SKRIPSI

PENGEMBANGAN APLIKASI PENCARIAN RUTE TERPENDEK DENGAN METODE ALGORITMA A* BERBASIS WEB

Disusun Oleh :

REYDIKA ILHAM

06.12.658

MILIK PERPUSTAKAAN ITN MALANG

JURUSAN TEKNIK ELEKTRO S-1

KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL MALANG

2010

LEMBAR PERSETUJUAN

PENGEMBANGAN APLIKASI PENCARIAN RUTE TERPENDEK DENGAN METODE ALGORITMA A* BERBASIS WEB

SKRIPSI

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

Disusun Oleh :

REYDIKA ILHAM

NIM : 06.12.658

Diperiksa dan Disetujui

Dosen Pembimbing I	Dosen Pembimbing II
Dr. Aryuanto S, ST, MT	**Ahmad Faisol, ST**
NIP.Y. 1030800417	

Mengetahui,

Ketua Jurusan Teknik Elektro S-1

Ir. Yusuf Ismail Nakhoda, MT

NIP.Y. 1018800189

JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2010

PENGEMBANGAN APLIKASI PENCARIAN RUTE TERPENDEK DENGAN METODE ALGORITMA A* BERBASIS WEB

Reydika Ilham
Konsentrasi Teknik Komputer dan Informatika, Jurusan Teknik Elektro S-1
Fakultas Teknologi Industri, Institut Teknologi Nasional Malang
Jl. Raya Karanglo Km 2 Malang
e-mail : *cibud.chi@gmail.com*

Abstrak

Melihat adanya angka kepadatan kendaraan bermotor di Kota Malang yang semakin meningkat dan menimbulkan kemacetan pada ruas-ruas jalan tertentu, maka dibutuhkan suatu cara untuk mengatasinya. Salah satunya adalah mencari jalur alternatif, dengan mencari jalur lintasan terpendek dari lokasi asal ke lokasi tujuan.

Di zaman modern seperti saat ini peta masih digunakan oleh sebagian besar orang untuk mencari jalur alternatif dengan menelusuri jalan-jalan mana saja yang dapat dilalui dari satu lokasi ke lokasi lainnya. Oleh karena semakin berkembangnya internet serta kemudahan dalam mengakses internet, maka dalam pembuatan skripsi ini dirancanglah suatu aplikasi web yang dapat memberikan informasi kepada penggunanya jalur-jalur mana saja yang sebaiknya dilalui.

Pencarian rute jalan pada aplikasi web tersebut dilakukan dengan menggunakan metode algoritma A yang biasa digunakan dalam game-game petualangan, dimana algoritma akan berjalan dengan melakukan scanning jalur-jalur yang dapat dilalui dan melakukan perhitungan bobot yang akan menghasilkan pencarian jalan yang lebih akurat.*

Dalam penggunaan aplikasi ini, user dapat menginputkan jalan asal beserta jalan tujuan dengan hasil dari pencarian rute jalan tersebut berupa visual yang ditampilkan didalam web.

Kata kunci: *pencarian rute, peta digital, A*, rute tercepat.*

Abstract

Seeing the number density of vehicles in the city of Malang on the rise with congestion on certain road sections, so needed a way to cope. One is looking for an alternative route, by finding the shortest path from point of origin to the destination. location.

*In modern times like this when the map is still used by most people to seek alternative routes to explore the streets of anywhere that can be passed from one location to another. Because of the growing Internet and the ease in accessing the internet, so in making this paper designed a web application that can provide information to users anywhere pathways that should pass. Route search on the web application is done using A * algorithm method commonly used in adventure games, where the algorithm will run with the scanning lines that can go through and do the calculation of weights that will produce more accurate search path.*

In these applications, users can menginputkan way home along the road with the results of a search destination route is displayed in visual form on the web.

Keywords: *pathfinding, digital map, A*, fastest path.*

KATA PENGANTAR

Dengan mengucap syukur kehadirat Tuhan YME yang telah memberikan kekuatan, kesabaran, bimbingan dan perlindungan sehingga penulis dapat menyelesaikan laporan skripsi dengan judul **"PENGEMBANGAN APLIKASI PENCARIAN RUTE TERPENDEK DENGAN METODE ALGORITMA A* BERBASIS WEB"**

Pembuatan skripsi ini disusun guna memenuhi syarat akhir kelulusan pendidikan jenjang Strata I di Institut Teknologi Nasional Malang. Dalam penyusunan skripsi ini penulis banyak mendapat bantuan baik moril maupun materiil, saran dan dorongan semangat dari berbagai pihak, untuk itu penulis mengucapkan terima kasih kepada :

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE selaku rektor ITN Malang
2. Bapak Ir. Sidik Noertjahjono, MT selaku Dekan Fakultas Teknologi Industri.
3. Bapak Ir. Yusuf Ismail Nakhoda, MT selaku Ketua Jurusan Teknik Elektro S-1 ITN Malang.
4. Bapak Dr. Eng. Aryuanto S, ST, MT selaku Dosen Pembimbing I.
5. Bapak Ahmad Faisol, ST selaku Dosen Pembimbing II.
6. Semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan. Oleh sebab itu kritik dan saran yang membangun sangat diharapkan.

Akhir kata, penulis mohon maaf kepada semua pihak bilamana selama penyusunan skripsi ini penyusun membuat kesalahan secara tidak sengaja dan semoga skripsi ini dapat bermanfaat bagi kita semua.

Malang, September 2010

Penulis

DAFTAR ISI

DAFTAR TABEL

BAB II DASAR TEORI

BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

DAFTAR GAMBAR

BAB II DASAR TEORI

BAB III PERANCANGAN SISTEM

BAB IV IMPLEMENTASI DAN PENGUJIAN HASIL

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dengan meningkatnya angka kepadatan pengguna jalan di Kota Malang membuat terjadinya kemacetan di beberapa ruas jalan yang sering mengganggu kegiatan sehari-hari. Setiap pengguna jalan pastinya menginginkan sampai di tempat tujuan tepat waktu, Oleh karena itu, dibutuhkan suatu cara untuk menanggulangi kemacetan tersebut. Salah satunya adalah mencari lintasan terpendek dari tempat asal ke tempat tujuan, dimana lintasan terpendek ini memperhitungakn titik-titik kemacetan yang sering terjadi.

Di zaman yang serba modern saat ini peta masih digunakan oleh sebagian besar orang baik itu untuk mencari rute yang harus dilalui dari suatu lokasi ke lokasi lainnya, ataupun untuk sekedar mencari suatu lokasi atau tempat tertentu dalam suatu daerah. Untuk mencari rute yang terpendek dari suatu lokasi ke lokasi yang lainnya, kita harus menelusuri gambar jalan yang terdapat pada peta satu persatu dan menghitung jarak dari rute yang dapat kita lewati dengan melakukan perhitungan skala. Jika kita melakukan pencarian dan perhitungan dengan manual maka akan menghabiskan waktu yang cukup lama dan memerlukan ketelitian karena banyaknya kemungkinan rute yang bisa dilewati untuk mencapai suatu lokasi tertentu dari lokasi yang lainnya.

Seiring perkembangan internet yang semakin pesat dan kemudahan akses internet pada saat ini, maka dalam tugas akhir pencarian rute terpendek ini dibuat dengan berbasiskan web menggunakan metode A* (a star) dengan hasil pencarian yang lebih optimum dari algoritma dijkstra[1]. Dari segi kecepatan, pencarian dengan menggunakan algoritma dijkstra lebih lambat dari algoritma A*[8].

Dengan adanya pencarian rute via web ini nantinya diharapkan dapat digunakan sebagai sarana untuk melakukan pencarian rute yang lebih efektif dan efisien.

1.2 Rumusan Masalah

Berdasarkan uraian pada latar belakang di atas maka dalam tugas akhir ini dapat dirumuskan perumusan masalah yaitu bagaimana cara membuat program pencari rute terpendek dengan banyaknya kemungkinan yang ada serta melewati jalan mana saja.

1.3 Tujuan

Tujuan dari skripsi ini adalah untuk membuat suatu aplikasi yang dapat membimbing seseorang/pengemudi dalam berkendara di wilayah kota Malang, sehingga bisa mencapai tujuan dengan melalui rute terpendek yang ditentukan.

1.4 Batasan Masalah

Batasan masalah yang diambil pada penulisan skripsi ini diharapkan mampu membatasi pembahasan agar sesuai dengan tujuan penelitian itu sendiri. Adapun batasan masalah yang diajukan adalah sebagai berikut :

1. Lokasi peta yang digunakan dan penunjuk arah kendaraan hanya di daerah kota Malang.
2. Aplikasi mencari rute terpendek dengan menggunakan Algoritma A* (A-star).
3. Panduan arah dengan menggunakan bantuan garis yang menghubungkan antar titik-titik jalan pada peta.
4. Jalan searah dan dua arah diperhitungkan.
5. Pencarian rute jalan hanya berlaku untuk jalan besar/jalan raya, tidak berlaku untuk jalan kecil/gang.
6. Terdapat fasilitas tambahan : Daftar lokasi/tempat-tempat menarik (Hotel, Mall, Plaza, dll) dan daftar lokasi/tempat-tempat penting (Rumah sakit, Kantor Polisi).

1.5 Metodologi Penelitian

Dalam menyelesaikan permasalahan yang dihadapi, dilakukan penelitian dengan menggunakan metode pengumpulan data dan metode implementasi.

1.5.1 Metode Pengumpulan Data

Data merupakan sumber atau bahan mentah yang sangat berharga bagi proses menghasilkan informasi. Oleh sebab itu dalam pengambilan data perlu dilakukan penanganan secara cermat dan hati-hati, sehingga data yang diperoleh dapat bermanfaat dan berkualitas.

Dalam pengumpulan data penyusun mengunakan metode sebagai berikut :

1. Studi Lapangan

   Dengan metode ini data-data diperoleh langsung dari sumber yang bersangkutan, dimana peneliti berhadapan langsung dengan obyek yang diteliti, yang dilakukan dengan cara :

   a. Survey

      Teknik pengumpulan data dengan cara terjun secara langsung dan mencatat secara sistematis terhadap obyek masalah.

   b. Wawancara / Interview

      Teknik pengumpulan data dengan jalan mengadakan komunikasi langsung dengan pihak yang bersangkutan.

      (ex : Dinas Perhubungan, dll).

2. Studi Pustaka / Literatur

   Pengumpulan data ini dilakukan dengan cara mencari bahan-bahan kepustakaan sebagai landasan teori yang ada hubunganya dengan permasalahan yang dijadikan obyek penelitian.

1.5.2 Metode Implementasi

Dalam implementasi GIS berbasis WEB untuk menentukan lokasi dan rute jalan tujuan terdekat ini ada beberapa langkah yang harus dilakaukan antara lain :

1. Analisis

   Tahap ini melakukan pengumpulan elemen-elemen yang berkaitan dengan proses implementasi dan data yang diperlukan, meliputi: data, metode, fungsi, proses atau procedure. Hasil akhir tahap ini adalah spesifikasi kebutuhan yang diperlukan, khususnya dalam hal perangkat lunak.

2. Design

   Spesifikasi perangkat lunak yang dihasilkan dari tahap analisa ditransformasikan kedalam bentuk arsitektur perangkat lunak yang memiliki karakteristik mudah dimengerti dan tidak sulit untuk diimplementasikan.

3. Coding / Pemrograman

   Tahap ini dilakukan implementasi hasil rancangan kedalam baris-baris kode program yang dapat dimengerti oleh mesin.

4. Testing

   Pengujian dilakukan untuk setiap modul. Jika hasil pengujian tidak menemukan adanya masalah, modul-modul yang terpisah tersebut diintegrasikan untuk mendapatkan perangkat lunak yang utuh. Kemudian, dilakukan pengujian ditingkat perangkat lunak yang memfokuskan pada masalah-masalah logika internal, fungsi eksternal, potensi masalah yang mungkin terjadi dan pemeriksaan hasil.

1.6 Sistematika Penulisan

Sistematika penulisan yang diuraikan dalam penyusunan skripsi ini adalah sebagai berikut :

BAB I : PENDAHULUAN

Bab ini berisi tentang latar belakang, tujuan, permasalahan, batasan masalah, dan sistematika pembahasan dari skripsi ini.

BAB II : TINJAUAN PUSTAKA

Bab ini berisi penjelasan tentang teori – teori yang mendukung dalam implementasi system yang meliputi teori GIS, PHP, MySQL, MapServer, Flex, dan metode yang digunakan dalam pencarian rute terpendek.

BAB III : PERANCANGAN SISTEM

Bab ini berisi tentang serangkaian langkah yang ditempuh dalam pelaksanaan kegiatan penelitian ini serta langkah perancangan model sistem yang diusulkan dari awal sampai akhir, rancangan struktur database yang dipergunakan dalam aplikasi serta desain masukan dan keluaran aplikasi (*design user interface*).

BAB IV : HASIL DAN ANALISA

Bab ini berisi pembahasan hasil pengujian dan analisa mengenai cara kerja dari aplikasi.

BAB V : PENUTUP

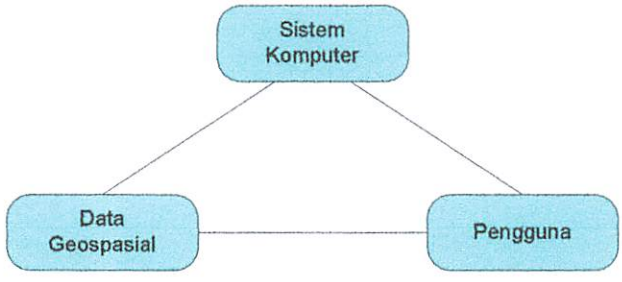
Bab ini berisi kesimpulan dan saran dari hasil pembahasan pada skripsi ini.

BAB II

DASAR TEORI

2.1 GIS (Geographic Information System)

GIS atau yang lebih dikenal dengan Geographic Information System atau Sistem Informasi Geografi (SIG) merupakan sistem yang dirancang untuk bekerja dengan data yang tereferensi secara spasial atau koordinat-koordinat geografi. GIS memiliki kemampuan untuk melakukan pengolahan data dan melakukan operasi-operasi tertentu dengan menampilkan dan menganalisa data. Pengembangan aplikasi GIS ke depannya mengarah kepada aplikasi berbasis web yang dikenal dengan web GIS. Hal ini disebabkan karena pengembangan aplikasi di lingkungan jaringan telah menunjukan potensi yang besar dalam kaitannya dengan geoinformasi. 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 masukan data, manajemen, analisis, dan representasi data. Komponen utama GIS adalah sistem komputer, data geospasial dan pengguna, seperti diperlihatkan pada Gambar 2.1



Gambar 2.1 Komponen Utama GIS

Sistem GIS terdiri dari:

1. Perangkat keras, perangkat lunak dan prosedur untuk penyusunan pemasukkan data, pengolahan, analisis, pemodelan, dan penayangan data geospatial.
2. Sumber-sumber data geospasial adalah petadigital, foto udara, citrasatelit, tabel statistik dan dokumen lain yang berhubungan. Data geospasial dibedakan menjadi data grafis (atau disebut juga data geometris) dan data atribut (data tematik). Data grafis mempunyai tiga elemen: titik (node), garis(arc) dan luasan(poligon) dalam bentuk vektor ataupun raster yang mewakili geometri topologi, ukuran, bentuk, posisi dan arah.
3. Pengguna merupakan spesifikasi pembeda dari fungsionalitas dari GIS itu sendiri. Beberapa contoh fungsionalitas GIS antara lain: Desain webGIS, updatingGIS, analisa spatial, dan lain sebagainya.

2.1.1 Data dalam GIS

Ada dua macam data dalam GIS (Geographic Information System), yaitu *data spatial* dan *data non-spatial (atribut).*

2.1.1.1 Data Spasial

Data spasial mempunyai dua bagian penting yang membuatnya berbeda dari data lain, yaitu informasi lokasi dan informasi atribut yang dapat dijelaskan sebagai berikut:

- Informasi lokasi atau informasi spasial. Contoh yang umum adalah informasi lintang dan bujur, termasuk diantaranya informasi datum dan proyeksi. Contoh lain dari informasi spasial yang bisa digunakan untuk mengidentifikasikan lokasi misalnya adalah Kode Pos.

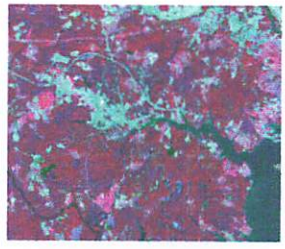
- Informasi deskriptif (atribut) atau informasi non spasial. Suatu lokalitas bisa mempunyai beberapa atribut atau properti yang berkaitan dengannya; contohnya jenis vegetasi, populasi, pendapatan pertahun, dsb.

Format data spasial

Dalam SIG, data spasial dapat direpresentasikan dalam dua format, yaitu:

a. Data Raster

Model data raster menampilkan, menempatkan, dan menyimpan data spasial dengan menggunakan struktur matriks atau pixel-pixel yang membentuk grid. Akurasi model ini sangat bergantung pada resolusi atau ukuran pixelnya. Entity spasial raster disimpan didalam layers yang secara fungsionalitas direlasikan dengan unsure-unsur petanya. Model data raster memberikan informasi spasial apa yang terjadi dimana saja dalam bentuk gambaran yang digeneralisir, seperti diperlihatkan pada Gambar 2.2

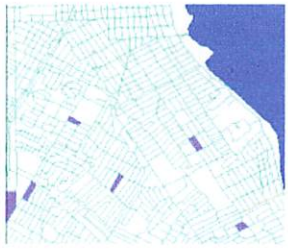


Gambar 2.2 Contoh Tampilan Data Raster

b. Data Vektor

Model data vector menampilkan, menempatkan, dan menyimpan data spasial dengan menggunakan titik-titik, garis, atau polygon beserta atribut-atributnya yang terlihat pada Gambar 2.3. Bentuk-bentuk dasar representasi data spasial ini didalam system model data vector didefinisikan oleh system koordinat kartesian dua dimensi (x, y). Pada model data vector terdapat tiga entity yaitu :

- Entity titik
- Entity garis
- Entity polygon



Gambar 2.3 Contoh Tampilan Data Vektor

2.1.1.2 Data Non-Spatial

Data non-spatial bersumber dari data sekunder dan catatan statistic atau sumber lainnya seperti hasil survey dan eksplorasi. Data non-spatial sifatnya sebagai data attribute atau data pelengkap bagi data spatial, yaitu sebagai deskripsi tambahan pada titik, garis, polygon. Data non-spatial dapat berupa table-tabel statistik, kependudukan, iklim, sumberdaya lahan, social ekonomi, kawasan politik yang dapat dikaitkan dengan luasan administrasi.

