

## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

Bab analisa dan perancangan membahas tentang perancangan sistem yang meliputi analisis permasalahan, perancangan diagram alir yang menunjukkan alur jalan dari sistem, desain arsitektur yang menunjukkan hubungan antar elemen, dan perancangan sistem informasi. Perancangan sistem informasi geografis penentuan lahan potensial pertumbuhan terdiri dari perancangan UML yang meliputi *use case diagram*, *activity diagram*, *sequence diagram* dan *class diagram*. Dalam bab ini juga dilengkapi dengan struktur tabel dan desain *input output* pada Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing*.

#### **1.1 Analisis Permasalahan**

Sebagai instansi pemerintah yang melayani dan membantu masyarakat dalam hal pertanian Dinas Pertanian membutuhkan data-data yang berkaitan dengan masalah pertanian salah satunya adalah daerah potensial yang ada di wilayahnya. Dinas Pertanian Propinsi Jawa timur menyadari akan hal tersebut, oleh karena nya mereka memerlukan sebuah sistem informasi yang mampu mengolah data-data yang telah mereka miliki menjadi data daerah potensial.

Saat ini Dinas Pertanian Jawa Timur hanya memiliki data berupa hasil produksi pertanian dari tiap-tiap kabupaten di Jawa Timur. Namun mereka belum memiliki data mengenai data daerah potensi lain diluar daerah yang sudah ada. Mereka juga memerlukan data daerah potensi lain di luar dari daerah yang menghasilkan untuk dikembangkan agar dapat menghasilkan hasil pertanian yang produktif sehingga meningkatkan kesejahteraan kabupaten tersebut.

Kendala yang dihadapi mereka adalah data yang mereka miliki masih berupa data raster atau data 'mentah'. Mereka harus merubah data tersebut menjadi data digital dengan proses digitasi agar data tersebut dapat diolah menggunakan komputer. Namun kendala lain muncul, mereka tidak memiliki tenaga ahli yang dapat melakukan proses digitasi tersebut.

Proses digitasi memerlukan keahlian dalam penggunaan software tertentu dan membutuhkan biaya yang besar untuk membiayai operator yang dapat melakukan proses tersebut. Saat ini Dinas Pertanian tidak ingin melakukan penambahan operator karena cpns tahun ini tidak ada penambahan.

Staf IT pada Dinas Pertanian menyadari pemanfaatan teknologi informasi yang menunjang mampu memberikan kontribusi bagi para staff yang lainnya dalam memahami atau menyelesaikan suatu masalah. Para staf ingin agar dalam tampilan daerah potensial dibuat visualisasi 3D. Mereka ingin sesuatu yang lebih dari sebuah peta 2D biasa, sehingga para staf dapat mengetahui secara pasti kontur permukaan bumi sesungguhnya. Mereka akan dapat melihat relief bumi secara nyata yang ditampilkan dalam bentuk 3D. Dengan adanya visualisasi 3D tersebut diharapkan akan dapat memudahkan para staf lain dalam memahami maksud dan tujuan dibuatnya Sistem Informasi ini.

Dari hasil permasalahan diatas mereka membutuhkan suatu sistem informasi geografis yang dapat mengolah data raster yang mereka miliki agar dapat menjadi output berupa daerah potensial dalam bentuk *vector*. Mereka memerlukan hasil *output* berupa daerah potensial. Daerah potensial adalah daerah yang tepat untuk ditanami suatu tanaman. Pada daerah tersebut tanaman jenis

tertentu akan dapat tumbuh dengan baik dimana tiap tanaman memiliki karakteristik jenis lahan yang berbeda-beda.

Untuk dapat mengolah data berupa raster atau citra. Perlu diterapkan metode *Image Processing* atau proses citra digital. Karena hanya dengan metode tersebut. Citra dapat di analisa dan bahkan dapat dibuat *vector*-nya berdasarkan kriteria *edge detect* dari gambar tersebut.

*Edge Detect* adalah proses citra digital dimana gambar akan ditemukan perberdaan antara pixel satu dengan *pixel-pixel* lain disekitarnya. Apabila ada perbedaan yang signifikan antara *pixel* tersebut dengan *pixel-pixel* di sekitarnya maka pixel tersebut dapat dinyatakan sebagai batas atau *edge*. Proses *edge detect* digunakan dalam aplikasi ini untuk merubah hasil *output* dari sistem berupa daerah potensial menjadi format ESRI Shapefile dan Google KML. Dimana kedua format tersebut adalah format yang paling sering digunakan dalam aplikasi SIG lainnya dan format tersebut dalam bentuk *vector*.

Data *raster* yang mereka miliki adalah peta curah hujan dan peta kelembaban yang didapat kan dari Badan Meteorologi dan Geofisika (BMKG). BMKG sendiri juga tidak dapat menyediakan data berupa *vector* karena mereka membuat data yang berasal dari citra satelit yang juga berupa *raster*. BMKG hanya menambahkan beberapa informasi tambahan yang melengkapi peta *raster* nya tersebut seperti nama daerah dan skala peta.

Proses digitasi dapat dilakukan secara otomatis dengan menerapkan *Image Processing*. Peran *Image Processing* adalah menentukan batas atau *edge* pada *raster* yang akan menjadi acuan untuk membentuk suatu *polygon* berdasarkan

pada *edge* yang telah ditemukan. Semakin besar resolusi gambar maka semakin akurat hasil dari proses *Image Processing* ini untuk menentukan batas atau *edge*.

Proses pertama yang dilakukan sistem informasi geografis ini pada data raster yang diinputkan adalah. Data akan dipisahkan berdasarkan warnanya. Karena struktur data *raster* peta kelembaban dan peta ketinggian berupa warna yang mewakili sebuah informasi mengenai kelembaban atau curah hujan. Ketika warna sudah di kelompokkan menurut warna, kemudian tiap warna diberi data informasi sesuai yang tertera pada gambar tersebut.

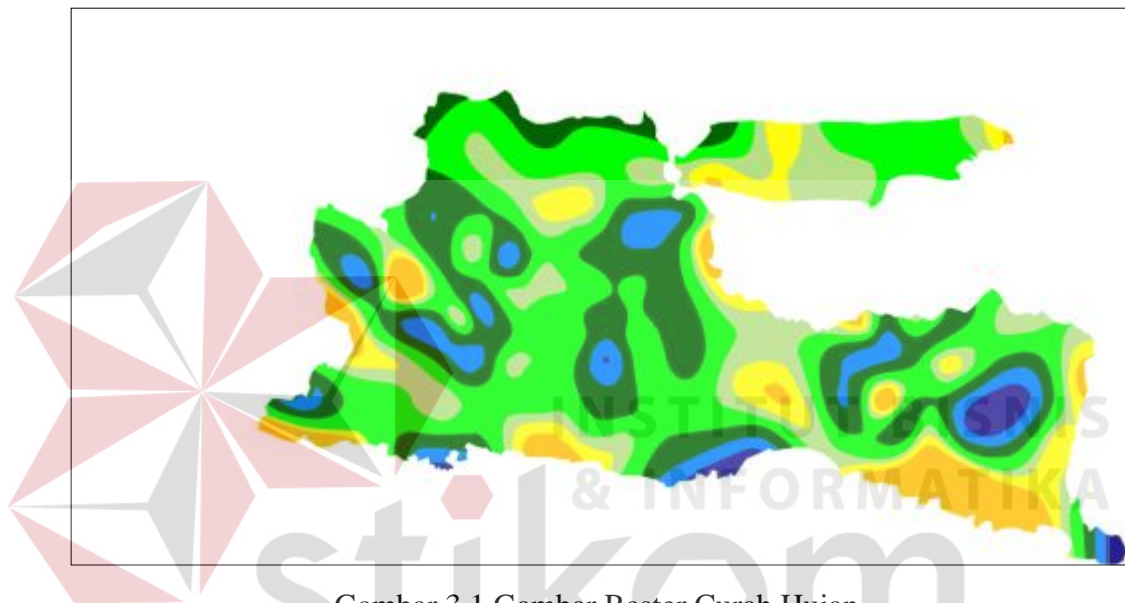
Proses selanjutnya adalah mencocokkan kriteria dari tanaman yang ingin diketahui daerah yang tepat untuk ditanami jenis itu. Untuk kriteria suhu, data yang dipergunakan adalah peta ketinggian, dimana terdapat korelasi antara suhu dan ketinggian

Dengan adanya Sistem Informasi Geografis yang menggunakan *Image Processing* ini, Dinas Pertanian akan dapat mengolah data mentah mereka menjadi data daerah potensial yang sangat berguna untuk perencanaan dan pengembangan daerah.

## **1.2 Perancangan Sistem Informasi Geografis Penentuan Lokasi Lahan Potensial Tumbuhan**

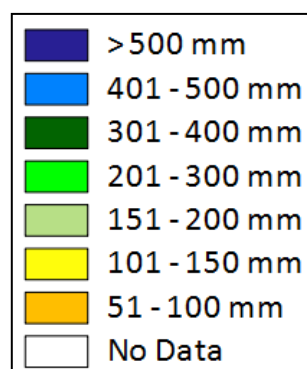
Sistem Informasi Geografis Penentuan Lahan Potensial dapat mengubah peta ketinggian dan peta permukaan yang berupa gambar/*raster* menjadi sebuah peta spasial/data *vector* yang dapat dimanfaatkan oleh program GIS lainnya. Untuk melakukan hal tersebut perlu beberapa tahap yang harus dilakukan untuk merubah data gambar/*raster* menjadi data daerah potensial yang berupa *vector*.

Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing* ini memanfaatkan data yang dimiliki oleh Dinas Pertanian berupa data raster curah hujan dan data raster peta kelembaban untuk propinsi Jawa Timur yang mereka dapatkan dari BMKG. Adapun data raster yang akan digunakan dalam Sistem ini nantinya dapat dilihat pada Gambar 3.1 dan Gambar 3.3 di halaman 33.

