

Buku Ajar

Sistem Informasi Geografis Lanjut



Eko Budiyanto, S.Pd., M.Si.

**Jurusan Pendidikan Geografi
Fakultas Ilmu Sosial dan Hukum
Universitas Negeri Surabaya
2016**

Rencana Perkuliahan Semester

Buku ajar Sistem Informasi Geografis Lanjut disusun untuk memenuhi kebutuhan dan pengayaan bahan ajar pada mata kuliah tersebut. Buku ajar ini dirancang secara ringkas, padat, dan memuat pokok materi yang harus dikuasai oleh mahasiswa. Proses pembelajaran dilakukan melalui metode diskusi dan praktek terbimbing. Adapun rencana perkuliahan semester disajikan dalam tabel berikut :

Kompetensi Mahasiswa	Materi	Pertemuan
Memahami konsep Sistem Informasi Geografis aplikatif	Bab 1	Pertemuan 1
Memahami proses-proses pengolahan vektor	Bab 2	Pertemuan 2,3
Memahami teknik analisis morfologi dasar	Bab 3	Pertemuan 4,5
Memahami teknik analisis geostatistik dasar	Bab 4	Pertemuan 6,7
Ujian Tengah Semester		Pertemuan 8
Menguasai keterampilan Web GIS berbasis Open Layer	Bab 5	Pertemuan 9, 10, 11
Menguasai keterampilan Web GIS berbasis MapServer	Bab 6	Pertemuan 12, 13, 14, 15
Ujian Akhir Semester		Pertemuan 16

Surabaya, Agustus 2016

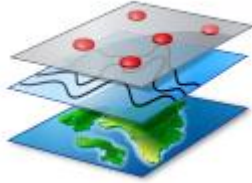
Eko Budiyanto, S.Pd., M.Si.

Daftar Isi

Bab 1. Aplikasi Sistem Informasi Geografis	1
A. Data Geospasial	2
B. Bentuk data Geospasial	3
C. Tipe data	3
D. Latihan	5
Bab 2. Geoprosesing Dasar	7
A. Buffer	7
B. Intersect	8
C. Union	9
D. Symetrical Difference	9
E. Clip	10
F. Dissolve	11
G. Elliminate Sliver Polygons	11
H. Latihan	12
Bab 3. Analisis Morfologi Dasar	13
A. Kontur	13
B. Relief	14
C. Aspect	15
D. Hillshade	17
E. Ruggedness Index	17
F. Latihan	18
Bab 4. Analisis Geostatistik Dasar	19
A. Scatterplot	19
B. Visualisasi data	21
C. Statistik Dasar	21
D. Nearest Neighbour Analysis	22
E. Latihan	24
Bab 5. Web GIS Berbasis OpenLayers	25
A. Web GIS	25
B. Persiapan	27
C. Instalasi Platform Web Server	27
D. Instalasi Notepad	29
E. Folder Proyek	31
F. Data Spasial	32
G. Konversi koordinat ke data vektor	33
H. Menyusun kode	35
I. Memanggil data vektor	38
J. Simbol data vektor	41
K. Simbol ikon	48
L. Label	49
M. Kontrol skala	51
N. Posisi mouse	51
O. Zoom Slider	53

P. OverviewMap	54
Q. FullScreen	55
R. Map Interaction	56
S. Overlay Layer	58
T. Latihan	51
Bab 6. Web GIS berbasis Mapserver	63
A. Konfigurasi Awal	63
B. Script Mapserver	64
B.1. Struktur script .map	64
B.2. Membuat script map sederhana	65
B.3. Menampilkan layer ganda	66
B.4. Menampilkan simbolisasi peta	68
B.5. Annotasi	69
B.6. Hubungkan dengan menu html	71
C. Tampilkan peta dengan template	72
D. Menambah properti peta	76
D.1. Pengatur ukuran tampilan peta	76
D.2. Skala Grafis	78
D.3. Legenda	79
E. Menggunakan Widget Chameleon	83
F. Latihan	91

Bab 1. Aplikasi Sistem Informasi Geografis



Sebuah sistem informasi geografis (SIG) mengintegrasikan perangkat keras, perangkat lunak, dan data untuk perolehan, pengelolaan, analisa, dan menampilkan berbagai bentuk informasi berreferensi geografis. Sistem informasi geografis menurunkan berbagai informasi dari dunia nyata di muka bumi. Dalam perkembangannya saat ini, Sistem informasi geografis menempatkan basis data informasi pada suatu ruang waktu geografis untuk mempertajam analisis dan pengambilan keputusan. Sistem informasi geografis lebih lanjut banyak diaplikasikan secara online baik sebagai dasar pengambilan keputusan ataupun sebagai media penyebaran informasi kepada masyarakat.

Teknologi sistem informasi geografis digunakan untuk mengatur dan memanfaatkan data geografis, dan secara luas dikenal sebagai alat dalam berbagai bidang seperti pengelolaan kehutanan, perencanaan perkotaan, teknik sipil, pengelolaan permukiman, bisnis, dan studi lingkungan hidup. Beberapa contoh aplikasi dalam sistem informasi geografis adalah sebagai berikut :

a. Pemetaan Lokasi

Fungsi ini adalah fungsi yang paling mendasar dan umum. Fungsi ini diwujudkan sebagai suatu proses pemetaan digital yang ada dalam SIG. Secara umum dikenal berbagai teknik digitasi dan pemrosesan data dasar lainnya hingga proses layouting.

b. Pemetaan informasi kuantitas

Fungsi ini mengaplikasikan data-data kuantitatif untuk memunculkan informasi tematik tertentu, seperti informasi tertinggi, terluas, terbesar atau sebaliknya. Fungsi ini juga mencakup tentang keterkaitan data dengan data yang lainnya dalam konteks ruang.

c. Pemetaan kepadatan

Fungsi ini menekankan pada model konsentrasi sebaran data. Informasi ditampilkan dalam bentuk diagram, warna ataupun tekstur pada peta poligon, garis, ataupun titik.

d. Pencarian informasi data spasial

Fungsi ini memberikan kemudahan dalam pencarian data yang memenuhi kriteria tertentu, baik terhadap informasi yang ada dalam satu data spasial ataupun lebih. Proses dalam fungsi ini banyak dikembangkan sebagai model analisis spasial dalam SIG.

e. Monitoring Perubahan

Data temporal dapat digunakan dalam analisis trend atau perubahan antar waktu. Analisis ini didasarkan pada perubahan kuantitas data ataupun perubahan spasialnya.

A. Data Geospasial

Data geospasial mencakup dua komponen yaitu komponen **spasial** dan komponen **tematik**. Kedua komponen tersebut saling terkait dan saling memperkuat informasi yang dikandung dalam data tersebut. Komponen spasial dan komponen tematik dapat dianalisis secara bersama ataupun terpisah dari masing-masing komponen tersebut.

Komponen spasial memberikan keterangan tentang lokasi dari keberadaan data tersebut. Terdapat dua bentuk dari aspek komponen spasial ini yaitu **lokasi absolut** dan **lokasi relatif**. Lokasi absolut merupakan lokasi yang mendasarkan pada posisi koordinat tertentu dalam sebuah sistem proyeksi. Koordinat menunjukkan lokasi data secara pasti yang tidak akan diterjemahkan berbeda antara satu pengguna dengan pengguna lainnya. Lokasi relatif menunjuk suatu lokasi berbanding pada suatu lokasi data lainnya. Misal, kampus Unesa terletak di sebelah barat dari jalan Ahmad Yani Surabaya. Informasi ini menunjuk pada arah tertentu yang dibandingkan dengan posisi fitur tertentu yang lain pada sebuah data.

Komponen tematik merujuk pada jenis informasi yang terkandung pada data spasial. Informasi ini dapat berbentuk kuantitatif ataupun deskriptif dan

terrekam sebagai data atribut. Komponen tematik memunculkan informasi tematik yang dimunculkan atas sebuah peta dasar.

B. Bentuk data Geospasial

Data geospasial sering diwujudkan dalam dua bentuk yaitu data digital dan data analog. Data digital dihasilkan atas suatu proses digital menggunakan perangkat komputer, sedangkan data analog dihasilkan dari proses manual ataupun cetak dari data digital. Berikut karakteristik dari dua bentuk data tersebut.

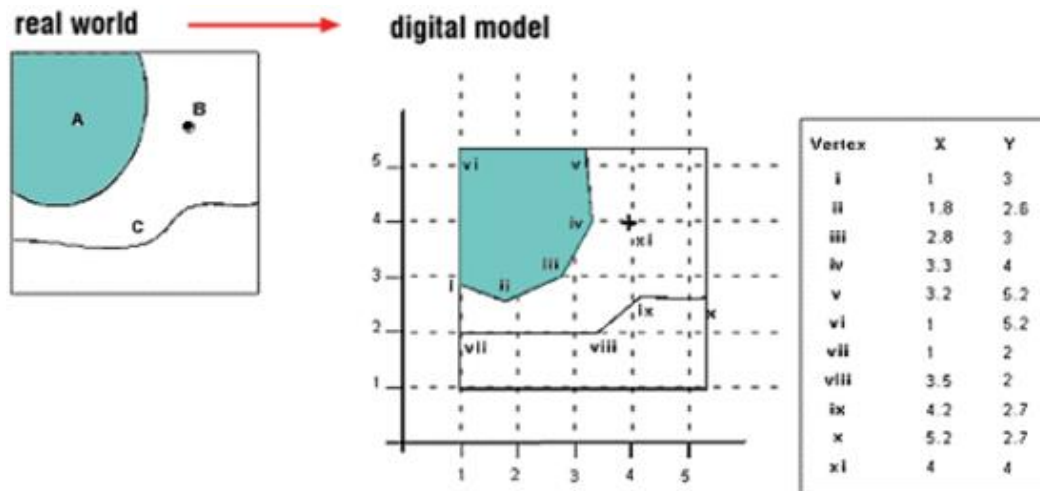
Tabel 1. Karakteristik data geospasial

Data Digital	Data Analog
Mudah diperbarui	Perbaikan harus meliputi seluruh peta
Mudah dan cepat dipindahkan	Sulit untuk proses pemindahan data
Ruang penyimpanan kecil	Diperlukan ruang luas untuk penyimpanan
Mudah di kelola	Diperlukan kertas cetak yang terpisah
Data dilakukan analisis terotomasi	Sulit dan kurang akurasi dalam proses analisis

C. Tipe data

Data geospasial yang digunakan dalam sistem informasi geografis diwujudkan dalam dua tipe data, yaitu tipe **vektor** dan tipe **raster**. Kedua tipe ini dapat digunakan secara bersamaan ataupun terpisah. Data vektor merupakan data yang lebih umum digunakan dalam berbagai fungsi sistem informasi geografis ini.

Data tipe vektor terdiri dari titik dan garis atau lengkungan (arc). Sebuah data vektor garis dibentuk oleh sebuah *start node* dan diakhiri dengan sebuah *end node*. Pada garis lengkung, terdapat beberapa node diantara start node dan end node tersebut. Node di antara start node dan end node disebut sebagai *vertex*. Garis digunakan untuk merepresentasikan fitur memanjang yang terlalu kecil untuk ditampilkan sebagai poligon pada skala tertentu seperti jalan dan sungai. Garis juga digunakan untuk merepresentasikan fitur yang tak memiliki informasi ruang seperti batas administratif desa, kota, atau negara.



Pada sebuah poligon, start node dan end node berada pada satu koordinat yang sama. Start node dan end node haruslah bertemu pada koordinat yang sama tersebut, karena jika tidak bertemu, maka hanya akan membentuk sebuah arc. Poligon digunakan untuk merepresentasikan sebuah area.

Titik atau point adalah sebuah abstraksi data tak berdimensi yang dibentuk dari sebuah koordinat xy. Titik biasanya digunakan untuk merepresentasikan fitur yang terlalu kecil dalam konteks skala untuk ditampilkan sebagai sebuah poligon atau garis, misal lokasi dari sebuah gedung, atau juga sebuah kota pada peta skala kecil.

raster representation

A	A	A	A	0	0	0	0
A	A	A	A	A	0	0	0
A	A	A	A	0	B	0	0
A	A	A	A	0	0	0	0
A	A	A	0	0	0	C	C
0	0	0	0	0	C	0	0
C	C	C	C	C	0	0	0
0	0	0	0	0	0	0	0

Data tipe raster merupakan data yang diwujudkan dalam sekumpulan grid piksel. Data raster merupakan abstraksi dunia nyata yang diekspresikan sebagai matrik sel atau sering disebut sebagai piksel dengan informasi posisi spasial yang tersimpan secara implisit pada susunan piksel tersebut. Data raster memiliki struktur tertentu tergantung pada sistem kompresi atau format datanya.

Data spasial yang secara umum disimpan dalam bentuk raster misalnya adalah data citra satelit. Informasi yang tersimpan pada data tipe raster ini akan menjadi data yang bersifat diskret. Setiap piksel mengandung informasi kuantitatif yang terpisah dengan piksel di sekitarnya. Data diskret tersebut dapat diklasifikasi sehingga dapat dihasilkan data kualitatif yang bersifat kontinyu. Analisis terhadap data raster dapat dilakukan melalui berbagai algoritma terotomasi sehingga dapat dilakukan dengan cepat dan obyektif.



D. Latihan

- Buatlah sebuah peta tematik tertentu pada wilayah lokasi kabupaten anda masing-masing. Cetak dalam sebuah kertas ukuran A3.
- Lama pengerjaan satu minggu dan dikerjakan secara individual
- Perangkat lunak yang digunakan adalah perangkat lunak opensource GIS seperti QGIS, MapWindow dan lain-lain.
- Tidak diperkenankan menggunakan perangkat lunak berbayar, kecuali anda memiliki lisensi resmi atas software yang terinstal tersebut.

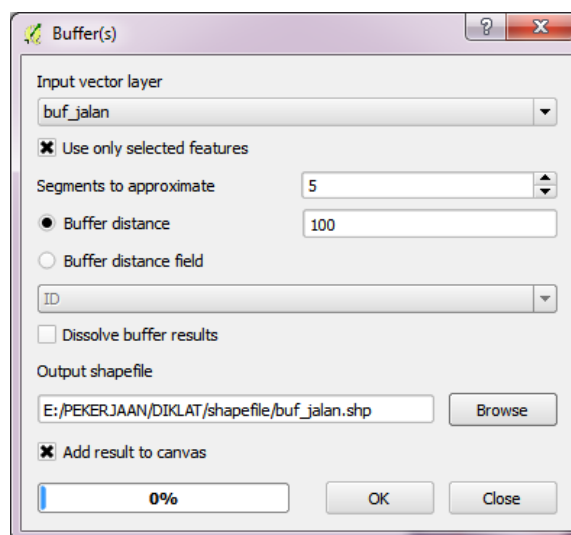
Bab 2. Geoprosesing Dasar

Geoprosesing adalah salah satu proses pengolahan data spasial. Quantum GIS memiliki fasilitas geoprosesing yang telah include pada menu sehingga sangat memudahkan pelaksanaan proses tersebut. Beberapa fasilitas geoprosesing yang disediakan pada Quantum GIS versi 2.6.0 ini adalah *buffer*, *intersect*, *union*, *symetrical difference*, *clip*, *difference*, *dissolve*, dan *eliminate sliver polygons*.

A. Buffer

Buffer adalah pembentukan area sekitar obyek dengan jarak tertentu dari obyek tersebut. Proses buffer dapat dilakukan terhadap seluruh ataupun sebagian obyek pada sebuah data spasial. Proses Buffer dilakukan dengan cara :

- ✓ Klik layer yang akan di-buffer
- ✓ Pilih menu Vector – Geoprocessing



- ✓ Pilih layer yang akan di-buffer pada Input vector layer
- ✓ Klik use only selected features, jika hanya beberapa obyek terpilih saja yang akan di-buffer.
- ✓ Isikan luas buffer yang diinginkan misal : 100 meter

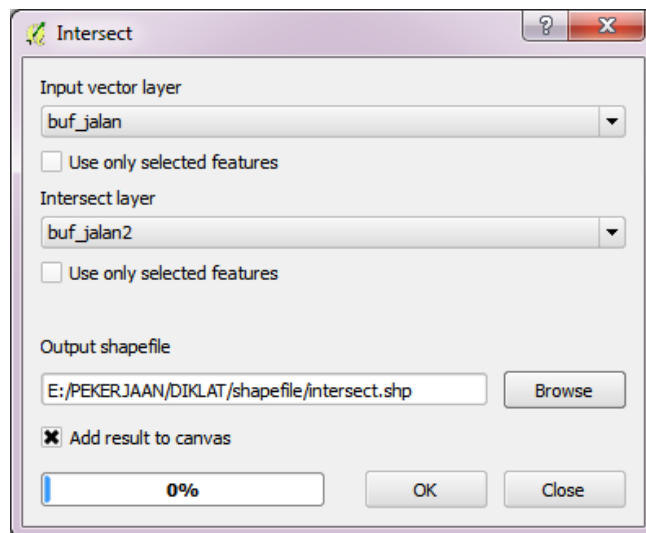
- ✓ Klik dissolve buffer results agar poligon yang dihasilkan proses ini menyatu.
- ✓ Tulis nama baru dan tentukan folder penyimpanannya pada baris Output Shapefile.
- ✓ Klik OK

Hasil proses buffer akan menghasilkan sebuah file baru yang berupa poligon. Hasil proses buffer tersebut akan menjadi seperti gambar berikut ini.

B. Intersect

Proses intersect dapat dilakukan atas dua file data poligon. Intersect akan menghasilkan perpotongan pada bagian poligon yang saling bertumpang tindih. Proses intersect dilakukan dengan cara berikut.

- ✓ Pilih menu Vector – Intersect

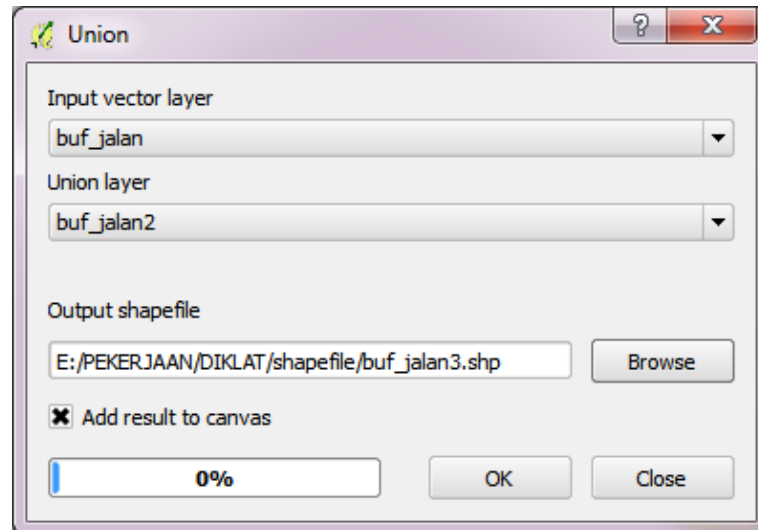


- ✓ Tentukan data vektor input dan intersect pada baris : Input vector layer dan Intersect layer
- ✓ Klik use only selected features jika diperlukan
- ✓ Tentukan folder penyimpanan hasil intersect dan namanya pada baris Output shapefile
- ✓ Klik OK

C. Union

Proses union adalah proses penggabungan dari dua data poligon. Langkah dari proses ini dilakukan dengan cara berikut.

- ✓ Pilih menu Vector – Union

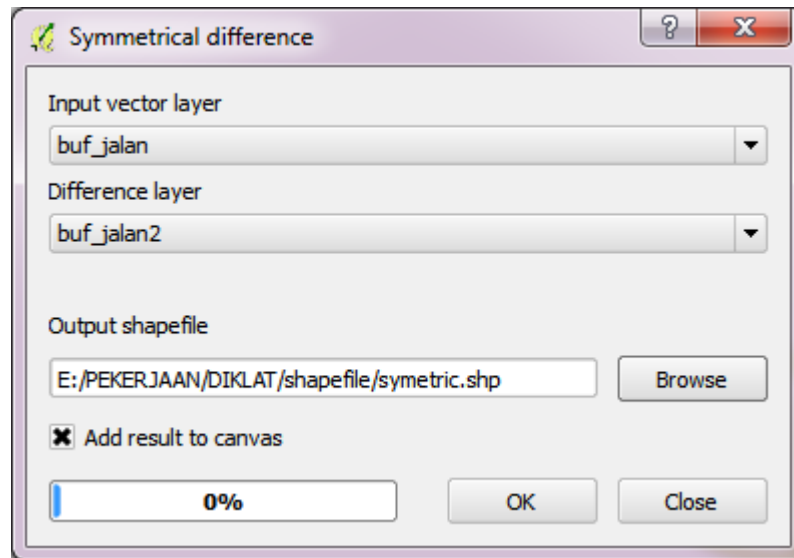


- ✓ Tentukan data input dan union pada baris Input vector layer dan Union layer
- ✓ Tentukan folder penyimpanan hasil union dan namanya pada baris Output shapefile
- ✓ Klik OK

D. Symmetrical Difference

Proses ini menghasilkan penggabungan dua buah file poligon dan mengosongkan data pada berbagai titik perpotongannya. Langkah Symmetrical Difference adalah sebagai berikut.

- ✓ Pilih menu Vector – Symmetrical Difference



- ✓ Tentukan data input dan difference pada baris Input vector layer dan Difference layer
- ✓ Tentukan folder penyimpanan hasil symetrical difference dan namanya pada baris Output shapefile
- ✓ Klik OK

E. Clip

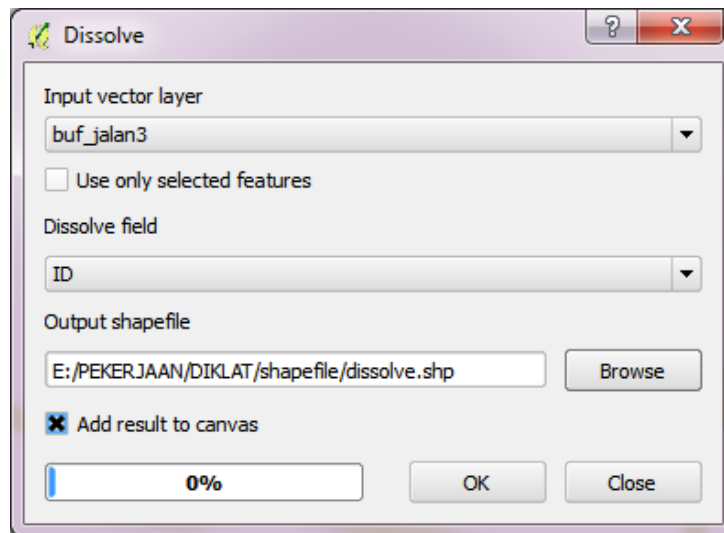
Proses clip akan memotong suatu poligon dari data spasial sumber pada lokasi yang beroverlap dengan poligon dari data yang lainnya. Secara spasial proses ini memberikan yang mirip dengan proses intersect. Perbedaannya adalah, pada proses intersect data atribut dari kedua data akan terrekam pada file baru hasil intersect. Sedangkan pada proses clip, data atribut yang terrekam hanyalah data atribut yang berasal dari data sumbernya. Proses clip dilakukan sebagai berikut.

