Analisa Sistem | 20 |
| 3.1.1 | Deskripsi Sistem | 20 |
| 3.1.2 | Pengguna Sistem | 20 |
| 3.1.3 | Spesifikasi Kebutuhan Sistem | 20 |
| 3.2 | Perancangan Sistem | 21 |
| 3.2.1 | Desain Sistem | 21 |
| 3.2.1.1 | Work Breakdown System (WBS) | 22 |
| 3.2.1.2 | Flowchart | 23 |
| 3.2.1.3 | Data Flow Diagram (DFD) | 24 |
| 3.2.2 | Desain Database | 26 |
| 3.2.2.1 | Entity Relationship Diagram (ERD) | 26 |
| 3.2.2.2 | Tabel Database | 26 |
| 3.2.3 | Desain Interface / Antarmuka..... | 28 |
| 3.2.3.1 | Tampilan Utama User | 28 |

| | |
|--|-----------|
| 3.2.3.2 Login Admin | 28 |
| 3.2.3.3 Tampilan Utama Admin | 29 |
| 3.2.3.4 Manipulasi Data Pesantren | 29 |
| 3.2.3.5 Tampilan Insert | 30 |
| 3.2.3.6 Tampilan Edit | 30 |
| BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM..... | 31 |
| 4.1 Implementasi Sistem | 31 |
| 4.1.1 Pembuatan Peta Digital | 31 |
| 4.1.2 Pembuatan Database | 39 |
| 4.1.3 Pembuatan Struktur File *.map..... | 43 |
| 4.1.4 Pembuatan Struktur Template | 44 |
| 4.2 Pengujian | 45 |
| 4.2.1 Pengujian Sistem..... | 45 |
| 4.2.1.1 Pengujian terhadap User | 45 |
| 4.2.1.2 Pengujian terhadap Admin | 52 |
| 4.2.2 Pengujian Browser | 57 |
| 4.2.3 Pengujian User | 59 |
| BAB V PENUTUP | 60 |
| 5.1 Kesimpulan | 60 |
| 5.2 Saran | 60 |
| DAFTAR PUSTAKA..... | 61 |

DAFTAR GAMBAR

| | | |
|-------------|---|----|
| Gambar 2.1 | Logo ArcView | 7 |
| Gambar 2.2 | Tampilan ArcView pertama kali dijalankan | 7 |
| Gambar 2.3 | Tampilan Project ArcView pada Notepad | 11 |
| Gambar 2.4 | <i>Entity</i> | 10 |
| Gambar 2.5 | Notasi Atribut | 11 |
| Gambar 2.6 | <i>One to One Relationship</i> | 11 |
| Gambar 2.7 | <i>One to Many Relationship</i> | 11 |
| Gambar 2.8 | <i>Many to Many Relationship</i> | 11 |
| Gambar 2.9 | <i>Mandaroty</i> | 11 |
| Gambar 2.10 | <i>Data Source</i> | 14 |
| Gambar 2.11 | <i>Data Flow</i> | 14 |
| Gambar 2.12 | Proses | 14 |
| Gambar 2.13 | <i>Data Stores</i> | 14 |
| Gambar 2.14 | Contoh <i>Flowchart</i> | 15 |
| Gambar 2.1 | Simbol <i>Flowchart</i> | 15 |
| Gambar 2.15 | Arsitektur <i>MapServer</i> | 17 |
| Gambar 2.16 | <i>Konfigurasi Chameleon digunakan dengan MapServer</i> | 18 |
| Gambar 3.1 | Rancangan Desain Tampilan Aplikasi | 22 |
| Gambar 3.2 | Flowchart <i>Admin</i> | 23 |
| Gambar 3.3 | Flowchart <i>User</i> | 24 |
| Gambar 3.4 | <i>Data Flow Diagram Level 0</i> | 24 |
| Gambar 3.5 | <i>Data Flow Diagram Level 1</i> | 25 |
| Gambar 3.6 | <i>Entity Relationship Diagram</i> | 26 |

| | | |
|-------------|---|----|
| Gambar 3.7 | Desain Tampilan Utama <i>User</i> | 28 |
| Gambar 3.8 | Desain Tampilan Login <i>Admin</i> | 28 |
| Gambar 3.9 | Desain Tampilan Utama (<i>Home</i>) <i>Admin</i> | 29 |
| Gambar 3.10 | Desain Tampilan Manipulasi Data Pesantren | 29 |
| Gambar 3.11 | Desain Tampilan Insert Pesantren | 30 |
| Gambar 3.12 | Desain Tampilan Edit Pesantren | 30 |
| Gambar 4.1 | Proses Pembuatan Peta Digital | 31 |
| Gambar 4.2 | Jendela “Welcome to Arcview GIS” | 32 |
| Gambar 4.3 | Jendela “Add Data” | 32 |
| Gambar 4.4 | Jendela “Add Theme” | 32 |
| Gambar 4.5 | File Peta Jombang pada Arcview | 33 |
| Gambar 4.6 | Mengaktifkan Ekstensi Register and Transform Tool | 33 |
| Gambar 4.7 | Proses Registrasi Peta | 34 |
| Gambar 4.8 | Memasang Titik GCP | 34 |
| Gambar 4.9 | Menyimpan File Koordinat GCP | 35 |
| Gambar 4.10 | New Theme | 36 |
| Gambar 4.11 | Feature Type | 36 |
| Gambar 4.12 | Menyimpan Theme | 36 |
| Gambar 4.13 | Menu Drop Down | 37 |
| Gambar 4.14 | Theme kecamatan.shp setelah dibagi menjadi 20 bagian..... | 37 |
| Gambar 4.15 | Menambahkan Tabel dan <i>Field</i> | 38 |
| Gambar 4.16 | Input Data pada Theme Kecamatan..... | 38 |
| Gambar 4.17 | Input Data pada Theme Jalan | 38 |
| Gambar 4.18 | Input Data pada Theme Pesantren Putra | 39 |
| Gambar 4.19 | Input Data pada Theme Pesantren Putri | 39 |

| | | |
|-------------|---|----|
| Gambar 4.20 | Input Data pada Theme Pesantren Putra-Putri | 39 |
| Gambar 4.21 | Membuat User Baru | 40 |
| Gambar 4.22 | Membuat Database Baru | 41 |
| Gambar 4.23 | Konversi File *.shp menjadi *.sql | 42 |
| Gambar 4.24 | Query Tools | 42 |
| Gambar 4.25 | Hasil Konversi Database | 43 |
| Gambar 4.26 | Hasil Instalasi ms4w | 43 |
| Gambar 4.27 | Instalasi Chameleon | 44 |
| Gambar 4.28 | Hasil Instalasi Chameleon | 44 |
| Gambar 4.29 | Tampilan Awal Aplikasi | 46 |
| Gambar 4.30 | Tampilan peta setelah dilakukan Zoom In..... | 47 |
| Gambar 4.31 | Tampilan peta setelah dilakukan Zoom Out | 47 |
| Gambar 4.32 | Tampilan peta setelah dilakukan Zoom to Full Extents | 48 |
| Gambar 4.33 | Tampilan peta setelah dilakukan Pan Map | 48 |
| Gambar 4.34 | Tampilan peta setelah dilakukan Identify Feature | 49 |
| Gambar 4.35 | Tampilan peta setelah dilakukan Map Size "Small" | 50 |
| Gambar 4.36 | Tampilan peta setelah dilakukan Map Size "Medium" | 51 |
| Gambar 4.37 | Tampilan peta setelah dilakukan Map Size "Large"..... | 52 |
| Gambar 4.38 | Tampilan Login Admin | 52 |
| Gambar 4.39 | Input Username & Password | 53 |
| Gambar 4.40 | Peringatan Salah <i>Username & Password</i> | 53 |
| Gambar 4.41 | Tampilan Awal Admin | 54 |
| Gambar 4.42 | Tampilan Manipulasi Informasi Pesantren Putra | 55 |
| Gambar 4.43 | Tampilan Manipulasi Informasi Pesantren putri | 55 |
| Gambar 4.44 | Tampilan Manipulasi Informasi putra putri | 56 |

| | | |
|-------------|---|----|
| Gambar 4.45 | Edit Tabel Pesantren putra - putri..... | 56 |
| Gambar 4.46 | Insert Tabel Pesantren | 57 |
| Gambar 4.47 | Hasil Insert Data Pesantren | 57 |

DAFTAR TABEL

| | | |
|-----------|---|----|
| Tabel 3.1 | Tabel <i>Database</i> Pesantren Putra | 26 |
| Tabel 3.2 | Tabel <i>Database</i> Pesantren Putri | 27 |
| Tabel 3.3 | Tabel Database Pesantren Putra-Putri | 27 |
| Tabel 3.4 | Tabel Database Jalan | 27 |
| Tabel 3.5 | Tabel Database Admin | 27 |
| Tabel 4.1 | Tabel Layer | 35 |
| Tabel 4.2 | Tabel Pengujian <i>Browser</i> | 58 |
| Tabel 4.3 | Hasil Pengujian User | 59 |

BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Sistem Informasi Geografis (SIG) merupakan suatu sistem pengolahan dan pengelolaan informasi geografis berupa peta *digital*. Semakin pesatnya perkembangan teknologi di zaman ini, menjadikan peta konvensional terlihat ketinggalan zaman, dan tidak efisien jika dibandingkan dengan peta *digital*.

Pencarian suatu tempat dengan menggunakan peta konvensional dirasakan masih menyusahakan bagi sebagian orang. Hal ini terkait dengan ukuran peta yang relatif besar sehingga kurang efektif dalam penggunaannya. Selain itu di dalam peta tidak terdapat simbol-simbol untuk bangunan-bangunan tertentu. Untuk itu dibutuhkan suatu pemetaan berupa proses pengukuran, perhitungan dan penggambaran permukaan bumi (terminologi geodesi) menggunakan metode tertentu sehingga didapatkan hasil berupa peta digital.

Kabupaten Jombang merupakan salah satu kabupaten yang mendapat julukan sebagai kota santri di Jawa Timur dengan cukup banyak pondok pesantren andalan. Salah satu pondok pesantren di Kabupaten Jombang adalah pondok pesantren Tebu Ireng, disamping menyajikan pelajaran mengenai pengetahuan agama Islam, ilmu syariat, dan bahasa Arab, pelajaran umum juga dimasukkan ke dalam struktur kurikulum pengajarannya. Pesantren Tebu Ireng telah banyak memberikan kontribusi dan sumbangan kepada masyarakat luas baik, terutama dalam dunia pendidikan Islam di Indonesia. Sektor pesantren yang beragam dan banyaknya dan didukung dengan fasilitas serta sarana transportasi yang tersedia di kawasan sekitar pesantren dapat memberikan *income* pemerintah yang cukup besar. Pemerintah Kabupaten Jombang telah melakukan promosi melalui media masa seperti surat kabar dan pamflet. Namun metode tersebut belum cukup untuk menginformasikan pondok pesantren secara meluas kepada calon pelajar lokal maupun interlokal. Para calon santri akan mengalami kesulitan untuk menentukan pilihan pondok pesantren karena gambaran daerah pesantren sekaligus fasilitas yang mendukung pada pesantren tersebut tidak tersedia seperti visualisasi tempat, jarak antar daerah, serta jalan yang akan dilalui. Oleh karena itu melalui perancangan dan pembuatan SIG pondok pesantren diharapkan dapat menampilkan gambaran peta pondok pesantren Kabupaten Jombang sehingga lebih menarik dan dapat dimanfaatkan oleh masyarakat luas. Penyajian informasi dalam

bentuk web akan memudahkan masyarakat untuk mengaksesnya.

1.2 RUMUSAN MASALAH

Berdasarkan latar belakang tersebut timbul suatu permasalahan, yaitu : bagaimana menampilkan lokasi beserta informasi yang berkaitan dengan objek pondok pesantren di kabupaten Jombang berbasis web pada sebuah peta tematik.

1.3 BATASAN MASALAH

Adapun batasan masalah dalam penyusunan skripsi ini adalah :

1. Merupakan sebuah aplikasi SIG objek Pondok Pesantren berbasis web,
2. Layanan yang diberikan adalah penunjukan lokasi tempat-tempat pondok pesantren di kabupaten Jombang,
3. Wilayah yang dicakup adalah kabupaten kabupaten Jombang,
4. Menggunakan *software* SIG yaitu *ArcView*, aplikasi *MapServer* untuk menampilkan peta di web dengan bahasa pemrograman HTML, PHP, dan *database* PostgreSQL, serta menggunakan browser *Mozilla* dan *Opera* pada proses pengujian.

1.4 TUJUAN

Pembuatan aplikasi SIG pondok pesantren di kabupaten Jombang ini memiliki beberapa tujuan, seperti di bawah ini :

1. Mempermudah publikasi objek pondok pesantren yang ada di kabupaten Jombang,
2. Menghasilkan produk yang informatif untuk mempermudah masyarakat khususnya calon santri lokal maupun interlokal dalam mendapatkan informasi tentang objek pondok pesantren di kabupaten Jombang,
3. Merupakan skripsi yang digunakan sebagai syarat untuk menyelesaikan studi S-1 Teknik Informatika di Institut Teknologi Nasional Malang.

1.5 METODOLOGI

Metodologi yang digunakan dalam penyusunan skripsi ini adalah :

1. Studi Literatur
Penelitian ini dilakukan dengan mempelajari buku-buku baik dari perpustakaan, internet, maupun milik pribadi.

2. Analisa Kebutuhan
 - a. *Hardware*
 1. Komputer
 - b. *Software*
 1. ArcView
 2. MapServer
 3. PostgreSQL
 4. Browser : Mozilla , Opera
3. Perancangan Sistem
Meliputi perancangan desain sistem, desain *database* dan desain *interface*.
4. Pembuatan Sistem
Tahap pembuatan sistem ini merupakan implementasi dari desain sistem yang telah direncanakan sebelumnya. Pada tahap ini meliputi tahap pembuatan *interface* dan fitur yang disediakan di dalam sistem ini.
5. Pengujian dan Analisis
Tahapan pengujian sistem merupakan tahap untuk melakukan pengujian terhadap sistem yang telah dibuat, sekaligus melakukan analisa untuk melakukan perbaikan apabila ada kekurangan dan kesalahan terhadap sistem yang telah dibuat.
6. Penarikan Kesimpulan
Pada tahap ini akan dibuat kesimpulan tentang sistem yang sudah dibuat berdasarkan perencanaan dan analisa yang sudah dilakukan.

1.6 SISTEMATIKA PENULISAN LAPORAN

Sistematika penyusunan skripsi ini dibagi menjadi 5 (lima) bab yaitu pendahuluan, dasar teori, analisa dan perancangan sistem, implementasi dan pengujian, serta penutup.

BAB 1 PENDAHULUAN

Bab ini berisi latar belakang, rumusan masalah, batasan masalah, tujuan, metodologi, dan sistematika penulisan laporan.

BAB 2 DASAR TEORI

Bab ini berisi tentang penjelasan teori-teori sebagai referensi dalam pembuatan aplikasi.

BAB 3 ANALISA DAN PERANCANGAN SISTEM

Bab ini berisi tentang perancangan sistem berupa analisa dan desain sistem. Proses analisa menjelaskan tentang deskripsi sistem, pengguna sistem, dan spesifikasi kebutuhan sistem. Sedangkan pada tahap desain dibagi menjadi desain dasar sistem secara keseluruhan.

BAB 4 IMPLEMENTASI DAN PENGUJIAN

Pada bab ini membahas tentang implementasi dan pengujian sistem yang dibuat secara keseluruhan.

BAB 5 PENUTUP

Berisi tentang kesimpulan dan saran yang didapat setelah melakukan analisa, desain, implementasi dan pengujian terhadap pembuatan sistem.

BAB II DASAR TEORI

2.1 SISTEM INFORMASI GEOGRAFIS (SIG)

2.1.1 Pengertian Sistem Informasi Geografis

Sistem Informasi Geografis (*Geographic Information System/GIS*) merupakan sistem informasi berbasis komputer yang digunakan untuk mengolah dan menyimpan data atau informasi geografis. SIG adalah sistem informasi yang digunakan untuk memasukkan, menyimpan, memanggil kembali, mengolah, menganalisa dan menghasilkan data bereferensi geografis atau data spasial.

Teknologi Sistem Informasi Geografis dapat digunakan untuk investigasi ilmiah, pengelolaan sumber daya, perencanaan pembangunan, kartografi dan perencanaan rute. Misalnya, SIG bisa membantu perencana untuk secara cepat menghitung waktu tanggap darurat saat terjadi bencana alam, atau SIG dapat digunakan untuk mencari lahan basah (*wetlands*) yang membutuhkan perlindungan dari polusi. [1]

Dengan menggunakan SIG, di dapat keuntungan sebagai berikut : [1]

- a. Penanganan data *geospatial* menjadi lebih baik,
- b. Merubah dan memperbaharui data menjadi lebih mudah,
- c. Data *geospatial* lebih mudah dianalisis dan direpresentasikan,
- d. Informasi lebih mudah dicari,
- e. Menjadi produk bernilai tambah,
- f. Data *geospatial* dapat dipertukarkan,
- g. Penghematan waktu dan biaya,
- h. Keputusan yang akan diambil menjadi lebih baik.

2.1.2 Macam-macam Data dalam SIG

Data SIG dapat dibagi menjadi dua macam, yaitu data grafis dan data atribut atau tabular. Data grafis adalah data yang menggambarkan bentuk atau kenampakan objek di permukaan bumi. Sedangkan data tabular adalah data deskriptif yang menyatakan nilai dari data grafis tersebut.[1]

a) **Data Grafis**

Secara garis besar, data grafis dibedakan menjadi tiga macam yaitu data titik (*point*), garis (*line*) dan area (*region* atau *polygon*). Data grafis titik biasanya digunakan untuk mewakili objek kabupaten, bangunan, kondisi lahan. Data garis

dapat dipakai untuk menggambarkan jalan, sungai, jaringan atau saluran dan lain-lain. Sementara data area digunakan untuk mewakili batas *administrasi*, penggunaan lahan, kemiringan lereng dan lain-lain. Sedangkan struktur data SIG ada dua macam yaitu vector dan raster. Pada struktur data vector posisi objek dicatat pada system koordinat. Sedangkan objek pada struktur data raster disimpan pada grid dua dimensi yaitu baris dan kolom.

b) **Data Atribut**

Data atribut atau tabular merupakan data yang menyimpan informasi mengenai nilai atau besaran dari data grafis. Untuk struktur data vector, data atribut tersimpan secara terpisah dalam bentuk tabel. Sementara pada stuktur data raster nilai data grafis-nya tersimpan langsung pada nilai grid atau piksel tersebut.

2.1.3 *Software* SIG

Software yang mendukung SIG ini, diantaranya adalah sebagai berikut : [2]

a) **MapInfo**

MapInfo Professional adalah Sistem Informasi Geografis terkemuka, yang digunakan untuk analisis georeferensi untuk menghasilkan sebuah peta atau jenis lain bentuk data spasial.

Baru-baru ini, MapInfo Profesional tidak hanya digunakan sebagai alat untuk menganalisis data spasial sumber daya alam, tetapi juga diterapkan di hampir semua bidang seperti ekonomi dan perdagangan, sosial dan budaya, pondok pesantren, polisi, selular, layanan, dan lainnya.

b) **ArcGIS**

Penggunaan ArcGIS di semua jenis organisasi untuk meningkatkan alur kerja dan memecahkan masalah-masalah yang paling menantang.

c) **ArcView**

ArcView adalah sistem informasi geografis untuk visualisasi, mengelola, menciptakan dan menganalisa data geografis, memahami konteks geografis data, dan memungkinkan untuk melihat hubungan dan mengidentifikasi pola-pola dalam cara-cara baru.

Ada beberapa *software* yang dapat digunakan dalam proses pengolahan data pada sistem informasi geografis seperti pada uraian di atas. Namun dalam proses pembuatan aplikasi kali ini akan digunakan *software* SIG ArcView dalam proses pembuatan peta digital.

2.2 ARCVIEW



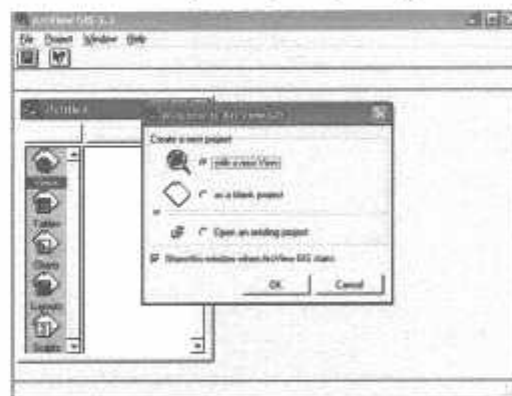
Gambar 2.1 Logo ArcView [3]

Arcview adalah salah satu perangkat lunak SIG yang paling populer dan paling banyak digunakan untuk mengolah data spasial. *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. Arcview terdiri dari 6 modul utama yang disebut dokumen, yaitu : *Project, View, Table, Chart, Layout, dan Script*. Masing-masing dokumen tersebut mempunyai GUI (*Graphical UserInterface*) yang spesifik. GUI adalah tempat pengguna berinteraksi dengan dokumen, misalnya melihat, memilih, memperbesar, mengisi tabel, dsb. Untuk melakukan tujuan tersebut tiap GUI dari tiap dokumen mempunyai *menu, botton*, dan *tools* tersendiri.

a) *Project ArcView*

Apabila telah meng-*install software* ArcView pada komputer, tampilan ArcView saat pertama kali dijalankan akan nampak seperti pada gambar di bawah ini.