Agar data non-spatial dapat digunakan pada GIS, maka perlu diubah terlebih dahulu dengan suatu software yang mendukung Data Base Management System (DBMS). DBMS merupakan sistem yang digunakan untuk memudahkan pembuatan dan pemeliharaan basis data yang terkomputasi. Sistem ini bertujuan untuk mengolah data yang digunakan secara bersamaan dengan satu tujuan, dan terintegrasi kedalam basis data.

DBMS merupakan "*interface*" yang mengatur:

a. Bagaimana setruktur data tersebut akan disimpan dan dapat dipergunakan kembali dengan mudah, misalnya mencari kembali data (retrival data).
b. Prosedur untuk mengakses data.
c. Pembentukan file, modifikasi, penyimpanan, up-dating dan proteksi file.

Contoh perngkat lunak yang mendukung DBMS adalah: Ms Access, Dbase, SQL Server, My SQL dan lain-lain.

2.1.2 Perkembangan GIS

Seiring dengan perkembangan jaringan internet menyebabkan banyak pihak termotivasi untuk merancang dan mengembangkan aplikasi GIS tersebut hingga saai ini. Diantaranya adalah Kebutuhan Pengembang (***Produsen***) dan Kebutuhan Pengguna (***Clients***). Kebutuhan Pengenbang (Produsen) pada saat ini banyak yang memrlukan publikasi atas produk-produknya (baik barang atau jasa), apalagi yang menyangkut produk terkait informasi spasial, banyak produsen yang memerlukan sarana dan pelayanan distribusi dan yang efektif dan murah. Secara tidak langsung mereka akan memilih internet sehingga produsen dapat menghadirkan produk (terutama yang bersifat GIS) ke hadapan calon konsumennya.

Dari penjelasan diatas kita dapat melihat keuntungan dari aplikasi Web GIS dan Desktop GIS, dan dari keuntungan tersebut dapat dilihat pula perbedaannya, antara lain:

a. Pengembangan aplikasi *webbased* GIS yang didasarkan pada konsep arsitektur *web client-server* (Gambar 2.4) menjadikannya tidak mudah untuk dibandingkan secara sederhana dengan *desktop based.*

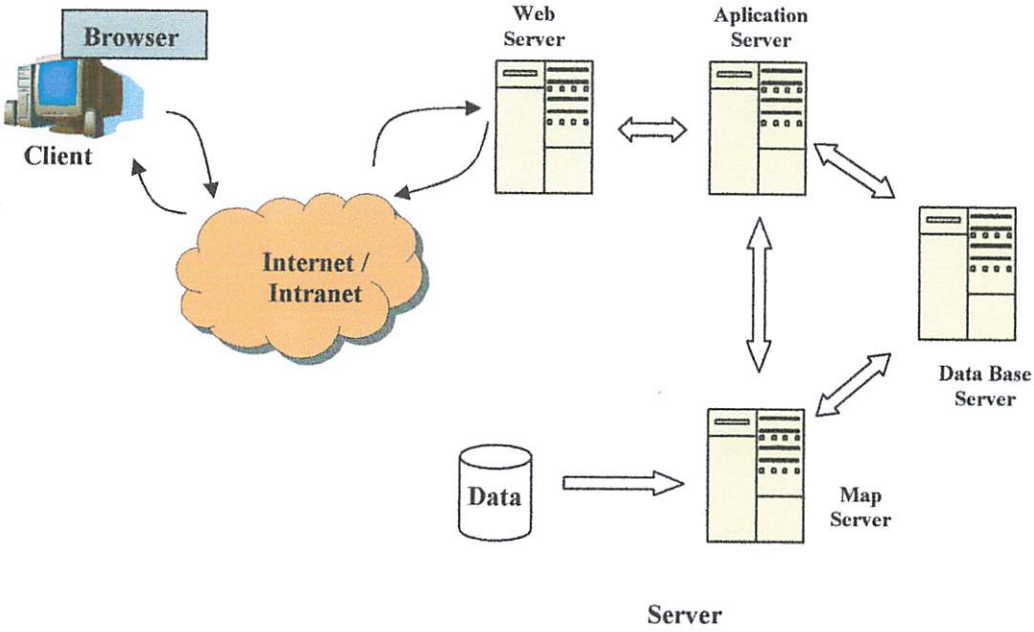
b. Kecepatan akses ke jaringan internet, kondisi existing volume lalulintas di jaringan internet terkait, dan unjuk kerja server yang bersangkutan selalu menjadi faktor kendala bagi aplikasi *webbased* GIS. Sementara *desktop based* GIS tidak mengalaminya.

c. Pengguna bebas menjalankan *query* dan analisis spasialnya (*geoprocessing*) di aplikasi *desktop based* GIS. Ia bebas menjalankan fungsi-fungsi terkait selama perangkat lunak yang bersangkutan menyediakannya. Tetapi di aplikasi *webbased* GIS, fungsionalitasnya yang sama akan sangat bergantung pada komponen *map server aplication server* [akan di bahas pada poin berikutnya]. Biasanya ukuran komponen *map server* lebih kecil dari pada *desktop based* GIS. Selain itu, tidak semua fungsionalitas yang terdapat di dalamnya selalu digunakan oleh pengembang dalam mengimplementasikan *application server*-nya. Atau dengan kata lain, fungsionalitas *webbased* GIS sangat terbatas di banding *desktop based* GIS.

d. Pada *desktop* GIS, pengguna berinteraksi langsung dengan *user interface & engine*-nya (tanpa sekat). Sementara pada *webbased* GIS, pengguna (client) tidak dapat berhubungan langsung dengan *engine*-nya. Ia terlebih dahulu harus berhubungan dengan *web server* dan *server* aplikasinya.

e. Aplikasi GIS yang berjalan di internet cenderung hanya menampilkan pete-peta digital dengan simbol-simbol & legenda berwarna, dan tabel

attribut juga menyediakan beberapa fungsi manipulasi tampilan: *zoom in, zoom out, pan*, menjalankan fungsi pencarian dan *quey* interaktif sederhana, dan menyediakan beberapa file untuk di *download*. Sementara *desktop based* GIS menyediakan semua fungsi-fungsi diatas secara penuh plus beberapa tambahan lainnya minus fasilitas *download*.



Gambar 2.4 Arsitektur aplikasi *Web Based GIS*

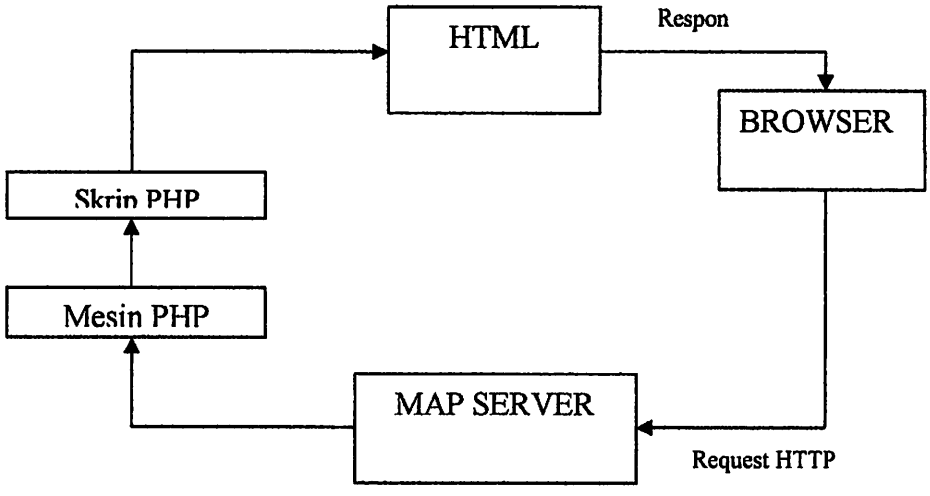
2.2 Aplication Server (PHP)

PHP adalah skrip bersifat server – side yang ditambahkan kedalam HTML. PHP sendiri merupakan singkatan dari Personal Home Page tools. Skrip ini akan membuat suatu aplikasi dapat diintegrasikan ke dalam HTML sehingga suatu halaman web tidak lagi bersifat statis, namun menjadi bersifat dinamis. Sifat server – side berarti pengerjaan skrip akan dilakukan di server, baru kemudian hasilnya dikirimkan ke browser.

Keunggulan dari sifatnya yang server – side tersebut antara lain:

- Tidak diperlukan kompatibilitas browser atau harus menggunakan browser tertentu, karena serverlah yang akan mengerjakan skrip PHP. Hasil yang dikirimkan kembali ke browser umumnya bersifat text atau gambar saja sehingga pasti dikenal oleh browser apapun.
- Dapat memanfaatkan sumber – sumber aplikasi yang dimiliki oleh server, misalnya koneksi ke database.
- Kode yang menyusun program tidak perlu diedarkan ke pemakai sehingga kerahasiaan kode dapat dilindungi (skrip tidak dapat "diintip" dengan menggunakan fasilitas view HTML source).

Prinsip kerja dari PHP seperti diperlihatkan pada Gambar 2.5



Gambar 2.5 Prinsip kerja PHP

Variabel pada PHP berfungsi untuk menyimapan suatu nilai dan nilai yang ada didalamnya dapat diubah sewaktu-waktu. Dalam membuat suatu nama variabel, nama yang dipilih harus memenuhi aturan pengenal. Pengenal (identifier) banyak digunakan dalam program untuk memberi nama variabel, fungsi atau kelas. Aturan yang berlaku untuk pengenal:

1. Karakter yang dapat digunakan yaitu huruf, angka atau garis bawah (_).
2. Karakter yang pertama harus berupa huruf atau garis bawah.
3. Panjang pengenal bisa berapa saja.
4. huruf kecil dan huruf kapital dibedakan.

Variabel pada PHP selalu diawalai dengan tanda $.

Sintaks Dasar PHP digunakan sebagai penanda yang memberitahukan parser PHP bahwa sebuah skrip menggunakan bahasa PHP dapat dilakukan dengan cara berikut :

- Menggunakan tanda <? .. ?>
- Menggunakan tanda <?php ... ?>
- Menggunakan perintah seperti pada Javascript

Contoh :

<script language="php"> ... </script>

- Menggunakan tanda <% ... %>

Dari seluruh cara tersebut dapat dilihat bahwa tanda penutup harus sama dengan tanda pembukanya. Misalnya jika tanda pembukanya (<?) maka tanda pentutpnya adalah (?>). Fungsi dari tanda penutup ini adalah untuk memberikan tanda bahwa sebuah statement atau skrip PHP telah selesai dilengkapi.

2.3 Database Server (MySQL)

MySQL merupakan software resmi yng dikembangkan oleh perusahaan Swedia bernama MySQL AB, yang waktu itu bernama TcX Data Konsult AB.Pada awalnya MySQL memakai nama mSQL atau "mini SQL" sebagai

antarmuka yang digunakan, ternyata dengan menggunakan mSQL itu mengalami banyak hambatan, yaitu sangat lambat dan tidak fleksibel. Oleh karena itu, Michael Widenius ("Monty"), panggilan akrabnya, berusaha mengembangkan interface yang tersebut hingga ditemukan MySQL. Kala itu, MySQL didistribusikan secara khusus, yakni untuk keperluan nonkomersial bersifat gratis sedangkan untuk kebutuhan komersial diharuskan membayar lisensi. Barulah sejak versi 3.23.19, MySQL dikategorikan software berlisensi GPL, yakni dapat dipakai tanpa biaya untuk kebutuhan apapun.

Hingga kini, MySQL dapat dijalankan di berbagai sistem operasi misalnya Linux, Unix, Windows. Kelebihan MySQL adalah pada kecepatan akses, biaya, konfigurasi, tersedia source code karena MySQL dibawah Open Source License dan menjadi database open source yang sangat popular. MySQL merupakan database relational (RDBMS) yang mempunyai kemampuan yang sangat cepat untuk menjalankan SQL dengan multithread dan multiuser. Oleh karena itu, dengan dengan melihat begitu tingginya potensi MySQL untuk dijadikan sebagai database yang andal, segala feature pendukung terus dikembangkan agar penggunaan MySQL dapat lebih optimal lagi.

Ada beberapa alasan mengapa MySQL menjadi database yang sangat populer dan digunakan oleh banyak orang, antara lain ialah :

- MySQL merupakan database yang memiliki kecepatan yang tinggi dalam melakukan pemrosesan data, dapat diandalkan dan mudah digunakan serta mudah dipelajari.
- MySQL mendukung banyak bahasa pemrograman seperti Java, C, C++, Phyton, dan PHP. Kita dapat menggunakan bahasa pemrograman tersebut

untuk berinteraksi maupun berkomunikasi dengan MySQL server, atau dapat juga digunakan sebagai komponen pembentuk antarmuka (interface) dari suatu database MySQL.

- Koneksi, kecepatan dan keamanan membuat MySQL sangat cocok diterapkan untuk pengaksesan database melalui internet dengan menggunakan bahasa pemrograman Perl atau PHP sebagai interfacenya.

2.4 Web Server (Apache)

Web Server adalah software server yang menjadi tulang belakang dari World Wide Web (WWW). Web server menunggu permintaan dari client yang menggunakan browser seperti netscape navigator, Internet Explorer, modzilla, dan program browser lainnya. Jika ada permintaan dari browser, maka web server akan memproses permintaan itu dan kemudian memberikan hasil prosesnya berupa data yang diinginkan kembali ke browser.

APACHE adalah web server standar yang dipaket oleh berbagai macam distribusi Linux. Apache dibuat berbasiskan kode sumber dan ide-de yang ada pada web server leluhurnya yaitu web server NCSA. NCSA dibuat oleh National Center for Supercomputing Applications dan dibiayai oleh pemerintah Amerika. Tidak seperti proyek leluhurnya, Apache adalah software berlisensi GPL (General Public License) atau free software.

APACHE memiliki beberapa kelebihan di banding Web server yang lainnya, antara lain:

a. Apache termasuk dalam kategori freeware (software gratisan).

b. Apache mudah sekali proses instalasinya jika dibanding web server lainnya seperti NCSA, IIS, dan lain-lain.

c. Mampu beroperasi pada berbagai platform sistem operasi seperti : AUX 3.1, BSDI 2.0, FreeBSD 2.1, HP-UX 9.07, IRIX 5.3, Linux, NetBSD 1.1, NEXTSTEP, SolarisX86 2.5, Solaris 2.4, Solaris 2.5, SunOS 4.1.3, UnixWare 1.1.2.

d. Apache mudah untuk mengkonfigurasinya karena hanya mempunyai satu file konfigurasi.

e. Mudah dalam menambahkan periferal lainnya ke dalam platform web servernya, misalnya : untuk menambahkan modul, cukup hanya menset file konfigurasinya agar mengikutsertakan modul itu kedalam kumpulan modul lain yang sudah dioperasikan.

2.5 Map Server

MapServer merupakan aplikasi freeware dan open source yang memungkinkan kita menampilkan data spasial (peta) di web. Saat ini, karena sifatnya yang terbuka (open source), pengembangan MapServer dilakukan oleh pengembang dari berbagai negara. Pengembangan MapServer menggunakan berbagai aplikasi open source atau freeware seperti Shapelib (http://shapelib.maptools.org) untuk baca/tulis format data Shapefile, FreeType (http://www.freetype.org) untuk merender karakter, GDAL/OGR (http://www.remotesensing.org/gdal) untuk baca/tulis berbagai format data vektor maupun raster, dan Proj.4 (http://www.remotesensing.org/proj) untuk menangani beragam proyeksi peta.

Pada bentuk paling dasar, MapServer berupa sebuah program CGI (Common Gateway Interface). Program tersebut akan dieksekusi di web server, dan berdasarkan beberapa parameter tertentu (terutama konfigurasi dalam bentuk file *.MAP) akan menghasilkan data yang kemudian akan dikirim ke web browser, baik dalam bentuk gambar peta maupun bentuk lain.. Selain itu MapServer juga sangat unggul didalam me-render data spasial (citra, data vektor dan peta digital lainnya) untuk aplikasi web seperti pada Gambar 2.6

Gambar 2.6 Diagram Kerja MapServer

MapServer mempunyai fitur-fitur berikut :

- Menampilkan data spasial dalam format vektor seperti Shapefile (ESRI), ArcSDE (ESRI), PostGIS dan berbagai format data vektor lain dengan menggunakan library OGR
- Menampilkan data spasial dalam format raster seperti : TIFF/GeoTIFF, EPPL7 dan berbagai format data raster lain dengan menggunakan library GDAL
- Menggunakan quadtree dalam indexing data spasial, sehingga operasi-operasi spasial dapat dilakukan dengan cepat

- Dapat dikembangkan (customizable), dangan tampilan keluaran yang dapat diatur menggunakan file-file template
- Dapat melakukan seleksi objek berdasar nilai, titik, area, atau berdasar sebuah objek spasial tertentu
- Mendukung rendering karakter berupa font TrueType
- 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)
- Dapat menggambarkan elemen peta secara secara otomatis : skala grafis, peta indeks, dan legenda peta
- Dapat menggambarkan peta tematik yang dibangun menggunakan ekspresi logik maupun ekspresi reguler

2.6 Flex

Flex, atau lebih dikenal sebagai Adobe Flex merupakan suatu teknologi yang membawa perubahan paradigma dalam pengembangan aplikasi /perangkat lunak, khususnya aplikasi web. Flex sendiri merupakan gabungan dari teknologi yang telah dimiliki Adobe sebelumnya, dengan bertumpu pada platform Adobe Flash. Flex juga merupakan RIA (Rich Internet Application), yaitu sebuah aplikasi web yang dapat memiliki fitur dan menerapkan fungsi-fungsi selayaknya aplikasi berbasis desktop.