- ✓ Pilih menu Vector – Clip
- ✓ Tentukan data input dan clip pada baris Input vector layer dan Clip Layer.
- ✓ Tentukan folder penyimpanan hasil clip dan namanya pada baris Output shapefile
- ✓ Klik OK

F. Dissolve

Proses Dissolve akan menggabungkan poligon-poligon yang saling bertumpangsusun pada sebuah file data spasial berdasar pada kesamaan informasi yang ditentukan. Proses dissolve adalah sebagai berikut.

- ✓ Pilih menu Vector – Dissolve

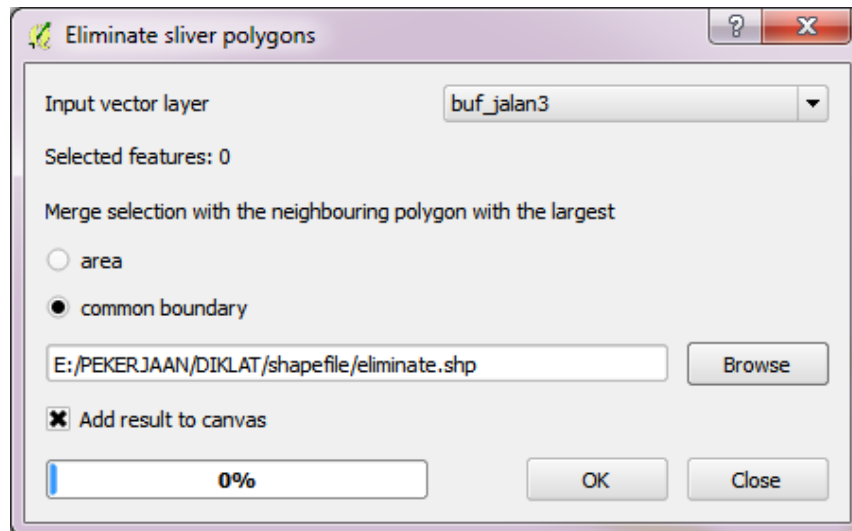


- ✓ Tentukan data input yang akan didissolve pada baris Input vector layer
- ✓ Tentukan field yang akan digunakan sebagai dasar penyatuan poligon
- ✓ Tentukan folder penyimpanan hasil dissolve dan namanya pada baris Output shapefile
- ✓ Klik OK

G. Eliminate Sliver Polygons

Proses eliminate ini akan menggabungkan poligon – poligon kecil ke dalam poligon disekitarnya yang berbatasan yang paling dominan (luas). Langkahnya adalah sebagai berikut.

- ✓ Pilih poligon yang akan dieliminate
- ✓ Pilih menu Vector – Eliminate Sliver Polygon



- ✓ Tentukan data input pada baris Input vector layer
- ✓ Pilih dasar penggabungannya : luas area atau batas yang paling dominan
- ✓ Tentukan folder penyimpanan hasil elliminate dan namanya pada baris Output shapefile
- ✓ Klik OK

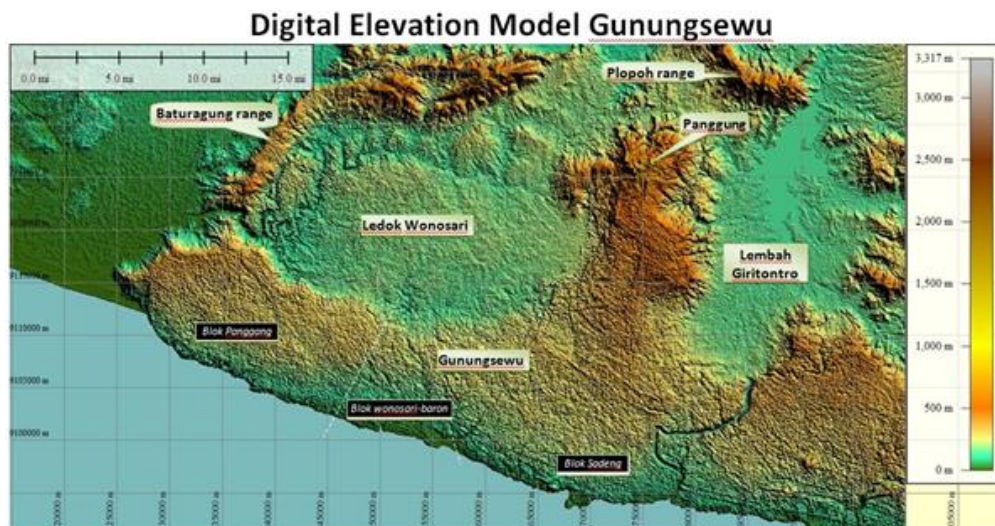
H. Latihan

1. Siapkan data jalan, sebaran rumah penduduk, dan penggunaan lahan daerah anda masing-masing
2. Carilah informasi gambaran informasi harga lahan di daerah anda.
3. Buatlah analisis spasial dan biaya sendiri pada salah satu ruas jalan yang utama pada data anda tersebut akan dilakukan perluasan jalan masing-masing 1 meter ke arah kanan dan kiri jalan tersebut.
4. Laporan dibuat dalam sebuah paper yang menyertakan deskripsi singkat wilayah penelitian, data-data, peta dan gambar, hitungan-hitungan, dan kesimpulan.

Kerjalan latihan ini secara individu dan dikerjakan selama satu minggu.

Bab 3. Analisis Morfologi Dasar

Analisis morfologi dapat dilakukan menggunakan plugin-plugin yang terdapat dalam perangkat lunak sistem informasi geografis. Analisis morfologi yang umum dilakukan adalah konturing, slope, aspect, hillshade, volume, kelurusan, visualisasi tiga dimensional dan lain-lain. Data dasar yang sering digunakan adalah data DEM (*Digital Elevation Model*) dan grid xyz. Data DEM yang banyak digunakan berasal dari turunan data penginderaan jauh seperti SRTM atau ASTER GDEM. DEM juga dapat dibentuk dari peta topografi. Perangkat lunak GIS secara umum dapat digunakan untuk proses ini.



Gambar di atas adalah visualisasi data DEM yang dihasilkan dari SRTM. Data ini menampilkan informasi elevasi suatu tempat secara spasial. Visualisasi data ini menampilkan informasi morfologi suatu lokasi.

A. Kontur

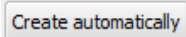
Peta kontur dapat diturunkan dari sebuah data citra SRTM ataupun ASTER GDEM. Berikut cara untuk membentuk kontur dari sebuah citra radar interferrometrik.

- ✓ Pilih Raster dari menu utama
- ✓ Pilih Extraction
- ✓ Pilih Contour

- ✓ Tentukan file input yang akan digunakan untuk membentuk kontur pada baris Input file (raster)
- ✓ Tentukan folder penyimpanan dan nama file konturnya pada baris Output file for contour lines (vector)
- ✓ Tentukan kontur interval pada baris Interval between contour lines. Misal 50 meter
- ✓ Klik pada kotak kecil di depan Attribute name agar pada file vektor kontur yang dihasilkan terbentuk data atribut ketinggian garisnya.
- ✓ Klik OK

B. Relief

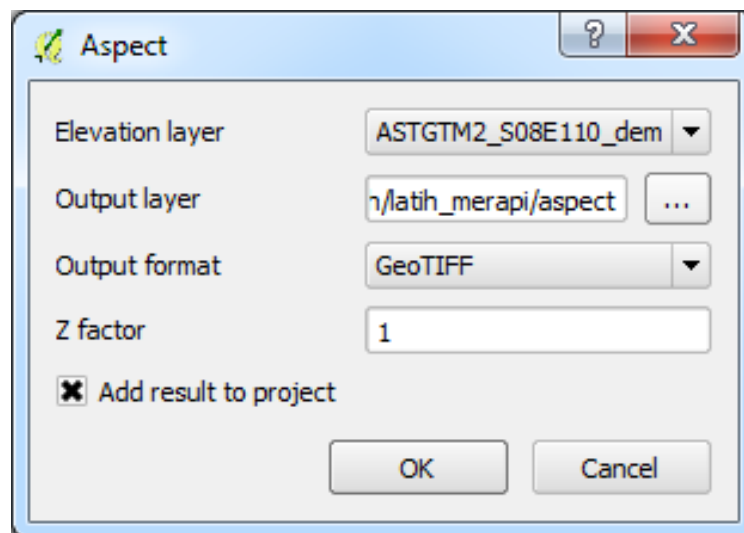
Morfologi permukaan bumi dapat digambarkan melalui pembentukan relief dari data Aster GDEM dengan menggunakan Quantum GIS. Berikut langkah pembentukan relief pada Quantum GIS ini.

- ✓ Pilih Raster dari menu utama
- ✓ Pilih Terrain Analysis
- ✓ Pilih Relief dan akan muncul jendela berikut ini
- ✓ Tentukan file Aster GDEM pada baris Elevation layer
- ✓ Tentukan folder dan nama file hasil proses relief
- ✓ Jika menginginkan simbolisasi warna kategori elevasi, buatlah beberapa kelas dengan cara
 - klik tombol + pada kotak Relief colors
 - Klik dobel pada lower bound lalu tulislah angka ketinggian minimal
 - Klik dobel pada upper bound lalu tulislah angka ketinggian maksimal
 - Klik dobel pada warna lalu pilih simbol warna yang diinginkan untuk kelas ketinggian tersebut
 - Lakukan beberapa kali langkah diatas untuk mendapatkan beberapa kelas yang lainnya.
- ✓ Pembuatan kelas tersebut juga dapat dilakukan secara otomatis dengan klik tombol Create automatically 
- ✓ Klik OK dari jendela Relief

C. Aspect

Aspect adalah proses untuk menghasilkan informasi kelas arah kemiringan lereng seperti miring ke utara, ke timur, dan seterusnya. Langkah pembentukan peta arah kemiringan lereng (aspect) ini adalah sebagai berikut.

- ✓ Pilih Raster dari menu utama
- ✓ Pilih Terrain Analysis
- ✓ Pilih Aspect dan akan muncul jendela berikut ini



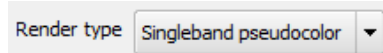
- ✓ Tentukan file Aster GDEM pada baris Elevation layer
- ✓ Tentukan folder dan nama file hasil proses aspect
- ✓ Klik OK

Proses tersebut menghasilkan citra raster yang menunjukkan kelas arah kemiringan lerengnya dalam mode greyscale. Untuk lebih memudahkan pembacaan citra tersebut, perlu dilakukan klasifikasi dan simbolisasi terhadap citra tersebut. Misal untuk kemiringan lereng dengan arah antara 0 hingga 90 derajat disimbolkan dengan warna merah, arah hadap lereng antara 90 hingga 180 derajat disimbolkan dengan warna orange, dan seterusnya.

Langkah pengkelasan dan simbolisasi warna terhadap arah hadap lereng tersebut dapat dilakukan dengan cara sebagai berikut.

- ✓ Klik double pada layer Aspect pada kotak Layers
- ✓ Akan muncul jendela Layer properties
- ✓ Klik Style pada kotak sebelah kiri

- ✓ Pada baris Render type pilih Singleband pseudocolor



- ✓ Pada kotak Generater new color map atur sebagai berikut
 - Pada baris Mode pilih Equal interval
 - Pada baris Classes ubah nilai menjadi 4
 - Pada baris Min isi dengan 0 dan Max dengan 360
 - Klik tombol Classify
- ✓ Akan muncul empat klasifikasi pada kotak sebelah kiri
- ✓ Klik dobel pada baris paling atas dibawah kolom, dan ubah nilainya menjadi 90. Lakukan pada baris-baris selanjutnya menjadi 180, 270, dan 360
- ✓ Klik dobel pada baris teratas di bawah kolom Label dan tuliskan 0 – 90. Lakukan pada baris-baris berikutnya menjadi 90 – 180, 180 – 270, dan 270 – 360.
- ✓ Klik tombol Apply
- ✓ Klik tombol OK

D. Hillshade

Hillshade adalah citra turunan yang memberikan informasi efek bayang-bayang dari suatu ketinggian tempat. Tempat yang cerah menunjukkan tempat yang berhadapan dengan arah datang sinar matahari. Sementara lokasi yang gelap merupakan tempat yang terkena bayang-bayang dari obyek-obyek tinggi di sampingnya yang berada searah dengan arah datang sinar matahari. Posisi sumber cahaya dan ketinggiannya dapat diatur pada saat pembentukan hillshade ini. Berikut adalah langkah-langkah untuk pembentukan hillshade.

- ✓ Pilih Raster dari menu utama
- ✓ Pilih Terrain Analysis
- ✓ Pilih Hillshade dan akan muncul jendela berikut ini
- ✓ Tentukan file Aster GDEM pada baris Elevation layer
- ✓ Tentukan folder dan nama file hasil proses hillshade

- ✓ Tentukan posisi arah sumber cahaya pada baris Azimuth (horizontal angle)
- ✓ Tentukan ketinggian sumber cahaya pada baris Vertical angle.
- ✓ Klik OK

E. Ruggedness Index

Ruggedness Index dapat digunakan untuk mengetahui kelas kekasaran permukaan. Berikut langkah-langkah untuk pembuatan Ruggedness Index.

- ✓ Pilih Raster dari menu utama
- ✓ Pilih Terrain Analysis
- ✓ Pilih Ruggedness Index, akan muncul jendela berikut.
- ✓ Tentukan file Aster GDEM pada baris Elevation layer
- ✓ Tentukan folder dan nama file hasil proses ruggedness index
- ✓ Klik OK

F. Latihan

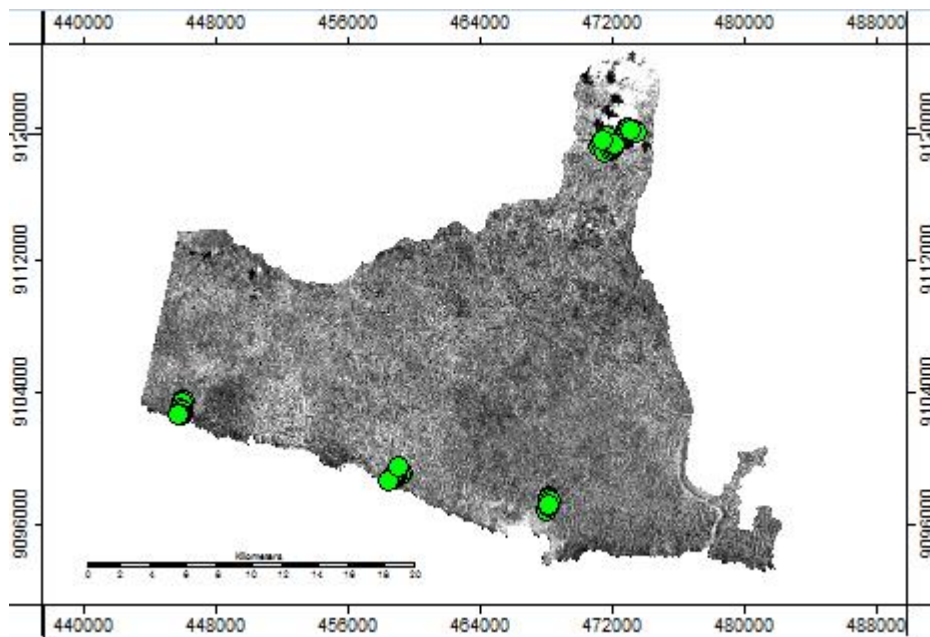
- Buatlah sebuah peta kontur dengan CI yang sesuai pada peta administrasi kabupaten masing-masing
- Buatlah visualisasi 3 dimensional dari data tersebut.
- Buat laporan tertulis yang berisi langkah pembuatan dan cetakan peta tersebut

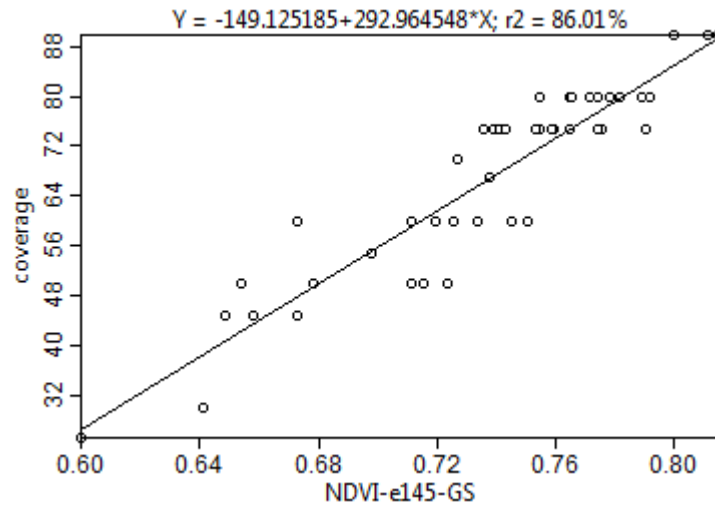
Pekerjaan dilakukan selama 1 minggu secara individual

Bab 4. Analisis Geostatistik Dasar

A. Scatterplot

Analisis geostatistik dasar digunakan sebagai metode analisis kuantitatif terhadap dua atau lebih sebaran data geografis. Data yang saling dikorelasikan dapat berupa data vektor dengan data vektor, data vektor dengan data raster, dan data raster dengan data raster. Contoh kasus adalah analisis keterkaitan antara nilai NDVI dari suatu data citra satelit dengan data kerapatan vegetasi yang diambil dari survey lapangan. Analisis ini dapat menghasilkan nilai korelasi antara kedua data tersebut. Berikut adalah satu contoh analisis geostatistik dari sebuah data citra satelit dengan data pengukuran lapangan.



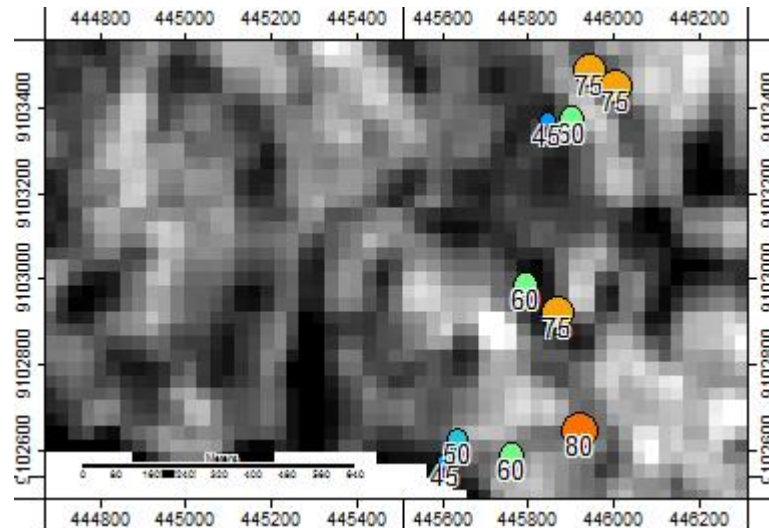


Gambar sebelah menunjukkan data citra satelit dan sebaran titik observasi dari pengukuran lapangan. Analisis korelasi terhadap kedua data tersebut menghasilkan nilai korelasi sebesar $R^2 = 86\%$. Analisis tersebut dihasilkan melalui scatterplot pada perangkat lunak GIS. Berikut langkah dalam membentuk analisis scatterplot melalui SAGA GIS.

- Panggil data raster misal NDVI
- Panggil data shapefile tipe point yang mengandung data kuantitas tertentu, misal kerapatan vegetasi
- Arahkan pointer pada nama layer NDVI
- Klik kanan lalu pilih Scatterplot
- Akan muncul jendela parameter scatterplot
- Ubah parameter pada baris Compare with.. ganti dengan isian: Point
- Ubah parameter pada baris Points dengan isian : nama shapefile point
- Ubah parameter pada baris Attribute dengan isian : data yang akan dikorelasikan
- Klik Okay

B. Visualisasi data

Data dapat ditampilkan dalam visualisasi grafis yang merepresentasikan kuantitas dan kualitas data pada suatu lokasi. Data ditampilkan dalam bentuk simbol dengan ukuran atau warna yang berbeda untuk menyatakan perbedaan tingkatan data tersebut. Hal ini dapat dilihat seperti pada gambar berikut ini.



C. Statistik Dasar

Analisis statistik dasar dapat dilakukan terhadap sebuah data vektor. Bentuk output statistik diantaranya adalah mean, standard deviasi, summary, min, max, N, unique value, range, dan median. Berikut adalah contoh langkah menampilkan output analisis statistik dasar atas suatu data vektor berupa titik yang memuat informasi tutupan lahan oleh vegetasi hasil dari pengukuran lapangan. Analisis dilakukan dengan menggunakan modul Basic Statistics pada QGIS.

- Panggil shapefile bertipe vektor. Misal : Tutupan lahan
- Tunjuk layer vektor yang akan dihitung statistik
- Pilih menu Vector
- Pilih Analysis Tools
- Pilih Basic Statistics
- Akan keluar jendela Basic Statistics
- Pastikan baris Input Vector Layer
- Tentukan baris Target Field. Misal : coverage

- Klik OK
- Hasil statistik akan muncul pada jendela Statistics Output. Contoh hasil nampak seperti gambar berikut berikut.

Mean	76.5
StdDev	6.34428877022
Sum	765.0
Min	60.0
Max	85.0
N	10.0
CV	0.0829318793493
Number of unique values	4
Range	25.0
Median	77.5

D. Nearest Neighbour Analysis

Nearest Neighbour Analysis dapat dilakukan terhadap data sejenis dengan langkah sebagai berikut.

- Panggil shapefile bertipe vektor. Misal : Tutupan lahan
- Tunjuk layer vektor yang akan dihitung statistik
- Pilih menu Vector
- Pilih Analysis Tools
- Pilih Nearest Neighbour Analysis
- Akan keluar jendela Nearest Neighbour Analysis
- Pastikan baris Input Vector Layer
- Klik OK
- Hasil analisis akan muncul pada kotak di sebelah bawah Nearest Neighbour Analysis.

Contoh hasil dapat dilihat seperti pada gambar berikut ini.