Gambar 2.2 Tampilan ArcView pertama kali dijalankan [3]

Pada kotak dialog pembuka *Welcome to ArcView GIS*, ArcView menanyakan kepada kita apakah kita akan membuat proyek dengan membuka view kosong (*with a*

new View), membuka projek kosong tanpa membuka view (*as blank projek*), atau mengambil projek yang pernah disimpan (*Open an existing project*).

Kotak dialog pembuka ini akan selalu ditampilkan setiap kali kita menjalankan ArcView. Bila anda tidak menginginkan kotak dialog ini terus dimunculkan saat mengaktifkan ArcView, hilangkan tanda rumput pada kotak cek *Show this window when ArcView GIS start*.

Setiap projek terdiri dari 5 komponen, yaitu Views, Tables, Charts, Layouts, dan Scripts.

1. **Views** digunakan untuk mengelola data grafis,
2. **Tables** dipakai untuk manajemen data atribut,
3. **Charts** digunakan untuk mengelola grafik (bukan data grafis),
4. **Layouts** dipakai untuk membuat komposisi peta untuk dicetak, dan
5. **Scripts** digunakan untuk membuat modul yang berisikan kumpulan perintah ArcView yang ditulis dalam bahasa pemrograman *Avenue*.

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



```
File Edit Format View Help
v3.2
(CDD,1
  Dependencies: "E:\AVEXT\spatial\src\src\AVEXT\lognode.exe"
  FirstDocClassName: "Project"
  Root: 2
  Version: 33
)

(Project_2
  Name: "Pusatok.ape"
  CreatedDate: "08 October 2004 8:06:24"
  GUIDName: "Project"
  Map: 3
  CDDs: 4
  DocEnter: 3
  ViewWidth: 70
  Doc: 4
  Doc: 182
  Doc: 85
  Doc: 289
  Doc: 314
  Doc: 340
  Doc: 575
  Doc: 583
  Doc: 243
)
```

Gambar 2.3 Tampilan Project ArcView pada Notepad [3]

b) View

Views pada proyek ArcView yang dipakai untuk mengelola data grafis baik vektor 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.

c) **Tables**

Fasilitas *Tables* (tabel) pada proyek ArcView digunakan untuk memajemen data atribut atau tabular. Membuat tabel baru, menambah *field* dan *record*, join antar tabel merupakan beberapa pekerjaan yang dapat dilakukan pada *Tables*. Pada struktur data vektor, 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.

d) **Layouts**

Fasilitas *Layouts* 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 ascoris lainnya.

e) **Scripts**

Scripts pada proyek 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 analisis sesuai dengan kebutuhan. Di samping itu, proses sifatnya yang berulang-ulang dapat dibuatkan *script* sehingga pekerjaan menjadi lebih efektif dan efisien.

2.3 BASIS DATA

2.3.1 Data dan Informasi

Data berupa catatan historis yang dicatat dan diarsipkan tanpa maksud dan segera diambil kembali untuk pengambilan keputusan. Data yang telah diletakkan dalam konteks yang lebih berarti dan berguna yang dikomunikasikan kepada penerima untuk digunakan di dalam pembuatan keputusan disebut informasi. [4]

2.3.2 Pengertian Basis Data

Basis data dapat didefinisikan dalam beberapa sudut pandang, seperti :

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.

2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundansi*) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan *file* / tabel / arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.[5]

Salah satu cara menyajikan data untuk mempermudah modifikasi adalah dengan cara pemodelan data. Model yang akan dipergunakan pada perencanaan aplikasi ini adalah *Entity Relationship Model*.

2.3.3 Entity Relationship Diagram (ERD)

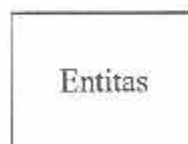
ERD adalah representasi data dari suatu organisasi secara detail, masuk akal dan digambarkan dalam bentuk grafik. *ERD* merupakan model dari *entity* dalam suatu elemen bisnis, relasi antara *entity* dan atribut atau *property* dari *entity* dan relasinya. Berikut adalah simbol yang digunakan dalam membuat suatu *ERD* [6] :

a) *Entity*

Suatu *entity* adalah seseorang, tempat, objek, kejadian atau konsep dalam lingkungan *user*, dimana suatu organisasi membutuhkan untuk memelihara data. *Entity* mempunyai identitas sendiri, dimana antara satu *entity* dengan yang lain berbeda. Beberapa contoh *entity* adalah :

1. Orang : PEGAWAI, PELAJAR, PASIEN
2. Tempat : NEGARA, DAERAH, KABUPATEN
3. Objek : MESIN, BANGUNAN, PRODUK
4. Kejadian : PENJUALAN, PEMBELIAN, REGISTRASI
5. Konsep : AKUNTANSI

Suatu *entity* digambarkan dengan kabupatenk, dimana terdapat nama yang mengidentifikasi *entity* tersebut, seperti pada gambar di bawah ini :



Gambar 2.4 Entity [6]

b) *Attributes*

Setiap *entity* mempunyai beberapa *property* atau karakteristik. Atribut menggambarkan tentang rincian dari *entity* tersebut (atribut yang berfungsi sebagai *key*

diberi garis bawah). Atribut dinotasikan seperti gambar di bawah ini :



Gambar 2.5 Notasi Atribut [6]

c) **Identifier atau Primary Key**

Primary key adalah atribut *unique* yang mengidentifikasi setiap *entity*. *Unique* di sini yang dimaksud adalah dalam *entity* tersebut tidak boleh ada *primary key* yang kembar atau sama.

d) **Relationship**

Relationship adalah relasi atau hubungan antar *entity*. Berikut adalah beberapa istilah dalam penggolongan jenis relasi antar *entity* : [6]

1. **Cardinality**

Menandai jumlah *entity* yang muncul pada *entity* lainnya. Misalkan, terdapat dua *entity* A dan B, *cardinality* adalah angka dari B yang dapat diasosiasikan dengan *entity* A dan sebaliknya. Nilai untuk *cardinality* ini ada dua yaitu *one* atau *many*. Berikut adalah relasi dengan *cardinality* :

a. **One to one**



Gambar 2.6 One to One Relationship [7]

b. **One to many**



Gambar 2.7 One to Many Relationship [7]

c. **Many to many**



Gambar 2.8 Many to Many Relationship [7]

2. **Mandatory**

Sedang *mandatory* adalah untuk menunjukkan apakah *entity* tersebut harus berelasi dengan *entity* lain atau tidak. Jika iya bernilai 1 sedangkan jika tidak maka bernilai 0.



2.3.4 Database Management System (DBMS)

Secara umum DBMS diartikan sebagai suatu program komputer yang digunakan untuk memasukkan, mengubah, menghapus, memanipulasi, dan memperoleh data atau informasi dengan praktis dan efisien. Tujuan DBMS adalah untuk mempermudah penciptaan struktur data dan membebaskan pemrograman dari masalah penyusutan *file* yang kacau.

Dibandingkan dengan sistem yang berbasis kertas, *DBMS* memiliki empat keunggulan, yaitu :[8]

1) Kepraktisan

Sistem yang berbasis kertas akan menggunakan kertas yang sangat banyak untuk menyimpan informasi, sedangkan *DBMS* menggunakan media penyimpanan sekunder yang berukuran kecil tetapi padat informasi. Sehingga *DBMS* memiliki kepraktisan dalam penggunaan media penyimpanan.

2) Kecepatan

DBMS menggunakan mesin yang dapat mengambil atau mengubah data jauh lebih cepat daripada yang dilakukan oleh manusia.

3) Mengurangi kejemuian

Orang cenderung menjadi bosan kalau melakukan tindakan berulang-ulang dengan menggunakan tangan (misalnya harus mengganti suatu informasi). *DBMS* ini dapat mengurangi kejemuian karena penggunaanya tidak terus menerus mengganti suatu informasi dengan cara manual, tetapi pengguna *DBMS* tinggal memberikan perintah kepada komputer untuk menggantinya dengan informasi yang baru.

4) Kekinian

Informasi yang tersedia pada *DBMS* akan bersifat mutakhir dan akurat setiap saat. Kebanyakan *DBMS* menyediakan mekanisme pengaturan keamanan terhadap basis data berdasarkan wewenang pengguna. Sebagai contoh si A hanya boleh membaca suatu data, tetapi si B selain membaca suatu data juga boleh mengubah data tersebut.

Ada beberapa DBMS yang dapat digunakan dalam proses pengolahan data seperti Ms. Access, PostgreSQL, Visual FoxPro, Sybase, MySQL, Oracle, dll. Namun dalam proses pembuatan aplikasi kali ini, kami menggunakan PostgreSQL.

2.3.5 PostgreSQL

PostgreSQL atau sering disebut Postgres merupakan salah satu dari sejumlah *database open source* yang menawarkan skalabilitas, keluwesan, dan kinerja yang tinggi. SQL di PostgreSQL tidaklah seperti yang kita temui pada RDBMS umumnya. Perbedaan antara PostgreSQL dengan sistem relasional standar adalah arsitektur PostgreSQL yang memungkinkan *user* untuk mendefinisikan sendiri SQL-nya, terutama pada pembuatan *function* atau biasa disebut sebagai *stored procedure*. Hal ini dimungkinkan karena informasi yang disimpan oleh PostgreSQL bukan hanya tabel dan kolom, melainkan tipe, fungsi, metode akses, dan banyak lagi yang terkait dengan tabel dan kolom tersebut. Semuanya terhimpun dalam bentuk *object oriented* yang bisa diubah *user*.

Untuk *platform* Windows, PostgreSQL hanya bisa berjalan jika tipe format *harddisk* yang digunakan adalah NTFS, jika tipe format FAT/FAT32 PostgreSQL tidak bisa dijalankan. Sedangkan untuk *platform* yang lain, PostgreSQL bisa berjalan tanpa syarat khusus. [8]

2.3.6 PostGIS

PostGIS adalah *extension* dari PostgreSQL yang bersifat *object relational database server* yang mempunyai kemampuan untuk menyimpan fitur SIG dalam *database server*. PostGIS adalah *software open source* yang tidak perlu membeli lisensi untuk menggunakannya. PostGIS dikembangkan oleh *Refractions Research of Victoria* sebagai proyek penelitian teknologi *database* spasial.

PostGIS mempunyai karakteristik unik tersendiri yang membedakannya dengan *database* yang lain, seperti :

- PostGIS mendukung semua fitur OGC (*Open GIS Consortium*) seperti : titik, garis, *polygon*, *multipoint*, *multiline*, *multipartigon*, dan *Geometry Collection*,
- PostGIS menggunakan teks format OGC dalam perintah SQL untuk merepresentasikan fitur SIG,
- PostGIS menyediakan proses *indexing* secara cepat dengan menggunakan GiST (*Generalized Search Tree*) atau *R-Tree indexes*. [8]

2.4 DATA FLOW DIAGRAM (DFD)

DFD adalah representasi secara grafik untuk menjelaskan aliran data dari suatu

organisasi. *DFD* digunakan untuk mendokumentasikan sistem yang sudah ada. Suatu *DFD* terdiri dari empat elemen utama yaitu data *source* dan tujuan, *data flow*, proses *transformasi* dan penyimpanan data. Setiap elemen digambarkan dengan satu simbol. Keempat simbol ini digabungkan untuk menunjukkan bagaimana suatu data diproses. Elemen tersebut digambarkan dengan simbol sebagai berikut :

a) **Data Source dan Tujuan**

Data *source* dan tujuan menggambarkan suatu organisasi atau individu yang mengirim atau menerima data yang digunakan atau dihasilkan oleh sistem. Sebuah *entity* dapat menjadi sumber atau tujuan dari data. Data *source* dan tujuan digambarkan oleh suatu persegi.



Gambar 2.10 Data Source [8]

b) **Data Flow**

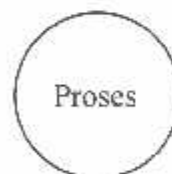
Data *flow* menggambarkan aliran data antarproses, data *stores*, data *sources* dan tujuan. Data *flow* menggambarkan aliran data secara *logical*. Suatu data yang akan melewati data *stores* dan data *sources* atau tujuan harus melewati suatu proses terlebih dahulu yang dihubungkan dengan suatu data *flow*. Data *flow* digambarkan dengan panah dimana untuk setiap aliran data diberi label untuk mengetahui aliran apa yang sedang terjadi dalam suatu kejadian.



Gambar 2.11 Data Flow [8]

c) **Proses**

Suatu proses adalah suatu kejadian yang menyebabkan suatu data ditransformasi, disimpan atau didistribusikan. Proses digambarkan dalam bentuk lingkaran dimana label akan dicantumkan di dalam lingkaran tersebut untuk menunjukkan proses yang dilakukan.



Gambar 2.12 Proses [8]

d) **Data Stores**

Data *stores* adalah tempat penyimpanan data secara permanen maupun *temporary*. *DFD* tidak menunjukkan media penyimpanan secara fisik (kertas dan disk) untuk

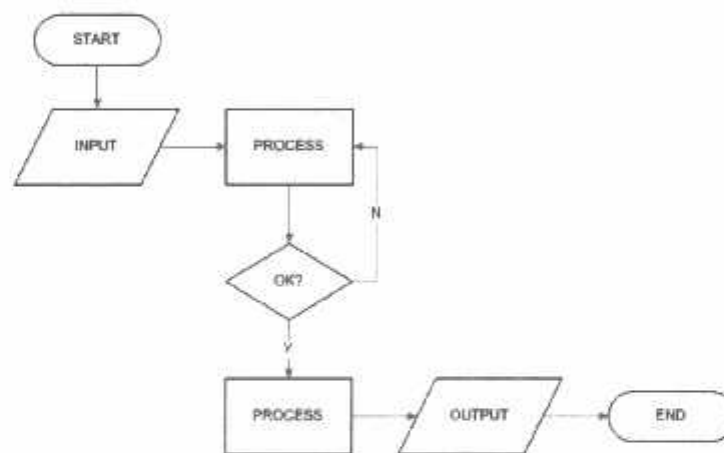
menyimpan data. Nama dari data *stores* adalah deskripsi dari data apa yang disimpan. Simbol dari data *stores* adalah dua garis horizontal seperti pada gambar di bawah ini :



Gambar 2.13 *Data Stores* [8]

2.5 FLOWCHART

Flowchart merupakan representasi grafik dari algoritma dengan menggunakan simbol-simbol tertentu yang masing-masing mempunyai fungsi yang khusus. *Flowchart* menggambarkan logika atau langkah dari sistem (proses, operasi, fungsi, atau aktifitas). *Flowchart* mempunyai beberapa simbol yang digunakan, contoh untuk *flowchart* ditunjukkan pada gambar di bawah ini :






Gambar 2.14 Contoh *Flowchart* [9]

Gambar di atas menunjukkan simbol-simbol dasar yang dipakai dalam pembuatan *flowchart*. Beberapa simbol lain yang dipakai dan penjelasannya dapat dilihat pada tabel di bawah ini :

Gambar 2.15 Simbol *Flowchart* [9]

| Simbol | Nama Simbol | Keterangan |
|--------|---------------------------------|---|
| | Terminator | Untuk memulai atau mengakhiri program |
| | Garis Alir (<i>Flow Line</i>) | Arah aliran suatu proses dalam program |
| | Proses | Proses perhitungan atau pengolahan data |
| | Decision | Penyeleksian data yang |

| | | |
|---|--------------------------------------|---|
| | (keputusan dalam pengujian) | memberikan pilihan untuk langkah selanjutnya. |
|  | Input/Output Data | Proses masukan atau keluaran data, parameter, informasi |
|  | Document | Untuk langkah proses yang menghasilkan dokumen |
|  | Magnetic Disk (<i>Database</i>) | Simbol yang paling universal untuk lokasi penyimpanan data. |

2.6 MAPSERVER

2.6.1 Pengenalan MapServer

MapServer merupakan aplikasi *freeware* dan *open source* yang memungkinkan kita menampilkan data spasial (peta) di *web*. Aplikasi ini pertama kali dikembangkan di Universitas Minesotta, Amerika Serikat untuk proyek *ForNet* (sebuah proyek untuk manajemen sumber daya alam) yang disponsori NASA (*Nasional Aeronautics and Space Administration*).[10]

Pada bentuk paling dasar, *MapServer* berupa sebuah program CGI (*Common Gateway Interface*). Program tersebut akan dieksekusi di *webserver* dan berdasarkan beberapa parameter tertentu (terutama konfigurasi dalam bentuk file *.MAP) akan menghasilkan data yang kemudian akan dikirim ke *web browser*, baik dalam bentuk gambar peta atau bentuk lain. *MapServer* mempunyai beberapa fitur sebagai berikut :

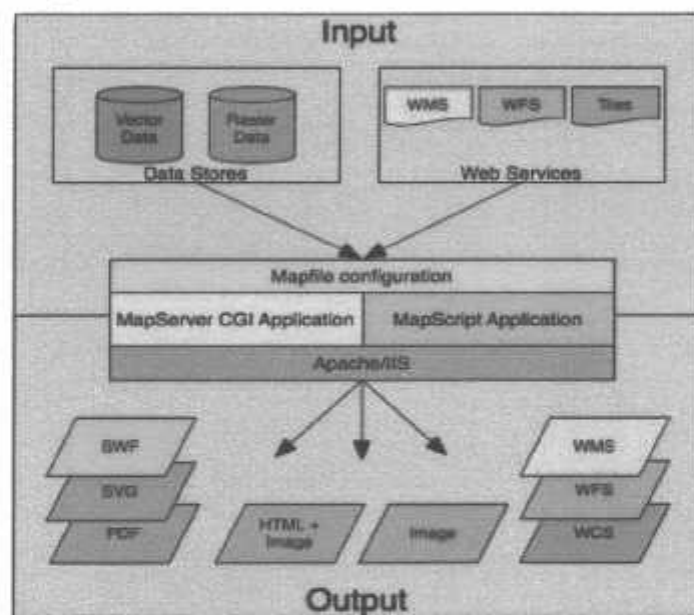
1. Menampilkan data spasial dalam format vektor seperti : Shapefile (ESRI), ArcSDE (ESRI), PostGIS dan berbagai format data vektor lain dengan menggunakan *library* OGR,
2. Menampilkan data spasial dalam format *raster* seperti TIFF/GeoTIFF, EPPL7 dan berbagai format data *raster* lain dengan menggunakan *library* GDAL,
3. Menggunakan *quadtree* dalam *indexing* data spasial, sehingga operasi-operasi spasial dapat dilakukan dengan cepat,
4. Dapat dikembangkan (*customizable*), dengan tampilan keluaran yang dapat diatur menggunakan *file-file template*,
5. Dapat melakukan seleksi objek berdasar nilai, berdasar titik, area, atau berdasar sebuah objek spasial tertentu,
6. Mendukung penggunaan data raster maupun vektor yang di-tiled (dibagi-bagi menjadi sub bagian yang lebih kecil sehingga proses untuk mengambil dan

menampilkan gambar dapat dipercepat),

7. Dapat menggambarkan elemen peta secara otomatis , skala grafis, peta indeks dan legenda peta,
8. Dapat menampilkan label dari objek spasial, dengan label dapat diatur sedemikian rupa sehingga tidak saling tumpang tindih.

Saat ini, selain dapat mengakses *MapServer* sebagai program CGI, kita dapat mengakses *MapServer* sebagai modul *MapScript*, melalui berbagai bahasa *script* : PHP, Perl, Python atau Java. Akses fungsi-fungsi *MapServer* melalui *script* akan lebih memudahkan pengembangan aplikasi. Pengembang dapat memilih bahasa yang paling familiar. [10]

2.6.2 Arsitektur *MapServer*



Gambar 2.15 Arsitektur *MapServer* [10]

Arsitektur aplikasi pemetaan di *web* dibagi menjadi dua pendekatan sebagai berikut :

a. Pendekatan *Thin Client*

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

fleksibel.

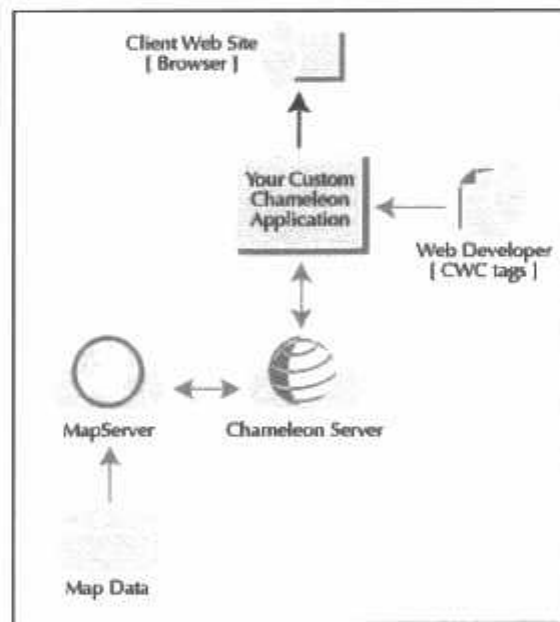
b. Pendekatan *Thick Client*

Pada pendekatan ini, pemrosesan data dilakukan di sisi klien menggunakan beberapa teknologi seperti kontrol *ActiveX* atau *applet*. Kontrol *ActiveX* atau *applet* akan dijalankan di klien untuk memungkinkan *web browser* dengan format data yang tidak dapat ditangani oleh *web browser* dengan kemampuan *standard*. Dengan adanya pemrosesan di klien, maka transfer data antara klien dengan *web server* akan berkurang.