Dikarenakan bertumpu pada platform Flash, aplikasi web yang Flex-based tidak melakukan proses HTML Rendering ketika mempresentasikan dirinya melalui browser. Selain itu juga, Flex dikembangkan dengan konsep n-tier , yaitu

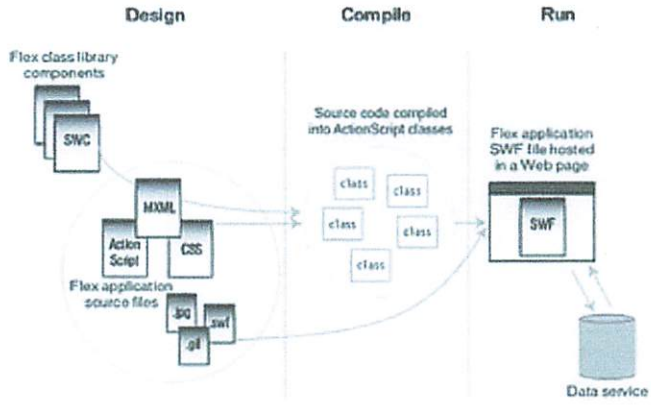
pemisahan layer antara data access, presentation, business process dikarenakan flash tidak dapat melakukan *direct connection* ke data source semisal DBMS ataupun data source dalam bentuk lain.

Untuk memulai pengembangan aplikasi Flex, Adobe telah menyediakan Flex SDK (Software Development Kit) sebagai koleksi / pustaka kelas yang dapat dipakai baik melalui IDE ataupun secara traditional. IDE yang telah disediakan khusus oleh Adobe adalah Adobe Flex Builder (hingga saat ini, sudah memasuki versi 4), yang memungkinkan developer membuat aplikasi Flex secara visual, layaknya pengembangan aplikasi desktop dengan IDE nya masing-masing. Dahulunya, beberapa developer web (termasuk saya),sering mengalami kesulitan ketika menyediakan content berbasis flash, dikarenakan mekanisme yang dilalui tidak sesingkat pembuatan content HTML biasa. Namun kini dengan Adobe Flex Builder, masa-masa sulit tersebut dapat dilalui dengan usaha selayaknya pengembangan aplikasi web secara tradisional.

Keunggulan Flex :

1. Kemudahan dalam melakukan layouting aplikasi dengan fitur drag n drop komponen, tersedianya panel properties dan event untuk masing-masing komponen, serta pengaturan posisi komponen dan tampilan aplikasi secara keseluruhan melalui fitur constraint.
2. Kemudahan koneksi ke sumber data dengan adanya fungsi yang terintegrasi untuk menyediakan RPC Service.
3. Konsep multi layer /page pada presentasi dengan menggunakan View States.

4. Kemudahan dalam membuat animasi transisi dan effect pada tiap-tiap komponen dan layer yang digunakan dalam pengembangan aplikasi.
5. Compiler dan debugger yang terintegrasi, dengan beberapa opsi pilihan



Gambar 2.7 Bagaimana program Flex di compile dan dikembangkan

Pada Gambar 2.7, dijelaskan bahwa Kelas HTTPService memperbolehkan kita untuk merequest dari URL dan menerima respon. Penggunaan HTTPService untuk memanggil file PHP dan menghasil file XML dapat digambarkan sebagai berikut. Pertama MXML dikompile menjadi SWF dan publish diweb, kemudian swf membuat request dengan HTTPService melalui Apache, apache meterjemahkan request tadi dan meminta file PHP untuk menggenerate file XML, bila file XML sudah terbentuk maka apache merespon request tersebut dengan memberikan data XML ke SWF.

2.7 Pencarian Rute Terpendek

Pencarian Rute Terpendek dapat menggunakan 2 metode, yaitu metode algoritma konvensional dan metode heuristik. Metode algoritma konvensional diterapkan dengan cara perhitungan matematis seperti biasa, sedangkan metode

heuristik diterapkan dengan perhitungan kecerdasan buatan, dengan menentukan basis pengetahuan dan perhitungannya.

a. Metode konvensional

Metode konvensional berupa algoritma yang menggunakan perhitungan matematis biasa. Metode konvensional yang biasa digunakan untuk melakukan pencarian jalur terpendek, diantaranya algoritma Djikstraa, algoritma Floyd-Warshall, dan algoritma Bellman-Ford

b. Metode heuristik

Adalah sub bidang dari kecerdasan buatan yang digunakan untuk melakukan pencarian dan penentuan jalur terpendek. Ada beberapa algoritma pada metode heuristik yang biasa digunakan dalam pencarian jalur terpendek. Namun dalam penelitian yang biasa dilakukan para peneliti dibatasi hanya membahas dua macam algoritma yaitu algoritma semut dan algoritma genetika.

2.7.1 Algortima A*

Algoritma A* pertama kali diperkenalkan pada tahun 1968 oleh Peter Hart, Nils Nilsson dan Bertram Raphael. Dalam tulisan mereka, algoritma ini disebut sebagai algoritma A.

Algoritma A merupakan salah satu algoritma pencarian graf, yaitu menemukan jalur terpendek dari titik awal sampai ke titik tujuan dalam suatu graf. Algoritma ini menggunakan perkiraan heuristic h(x) yang menyusun tiap titik x berdasarkan perkiraan rute terbaik yang melalui titik tersebut. Kemudian titik yang dikunjungi adalah titik dengan perkiraan terbaik tersebut.

Algoritma A bekerja dengan menyimpan sebuah antrian prioritas dari garis-garis yang melalui graf tersebut dimulai dari titik awal. Misalkan dari titik awal (start node) ke titik A terdapat sebuah garis, prioritas pada garis x berdasarkan Persamaan 2.1

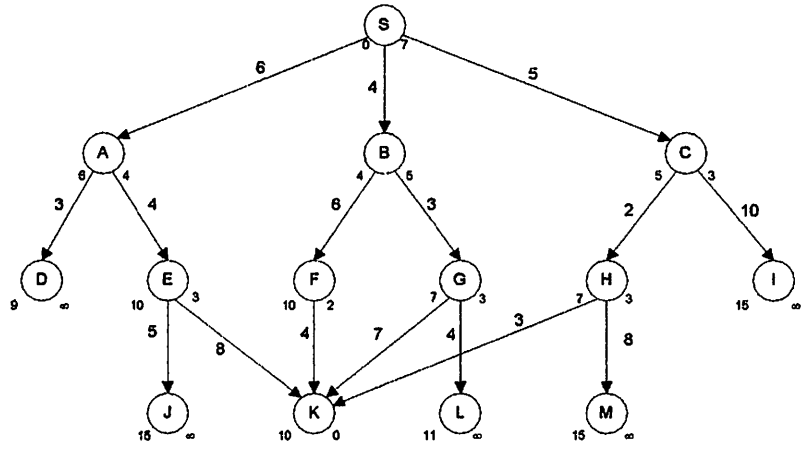
$$f(x) = g(x) + h(x) \qquad \text{.....} \quad (2.1)$$

Pada Persamaan 2.1, g(x) merupakan biaya yang dimiliki oleh garis yang menghubungkan dari titik awal menuju ke titik x. Sedangkan h(x) adalah biaya minimum yang sebenarnya untuk mencapai titik tujuan dari titik x. Nilai f(x) yang paling rendah mendapat prioritas paling tinggi dalam antrian tersebut.

Algoritma A ini kemudian dikembangkan sehingga menjadi algoritma A*. Jika dalam algoritma A digunakan nilai h(x) maka dalam algoritma A* digunakan nilai h'(x) yang merupakan perkiraan biaya untuk sampai pada suatu tujuan dari titik x. Dengan syarat $h'(n) \leq h(x)$. Untuk evaluasinya algoritma ini menggunakan fungsi seperti yang terlihat pada Persamaan 2.2

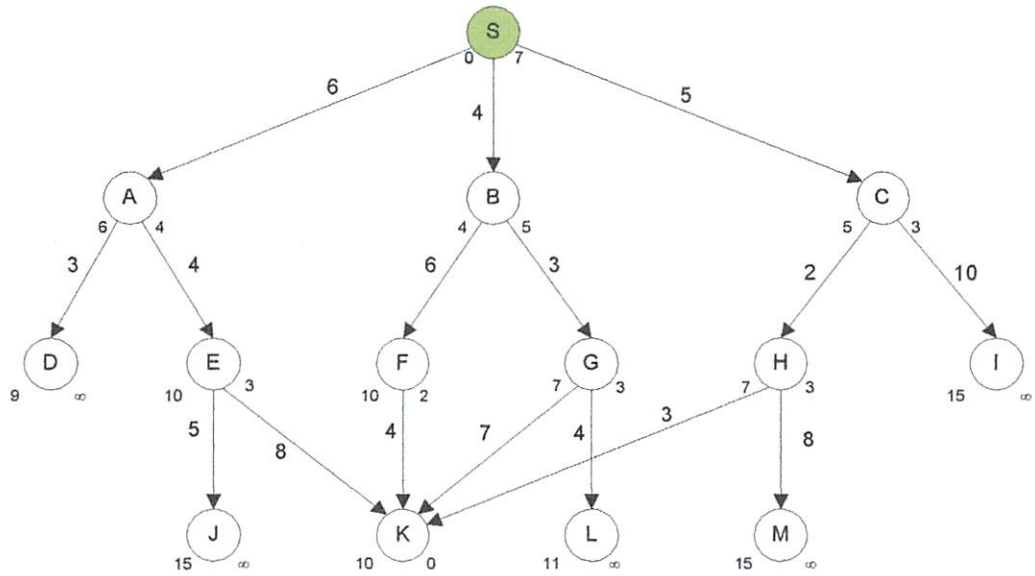
$$f'(x) = g(x) + h'(x) \qquad \text{.....} \quad (2.2)$$

Untuk lebih jelasnya, berikut ini adalah contoh penggunaan algoritma A* untuk graf seperti yang terlihat pada Gambar 2.8



Gambar 2.8 Graf Contoh Pencarian Rute dengan Algoritma A*

Node S merupakan start node sedangkan node K merupakan goal node. Nilai edge dari S ke A merupakan biaya yang dibutuhkan dari node S menuju ke node A. Nilai g(x) diperoleh dari biaya edge minimal, sedangkan nilai h'(x) diperoleh dari perkiraan biaya yang dibutuhkan dari node tersebut menuju goal node (nilai ini tidak menjamin keberadaan solusi, hanya perkiraan saja). h'(x) bernilai ∞ jika sudah jelas tidak ada hubungan antara node x dengan goal node (jalan buntu). Misalkan untuk node S memiliki nilai g(x) = 0 dan h'(x) = 7, node A memiliki nilai g(x) = 6 dan h'(x) = 4. Mula-mula start node(S) dimasukkan dalam closed list kemudian dicari anak-anaknya yaitu A, B dan C. Karena ketiga node tersebut belum pernah masuk dalam open list maupun closed list maka ketiganya langsung dimasukkan dalam open list dan masing-masing diinisialisasi parentnya yaitu S. Antrian dalam open list berdasarkan nilai f'(x) dari kiri ke kanan adalah dari kecil ke besar. Warna hijau melambangkan bahwa node tersebut telah diuji atau telah dimasukkan dalam closed list. Seperti terlihat dalam Gambar 2.9 dan Tabel 2.1

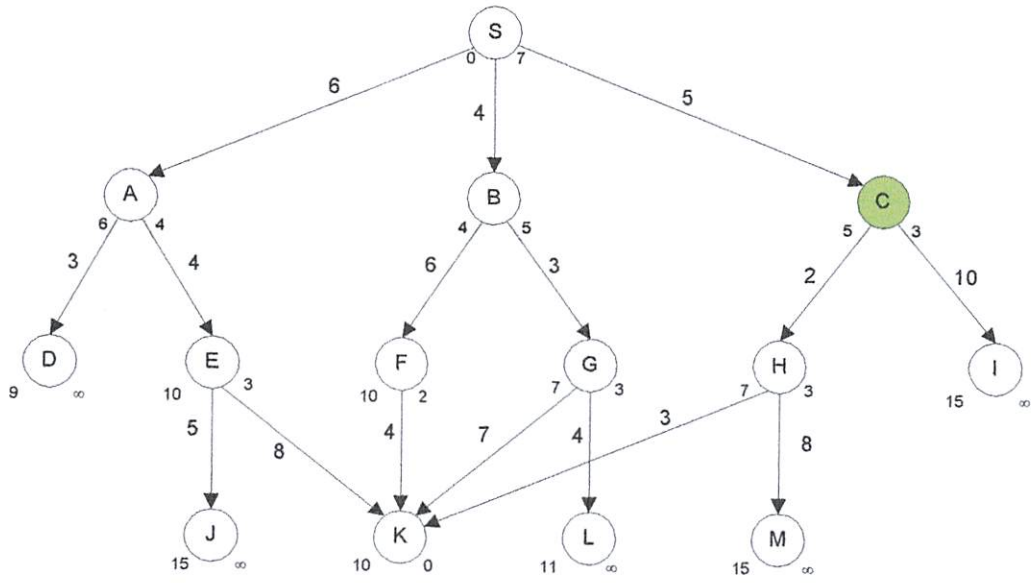


Gambar 2.9 Langkah ke-1 Pencarian Rute dengan Algoritma A*

Tabel 2.1 *Open List* dan *Closed List* pada Langkah-1

Open List	*Closed List*
{C(8), B(9), A(10)}	{S}

Selanjutnya dari antrian dalam open list diambil node dengan nilai f'(x) terkecil yaitu node C. Node tersebut kemudian dimasukkan dalam closed list. Lalu dilakukan pengecekan pada anak-anak C yaitu H dan I. Karena keduanya belum pernah masuk dalam closed list maupun open list maka langsung dimasukkan dalam open list dan node C menjadi parent H dan I. Seperti terlihat dalam Gambar 2.10 dan Tabel 2.2

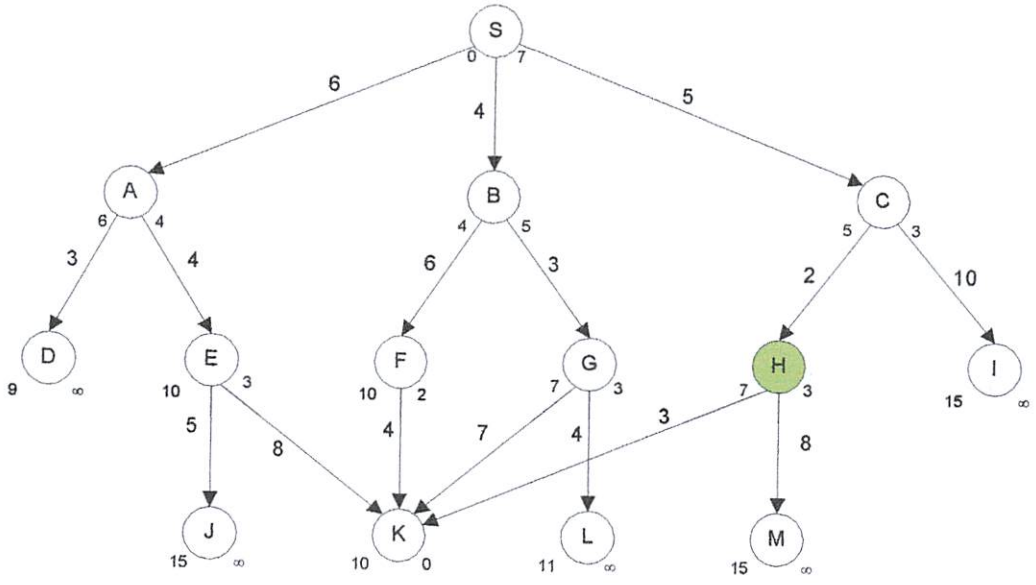


Gambar 2.10 Langkah ke-2 Pencarian Rute dengan A*

Tabel 2.2 Open List dan Closed List pada Langkah ke-2

Open List	Closed List
{H(9), B(9),A(10), I(∞)}	{C, S}

Dari antrian yang terdapat dalam open list maka diambil node dengan nilai f'(x) paling kecil yaitu H. Selanjutnya H dimasukkan ke dalam closed list. Kemudian dilakukan pengecekan terhadap anak dari node H dan diperoleh M serta K. Karena keduanya belum pernah masuk dalam closed list maupun open list maka keduanya langsung dimasukkan dalam open list. Seperti yang terlihat pada Gambar 2.11 dan Tabel 2.3.



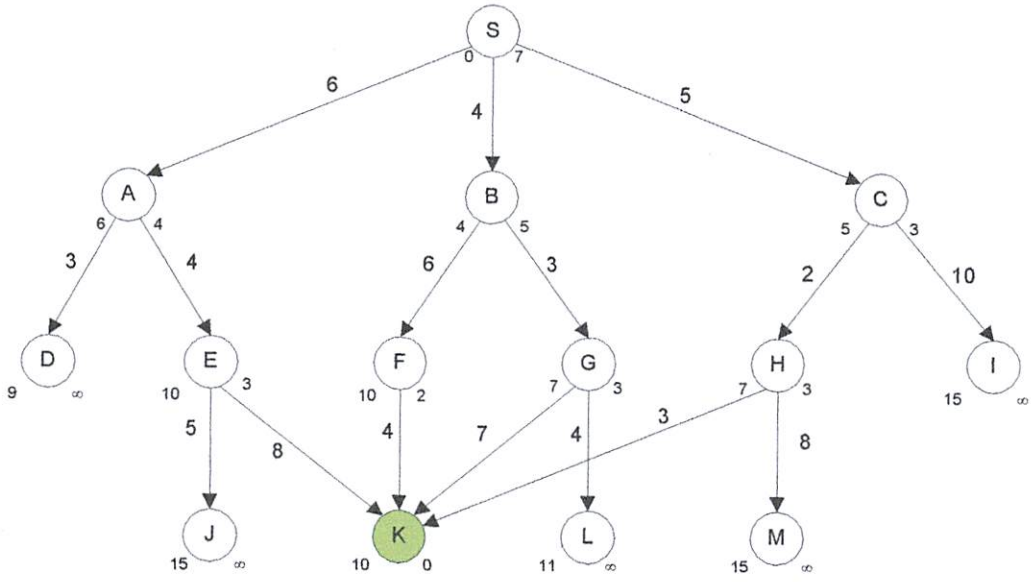
Gambar 2.11 Langkah ke-3 Pencarian Rute dengan A*

Tabel 2.3 Open List dan Closed List pada langkah ke-3

Open List	Closed List
{K(9), B(9), A(10), M(∞)}	{H, C, S}

Kemudian dilakukan pengecekan lagi dari antrian yang ada dalam open list dan diperoleh node dengan nilai f'(x) paling kecil yaitu K. Kemudian K dimasukkan ke dalam closed list seperti diperlihatkan dalam Gambar 2.12 dan Tabel 2.4.