Observed mean distance	93.5996578227
Expected mean distance	66.9749953341
Nearest neighbour index	1.39753138251
N	10
Z-Score	2.40492923576

Untuk menunjukkan nilai unik yang ada pada data tersebut dapat dilakukan dengan menggunakan modul List Unique Value pada QGIS. Berikut langkah yang dapat dilakukan dalam QGIS :

- Panggil shapefile bertipe vektor. Misal : Tutupan lahan
- Tunjuk layer vektor yang akan dihitung statistik
- Pilih menu Vector
- Pilih Analysis Tools
- Pilih List Unique Values
- Akan keluar jendela isian
- Pastikan baris Input Vector Layer
- Tentukan Target Field
- Klik OK

Hasil dari proses tersebut akan menampilkan hasil seperti gambar berikut.



E. Latihan

- Potonglah sebuah data citra satelit, kemudian bentuklah citra indeks vegetasi.
- Buat plot sampel pada citra tersebut.
- Lakukan pengukuran kepadatan vegetasi pada satu plot.
- Hitung korelasi nilai spektral citra dengan nilai kepadatan vegetasi yang dihitung secara langsung di lapangan.

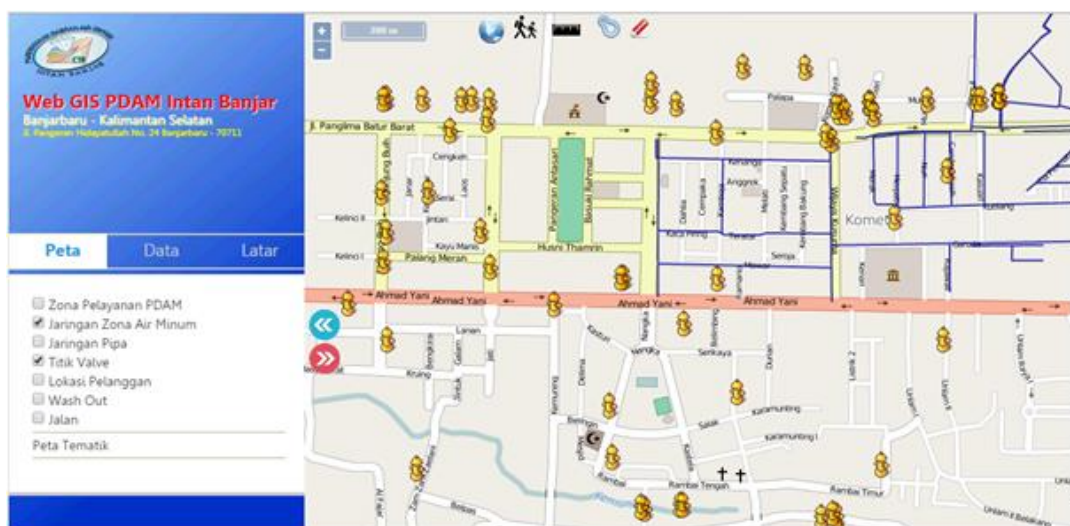
-
- e. Buat serangkaian data citra multitemporal dan berdasar hasil perhitungan korelasi tersebut, buatlah analisa terhadap kecenderungan kondisi tutupan vegetasi di masa depan.

Bab 5. Web GIS Berbasis OpenLayers

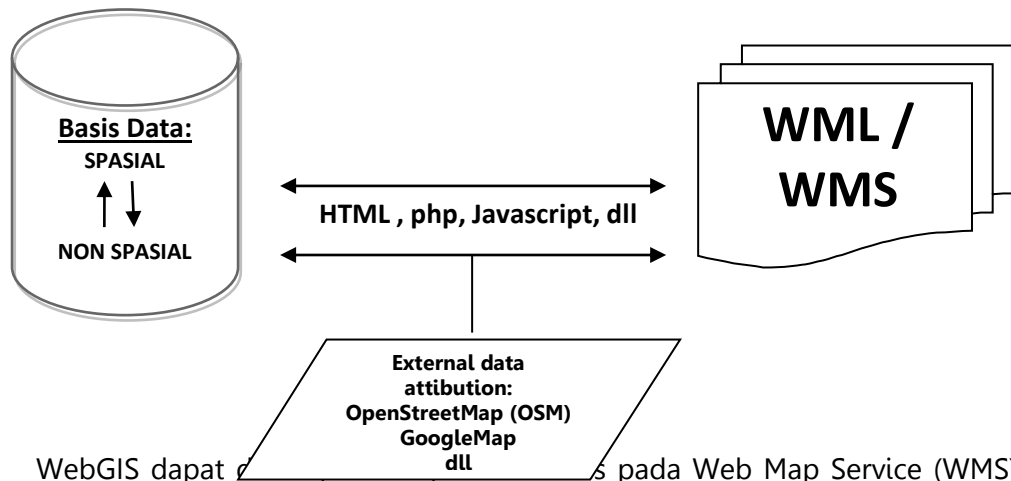
A. Web GIS

Pada masyarakat modern saat ini penggunaan data-data spasial seperti peta dan citra telah biasa dilakukan, dan tidak terbatas pada kalangan-kalangan berlatarbelakang keilmuan atau pekerjaan tertentu. Hal ini sangat didukung oleh media komunikasi canggih yang beredar di masyarakat yang sebagian besar memiliki aplikasi informasi spasial. Masyarakat modern cenderung akan menjatuhkan pilihannya pada pihak-pihak yang dapat dengan mudah ditemukan informasi dan lokasinya. Dengan demikian institusi yang memiliki kepentingan dengan masyarakat sangat tertuntut untuk memiliki sebuah sistem yang dapat mendistribusikan data-data spasialnya dengan mudah, cepat, dan tepat. Dalam kondisi seperti ini, pemanfaatan teknologi WebGIS adalah suatu solusi yang paling tepat untuk mendekatkan institusi dengan masyarakat ataupun sebaliknya.

Dalam perkembangannya saat ini, WebGIS telah memberikan banyak kemudahan dan keuntungan bagi masyarakat dan institusi penyedia layanan. WebGIS adalah suatu bentuk sistem penyebaran informasi yang paling tidak terbangun oleh adanya sebuah server dan klien, dimana server tersebut adalah server GIS dan klien adalah browser web, aplikasi desktop, atau aplikasi *mobile* (ESRI, 2016).



Selain memiliki fungsi pokok yaitu penampil data-data spasial, beberapa fungsionalitas GIS desktop sering disertakan dalam aplikasi WebGIS yang disesuaikan dengan keperluan masing-masing institusi. Dengan demikian masyarakat dapat menerima ataupun memberikan informasi spasial dengan mudah.



WebGIS dapat diakses pada Web Map Service (WMS) ataupun dengan memanfaatkan paket-paket *web mapping library* seperti OpenLayers. OpenLayers adalah salah satu paket library yang saat ini banyak diaplikasikan dalam pembangunan webGIS di seluruh dunia. OpenLayers memiliki fitur-fitur yang banyak dan dapat diaplikasikan bersamaan dengan WMS lainnya. Bahasa yang digunakan dalam pembangunan webGIS sebagian besar bertumpu pada kode-kode HTML, php, dan Javascript.

B. Persiapan

Persiapan yang perlu dilakukan dalam pembuatan webGIS berbasis OpenLayers ini adalah :

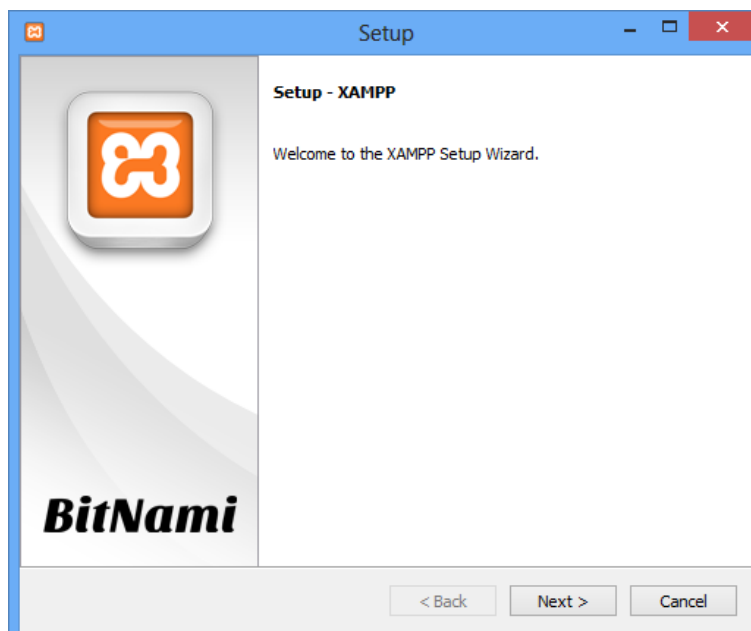
- Jaringan terkoneksi internet.
- Platform web server seperti XAMPP ataupun paket Apache.
- Sistem basis data seperti MySQL, Oracle, dan lain-lain.
- Pengolah kata (text) seperti notepad untuk penulisan kode program.
- Web Mapping Library seperti OpenLayers, atau Web Mapping Service seperti MapServer.
- GIS software seperti QGIS, Arc GIS, Arc View, dan lain-lain untuk pembuatan atau editing data spasial

- Data spasial ataupun non spasial yang akan ditampilkan dalam webGIS tersebut.

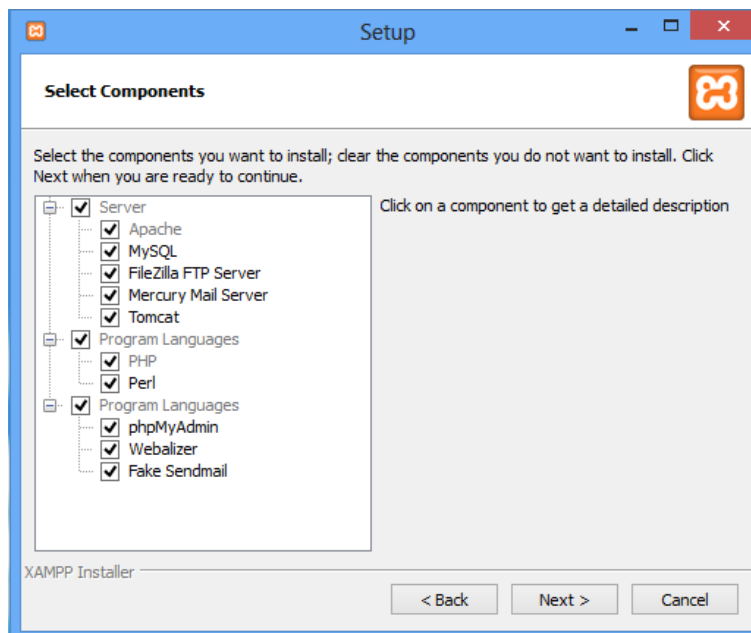
C. Instalasi Platform Web Server

Web Server perlu diinstal pada PC untuk mengoperasikan komputer sebagai server. Langkah instal dilakukan dengan cara berikut :

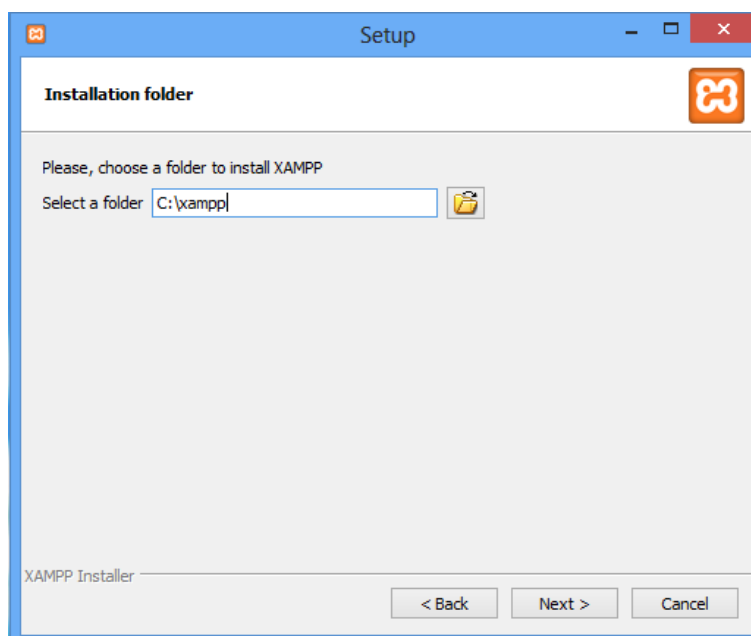
- Klik double master Xampp, kemudian akan muncul jendela Setup seperti gambar berikut



- Klik Next, akan muncul jendela berikut.



- Pilih komponen-komponen yang diperlukan, lalu tekan next. Akan muncul jendela selanjutnya.

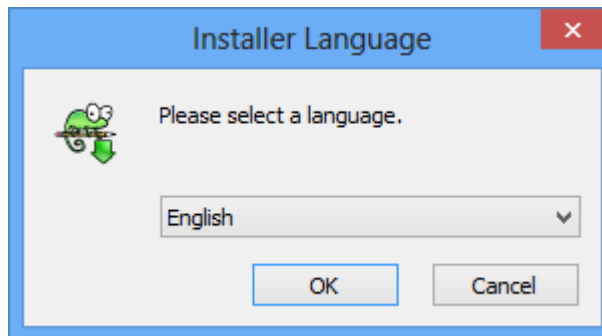


- Tentukan lokasi drive yang akan diinstal Xampp, Klik Next
- Proses instalasi akan berjalan

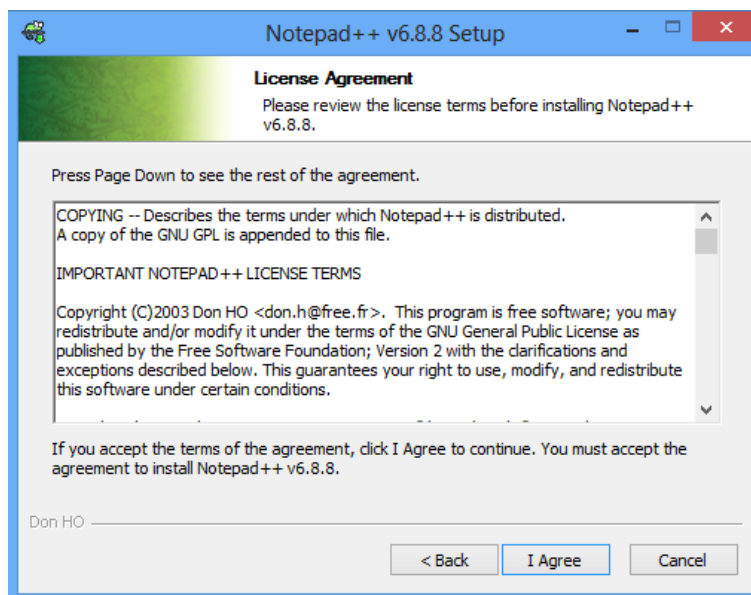
D. Instalasi Notepad

Notepad digunakan untuk menyusun kode-kode program. Notepad merupakan perangkat lunak yang cukup memadai untuk pembuatan script program. Langkah instalasi dilakukan dengan cara berikut :

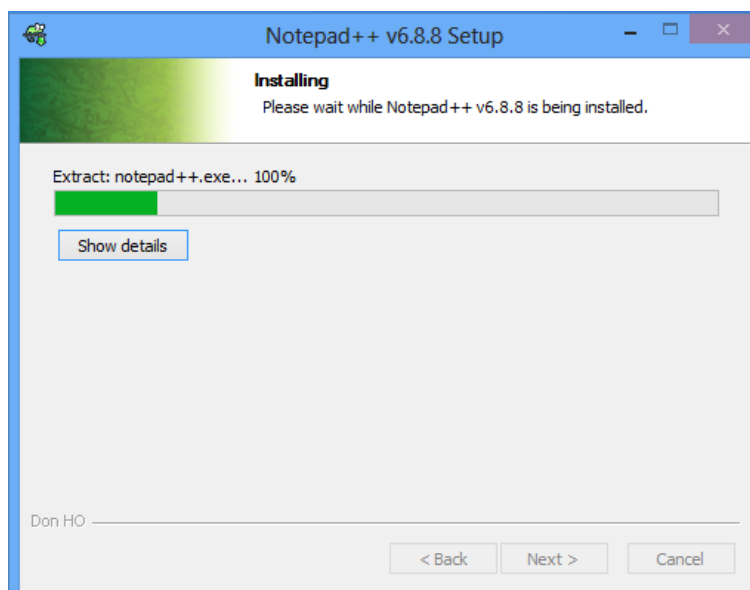
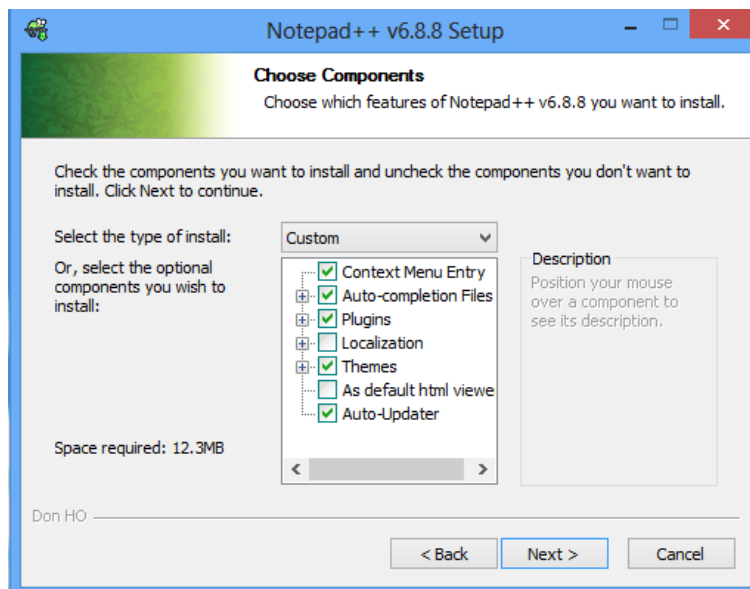
- Klik double pada file installer notepad++. Akan muncul jendela pemilihan bahasa seperti gambar berikut



- Pilih bahasa yang akan digunakan, lalu klik tombol OK.
- Klik Next pada jendela berikutnya
- Akan muncul jendela persetujuan seperti pada gambar berikut

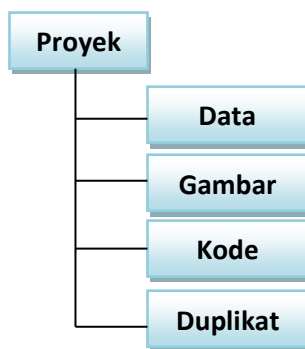


- Klik tombol I Agree dan akan muncul jendela pemilihan lokasi instal.
- Pilih lokasi drive yang akan diinstal notepad lalu klik tombol Next.
- Selanjutnya akan muncul jendela opsi komponen. Pilih komponen yang diperlukan lalu klik tombol Next.
- Pada jendela selanjutnya klik tombol Install



E. Folder Proyek

Struktur folder perlu diatur sebaik-baiknya sehingga berbagai data yang diperlukan tidak saling bercampur. Demikian pula antara data dengan kode-kode yang digunakan. Beberapa folder yang diperlukan diantaranya adalah data, gambar, kode, duplikat, dan lain-lain. Lakukan penyimpanan berkas pada folder yang sesuai. Folder-folder tersebut disatukan dalam sebuah folder proyek hingga nampak seperti diagram dibawah ini :



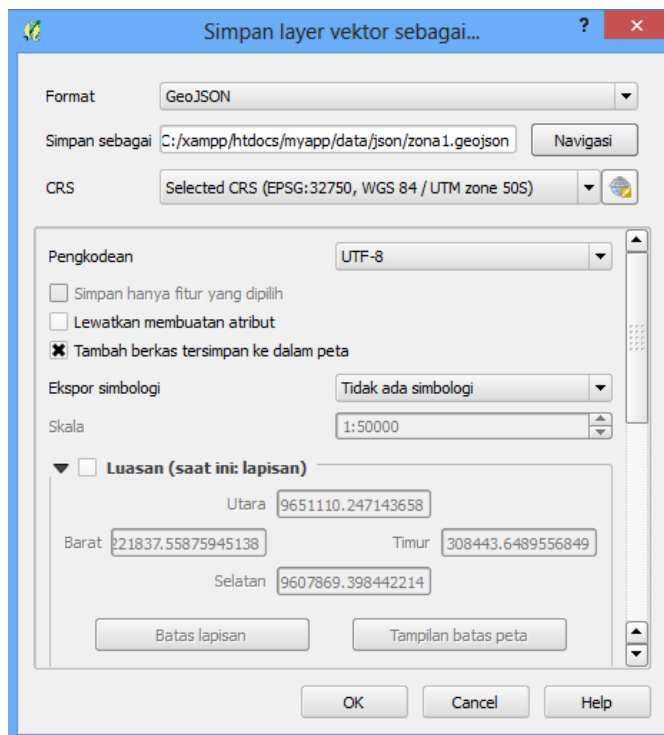
F. Data Spasial

Data spasial yang digunakan dalam WebGIS dapat berasal dari beberapa sumber yaitu data internal seperti peta-peta yang diproduksi oleh institusi dan berbagai tabel serta data external seperti data atribusi dari OSM, Bings Map, dan lain-lain.

Platform webGIS memiliki aturan-aturan yang harus dipenuhi agar data-data tersebut dapat muncul. Beberapa hal diantaranya adalah terkait dengan sistem proyeksi dan tipe data. Sebagian besar webGIS menggunakan sistem EPSG3857 yang juga digunakan oleh OpenLayers. Data spasial dengan sistem proyeksi lain harus terlebih dahulu ditransformasikan ke sistem EPSG3857 tersebut. Berikut langkah perubahan tipe data spasial sekaligus transformasi sistem proyeksinya dengan menggunakan QGIS.

- Buka QGIS Desktop
- Panggil data sehingga masuk pada daftar lapisan (layer)
- Klik file pada kotak layer tersebut kemudian klik kanan
- Pilih Simpan sebagai (Save As)

- Akan muncul jendela Simpan layer vector sebagai (Save As)
- Tentukan format baru yang dapat digunakan dalam webGIS. Misal GeoJSON dengan cara klik dropdown pada baris Format
- Tentukan folder penyimpanan dan nama file barunya dengan klik pada tombol Navigasi
- Ubah proyeksi pada baris CRS menjadi EPSG 3857
- Klik tombol OK
- File akan tersimpan menjadi file baru dengan tipe GeoJSON dengan EPSG 3857.