MapServer menggunakan pendekatan *thin client*. Semua pemrosesan dilakukan di sisi *server*. Informasi peta dikirimkan ke *web browser* di sisi klien dalam bentuk file gambar (JPG, PNG, GIF atau TIFF).

2.6.3 Chameleon

Chameleon adalah *framework* yang dapat digunakan dengan baik pada *webGIS*. Dapat digunakan secara berdampingan atau *full integrated* dengan dengan *MapServer* berdasarkan spesifikasi yang ditentukan oleh *Open Geospatial Consortium (OGC)*. Gambar berikut mengilustrasikan konfigurasi nya :



Gambar 2.16 Konfigurasi Chameleon digunakan dengan *MapServer* [10]

DM *Solution Group* (Ottawa, Canada) adalah pengembang komponen yang *reusedable* pada Chameleon. Komponen-komponen yang dikembangkan tersebut membangun Chameleon *framework*.

Chameleon sebagai sebuah produk dari *open source* yang dibangun dengan

bahasa pemrograman PHP. Chameleon memberikan akses yang sederhana ke beberapa fitur yang hanya bisa diakses dalam *map script* dimana telah disediakan sebuah *script* yang telah jadi sebagai komponen yang dapat di gunakan. Dengan Chameleon seorang yang bukan *programmer* memungkinkan untuk memasukkan komponen pada aplikasi *webGIS*. [10]

BAB III

ANALISA DAN PERANCANGAN SISTEM

Pada bab ini akan diuraikan tentang perancangan sistem berupa analisa dan desain sistem. Proses analisa menjelaskan tentang deskripsi sistem, pengguna sistem, dan spesifikasi kebutuhan sistem.

Sedangkan perancangan sistem dalam proses pembuatan aplikasi ini meliputi perancangan desain sistem, desain *database*, dan desain *interface* atau antarmuka.

3.1 ANALISA SISTEM

3.1.1 Deskripsi Sistem

Aplikasi layanan webgis pondok pesantren ini merupakan sistem informasi geografis yang tujuan pembuatannya adalah untuk menampilkan semua informasi tentang tempat pesantren di kabupaten Jombang berkaitan dengan data-data mengenai pondok pesantren itu sendiri dan penunjukan lokasi. Dengan memanfaatkan media webgis untuk menggantikan media konvensional (peta) sehingga lebih efektif.

Berdasarkan tugas dan fungsinya di dalam sistem, pengguna aplikasi dibedakan menjadi 2 pelaku yaitu *administrator*, dan pengguna layanan aplikasi. Adapun tugas dan fungsi dari masing-masing pengguna sistem tersebut adalah :

1. *Administrator*

Merupakan pelaku yang memiliki hak akses tertinggi dalam aplikasi ini. Adapun tugas dan fungsinya yaitu mengelola keseluruhan menu dan fasilitas yang ada di dalam sistem.

2. *Pengguna Layanan Aplikasi (user)*

Pengguna layanan aplikasi hanya dapat melihat dan memperoleh informasi yang ditampilkan oleh aplikasi webgis.

Secara garis besar pengguna layanan hanya memiliki hak akses saja. Sedangkan untuk *administrator* diberikan hak akses penuh untuk menambah, merubah, dan menghapus data.

3.1.2 Pengguna Sistem

Pengguna aplikasi SIG pemetaan objek pesantren ini ditujukan kepada masyarakat umum, dan khususnya para santriwan-santriwati yang ingin mengetahui objek pondok pesantren di kabupaten Jombang.

3.1.3 Spesifikasi Kebutuhan Sistem

Sistem yang dibutuhkan untuk membangun suatu aplikasi *webbased* GIS dengan menggunakan *mapserver* terdiri atas perangkat keras (*hardware*) dan perangkat lunak (*software*).

1. Perangkat Keras (*hardware*)

Perangkat keras yang digunakan adalah PC (*Personal Computer*). Spesifikasi perangkat keras yang harus dipenuhi untuk pengembangan perangkat lunak adalah minimal dapat menjalankan perangkat lunak sesuai kebutuhan. Perangkat lunak yang dibutuhkan bergantung pada kebutuhan aplikasi itu sendiri.

2. Perangkat Lunak (*software*)

- a. Sistem operasi komputer
- b. Arcview
- c. PostgreSQL dan Postgis
- d. Ms4w dan Chameleon
- e. *Text Editor* : Notepad++, Dreamweaver
- f. *Browser* internet : Mozilla, Opera

3. Data Spasial dan Data Atribut

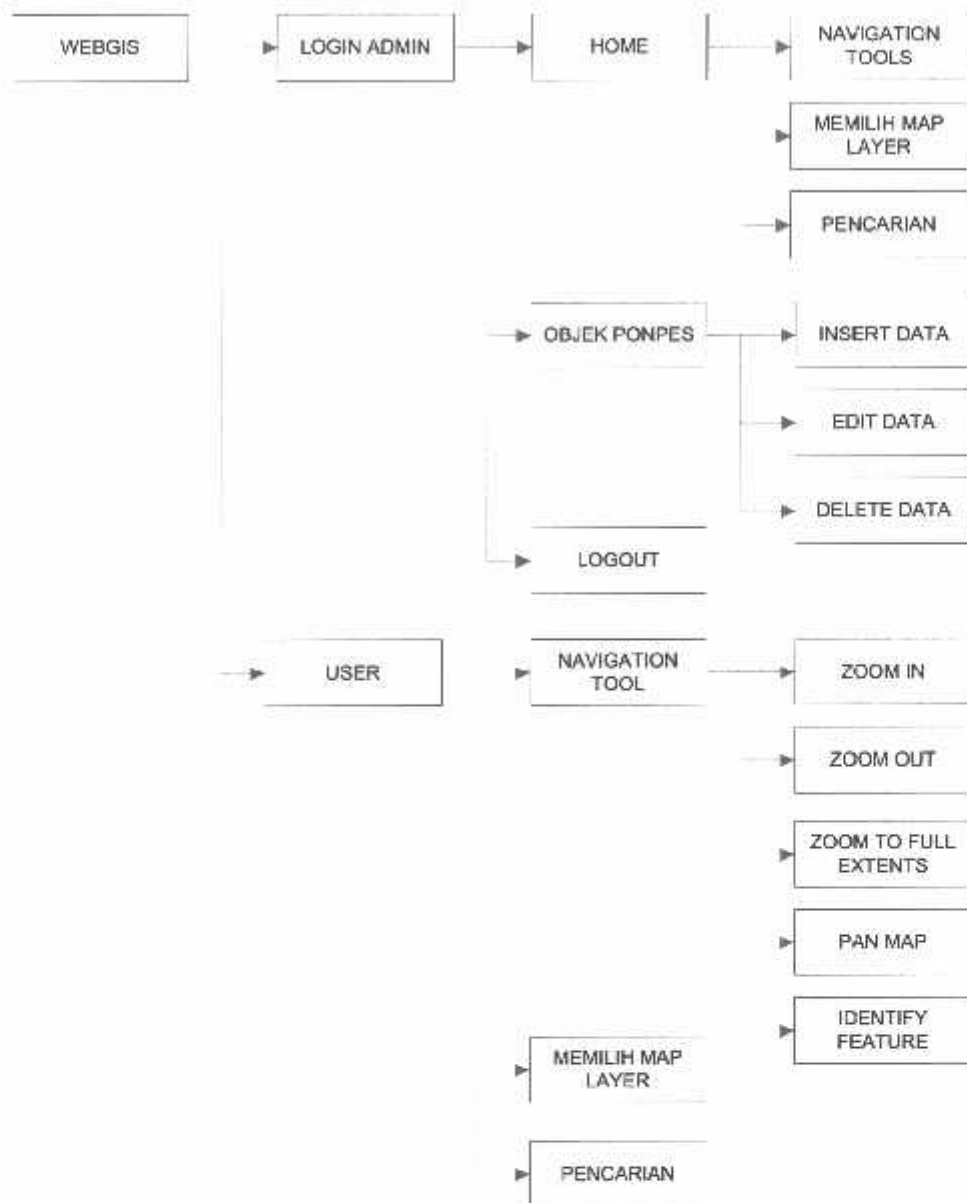
Data spasial berupa gambar peta digital yang akan dibuat dalam *software* Arcview, sedangkan data atribut berupa data yang menerangkan informasi-informasi dari suatu objek (lokasi dan posisi) yang ditunjukkan oleh data spasial.

3.2 PERANCANGAN SISTEM

3.2.1 DESAIN SISTEM

Desain sistem aplikasi sistem informasi geografis pemetaan objek pondok pesantren di kabupaten Jombang ini dibagi menjadi tiga, yaitu *WBS* (*Work Breakdown System*), *Flowchart*, dan *DFD* (*Data Flow Diagram*).

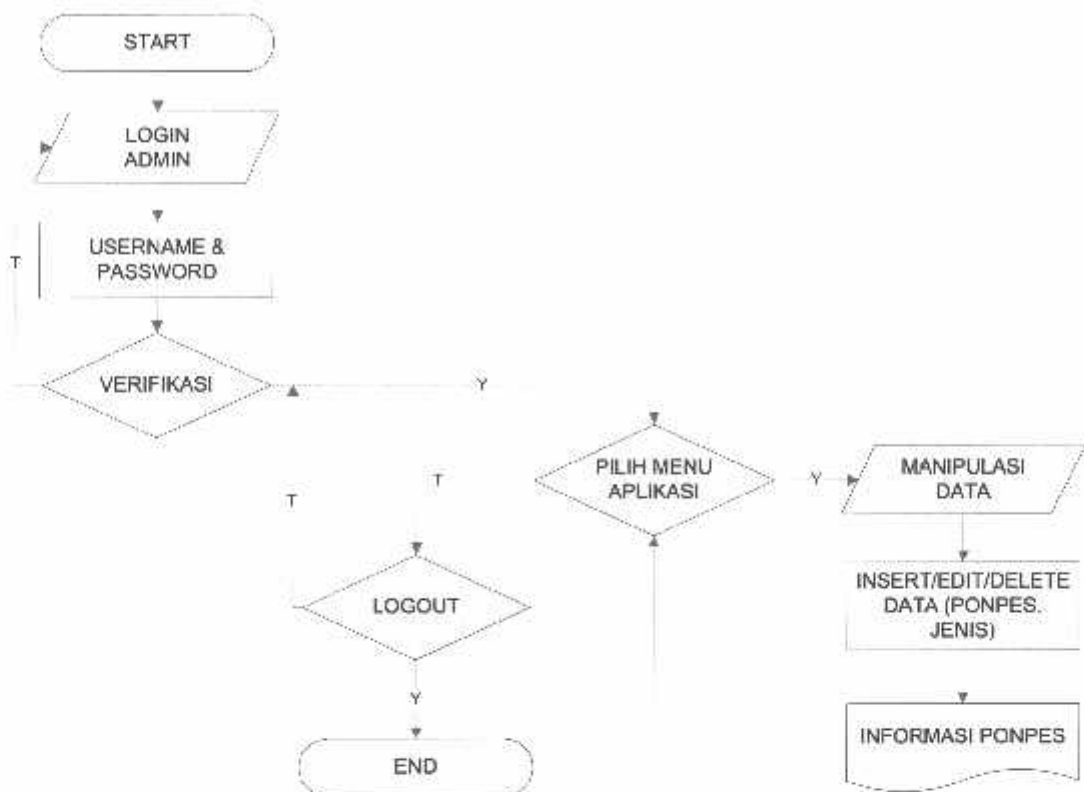
3.2.1.1 Work Breakdown System (WBS)



Gambar 3.1 Rancangan Desain Tampilan Aplikasi

Dari gambar desain aplikasi sistem di atas, menunjukkan menu-menu yang dapat diakses oleh masing-masing pengguna sistem (*administrator* dan *user* umum). *Administrator* memiliki menu-menu yang dibutuhkan untuk semua pengaksesan data dalam aplikasi sistem ini. Sedangkan *user* umum hanya dapat melihat informasi yang tersedia dalam peta, dan mendapatkan beberapa fasilitas, seperti pencarian, memilih map layer, dan *navigation tools* berupa *zoom in*, *zoom out*, *zoom to full extents*, *pan map*, dan *identify feature*.

3.2.1.2 Flowchart



Gambar 3.2 Flowchart Admin

Proses *admin* dimulai dari *admin* melakukan *login* sebagai *admin* dengan memasukkan *username* dan *password*. Kemudian diverifikasi oleh sistem, jika *username* dan *password* benar maka akan diarahkan pada menu aplikasi sistem, namun jika salah akan dikembalikan pada *login admin*. Setelah masuk pada menu aplikasi, *admin* bisa melakukan proses manipulasi data objek pesantren yang akan menghasilkan data informasi objek pesantren baru dimana data tersebut akan mengganti data lama yang tersedia pada menu aplikasi. Jika sudah selesai, maka *admin* mengakhiri dengan *logout admin* terlebih dahulu.



Gambar 3.3 Flowchart *User*

Pada sisi *user* (pengguna umum), langsung diarahkan pada menu aplikasi tanpa harus melakukan *login* terlebih dahulu. *User* bisa memilih map layer sesuai yang dibutuhkan, kemudian juga disediakan beberapa *navigation tools* untuk mempermudah melakukan pencarian objek pesantren di dalam peta, hingga akhirnya *user* mendapatkan informasi objek pesantren yang diinginkan.

3.2.1.3 DFD (*Data Flow Diagram*)

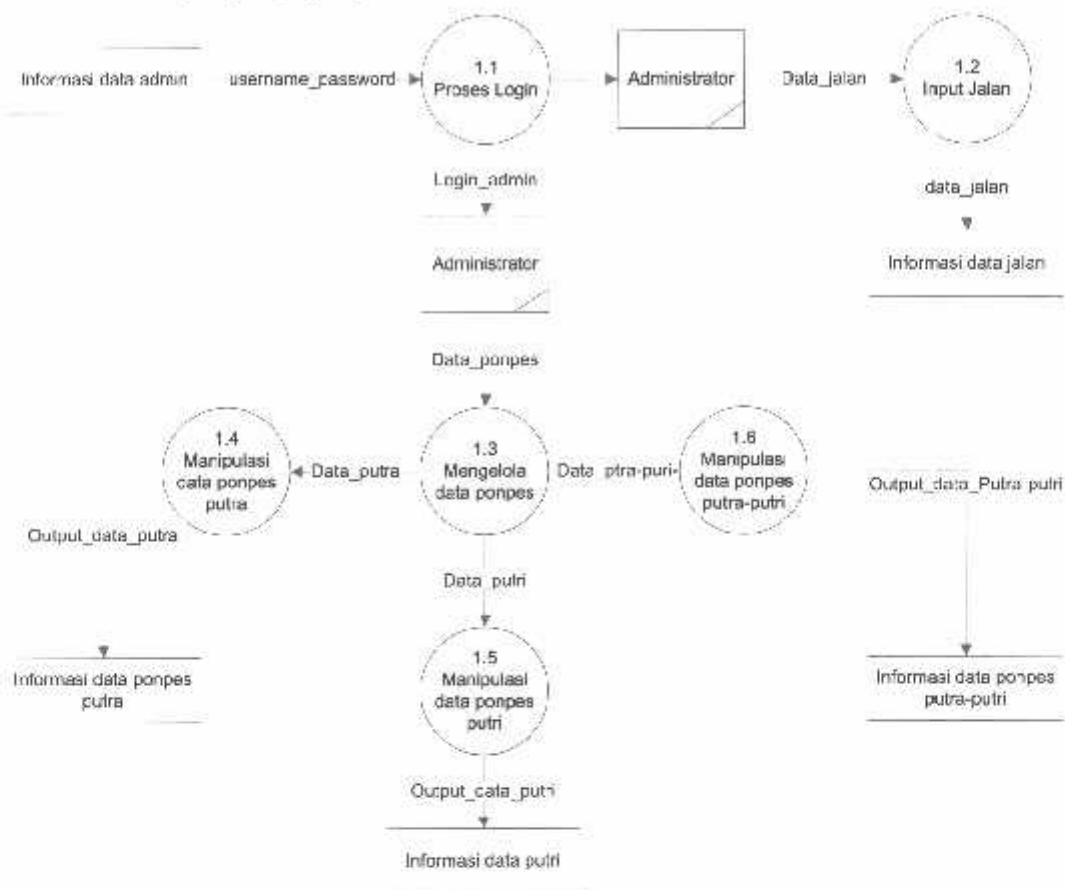


Gambar 3.4 *Data Flow Diagram* Level 0

Sistem berinteraksi dengan 2 pelaku, yaitu :

1. *Admin* memberikan informasi data pesantren sebagai masukan ke dalam sistem dan oleh sistem akan diolah dan dihasilkan output berupa informasi data pesantren,
2. *User* (pengguna sistem) memberikan *query* tentang informasi objek pesantren ke dalam sistem. Hasil keluaran dari sistem kepada *user* berupa *result* informasi data

sesuai dengan *query* yang telah dimasukkan tadi.



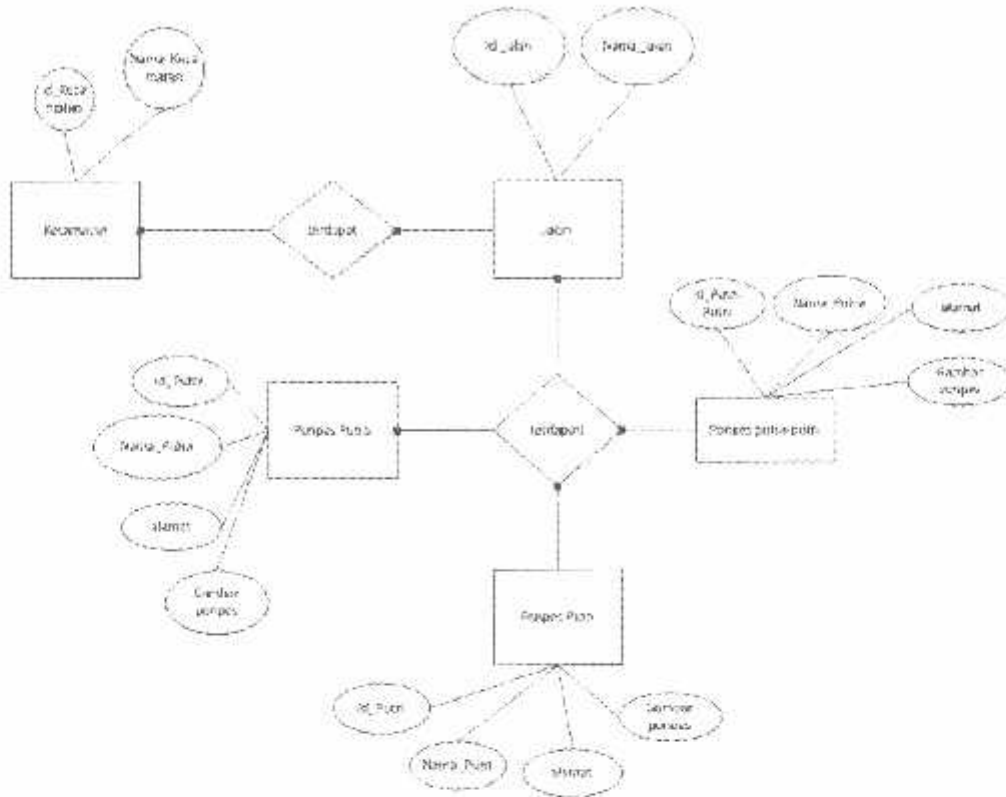
Gambar 3.5 Data Flow Diagram Level 1

Proses pada DFD level 1 ini adalah :

- 1) Proses login dilakukan dengan memasukkan username dan password yang telah tersimpan dalam sistem,
- 2) Administrator yang telah *login* dalam sistem memasukkan data informasi jalan,
- 3) Administrator melakukan proses pengelolaan data pondok pesantren dengan memasukkan data pondok pesantren ke dalam sistem,
- 4) Sistem akan menghasilkan data pesantren yang kemudian akan dimanipulasi oleh sistem (update, delete), yang menghasilkan output berupa informasi data ponpes putra,
- 5) Sistem akan menghasilkan data plasa yang kemudian akan dimanipulasi oleh sistem (update, delete), yang menghasilkan output berupa informasi data ponpes putri,
- 6) Sistem akan menghasilkan data pesantren yang kemudian akan dimanipulasi oleh sistem (update, delete), yang menghasilkan output berupa informasi data ponpes putra-putri.

3.2.2 DESAIN DATABASE

3.2.2.1 Entity Relationship Diagram (ERD)



Gambar 3.6 Entity Relationship Diagram

Desain *database* di atas mempunyai lima entitas, yaitu kecamatan, jalan, pesantren, putra, putri dan putra-putri. Dimana masing-masing memiliki atribut seperti yang sudah disajikan pada gambar di atas.