Karena K merupakan goal node maka proses berhenti. Dan untuk mendapatkan jalurnya bisa dilakukan backward. Yaitu proses untuk mencari jalur dari goal node sampai ke start node dengan melihat parentnya. Pada kasus ini proses backward dimulai dari node K. Node K mempunyai parent H, node H mempunyai parent C, node C mempunyai parent S. Karena M merupakan start node maka proses backward berhenti. Dan diperoleh jalurnya adalah S-C-H-K. Seperti terlihat pada Gambar 2.12 dan Tabel 2.4



Gambar 2.12 Langkah ke-4 Pencarian Rute dengan A*

Tabel 2.4 Antrian Open dan Closed pada langkah ke-4

Antrian Open	Antrian Closed
{B(9), A(10), I(∞), M(∞)}	{K, H, C, S}

BAB III

PERANCANGAN SISTEM

Pada bab ini akan dibahas metode, rancangan yang digunakan dan langkah-langkah yang dilakukan dalam penelitian penentuan rute. Penelitian dilakukan dengan tahapan-tahapan sebagai berikut :

1. Mempelajari algoritma yang digunakan pada sistem penentuan rute terpendek, yaitu algoritma A*.
2. Menganalisa dan merancang perangkat lunak penentuan rute yang akan dilalui dengan analisis pencarian rute terpendek berdasarkan jarak lintasan yang mempertimbangkan titik-titik kemacetan.
3. Pada analisa hasil juga akan dilakukan uji coba dengan membandingkan analisa yang didapat pada point 2, yang ditujukan untuk mengetahui apakah dengan analisa pencarian lintasan tependek yang dilakukan dapat menghasilkan jalur terpendek, apabila mempertimbangkan variabel kemacetan.
4. Uji coba perangkat lunak yang telah dibuat dan menjalankannya di browser.

3.1 Deskripsi Umum Sistem

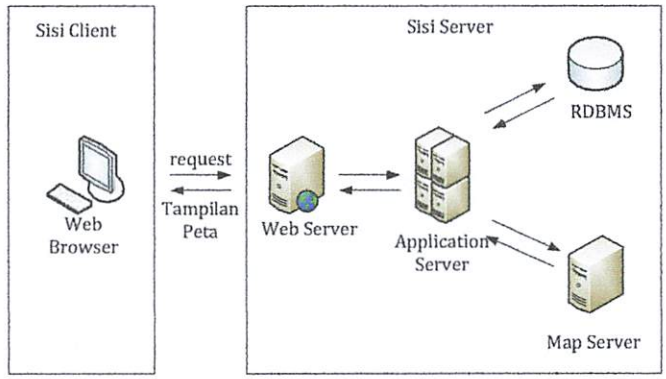
Sistem yang akan dibuat merupakan penentuan rute jalan terpendek yang dapat mempermudah pengendara kendaraan melakukan perjalanan ke berbagai tempat di Kota Malang dengan cara mengumpulkan informasi, pencarian data yang dilakukan dengan cara merancang database dan membuat sistem. Dengan interface yang dapat mempermudah user, user juga dapat memperoleh informasi

geografis mengenai objek-objek/tempat-tempat tersebut melalui peta yang telah disediakan.

Dalam pembuatan aplikasi ini, informasi yang telah dikumpulkan berupa kondisi jalan dan lokasi / tempat-tempat di Kota Malang. Metode yang digunakan pada sistem ini adalah *Algoritma A-star* yang merupakan salah satu algoritma terbaik dalam penentuan pencarian lintasan terpendek dari simpul tertentu ke semua simpul yang lain (*single-source shortest path*). Dengan metode *single-source shortest path,* maka akan didapat jarak terdekat ke semua node / simpul. Sehingga dapat dibandingkan untuk mendapatkan jarak yang paling dekat / terdekat. Algoritma ini lebih efisiensi di banding algortima konvensional lainnya (Dijkstra, Floyd-Warshall, Bellman-Ford).

Algoritma A* juga lebih menguntungkan dari sisi efisiensi waktu. Algoritma A* dapat memfokuskan pencarian pada node-node yang berada pada arah yang mendekati node tujuan. Kemudian pencarian dapat dimaksimalkan pada waktu node tujuan diperiksa. Hal ini dapat meminimalisasikan jumlah node yang harus diperiksa.

Output dari sistem, berupa informasi visual peta rute jalan dan jarak terdekat, data geografis jalan (nama jalan, panjang jalan, foto dll), dan data lokasi (alamat, foto, no.telp, dll) yang diperlukan. Ada beberapa pertimbangan didalam pencarian rute terdekat, yaitu kemacetan ruas jalan dan arus jalan (1 atau 2 arah). Jika dalam pencarian rute terdapat ruas kemacetan yang dilewati, maka secara otomatis sistem akan memberikan rute alternatif untuk menghindari ruas kemacetan tersebut.



Gambar 3.1 Arsitektur Sistem

Seperti pada Gambar 3.1, didalam implementasi sistem sesungguhnya terdapat tiga komponen utama yaitu: *server* yang memproses request dari client dan mengirimkan kembali hasil request (respon) ke client, *internet* sebagai jaringan global yang menghubungkan client ke server dan *client* yang melakukan request ke server. Aplikasi yang di bangun akan di pasang pada Server, sehingga user / client hanya tinggal mengakses ke server untuk menjalankannya.

Permintaan user atas informasi rute terdekat pertama kali akan di terima oleh HTTP Server. Kemudian akan dilakukan pengecekan apakah ada script PHP yang perlu diproses atau dieksekusi. Script PHP akan diproses dan dieksekusi oleh PHP + MapScript dengan dukungan Map files, HTML Template dan MySQL sebagai data external melalui API yang disediakan, dan hasilnya akan dikirim kembali ke client oleh HTTP Server yang sudah berupa HTTP respon.

Di dalam server terdapat bagian utama yang berperan penting dalam kinerja aplikasi yang dibangun. Bagian-bagian tersebut saling terkait, yaitu:

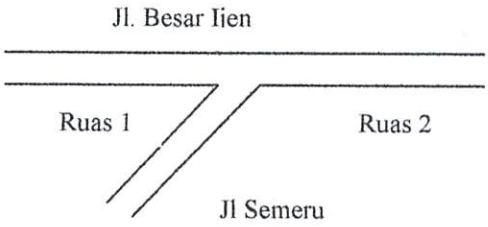
- Web Server : Yang menunggu request dari client dan mengirimkan respon ke admin setelah request di proses.

- MapServer (PHP MapScript) : Yang melakukan fungsional GIS (mapping tools).
- Map File : Merupakan file GIS yang digunakan untuk menajamen peta, standart format yang digunakan adalah .shp.
- PHP : Sebagai script yang mengeksekusi PHP Script.
- Map Data : Menyangkut data-data geografis jalan dan data pendukung lainnya (Data Attribut).
- External Data : Merupakan data pendukung yang tersimpan di Database MySQL

Di dalam penentuan rute dan urutan rute jalan yang dibuat berdasarkan pada analisis yang dilakukan terdapat 2 aspek, yaitu:

1. Berdasarkan graf berbobot yang bobotnya merupakan panjang lintasan (jarak).
2. Berdasarkan graf berbobot yang bobotnya merupakan titik ruas kemacetan .

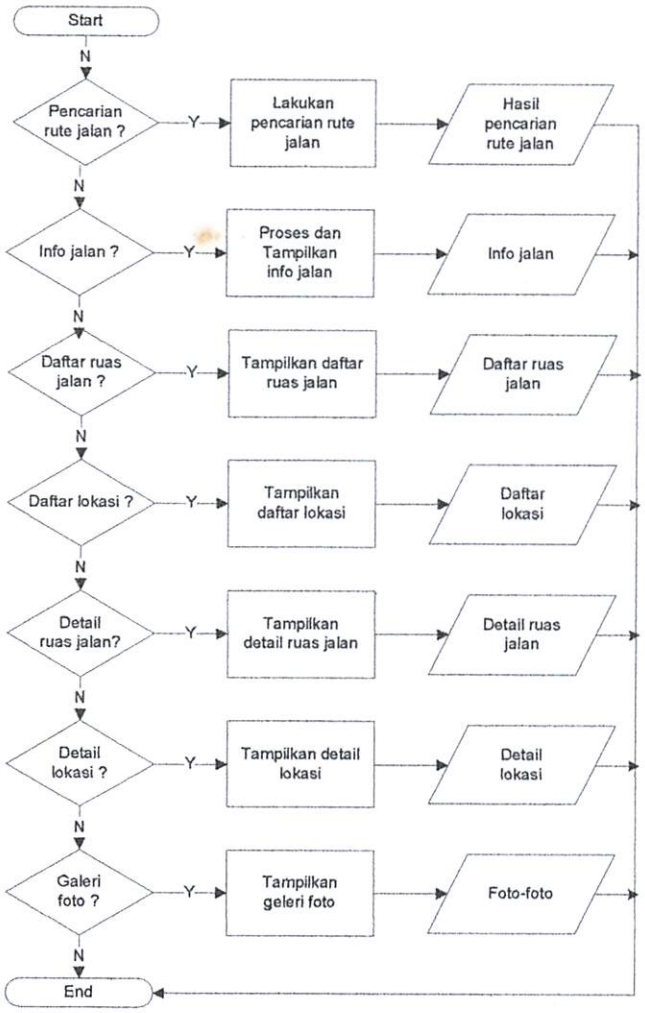
Setiap ruas jalan dan lokasi memiliki ID yang berbeda. Pemberian ID jalan dilakukan pada setiap ruas, dan pemisahan ruas berdasarkan percabangan jalan (pertigaan, perempatan, perlimaan dll). Hal ini dilakukan untuk memudahkan user dalam mencari ruas jalan yang ada., sebagai contoh "Jalan Besar Ijen", jalan ini terdiri dari dua ruas yang dipisahkan oleh pertigaan antara "Jalan Besar Ijen" dan "Jalan Semeru". Seperti terlihat pada Gambar 3.2



Gambar 3.2 Pemisahan ruas jalan

3.2 Alur Proses Aplikasi

Untuk lebih memperjelas alur dari sistem, maka alur proses aplikasi akan digambarkan pada flowchart Gambar 3.3



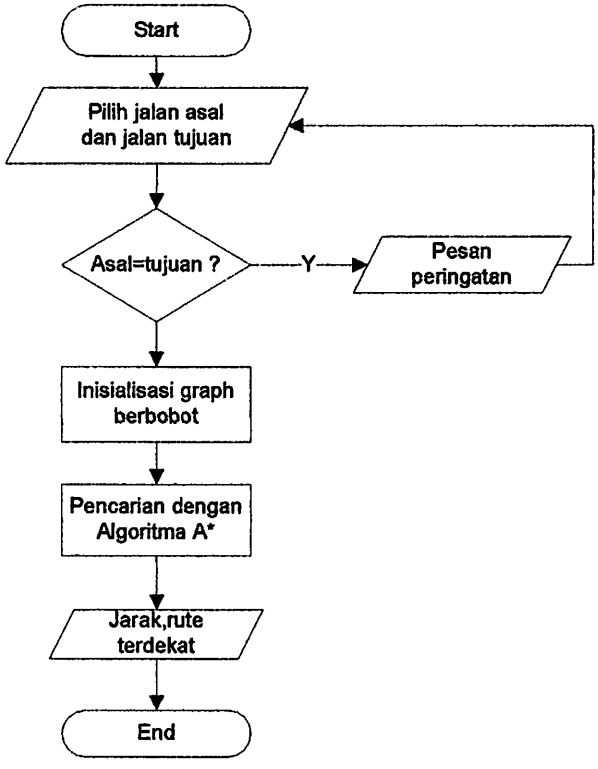
Gambar 3.3 Flowchart Alur Aplikasi

Flowchart alur proses tersebut menjelaskan rancangan urutan proses yang terjadi pada aplikasi. Dimulai dari proses pemasukan input, pencarian rute terdekat dengan algortima A* maupun proses lain, hingga output. Dimana inputan pertama berupa ruas jalan asal dan ruas jalan tujuan. ID jalan dan ID ruas jalan dibedakan, dimana ID ruas jalan diawali dengan huruf atau character "n" sedangkan ID lokasi diawali dengan huruf atau character "L". Hal ini bertujuan untuk memisahkan antara ruas jalan dan lokasi.

Di dalam aplikasi terdapat kotak saran yang dapat disi oleh setiap pengunjung yang digunakan untuk menampung seluruh kritik, saran, dan masukan dari pengunjung yang nantinya akan dibaca oleh *Administrator*.

Seorang admin memiliki hak akses untuk melakukan *maintenance* (create, update, delete) data. Untuk dapat masuk ke dalam halaman admin, seorang user harus login terlebih dahulu sebagai admin. Hal ini bertujuan untuk menjaga keamanan data yang ada di dalam database.

3.2.1 Penentuan Rute Jalan



Gambar 3.4 Flowchart pencarian rute jalan terdekat

Penjelasan flowchart pencarian rute jalan terdekat dari Gambar 3.4 adalah sebagai berikut:

a. User menentukan jalan asal dan jalan tujuan sebagai inputan dalam pencarian dengan menggunakan algortima A^*. Ruas jalan asal dan tujuan tidak boleh sama, jika sama akan muncul pesan error.

b. Graph berbobot di sini dipresentasikan sebagai struktur data bertype array 2 dimensi (Matrik ketetanggaan) yang menggambarkan hubungan node-node yang saling berkaitan yang nantinya digunakan sebagai inputan pencarian dengan algortima A^*. Dalam permasalahan kali ini bobot dari graph adalah jarak antara node-node yang saling berhubungan Semua data graph berbobot

(node, bobot / jarak antar node) ini didapat dari data yang sudah tersimpan di dalam database.

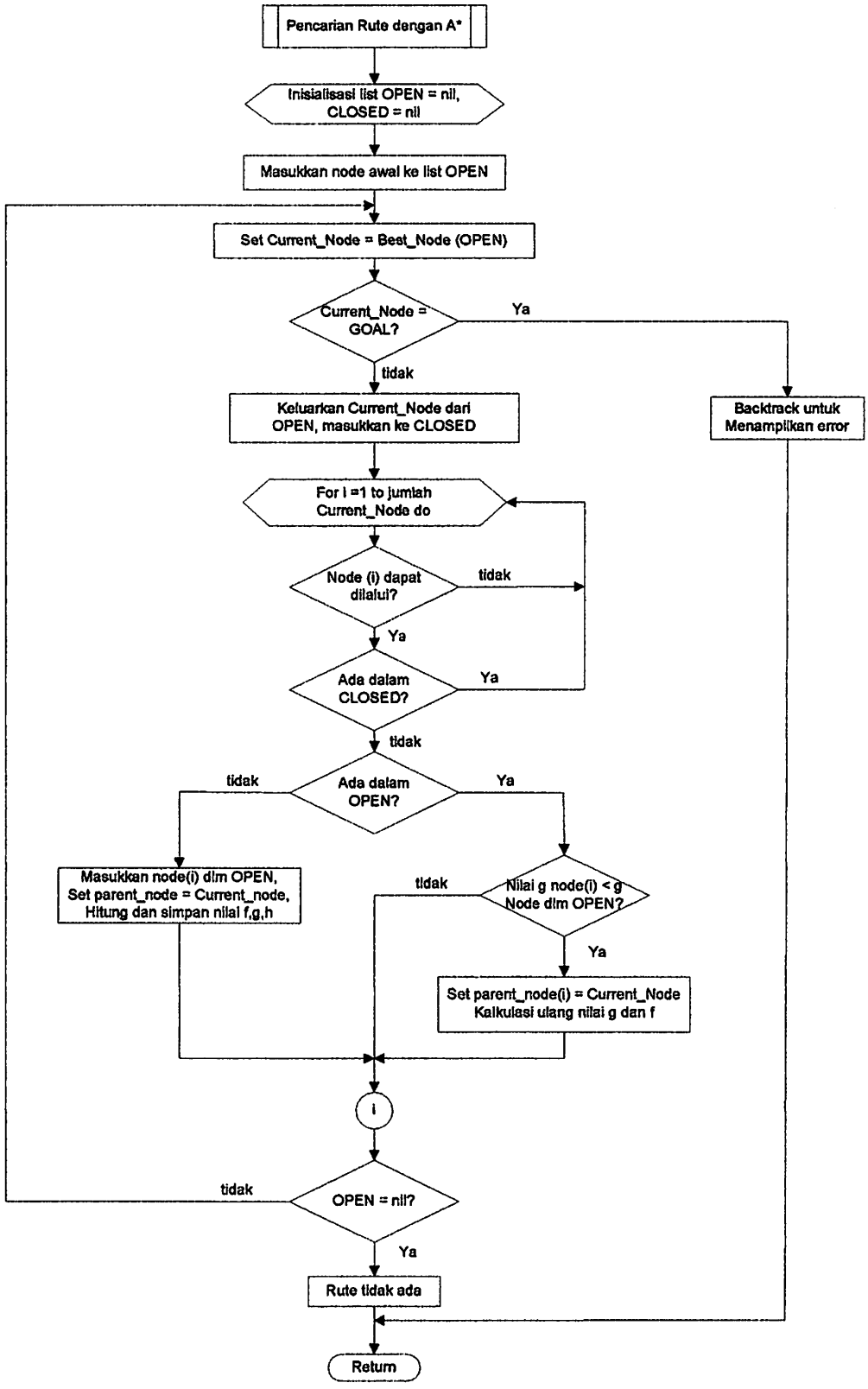
c. Melakuan pencarian dengan algoritma A*. Inputan dari pencarian dengan algortima A* adalah graph berbobot, node asal, node tujuan. Dan outputnya atau hasil pencarian adalah jarak dan rute jalan terdekat.

d. Data ruas yang macet akan digunakan sebagai pertimbangan dalam menentukan rute alternatif. Jika didalam hasil pencarian rute terdekat terdapat ruas macet, maka program otomatis akan memberikan rute solusi untuk menghindari kemacetan. Node-node yang terdapat pada ruas kemacetan akan dimasukkan ke dalam graph berbobot.

e. Hasil pencarian akan ditampilkan pada peta.