G. Konversi koordinat ke data vektor

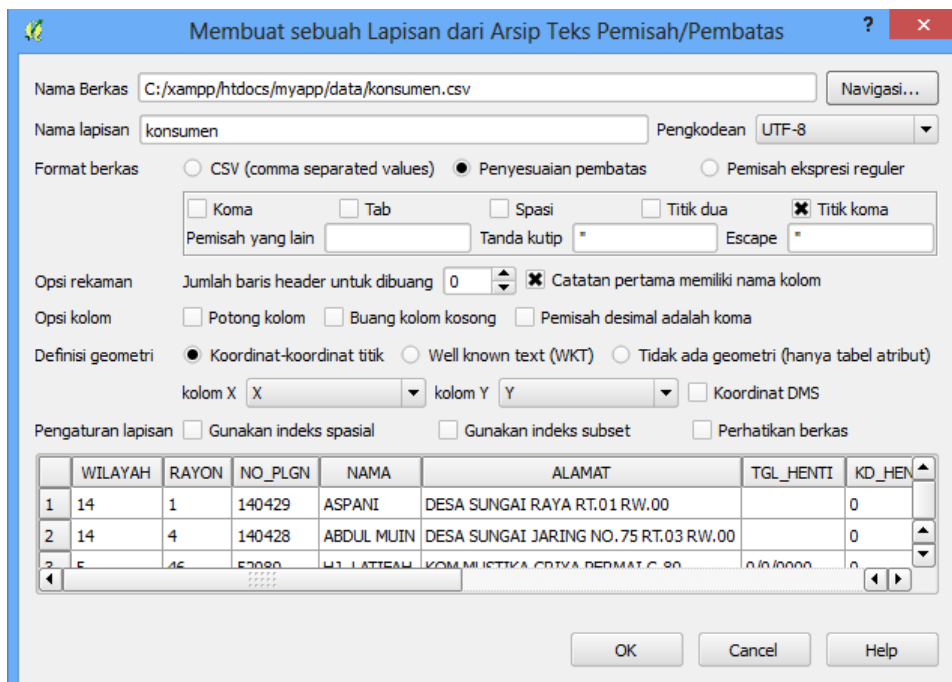
Data koordinat dapat diperoleh dari lapangan menggunakan peralatan navigasi seperti GPS. Data koordinat tersebut selanjutnya perlu dipetakan dengan terlebih dahulu mengubahnya menjadi sebuah peta titik. Berikut langkah perubahan data koordinat menjadi sebuah peta titik.

- Tuliskan daftar koordinat tersebut dengan menggunakan worksheet seperti excel dan lain-lain
- Baris pertama merupakan header dengan kolom minimal X dan Y

- Baris selanjutnya adalah angka-angka koordinat X dan Y

X	Y
261605	9616866
262518	9619646
248281	9619994
248637	9621022

- Simpan dalam format CSV (Comma Delimited).
- Panggil file CSV tersebut dalam QGIS dengan menggunakan ikon 



Membuat sebuah Lapisan dari Arsip Teks Pemisah/Pembatas

Nama Berkas: C:/xampp/htdocs/myapp/data/konsumen.csv

Nama lapisan: konsumen Pengkodean: UTF-8

Format berkas: CSV (comma separated values) Penyesuaian pembatas Pemisah ekspresi reguler

Koma Tab Spasi Titik dua Titik koma

Pemisah yang lain: Tanda kutip: " Escape: "

Opsi rekaman: Jumlah baris header untuk dibuang: 0 Catatan pertama memiliki nama kolom

Opsi kolom: Potong kolom Buang kolom kosong Pemisah desimal adalah koma

Definisi geometri: Koordinat-koordinat titik Well known text (WKT) Tidak ada geometri (hanya tabel atribut)

kolom X: X kolom Y: Y Koordinat DMS

Pengaturan lapisan: Gunakan indeks spasial Gunakan indeks subset Perhatikan berkas

	WILAYAH	RAYON	NO_PLGN	NAMA	ALAMAT	TGL_HENTI	KD_HENI
1	14	1	140429	ASPANI	DESA SUNGAI RAYA RT.01 RW.00		0
2	14	4	140428	ABDUL MUJIN	DESA SUNGAI JARING NO.75 RT.03 RW.00		0
3	5	46	50080	HJ. LATIFAH	KOM. MUKTIKA CRTVA PERMAT. C. 80	0/0/0000	0

OK Cancel Help

- Tentukan nama file dengan klik Navigasi pada baris Nama Berkas
- Buat nama layer baru dengan mengisi pada baris Nama Lapisan
- Tentukan kolom X dan kolom Y
- Klik OK

Hasil proses konversi tersebut menghasilkan layer spasial bertipe titik. Lakukan perubahan layer tersebut menjadi shapefile dengan menggunakan langkah seperti bagian di atas (Save As / Simpan Sebagai) dengan menggunakan QGIS.

H. Menyusun kode

WebGIS dituliskan dalam kode-kode HTML/php atau script Javascript secara terpisah ataupun bersamaan. Skema kode-kode tersebut secara opsional dapat berbentuk seperti dibawah ini.

```
<?php
....
<html>
  <head>
    ....
      <script>
        ....
      </script>
  </head>

  <body>
    ....
      <script>
        ....
      </script>
  </body>

</html>
?>
```

Contoh kode webGIS sederhana dapat dilihat sebagai berikut :

```
<html>

<head>
  <title>Program ke 1</title>
  <link rel="stylesheet" href="http://openlayers.org/en/v3.13.0/css/ol.css" type="text/css">
  <script src="http://openlayers.org/en/v3.13.0/build/ol.js"></script>
</head>

<body>
  <div id="map" class="map"></div>

  <script>
    var map = new ol.Map
    ({
      target: 'map',
      layers: [new ol.layer.Tile({source: new ol.source.OSM()})],
      view: new ol.View({center: ol.proj.transform([114.8376,-3.4389], 'EPSG:4326',
'EPSG:3857'),zoom: 10})
    });
  </script>

</body>
```

</html>

Kode tersebut akan menampilkan sebuah data eksternal atribusi yang berasal dari OpenStreetMap.org. Eksekusi dari kode tersebut akan nampak seperti pada gambar berikut :



Keseluruhan kode tersebut terdiri dari beberapa bagian penting yaitu :

1. Deklarasi library yang tuliskan pada bagian head (teks warna merah)
2. Deklarasi obyek Map (teks warna hijau)
3. Uraian modul OpenLayers yang akan diberikan pada obyek Map (teks warna biru).

Pada bagian script modul map OpenLayers terdapat pula tiga bagian penting yang harus ada agar layer dapat muncul pada obyek Map. Bagian tersebut adalah parameter **target**, **layers**, dan **view**.

```
var map = new ol.Map
({
  target: 'map',
  layers: [ new ol.layer.Tile( {source: new ol.source.OSM()} ) ],
  view: new ol.View({center: ol.proj.transform([114.8376,-3.4389], 'EPSG:4326', 'EPSG:3857'),zoom: 10})
});
```

Modul map dibentuk dengan kode baku `new ol.Map ({});` dan disimpan dalam sebuah variabel. Modul ini selanjutnya diisi dengan deklarasi parameter dari modul map yang pokok yaitu target, layers, dan view.

- Parameter target : untuk mengarahkan hasil eksekusi modul Map

- Parameter `layers` : untuk mendeklarasikan layer-layer yang akan ditampilkan
- Parameter `view` : untuk menentukan titik tengah (center) dari display layer, sistem proyeksi yang digunakan, serta perbesaran tampilan defaultnya.

Kode tersebut dapat dimodifikasi dengan menghasilkan tampilan yang sama seperti dengan memasukkan beberapa bagian script menjadi sebuah variabel. Berikut adalah contoh dari modifikasi tersebut.

```
<html>
<head>
  <title>Program ke 1</title>
  <link rel="stylesheet" href="http://openlayers.org/en/v3.13.0/css/ol.css" type="text/css">
  <script src="http://openlayers.org/en/v3.13.0/build/ol.js"></script>
</head>
<body>
  <div id="map" class="map"></div>

  <script>
    var layerku = new ol.layer.Tile({source: new ol.source.OSM()});

    var map = new ol.Map
    ({
      target: 'map',
      layers: [layerku],
      view: new ol.View({center: ol.proj.transform([114.8376,-3.4389], 'EPSG:4326',
'EPSG:3857'),zoom: 10})
    });
  </script>
</body>
</html>
```

Parameter **view** terdiri dari paling tidak dua bagian penting yang harus ada, yaitu **center**, dan **zoom**. Center digunakan untuk mengontrol titik pusat tampilan map, sedangkan zoom digunakan untuk mengontrol perbesaran default dari map tersebut. Modifikasi lain dari kode di atas yang dilakukan terhadap parameter view dapat dilihat seperti contoh berikut

```
<script>
var layerku = new ol.layer.Tile({source: new ol.source.OSM()});

var proyeksi_awal = 'EPSG:4326';
var proyeksi_akhir = 'EPSG:3857';
var X = 114.8376;
var Y = -3.4389;
var pusat = ol.proj.transform([X,Y], proyeksi_awal, proyeksi_akhir);
```

```

var zooms = 10;

var views = new ol.View
({
  center: pusat,
  zoom: zooms
});
var map = new ol.Map
({
  target: 'map',
  layers: [layerku],
  view: views
});
</script>

```

Hasil eksekusi dari kode di atas jika tidak terjadi kesalahan apapun, akan memberikan hasil tampilan yang sama dengan kode terdahulu.

I. Memanggil data vektor

WebGIS memiliki fungsi pokok untuk menampilkan data spasial yang salah satunya berupa peta. Peta sering berbentuk data vektor seperti shapefile. Data vektor yang akan ditampilkan harus dipanggil dalam script dan disimpan sebagai sebuah layer. Perintah pokok untuk memanggil data vektor tersebut adalah sebagai berikut :

new ol.layer.Vector ({ *source* })

Parameter minimal yang harus ada pada deklarasi layer yaitu **source** . Parameter ini berfungsi untuk menetapkan file yang akan dipanggil, format file, serta lokasinya. Source dideklarasikan dengan perintah : **new ol.source.Vector({ *url data vektor* , *format data* })**.

Variabel layer tersebut selanjutnya dimasukkan pada array layer modul map OpenLayers. Untuk lebih jelas, deklarasi layer data vektor dapat dilihat pada contoh berikut ini.

```

vector1 = new ol.layer.Vector
({
  source: new ol.source.Vector({url: './data/json/zona.geojson',format: new
ol.format.GeoJSON()});
});

```

Contoh lengkap dalam sebuah kode HTML adalah sebagai berikut :

```
<html>
<head>
  <link rel="stylesheet" href="http://openlayers.org/en/v3.13.0/css/ol.css"
type="text/css">
  <script src="http://openlayers.org/en/v3.13.0/build/ol.js"> </script>
</head>

<body>
  <div id="map" class="map"> </div>

  <script>

    var vector1 = new ol.layer.Vector
    ({
      source: new ol.source.Vector({url: './data/json/zona.geojson', format: new
ol.format.GeoJSON()})
    });
    var layerku = new ol.layer.Tile({source: new ol.source.OSM()});
    var proyeksi_awal = 'EPSG:4326';
    var proyeksi_akhir = 'EPSG:3857';
    var X = 114.8376;
    var Y = -3.4389;
    var pusat = ol.proj.transform([X,Y], proyeksi_awal, proyeksi_akhir);
    var zooms = 10;

    var views = new ol.View
    ({
      center:pusat,
      zoom: zooms
    });

    var map = new ol.Map
    ({
      target: 'map',
      layers: [layerku, vector1],
      view: views
    });

  </script>

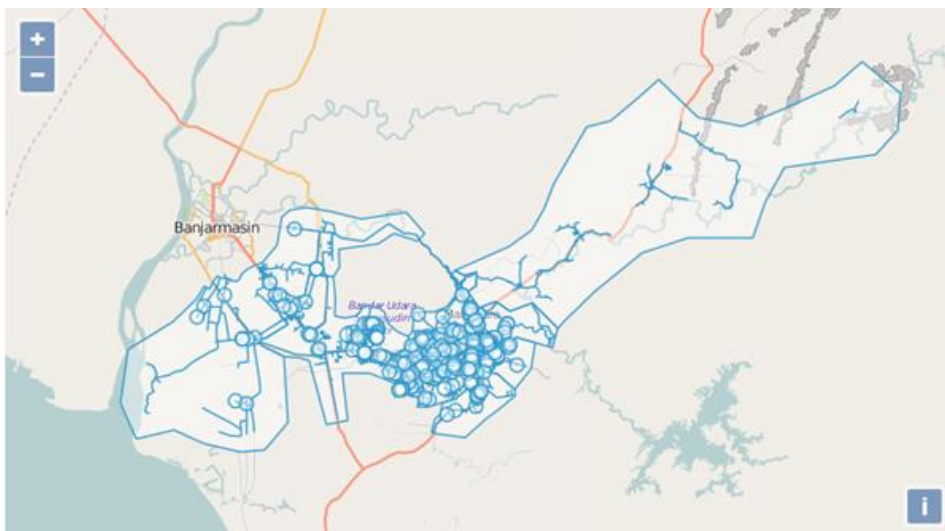
</body>
</html>
```

Eksekusi dari kode HTML tersebut di atas akan menampilkan data vektor beserta data atribusi OSM seperti dapat dilihat pada gambar berikut



LATIHAN :

Coba lakukan penambahan beberapa layer vektor lain ke dalam tampilan peta di atas dengan menggunakan data yang ada.



J. Simbol data vektor

Simbol pada sebuah peta memiliki peran yang sangat penting. Deklarasi simbol pada modul OpenLayers dituliskan di dalam deklarasi layer dengan menambah parameter **style**. Parameter style dideklarasikan dengan perintah **new ol.style.Style ({...})**.

1. Simbol data bertipe poligon

Dalam parameter tersebut terdapat sub parameter **fill** dan **stroke**. Sub parameter fill digunakan untuk menentukan warna isi, sedangkan sub parameter stroke digunakan untuk menentukan warna garis tepi.

```
style : new ol.style.Style ({ fill , stroke });
```

Sub parameter **fill** ditetapkan dengan menggunakan perintah **new ol.style.Fill({...})**. sub parameter **stroke** ditetapkan dengan menggunakan perintah **new ol.style.Stroke({...})**. Dalam sub parameter tersebut ditetapkan nilai warna yang akan digunakan dengan perintah berikut : **color: "rgba(nilai R, nilai G, nilai B, opasiti)"**.

```
fill: new ol.style.Fill({color: 'rgba(255,255,0,1)'}),      isi   berwarna
kuning
stroke: new ol.style.Stroke({color: 'rgba(0, 0, 255, 1)'}),  garis  tepi
berwarna biru
```

Aplikasi dalam deklarasi layer dapat dilihat pada kode berikut :

```
<html>
<head>
  <link rel="stylesheet" href="http://openlayers.org/en/v3.13.0/css/ol.css"
type="text/css">
  <script src="http://openlayers.org/en/v3.13.0/build/ol.js"> </script>
</head>
<body>
<div id="map" class="map"> </div>
<script>

var vector1 = new ol.layer.Vector
({
source: new ol.source.Vector({url: './data/json/zona.geojson',format: new
ol.format.GeoJSON()}),
style: new ol.style.Style
  ({
    fill: new ol.style.Fill({color: 'rgba(255,255,0,1)'}),
    stroke: new ol.style.Stroke({color: 'rgba(0, 0, 255, 1)'}),
  })
});
```

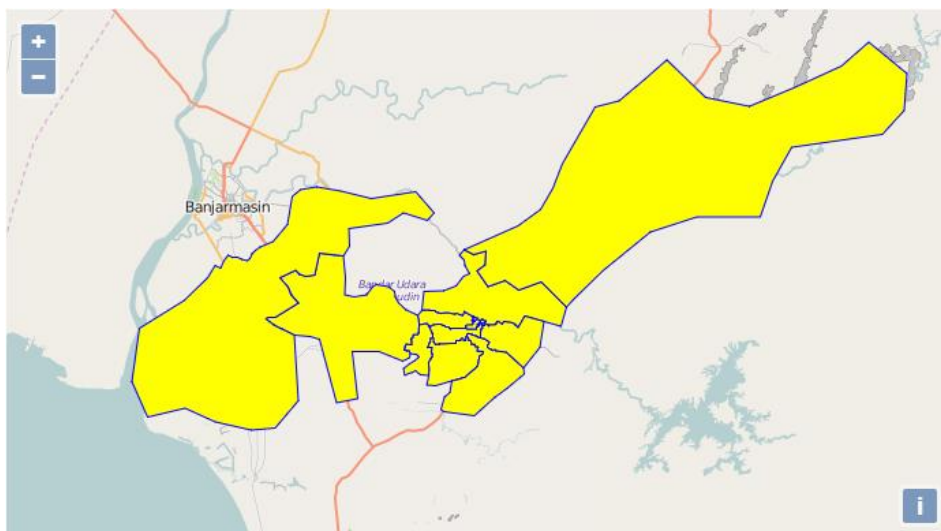
```

var layerku = new ol.layer.Tile({source: new ol.source.OSM()});
var proyeksi_awal = 'EPSG:4326';
var proyeksi_akhir = 'EPSG:3857';
var X = 114.8376;
var Y = -3.4389;
var pusat = ol.proj.transform([X,Y], proyeksi_awal, proyeksi_akhir);
var zooms = 10;

var views = new ol.View
  ({
    center:pusat,
    zoom: zooms
  });

var map = new ol.Map
  ({
    target: 'map',
    layers: [layerku, vector1],
    view: views
  });
</script>
</body>
</html>

```



2. Simbol data bertipe garis

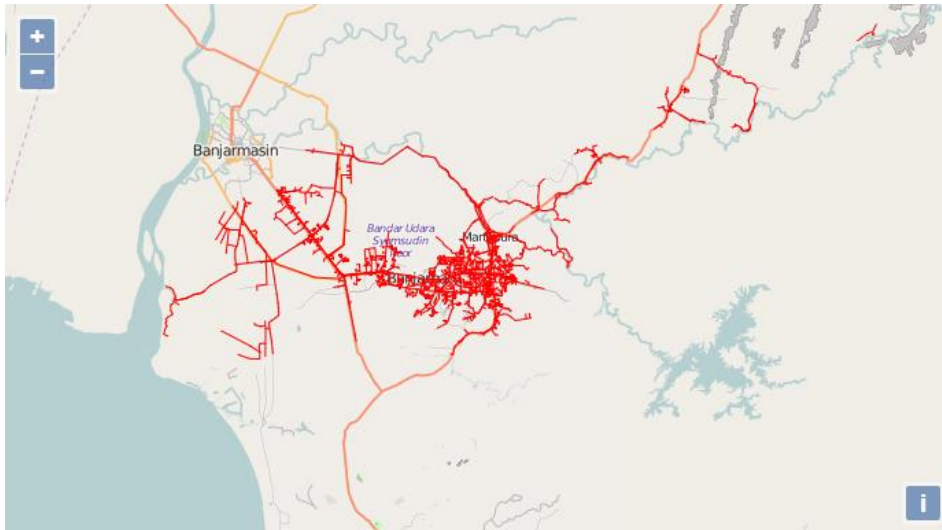
Pada layer dengan data vektor bertipe garis, pengaturan warna style hanya menggunakan sub parameter stroke saja.

```
var vector2 = new ol.layer.Vector
```

```

({
  source: new ol.source.Vector({url: './data/json/pipa.geojson',format: new
  ol.format.GeoJSON()}),
  style: new ol.style.Style
    (
      (
        stroke: new ol.style.Stroke({color: 'rgba(255, 0, 0, 1)}'),
      )
    )
});

```



3. Simbol data bertipe point

Pada layer dengan data vektor bertipe point, memiliki sub parameter yang berbeda yaitu dengan sub parameter **image**. Sub parameter tersebut memiliki sub parameter **radius**, **fill**, dan **stroke**. Radius digunakan untuk menentukan besar point tersebut, sedangkan fill dan stroke memiliki fungsi yang sama dengan simbol data bertipe poligon.

```

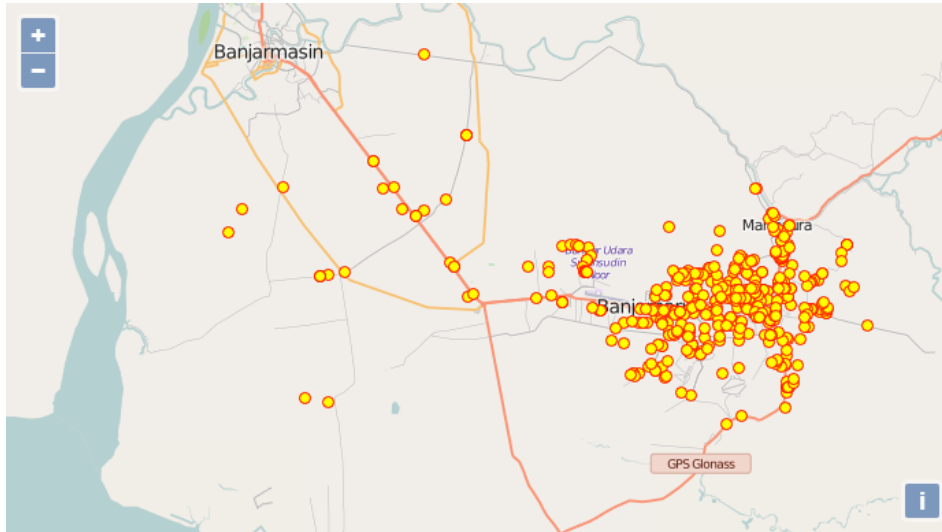
var vector3 = new ol.layer.Vector
({
  source: new ol.source.Vector({url: './data/json/valve.geojson',format: new
  ol.format.GeoJSON()}),
  style: new ol.style.Style
    (
      (
        Image : new ol.style.Circle
          (
            radius : 5,
            fill: new ol.style.Fill({color: 'rgba(255,255,0,1)}'),

```

```

stroke: new ol.style.Stroke({color: 'rgba(255, 0, 0, 1)}'),
})
})
});

```



4. Kategori simbol

Fenomena spasial sejenis dapat ditampilkan dalam bentuk kategori atau klasifikasi. Sebagai contoh, peta zona dapat ditampilkan dengan warna yang berbeda berdasar nama zona, dan lain-lain.