3.2.2.2 Tabel Database

a) Tabel Database Putra

Tabel 3.1 Tabel Database Pesantren Putra

| Nama Field | Type data | Panjang | Keterangan |
|------------|-----------|---------|---|
| Id_pp | Int | | Id objek |
| Nama_pp | Varchar | 50 | Nama objek |
| Jenis_pp | Varchar | 20 | Jenis Tempat |
| Alamat_pp | Varchar | 50 | Alamat objek |
| Detail_pp | Varchar | 1000 | Berisi keterangan mengenai objek |
| Gambar_pp | Varchar | 1000 | Berisi data gambar yang menampilkan visualisasi objek |

b) Tabel Database Putri

Tabel 3.2 Tabel Database Pesantren Putri

| Nama Field | Type data | Panjang | Keterangan |
|-------------------|------------------|----------------|---|
| Id_pp | Int | | Id objek |
| Nama_pp | Varchar | 50 | Nama objek pesantren |
| Jenis | Varchar | 20 | Jenis tempat pondok pesantren |
| Alamat_pp | Varchar | 50 | Alamat objek |
| Detail_pp | Varchar | 1000 | Berisi keterangan mengenai objek |
| Gambar_pp | Varchar | 1000 | Berisi data gambar yang menampilkan visualisasi objek |

c) Tabel Database Putra-Putri

Tabel 3.3 Tabel Database Pesantren Putra-Putri

| Nama Field | Type data | Panjang | Keterangan |
|-------------------|------------------|----------------|---|
| Id_pp | Int | | Id |
| Nama_pp | Varchar | 50 | Nama objek |
| Jenis | Varchar | 20 | Jenis tempat pondok pesantren |
| Alamat_pp | Varchar | 50 | Alamat objek |
| Detail_pp | Varchar | 1000 | Berisi keterangan mengenai objek |
| Gambar_pp | Varchar | 1000 | Berisi data gambar yang menampilkan visualisasi objek |

d) Tabel Database Jalan

Tabel 3.4 Tabel Database Jalan

| Nama Field | Type data | Panjang | Keterangan |
|-------------------|------------------|----------------|-------------------|
| Id_jln | Int | | Id |
| Nama_jln | Varchar | 50 | Nama jalan |

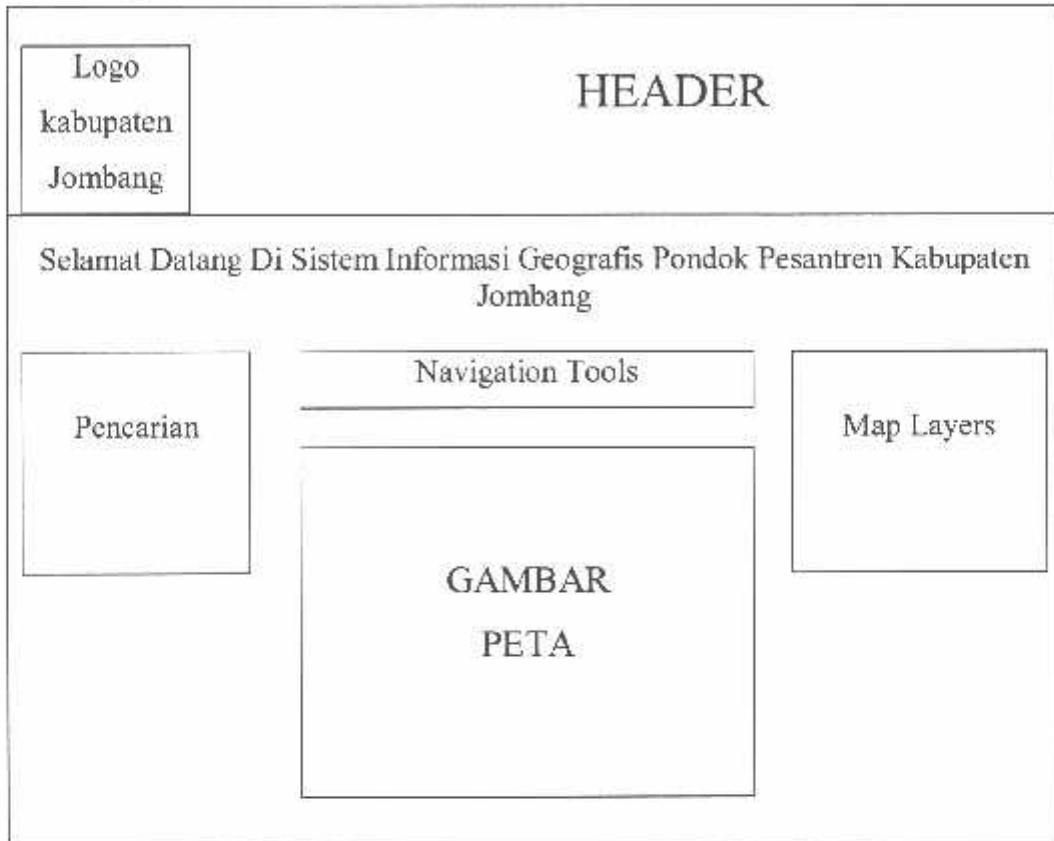
e) Tabel Database Admin

Tabel 3.5 Tabel Database Admin

| Nama Field | Type data | Panjang | Keterangan |
|-------------------|------------------|----------------|--------------------------|
| Id_admin | Int | 10 | Kode sebagai primary key |
| Username | Varchar | 30 | Nama lengkap admin |
| Password | Varchar | 30 | Password admin |

3.2.3 DESAIN INTERFACE / ANTARMUKA

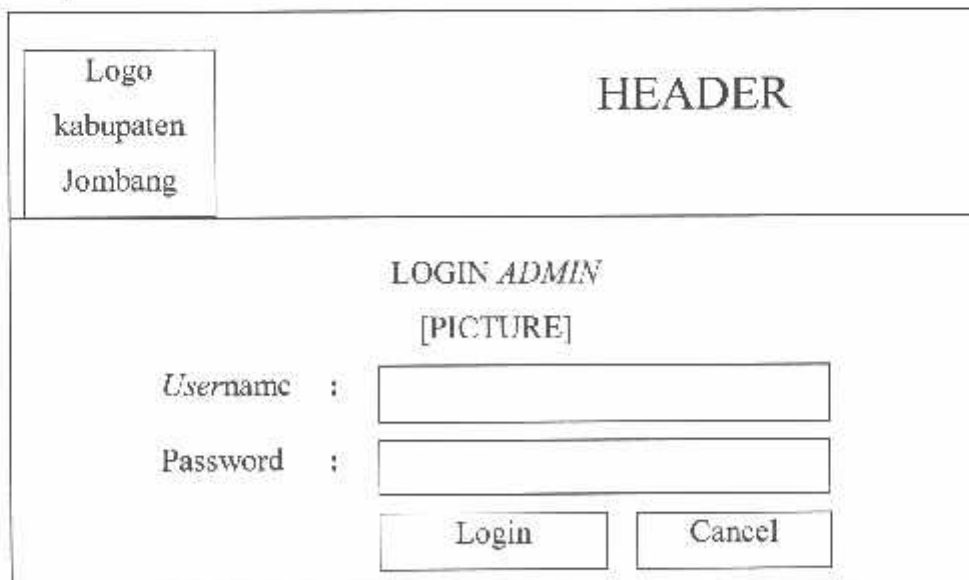
3.2.3.1 Tampilan Utama *User*



Gambar 3.7 Desain Tampilan Utama *User*

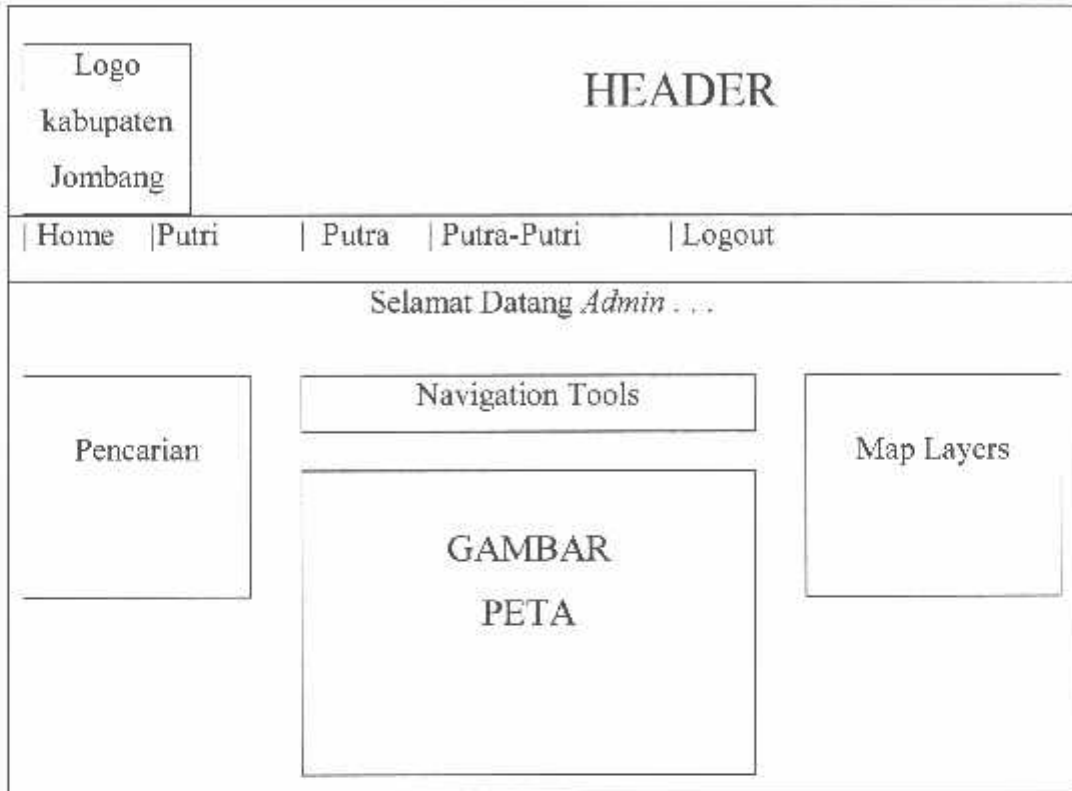
Desain tampilan awal aplikasi ini, yaitu pada *header* terdapat logo instansi. Sedangkan pada halaman isi (bawah *header*) terdapat informasi berupa gambar peta, *maplayer*, dan alat navigasi serta fungsi pencarian.

3.2.3.2 Login *Admin*



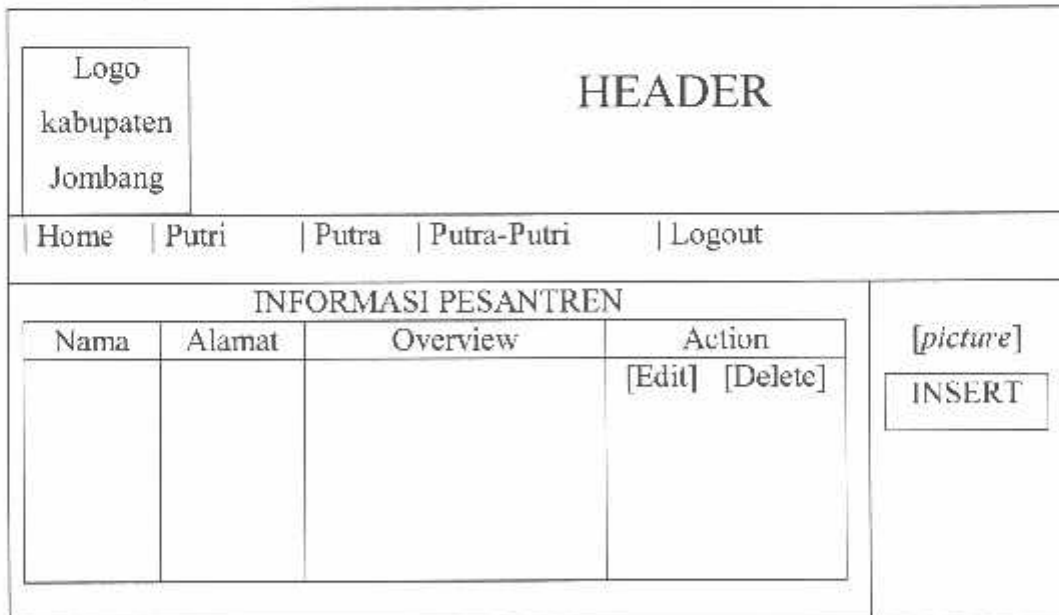
Gambar 3.8 Desain Tampilan Login *Admin*

3.2.3.3 Tampilan Utama *Admin*



Gambar 3.9 Desain Tampilan Utama (*Home*) *Admin*

3.2.3.4 Manipulasi Data Pesantren Putri / Putra / Putra-Putri



Gambar 3.10 Desain Tampilan Manipulasi Data Pesantren

3.2.3.5 Tampilan Insert

| | | |
|------------------------------|---------------------------------------|---------------------------------------|
| Logo kabupaten Jombang | HEADER | |
| Insert Informasi Pesantren | | |
| [picture] INSERT | Nama : | <input type="text"/> |
| | Alamat : | <input type="text"/> |
| | Jenis : | <input type="text"/> |
| | Overview : | <input type="text"/> |
| | Koordinat Lintang (x) : | <input type="text"/> |
| | Koordinat Bujur (y) : | <input type="text"/> |
| | Gambar : | <input type="text"/> |
| | <input type="button" value="Simpan"/> | <input type="button" value="Browse"/> |

Gambar 3.11 Desain Tampilan Insert Pesantren

3.2.3.6 Tampilan Edit

| | | |
|------------------------------|---------------------------------------|---------------------------------------|
| Logo kabupaten Jombang | HEADER | |
| Edit Informasi Pesantren | | |
| [picture] EDIT | Nama : | <input type="text"/> |
| | Alamat : | <input type="text"/> |
| | Jenis : | <input type="text"/> |
| | Koordinat : | <input type="text"/> |
| | Gambar : | <input type="text"/> |
| | <input type="button" value="Simpan"/> | <input type="button" value="Browse"/> |

Gambar 3.12 Desain Tampilan Edit Pesantren

BAB IV IMPLEMENTASI DAN PENGUJIAN

Bab ini membahas tentang implementasi dan pengujian terhadap aplikasi webgis objek pondok pesantren di kabupaten Jombang.

4.1 IMPLEMENTASI

Implementasi aplikasi webgis ini terdiri dari beberapa tahap, yaitu : tahap pembuatan peta digital (berupa data spasial dan data *attribute*) menggunakan *software* Arcview, pembuatan *database* pada postgresQL, pembuatan struktur file *.map, dan pembuatan struktur template menggunakan *framework* chameleon.

4.1.1 Pembuatan Peta Digital (Data Spasial dan Data Atribut)

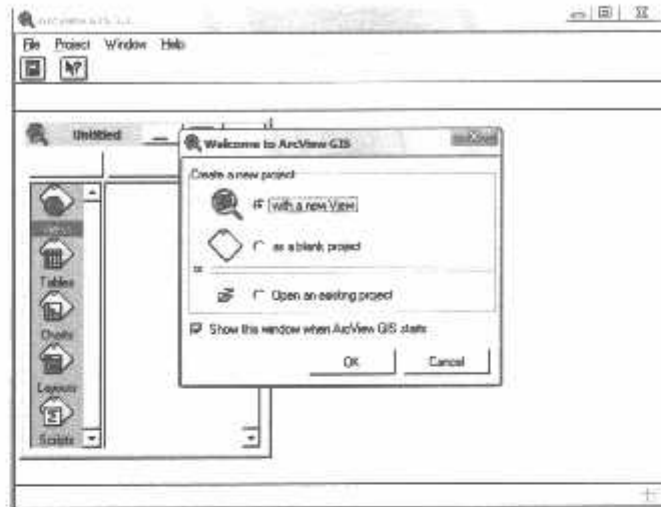
Proses pembuatan peta digital dilakukan dengan menggunakan *software* Arcview. *Software* ini bisa didapatkan secara gratis di *website* berikut www.esri.com. Data yang diperlukan dalam pembuatan peta digital ini adalah data gambar peta mati atau peta yang belum terdapat koordinat *latitude* dan *longitude* di dalamnya. Dalam hal ini digunakan peta kabupaten Jombang berformat *.bmp. Dimana peta tersebut akan dijadikan acuan untuk menggambar peta digital.

Secara garis besar, pembuatan peta digital dilakukan dalam rangkaian proses berikut :



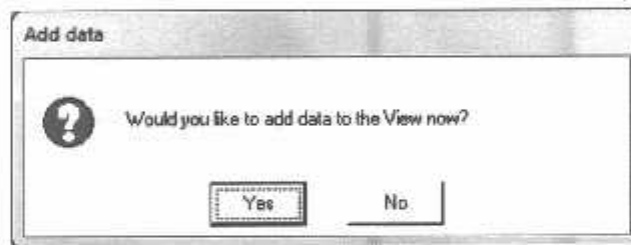
Gambar 4.1 Proses Pembuatan Peta Digital

1. Masuk pada *software* Arcview, membuat project baru dan memasukkan data gambar peta mati yang telah disediakan tadi, sebagai acuan pembuatan peta digital.



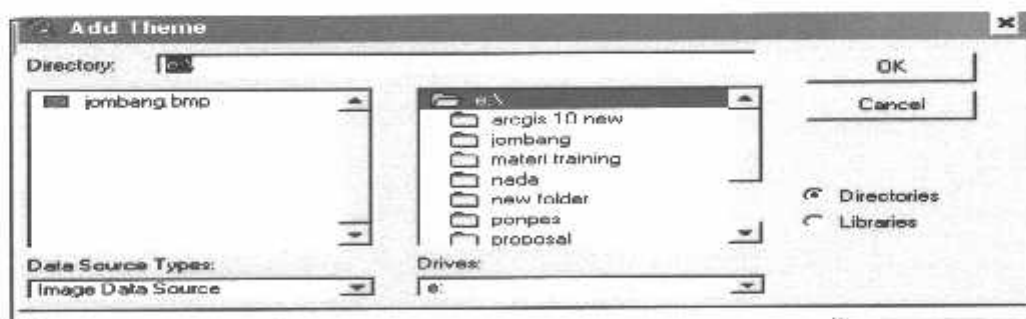
Gambar 4.2 Jendela "Welcome to Arcview GIS"

Pada jendela "Welcome to Arcview GIS" terdapat tiga pilihan. pilih "with a new View". Kemudian muncul jendela "Add data".



Gambar 4.3 Jendela "Add Data"

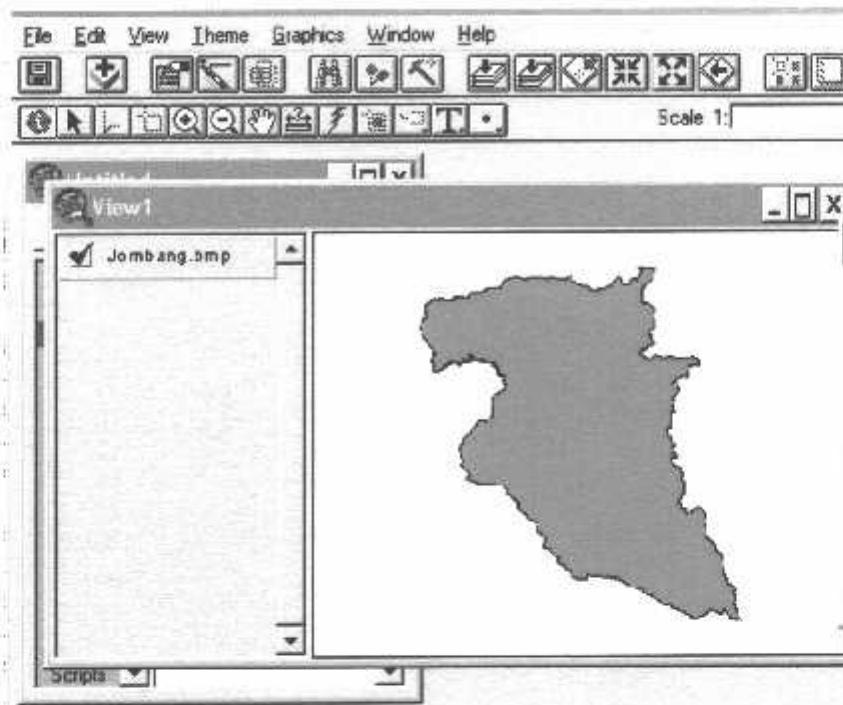
Pilih yes, dan setelah itu akan diarahkan ke "Add Theme". Disini kita mengambil gambar peta Jombang (berformat *.bmp) yang akan digunakan sebagai acuan untuk menggambar peta di Arcview.



Gambar 4.4 Jendela "Add Theme"

Ubah "*Data Source Type*" menjadi "*Image Data Source*", dan pilih file gambar yang tersimpan pada "*Drives*".