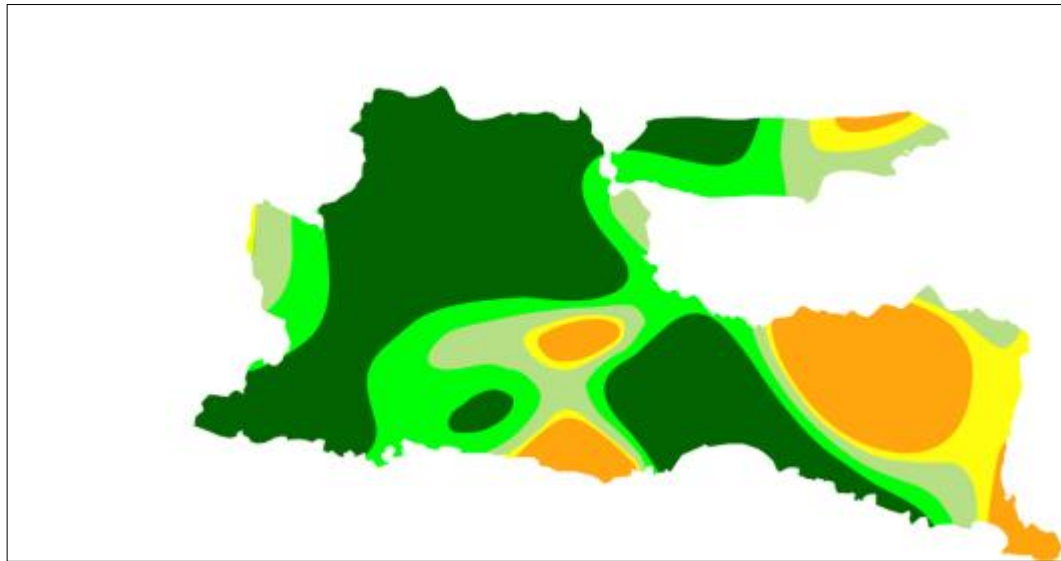


Gambar 3.1 Gambar Raster Curah Hujan

Setiap warna dari gambar tersebut memiliki makna curah hujan. Legenda warna dari gambar tersebut dapat dilihat pada Gambar 3.2.

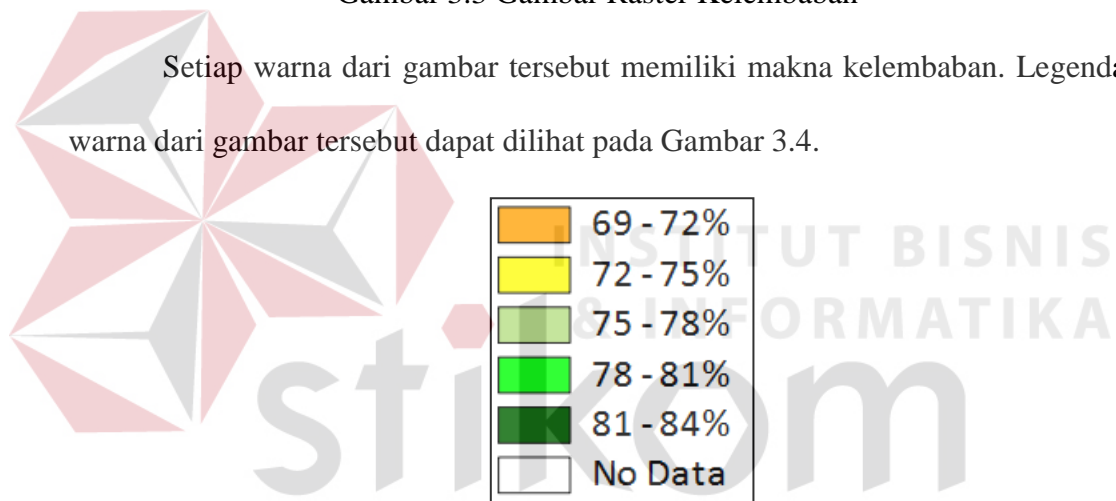


Gambar 3.2 Keterangan dari gambar raster curah hujan



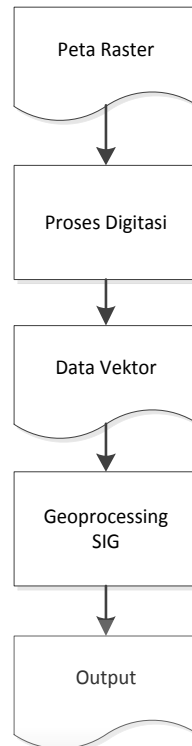
Gambar 3.3 Gambar Raster Kelembaban

Setiap warna dari gambar tersebut memiliki makna kelembaban. Legenda warna dari gambar tersebut dapat dilihat pada Gambar 3.4.



Gambar 3.4 Keterangan Dari Gambar Raster Kelembaban

Sistem Informasi Geografis ini sedikit berbeda dari Sistem informasi Geografis lainnya. Pada umumnya Sebuah sistem informasi geografis hanya dapat melakukan *geoprocessing* apabila data inputannya berupa data *vector* apabila data nya masih berupa data raster, maka harus dirubah dulu formatnya menjadi data *vector* menggunakan aplikasi SIG lain atau fitur lain yang disebut dengan proses digitasi. Diagram SIG pada umumnya dapat dilihat pada Gambar 3.5 di halaman 34.

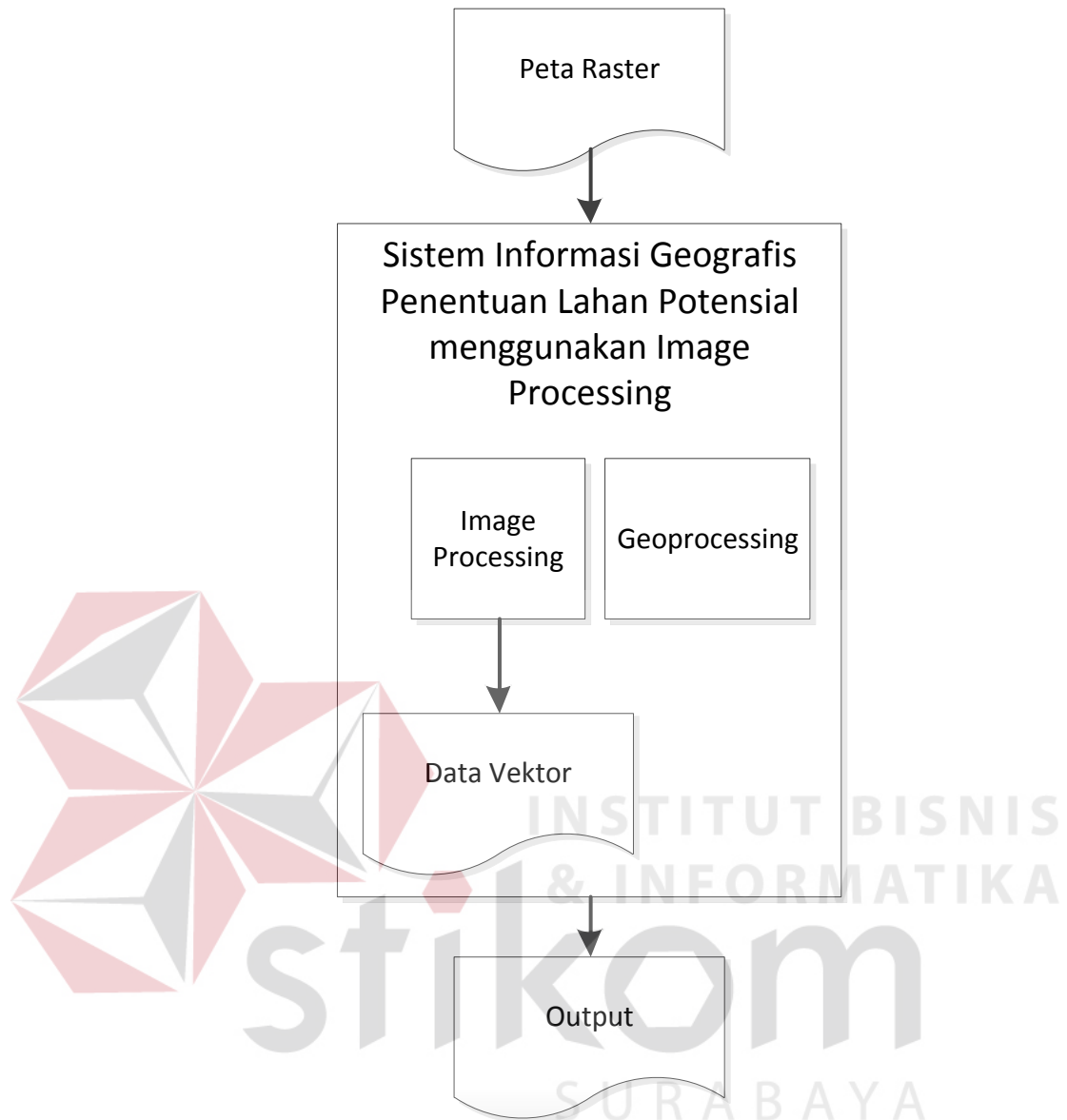


Gambar 3.5 Proses Penggunaan SIG Pada Umumnya

Sedangkan dalam Sistem Informasi Geografis Penentuan Lahan Potensial menggunakan *Image Processing* ini mampu melakukan *geoprocessing* pada data *raster* karena proses digitasi dilakukan oleh proses citra visual yang mampu menemukan garis, dan *polygon* yang terdapat dalam gambar. Gambar sitematika kerja Sistem informasi ini dapat dilihat pada Gambar 3.6 di halaman 35.