Kategori simbol dibentuk dengan mendeklarasikan simbol dalam sebuah fungsi. Fungsi tersebut selanjutnya dimasukkan ke dalam parameter style dalam deklarasi layer. Berikut contoh listing kode fungsi kategori simbol.

```

var simbolpoligon = function()
{
  return function(feature)
  {
    var style1 = new ol.style.Style
      ({
        stroke: new ol.style.Stroke({color: 'blue',width: 1}),
        fill: new ol.style.Fill({color: 'rgba(255,0,0,0.5)}'),
      });

    var style2 = new ol.style.Style
      ({

```

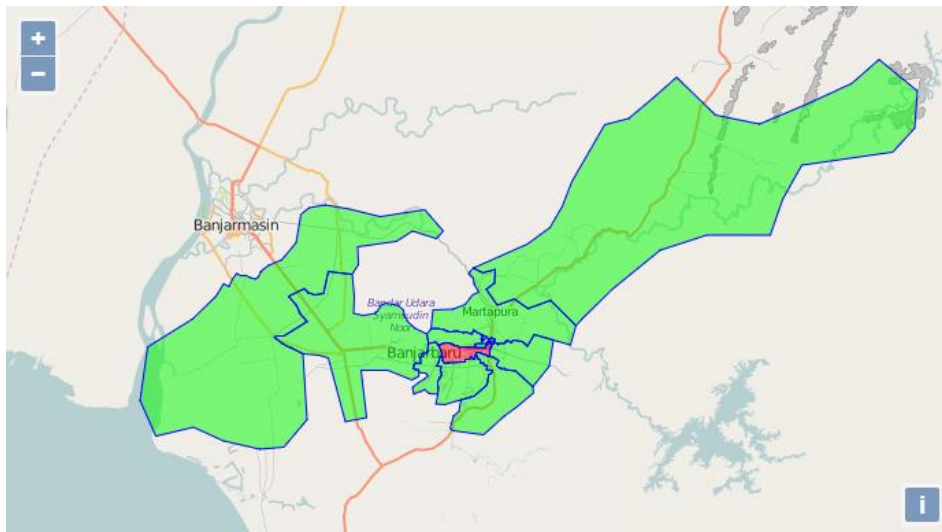
```

        stroke: new ol.style.Stroke({color: 'blue',width: 1}),
        fill: new ol.style.Fill({color: 'rgba(0,255,0,0.5)}),
    });

    if (feature.get('ZONA') == '1') {return [style1];}
    else {return [style2];}
};

};
var vector1 = new ol.layer.Vector
({
source: new ol.source.Vector({url: './data/json/zona.geojson',format: new
ol.format.GeoJSON()}),
style: simbolpoligon()
});

```



Deklarasi kategori simbol untuk data vektor bertipe garis memiliki bentuk yang hampir sama dengan data vektor bertipe poligon. Berikut deklarasi fungsi :

```

var simbolgaris = function()
{
    return function(feature)
    {
        var style1 = new ol.style.Style
        ({
            stroke: new ol.style.Stroke({color: 'blue',width: 1}),
        });
    }
}

```

```

var style2 = new ol.style.Style
    ({
        stroke: new ol.style.Stroke({color: 'blue',width: 1}),
    });
if (feature.get('DIAMETER') == '25') {return [style1];}
else {return [style2];}
};
};
var vector2 = new ol.layer.Vector
({
source: new ol.source.Vector({url: './data/json/pipa.geojson',format: new
ol.format.GeoJSON()}),
style: simbolgaris()
});

```



Untuk data bertipe point deklarasi fungsi dibentuk dengan cara sebagai berikut :

```

var simboltitik = function()
{
    return function(feature)
    {
        var style1 = new ol.style.Style
            ({
                image : new ol.style.Circle
                    ({
                        stroke: new ol.style.Stroke({color: 'blue',width: 1}),
                        fill: new ol.style.Fill({color: 'rgba(0,255,0,0.5)'}),
                    })
            });
    };
};

```

```

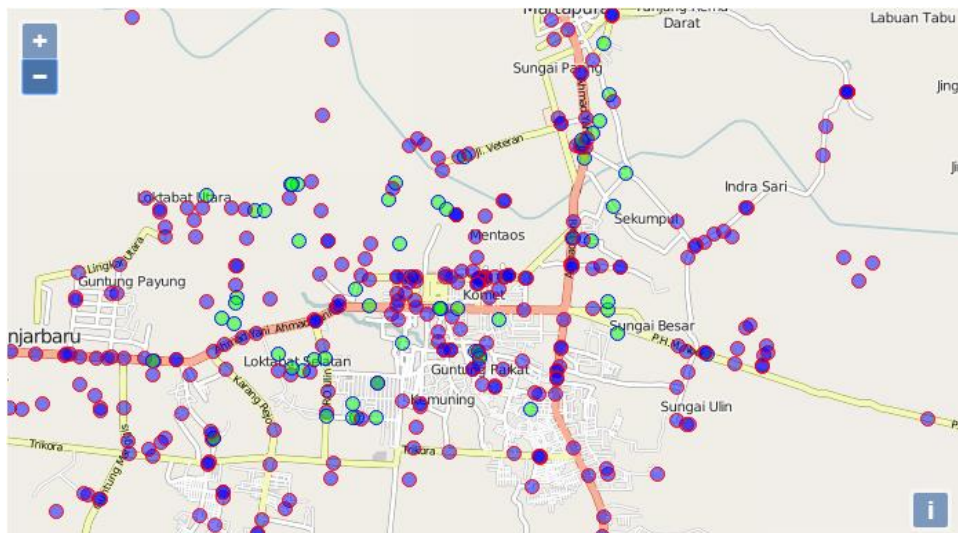
var style2 = new ol.style.Style
({
  image : new ol.style.Circle

    ({
      stroke: new ol.style.Stroke({color: 'red',width: 1}),
      fill: new ol.style.Fill({color: 'rgba(0,0,255,0.5)}),
    })
});

if (feature.get('DIAMETER') == '25') {return [style1];}
else {return [style2];}
};

var vector2 = new ol.layer.Vector
({
  source: new ol.source.Vector({url: './data/json/valve.geojson',format: new
ol.format.GeoJSON()}),
  style: simboltitik()
});

```



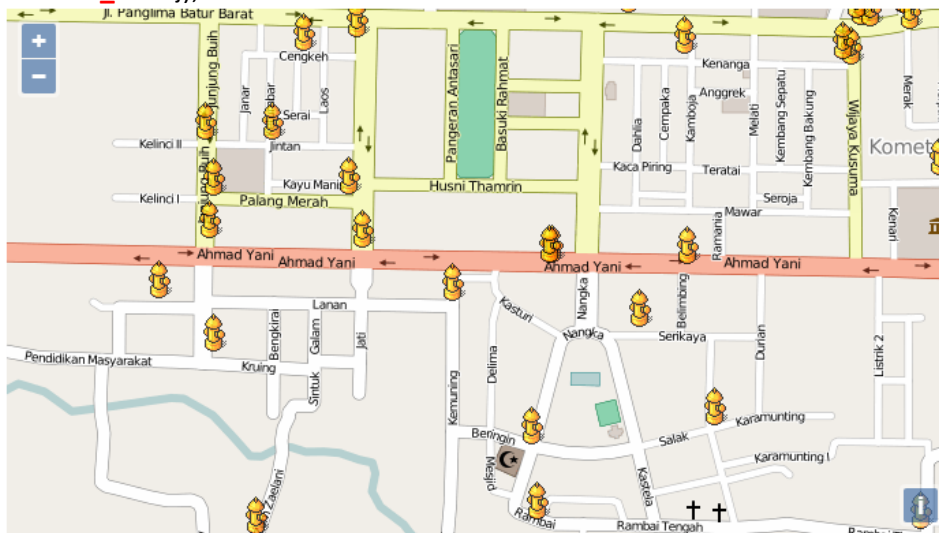
K. Simbol ikon

Simbol pada data berbentuk titik dapat dibentuk dengan menggunakan ikon atau image. Ikon atau gambar dideklarasikan sebagai variabel dan selanjutnya dimasukkan sebagai nilai dari parameter style dalam deklarasi layer. Simbol ikon

dibentuk dengan perintah pokok : **new ol.style.Icon ({ ... })** . Berikut contoh deklarasi simbol ikon.

```
var simbol_ikon = new ol.style.Style
({
  image: new ol.style.Icon
    ({
      anchor: [0.5, 1.1],
      anchorXUnits: 'fraction',
      anchorYUnits: 'fraction',
      opacity: 0.75,
      src: 'images/valve1.png'
    })
});

var vector1 = new ol.layer.Vector
({
  source: new ol.source.Vector({url: './data/json/valve.geojson',format: new
ol.format.GeoJSON()}),
  style: simbol_ikon });
```



L. Label

Label merupakan simbol teks yang dimunculkan dari sebuah fitur data. Sebagai contoh label adalah nama wilayah, nama jalan, kode, dan lain-lain. Label fitur ditampilkan dengan melalui beberapa langkah:

- Buat sebuah fungsi untuk menentukan teks label yang akan dimunculkan
- Teks label diubah menjadi fitur teks OpenLayers melalui sebuah fungsi
- Tangkap teks label dengan sebuah fungsi. Fitur teks openlayers digunakan sebagai dasar penangkap teks label dari data vektor.

- Masukkan fungsi teks label ke dalam parameter style dalam deklarasi layer

Berikut contoh kode untuk menampilkan label tersebut

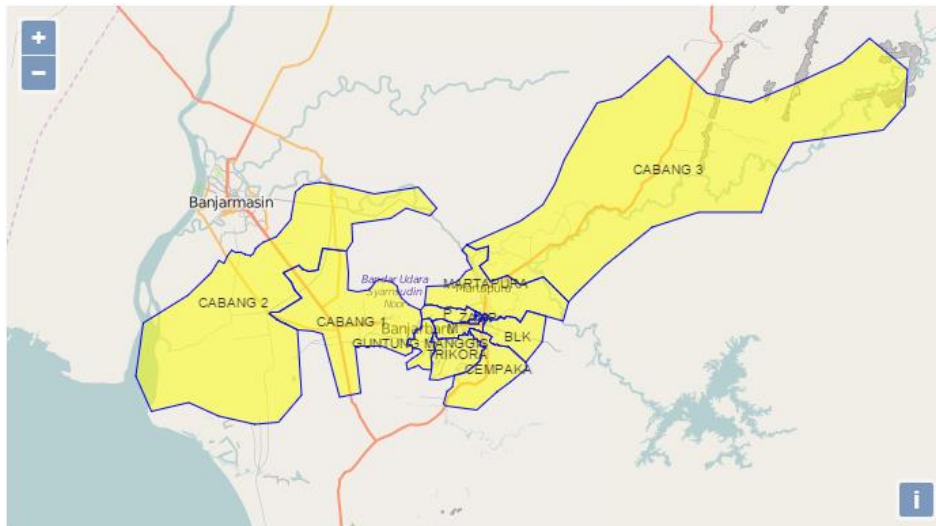
```

var tekslabel = function(feature) {var text = feature.get('NAMA ZONA');return text;};
var buatteks = function(feature)
    {return new ol.style.Text({text: tekslabel (feature)}});

var labelku = function()
{ return function(feature)
    {
        var style1 = new ol.style.Style
            ({
                stroke: new ol.style.Stroke({color: 'blue',width: 1}),
                fill: new ol.style.Fill({color: 'rgba(255,255,0,0.5)'}),
                text: buatteks (feature)
            });
        if (feature) {return [style1];}
    };
};

var vector1 = new ol.layer.Vector
({
    source: new ol.source.Vector({url: './data/json/zona.geojson',format: new
ol.format.GeoJSON()}),
    style: labelku()
});

```

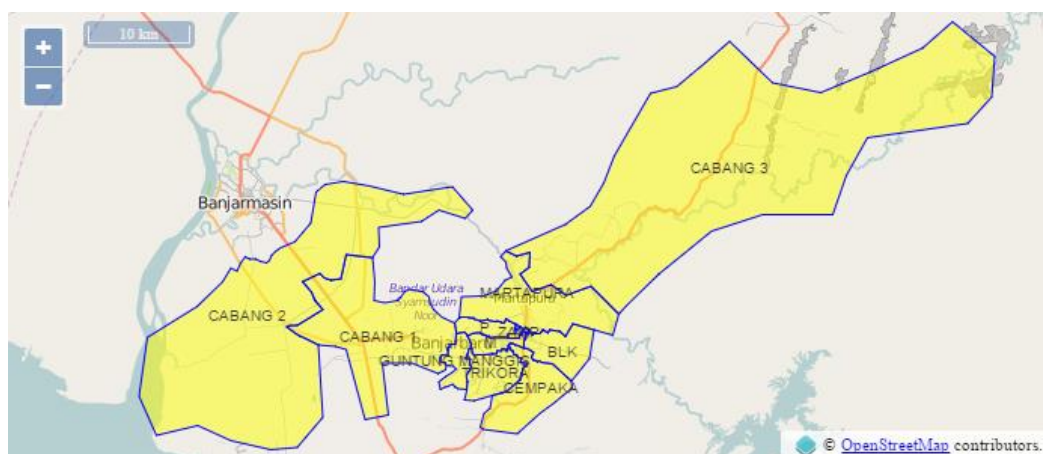


M. Kontrol skala

Fitur skala termasuk sebagai Control OpenLayers. Visualisasi kontrol diwujudkan dalam sebuah obyek div, sehingga terlebih dahulu perlu didefinisikan sebuah div untuk menampilkan kontrol tersebut.

```
<html>
<head></head>
<body>
<div id='skala'></div>
....
<script>
var skala = new ol.control.ScaleLine({target:
document.getElementById('skala')});
.....
</body>
</html>
```

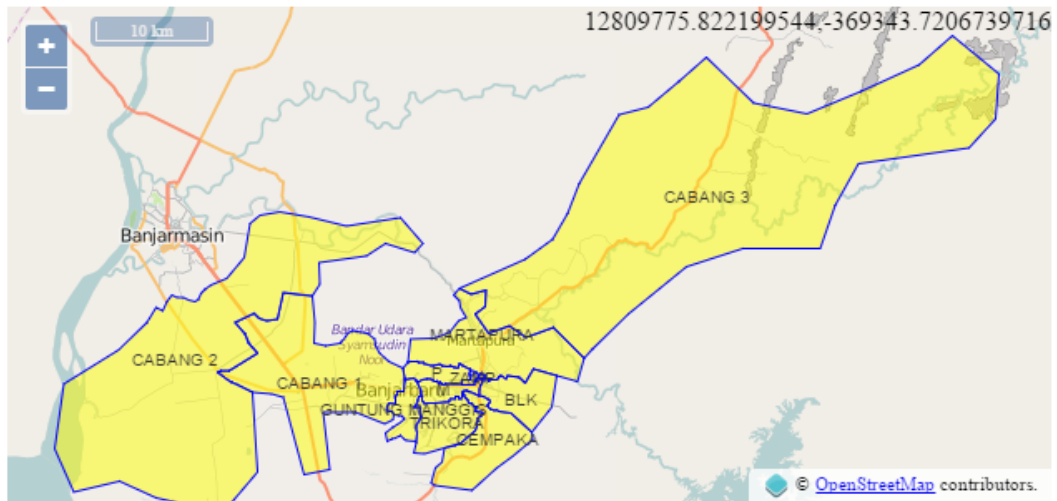
Seperti pada listing kode di atas, fitur kontrol diwujudkan pada sebuah div html. Isi dari fitur kontrol dideklarasikan dalam sebuah variabel. Kontrol skala dibentuk dengan menggunakan perintah pokok **new ol.control.Scaleline({...})**. Variabel kontrol selanjutnya dimasukkan pada parameter **controls** pada fitur map sebagai extent.



N. Posisi mouse

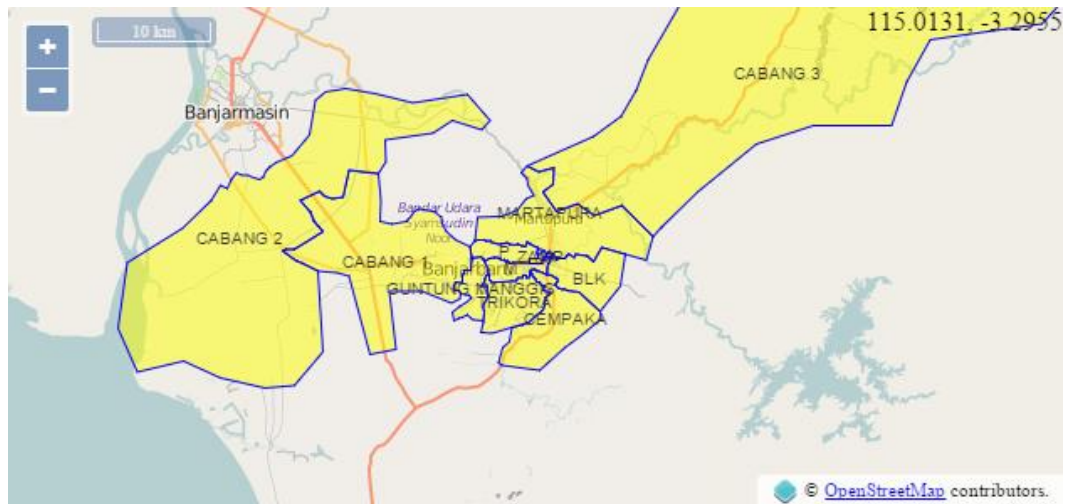
Posisi mouse sangat bermanfaat dalam webGIS. Aplikasi fitur ini mirip dengan cara pemasangan fitur skala di atas. Fitur ini akan memberikan hasil ketika mouse digerakkan pada display map. Perintah pokok untuk dalam

deklarasi kontrol posisi mouse ini adalah **new ol.control.MousePosition({...})**. Deklarasi fitur dimasukkan pada sebuah variabel dan selanjutnya diteruskan sebagai nilai dari parameter controls pada fitur map.



Secara default koordinat posisi mouse ditampilkan dalam EPSG3857, sehingga jika akan ditampilkan dalam sistem lain perlu terlebih dahulu ditransformasikan ke EPSG yang diinginkan. Perubahan dapat dilakukan dengan menambahkan parameter projection di dalam deklarasi fitur ini. Berikut contoh dari deklarasi fitur koordinat posisi mouse.

```
var koordinat = new ol.control.MousePosition
({
  target:document.getElementById('koord'),
  projection: 'EPSG:4326',
  coordinateFormat: ol.coordinate.createStringXY(4),
});
```



Hasil eksekusi dari kode di atas akan nampak koordinat dalam satuan geografis atau EPSG 4326. Jumlah digit diatur sejumlah empat digit dibelakang koma dengan pengaturan parameter **coordinateFormat**.

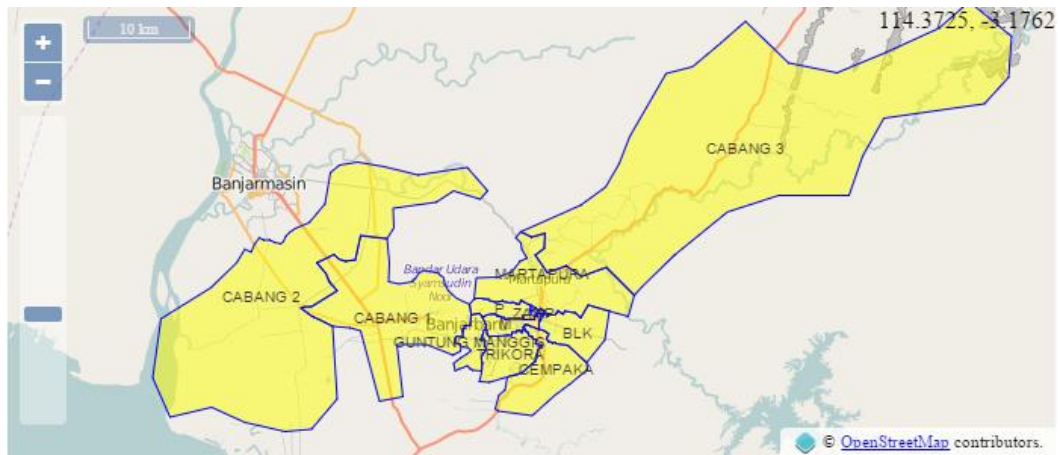
O. Zoom Slider

Zoom slider sering dimanfaatkan dalam aplikasi WebGIS untuk mengubah ukuran tampilan peta. Zoom slider termasuk sebagai kontrol OpenLayers sehingga aplikasinya hampir sama dengan kontrol yang lain. Perintah pokok dari zoom slider ini adalah : **new ol.control.ZoomSlider({...})** . Berikut deklarasi kontrol zoom slider pada sebuah variabel.

```

...
<div id='slider'></div>
....
<script>
var myslider = new ol.control.ZoomSlider
    (
    {
    target:document.getElementById('slider'),
    });
....
Var map = ....
    controls: ol.control.defaults({attributionOptions: ({collapsible: false})
    }).extend([kontrolskala, koordinat, myslider]),

```



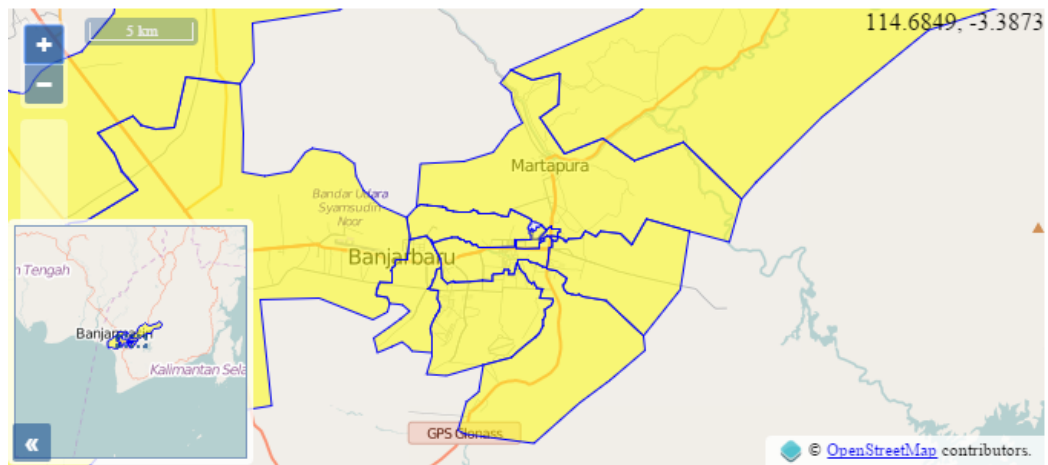
Untuk membatasi level zoom agar zoom tidak terlalu besar ataupun kecil, maka pada deklarasi view perlu ditambahkan parameter **minZoom** dan **maxZoom**. Sebagai contoh adalah seperti pada listing kode dibawah ini.

```
var views = new ol.View
({
  center:pusat,
  zoom: zooms,
  minZoom: 9,
  maxZoom: 13
});
```

P. OverviewMap

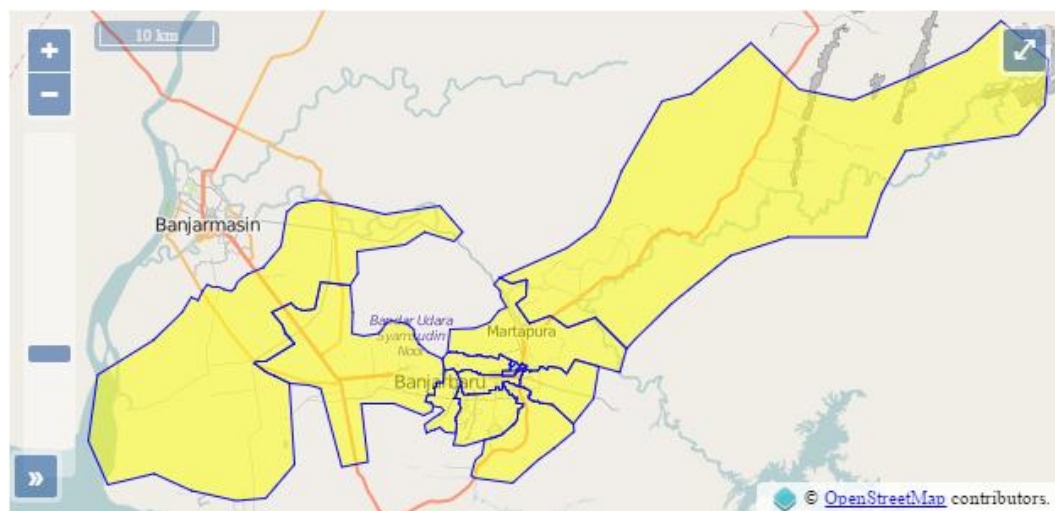
Kontrol overviewmap memiliki fungsi yang mirip dengan inset dalam sebuah peta yang tercetak. Perintah pokok untuk menampilkan kontrol tersebut adalah **new ol.control.OverviewMap ({...})** .

```
...
<div id='inset'></div>
....
<script>
var myinset = new ol.control.OverviewMap
({
  target:document.getElementById('inset'),
  });
....
Var map = ....
  controls: ol.control.defaults({attributionOptions: ({collapsible: false})
    }).extend([kontrolskala, koordinat, myslider, myinset]),
```



Q. FullScreen

Kontrol fullscreen digunakan untuk menampilkan display map secara penuh. Kontrol beroperasi secara bolak-balik, dimana ketika display map dalam status fullscreen, akan dimunculkan sebuah ikon untuk kembali ke ukuran semula. Perintah pokok untuk menampilkan kontrol fullscreen ini adalah **new ol.control.FullScreen({..})**. Secara default posisi kontrol berada di kanan atas display map.