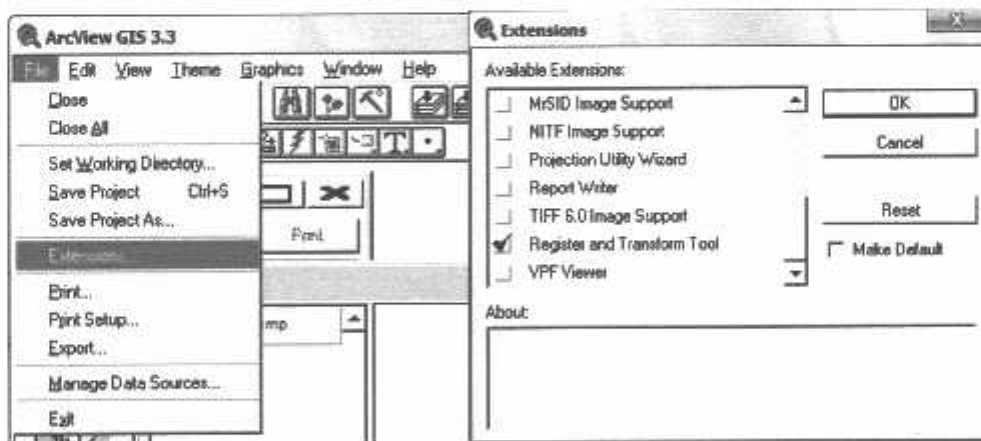
Setelah file berhasil di tambahkan pada *theme*, file dalam status tidak aktif. Untuk mengaktifkan theme klik *checkbox* disebelah nama file.



Gambar 4.5 File Peta Jombang pada Arcview

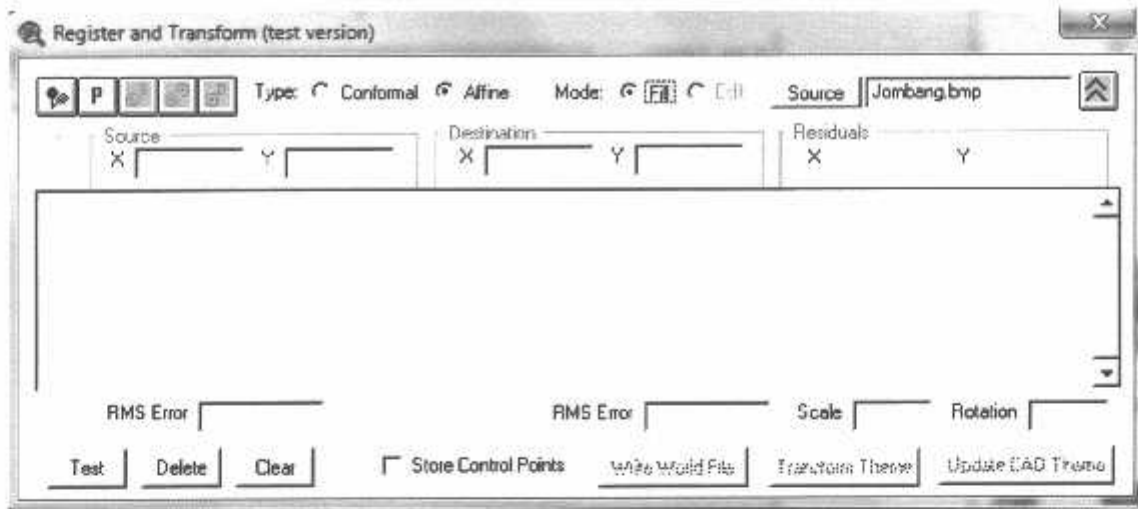
2. Melakukan proses registrasi peta dasar.

Pilih menu *File > Extension > (centang) Register and Transform Tool*.



Gambar 4.6 Mengaktifkan Ekstensi *Register and Transform Tool*

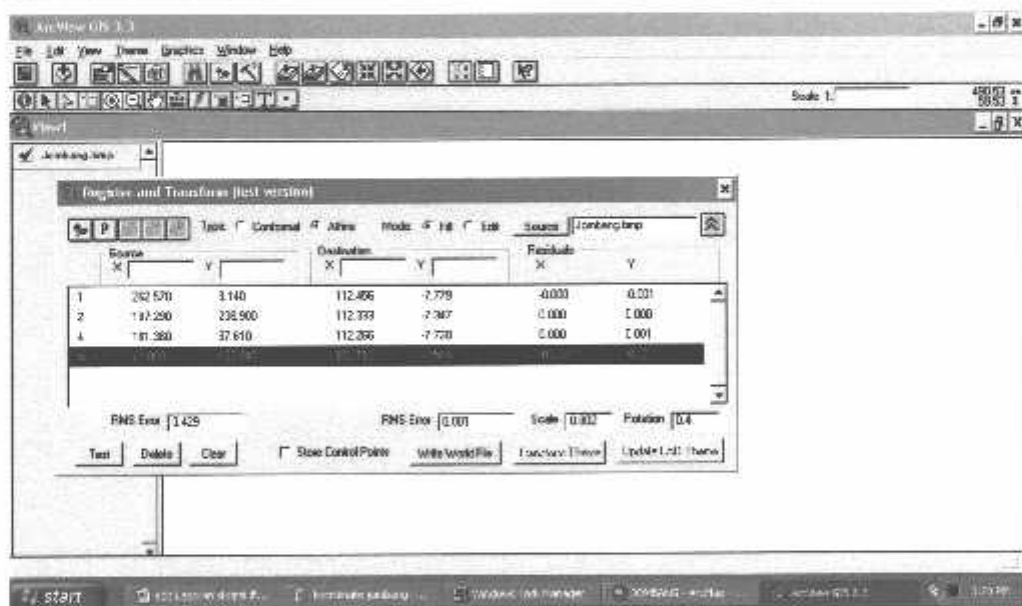
Untuk menampilkan pilih menu *View > Register and Transform Tool*.



Gambar 4.7 Proses Registrasi Peta

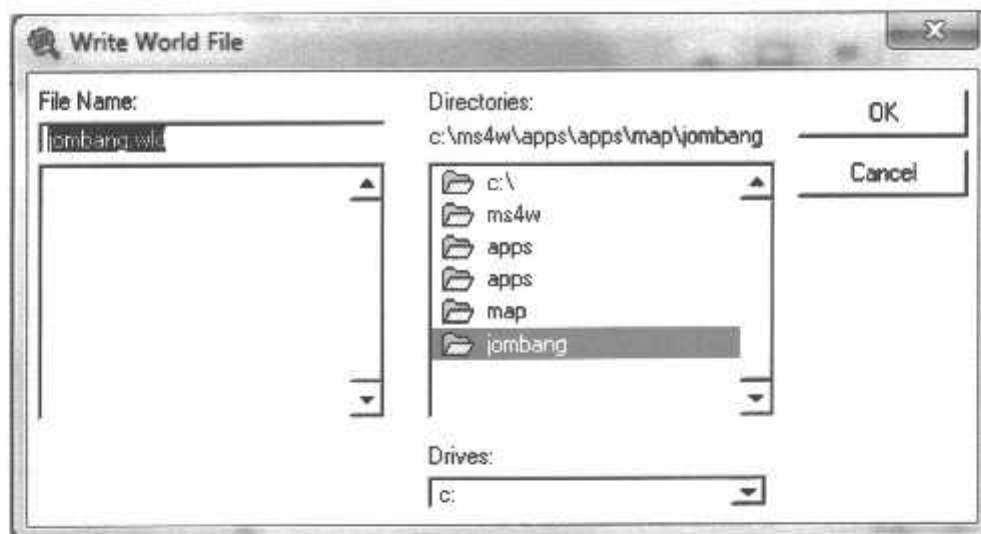
Klik tombol *source point* yang bersimbol P untuk memasang koordinat bertipe *Universal Transverse Mercator (UTM)* atau untuk memasukkan koordinat pada titik yang telah diketahui koordinatnya.

Lakukan pada empat titik yang berbeda, dan lihat nilai *RMS Error* nya. Nilai ini menunjukkan tingkat ketelitian yang terjadi pada saat registrasi. Nilai RMS yang diijinkan maksimal 0,5. Semakin besar nilai RMS ini akan mengakibatkan akurasi yang semakin rendah.



Gambar 4.8 Memasang Titik GCP

Setelah didapatkan nilai RMS terkecil, pilih tombol *“Write World File”* sehingga muncul jendela seperti di bawah ini. Jangan pindahkan folder, dan segera pilih tombol *OK*.



Gambar 4.9 Menyimpan File Koordinat GCP

Setelah tombol ini dipilih maka gambar pada view akan hilang. Dengan hilangnya gambar tersebut berarti pemasangan GCP sudah benar.

Masukkan kembali data peta Jombang dengan memilih *View > Add Theme*.

Tanda bahwa koordinat peta dasar ini telah sesuai dengan koordinat bumi adalah dengan adanya nilai dari derajat peta pada pojok kanan atas yang sudah sesuai.

3. Melakukan proses digitasi feature.

Membuat lima layer baru (*View > New Theme*), dengan ketentuan sebagai berikut :

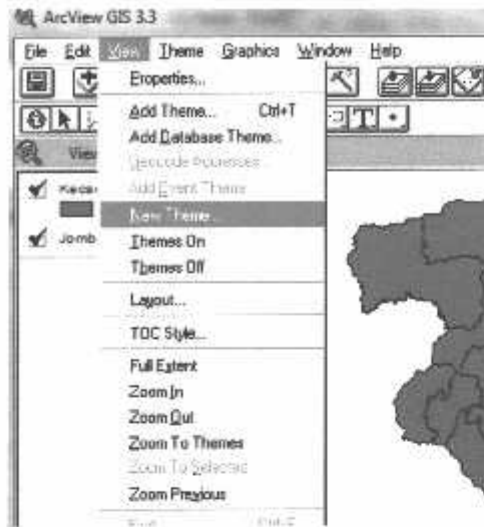
Tabel 4.1 Tabel Layer

| No. | Layer | Feature Type |
|-----|-------------|--------------|
| 1 | Kecamatan | Polygon |
| 2 | Jalan | Line |
| 3 | Putra | Point |
| 4 | Putri | Point |
| 5 | Putra-putri | Point |

Proses menggambar bisa dimulai dengan memilih menu *drop down "draw polygon"* untuk menggambar peta kecamatan, *"draw line"* untuk menggambar peta jalan, *"draw point"* untuk menggambar peta pesantren putra, pesantren putri, dan pesantren putra-putri. Masing-masing layer tersebut disimpan dalam format file *.shp

Di bawah ini akan diberikan salah satu contoh dari proses pembuatan masing-masing layer yang telah disebutkan di atas, yaitu pembuatan layer kecamatan.shp.

a) Membuat layer baru (*View > New Theme*)



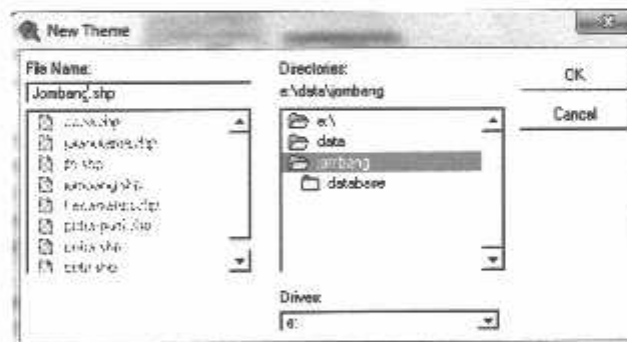
Gambar 4.10 *New Theme*

- b) Menentukan *feature type* sesuai dengan yang akan dibuat. Jika membuat sebuah area menggunakan *feature type polygon*, untuk membuat garis menggunakan *line*, dan untuk membuat titik menggunakan *point*.



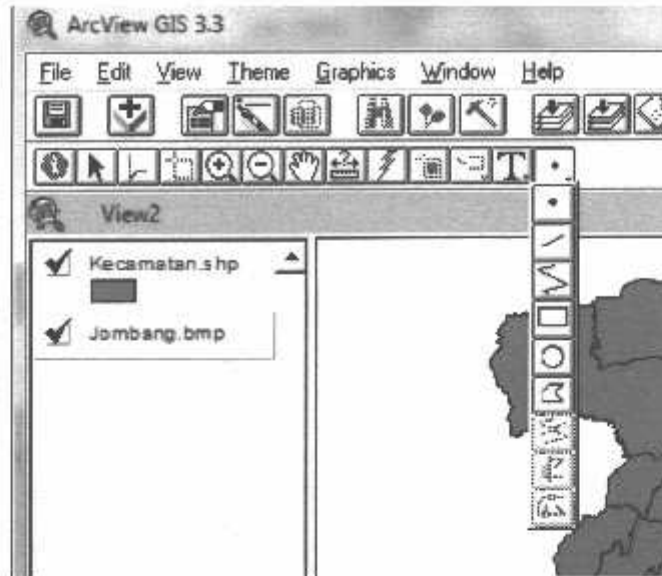
Gambar 4.11 *Feature Type*

- c) Menyimpan *theme* dengan format file *.shp



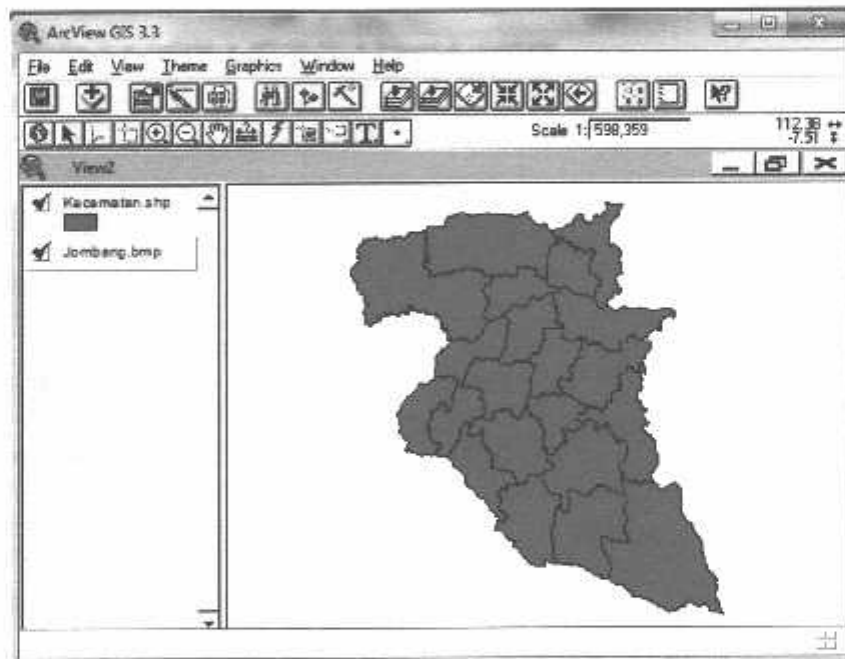
Gambar 4.12 *Menyimpan Theme*

- d) Proses menggambar bisa dimulai dengan memilih menu *drop down "draw polygon"*.



Gambar 4.13 Menu Drop Down

- c) Untuk membagi *polygon* menjadi beberapa bagian, gunakan tool “draw line to split polygon”. Dan hasilnya bisa dilihat pada gambar di bawah ini :



Gambar 4.14 Theme kecamatan.shp setelah dibagi menjadi 20 bagian

4. Input Data Atribut.

Mengisi data pada masing-masing bagian layer, dengan memilih menu *Theme > Table*. Tambahkan kolom yang dibutuhkan dengan mengklik *Edit > Add Field*, dan masukkan datanya.



Gambar 4.15 Menambahkan Tabel dan *Field* (kolom)

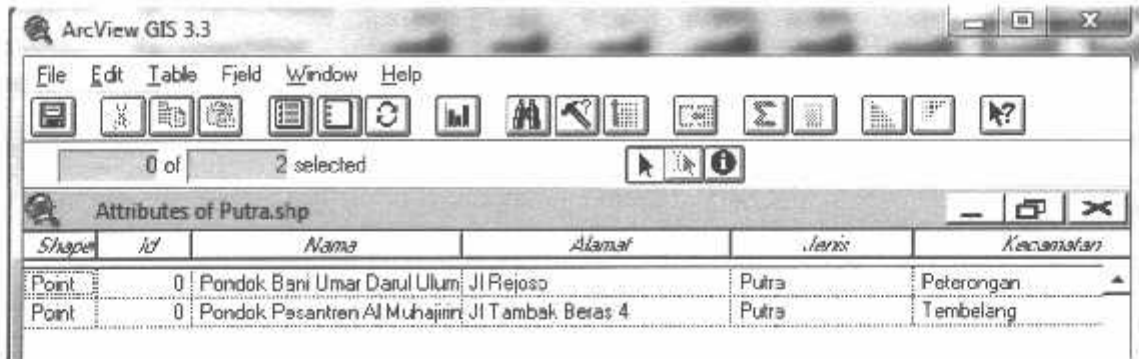
Lakukan hal yang sama untuk membuat layer (data spasial dan attribute) Jalan, Putra, Putri, dan Putra - Putri.

| Shape | Area | Perimeter | Admin_cod | Admin_nm | Kecamatan |
|---------|------------------|--------------|-----------|----------|-----------------------|
| Polygon | 35450427.169434 | 34484.023120 | 1 | | 0 BANDAR KEDUNG MULYO |
| Polygon | 64570132.081029 | 43985.289276 | 2 | | 0 BAFENG |
| Polygon | 93295440.089449 | 42001.656172 | 3 | | 0 DIMEK |
| Polygon | 38873881.201172 | 41077.929107 | 4 | | 0 GUDO |
| Polygon | 28931381.511200 | 32286.042032 | 5 | | 0 JOGJOPRO |
| Polygon | 32320727.370293 | 34963.230548 | 6 | | 0 JOMBANG |
| Polygon | 32210072.620117 | 48992.783439 | 7 | | 0 KABUH |
| Polygon | 54976184.860362 | 44849.689877 | 8 | | 0 KESAMBEN |
| Polygon | 28963896.318236 | 27678.701602 | 9 | | 0 KUDU |
| Polygon | 31890184.300366 | 30820.046495 | 10 | | 0 MEGALIH |
| Polygon | 52700062.950195 | 59265.964130 | 11 | | 0 MELULUNG |
| Polygon | 62495906.739352 | 60063.189187 | 12 | | 0 MOWARNO |
| Polygon | 52614203.473340 | 48113.729001 | 13 | | 0 NGCRO |
| Polygon | 3860788.948707 | 47030.855994 | 14 | | 0 NGLISAN |
| Polygon | 29662114.369220 | 32888.603740 | 15 | | 0 PERAK |
| Polygon | 31513242.994229 | 32729.997679 | 16 | | 0 PETERONGAN |
| Polygon | 119851897.797324 | 74014.414010 | 17 | | 0 PLANDAN |
| Polygon | 38713372.918892 | 27506.851525 | 18 | | 0 PLOSO |
| Polygon | 47127429.310701 | 45887.319918 | 19 | | 0 SUWIRAN |

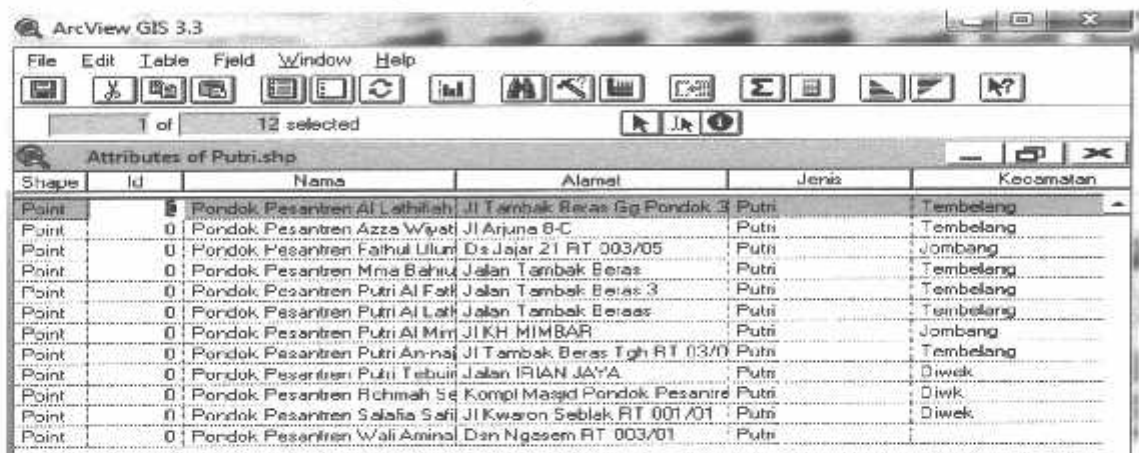
Gambar 4.16 Input Data pada *Theme* Kecamatan

| Shape | Entity | Layer |
|-----------|----------|------------------|
| PolyLineZ | Polyline | Jalan Lingkungan |
| PolyLineZ | Polyline | Jalan Lingkungan |
| PolyLineZ | Polyline | Jalan Lingkungan |
| PolyLineZ | Polyline | Jalan Lingkungan |
| PolyLineZ | Polyline | Jalan Lingkungan |
| PolyLineZ | Polyline | Jalan Lingkungan |
| PolyLineZ | Polyline | Jalan Lingkungan |
| PolyLineZ | Polyline | Jalan Lingkungan |
| PolyLineZ | Polyline | Jalan Lingkungan |
| PolyLineZ | Polyline | Jalan Lingkungan |

Gambar 4.17 Input Data pada *Theme* Jalan



Gambar 4.18 Input Data pada Theme Pesantren Putra



Gambar 4.19 Input Data pada Theme Pesantren Putri



Gambar 4.20 Input Data pada Theme Pesantren Putra-Putri

4.1.2 Pembuatan Database

Proses pembuatan *database* dilakukan dengan mengkonversi file *.shp menjadi file *.sql menggunakan *software* PostgreSQL dan Postgis. *Software* ini bisa didapatkan secara gratis di *website* berikut www.postgresql.org untuk PostgreSQL, dan

www.postgis.refrains.net untuk Postgis.