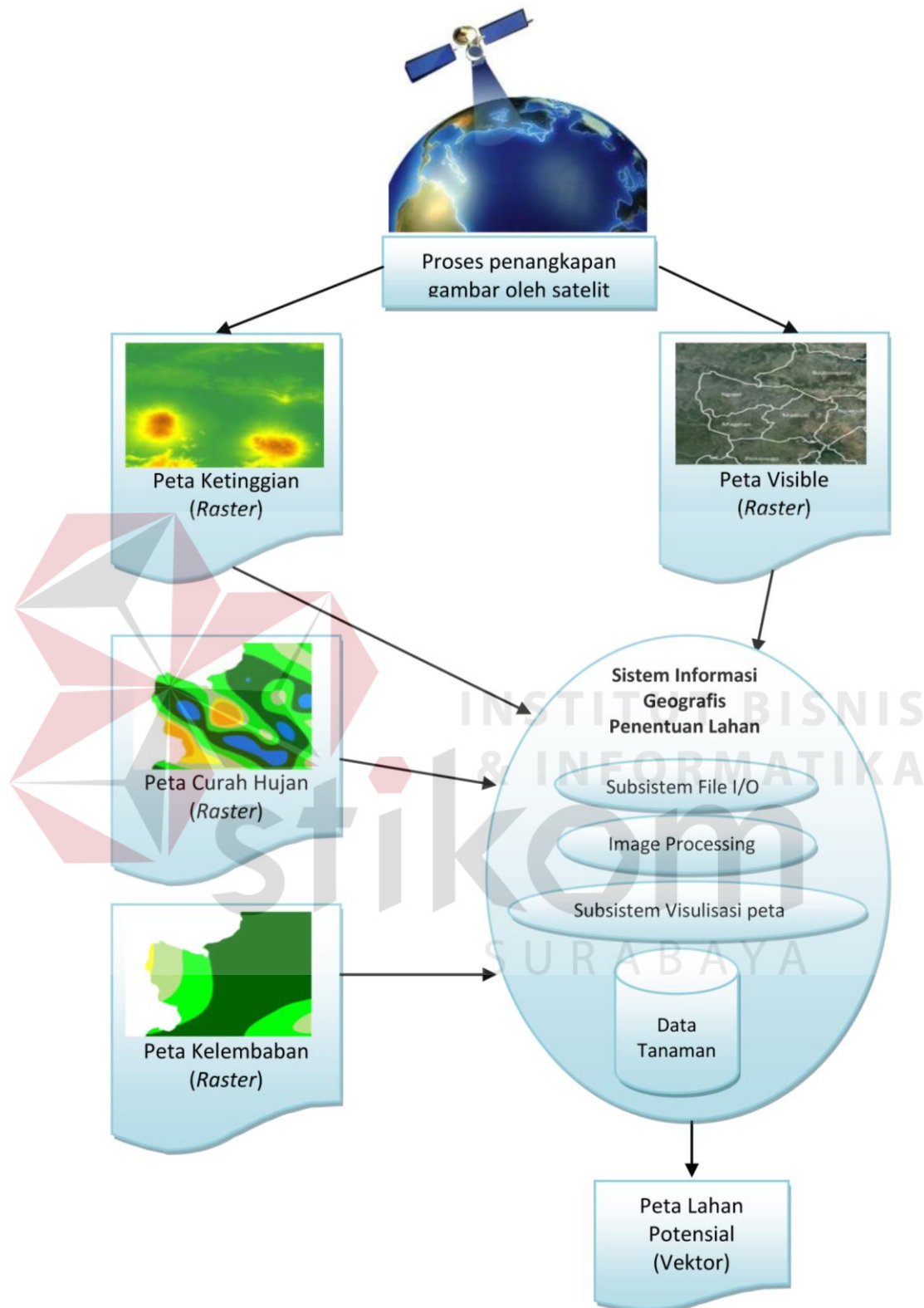
### 1.2.1 Gambaran Umum Sistem

Secara umum Sistem informasi ini mengolah data *raster* berupa data ketinggian dan data *raster* lainnya. Sistem ini tidak membahas bagaimana mendapatkan gambar dari citra satelit. Gambaran umum Sistem informasi geografis penentuan lahan potensial menggunakan *Image Processing* dapat dilihat pada Gambar 3.7 di halaman 36.



Gambar 3.6 Sistemika Kerja SIG Menggunakan *Image Processing*



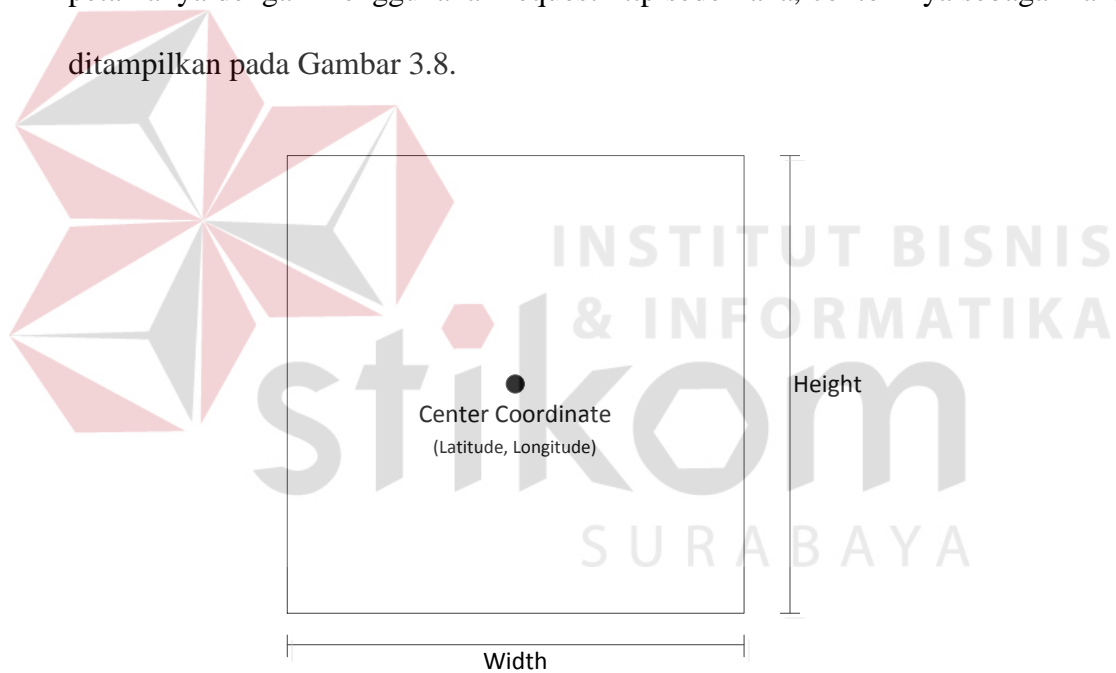


Gambar 3.7 Gambaran Umum Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing*

Sistem informasi geografis ini memiliki beberapa metode tambahan yang merupakan fitur dalam aplikasi ini. berikut adalah penjelasan dari beberapa fitur yang menunjang kinerja dari Sistem informasi Geografis ini.

### 1. Metode Menggunakan Gambar dari GoogleMaps Dengan Menggunakan Koneksi Internet

Dalam aplikasi ini terdapat fitur untuk melakukan proses *download* gambar dari Google Maps dengan menggunakan fasilitas dari Google yaitu Google Maps Static API. Google memberi kemudahan *user* untuk mendapatkan peta hanya dengan menggunakan request http sederhana, contoh nya sebagaimana ditampilkan pada Gambar 3.8.



Gambar 3.8 Ilustrasi Metode *Google Maps Static API*

Google Maps Static API sepenuhnya menggunakan *Request* HTTP dalam bentuk URL. Alamat yang disediakan Google menggunakan alamat <http://maps.googleapis.com> dengan beberapa parameter tambahan sesuai dengan kebutuhan peta yang diinginkan *user*. Format *request* dari Google Maps Static API dapat dilihat pada Gambar 3.9 di halaman 38.

```

http://maps.googleapis.com/maps/api/staticmap?
center=latitude,longitude
&zoom=zoomlevel
&maptype=satellite
&size=heightxwidth
&sensor=false

```

Gambar 3.9 Format *Request GoogleMaps Static API*

Sebagai contoh apabila *user* melakukan sebuah *request* dengan menggunakan *GoogleMaps Static API* pada Gambar 3.10.

```

http://maps.googleapis.com/maps/api/staticmap?
center=-6.99958,111.548
&zoom=10
&maptype=satellite
&size=400x400
&sensor=false

```

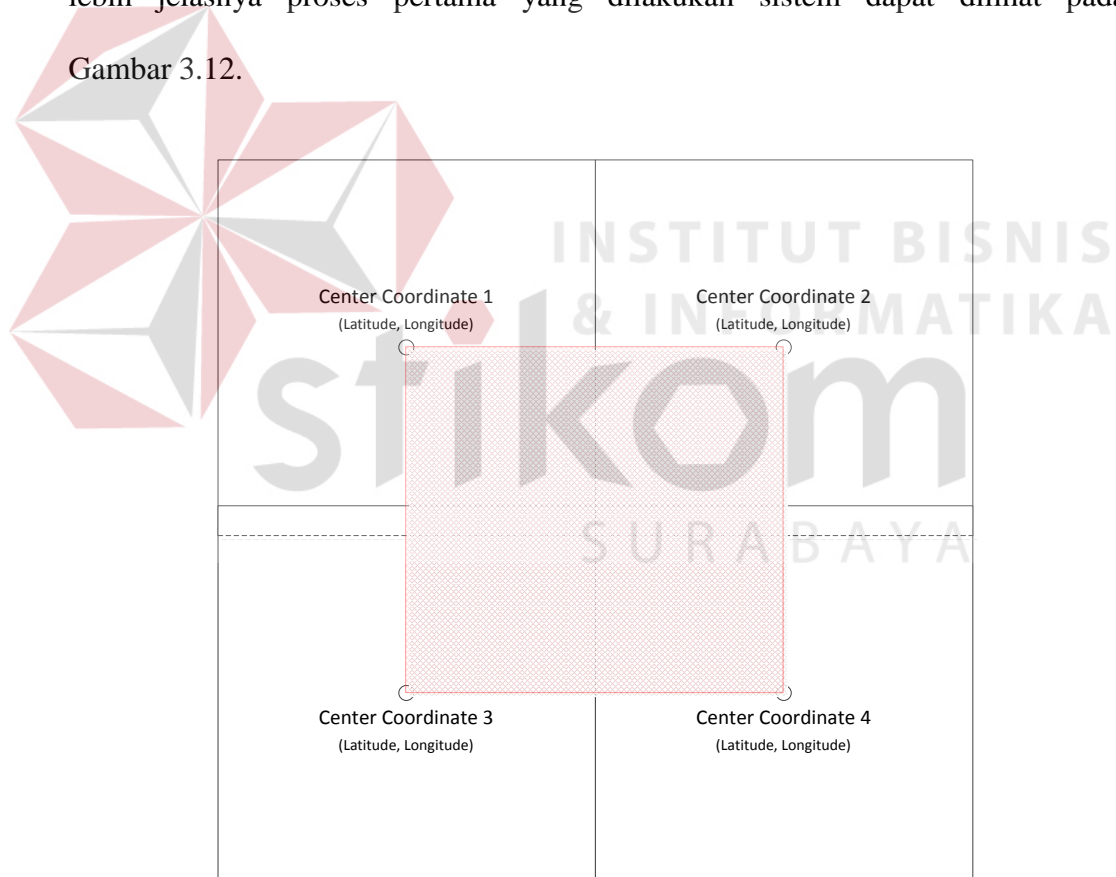
Gambar 3.10 Contoh *Request GoogleMaps Static API*

Maka hasil dari *request* tersebut apabila dijalankan dengan menggunakan *browser* akan didapat sebuah gambar dengan format png seperti pada Gambar 3.11.