R. Map Interaction

Interaksi merupakan bentuk komunikasi antarmuka antar pengguna webGIS dengan sistem webGIS tersebut. Bentuk interaksi dapat berupa klik, dobel klik, ataupun pergeseran pointer. Interaksi dibentuk melalui sebuah fungsi dan diaplikasikan menjadi sebuah event dalam OpenLayers. Dengan demikian untuk membentuk suatu interaksi terlebih dahulu harus mendeklarasikan suatu fungsi yang memuat perintah-perintah baku terhadap proses interaksi tersebut. Alur dalam penyusunan modul interaksi dilakukan dalam beberapa tahap sebagai berikut :

Langkah	Contoh
<div> obyek overlay	<code><div id="popku" class="pop1"></div></code>
Deklarasi fungsi overlay	<code>var popku = new ol.Overlay ({ element: document.getElementById('popku') }); map.addOverlay(popku);</code>
Even on click	<code>map.on('click', function(evt) {...} var element = popku.getElement(); if (evt.dragging) {return;}</code>
Definisi element overlay (koordinat, piksel, fitur)	<code>var coordinate = evt.coordinate; var pixel = map.getEventPixel(evt.originalEvent); var feature = map.forEachFeatureAtPixel(evt.pixel, function(feature) {return feature;});</code>
Tangkap data fitur	<code>var data1 = feature ? feature.get('ZONA');' var data2 = feature ? feature.get('NAMA ZONA');'</code>
Definisi posisi elemen overlay	<code>popku.setPosition(coordinate); \$(element).popover ({</code>
Definisi isi elemen overlay	<code> 'html': true, 'content': ' Zona : '+data1 + 'Nama Zona : '+data2 });</code>
Tampilkan elemen overlay	<code>\$(element).popover('show');</code>

Berikut contoh listing kode interaksi pada event klik pada display map

```
map.on('pointermove', function(evt)
{
```



```

var element = popku.getElement();
if (evt.dragging) {return;}

var coordinate = evt.coordinate;
var pixel = map.getEventPixel(evt.originalEvent);
var feature = map.forEachFeatureAtPixel(evt.pixel, function(feature)
    {return feature;});

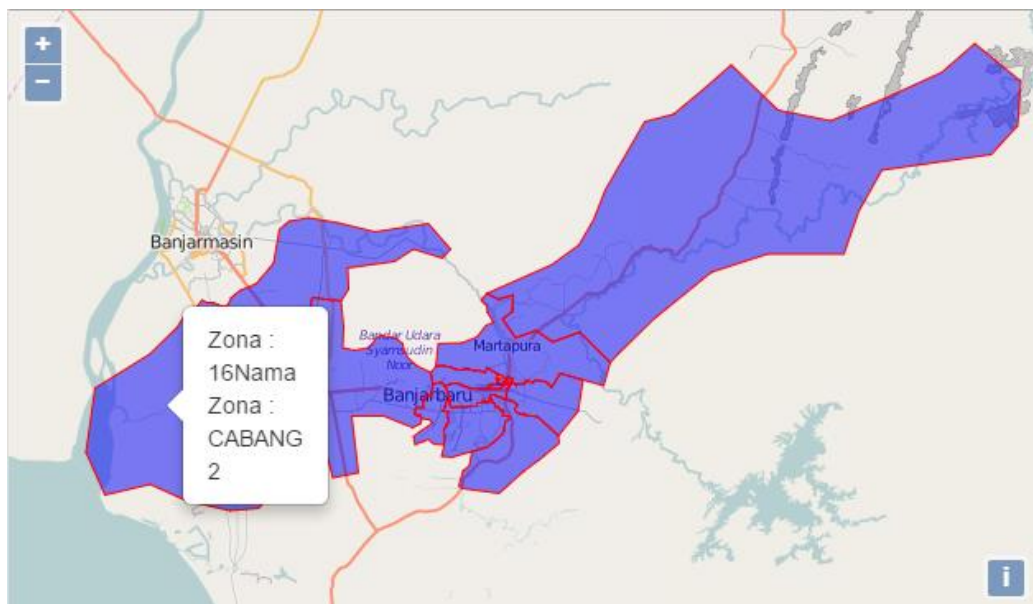
var data1 = feature ? feature.get('ZONA');";
var data2 = feature ? feature.get('NAMA ZONA');";

if (feature)
{
    popku.setPosition(coordinate);
    $(element).popover
    ({
        'html': true,
        'content':      'Zona : '+data1 +
                        'Nama Zona : '+data2

    });
    $(element).popover('show');
}
else
    {$(element).popover('destroy')}
});

```

Hasil eksekusi dari kode tersebut akan nampak seperti pada gambar berikut ini



S. Overlay Layer

Overlay layer sering diaplikasikan dalam webGIS untuk menampilkan data baik vektor maupun raster secara bersamaan atau bergantian dengan sebuah obyek kendali. Overlay layer ini sangat bermanfaat dalam visualisasi informasi ataupun analisis spasial melalui media WebGIS.

Metode overlay layer ini mengaplikasikan kendali terhadap parameter layer dalam sebuah deklarasi map. Prosedur kendali layer dideklarasikan dalam sebuah fungsi. Berikut langkah dari Overlay Layer tersebut.

Langkah	Contoh
Deklarasi modul jquery dan openlayers	<pre><script src="https://code.jquery.com/jquery-1.11.2.min.js"></script> <link rel="stylesheet" href="http://openlayers.org/en/v3.12.1/css/ol.css" type="text/css"> <script src="http://openlayers.org/en/v3.12.1/build/ol.js"></script> <div id="layer" class="layer"> <input id="pc1" type="checkbox" value="1" > Zona Pelayanan PDAM
 <input id="pc2" type="checkbox" value="2" > Jaringan Zona Air Minum
 </div></pre>
Definisi obyek <div> layer dan isi layer	<pre>vector1 = new ol.layer.Vector ({ visible :true, source: new ol.source.Vector({url: './data/json/zona.geojson', format: new ol.format.GeoJSON()}), style : createPolygonStyleFunction() });</pre>
Deklarasi data masing-masing layer	<pre>vector2 = new ol.layer.Vector ({ visible :false, source: new ol.source.Vector({url: './data/json/zam.geojson', format: new ol.format.GeoJSON()}), style : createLineStyleFunction1() });</pre>
Even on click	<pre>\$('#pc1').change(function () {if (\$(this).is(':checked')){vector1.setVisible(true);} else {vector1.setVisible(false);} }); \$('#pc2').change(function () {if (\$(this).is(':checked')){vector2.setVisible(true);} else {vector2.setVisible(false);} });</pre>
Definisi variabel layer Deklarasi parameter layer pada fitur map	<pre>..... var mylayers = [vector1, vector2, ...]; layers: mylayers,</pre>

Contoh listing kode overlay layer dapat dilihat pada kode di bawah ini

```

<html>
<head>
<script src="https://code.jquery.com/jquery-1.11.2.min.js"></script>
<link rel="stylesheet" href="http://openlayers.org/en/v3.12.1/css/ol.css" type="text/css">
<script src="http://openlayers.org/en/v3.12.1/build/ol.js"></script>
</head>

<body>
  <div id="map" class="map"></div>

  <div id="layer" class="layer">
    <input id="pc1" type="checkbox" value="1" checked> OSM
    <input id="pc2" type="checkbox" value="2" > Zona Pelayanan
    <input id="pc3" type="checkbox" value="3" > Valve
  </div>

  <script>

    var vector1 = new ol.layer.Vector
    ({
      visible : false,
      source:   new    ol.source.Vector({url: './data/json/zona.geojson',format:   new
ol.format.GeoJSON()})
    });

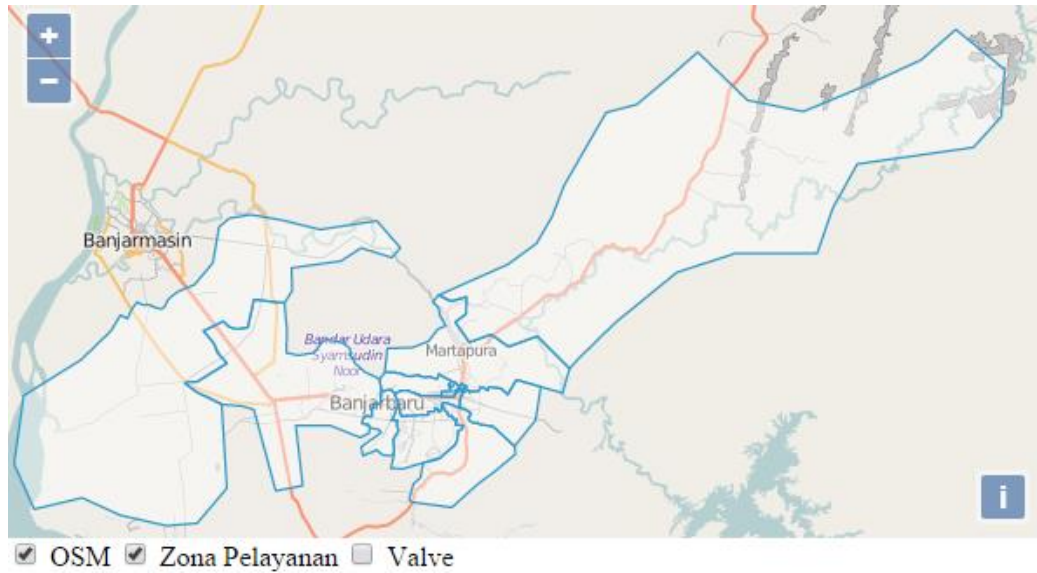
    var vector2 = new ol.layer.Vector
    ({
      visible : false,
      source:   new    ol.source.Vector({url: './data/json/pipa.geojson',format:   new
ol.format.GeoJSON()})
    });

    var myOSM = new ol.layer.Tile({visible : true,source: new ol.source.OSM()});

    $('#pc1').change(function ()
      {if ($(this).is(':checked')){myOSM.setVisible(true);}
      else {myOSM.setVisible(false);}
    });
    $('#pc2').change(function ()
      {if ($(this).is(':checked')){vector1.setVisible(true);}
      else {vector1.setVisible(false);}
    });
    $('#pc3').change(function ()
      {if ($(this).is(':checked')){vector2.setVisible(true);}
      else {vector2.setVisible(false);}
  
```

```
    });  
  
    var proyeksi_awal = 'EPSG:4326';  
    var proyeksi_akhir = 'EPSG:3857';  
    var X = 114.8376;  
    var Y = -3.4389;  
    var pusat = ol.proj.transform([X,Y], proyeksi_awal, proyeksi_akhir);  
    var zooms = 10;  
  
    var views = new ol.View  
    ({  
      center:pusat,  
      zoom: zooms  
    });  
  
    var map = new ol.Map  
    ({  
      target: 'map',  
      layers: [myOSM, vector1, vector2],  
      view: views  
    });  
  
</script>  
  
</body>  
</html>
```

Eksekusi dari listing kode di atas akan menghasilkan tampilan seperti pada gambar berikut ini. Layer dapat dikendalikan dengan cara melakukan klik pada kontrol di bawah display map.



T.Latihan :

1. Buatlah script untuk menampilkan Openstreetmap.
2. Buatlah script untuk menampilkan data spasial bertipe titik, garis, dan poligon
3. Buatlah visualisasi dengan kontrol display peta, label, dan warna
4. Buatlah WebGIS dengan server XAMPP

Pengerjaan :

- a. Latihan di selesaikan maksimal dalam 2 minggu.
- b. Gunakan peta administrasi daerah masing-masing.
- c. Pekerjaan dapat dilakukan secara individu atau berkelompok maksimal 3 orang

Bab 6. Web GIS berbasis Mapserver

A. Konfigurasi Awal

A.1. Instalasi Mapserver

Instalasi dapat dilakukan dengan menggunakan bundel yang telah ada dan dikerjakan dengan proses copy-paste, atau instal secara online dari Maptool.org. Pastikan sistem apache lain atau IIS yang aktif terlebih dahulu dimatikan untuk menghindari konflik sistem. Proses instalasi secara detil akan dipandu dalam pelatihan.

A.2. Folder yang diperlukan adalah :

Untuk membuat sebuah proyek web gis diperlukan sebuah folder project yang dibentuk dibawah c:\ms4w\apps\. Dalam folder tersebut paling tidak dibentuk tiga folder sebagai berikut :

- **Data** : untuk menyimpan semua shapefile
- **Map** : untuk menyimpan script .map
- **htdocs** : untuk menyimpan file .html dan .phtml

Ketiga folder tersebut dimasukkan ke dalam sebuah folder aplikasi dibawah c:\ms4w\apps\. Selain itu diperlukan pula file temp yang berada dibawah c:\ms4w

A.3. Konfigurasi path

Buat file dengan nama **httpd_peta.conf** dan simpan di folder **httpd.d** dibawah **c:/ms4w**. Listing program adalah sebagai berikut :

```
Alias /pdam/ "/ms4w/apps/myapp/htdocs/"
<Directory "/ms4w/apps/pdam/htdocs/">
  AllowOverride None
  Options Indexes FollowSymLinks Multiviews
  Order allow,deny
  Allow from all
</Directory>
```

B. Script Mapserver

Program web gis berbasis mapserver dibangun dengan menggunakan perintah-perintah dalam bahasa html dan php. Perintah diwujudkan menjadi file .map, .html, dan .phtml. Untuk tahap awal sebuah file map saja dapat digunakan untuk melihat hasil sementara tampilan peta pada browser (dalam mode *map*). Sedangkan dalam sebuah webgis yang sempurna (ditampilkan dalam browser dengan mode *browse*) memerlukan bantuan file .html dan .phtml untuk merancang tampilan ataupun pengendalian alur program.

Program dituliskan dengan menggunakan notepad dan disimpan dengan akhiran yang diperlukan (.map , .html , .phtml). Tata aturan penulisan script html dan phtml adalah berlaku untuk penulisan script html dalam mapserver ini.

B.1. Struktur script .map

Map

.....

Layer

.....

Class

.....

Label

.....

Style

.....

.....

End

End

Style

.....

.....

End

End

End

End

B.2. Membuat script map sederhana

Buatlah script berikut dengan notepad dan simpan dalam folder Map dengan nama petadas.map.

MAP

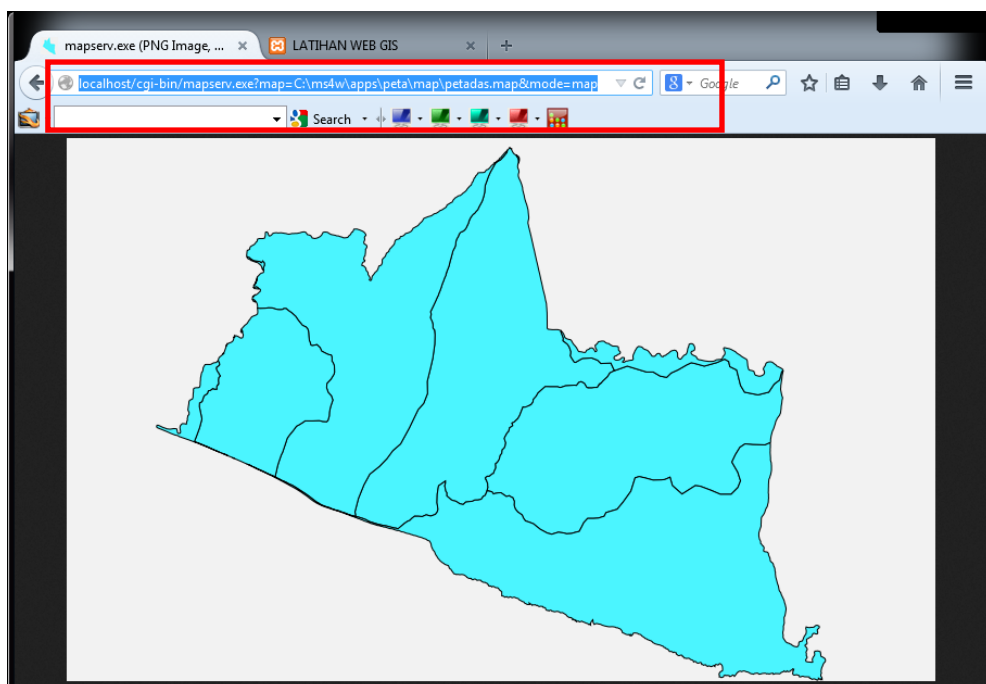
name "PROYEK DAS"

extent 386378 9093153 488693 9167686

```
status on
size 800 500
shapepath "../data"
imagetype png
imagecolor 242 242 242
units meters
layer
  name "PETA_DAS"
  connectiontype ogr
  connection "../das.shp"
  status default
  type polygon
  class
    name "DAS"
    style
      color 0 0 255
      SIZE 5
    end
  end
end
END
```

Script tersebut dapat dieksekusi melalui browser dengan mengetikkan sebuah perintah berikut.

<http://localhost/cgi-bin/mapserv.exe?map=C:\ms4w\apps\peta\map\petadas.map&mode=map>



B.3. Menampilkan layer ganda

Berikut ini adalah script map yang menggabungkan beberapa layer untuk ditampilkan. Layer yang ditulis paling atas akan ditampilkan paling bawah, sehingga kemungkinan akan tertutupi oleh layer yang dituliskan berikutnya pada file map.

```
MAP
name "administrasi "
extent 679113 9178346 688192 9185361
status on
size 800 500
shapepath "../data"
imagetype png
imagecolor 242 242 242
units meters

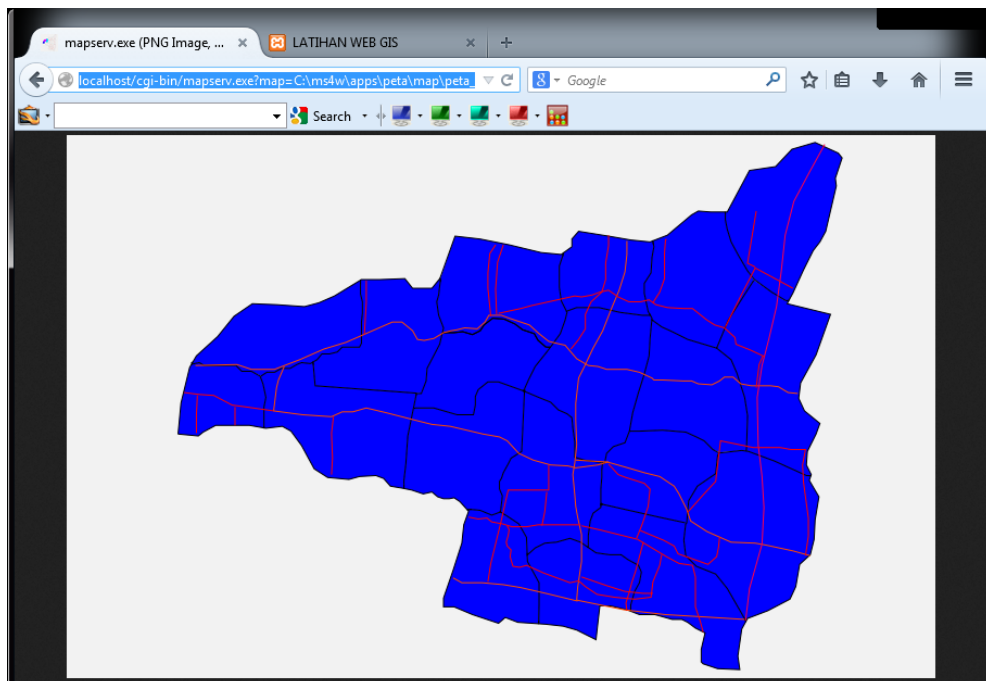
# Peta administrasi tipe poligon
layer
  name "administrasi"
  connectiontype ogr
  connection "./adm_sukodono.shp"
  status default
  type polygon
  class
    name "desa"
    style
      color 0 0 255
    end
  end
end

# peta administrasi tipe garis
layer
  name "batas_administrasi"
  connectiontype ogr
  connection "./administrasi_ln.shp"
  status default
  type line
  class
    name "id"
    style
      color 0 0 0
    end
  end
end
```

```
# peta jalan tipe garis
layer
  name "Peta Jalan"
  connectiontype ogr
  connection "./jalan.shp"
  status default
  type line
  class
    name "jalan"
    style
      color 255 0 0
    end
  end
end
END
```

Simpan dengan nama administrasi.map pada folder htdocs dibawah c:\ms4w\apps\myapp\.