Data atribut yang digunakan mengacu pada data yang diisikan pada waktu pembuatan peta.

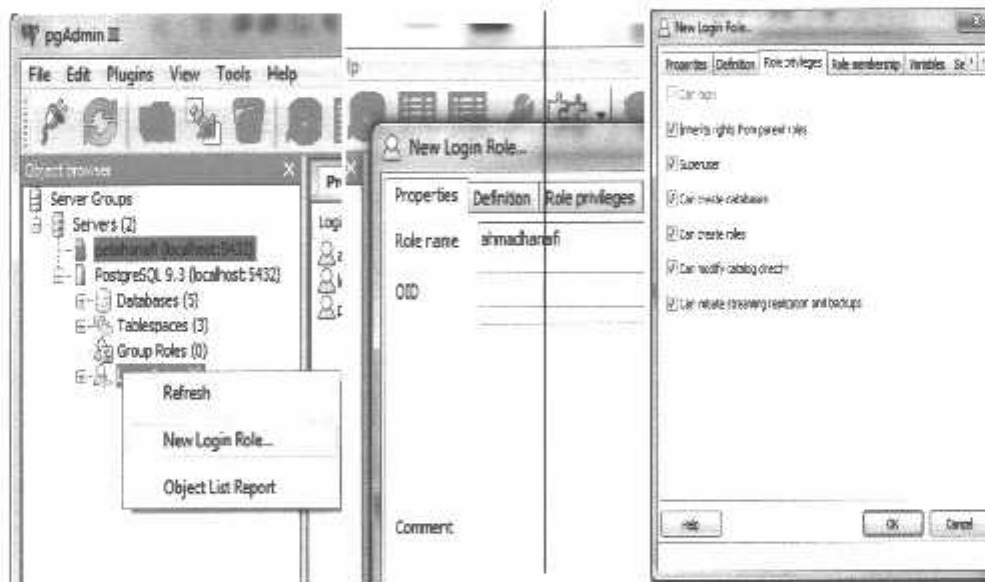
1. Jalankan (install) PostgreSQL dan Postgis. PostgreSQL yang digunakan adalah PostgreSQL versi 8.3 dan Postgis versi 1.3.5 for PostgreSQL 8.3. Pada saat menginstal PostgreSQL catat *password*-nya dan masukkan *password* tersebut pada saat instalasi Postgis.

2. Pada pembahasan diatas kita telah mempunyai beberapa data *.shp yang digunakan dalam aplikasi WebGIS ini, yaitu :

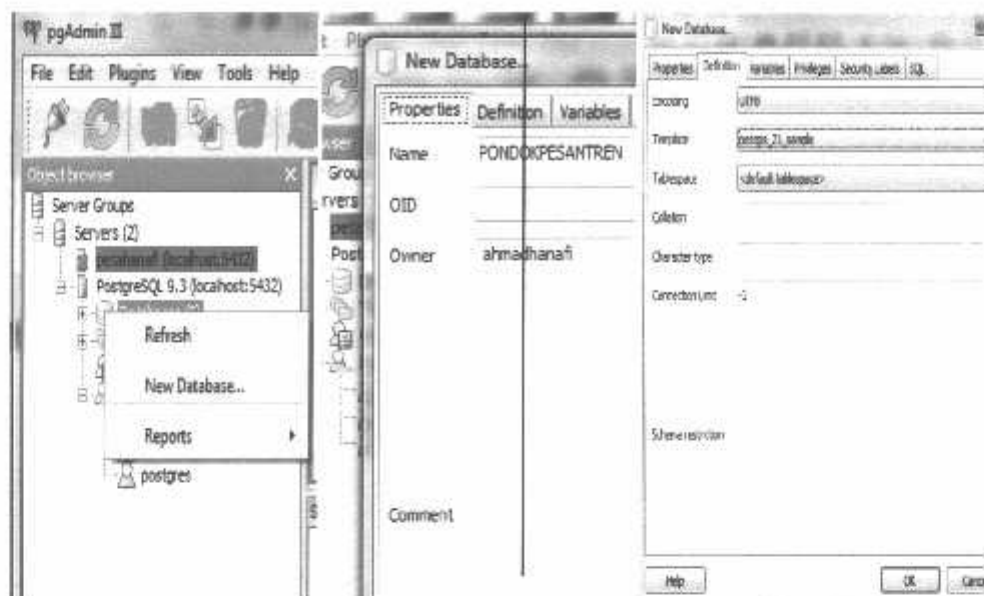
1. Layer polygon = Kecamatan.shp
2. Layer line = Jln.shp
3. Layer point = Putra.shp, Putri.shp, Putra-Putri.shp

Selanjutnya kita akan merubah data-data *.shp di atas menjadi bentuk tabel dalam database PostgreSQL.

3. Membuat *database* baru di PostgreSQL. Buka PgAdmin lalu membuat *user* baru dan *database* baru dengan nama pondok pesantren.



Gambar 4.21 Membuat *User* Baru



Gambar 4.22 Membuat Database Baru

4. Copykan file-file peta berformat *.shp ke dalam direktori bin pada instalasi server PostgreSQL (*C:\Program File\PostgreSQL\8.3\bin*)
5. Jalankan *command prompt*, dan lakukan konversi dari semua file shp ke dalam bentuk *.sql, sehingga diperoleh file-file *.sql. Caranya pada *command prompt* masuk ke direktori *C:\Program Files\PostgreSQL\8.3\bin* dan ketikkan perintah dibawah ini :

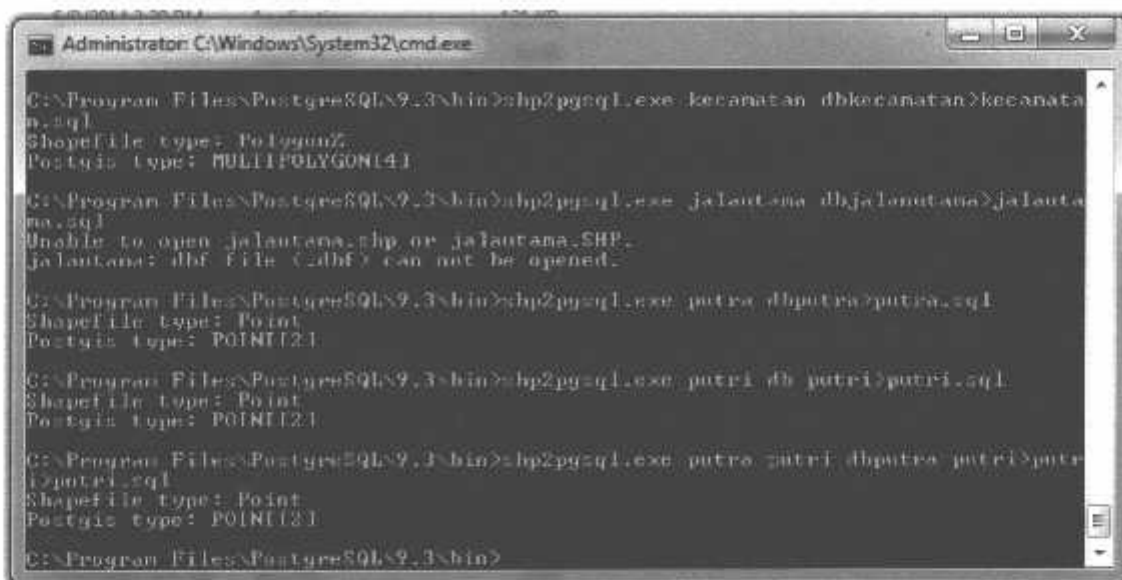
`"shp2pgsql.exe kecamatan dbkecamatan > kecamatan.sql "`

Penjelasan perintah :

- shp2pgsql.exe : Aplikasi pgsqf yang digunakan untuk mengkonversi file *.shp menjadi *.sql, yang letaknya berada di "postgreSQL\8.3\bin".
- kecamatan : Nama file *.shp (data *.shp dan pendukungnya seperti *.dbf, *.shx) yang akan dikonversi. Karena data tersebut sudah berada pada folder \bin maka tidak perlu menuliskan nama filenya. Jika file *.shp berada di lain folder maka alamatnya harus ditulis secara lengkap, [C:\Jombang\kecamatan.shp].
- dbkecamatan : Nama tabel pada database.
- kecamatan.sql : Nama file hasil konversi berekstensi *.sql secara default akan tersimpan pada folder \bin. Jika ingin menyimpan di tempat lain maka alamat penyimpanannya harus ditulis secara lengkap, [C:\Jombang\kecamatan.sql]

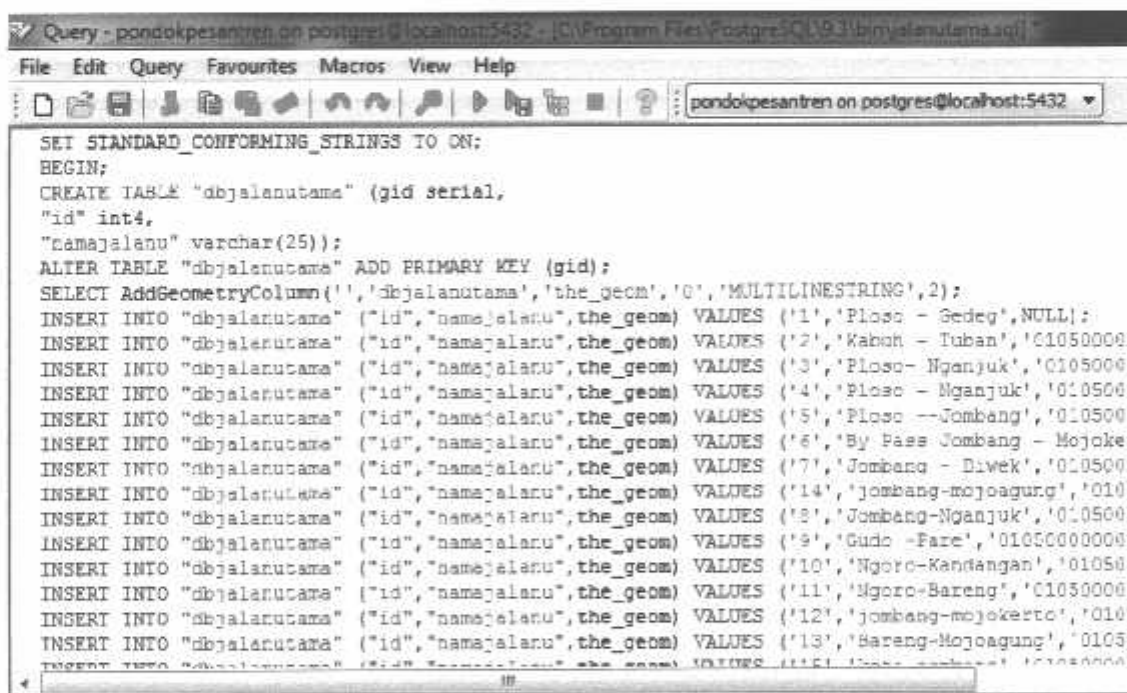
Lakukan hal yang sama untuk mengkonversi file *.shp yang lain. Hasilnya bisa

dilihat pada gambar di bawah ini :



Gambar 4.23 Konversi File *.shp menjadi *.sql

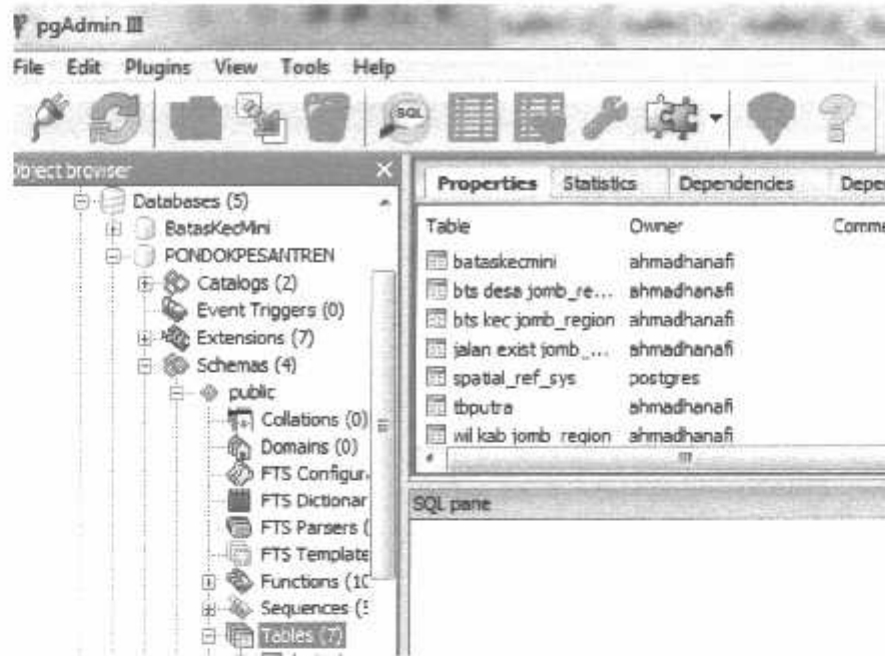
6. Load semua file sql tersebut kedalam *database* postgresQL dengan menggunakan menu *Tools > QueryTools*. Load masing-masing file sql tersebut untuk selanjutnya dieksekusi.



Gambar 4.24 Query Tools

Lakukan langkah yang sama untuk *database* yang berasal dari file *.shp

7. Hasil dari konversi tersebut bisa dilihat di *database* pondok pesantren yang sebelumnya telah dibuat pada PostgreSQL tadi, seperti gambar di bawah ini :



Gambar 4.25 Hasil Konversi Database

4.1.3 Pembuatan Struktur File *.map

1. Instalasi MapServer

Instalasi mapserver cukup dilakukan dengan mengekstrak file *mapserver* berextensi *.zip ke dalam root direktori (di windows menggunakan direktori *c:/*).

Kemudian *double click apache_install.bat*.



Gambar 4.26 Hasil Instalasi ms4w

2. Mapserver membutuhkan sebuah *mapfile* yang berektensi *.map yang berfungsi sebagai file konfigurasi peta. Didalam *mapfile* didefinisikan objek-objek yang akan digunakan, antara lain objek *map*, objek *layer*, objek *class*, objek *label* dan objek *style*. Objek *map* mendefinisikan parameter-parameter yang berlaku umum untuk keseluruhan peta.

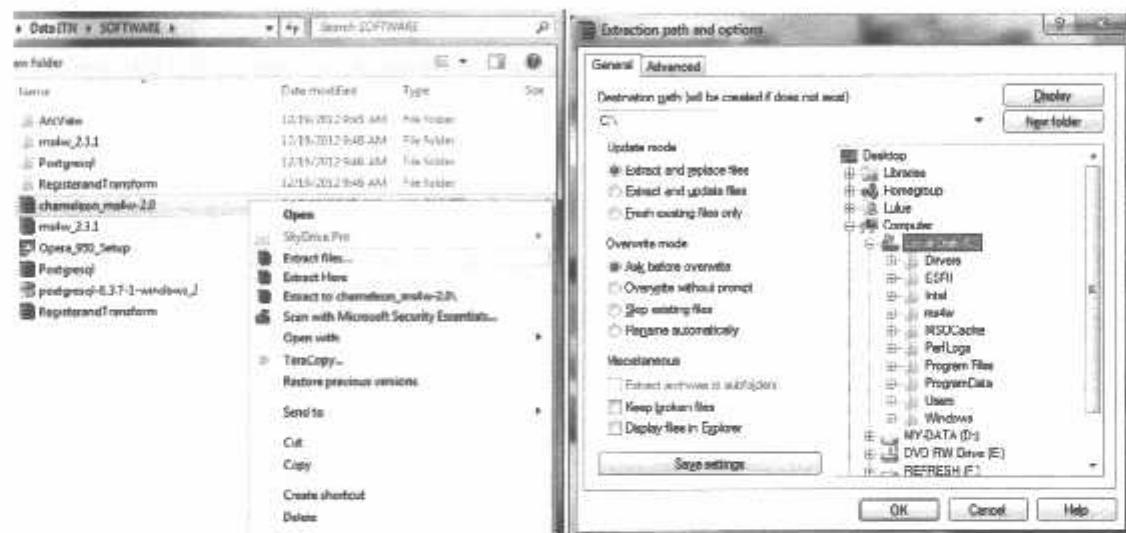
Pendefinisian objek map untuk aplikasi web-GIS yang menyajikan peta kabupaten Jombang dalam bentuk *mapfile* terbagi dalam beberapa *object*, yaitu:

- a) *Object Map*
- b) *Object Web*
- c) *Object Layer*

Script pada masing-masing objek, bisa dilihat pada halaman lampiran.

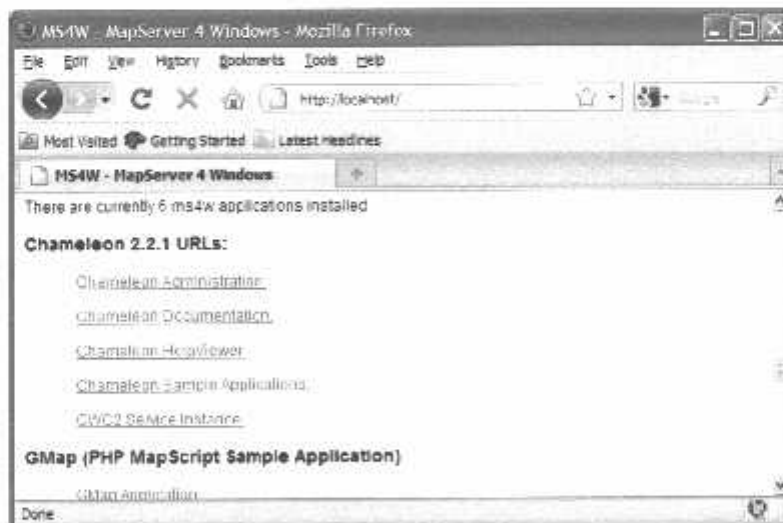
4.1.4 Pembuatan Struktur Template

Proses instalasi Chameleon cukup dilakukan dengan meng-*extract* file *.zip yang didapat kedalam root direktori.



Gambar 4.27 Instalasi Chameleon

Jika instalasi berhasil maka akan tampil link Chameleon pada halaman *localhost*.



Gambar 4.28 Hasil Instalasi Chameleon

Membuat file *template* chameleon sama dengan membuat file html biasa. Hanya

dalam file *template* chameleon dilengkapi dengan pemanggilan terhadap fungsi *CWC2OnLoadFunction()*.

Sementara itu, link *stylesheet*-nya juga perlu merujuk pada file "*cwc2.css*" yang mendefinisikan *style* yang diperlukan oleh *widget*. Kemudian dalam file *template* di sisipkan *widget* chameleon yang merepresentasikan objek *map*. *Widget* yang digunakan pada *template* aplikasi ini terdapat dalam lampiran pada halaman terakhir.

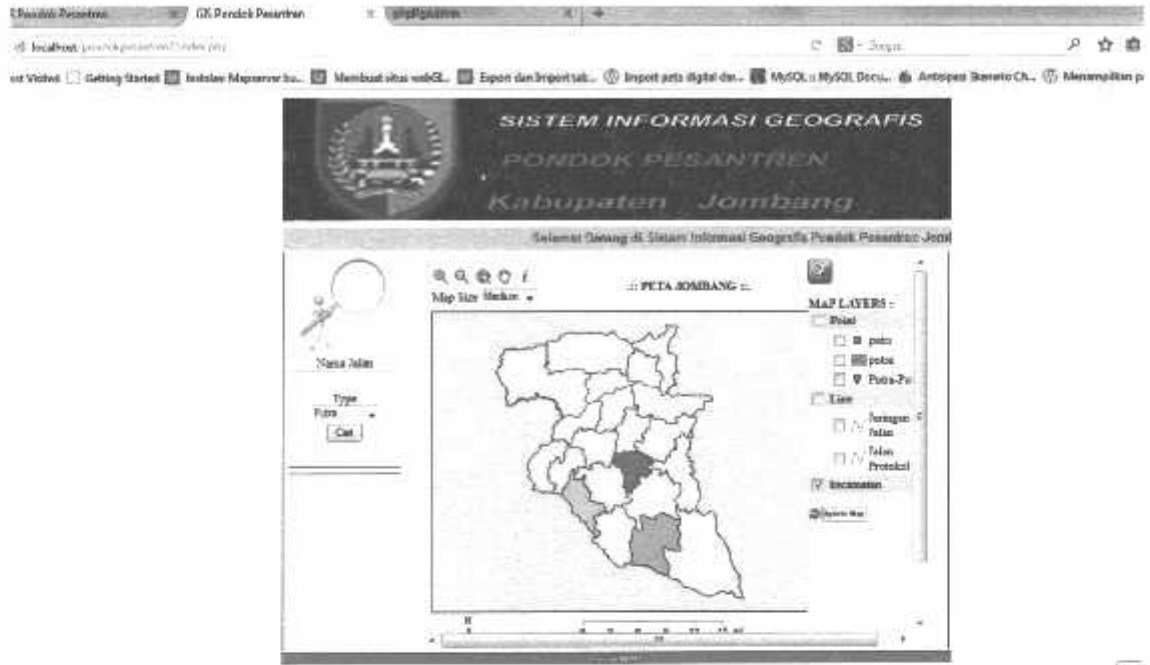
Fungsi dari masing-masing *widget* adalah sebagai berikut :

- a) *Zoom In* : Tombol navigasi untuk memperbesar tampilan map
- b) *Zoom Out* : Tombol navigasi untuk memperkecil tampilan map.
- c) *Zoom to Full Extents* : Tombol navigasi untuk mengembalikan tampilan map ke posisi awal.
- d) *Pan Map* : Tombol navigasi untuk pengguna supaya bisa bebas menggeser tampilan map.
- e) *Identify Feature* : Tombol navigasi untuk melihat detail bagian map yang ditunjuk.
- f) *Map Size* : Navigasi untuk merubah tampilan map (small, medium, dan large).
- g) *Scale Bar* : Mempresentasikan nilai skala actual tampilan peta dalam bentuk grafis
- h) *Legend* : Menampilkan semua layer dalam map
- i) *Kursor Position* : Menunjukkan posisi dalam bentuk koordinat peta bumi sesungguhnya.