Gambar 3.11 Hasil Contoh *Request GoogleMaps Static API*

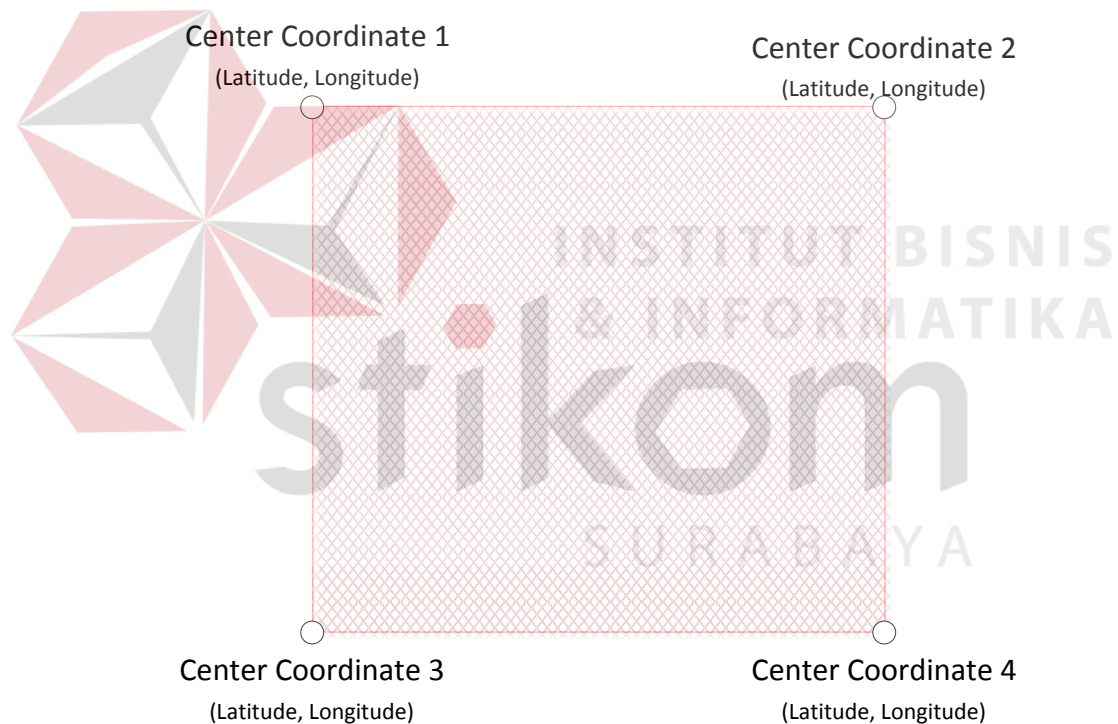
Sementara itu ada beberapa masalah dalam penggunaan *GoogleMaps Static API* yaitu ukuran gambar hanya dibatasi sampai 1024x1024 *pixel*. Hal ini tentu tidak bisa digunakan dalam aplikasi dengan menggunakan satu *request*, karena aplikasi pasti nantinya membutuhkan gambar peta dalam ukuran yang besar. Untuk mengatasi hal tersebut Sistem ini menerapkan multiple request dengan mengatur posisi gambar sesuai dengan ukuran dan letak geografisnya. Untuk mendapatkan peta daerah pada *range* area tertentu sistem pertama-tama menyiapkan beberapa *request* yang hasil *outputnya* disusun seperti *matrix*. Untuk lebih jelasnya proses pertama yang dilakukan sistem dapat dilihat pada Gambar 3.12.



Gambar 3.12 Blok *Matrix* Yang Disiapkan Untuk *GoogleMaps Static API*

Dapat dilihat pada Gambar 3.12, dua gambar dibawah sedikit lebih keatas hal ini dimaksudkan untuk menutup logo *GoogleMaps* yang selalu muncul setiap

kali melakukan *request* peta pada *GoogleMaps Static API*. Ilustrasi dilihat pada Gambar 3.12 di halaman 39 berjumlah empat *block* sehingga sistem akan melakukan *request* gambar peta menggunakan *GoogleMaps Static API* sebanyak 4 kali dengan parameter yang berbeda. Daerah yang di arsir adalah hasil *output*. Setelah selesai mendownload keempat *block* tersebut, sistem akan memotong gambar (*cropping*) daerah yang di arsir tersebut. Sehingga hasil akhir dari gambar tersebut akan memiliki data georefrensi di ke empat ujungnya. Ilustrasi dari hasil gambar hasil *cropping* dapat dilihat pada Gambar 3.13.



Gambar 3.13 Hasil Akhir Metode *Import GoogleMaps Static API*

## **2. Generate Terrain Merubah Gambar Raster Dua Dimensi (2D) Menjadi Tiga Dimensi (3D)**

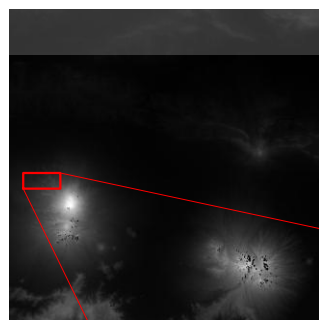
Sistem Informasi ini memiliki fitur visualiasi terrain dalam bentuk 3D. *Terrain* di generate dari *file* elevasi yaitu DTED (*Digital Terrain Elevation Data*)

yaitu *file raster* yang menyimpan ketinggian peta. Contoh *file* dari DTED dapat dilihat pada Gambar 3.14.



Gambar 3.14 Representasi *Raster* Dari DTED *File*

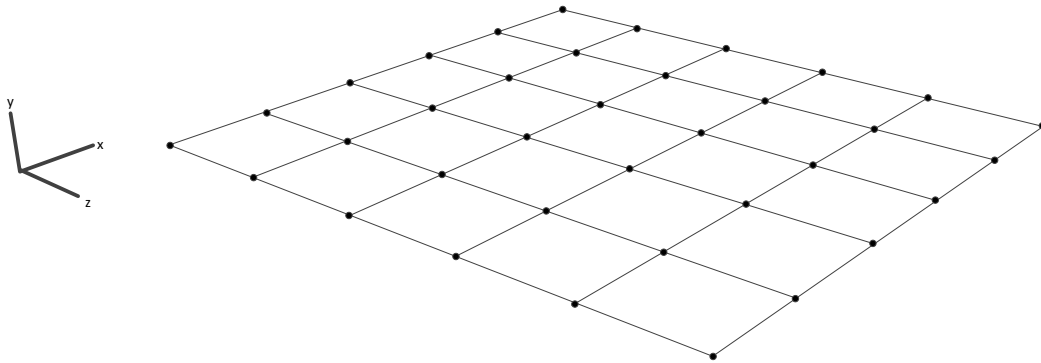
DTED dapat menyimpan ketinggian dari permukaan tanah dengan format dua dimensi. Warna putih di Gambar 3.14 menandakan daerah tersebut daerah pegunungan. Ilustrasi data yang disimpan *DTED* dapat dilihat pada Gambar 3.15.



1500	1502	981	871	943	978	1462	1542	1453
1542	1240	1105	1240	1438	1243	1324	1544	1477
1705	1457	1204	1140	1547	1467	1465	1522	1787
1508	1254	1424	1304	897	1547	1432	154	1987

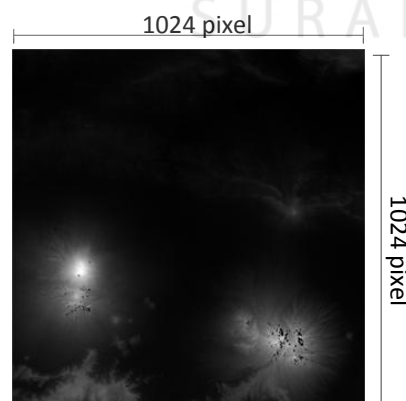
Gambar 3.15 Data Ketinggian Yang Disimpan DTED *File*

Proses pertama yang dilakukan adalah menyiapkan bidang tiga dimensi atau dikenal dengan istilah *plane*. Ilustrasi dari *plane* dapat dilihat pada Gambar 3.16.