A* akan memulai pencarian dengan looping pada semua node/vertex yang terhubung dengan node di posisi sekarang (berawal dari posisi asal). Kemudian cari bobot/jarak tekecil dari node-node yang terhubung, dimana node yang terhubung belum pernah dikunjungi atau bobot yang baru lebih kecil dari bobot yang lama. kemudian menyimpan jarak/bobot dan rute pada array. Setiap node/vertex yang telah dipilih akan disimpan guna menghindari *back step* (kembali ke node sebelumnya). Dilakukan berulang-ulang hingga semua node yang ada telah dipilih/dikunjungi semua.

Pada kasus kali ini pencarian akan dilakukan ke semua pasang node/simpul/vertex (*all pairs shortest path*), jadi semua node akan diuji sehingga benar-benar akan mendapatkan rute dan jarak terpendek. Seperti terlihat pada Gambar 3.5



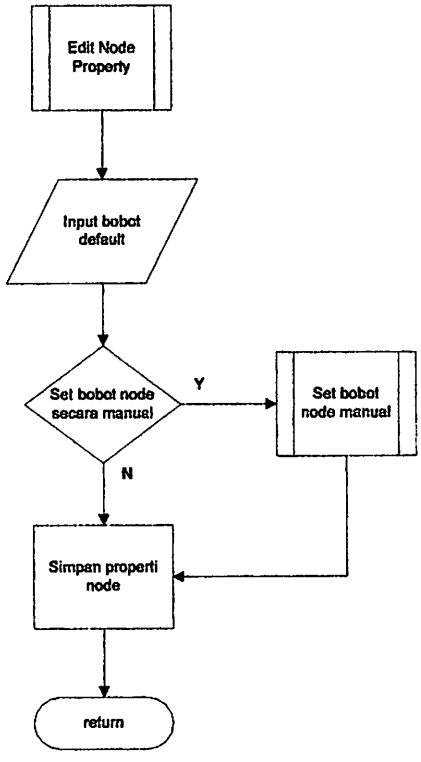
Gambar 3.5 Flowchart Algoritma A*

3.2.2 Pengaturan Bobot Node Peta

Pengaturan bobot node-node pada peta dapat dikatakan adalah salah satu hal paling penting dalam aplikasi ini. Pemberian bobot ini sangat berpengaruh pada ketepatan pencarian rute optimal pada peta yang bersangkutan. Pada pengaturan bobot node peta ini, pengguna dapat mengatur bobot default node peta. Untuk jalan yang dapat dilalui pada peta, sebaiknya bobotnya diset kecil. Sedangkan untuk daerah yang tidak dapat dilewati, bobot diset besar.

Selanjutnya pengguna dapat mengatur pemberian bobot dan pengaturan properti node lainnya kepada node-node yang dipilihnya secara manual. Pengaturan properti node lain meliputi pemberian nama jalan, kondisi jalan dan arah jalan.

Flowchart untuk pengaturan bobot dan properti lain node pada peta dapat dilihat pada Gambar 3.6



Gambar 3.6 Pengaturan bobot node pada peta

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

Implementasi merupakan proses transformasi representasi rancangan ke bahasa pemrograman yang dapat dimengerti oleh komputer. Pada bab ini akan dibahas hal-hal yang berkaitan dengan implementasi sistem penentuan rute terpendek dengan algoritma A-star.

4.1 Implementasi

Implementasi yang akan dipaparkan disini meliputi perangkat keras dan perangkat lunak.

4.1.1 Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan penentuan rute dan urutan perjalanan ini adalah sebagai berikut :

1. Prosesor Intel Core2 Duo 1.66 Ghz
2. RAM 2038 MB
3. Harddisk dengan kapasitas 185 GB
4. Monitor 12"
5. Keyboard
6. Mouse

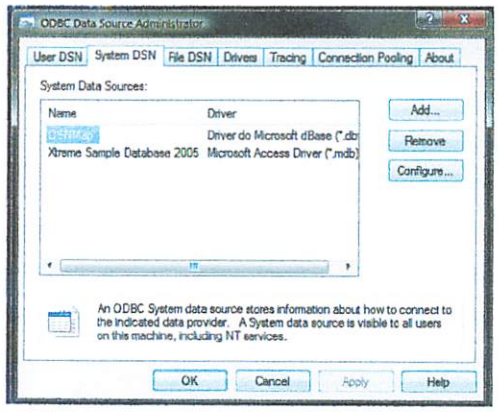
4.1.2 Perangkat Lunak

Perangkat lunak yang digunakan dalam pengembangan penentuan rute dan urutan perjalanan ini adalah :

1. Sistem Operasi Windows Vista
2. Macromedia Dreamweaver 8
3. NuSphere PhpED
4. Macromedia Flash MX
5. ArcView GIS 3.3
6. Apache 2.0.55
7. PHP 4.4.0
8. Map Server 4.6.1
9. MySQL Server
10. Adobe Photoshop
11. SQLyog Enterprise

4.2 Implementasi Database

Langkah awal yang harus dilakukan untuk mengatur konfigurasi koneksi database adalah membuat DSN (*Data Source Name*) pada ODBC yang diberi nama "DSNMap". Hal ini bertujuan untuk mengkoneksikan data pada *attribut* dBase (.dbf). Seperti terlihat pada Gambar 4.1



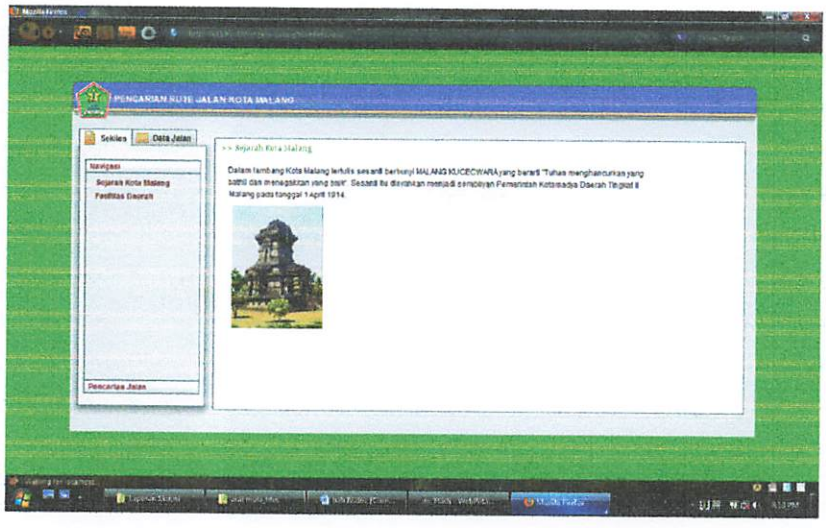
Gambar 4.1 Koneksi dengan ODBC

4.3 Antarmuka

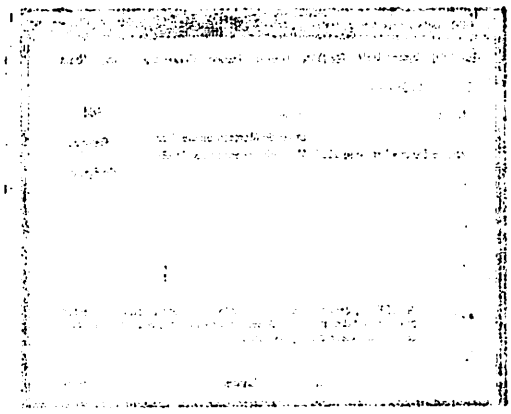
Pada sub bab ini akan dijelaskan tentang penggunaan aplikasi tiap-tiap sistem, dari tampilan aplikasi, fungsi, hingga pada cara penggunaannya.

4.3.1 Tampilan Menu Utama

Rancangan Tampilan Menu Utama yang dihasilkan adalah terlihat seperti pada Gambar 4.2



Gambar 4.2 Tampilan Menu Utama

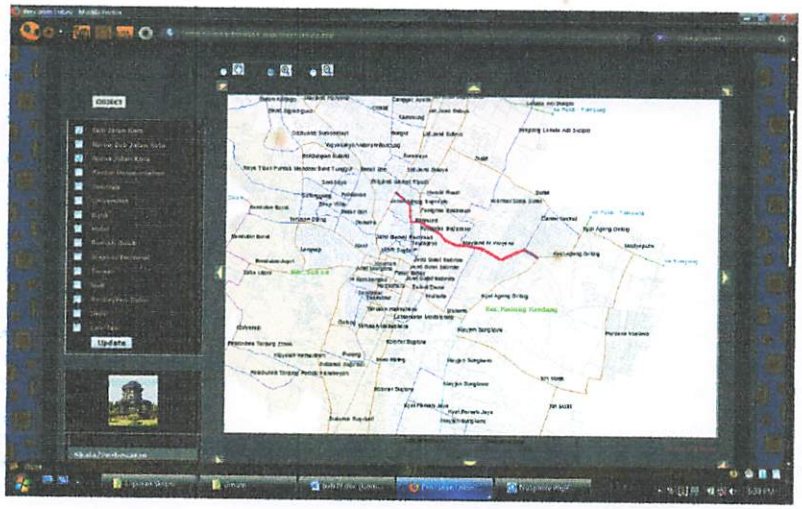


4.3.2 Tampilan Pencarian Rute Jalan Terdekat

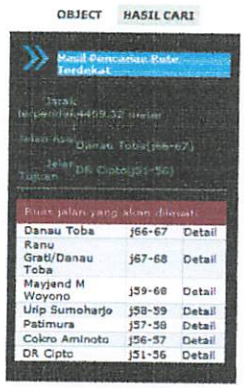
Seperti terlihat pada Gambar 4.3, di dalam halaman ini user akan diberikan pilihan parameter pencarian yang merupakan ruas jalan asal dan ruas jalan tujuan. Ruas jalan asal dan tujuan dipih dengan menggunakan combobox yang telah berisi daftar ruas-ruas jalan yang dapat dipilih. Hasil pencarian setelah button "Cari Rute" diklik maka akan ditampilkan ruas – ruas jalan mana saja yang akan dilewati yang ditampilkan secara visual pada peta, seperti terlihat pada Gambar 4.4. Hasil pencarian rute terdekat akan ditampilkan ke dalam peta beserta nama ruas jalan yang akan dilewati dan jarak tiap-tiap ruas jalan yang dilewati yang nantinya menghasilkan jarak total dari keseluruhan ruas jalan yang dilewati.

Di halaman ini juga terdapat tools untuk melakukan menejemen peta seperti halnya standart aplikasi GIS yang lain, diantaranya adalah :

- Pan : Untuk mengeser tampilan peta.
- Zoom in : Untuk perbesaran tampilan peta.
- Zoom out : Untuk perkecilan tampilan peta.
- Perbesaran : Untuk berapa kali peta akan di perbesar atau diperkecil.
- Skala : Menampilkan skala peta.
- Grid : Menampilkan grid pada peta
- Query peta : Digunakan untuk memilih ruas jalan pada peta, jika dalam kondisi aktif.



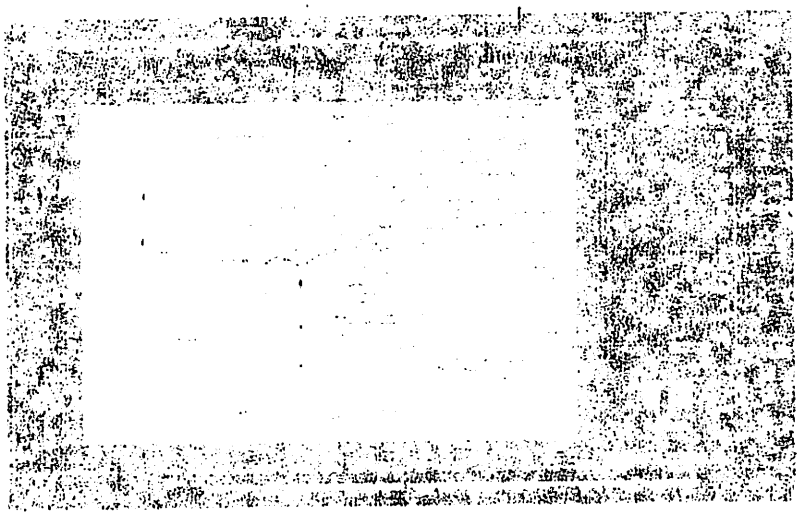
Gambar 4.3 Tampilan pencarian rute jalan terdekat

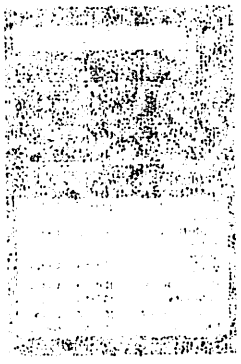


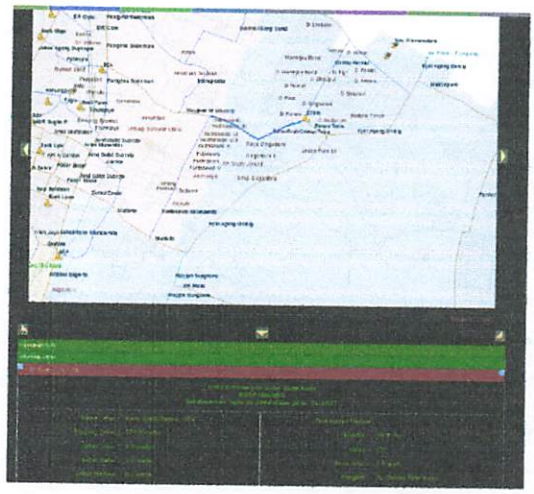
Gambar 4.4 Tampilan hasil pencarian rute jalan terdekat

4.3.3 Tampilan Info Jalan

Di dalam halaman info jalan akan ditampilkan info dari karakteristik jalan yang terseleksi. Seperti terlihat pada Gambar 4.5



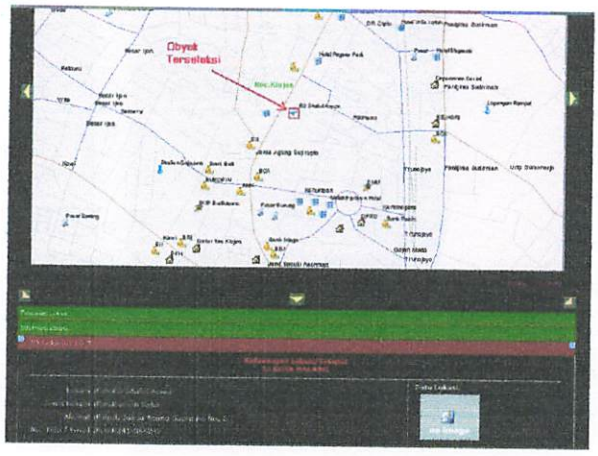




Gambar 4.5 Tampilan info jalan

4.3.4 Tampilan Info Objek

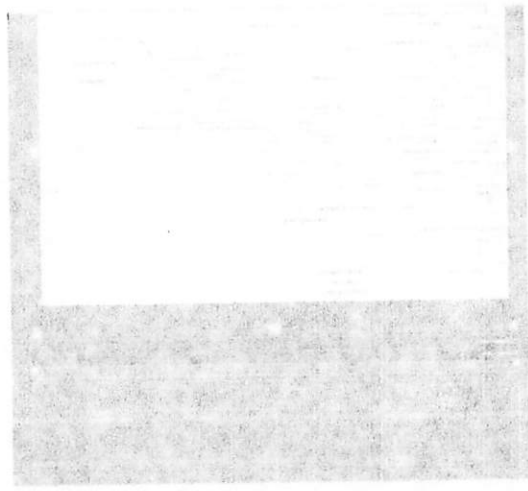
Di dalam halaman info obyek akan ditampilkan info dari karakteristik obyek yang terseleksi. Seperti terlihat pada Gambar 4.6



Gambar 4.6 Tampilan Info Obyek

4.3.5 Tampilan Login Administrator

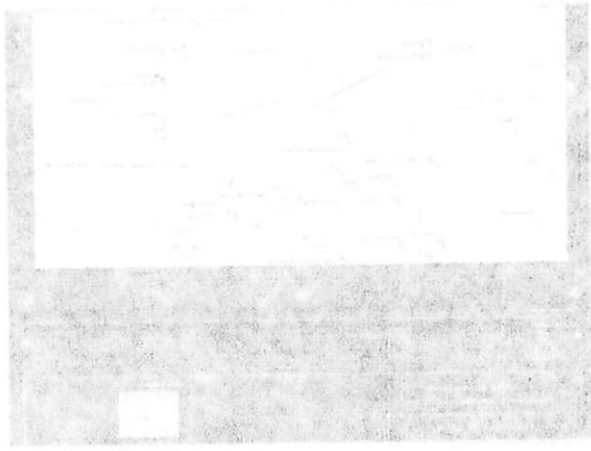
Halaman *admin* merupakan halaman yang dikhususkan untuk user dengan status "admin". Khusus pada halaman admin, ditambahkan aplikasi flex yang akan ditaruh pada frame. Seperti terlihat pada Gambar 4.7



Gambar 4.5 Tampilan info jalan

4.3.4 Tampilan Info Objek

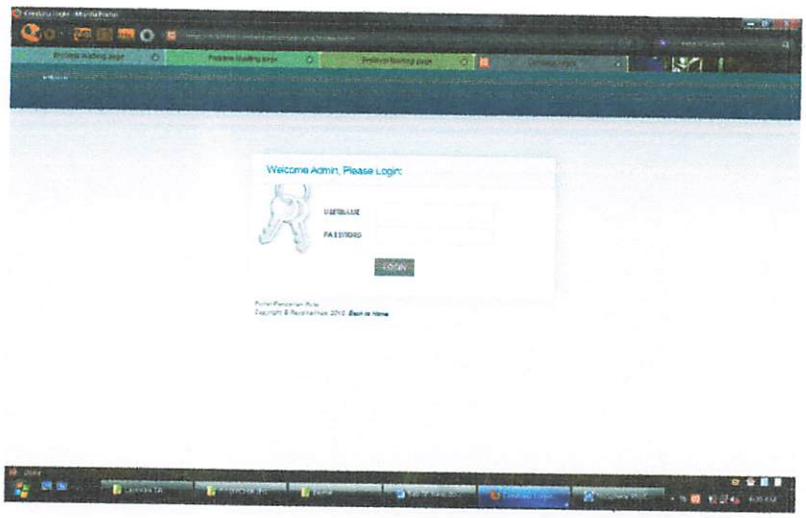
Di dalam halaman info objek, akan ditampilkan info dari kawasrisiko objek yang terseleksi, seperti terlihat pada Gambar 4.6