4.2 PENGUJIAN

4.2.1 Pengujian Sistem

4.2.1.1 Pengujian terhadap *User*



Gambar 4.29 Tampilan Awal Aplikasi

Aplikasi di atas merupakan tampilan awal bagi pengguna umum (*user* umum), dimana pada aplikasi tersebut, *user* bisa melihat informasi tentang pondok pesantren di kabupaten Jombang yang sudah dikelompokkan menjadi ponpes putra, ponpes putri, dan ponpes putra-putri.

Pengujian pada masing-masing *navigation tools*, adalah sebagai berikut :

1) *Zoom In*

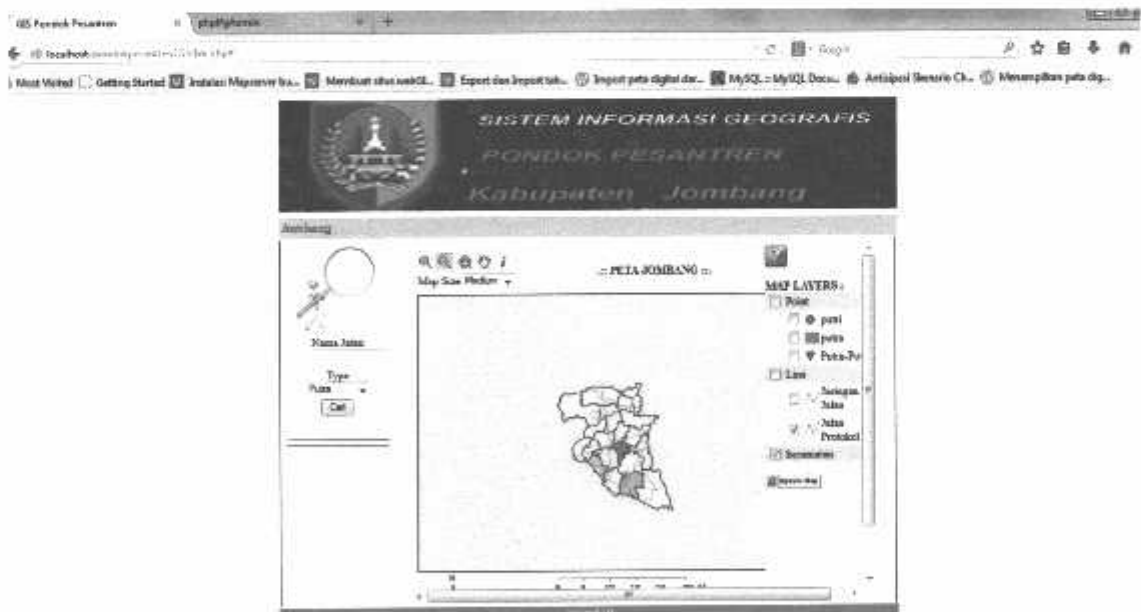
Jika seorang *user* memilih tombol “*Zoom In*” dan mengkliknya pada peta, maka yang terjadi adalah peta tersebut akan bertambah besar seperti pada gambar 4.30 di bawah ini.



Gambar 4.30 Tampilan peta setelah dilakukan *Zoom In*

2) *Zoom Out*

Jika seorang *user* memilih tombol “*Zoom Out*” dan mengkliknya pada peta, maka yang terjadi adalah peta tersebut akan bertambah kecil seperti pada gambar 4.31 di bawah ini.



Gambar 4.31 Tampilan peta setelah dilakukan *Zoom Out*

3) *Zoom to Full Extents*

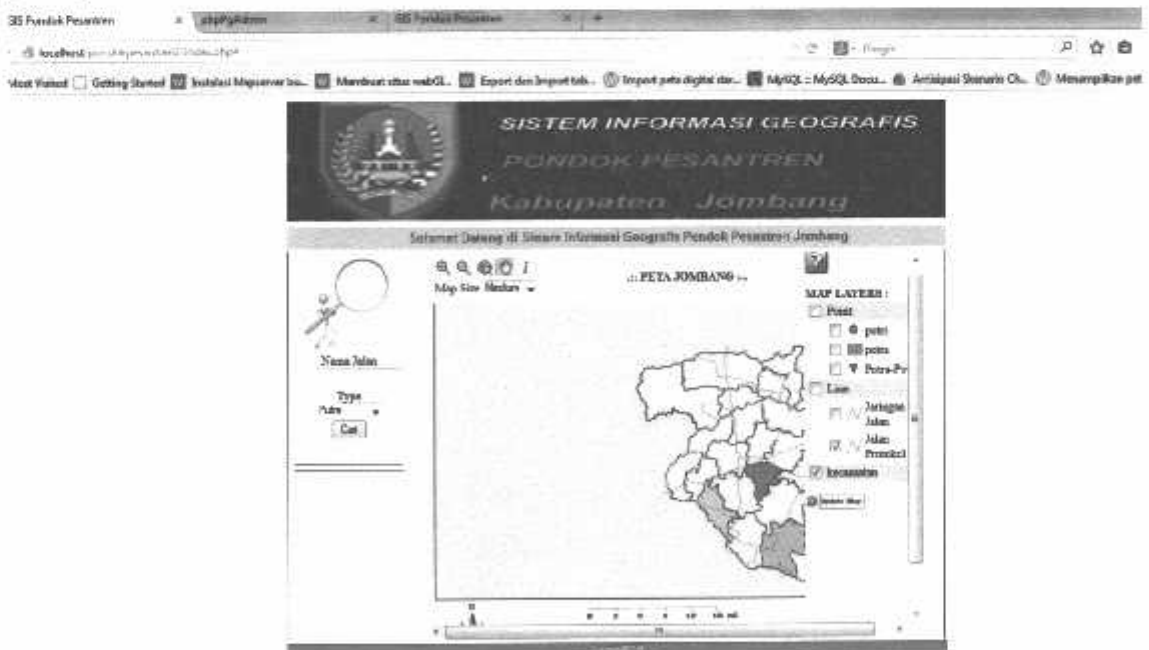
Jika seorang *user* memilih tombol “*Zoom to Full Extents*” dan mengkliknya pada peta, maka yang terjadi adalah peta tersebut akan kembali pada keadaan atau ukuran awal seperti pada gambar 4.32 di bawah ini.



Gambar 4.32 Tampilan peta setelah dilakukan *Zoom to Full Extents*

4) *Pan Map*

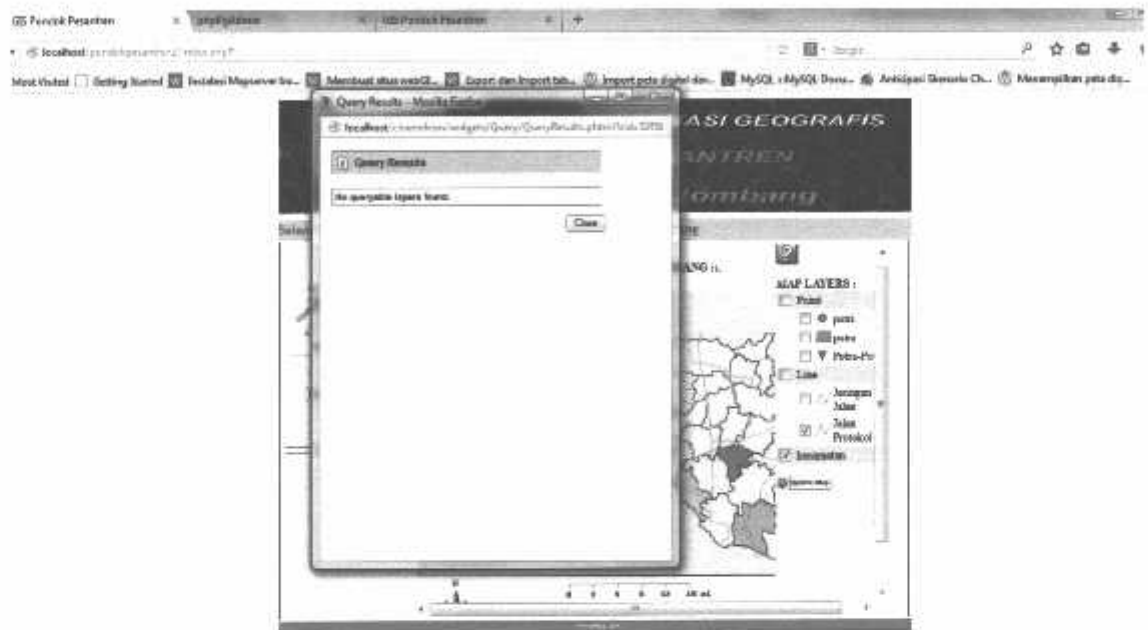
Jika seorang *user* memilih tombol “*Pan Map*” dan mengkliknya pada peta, maka yang terjadi adalah peta tersebut dapat digeser atau dipindah dengan bebas seperti pada gambar 4.33 di bawah ini.



Gambar 4.33 Tampilan peta setelah dilakukan *Pan Map*

5) *Identify Feature*

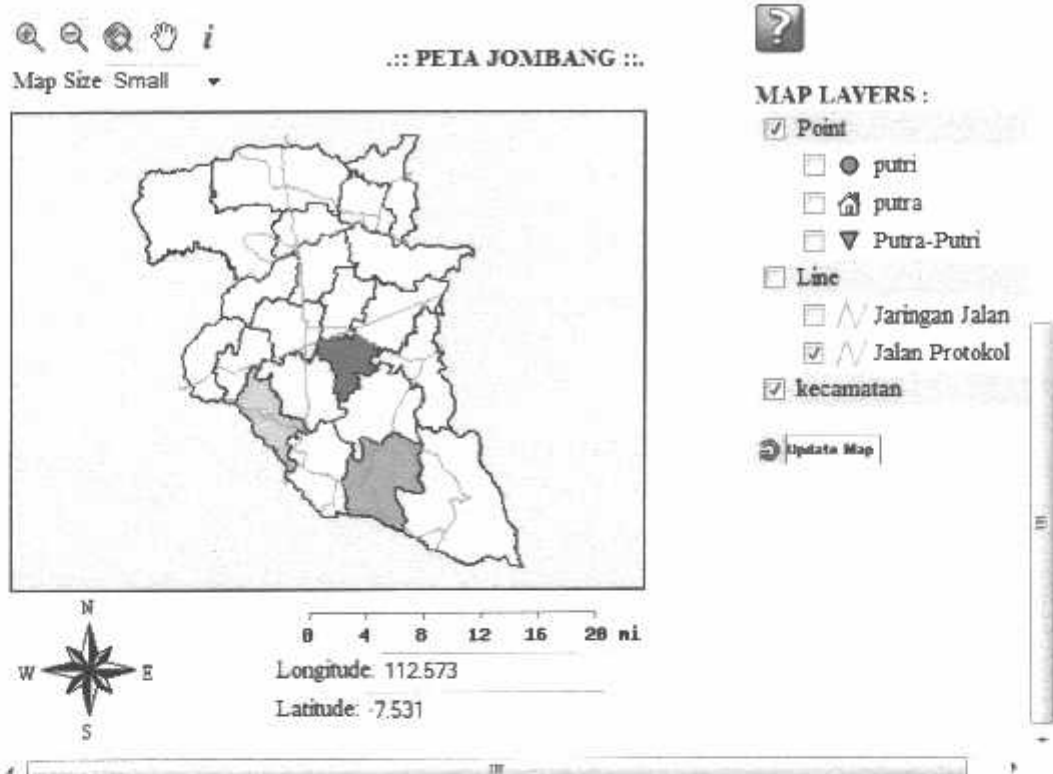
Jika seorang *user* memilih tombol "*Identify Feature*" dan mengkliknya pada salah satu tombol ponpes putra atau ponpes putri atau ponpes putra-putri yang ditunjukkan pada peta, maka yang terjadi akan ditampilkan *Query Result* yang menampilkan informasi sesuai dengan data yang dipilih tadi, seperti pada gambar 4.34 di bawah ini.



Gambar 4.34 Tampilan peta setelah dilakukan *Identify Feature*

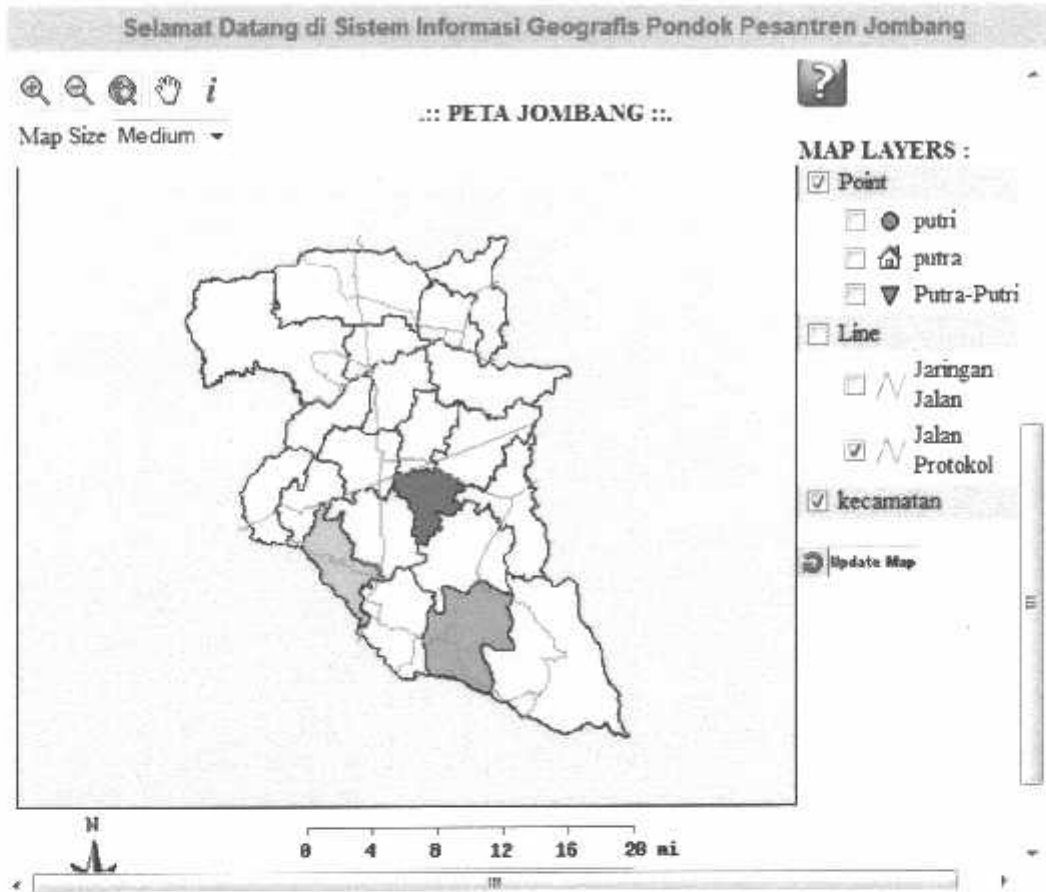
6) *Map Size*

Jika seorang *user* memilih pilihan *Map Size* "*Small*", maka yang terjadi adalah ukuran peta tersebut menjadi kecil atau sesuai dengan ukuran *small* yang telah diatur dalam program yaitu 400 x 300, seperti pada gambar 4.35 di bawah ini.



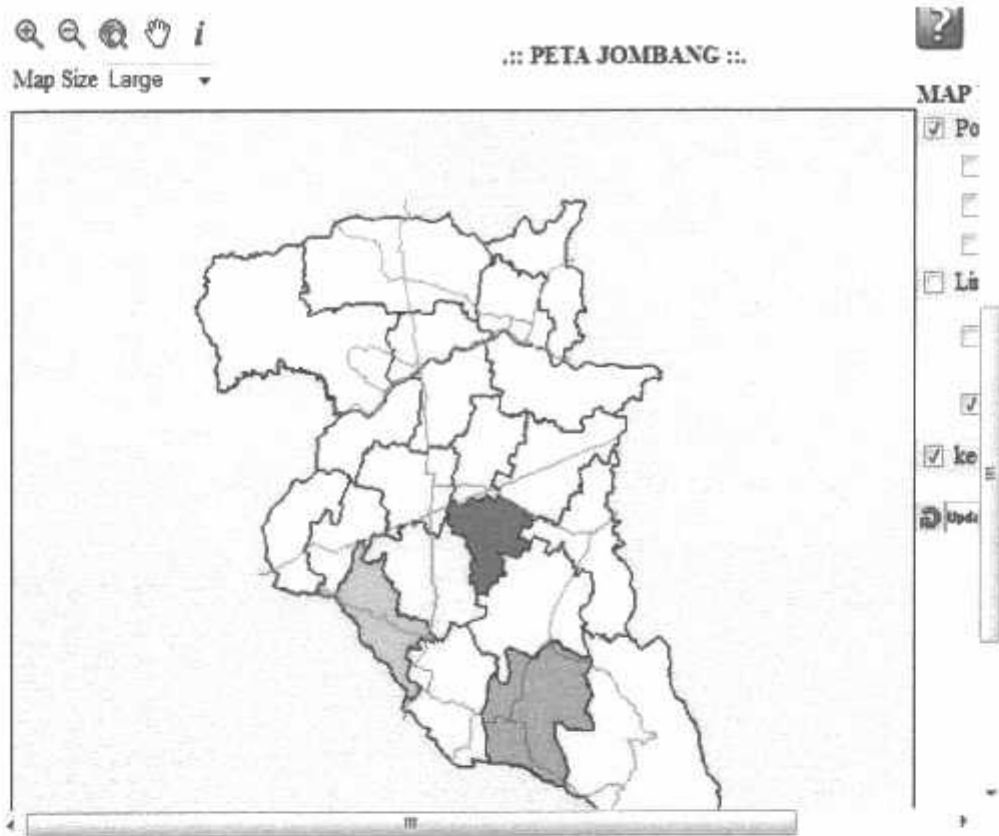
Gambar 4.35 Tampilan peta setelah dilakukan *Map Size "Small"*

Jika seorang *user* memilih pilihan *Map Size "Medium"*, maka yang terjadi adalah ukuran peta tersebut menjadi ukuran sedang atau sesuai dengan ukuran *medium* yang telah diatur dalam program yaitu 500 x 400, seperti pada gambar 4.36 di bawah ini.



Gambar 4.36 Tampilan peta setelah dilakukan *Map Size* "Medium"

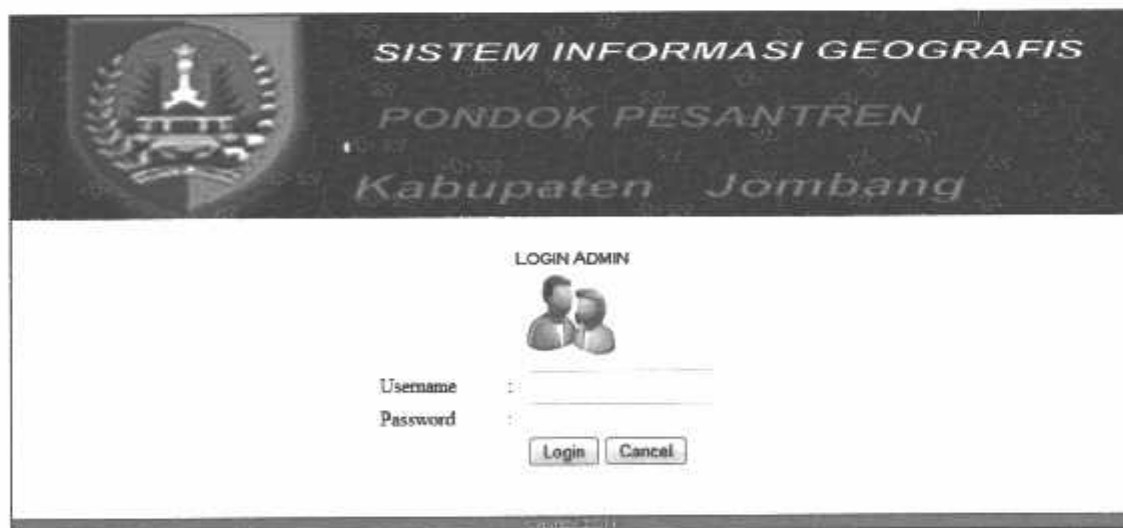
Jika seorang *user* memilih pilihan *Map Size* "Large", maka yang terjadi adalah ukuran peta tersebut menjadi besar atau sesuai dengan ukuran *large* yang telah diatur dalam program yaitu 600 x 500, seperti pada gambar 4.37 di bawah ini.