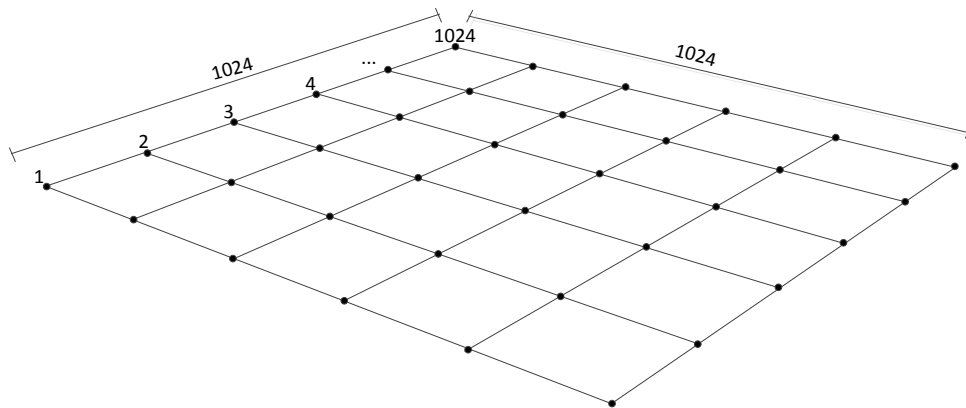


Gambar 3.16 Ilustrasi *Plane* Dalam Bidang Tiga Dimensi

Jumlah titik dari *plane* tersebut harus sama dengan jumlah pixel pada *file* DTED. Sehingga apabila dalam *file* DTED memiliki ukuran 1024 x 1024 pixel maka jumlah titik pada *plane* juga 1024 x 1024 pixel. Ilustrasi dari sumber DTED dapat dilihat pada Gambar 3.17. Sedangkan ilustrasi *plane* tiga dimensi dapat dilihat pada Gambar 3.18. di halaman 43.

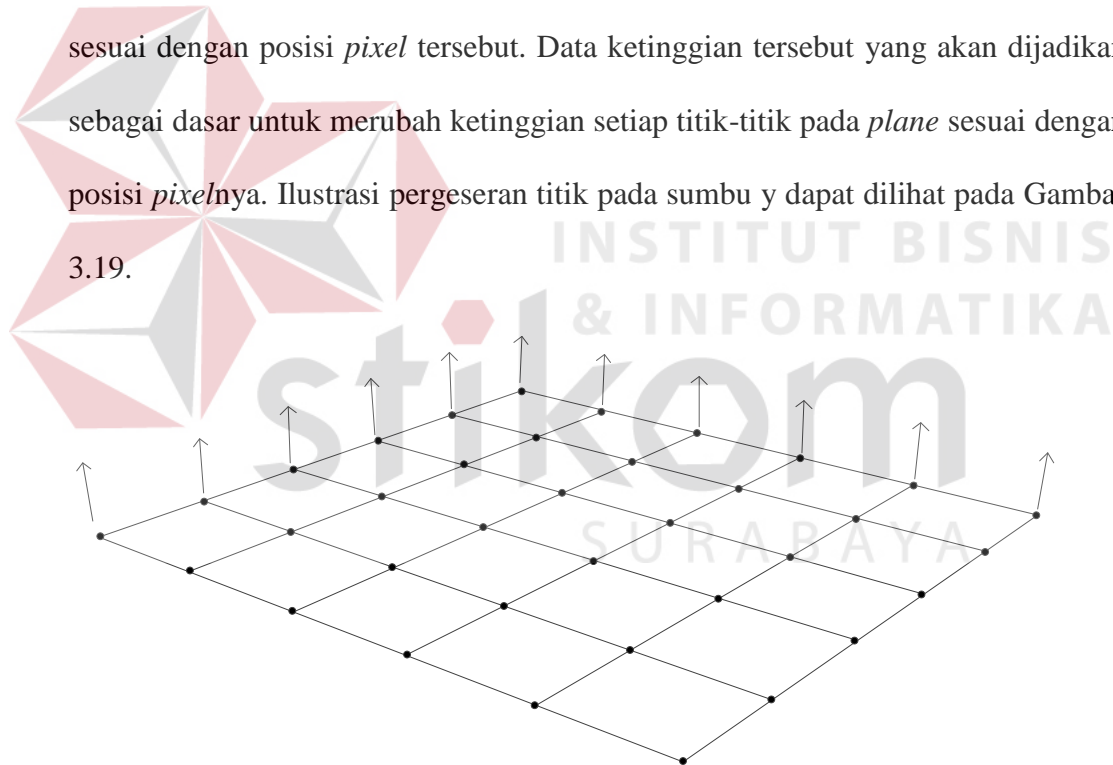


Gambar 3.17 Ilustrasi Data Peta Ketinggian Yang Akan Dirubah



Gambar 3.18 Ilustrasi *Plane* Dalam Bidang Tiga Dimensi

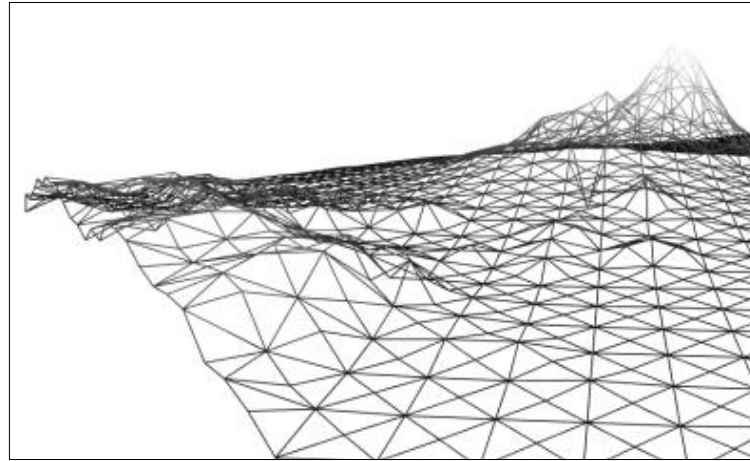
Tahap selanjutnya, tiap *pixel* pada DTED memiliki data ketinggian yang sesuai dengan posisi *pixel* tersebut. Data ketinggian tersebut yang akan dijadikan sebagai dasar untuk merubah ketinggian setiap titik-titik pada *plane* sesuai dengan posisi *pixel*nya. Ilustrasi pergeseran titik pada sumbu y dapat dilihat pada Gambar 3.19.



Gambar 3.19 Ilustrasi *Plane* Yang Akan Dirubah Ketinggiannya

Sehingga hasil dari penggeseran titik-titik tersebut akan tampak seperti pada Gambar 3.20 di halaman 44.



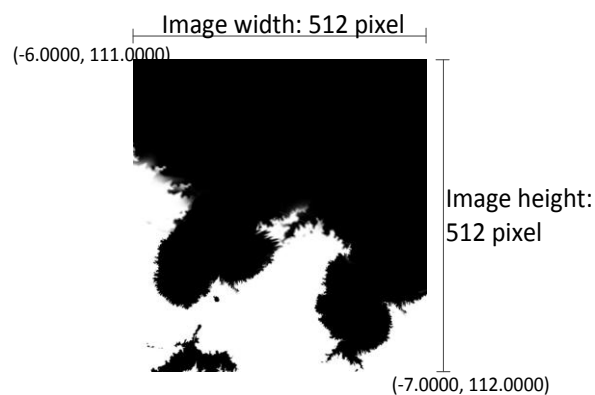


Gambar 3.20 Hasil Ilustrasi *Plane* Dalam Bidang Tiga Dimensi

### 3. Proses Transformasi Data Gambar *Raster* Menjadi Data *Vector*

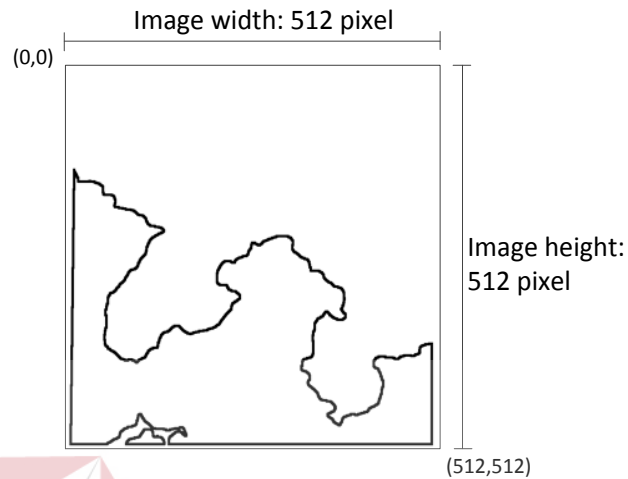
Hasil dari Sistem Informasi Geografis Penentuan Lahan Potensial menggunakan *Image Processing* adalah data berupa *vector*, sedangkan *input* dan beberapa proses dari aplikasi ini masih menggunakan data *raster*. Untuk itu diperlukan suatu proses yang dapat merubah format data dari *raster* menjadi *vector* dengan bantuan *Image Processing*.

Langkah pertama adalah membuka *layer* data *raster* yang akan dirubah formatnya menjadi *vector*. Contoh gambar yang akan dirubah formatnya dapat dilihat pada Gambar 3.21.



Gambar 3.21 Gambar *Layer* Yang Akan Dirubah ke *Vector*

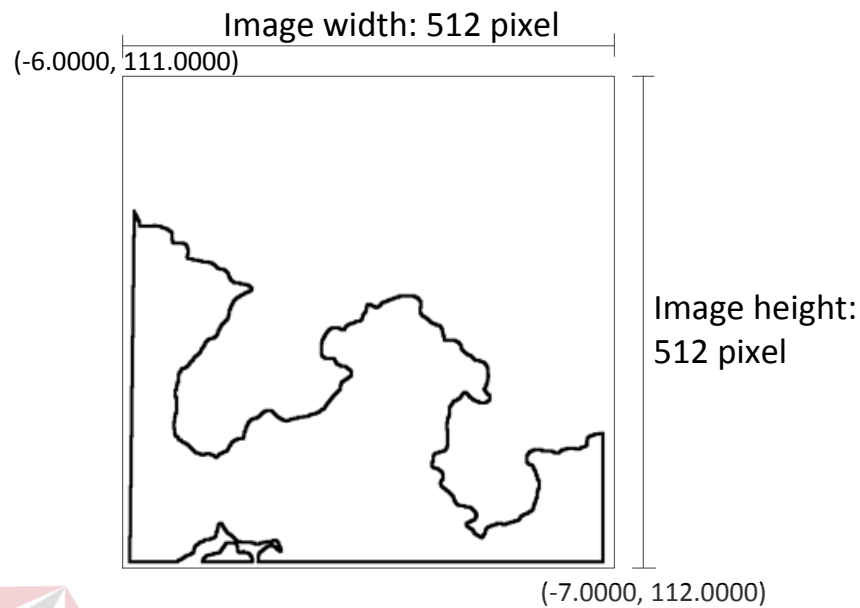
Kemudian dengan bantuan *Image Processing*, *find contour* yang disediakan oleh *library opencv* maka gambar tersebut akan diproses sehingga hasilnya dapat dilihat pada Gambar 3.22.