Gambar 4.6 Tampilan Info Objek

4.3.5 Tampilan Login Administrator

Halaman admin merupakan halaman yang diklusus untuk user dengan status "admin". Khusus pada halaman admin, ditambahkan aplikasi flex yang akan dijumpai pada frame, seperti terlihat pada Gambar 4.7

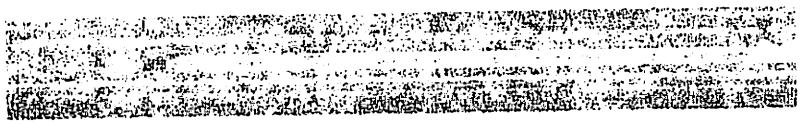


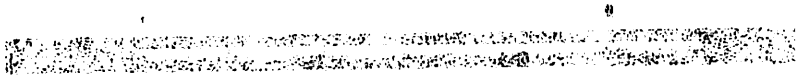
Gambar 4.7 Tampilan login admin

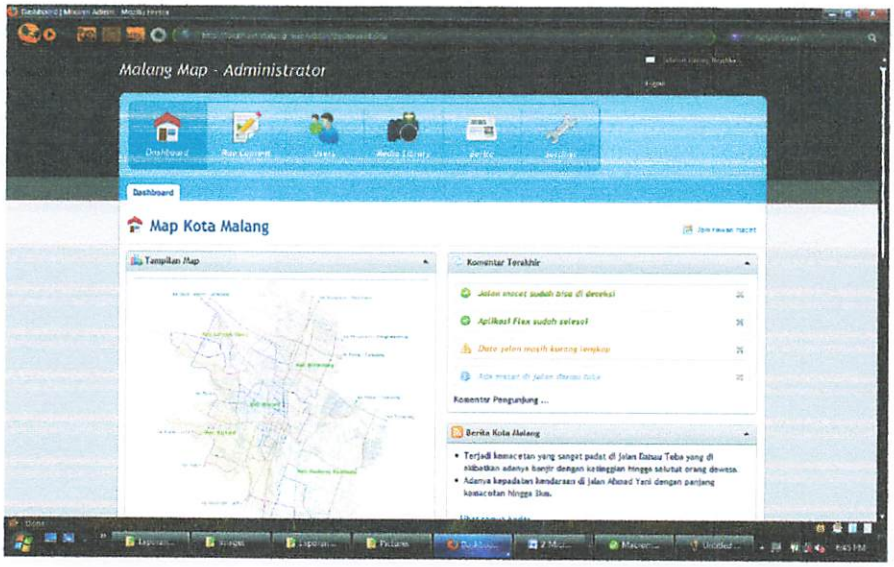
4.3.6 Tampilan Maintenance Data pada Admin

Hak utama admin disini adalah *maintenance (insert, update, delete)* data. Seperti terlihat pada Gambar 4.8, di halaman ini admin / administrator dapat melakkukan hal tersebut. Selain itu admin berhak mengaktifkan dan menonaktifkan user yang telah terdaftar. Data yang dapat di maintenance oleh admin antara lain :

- File attribut GIS
- Data node/vertex
- Data ruas jalan
- Data jalan
- Data lokasi
- Data admin



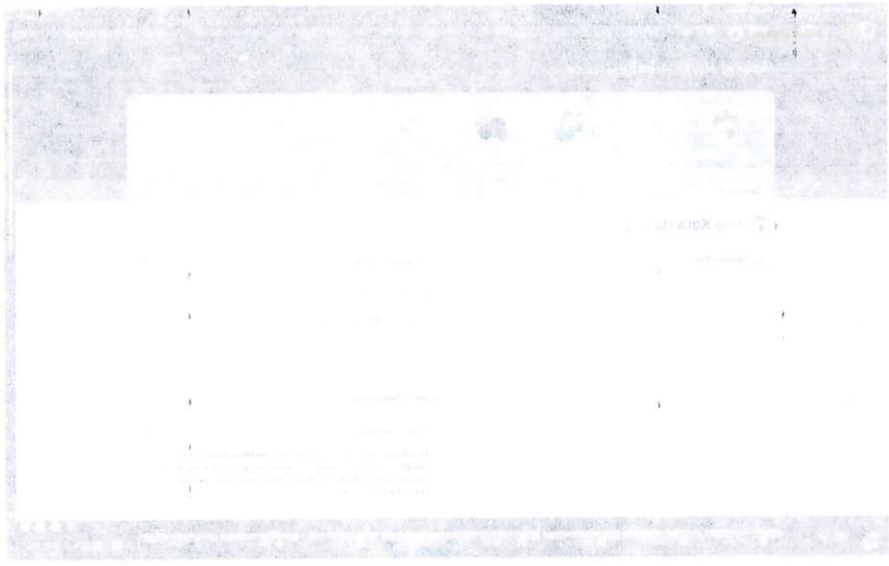




Gambar 4.8 Tampilan maintenance data

4.4 Pengujian Aplikasi

Pengujian dilakukan bertujuan untuk mengetahui apakah aplikasi yang telah dibangun telah berjalan dengan baik dan memenuhi spesifikasi yang telah ditentukan. Pengujian pada aplikasi *web based* GIS penentuan rute terdekat berfokus pada kemampuan pencarian rute, kinerja algoritma $A*$ yang digunakan dan kemapuan dari GIS pada aplikasi.



Gambar 4.8 Tampilan maintenance data

4.4 Pengujian Aplikasi

Pengujian dilakukan bertujuan untuk mengecekan apakah aplikasi yang telah dibangun telah berjalan dengan baik dan memenuhi spesifikasi yang telah ditentukan. Pengujian pada aplikasi *web A.wo/GIS* pemantauan mesin mudah berfokus pada kemampuan pencarian rute, antara algoritma A* yang digunakan dan kemampuan dari GIS pada aplikasi.

4.4.1 Pengujian Perbandingan Jarak Hasil Pencarian pada Aplikasi dengan Jarak Sebenarnya (Jarak Real)

Di dalam pengujian kali ini akan dilakukan pencarian rute terdekat dari ruas jalan "Balai Arjosari(n1-2)" menuju "Soekarno-Hatta(n17-27)" terlihat pada Tabel 4.1, "Danau Toba (n66-67)" menuju "Tugu(n76-77)" terlihat pada Tabel 4.2, dan "DR Cipto(n51-56)" menuju "Semeru(n46-48)" terlihat pada Tabel 4.3. Yang akan dijadikan perbandingan adalah jarak hasil pencarian pada sistem dengan jarak sesungguhnya hasil pengukuran *Spedometer motor* (Jarak Real).

Tabel 4.1 Hasil Pengujian dari "Balai Arjosari (n1-2)" menuju "Soekarno-Hatta(n17-27)" pada aplikasi web

Lokasi Awal	Lokasi Tujuan	Hasil Routing	Jarak
Balai Arjosari (n1-2)	Soekarno Hatta (n17-27)	Jend A Yani Utara (n2-10)	1619.65
		Jend A Yani Utara (n10-13)	150.36
		Borobudur (n13-16)	2385.07
		Soekarno Hatta (n16-17)	949.45
		Soekarno Hatta (n17-27)	802.84

Tabel 4.2 Hasil Pengujian dari "Danau Toba (n66-67)" menuju "Tugu(n76-77)" pada aplikasi web

Lokasi Awal	Lokasi Tujuan	Hasil Routing	Jarak
Danau Toba (n66-67)	Tugu (n76-77)	Ranugrati (n67-68)	372.53
		Mayjend M Wiyono (n59-68)	420.89
		Urip Sumoharjo (n58-59)	1147.74
		Pattimura (n57-58)	244.86
		Trunojoyo (n57-74)	487.37
		Kertanegara (n74-75)	262.72
		Tugu (n75-76)	87.36
		Tugu (n76-77)	74.40

Tabel 4.3 Hasil Pengujian dari "DR Cipto(n51-56)" menuju "Semeru(n46-48)" pada aplikasi web

Lokasi Awal	Lokasi Tujuan	Hasil Routing	Jarak
DR Cipto (n51-56)	Semeru (n46-48)	Jaksa Agung Suprapto (n50-51)	464.08
		Jaksa Agung Suprapto (n49-50)	465.73
		Jaksa Agung Suprapto (n48-49)	251.35
		Semeru (n46-48)	1047.14

Setelah dilakukan pengujian sebanyak 3 kali dengan rute yang sama seperti pada tabel diatas, maka didapat perbandingan jarak hasil pencarian dari sistem dengan jarak sesungguhnya hasil pengukuran *Speedometer*. Hasilnya adalah sebagai berikut :

Selisih jarak hasil dengan Jarak Real : [Jarak Hasil-Jarak Real]

- Percobaan No 1 : [5907.37-5500] = 407.37
- Percobaan No 2 : [4453.8-4400] = 53.8
- Percobaan No 3 : [2692.16-2600] = 92.16

Berdasarkan hasil perhitungan diatas maka selisih rata-rata jarak hasil dengan jarak real adalah 184.44 meter.

4.4.2 Pengujian Implementasi GIS

Untuk pengujian GIS dilakukan pada proses pensekalaan, spaning, zooming, query dan maping. Hasil uji dapat dilihat langsung berupa perubahan yang terjadi pada tampilan peta.

4.4.2.1 Pengujian Spaning (Penggeseran)

Fungsi spaning digunakan untuk menggeser tampilan dari peta. Pengujian dilakukan dengan menggeser tampilan peta secara horisontal maupun vertikal. Hasil pengujian berupa perubahan posisi tampilan peta / extent (minX, minY, maxX, maxY).

- Kondisi awal tampilan peta (Extent: 574.608069853, 2527.53 15289.7119301, 19559.48). Seperti terlihat pada Gambar 4.9

Gambar 4.9 Kondisi awal peta

- Percobaan 1 (Extent: 574.608069853, 2527.53, 15289.7119301, 19559.48). Seperti terlihat pada Gambar 4.10

Gambar 4.10 Hasil percobaan spaning 1

- Percobaan 2 (Extent: 1106.85650733, 7943.94056985, 15821.9603676, 24975.8905699). Seperti terlihat pada Gambar 4.11

Gambar 4.11 Hasil percobaan spaning 2

4.4.2.2 Pengujian Zooming dan Penskalaan

Fungsi *zooming* digunakan untuk merubah ukuran tampilan peta. Ada dua macam *zooming,* yaitu *zoomin* dan *zoomout*. Dalam percobaan ini akan dilakukan fungsi *zoomin* dan *zoomout* pada nilai penskalaan 4, 5 dan 6. Penskalaan merupakan nilai berapa kali tampilan akan di perbesar atau diperkecil. Hasil dari

percobaan ini adalah perubahan ukuran tampilan peta berdasarkan perbesaran yang dilakukan sekaligus nilai skala dari peta.

- Tampilan awal peta (Skala 1:88781). Seperti terlihat pada Gambar 4.12

Gambar 4.12 Peta awal skala 1:88781

- Percobaan 1, zoomin penskalaan 4 kali

  Skala yang ditampilkan 1: 5486

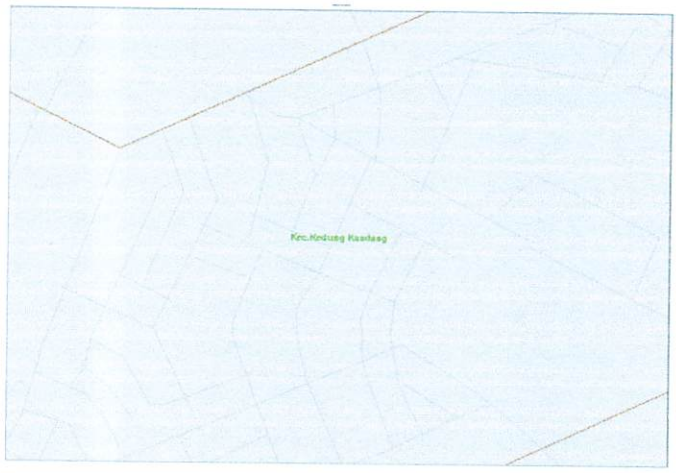
  Seperti terlihat pada Gambar 4.13

Gambar 4.13 Peta dengan skala 1: 5486

- Percobaan 2, zoomin penskalaan 5 kali

  Skala yang ditampilkan 1: 2743

  Seperti terlihat pada Gambar 4.14

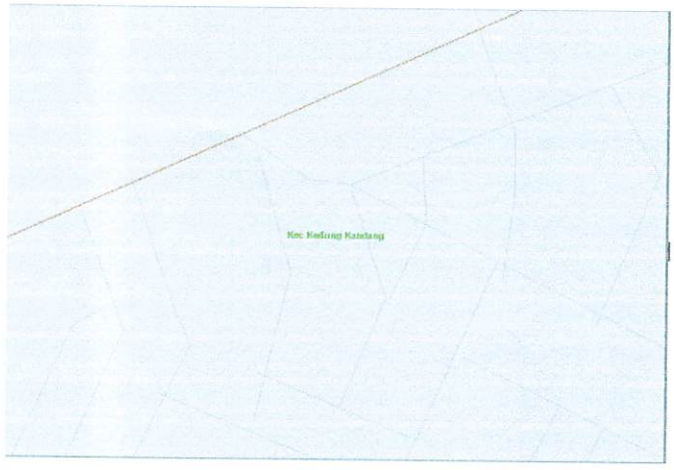


Gambar 4.14 Peta dengan skala 1: 2743

- Percobaan 3, zoomin penskalaan 6 kali

  Skala yang ditampilkan 1: 1371

  Seperti terlihat pada Gambar 4.15



Gambar 4.15 Peta dengan skala 1: 1371

- Percobaan 4, zoomout penskalaan 1 kali

  Skala yang ditampilkan 1: 175562

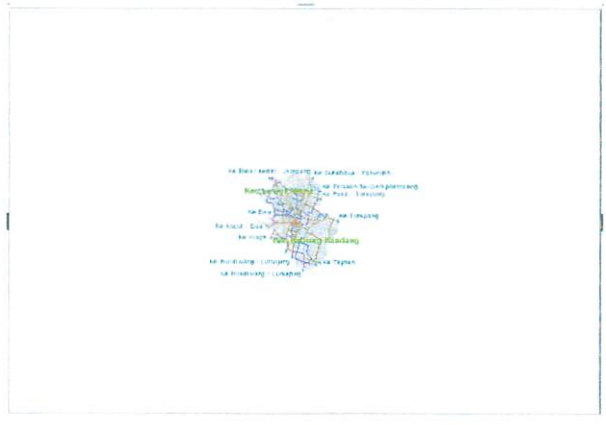
  Seperti terlihat pada Gambar 4.16

Gambar 4.16 Peta dengan skala 1: 175562

- Percobaan 5, zoomout penskalaan 2 kali

  Skala yang ditampilkan 1: 351124

  Seperti terlihat pada Gambar 4.17

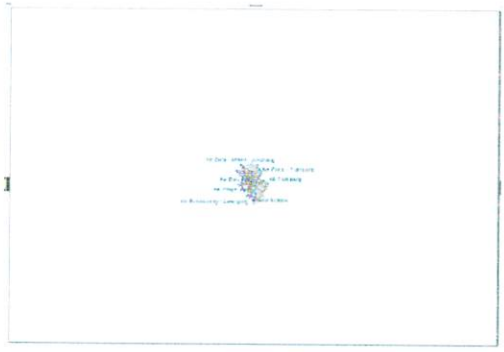


Gambar 4.17 Peta dengan skala 1: 351124

- Percobaan 6, zoomout penskalaan 3 kali

  Skala yang ditampilkan 1: 702248

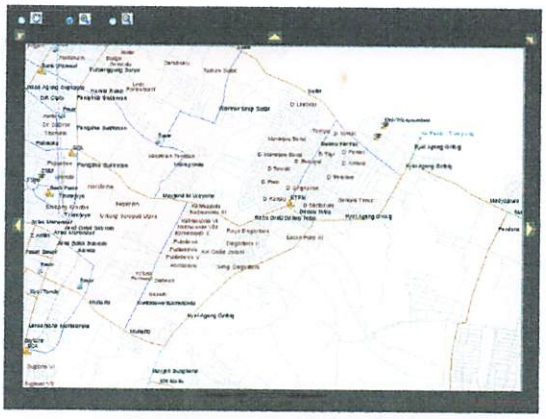
  Seperti terlihat pada Gambar 4.18



Gambar 4.18 Peta dengan skala 1: 702248

4.4.2.3 Pengujian Query Peta

Pengujian Query peta dilakukan dengan memilih ruas jalan yang diklik / dipilih langsung pada tampilan peta. Untuk pengujian ruas jalan yang dipilih adalah "Danau Toba(n66-67)". Hasil percobaan adalah ruas jalan "Danau Toba(n66-67)" yang terblok dengan warna biru pada tampilan peta. Seperti terlihat pada Gambar 4.19



Gambar 4.19 Hasil Query Peta

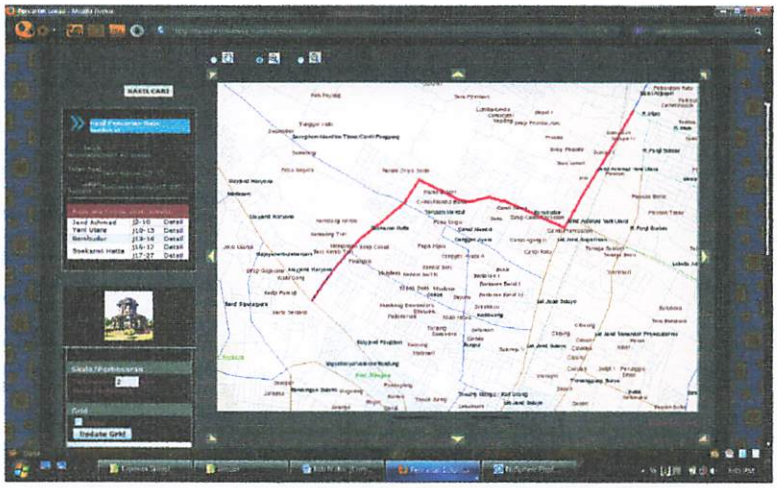
Jika tidak ada ruas yang terpilih maka akan muncul peringatan kesalahan bahwa tidak ada ruas yang terpilih. Seperti terlihat pada Gambar 4.20