Gambar 4.37 Tampilan peta setelah dilakukan *Map Size "Large"*

4.2.1.2 Pengujian terhadap Admin

Seorang *admin* harus melakukan *login admin* terlebih dahulu untuk dapat mengakses data yang ada dalam *database*. Setelah *login*, akan ditampilkan halaman *login admin* seperti gambar 4.38.



Gambar 4.38 Tampilan *Login Admin*

Setelah halaman awal *admin* muncul, *admin* bisa melakukan *login* dengan memasukkan *username* dan *password* seperti pada gambar 4.39.



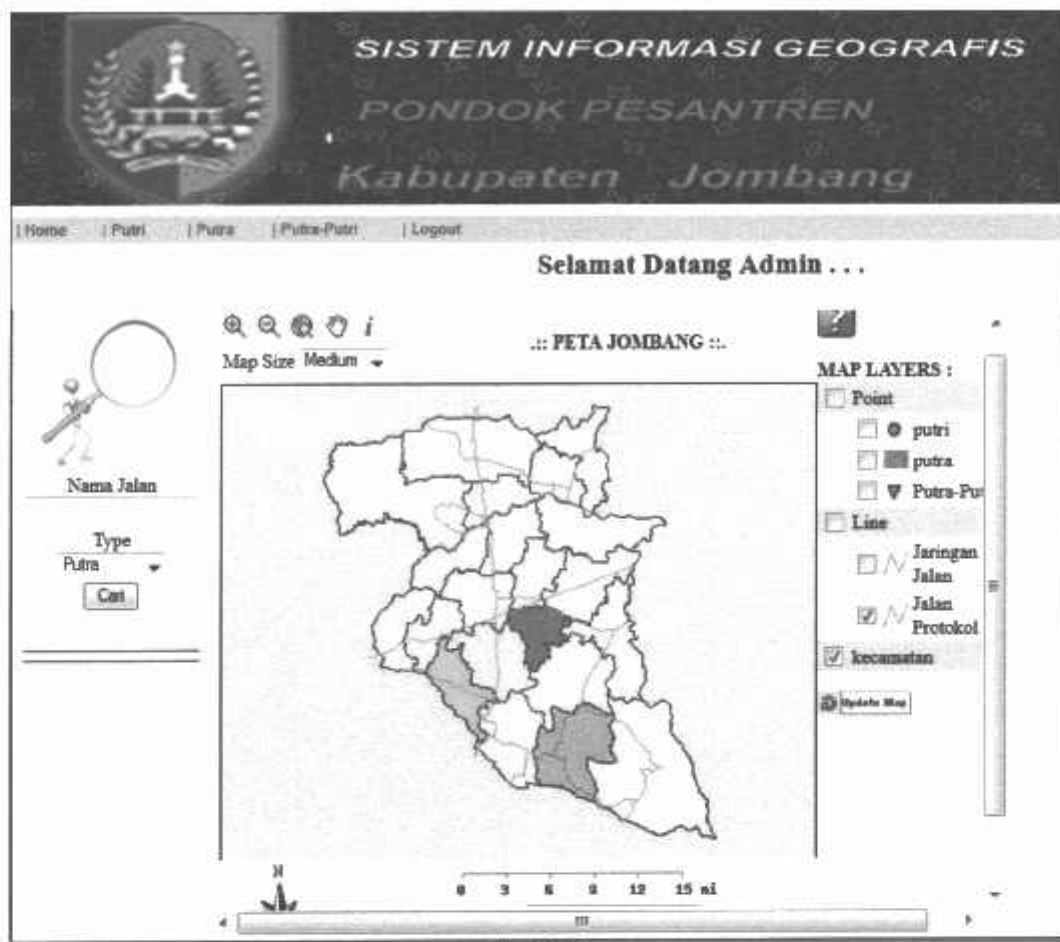
Gambar 4.39 *Input Username & Password*

Jika *username* dan *password* salah maka akan ditampilkan pesan peringatan seperti pada gambar 4.40. Dan disana terdapat perintah “Ulangi Login”, yang akan mengarah pada halaman *login admin*.



Gambar 4.40 Peringatan Salah *Username & Password*

Jika *username* dan *password* benar, maka *admin* akan diarahkan ke halaman awal *admin* seperti pada gambar 4.41.



Gambar 4.41 Tampilan Awal *Admin*

Secara keseluruhan, tampilan awal *admin* sama dengan tampilan awal *user*. Yang membedakan keduanya adalah pada halaman awal *admin* terdapat beberapa fasilitas menu di bawah *header* yang dapat digunakan untuk proses manipulasi data pada peta. Terdapat lima menu, yaitu *home*, *putra*, *putrid*, *putra-putri* dan *logout*.

a) *Home*

Tampilan awal *admin* yang bisa digunakan untuk melihat peta beserta informasinya.

b) *Putra*

Fasilitas yang digunakan untuk melihat data jenis pesantren putra secara keseluruhan, dan bisa dilakukan proses manipulasi data pesantren putra (*insert*, *edit*, dan *delete*).

c) *Putri*

Fasilitas yang digunakan untuk melihat data pesantren putri secara keseluruhan, dan bisa dilakukan proses manipulasi data pesantren putri (*insert*, *edit*, dan *delete*).

d) *Putra-Putri*

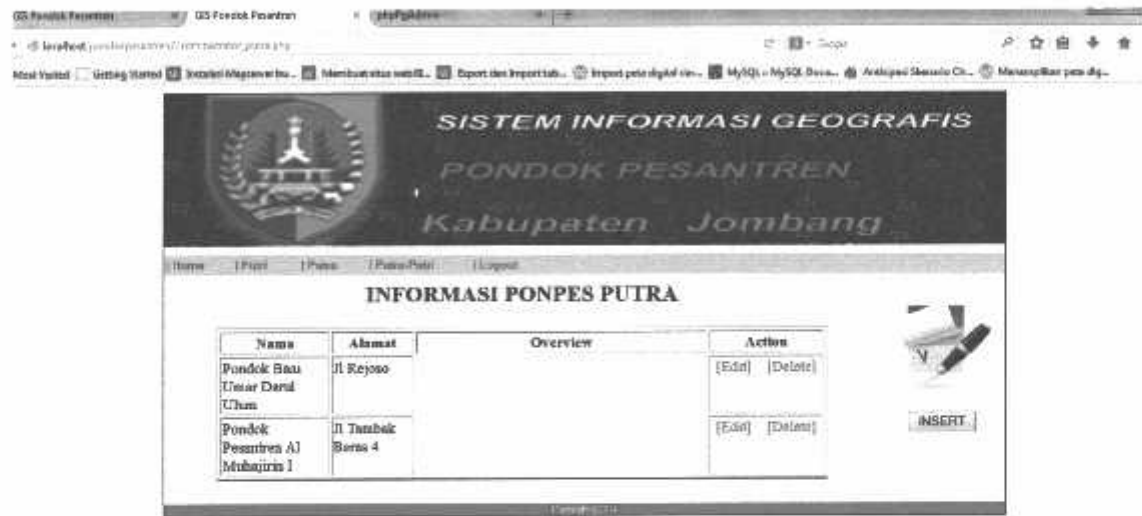
Fasilitas yang digunakan untuk melihat data pesantren putra-putri secara

keseluruhan, dan bisa dilakukan proses manipulasi data pesantren putra-putri (*insert, edit, dan delete*).

c) *Logout*

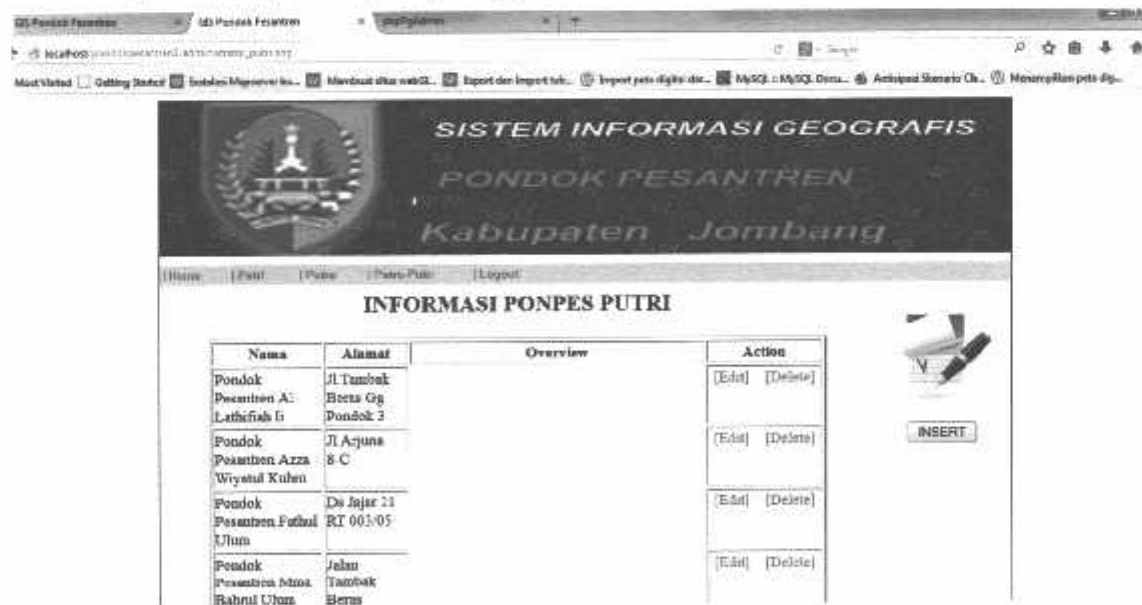
Fasilitas untuk keluar dari sistem, dan akan diarahkan ke halaman *login admin*.

Jika seorang *admin* memilih menu “putra”, maka akan ditampilkan halaman informasi pesantren seperti pada gambar 4.42 di bawah ini.



Gambar 4.42 Tampilan Manipulasi Informasi Pesantren Putra

Jika seorang *admin* memilih menu “putri”, maka akan ditampilkan halaman informasi pesantren putri seperti pada gambar 4.43 di bawah ini.



Gambar 4.43 Tampilan Manipulasi Informasi Pesantren putri

Jika seorang *admin* memilih menu “putra-putri”, maka akan ditampilkan halaman informasi putra-putri seperti pada gambar 4.44 di bawah ini.



Gambar 4.44 Tampilan Manipulasi Informasi putra putri

Pada masing-masing tampilan informasi data di atas, terdapat fasilitas untuk proses manipulasi data yaitu, *insert*, *edit*, dan *delete*. Jika menu *edit* pada *action* dipilih, maka akan ditampilkan halaman seperti pada gambar 4.45.



Gambar 4.45 Edit Tabel Pesantren putra - putri

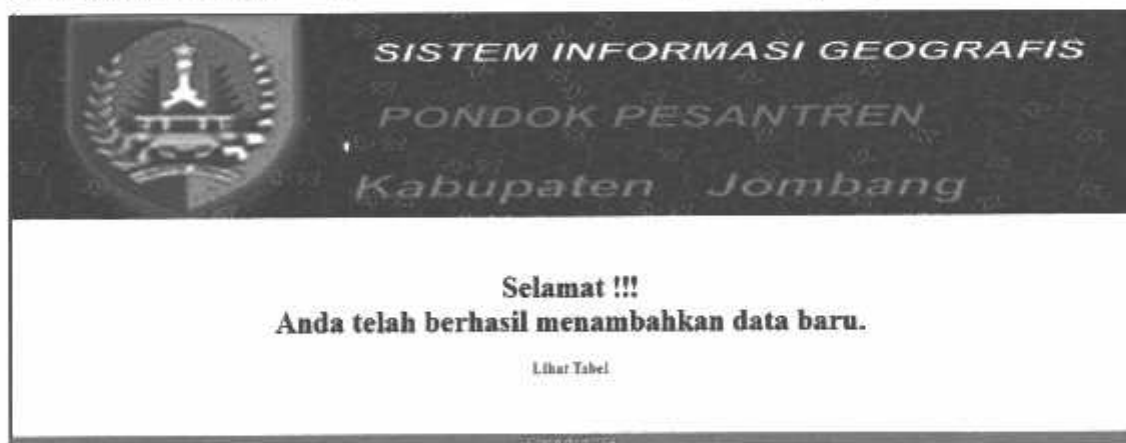
Jika menu *insert* dipilih, maka akan ditampilkan halaman seperti pada gambar

4.46. Disini akan dilakukan proses penambahan data baru, misalkan untuk menambahkan data pesantren, maka *admin* akan memasukkan nama pesantren, alamat pesantren, *overview* atau keterangan, lalu koordinat lintang dan bujur, serta menambahkan gambar. Koordinat lintang dan bujur, bisa didapatkan di *google map*.



Gambar 4.46 Insert Tabel Pesantren

Jika form di atas telah terisi semua klik tombol “Simpan”. Dan untuk melihat data yang telah di rubah klik pada “menu” tabel yang telah di tambah dengan data baru di atas. Atau bisa juga dengan melihat tabel database pada PostgreSQL.



Gambar 4.47 Hasil Insert Data Pesantren

4.2.2 Pengujian *Browser*

Pengujian ini dilakukan terhadap tiga *browser*, yaitu *Internet Explorer*, *Mozilla Firefox*, dan *Opera*. Pada masing-masing *browser*, dilakukan proses pengujian

keseluruhan sistem dengan keterangan hasil SB (Sangat Baik), B (Baik), dan K (Kurang Baik). Berikut hasil dari pengujian *browser* yang telah dilakukan.

Tabel 4.2 Tabel Pengujian *Browser*

| BROWSER | | Internet Explorer | Mozilla Firefox | Opera |
|-----------|---|-------------------|-----------------|-------|
| PENGUJIAN | | | | |
| 1. | <i>USER</i> | | | |
| | a. <i>Navigation Tools</i> | | | |
| | 1. <i>Zoom In</i> | ✓ | ✓ | ✓ |
| | 2. <i>Zoom Out</i> | ✓ | ✓ | ✓ |
| | 3. <i>Zoom to Full Extents</i> | ✓ | ✓ | ✓ |
| | 4. <i>Pan Map</i> | ✓ | ✓ | ✓ |
| | 5. <i>Identify Feature</i> | ✓ | ✓ | ✓ |
| | 6. <i>Map Size</i> | ✓ | ✓ | ✓ |
| | b. <i>Memilih Map Layer</i> | ✓ | ✓ | ✓ |
| | c. <i>Pencarian</i> | ✓ | ✓ | ✓ |
| 2. | <i>ADMIN</i> | | | |
| | a. <i>Login Admin</i> | ✓ | ✓ | ✓ |
| | b. <i>Home</i> | | | |
| | 1. <i>Navigation Tools</i> | ✓ | ✓ | ✓ |
| | 2. <i>Memilih Map Layer</i> | ✓ | ✓ | ✓ |
| | 3. <i>Pencarian</i> | ✓ | ✓ | ✓ |
| | c. <i>Pesantren Putra</i> | | | |
| | 4. <i>Insert Pesantren Putra</i> | ✓ | ✓ | ✓ |
| | 5. <i>Edit Pesantren Putra</i> | ✓ | ✓ | ✓ |
| | 6. <i>Delete Pesantren Putra</i> | ✓ | ✓ | ✓ |
| | d. <i>Pesantren Putri</i> | | | |
| | 7. <i>Insert Pesantren Putri</i> | ✓ | ✓ | ✓ |
| | 8. <i>Edit Pesantren Putri</i> | ✓ | ✓ | ✓ |
| | 9. <i>Delete Pesantren Putri</i> | ✓ | ✓ | ✓ |
| | e. <i>Pesantren Putra-Putri</i> | | | |
| | 10. <i>Insert Pesantren Putra-Putri</i> | ✓ | ✓ | ✓ |
| | 11. <i>Edit Pesantren Putra-Putri</i> | ✓ | ✓ | ✓ |
| | 12. <i>Delete Pesantren Putra-Putri</i> | ✓ | ✓ | ✓ |

Keterangan : ✓ = Berfungsi

× = Tidak Berfungsi

Dari hasil tabel pengujian di atas, dapat disimpulkan bahwa aplikasi SIG objek pondok pesantren ini mampu berfungsi dengan baik.

4.2.3 PENGUJIAN USER

Pengujian User dilakukan kepada 20 responden yaitu kepada masyarakat umum. Hasil pengujian user ditunjukkan table 4.3.

Tabel 4.3 Hasil Pengujian User.

| Pertanyaan | K | C | B |
|--|---|---|----|
| 1. Apakah system yang dibuat mampu memberikan informasi pondok pesantren | 2 | 3 | 15 |

Keterangan : K = Kurang

C = Cukup

B = Baik

$$B = \frac{15}{20} \times 100 = 75\%$$

$$C = \frac{3}{20} \times 100 = 15\%$$

$$K = \frac{2}{20} \times 100 = 10\%$$

BAB V

PENUTUP

5.1 KESIMPULAN

Dari hasil analisa dan desain serta implementasi dan pengujian aplikasi Sistem Informasi Geografis Pemetaan Objek Pondok pesantren di Jombang, maka dapat dibuat kesimpulan sebagai berikut:

- 1) Pemetaan lokasi pondok pesantren berdasarkan kategorinya yaitu, Ponpes Putra, Ponpes Putri, dan Ponpes Putra Putri beserta penyajian informasi pendukung lainnya mengenai pondok pesantren tersebut dapat dilakukan dengan menggunakan Sistem Informasi Geografis (SIG) dan dapat berjalan 100% sesuai fungsinya di browser *Internet Explorer*, *MozillaFirefox* dan *Opera*.
- 2) Dari hasil pengujian *User* menunjukkan bahwa 80% aplikasi dapat berfungsi sebagai pendukung informasi lokasi pondok pesantren di Jombang.

5.2 SARAN

- 1) Pada aplikasi ini, wilayah yang dicakup hanya Jombang. Kedepannya bisa dikembangkan untuk wilayah yang lebih luas, seperti misalnya mencakup propinsi Jawa Timur secara keseluruhan. Sehingga informasi yang disajikan lebih beragam, dan memberikan banyak pilihan pondok pesantren bagi masyarakat.
- 2) Untuk berikutnya dapat disediakan update berkala tentang informasi dan konten terbaru dalam mendapatkan informasi tentang objek pondok pesantren Jombang.
- 3) Pengembangan selanjutnya bisa ditambahkan informasi fasilitas akomodasi yang ada di objek pondok pesantren dan sekitarnya.

DAFTAR PUSTAKA

- [1] Riyanto , Putra, Prinali Eka. 2009. Pengembangan Aplikasi Sistem Informasi Geografis Berbasis Dekstop dan *Web*. Yogyakarta : Gava Media
- [2] Ganad. 2011. *Software yang Mendukung Sistem Informasi Geografis*. [Online] Tersedia : www.ganadp.wordpress.com [05 Juli 2014]
- [3] Budiyanto, Eko. 2010. Sistem Informasi Georafis dengan Arcview GIS. Yogyakarta : Andi
- [4] Suyanto. 2000. Pengertian Sistem Informasi. [Online] Tersedia : www.blog.re.or.id/pengertian-informasi.html. [05 Juli 2014]
- [5] Fathansyah, 2002. *Basis Data*. Bandung : Informatika
- [6] Valacich, George, and Hoffer, 2004. Entity Relationship Diagram (ERD).[Online] Tersedia : www.digilib.petra.ac.id [05 Juli 2014]
- [7] Kendall, Kenneth E., 2002. Relationship and DFD. [Online] Tersedia : www.digilib.petra.ac.id [05 Juli 2014]
- [8] Kadir, Abdul. 1999. *Konsep dan Tuntunan Praktis Basis Data*. Yogyakarta : Andi.
- [9] Arsana, 2008. Flowchart. [Online] Tersedia : www.digilib.petra.ac.id. [05 Oktober 2013]
- [10] Prahasta, Eddy. 2007. Membangun Aplikasi Web-based GIS dengan *Mapserver*. Bandung : Informatika

LAMPIRAN



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Raya Karanglo Km. 2 Malang

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : Ahmad Hanafi
NIM : 0918902
Jurusan : Teknik Informatika S-1
Judul : Rancang Bangun Aplikasi Sistem Informasi Geografis Berbasis Web
Untuk Pemetaan Pondok Pesantren Di Jombang

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :
Hari : Kamis
Tanggal : 25 Agustus 2014

Panitia Ujian Skripsi :

Ketua Majelis Penguji

Joseph Dedv Irawan, ST, MT
NIP.197404162005011002

Anggota Penguji :

Penguji Pertama

SONNY PRASETIO, ST.MT
NIP.P.1031000433

Penguji Kedua

FEBRIANA SANTI WAHYUNI, S.Kom.M.Kom
NIP.P. 1031000425