Gambar 3.22 Hasil Gambar *Layer* Yang Telah Menjadi *Vector*

Proses tidak berhenti sampai disini, bisa dilihat hasil dari proses hasil *vector* masih menggunakan koordinat gambar yaitu koordinat terdekat berada pada posisi 0,0 dan posisi terjauh berada pada posisi (512, 512). Untuk itu perlu dilakukan proses perubahan tiap titik koordinat atau dikenal dengan proses proyeksi. Gambar tersebut akan diproyeksi dengan data koordinat yang berada pada gambar *raster* yang bisa dilihat pada Gambar 3.23. di halaman 46.

Proses proyeksi merubah tiap titik dari hasil *vector* tersebut dan merubah skala dari tiap titik tetapi tidak merubah susunan pola antar titik. Yang berubah hanya skala dan posisi dari tiap titik tersebut. Hasil dari proses proyeksi dapat dilihat pada Gambar 3.23. di halaman 46.

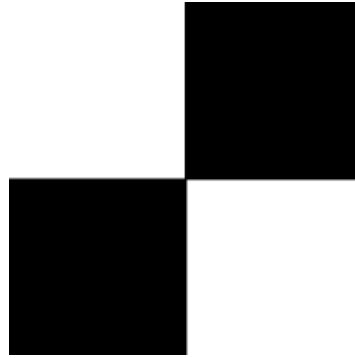


Gambar 3.23 *Vector* Yang Telah Diproyeksikan Dengan Koordinat Geografis

#### 4. Proses *Intersect* pada *binary image*

Proses *intersect* adalah proses penentuan dua daerah yang bersinggungan dimana daerah pertama dan daerah kedua berpotongan di daerah yang sama. Biasanya proses *intersect* dilakukan pada data *vector*. Namun pada sistem informasi geografis ini, proses *intersect* digunakan pada data *raster*. Hal ini karena data *raster* pada sistem informasi geografis ini memiliki data yang lebih presisi sehingga akan lebih akurat apabila masih diproses pada saat masih berupa data *raster*.

Data yang nanti akan dilakukan proses *intersect* adalah data *raster* dari elevasi dan data *raster* dari *surface map*. Pada penjelasan berikut akan digunakan ilustrasi bagaimana proses *intersect* dengan menggunakan data *raster* bisa dilakukan. Contoh data *raster* yang digunakan dalam penjelasan ini dapat dilihat pada Gambar 3.24 dan Gambar 3.25. di halaman 46.

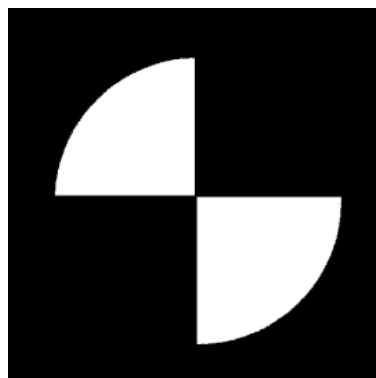


Gambar 3.24 Gambar Pertama Yang Akan Dilakukan Proses *Intersection*



Gambar 3.25 Gambar Kedua Yang Akan Dilakukan Proses *Intersection*

Hasil dari proses *intersect* adalah dengan membandingkan tiap *pixel* dari kedua gambar apabila pada gambar pertama dan gambar kedua warna *pixel* sama dengan putih atau dengan *rgb* (255,255,255) maka pada titik tersebut warna hasil gambar adalah putih. Selain dari pada warna itu maka warna hasil adalah hitam. hasil gambar dari proses *intersect* dapat dilihat pada Gambar 3.26.



Gambar 3.26 Hasil Proses *Intersection*

### 1.2.2 Perancangan UML

Aplikasi ini dibuat dengan menggunakan bahasa pemrograman C++ yang erat kaitannya dengan *object oriented* dan *class*. Untuk itu diperlukan suatu model perancangan yang mampu menangani masalah *object oriented* dan *class*. UML adalah singkatan dari *Unified Modeling Language* merupakan suatu bahasa pemodelan untuk menterjemahkan sistem yang menerapkan *object oriented*.

Menurut Sholih (2010: 18) Notasi UML dibuat sebagai kolaborasi dari beberapa pakar. Para pakar tersebut telah menulis tentang bagaimana mendapatkan persyaratan-persyaratan sistem dalam paket-paket transaksi yang disebut *use case*. UML menyediakan beberapa diagram visual yang menunjukkan berbagai aspek dalam sistem.

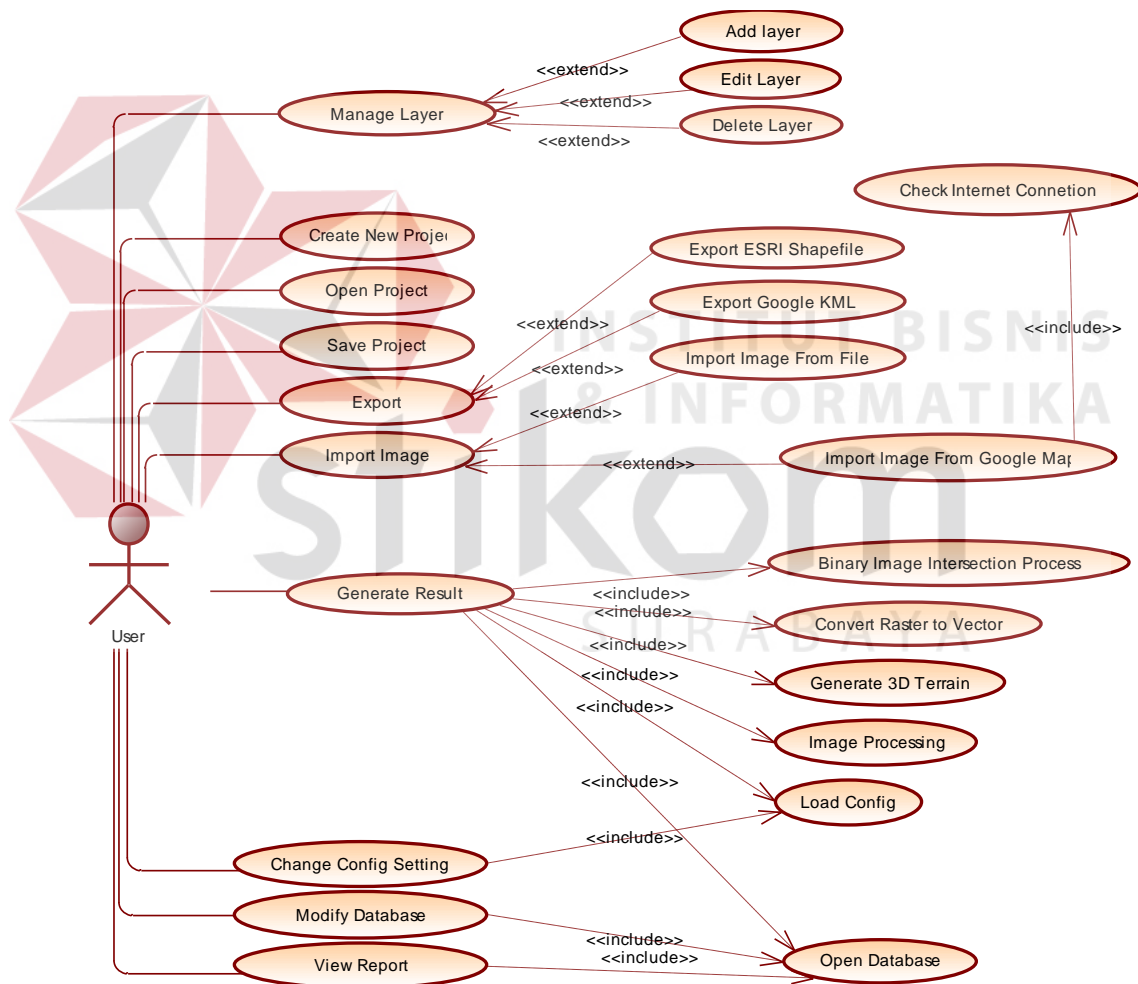
Dalam UML terdapat beberapa diagram yang dapat menjelaskan sistem antara lain: *Use Case Diagram*, *Interaction Diagram*, *Class Diagram*, *Sequence Diagram*, *Collaboration Diagram*, *Statechart Diagram*, *Component Diagram*, dan *Development Diagram*. Namun pada laporan ini hanya empat diagram saja yang digunakan yaitu:

1. *Use Case Diagram*
2. *Activity Diagram*
3. *Sequence Diagram*
4. *Class Diagram*

Berikut adalah penjelasan masing-masing diagram pada perancangan Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing*.

### A. Use Case Diagram

*Use Case* diagram menunjukkan beberapa *use case* dalam sistem, beberapa aktor dalam sistem dan relasi antar mereka. *Use case* adalah potongan fungsionalitas tingkat tinggi yang disediakan oleh sistem aktor adalah seorang atau sesuatu yang berinteraksi terhadap sistem yang akan dibangun *Use Case Diagram* Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing* dapat dilihat pada Gambar 3.27.