Lokasi Awal/Tujuan Belum terseleksi

Gambar 4.20 Pesan kesalahan Query peta

4.4.3 Pengujian Pencarian Rute dengan Parameter Jarak dan Bobot

Di dalam pengujian 1 akan dilakukan pencarian rute terdekat dari "Jl. Balai Arjosari" ke "Jl. Soekarno Hatta". Hasil pengujiannya seperti terlihat pada Gambar 4.21 dan Tabel 4.4

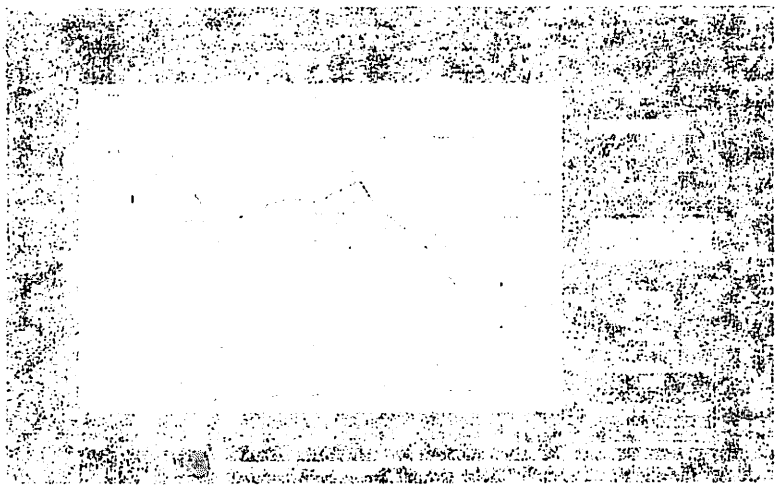


Gambar 4.21 Hasil pengujian 1

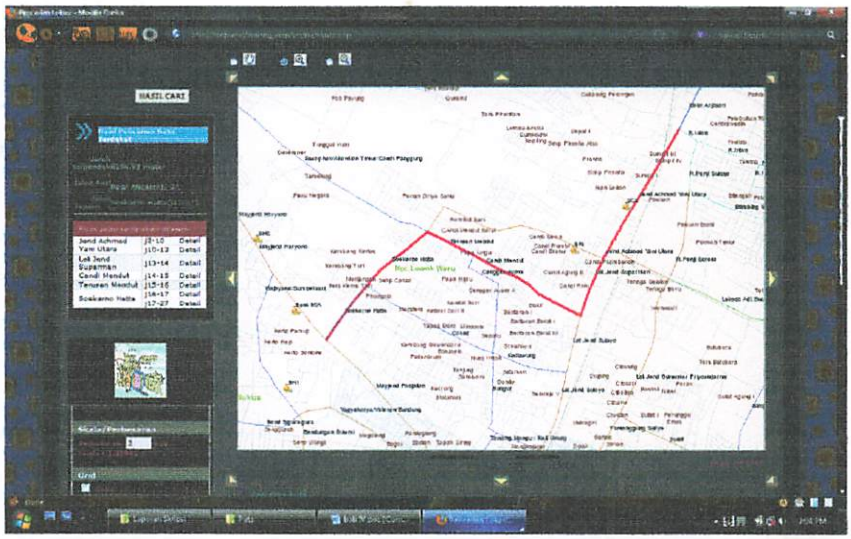
Tabel 4.4 Hasil Pengujian 1 dari "Jl. Balai Arjosari" menuju "Jl. Soekarno Hatta"

Lokasi Awal	Lokasi Tujuan	Hasil Routing	Jarak
Balai Arjosari (n1-2)	Soekarno Hatta (n17-27)	Jend A Yani Utara (n2-10)	1619.65
		Jend A Yani Utara (n10-13)	150.36
		Borobudur (n13-16)	2385.07
		Soekarno Hatta (n16-17)	949.45
		Soekarno Hatta (n17-27)	802.84

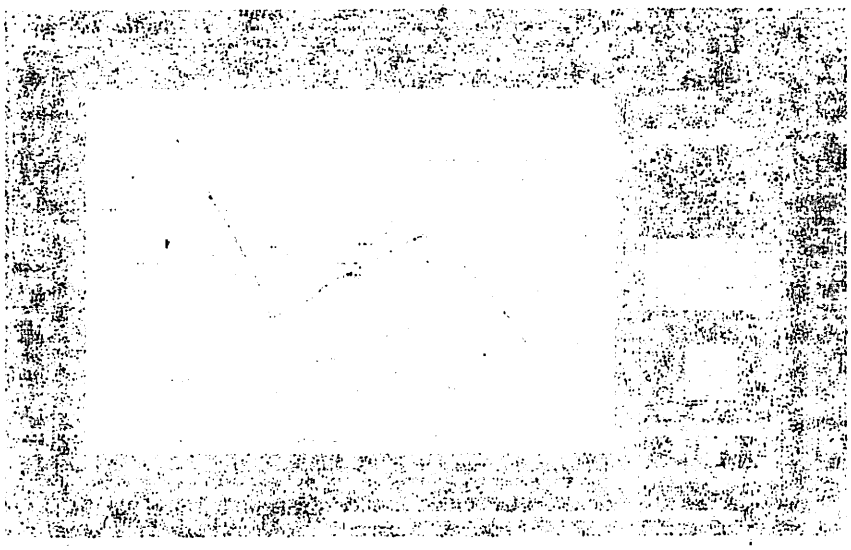
Jarak Total : 5907.37 meter



Untuk pengujian 2 dilakukan pencarian rute terdekat dari "Jl. Balai Arjosari" ke "Jl. Soekarno Hatta" dengan menambahkan bobot karena terjadi kemacetan pada node 13-16 yang merupakan Jl. Borobudur. Hasil pengujiannya seperti terlihat pada Gambar 4.22 dan Tabel 4.5



Gambar 4.22 Hasil pengujian 2



Tabel 4.5 Hasil Pengujian 2 dari "Jl. Balai Arjosari" menuju "Jl. Soekarno Hatta"

Lokasi Awal	Lokasi Tujuan	Hasil Routing	Jarak
Balai Arjosari (n1-2)	Soekarno Hatta (n17-27)	Jend A Yani Utara (n2-10)	1619.65
		Jend A. Yani Utara (n10-13)	150.36
		Let Jend Suparman (n13-14)	715.18
		Candi Mendut (n14-15)	1450.4
		Terusan Mendut (n15-16)	608.75
		Soekarno Hatta (n16-17)	949.45
		Soekarno Hatta (n17-27)	802.84

Jarak Total : 6296.93 meter

BAB V

PENUTUP

5.1 Kesimpulan

Dari pembuatan dan pengujian aplikasi pencarian rute terpendek dengan menggunakan algoritma A* ini, dapat diambil kesimpulan sebagai berikut :

1. Pencarian lintasan terpendek dapat menghasilkan jalan tersingkat untuk menuju suatu tempat dimana terdapat suatu variabel lain yang juga mempengaruhi. Dalam hal ini adalah titik kemacetan yang dapat digunakan sebagai variabel yang mempengaruhi rute perjalanan.
2. Lamanya proses pencarian dan banyaknya node yang dikembangkan untuk pencarian rute dengan metode A*, tergantung pada besar kecilnya ukuran peta dan grid peta serta jarak antara node awal dan node tujuan pada peta.
3. Aplikasi webGIS ini dapat membantu pengguna jalan dalam melakukan pencarian mengenai tempat pelayanan umum di Kota Malang seperti Perguruan Tinggi, SMA, Rumah Sakit, Bank, Kantor-kantor pemerintahan, dll.

5.2 Saran

Perencanaan dan pembuatan aplikasi ini terdapat beberapa hal yang perlu diperhatikan dalam pengembangan lebih lanjut yaitu :

a. Aplikasi ini dapat dikembangkan lagi, dari skala kota menjadi skala propinsi dan dapat dikembangkan ke arah *Graphic User Interface (GUI)* yang lebih menarik dengan mengintegrasikan MapServer dan Flex.

b. Ditambahkan sistem pendukung seperti GPS yang digunakan untuk melakukan proses pembuatan Web GIS yang lebih akurat.

DAFTAR PUSTAKA

[1].Agus Riyanto, 2009. *Implementasi Geographic Information System Berbasis Web untuk Menentukan Rute Jalan Terdekat di Kota Malang*. Laporan Skripsi, Institut Teknologi Nasional, Malang.

[2].Drs.Suarga, M.Sc.,M.Math.,Ph.D, 2006. *Algoritma Pemrograman*. Andi, Yogyakarta.

[3].Eko Priyo Utomo,ST, 2008. *125 Tips Menguasai Bahasa PHP*. Yrama Widya, Bandung.

[4]. Kadir, Abdul. 2003. *Dasar Pemrograman WEB Dinamis Menggunakan PHP*. Andi, Yogyakarta.

[5].Munir,R. 2003. *Matematika Diskrit*, PT. Informatika Bandung, Bandung.

[6] .Nugroho, Bunafit. 2004. *Aplikasi Pemrograman Web Dinamis Dengan PHP Dan MySQL*. Penerbit Gava Media, Yogyakarta.

[7] .Nugroho, Bunafit. 2004. *Trik Dan Rahasia Membuat Aplikasi Web Dengan PHP*. Penerbit Gava Media, Yogyakarta.

[8].Nuryadin R, 2005, *Panduan Menggunakan MapServer*. Informatika, Bandung.

[9].Policyalmanac, A Star Pathfinding for Beginners, http://www.policyalmanac.org/games, tanggal akses : 20 juni 2010.

[10].Prahasta, Eddy. 2001. *Konsep-konsep Dasar Sistem Informasi Geografis*. Informatika Bandung, Bandung.

[11].Siang, J. J., 2006. *Matematika Diskrit dan Aplikasinya pada Ilmu Komputer*, PT. Andi Offset, Yogyakarta.

LAMPIRAN

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG

INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. BendunganSigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
KampusII : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa	: Reydika Ilham
NIM	: 06.12.658
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Komputer dan Informatika
Judul Skripsi	: Pengembangan Aplikasi Pencarian Rute Terpendek dengan Metode Algoritma A* berbasis Web

Dipertahankan dihadapan tim penguji skripsi jenjang Strata Satu (S-1) pada:

Hari	: Sabtu
Tanggal	: 21 Agustus 2010
Dengan Nilai	: 83 (A)

PANITIA UJIAN SKRIPSI

KETUA

Ir. Yusuf Ismail Nakhoda, MT
NIP.Y.1018800189

ANGGOTA PENGUJI

Dosen Penguji I

M. Ibrahim Ashari, ST. MT.
NIP. P. 1030100358

Dosen Penguji II

Sandy Nataly M, SKom.

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK
Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp (0341) 551431 (Hunting) Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo Km 2 Telp (0341) 417636 Fax (0341) 417635 Malang

BERITA ACARA UJIAN SKRIPSI

FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa : Revolta Tilaar
NIM : 04.[illegible].[illegible]
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer dan Informatika
Judul Skripsi : Pengembangan Aplikasi Pencarian Kode Tempat dengan Metode Algoritma A* berbasis Web

Dipertahankan dihadapan tim penguji skripsi jenjang Strata Satu (S-1) pada

Hari : Selasa
Tanggal : 21 Agustus 2007
Nilai : 83 (A)

PANITIA UJIAN SKRIPSI

KETUA

Ir. Abdul Syukor, MT
NIP.Y.1018500190

SEKRETARIS

Dosen Penguji I

Dosen Penguji II

Ir. Abdullah Alkafi, ST, MT.
NIP.P. 1030100358

Ir. Sudirman, MT, Ph.D.

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG

INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

r. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 23 April 2010

Nomor : ITN-606/I.TA/2/10
Lampiran : -
Perihal : BIMBINGAN SKRIPSI

Kepada : Yth. Sdr./i. **DR. ARYUANTO S, ST, MT**
Dosen Institut Teknologi Nasional Malang

Dosen Pembimbing
Jurusan Teknik Elektro S-
di
Malang

Dengan hormat
Sesuai dengan permohonan dan persetujuan dalam Proposal Skripsi
Untuk Mahasiswa :

Nama	: REYDIKA ILHAM
Nim	: 0612658
Fakultas	: Teknologi Industri
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik **Komputer & Informatika**

Maka dengan ini pembimbingan tersebut kami serahkan sepenuhnya kepada Saudara/i selama masa waktu (enam) 6 bulan, terhitung mulai tanggal :

9 April 2010 s/d 9 Oktober 2010

Sebagai satu syarat untuk menempuh ujian Sarjana Teknik,
Jurusan Teknik Elektro S-1
Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih

Ketua Jurusan
Teknik Elektro S-1

Ir. F. Yudi Limpraptono, MT
Nip. Y. 1039500274

Tembusan Kepada Yth :
1. Mahasiswa Yang Bersangkutan
2. Arsip
3. Coret yang tidak perlu

Form. S 4a

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG

INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

(PERSERO) MALANG
IK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 23 April 2010

nor : ITN-607/I.TA/2/10
mpiran : -

pada : Yth. Sdr./i. **AHMAD FAISOL. ST**
Dosen Institut Teknologi Nasional Malang

Dosen Pembimbing
Jurusan Teknik Elektro S-
di
Malang

Dengan hormat
Sesuai dengan permohonan dan persetujuan dalam Proposal Skripsi
Untuk Mahasiswa :

Nama	: REYDIKA ILHAM
Nim	: 0612658
Fakultas	: Teknologi Industri
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik **Komputer & Informatika**

Maka dengan ini pembimbingan tersebut kami serahkan sepenuhnya kepada Saudara/i selama masa waktu (enam) 6 bulan, terhitung mulai tanggal :

9 April 2010 s/d 9 OKTober 2010

Sebagai satu syarat untuk menempuh ujian Sarjana Teknik,
Jurusan Teknik Elektro S-1
Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih

Ketua Jurusan
Teknik Elektro S-1

Ir. F. Yudi Limpraptono, MT
Nip. Y. 1039500274

Tembusan Kepada Yth :
1. Mahasiswa Yang Bersangkutan
2. Arsip
3. Coret yang tidak perlu

Form. S 4a

INSTITUT TEKNOLOGI NASIONAL
Jl. Bendungan Sigura-gura No.2
MALANG

FORMULIR BIMBINGAN SKRIPSI

Nama : Reydika Ilham
NIM : 06.12.658
Masa Bimbingan : 9 April 2010 s/d 9 Oktober 2010
Judul Skripsi : Pengembangan Aplikasi Pencarian Rute Tercepat dengan Menggunakan Metode Algoritma A* berbasis Web

No	Tanggal	Uraian	Paraf Pembimbing
1	19 April 10	Deskripsi Algoritma A-star	
2	03 Mei 10	ACC Bab I	
3	06 Mei 10	ACC Bab II	
4	31 Mei 10	ACC Bab III	
5	19 Juli 10	Demo Program	
6	30 Agustus 10	ACC Bab IV	
7	31 Agustus 10	ACC Bab V	
8	06 September 10	Revisi Tampilan Peta	
9			
10			

Malang,
Dosen Pembimbing I

Dr. Aryuanto S, ST, MT
NIP.Y.1030800417

Form S-4b

INSTITUT TEKNOLOGI NASIONAL
Jl. Bendungan Sigura – gura No.2
MALANG

FORMULIR BIMBINGAN SKRIPSI

Nama : Reydika Ilham
NIM : 06.12.658
Masa Bimbingan : 9 April 2010 s/d 9 Oktober 2010
Judul Skripsi : Pengembangan Aplikasi Pencarian Rute Terpendek dengan Menggunakan Metode Algoritma A* berbasis Web

No	Tanggal	Uraian	Paraf Pembimbing
1		Demo program	
2		Acc Bab I, II	
3		Acc Bab III	
4		Acc Bab IV, V	
5			
6			
7			
8			
9			
10			

Malang,
Dosen Pembimbing II

Ahmad Faisol, ST

Form S-4b

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T.Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : Reydika Ilham
NIM : 0612 658
Perbaikan meliputi :

Tambahkan kalimat pengantar pd tabel dan gbr.
Daftar Pustaka

Malang, 21 agust 2010

M. Ibrahim Ashari, ST MT

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T.Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : RETDIKA ILHAM
NIM : 0612658
Perbaikan meliputi :

1. PENUNJUK RUTE ATAU ARAH WARNANYA DIBUAT KONTRAS ATAU LEBIH TEBAL
2. ~~[illegible]~~
3. PERBAIKI DAFTAR PUSTAKA UTK PENULISAN ARTIKEL DR WEBSITE

Malang, 21 - 08 - 2000

(_____)

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG

INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

I (PERSERO) MALANG
NK NIAGA MALANG

Kampus I : Jl. BendunganSigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
KampusII : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata satu (S-1) Jurusan Teknik Elektro konsentrasi Teknik Komputer dan Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : Reydika Ilham

Nim : 06.12.658

Jurusan : T.Elektro S-1

Konsentrasi : T.Komputer dan Informatika

Masa Bimbingan : 9 April 2010 s/d 9 Oktober 2010

Judul Skripsi : Pengembangan Aplikasi Pencarian Rute Terpendek dengan Metode Algoritma A* berbasis Web

Tanggal	Penguji	Uraian	Paraf
21 Agustus 2010	Penguji I	Tambahkan kalimat pengantar pada Tabel dan Gambar	
	Penguji II	1. Petunjuk arah atau arah warnanya dibuat kontras atau lebih tebal 2. Perbaiki daftar pustaka untuk penulisan artikel dari website	