Hasil penggabungan dari beberapa layer diatas menghasilkan tampilan dalam browser seperti gambar berikut ini.



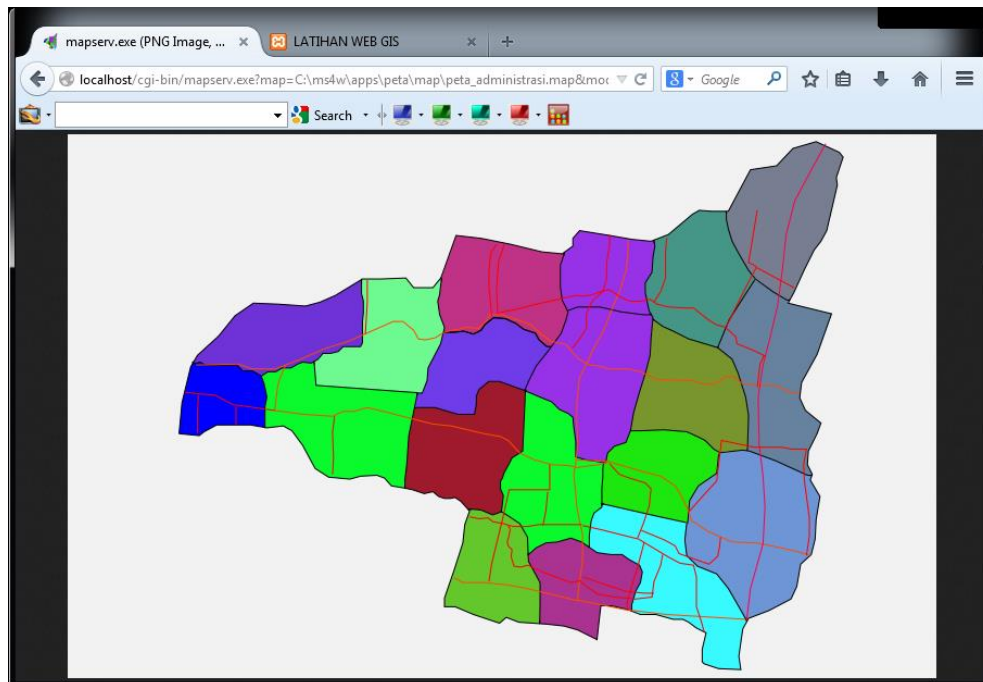
B.4. Menampilkan simbolisasi peta

Pembedaan simbol antar obyek dinyatakan pada bagian Class dengan memasukkan perintah expression sebagai syarat pada kelas tersebut.

.....

```
.....  
layer  
  name "administrasi"  
  CONNECTIONTYPE OGR  
  CONNECTION "./administrasi.shp"  
  status default  
  type polygon  
  class  
    name "DESA1"  
    expression ([ID] = 1)  
    style  
      color 0 0 255  
    end  
  end  
  class  
    name "DESA2"  
    expression ([ID] = 2)  
    style  
      color 110 50 215  
    end  
  end  
  class  
    name "DESA19"  
    expression ([ID] = 19)  
    style  
      color 192 50 133  
    end  
  end  
end
```

Hasilnya akan seperti pada gambar berikut ini.



B.5. Anotasi

Anotasi adalah label dari sebuah obyek. Anotasi sangat penting untuk pengenalan sebuah wilayah pada webgis terutama bagi masyarakat pengguna peta yang berasal dari lokasi yang jauh dan belum mengenal daerah tersebut.

Anotasi dibentuk dengan script pada bagian layer dengan tipe Annotation. Sebelum masuk pada kalang layer anotasi, terlebih dahulu diberikan pernyataan atau deklarasi pengenal huruf seperti berikut.

```
FONTSET "../etc/font/font.dat"
```

Letak deklarasi tersebut berada di bagian bawah sintax MAP dan diatas LAYER (*layer anotasi*). Sementara itu file font.dat dimasukkan ke dalam folder `c:\ms4w\apps\peta\etc\font`. Script pada layer anotasi adalah sebagai berikut.

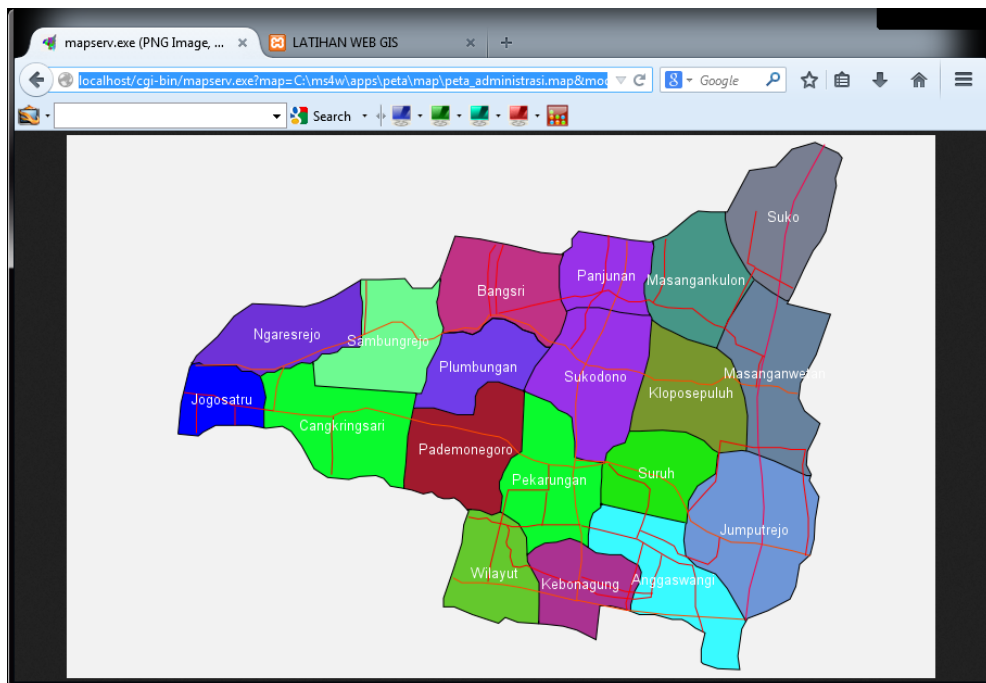
```
layer
  name "anotasi"
  DATA adm_sukodono
  status default
  type annotation
  labelitem "DESA"
  Class
    Label
      Color 255 255 255
      type truetype
      font arial
```

```

size 10
minsize 8
angle auto
position cc
antialias true
end
end
end
end
end

```

Hasil dari script tersebut akan menghasilkan tampilan sebagai berikut.



B.6. Hubungkan dengan menu html

Pemanggilan peta dengan menggunakan perintah yang panjang seperti diatas sangatlah tidak nyaman, apalagi bagi masyarakat umum yang hampir pasti tidak tahu alamat tersebut. Hal ini hampir tidak mungkin untuk dilakukan ketika webgis sudah online. Oleh karena itu penggunaan menu yang di tuliskan melalui html akan sangat membantu.

Buatlah script html berikut dengan menggunakan notepad dan simpan dengan nama menu.html dalam folder html.

```

<html>
<head>
    <title>LATIHAN WEB GIS</title>
</head>
<body>
<div class="img">

```

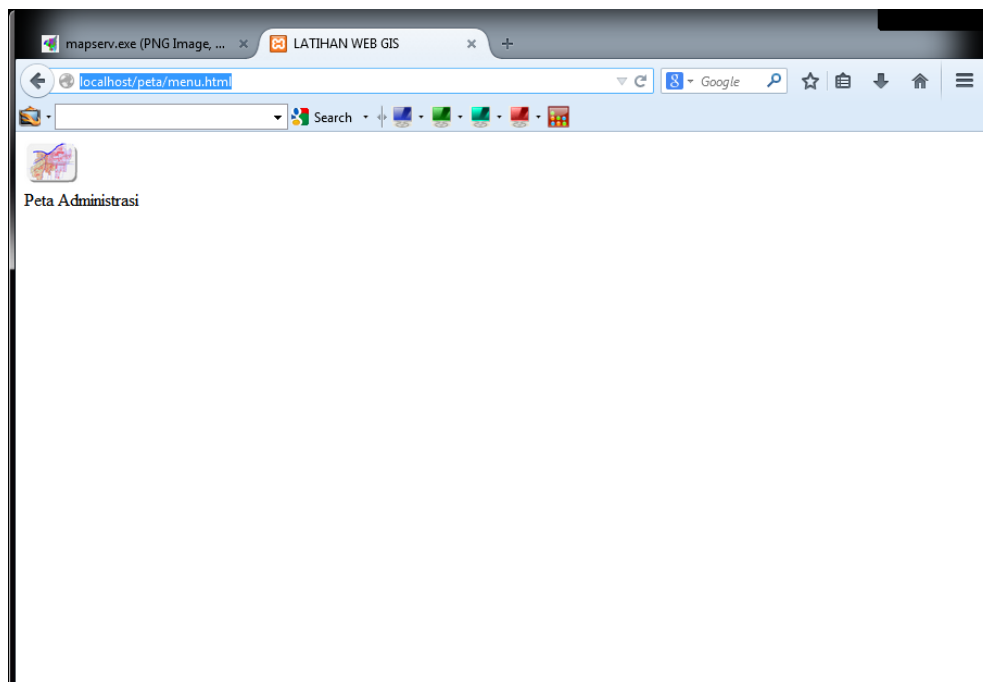
```

<a href=" http://localhost/cgi-
bin/mapserv.exe?map=C:\ms4w\apps\peta\map\administrasi.map&mode=map
">
 </a>
<div class="desc">Peta Administrasi</div>
</div>
</body>
</html>

```

Jika sudah selesai, panggilah pada browser dengan perintah :

<http://localhost/peta/menu.html> Jika berhasil akan menampilkan tampilan seperti gambar berikut ini.



Selanjutnya cobalah klik menu tersebut. Hasilnya adalah akan muncul peta administrasi seperti dimuka.

C. Tampilkan peta dengan template

Menampilkan peta seperti diatas adalah metode menampilkan peta pada browser dengan mode **map**. Tampilan tersebut digunakan dalam tahap awal persiapan pembuatan web gis. Tahap tersebut ditujukan untuk mengatur visualisasi peta yang akan disajikan kepada masyarakat pengguna web gis.

Tahap pengaturan tampilan selanjutnya adalah menampilkannya dalam mode **browse**. Mode ini adalah pengaturan lebih lanjut, dimana tata letak dari

peta dan berbagai propertinya akan diatur. Dalam mode ini diperlukan minimal dua file yaitu file .map dan file .html. Jika penampilannya akan dikaitkan dengan menu berarti memerlukan tiga file, yaitu file menu (html), file .map dan file template (html).

Struktur perintah pada file map harus ditambahkan deklarasi web yang mengkaitkan file map dengan file templatnya. Berikut contoh dari deklarasi tersebut.

```

WEB
    template "../htdocs/t_administrasi.html"
    imagepath "/ms4w/tmp/ms_tmp/"
    imageurl "/ms_tmp/"
END

```

Contoh lengkap dari file map-nya (administrasi.map) adalah sebagai berikut :

```

MAP
name "administrasi"
extent 679113 9178346 688192 9185361
status on
size 800 500
shapepath "../data"
imagetype png
imagecolor 242 242 242
units meters
    WEB
        template "../htdocs/t_administrasi.html"
        IMAGEPATH "/ms4w/tmp/ms_tmp/"
        IMAGEURL "/ms_tmp/"
    END
layer
name "administrasi"
connectiontype ogr
connection "./administrasi.shp"
status default
type polygon
class
    name "desa"
    style
        color 0 0 255
    end
end
end
layer
name "batas_administrasi"
connectiontype ogr

```

```

        connection "./administrasi_ln.shp"
        status default
        type line
        class
            name "id"
            style
                color 0 0 0
            end
        end
    end

layer
    name "Peta Jalan"
    connectiontype ogr
    connection "./jalan.shp"
    status default
    type line
    class
        name "jalan"
        style
            color 255 0 0
        end
    end
end
END

```

Selanjutnya buatlah file template yang digunakan untuk mengatur tampilan peta pada browser. Jika mengacu pada contoh deklarasi diatas, file template harus diberi nama **t_administrasi.html** dan disimpan pada folder c:\ms4w\apps\pdam\htdocs\

Berikut sebuah contoh file template yang sederhana :

```

<!mapserver template>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>
<head>

    <title>Web GIS</title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-
1">
</head>
<body>
    <img src=[img] width = 800 height = 500 ></img>
    <br>Skala tampilan peta saat ini 1:[scale]
</body>

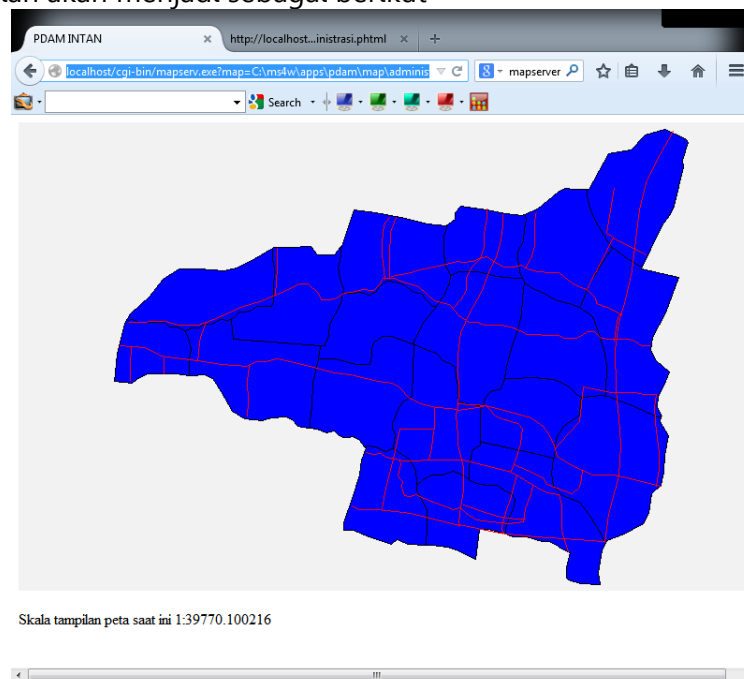
```


</html>

Simpan dengan nama : t_administrasi.html dan disimpan pada folder /htdocs. Selanjutnya cobalah panggil boleh dengan script menu ataupun ditulis langsung pada browser sebagai berikut :

[http://localhost/cgi-
bin/mapserv.exe?map=C:\ms4w\apps\myapp\map\administrasi.map&mode=bro
wse](http://localhost/cgi-bin/mapserv.exe?map=C:\ms4w\apps\myapp\map\administrasi.map&mode=bro
wse)

Hasil tampilan akan menjadi sebagai berikut



Pada tampilan kali ini posisi dan properti peta dapat dikendalikan seperti adanya teks, skala dan properti lainnya.

D. Menambah properti peta

Properti peta seperti berbagai kelengkapan peta dan pengatur tampilan peta didefinisikan pada file html yang berfungsi sebagai template.

D.1. Pengatur ukuran tampilan peta

Sekarang cobalah file template tersebut diubah sehingga menjadi seperti script dibawah ini.

```
<!mapserver template>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

```

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

<body>

<form name="mapserv" method="GET">
<input type="hidden" name="map" value="[map]">
<input type="hidden" name="imgext" value="[mapext]">
<input type="hidden" name="imgxy" value="225 150">
<input type="hidden" name="mode" value="browse">

<table>
<tr>
<! KOLOM PERTAMA ----- >
<td align = center >
<input type="image" name="img" src="[img]" width="450" height="300"
border="0">
<br>Skala tampilan peta saat ini 1:[scale]
<br>
</td>

<! KOLOM KEDUA ----- >

<td valign = top >
<! PILIHAN ZOOM ----- >
<b> TAMPILAN PETA </b><br>
  <select name="zoom" size = "1">
    <option value="3" [zoom_3_select]>
      Zoom In 3x
    </option>
    <option value="2" [zoom_2_select]>
      Zoom In 2x
    </option>
    <option value="1" [zoom_1_select]>
      Recenter
    </option>
    <option value="-2" [zoom_-2_select]>
      Zoom Out 2x
    </option>
    <option value="-3" [zoom_-3_select]>
      Zoom Out 3x
    </option>
  </select>
  <input align="center" type="submit" name="submit"
  value="Refresh"> </div>
<br>

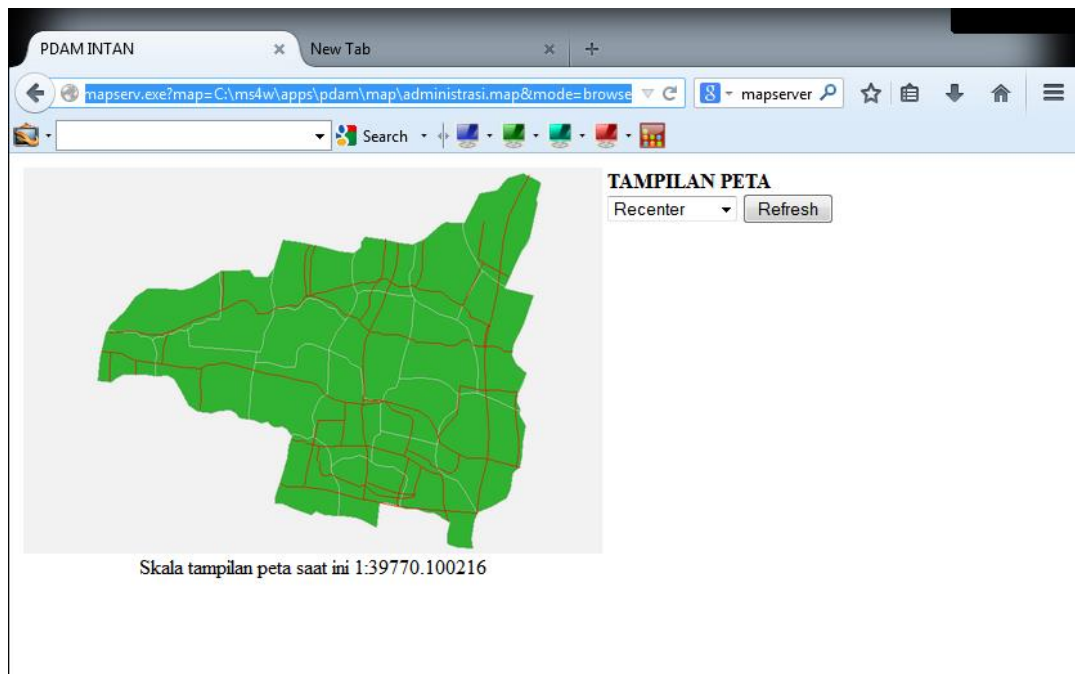
```

```

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

```

Template tersebut akan menghasilkan tampilan seperti di bawah ini. Beberapa tambahan akan muncul yaitu tombol untuk perbesaran dan perkecilan.



D.2. Skala Grafis

Skala grafis dapat ditambahkan pada template, namun harus dideklarasikan pada file map-nya. Tambahkan script berikut pada file administrasi.map

Scalebar

```

status on
label
    size medium
    color 0 0 0
    outlinecolor 255 255 255
end
style 0
intervals 4
size 200 5
color 0 0 0
backgroundcolor 255 255 255
outlinecolor 0 0 0

```

```

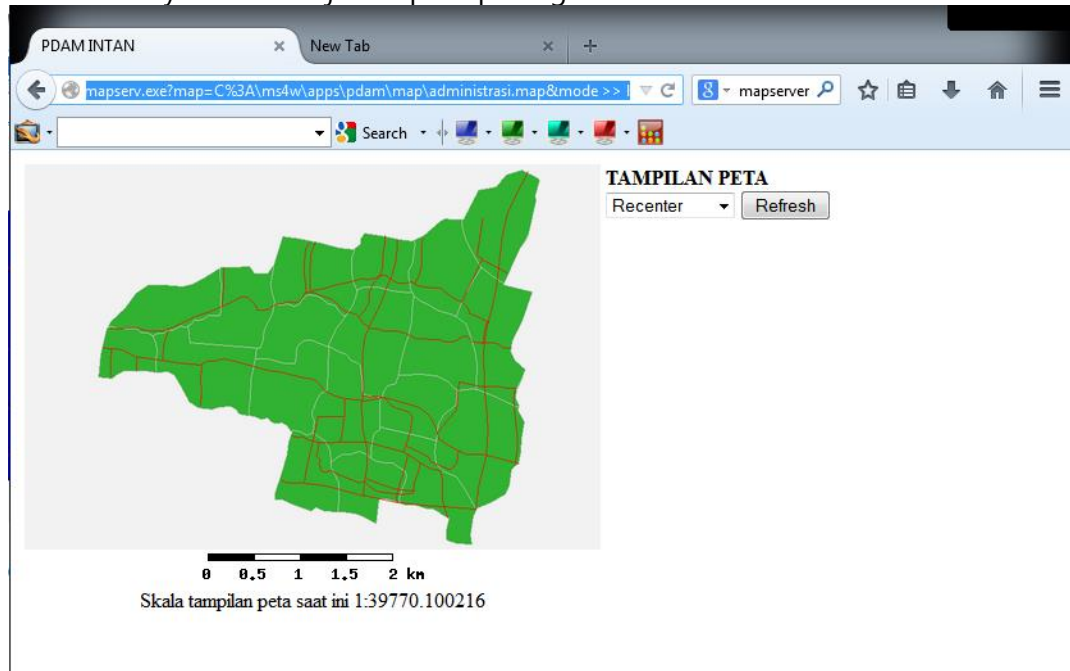
units kilometers
position LR
transparent true
end

```

Selanjutnya tambahkan perintah dibawah ini pada file templatennya. Posisi dibagian atas skala angka :

```
<br><img src=[scalebar]>
```

Hasilnya akan menjadi seperti pada gambar berikut ini.



D.3. Legenda

Sama halnya dengan skala grafis, obyek legenda juga harus didefinisikan pada file map. Contoh script untuk mendefinisikan legenda ini adalah sebagai berikut.

Legend

```

imagecolor 255 255 255
keysize 18 10
keyspacing 3 3
postlabelcache true
Status on
transparent off
label
    type truetype
    font arial
    color 0 0 0

```

```

        size 9
        antialias true
    end
    template "../htdocs/t_legend.html"
end

```

Kemudian pada file template ditambahkan perintah berikut.