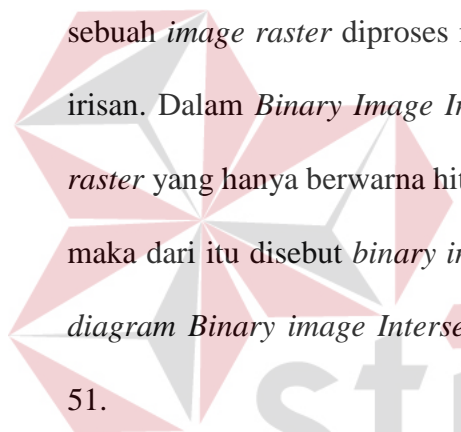
Gambar 3.27 *Use Case Diagram* Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing*

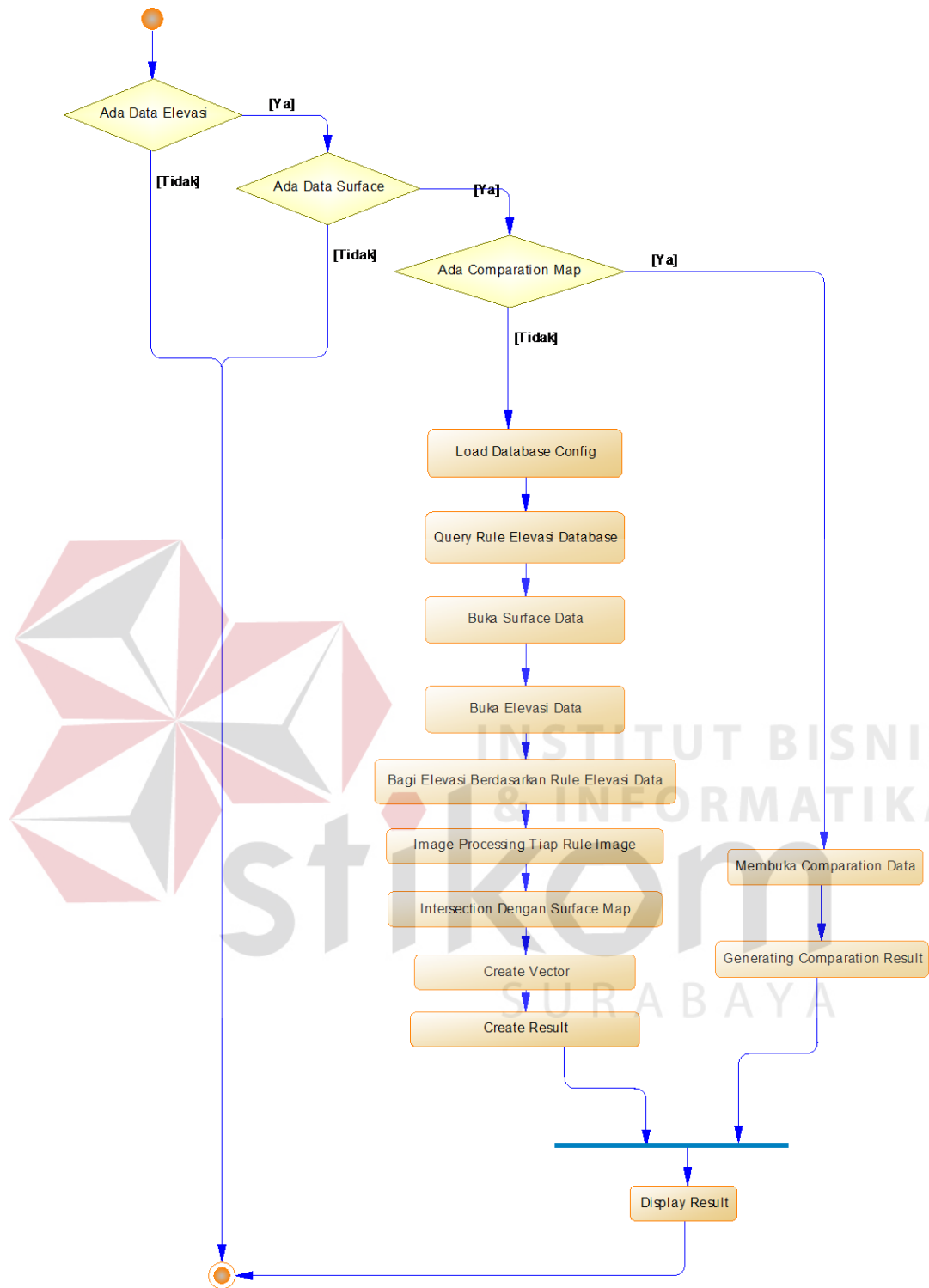
### **B. Activity Diagram Generating Potensial Area**

*Activity diagram* menunjukkan aktifitas yang dilakukan sebuah *use case* pada sistem, tahap-tahap apa saja yang dilakukan oleh *use case* tersebut dengan menjalankan beberapa sub proses. *Activity Diagram Generating Potensial Area* Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing* dapat dilihat pada Gambar 3.28. di halaman 51.

### **C. Activity Diagram Binary Image Intersection**

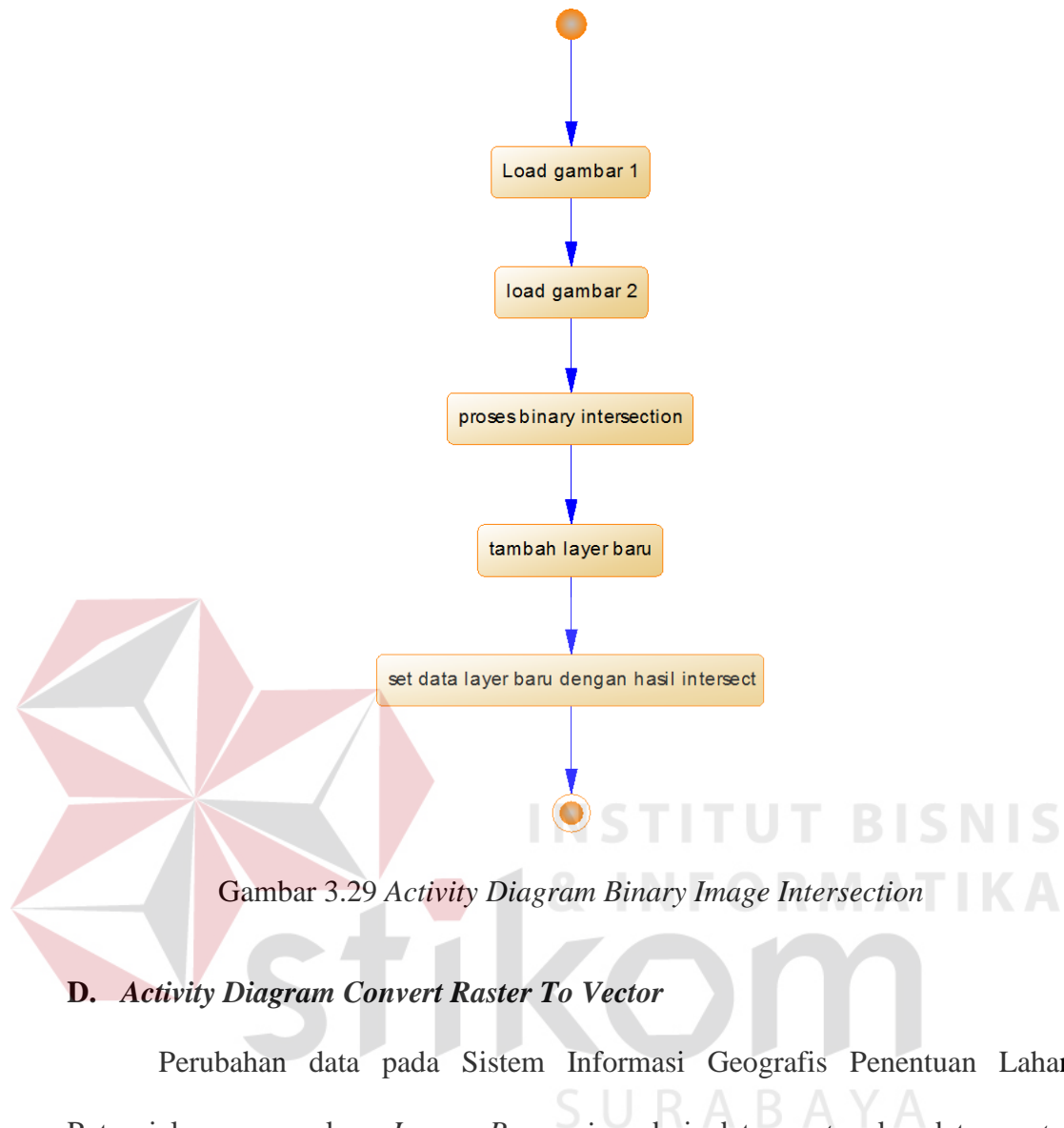
*Activity Diagram Binary Image Intersection* menunjukkan bagaimana sebuah *image raster* diproses menyerupai geometri yaitu proses *intersection* atau irisan. Dalam *Binary Image Intersection* gambar yang digunakan adalah gambar *raster* yang hanya berwarna hitam dan putih. Hitam bernilai 0 dan putih bernilai 1, maka dari itu disebut *binary image* karena hanya memiliki data 0 atau 1. *Activity diagram Binary image Intersection* dapat dilihat pada Gambar 3.29. di halaman 51.