Mengetahui

Dosen Pembimbing I

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

Dosen Pembimbing II

Ahmad Faisol, ST

Dosen Penguji,

Dosen Penguji I

M. Ibrahim Ashari, ST. MT.
NIP. P. 1030100358

Dosen Penguji II

Sandy Nataly Mantja, S.Kom

SOURCE CODE

```
<?php
include("../Library/Userlogin.php");
?>
<?php
include("../Library/function_ODBC.php");
include("../Library/function_MySQL.php");
include("../Library/function_mapquery.php");
include("../Library/function_GetJalan.php");
function CbxJalan($IsiCombo,$VarBtnReset,$Chk){
global $StatusQueryJalan,$HasilLakukanQueryJalan,$arr_isi_combo;
for($x=0;$x < sizeof($arr_isi_combo);$x++){
if (($arr_isi_combo[$x]==$IsiCombo) && !isset($VarBtnReset)){
    $select="selected='selected'";
}else{
    $select="";
}
if (($StatusQueryJalan=='on')&&($Chk=='on')){
    if($arr_isi_combo[$x]==$HasilLakukanQueryJalan){
        $select="selected='selected'";
    }else{
        $select="";
    }
}
echo ("<option ".$select.">".$arr_isi_combo[$x]."</option>");                    }
?>
<?php
```

```
include("../Library/keterangan.php");
include("../Library/isi_cbx.php");
include("../Library/load_Bobot.php");
include("../Library/load_BobotNode.php");
?>
$awal1=get_IDFromCombo1($_POST['cb_asal']);
$awal2=get_IDFromCombo2($_POST['cb_asal']);
$tujuan1=get_IDFromCombo1($_POST['cb_tujuan']);
$tujuan2=get_IDFromCombo2($_POST['cb_tujuan']);
include("../Library/astar.php");
    $tmp_HasilCari=astar($grap_cari,$awal1,$tujuan1);
    $tmp_bobotHasilCari=explode('#',$tmp_HasilCari);
    $tmp_bobotHasilCari=$tmp_bobotHasilCari[1];
    $valHidHasilCari=$tmp_HasilCari;
    $tmp_bobot=$tmp_bobotHasilCari;
}
if((isset($valHidHasilCari) || strlen($HTTP_POST_VARS['HidHasilCari'])>0) &&
!isset($HTTP_POST_VARS['btn_reset_cari']) && !isset($uji)){
if(isset($valHidHasilCari)){
    $hasil_cari=$valHidHasilCari;
}else{
    $valHidHasilCari=$HTTP_POST_VARS['HidHasilCari'];
    $hasil_cari=$valHidHasilCari;
}
$tmphasil_cari=explode('#',$hasil_cari);
 if ($StatusFungsi == "ON") {
        $objImagePeta = $objMap->DrawQuery();
        }
```

```
    else {
        $objImagePeta = $objMap->Draw();
        }
    $objUrlPeta = $objImagePeta->SaveWebImage();
    $objRect = $objMap->extent->minx." ".$objMap->extent->miny." "
            .$objMap->extent->maxx." ".$objMap->extent->maxy;
    $objImageSkala = $objMap->DrawScalebar();
    $objUrlSkala = $objImageSkala-> SaveWebImage();
    $objImageLegenda = $objMap->DrawLegend();
    $objUrlLegenda = $objImageLegenda-> SaveWebImage();
?>
if((isset($valHidIDLok))&&(strlen($valHidIDLok)>0)&&(!isset($HTTP_POST_VARS['btn_reset']))
  &&
(sizeof($error)==0)&&(cek_AdaDataODBC($valHidIDLok,'ID','Object')==true)){
   $objLayerLokPilih = ms_newLayerObj($objMap);
    $objLayerLokPilih->set ("name","LokPilih");
    $Arr_NamaLayer[]="LokPilih";
    $objLayerLokPilih->set ("type", MS_LAYER_POINT);
    $objLayerLokPilih->set ("status", MS_ON);
   $objLayerLokPilih->Set ("connectiontype", MS_OGR);
    $strSqlJoin = "SELECT * FROM Object LEFT JOIN  '".$map_path."/obj_gr.dbf'
            .obj_gr ON Object.Object_ID=obj_gr.Object_ID";
    $objLayerLokPilih->Set ("connection",$map_path."/Object.shp");
    $objLayerLokPilih->set ("data",$strSqlJoin);
    $objLayerLokPilih->set ("classitem","ID");
    $objClassLokPilih = ms_newClassObj ($objLayerLokPilih);
    $objClassLokPilih->Set ("name", "Lokasi Terpilih");
    $objClassLokPilih->SetExpression ($valHidIDLok);
    $objClassLokPilih->Set ("template",$web_path."/Desain/queryLokasi.html");
```

```
        $objStyleLokPilih = ms_newStyleObj ($objClassLokPilih);
        $objStyleLokPilih->Set ("size", 15);
        $objStyleLokPilih->Set ("symbolname", "Kotak");
        $objStyleLokPilih->color->setRGB (255,0,0);
 }
 $JmlRuteSolusi=sizeof($Arr_IDJalan);
 if (($JmlRuteSolusi>0)&&(!isset($_POST['btn_reset_cari']))){
        $objLayerJalan2 = ms_newLayerObj($objMap);
        $objLayerJalan2->set ("name", "Jalan2");
        $objLayerJalan2->set ("type", MS_LAYER_LINE);
        $objLayerJalan2->set ("status", MS_DEFAULT);
           $objLayerJalan2->set ("data","jalan.shp");
        $objLayerJalan2->set ("classitem","ID"); // field kelas
 for ($x=0;$x<$JmlRuteSolusi;$x++){
 $objClassJalan2Solusi = ms_newClassObj ($objLayerJalan2);
     }
     include("../Library/clear_TmpServer.php");
 $objImagePeta = $objMap->Draw();
 $objUrlPeta = $objImagePeta->SaveWebImage();
 $objRect = $objMap->extent->minx." ".$objMap->extent->miny." "
 $objMap->extent->maxx." ".$objMap->extent->maxy;
 $objImageSkala = $objMap->DrawScalebar();
 $objUrlSkala = $objImageSkala-> SaveWebImage();
 $objImageLegenda = $objMap->DrawLegend();
$objUrlPeta=$image_onServer.get_FileName($objUrlPeta);
 $objUrlSkala = $image_onServer.get_FileName($objUrlSkala);
 $objUrlLegenda = $image_onServer.get_FileName($objUrlLegenda);
?>
```

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Pencarian Lokasi</title>
<script type="text/javascript" src="jquery.js"></script>
<style type="text/css">
@import url(css.css);
</style>
<script type="text/javascript" src="../JS/js_page.js"></script>
<link rel=stylesheet href="../Style/Style.css" type=text/css>
<link rel="stylesheet" href="../js/stylecheck.css" type="text/css" media="screen"/>
<!-- login -->
    <link rel="stylesheet" href="../style/stylelogin.css" type="text/css" media="screen" />
    <link rel="stylesheet" href="../style/slidelogin.css" type="text/css" media="screen" />
    <script src="../js/jquery-1.3.2.min.js" type="text/javascript"></script>
    <script src="../js/slide.js" type="text/javascript"></script>
<!-- end login -->
<!-- Tab -->
   <link href="styletab.css" rel="stylesheet" type="text/css" />
   <script type="text/javascript" src="jquery-ui-personalized-1.5.2.packed.js"></script>
   <script type="text/javascript" src="sprinkle.js"></script>
<!-- End Tab -->
<!-- Accordion -->
   <script type="text/javascript" src="../js/accordion/menu.js"></script>
   <link rel="stylesheet" type="text/css" href="../js/accordion/style.css" />
<!-- End Accordion -->
</head>
<body onLoad="checkawal(form1.zoom,form1.btntool)">
```

```
<!-- Panel -->
<div id="toppanel">
    <div id="panel">
        <div class="content clearfix">
                <div class="left">
                        <h1>Login Panel</h1>
                        <h2>Sliding login panel</h2>
                        <p class="grey">Anda dapat mencari rute terpendek di wilayah kota
    malang</p>
                        <h2></h2>
                </div>
                <div class="left">
                        <!-- Login Form -->
                        <form class="clearfix" action="#" method="post">
                                <h1>Member Login</h1>
                                <label class="grey" for="log">Username:</label>
                                <input class="field" type="text" name="log" id="log" value=""
    size="23" />
                                <label class="grey" for="pwd">Password:</label>
                                <input class="field" type="password" name="pwd" id="pwd"
    size="23" />

                        <div class="clear"></div>
                                <input type="submit" name="btn_login" value="Login"
    class="bt_login" />
</form>
                </div>
                <div class="left right">
                        <!-- Register Form -->
                        <form action="#" method="post">
```

```
                    <h1>Not a member yet? Sign Up!</h1>
                    <label class="grey" for="signup">Username:</label>
                    <input class="field" type="text" name="signup" id="signup"
value="" size="23" />
                    <label class="grey" for="email">Email:</label>
                    <input class="field" type="text" name="email" id="email" size="23"
/>
                    <label>A password will be e-mailed to you.</label>
                    <input type="submit" name="submit" value="Register"
class="bt_register" />
                </form>
            </div>
        </div>
    </div> <!-- /login -->
    <!-- The tab on top -->
    <div class="tab">
        <ul class="login">
            <li class="left"> </li>
            <li>Hello Guest!</li>
            <li class="sep">|</li>
            <li id="toggle">
                <a id="open" class="open" href="#">Log In | Register</a>
                <a id="close" style="display: none;" class="close" href="#">Close Panel</a>
            </li>
            <li class="right"> </li>
        </ul>
    </div>
  <!-- / top -->
</div>
<!-- end panel -->
```

```
<br>
  <div id="wrapper">
  <h1>Pencarian Rute Jalan ... Kota Malang</h1><br>
  <ul id="nav">
    <li><a href="route.php">Cari Rute </a></li>
    <li><a href="object.php">Cari Lokasi</a></li>
    <li><a href="Detail_rute.php">Cari Jalan</a></li>
  </ul>
   <form name="form1" method=POST action=<?php echo
   $HTTP_SERVER_VARS['PHP_SELF']; ?> >
 <div id="content2" class="content2">
   <center>
   <table width="1100" border="0" cellpadding="2">
 <tr>
   <!-- rows.1 cols.1 -->
     <td width="300" valign="top">
   <br/><br/><br/><br/>
  <!-- table on rows.1 cols.1 -->
     <center>
       <table width="100%" border="0" cellpadding="5" cellspacing="2">
      <tr>
      <td>
              <!-- Tab Widget -->
         <div id="tabvanilla" class="widget">

     <ul class="tabnav">
       <li><a href="#popular">Object</a></li>
       <li><a href="#recent">Hasil Cari</a></li>
       <li><a href="#featured">Featured</a></li>
```

```
    </ul>


    <div id="popular" class="tabdiv">

      <table width="100%">

          <tr>

              <td width="15%"><input name="ChkJalan2" type="checkbox" id="ChkJalan2"
value="on"

                          <?php echo($StatChkNotJlnKota==1?"checked='checked'":""); ?>  /></td>

              <td width="85%"><font color="#E0E0E0"><font color="#E0E0E0">Sub Jalan
Kota</font></td>

          </tr>

          <tr>

            <td><?php

              if($StatChkNotJlnKota!=1){

                $dis='disabled';

              }

              if($StatChkLblNotJlnKota==1){

                $chk='checked';

              }

              ?>

              <input name="ChkNamaJalan2" type="checkbox" id="ChkNamaJalan2"

             value="on" <?php echo($dis.' '.$chk); ?> /></td>

            <td><font color="#E0E0E0">Nama Sub Jalan Kota</font></td>

          </tr>

          <tr>

            <td><input type="checkbox" name="ChkJalan" value="on"

              <?php echo($StatChkJalan==1?"checked='checked'":""); ?>  /></td>

            <td><font color="#E0E0E0">Nama Jalan Kota</font></td>

          </tr>
```

```
<tr>
<td><input type="checkbox" name="ChkSekolah" value="on"
<?php
echo($StatChkSekolah==1?"checked='checked'":""); ?> /></td>
<td><font color="#E0E0E0">Sekolah/Universitas</font></td>
</tr>
<tr>
<td><input type="checkbox" name="ChkBank" value="on"
<?php
echo($StatChkBank==1?"checked='checked'":""); ?> /></td>
<td><font color="#E0E0E0">Bank</font></td>
</tr>
<tr>
<td><input type="checkbox" name="ChkPasar" value="on"
<?php
echo($StatChkPasar==1?"checked='checked'":""); ?> /></td>
<td><font color="#E0E0E0">Pasar/Swalayan</font></td>
</tr>
<tr>
<td><input type="checkbox" name="ChkMuseum" value="on"
<?php
echo($StatChkMuseum==1?"checked='checked'":""); ?> /></td>
<td><font color="#E0E0E0">Museum</font></td>
</tr>
<tr>
<td><input type="checkbox" name="ChkHotel" value="on"
<?php
echo($StatChkHotel==1?"checked='checked'":""); ?> /></td>
<td><font color="#E0E0E0">Hotel</font></td>
</tr>
```

```
<tr>
<td><input type="checkbox" name="ChkStasiun" value="on"
<?php
echo($StatChkStasiun==1?"checked='checked'":""); ?> /></td>
<td><font color="#E0E0E0">Stasiun</font></td>
</tr>
<tr>
<td><input type="checkbox" name="ChkRS" value="on"
<?php
echo($StatChkRS==1?"checked='checked'":""); ?> /></td>
<td><font color="#E0E0E0">Rumah Sakit</font></td>
</tr>
<tr>
<td><input type="checkbox" name="ChkTerminal" value="on"
<?php
echo($StatChkTerminal==1?"checked='checked'":""); ?> /></td>
<td><font color="#E0E0E0">Terminal</font></td>
</tr>
<tr>
<td><input type="checkbox" name="ChkMonument" value="on"
<?php
echo($StatChkMonument==1?"checked='checked'":""); ?> /></td>
<td><font color="#E0E0E0">Monument</font></td>
</tr>
<tr>
<td><input type="checkbox" name="ChkInstansi" value="on"
<?php
echo($StatChkInstansi==1?"checked='checked'":""); ?> /></td>
<td><font color="#E0E0E0">Kantor/Instansi</font></td>
</tr>
```

```
                    <tr>
        <td><input type="checkbox" name="ChkPabrik" value="on"
                                        <?php
echo($StatChkPabrik==1?"checked='checked'":""); ?>  /></td>
        <td><font color="#E0E0E0">Pabrik</font></td>
      </tr>
                    <tr>
        <td><input type="checkbox" name="ChkLain2" value="on"
                                        <?php
echo($StatChkLain2==1?"checked='checked'":""); ?>  /></td>
        <td><font color="#E0E0E0">Lain-lain</font></td>
      </tr>
                    <tr>
        <td> </td>
        <td><input type=submit name="UpdateObject" value="Update" class="btm1"></td>
      </tr>


    </table>
    </div><!--/popular-->
  <div id="recent" class="tabdiv">
    <?php include("../Template/cari_rute/Template_HasilCari.php"); ?>
  </div><!--/recent-->
  <div id="featured" class="tabdiv">
    <ul>
    </ul>
  </div><!--featured-->
</div>
  <!-- End Tab Widget-->
    </td>
```

```
        </tr>
        <tr>
                <td></td>
        </tr>
        <tr>
        <td><?php include("../Template/cari_lokasi/Template_NavLokasi.php"); ?>
                                <br>
                                <?php include("../Template/umum/Template_hiden.php"); ?>
                        </td>
        </tr>
      </table>
    </center>
  <!-- end table on rows.1 cols.1 -->
    </td>
  <!-- end rows.1 cols.1 -->
<td colspan="2" width="800" valign="top">
      <center>
  <!-- Table Kanan -->
      <table width="95%" border="0" cellpadding="2">
      <tr>
      <td width="9%" class="brdnav1">
              <input type=radio name="zoom" value=0
onClick="check(form1.zoom,0,form1.btntool)"
                                <?php
                                        if
(($check_pan=='checked')||($HTTP_POST_VARS['zoom']==0)){
                                        echo ("checked");
                                }
                                ?>
```

```
/>

<img name="btntool" src= "../Image/bi_spaning.gif" width="20" height="20"
onClick="check(form1.zoom,0,form1.btntool)" onMouseOver="stylehand(this)">

</td>

<td width="8%" class="brdnav1">

<input type=radio name="zoom" id="zoomin" value=1
onClick="check(form1.zoom,1,form1.btntool)"

<?php

if
(($check_zoomin=='checked')||($HTTP_POST_VARS['zoom']==1)){

echo ("checked");

}

?>

/>

<img name="btntool" src= "../Image/bi_zoomin.gif" width="20" height="20"
onClick="check(form1.zoom,1,form1.btntool)" onMouseOver="stylehand(this)">

</td>

<td width="8%" class="brdnav1">

<input type=radio name="zoom" value=-1
onClick="check(form1.zoom,2,form1.btntool)"

<?php

if
(($check_zoomout=='checked')||($HTTP_POST_VARS['zoom']==-1)){

echo ("checked");

}

?>

/>

<img name="btntool" src= "../Image/bi_zoomout.gif" width="20" height="20"
onClick="check(form1.zoom,2,form1.btntool)" onMouseOver="stylehand(this)">

</td>

<td width="75%" colspan="2"> </td>
```

```
    </tr>
    <tr>
    <td colspan="5"><?php include("../Template/umum/Template_peta.php"); ?></td>
</tr>
    <tr>
    <td colspan="5">
   <!-- Accordion -->
   <ul id="menu_ac">
      <li>
         <a href="#">Pencarian Rute</a>
         <ul>
            <li><?php include("../Template/cari_rute/Template_InputCari.php"); ?> </li>
         </ul>
      </li>
      <li>
         <a href="#">Informasi Jalan</a>
         <ul>
            <li><?php include("../Template/detail_rute/frm_DetailRuas.php"); ?> </li>
         </ul>
      </li>
      <li>
         <a href="#">Pencarian Lokasi</a>
         <ul>
            <li><?php include("../Template/detail_lokasi/Input_DetailLokasi.php"); ?> </li>
         </ul>
      </li>
      <li>
```

```
            <a href="#">Informasi Lokasi</a>
            <ul>
              <li><?php include("../Template/detail_lokasi/frm_DetailLokasi.php"); ?> </li>
            </ul>
          </li>
        </ul>
        <!-- End Accordion -->
        </td>
      </tr>
        </table>
      <!-- End Table Kanan -->
        </center>
    </td>
    </tr>
    </table>
    </center>
  </div>
    </form>
</body>
</html>
```