```

<fontsize = +1><b> LEGENDA </b> </font><br>
<hr>[legend]<hr>

```

Selengkapnya, file template adalah sebagai berikut.

```

<!mapserver template>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

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

<body>

<form name="mapserv" method="GET">
<input type="hidden" name="map" value="[map]">
<input type="hidden" name="imgext" value="[mapext]">
<input type="hidden" name="imgxy" value="[center]">
<input type="hidden" name="mode" value="browse">

<table>
<tr>
<! KOLOM PERTAMA ----- >
<td align = center >

<input type="image" name="img" src="[img]" width="450" height="300"
border="0">
<br><img src=[scalebar]>

<br>Skala tampilan peta saat ini 1:[scale]
<br>
</td>

<! KOLOM KEDUA ----- >

<td valign = top >
<! PILIHAN ZOOM ----- >
<b> TAMPILAN PETA </b><br>
<select name="zoom" size = "1">
<option value="3" [zoom_3_select]>
Zoom In 3x

```

```

</option>
<option value="2" [zoom_2_select]>
  Zoom In 2x
</option>
<option value="1" [zoom_1_select]>
  Recenter
</option>
<option value="-2" [zoom_-2_select]>
  Zoom Out 2x
</option>
<option value="-3" [zoom_-3_select]>
  Zoom Out 3x
</option>
</select>
<div align="center"> <input type="submit" name="submit"
value="Refresh"> </div>
<br>
<fontsize = +1><b> LEGENDA </b> </font> <br>
<hr>[legend]<hr>
</td></tr>
</table>
</form>
</body>
</html>

```

Pada script tersebut nampak bahwa legenda itu sendiri memerlukan template khusus (t_legend.html) lagi yang berfungsi untuk mendefinisikan isi dari layer peta yang tampil saat itu. Legenda akan terhubung dengan class pada file map. Isi template legend adalah sebagai berikut.

```

[leg_layer_html order=ascending opt_flag=15]
  <input type="checkbox" name="legendlayername[]" value="[leg_layer_name]"
  [if name=layer_status oper=eq value=1]CHECKED[/if]
  [if name=layer_status oper=eq value=2]CHECKED[/if]>
  
  <span class="label"> [leg_layer_name] </span> </a>
  <br>
  [/if]
  [if name=wms_name oper=isset]
    <a href="#"
onclick="javascript:LegendTemplateLayerInfo('[leg_layer_name]');" >
  <span class="label"> [metadata name=WMS_TITLE] </span> </a>
  <br>
  [/if]

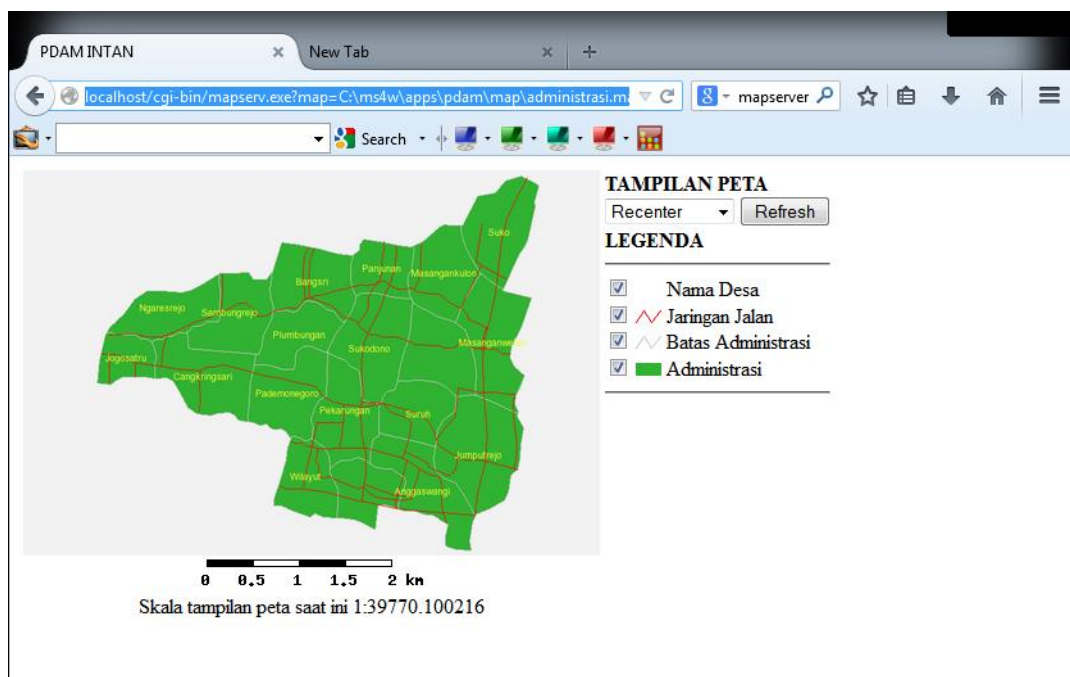
```

[/leg_layer_html]

Cobalah panggil pada dengan menggunakan menu ataupun diketikkan langsung pada baris alamat browser.

<http://localhost/cgi-bin/mapserv.exe?map=C:\ms4w\apps\pdam\map\administrasi.map&mode=browse>

Hasil dari pemanggilan tersebut akan menghasilkan tampilan seperti pada gambar berikut ini.



E. Menggunakan Widget Chameleon

Chameleon adalah piranti-piranti tambahan yang dikembangkan oleh DMSolution untuk pengembangan webgis. Piranti ini bersifat open source dan dapat digunakan untuk melengkapi beberapa fungsionalitas antarmuka dalam webgis. Widget Chameleon dapat diwujudkan menjadi berbagai tombol-tombol (ikon) seperti zoom, pan, identify, dan lain-lain.

Widget ini dapat diaplikasikan pada mapserver dengan menambahkan beberapa baris script yang merujuk pada obyek dan properties widget tersebut. Dalam pemanfaatan widget Chameleon ini ada beberapa persyaratan yang harus dipenuhi, yaitu :

- Chameleon telah terinstal bersama dengan mapserver tersebut
- Struktur folder aplikasi sebaiknya menyertakan folder turunan dengan nama map, data, htdocs, dan etc. Fungsi dari masing-masing folder telah diuraikan di depan
- Pada bagian awal script template menyertakan deklarasi berikut
 - ✚ Pada head : `<link href = "CWC2.css" rel ="stylesheet" type="text/css">`
 - ✚ Pada baris Body : `<body OnLoad = "CWC2OnLoadFunction()">`
- Masing-masing widget dinyatakan dengan `<CWC2 Type =/>`

Contoh :

```
<cwc2 type="MapDHTMLWidget"
  visible="true"
  width="350"
  height="270"
  allowresize="true"
  markeecolor="FF3333"
  markeewidth="2"
  minscale="1"
  bordercolor="#0000ff"
  borderwidth="0"/>
</cwc2>
```

- Dalam menampilkan peta diperlukan sebuah file map (.map), file template (.html), dan file inisiasi chameleon (.phtml).

Contoh file map : (nama file : adm_cham.map)

```
MAP
name "Peta Administrasi"
extent 679113 9178346 688192 9185361
status on
size 800 500
shapepath "../data"
SYMBOLSET "../etc/symbols.sym"
FONTSET "../etc/font/font.dat"
imagetype png
imagecolor 242 242 242
units meters
```



```
        WEB
            IMAGEPATH "/ms4w/tmp/ms_tmp/"
            IMAGEURL "/ms_tmp/"
        END

    layer
        name "Administrasi"
        connectiontype ogr
        connection "./administrasi.shp"
        status default
        type polygon
        class
            name "desa"
            style
                color 50 178 50
            end
        end
    end

    layer
        name "Batas Administrasi"
        connectiontype ogr
        connection "./administrasi_ln.shp"
        status default
        type line
        class
            name "batas"
            style
                color 200 200 200
                outlinecolor 200 200 200
            end
        end
    end

    layer
        name "Jalan"
        connectiontype ogr
        connection "./jalan.shp"
        status default
        type line
        class
            name "jalan"
            style
                color 255 0 0
            end
        end
    end
end
```

```
layer
  name "Desa"
  DATA administrasi
  status default
  type annotation
  labelitem "DESA"
  Class
    Label
      Color 247 252 50
      type truetype
      font arial
      size 10
      minsize 8
      angle auto
      position cc
      antialias true
    end
  end
end

Scalebar
  status on
  label
    size medium
    color 0 0 0
    outlinecolor 255 255 255
  end
  style 0
  intervals 4
  size 200 5
  color 0 0 0
  backgroundcolor 255 255 255
  outlinecolor 0 0 0
  units kilometers
  position LR
  transparent true
end

Legend
  imagecolor 255 255 255
  keysize 18 10
  keyspacing 3 3
  postlabelcache true
  Status on
  transparent off
  label
    type truetype
    font arial
```

```

        color 0 0 0
        size 9
        antialias true
    end
    template "../htdocs/t_legend.html"
end
END

```

Contoh file template : (nama file : t_peta_cham.html)

```

<!mapserver template>
<html>
<head>
    <title>Web GIS</title>
    <link href="cwc2.css" rel="stylesheet" type="text/css">

    <script type="text/javascript"
    src="/LocateByAttribute/LocateByAttribute_functions.js">
    </script>

</head>

<body OnLoad="CWC2OnLoadFunction()">
<form>
<table>
<tr>
<! KOLOM PERTAMA ----- >
<td align = center >
<cwc2 type="MapDHTMLWidget"
    visible="true"
    width="350"
    height="270"
    allowresize="true"
    marqueecolor="FF3333"
    marqueewidth="2"
    minscale="1"
    bordercolor="#0000ff"
    borderwidth="0"/>
</cwc2>

<br>Skala tampilan peta saat ini 1:[scale]
<br>
</td>

<! KOLOM KEDUA ----- >

<td valign = top >
<! PILIHAN ZOOM ----- >

```

 TAMPILAN PETA


```
<cwc2 type="ZoomOut"
  visible="true"
  imagewidth="24"
  imageheight="24"
  imagetip="Perkecil"
  toolset="Navigation">
  <image state="normal"
    image="buttons/button_zoomout_1.png"/>
  <image state="hover"
    image="buttons/button_zoomout_2.png"/>
  <image state="selected"
    image="buttons/button_zoomout_3.png"/>
</cwc2>
```

```
<cwc2 type="ZoomIn"
  visible="true"
  imagewidth="24"
  imageheight="24"
  imagetip="Perbesar"
  toolset="Navigation"
  default="true">
  <image state="normal"
    image="buttons/button_zoomin_1.png"/>
  <image state="hover"
    image="buttons/button_zoomin_2.png"/>
  <image state="selected"
    image="buttons/button_zoomin_3.png"/>
</cwc2>
```

```
<cwc2 type="zoomalllayers"
  visible="true"
  imagewidth="24"
  imageheight="24"
  imagetip="Tampil Utuh"
  toolset="Navigation">
  <image state="normal"
    image="buttons/button_zoomfull_1.png"/>
  <image state="hover"
    image="buttons/button_zoomfull_2.png"/>
  <image state="selected"
    image="buttons/button_zoomfull_3.png"/>
</cwc2>
```

```
<cwc2 type="recenter"
  visible="true"
  imagewidth="24"
```

```

        imageheight="24"
        imagetip="Posisi Tengah"
        toolset="Navigation">
        <image state="normal"
            image="buttons/button_recenter_1.png"/>
        <image state="hover"
            image="buttons/button_recenter_2.png"/>
        <image state="selected"
            image="buttons/button_recenter_3.png"/>
        </cwc2>

<br>
<span style = "font-family:arial narrow; font-size:11;">
    <cwc2 type="cursorpos"
        axis="x"
        label="Bujur : "
        Labelposition="8"
        labelvalign="middle"
        TEXTFIELDSIZE="10"/>
<span style = "font-family:arial narrow; font-size:11;">
    <cwc2 type="cursorpos"
        axis="Y"
        label="Lintang : "
        Labelposition="8"
        labelvalign="middle"
        TEXTFIELDSIZE="10"/>

<br>
<fontsize = +1> <b> LEGENDA </b> </font> <br>
<hr>[legend]<hr>

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

```

Contoh file phtml (nama file : p_administrasi.phtml)

```

<?php
/*
 * this is a sample of a minimal application script required to
 * get a chameleon application running with a template and a
 * local map file. Most applications should actually just
 * copy this file and modify the template and mapfile
 * parts and should not need to do a lot more here.
 */
include( "../chameleon/htdocs/chameleon.php" );
$szTemplate = "/t_peta_cham.html";

```

```

$szMapFile = "../map/adm_cham.map";

class latihan01 extends Chameleon
{
function latihan01()
{
parent::Chameleon();
$this->moMapSession = new MapSession_RW;
$this->moMapSession->setTempDir( getSessionSavePath());
}
}

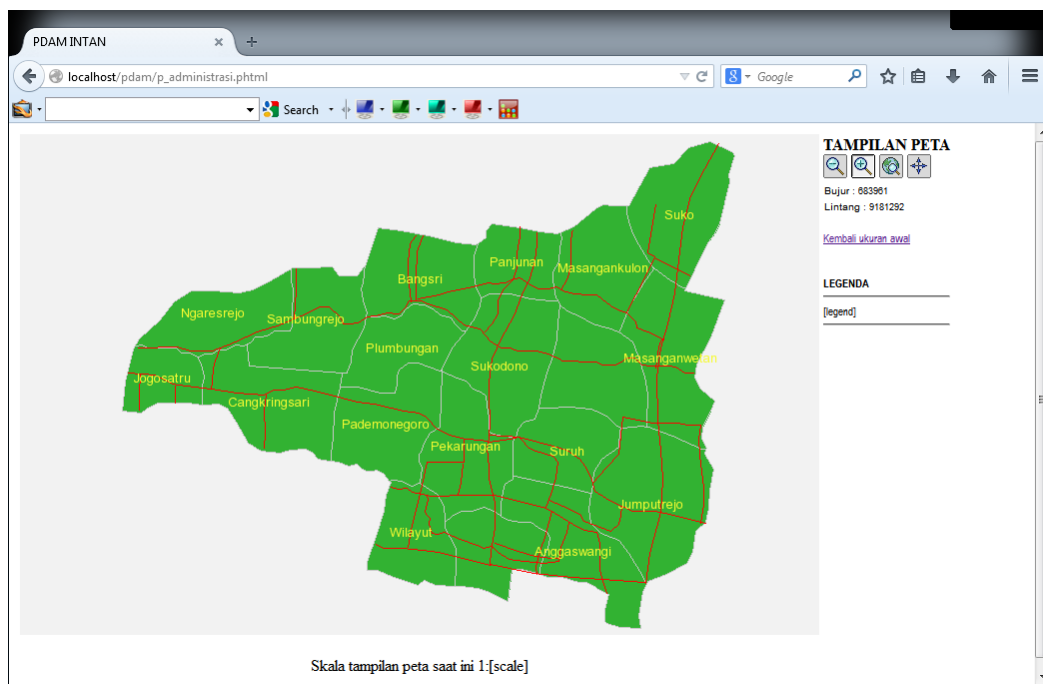
$oApp = new latihan01();
$oApp->registerSkin( 'skins/sample' );
$oApp->CWCInitialize( $szTemplate, $szMapFile );
$oApp->CWCExecute();
?>

```

Cobalah tulis ketiga file tersebut dengan notepad. File html dan phtml simpan dalam folder htdoc, sedangkan file map disimpan dalam folder map. Panggil file phtml (p_administrasi.phtml) pada browser dengan perintah :

[Localhost/myapp/p_administrasi.phtml](http://localhost/myapp/p_administrasi.phtml)

Hasil dari proses diatas akan menghasilkan visualisasi seperti gambar berikut ini.



Latihan :

5. Lakukan Instalasi Mapserver pada komputer masing-masing
6. Buatlah script yang dapat digunakan untuk menampilkan satu data spasial atau citra
7. Buatlah visualisasi berannotasi atau label
8. Buatlah visualisasi data dengan beberapa bentuk kontrol display.
9. Buatlah aplikasi webgis dengan memanfaatkan widget Chameleon.

Pengerjaan :

- d. Latihan dapat dilakukan secara individu atau berkelompok maksimal 2 orang
- e. Latihan di selesaikan maksimal dalam 2 minggu.
- f. Gunakan peta administrasi daerah masing-masing.

DAFTAR PUSTAKA

- Agosto, E., 2013. Vector–raster server-side analysis: a PostGIS benchmark, *Appl Geomat*. Vol. 5. Hal. 177–184. DOI 10.1007/s12518-013-0104-x
- Alesheikh, A.A., Helali, H., Behroz, H.A., 2002. Web GIS: Technologies and its applications. *Symposium on Geospatial Theory, Processing and Applications*. Ottawa.
- Alexis, S., Montero, L.G.G., Hernandez J., Abril, A.G., Pastor, J., 2010. Soil fertility and GIS raster models for tropical agroforestry planning in economically depressed and contaminated Caribbean areas (coffee and kidney bean plantations). *Agroforest Syst* . Vol. 79. Hal. 381–391. DOI 10.1007/s10457-009-9263-5
- Alhasanat, M.B., Kabir, S., Hussin, W.M.A.W., Eddison, E., 2012. Spatial analysis of a historical phenomenon: using GIS to demonstrate the strategic placement of Umayyad desert palaces. *GeoJournal*. Vol. 77. Hal. 343–359. DOI 10.1007/s10708-010-9392-4

- Alkobaisi, S., Bae, W.D., Vojtechovsky, P., Narayanappa, S., 2012. An interactive framework for spatial joins: a statistical approach to data analysis in GIS. *Geoinformatica* Vol. 16. Hal. 329–355. DOI 10.1007/s10707-011-0134-7
- Ates, M., 2013. Geography Teachers' Perspectives towards Geography Education with Geographic Information Systems (GIS). *International Journal of Innovative Research in Science, Engineering and Technology*. Vol. 2, Issue 10.
- Basnet, B.B., Apan, A.A., Raine, S.R., 2001. Selecting Suitable Sites for Animal Waste Application Using a Raster GIS. *Environmental Management*. Vol. 28, No. 4, hal. 519–531 DOI: 10.1007/s002670010241.
- Bednarz, S.W., 2004. Geographic information systems: A tool to support geography and environmental education?, *GeoJournal*. Vol. 60. Hal. 191–199,
- Bortoloti, F.D., Junior, R.M.C., Araujo, L.C., Morais, M.G.B., 2015. Preliminary landslide susceptibility zonation using GIS-based fuzzy logic in Vitoria, Brazil. *Environ Earth Sci*. Vol. 74. Hal. 2125–2141. DOI 10.1007/s12665-015-4200-6.
- Branch, R.M., 2009. *Instructional Design: The ADDIE Approach*. Springer. New York. DOI 10.1007/978-0-387-09506-6.
- Cheremia, E., Tokareva, N., Rische, N., 2012. Application of advance GIS technologies to environmental monitoring. NSF Supplement to IIP-0829576 for collaboration with I/UCRC-CAKE's Russian Site. State Research Center of the Russian Federation
- Delamater, P.L., Messina, J.P., Shortridge, A.M., Grady, S.C., 2012. Measuring geographic access to health care: raster and network-based methods. *International Journal of Health Geographics*, Vol. 11. Hal. 15.
- Esri. 2012. *GIS in Education: across campuses, inside facilities*. Esri. NewYork
- Hamzeh, M., Abbaspour, R.A., Davalou, R., 2015. Raster-based outranking method: a new approach for municipal solid waste landfill (MSW) siting. *Environ. Sci. Pollut. Res*. Vol. 22. Hal. 12511–12524. DOI 10.1007/s11356-015-4485-8.
- Isman, A., 2011. *Instructional Design In Education*. The Turkish Online Journal of Educational Technology. Vol. 10, Issue I. Hal. 136 – 142.
- Khan, M.N., Odman, M.T., Karimi, H., Goodchild, M., 2000. Developing and integrating advanced GIS Techniques in adaptive grid air quality model to reduce uncertainty. 4th International Conference on Integrating GIS and Environmental Modeling: Problems, Prospects and Research Needs. Alberta – Canada.

- Kilbrink, N., Bjurulf, V., Blomberg, I., Heidkamp, A., Hollsten, A.N., 2014. Learning specific content in technology education: learning study as a collaborative method in Swedish preschool class using hands-on material, *Int J Technol Des Educ*, Vol. 24. Hal. 241–259 DOI 10.1007/s10798-013-9258-4
- Li, L., Yang, Y., Wang, H., Dong, J., Zao, Y., He, J., 2011. Spatial Statistical Analysis in Cow Disease Monitoring Based on GIS, dalam Li, D., Liu, Y., Chen, Y., (Ed.), 2011, CCTA, Part II, IFIP AICT Vol. 345, hal. 561–566.
- Mahdjoubi, D., 2009. Four Types of R&D. Lecturing Materials. St. Edward's University, Stavanger - Norway.
- McElvany, N., Schroeder, S., Baumert, J., Schnotz, W., Horz, H., Ullrich, M., 2011. Cognitively demanding learning materials with texts and instructional pictures: teachers' diagnostic skills, pedagogical beliefs and motivation. *Eur J Psychol Educ*, Vol. 27. Hal. 403-420. DOI 10.1007/s10212-011-0078-1
- Ojanen, V., dan Vuola, O., 2003. Categorizing the Measures and Evaluation Methods of R&D Performance – A State-of-the-art Review on R&D Performance Analysis, Working Papers 16, Lappeenranta University of Technology, Lappeenranta.
- Prince, M., dan Felder, R.M., 2006. Inductive Teaching And Learning Methods: Definitions, Comparisons, And Research Bases, *J. Engr. Education*, Vol. 95(2), hal. 123–138
- Rao, S., dan Vinay, S., 2009. Choosing the right GIS framework for an informed Enterprise Web GIS Solution. CIESIN, Columbia University & NASA. New York, USA
- Simsek, A., 2012. Locus of Instructional Control, Learner Control. Dalam: Seel, N., 2012. *Encyclopedia of the Sciences of Learning*. DOI 10.1007/978-1-4419-1428-6.
- Singh, P.S., Chuti, D., Sudhakar, S., 2012. Development of a Web Based GIS Application for Spatial Natural Resources Information System Using Effective Open Source Software and Standards. *Journal of Geographic Information System*, Vol. 4, hal. 261-266.
- Sudaryanti, Kusrahmadi, S.D., 2011. Pengembangan model bahan ajar pendidikan lingkungan hidup berbasis lokal mata pelajaran ilmu pengetahuan sosial. Universitas Negeri Yogyakarta. Yogyakarta.
- UNESCO. 2014 . Guide to Conducting an R&D Survey: For countries starting to measure research and experimental development. UNESCO Institute for Statistics. Montreal