Gambar 3.28 Activity Diagram Generating Potential AreaResult

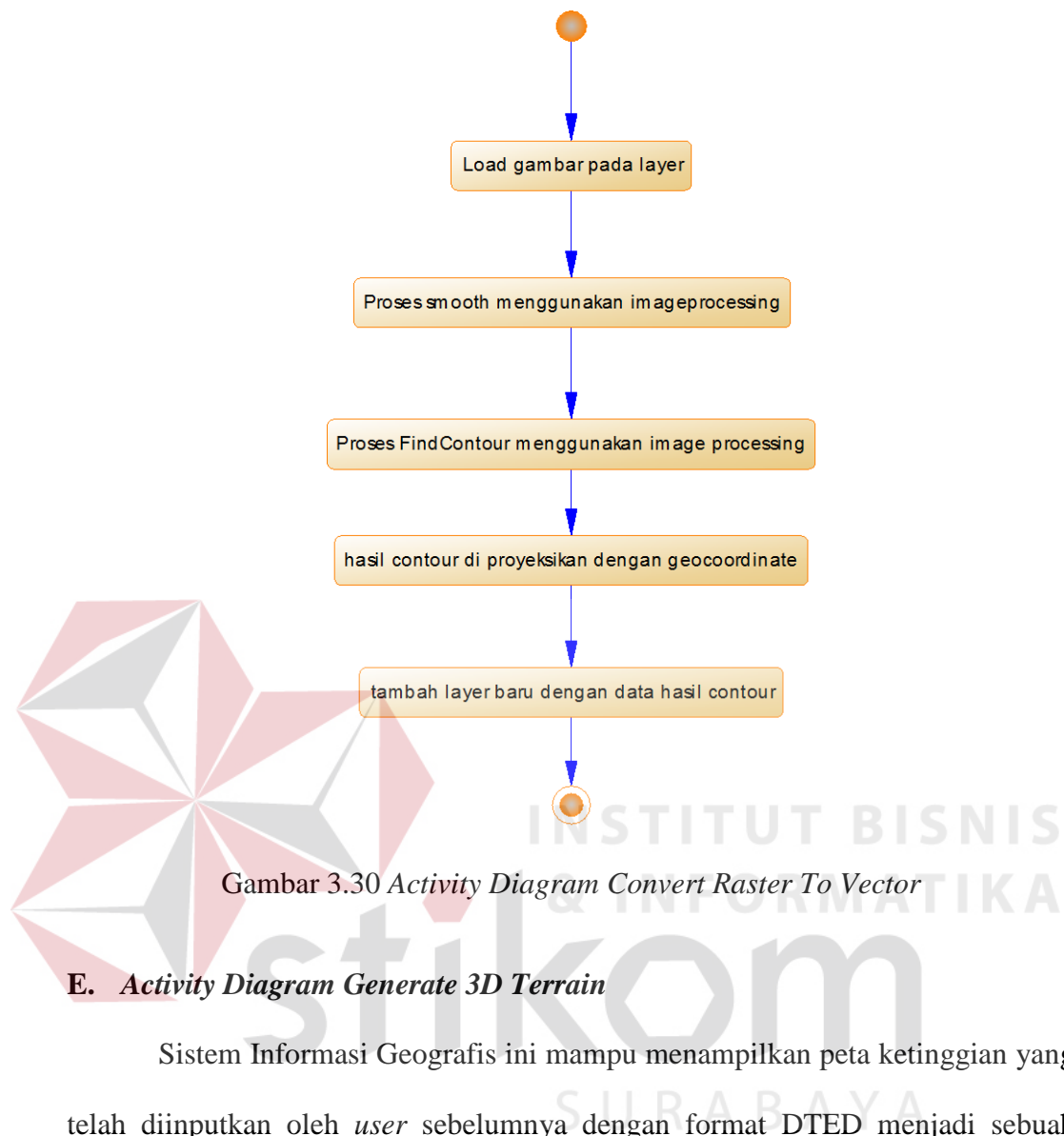




Gambar 3.29 Activity Diagram Binary Image Intersection

#### D. Activity Diagram Convert Raster To Vector

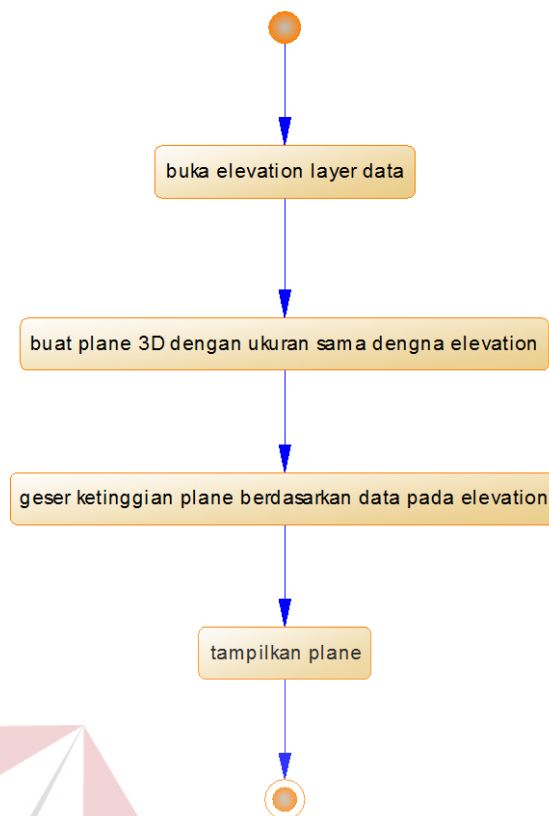
Perubahan data pada Sistem Informasi Geografis Penentuan Lahan Potensial menggunakan *Image Processing* dari data *raster* ke data *vector* memerlukan beberapa tahapan. Setiap data *raster* yang diinputkan pada sistem ini semua akan dirubah menjadi data *vector*. *Activity Diagram* untuk *Convert raster to Vector* dapat dilihat pada Gambar 3.30. di halaman 53.



Gambar 3.30 Activity Diagram Convert Raster To Vector

#### E. Activity Diagram Generate 3D Terrain

Sistem Informasi Geografis ini mampu menampilkan peta ketinggian yang telah diinputkan oleh *user* sebelumnya dengan format DTED menjadi sebuah *terrain* tiga dimensi. Ada beberapa aktifitas yang dilakukan oleh sistem untuk merubah peta ketinggian menjadi *terrain* tiga dimensi. Activity Diagram Generate 3D terrain dapat dilihat pada Gambar 3.31. di halaman 54.



Gambar 3.31 Activity Diagram Generate 3D Terrain

#### F. Activity Diagram Import From Google Maps

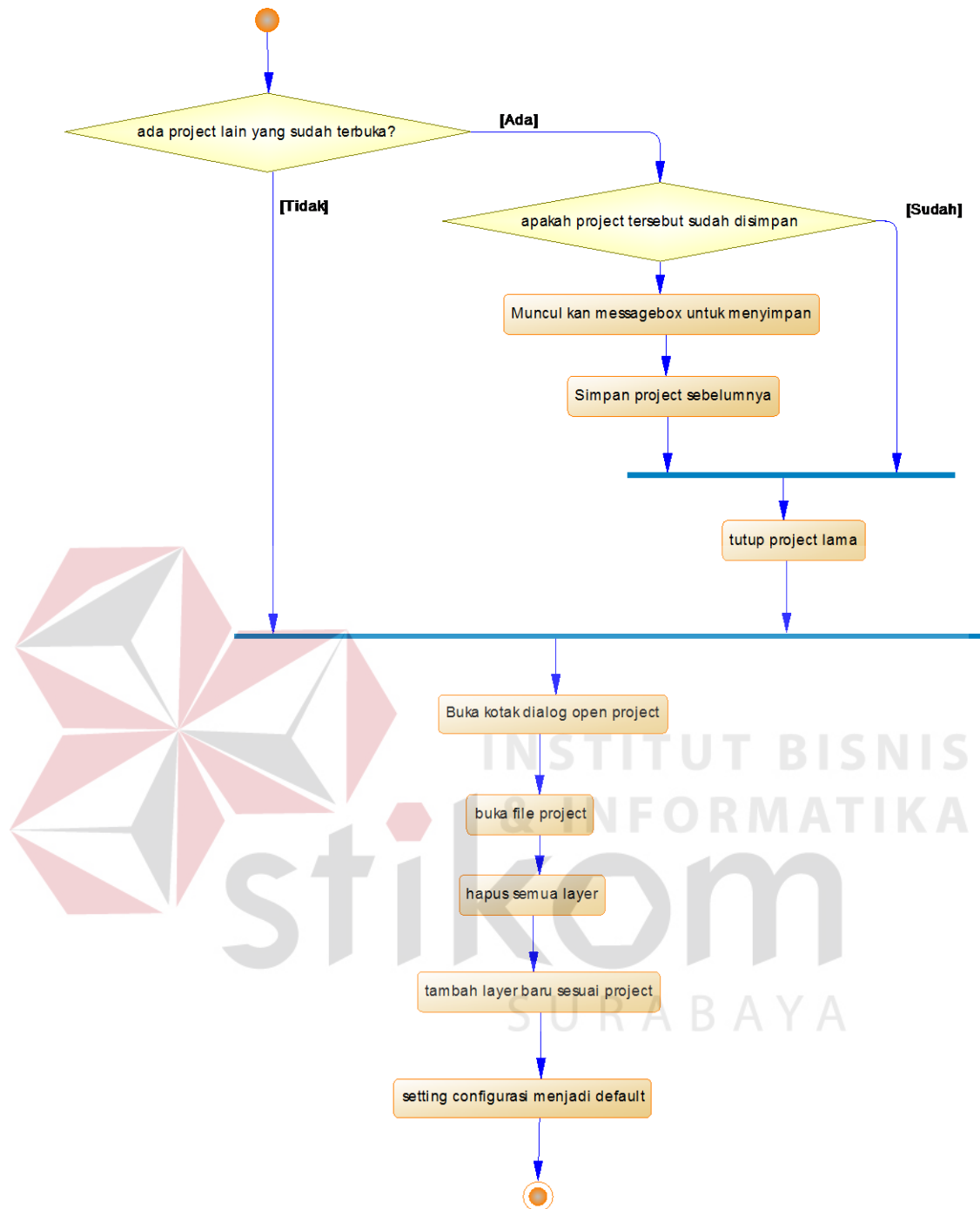
Peta permukaan atau *surface map* adalah syarat kedua sebelum dapat melakukan analisa daerah potensial pada sistem informasi ini. *User* dapat memberi *input* peta permukaan bumi berupa gambar, tetapi untuk mencegah kesalahan dalam memasukkan gambar. Sistem memiliki fitur yang dapat langsung mengunduh gambar dari *Google maps* disertai dengan titik koordinat dari gambar yang telah diunduh. Fungsi ini hanya akan dapat berjalan apabila sudah terhubung dengan Internet. *Activity Diagram Import From GoogleMaps* dapat dilihat pada Gambar 3.32 di halaman 55.



Gambar 3.32 Activity Diagram Import Google Maps

### G. Activity Diagram Open Project

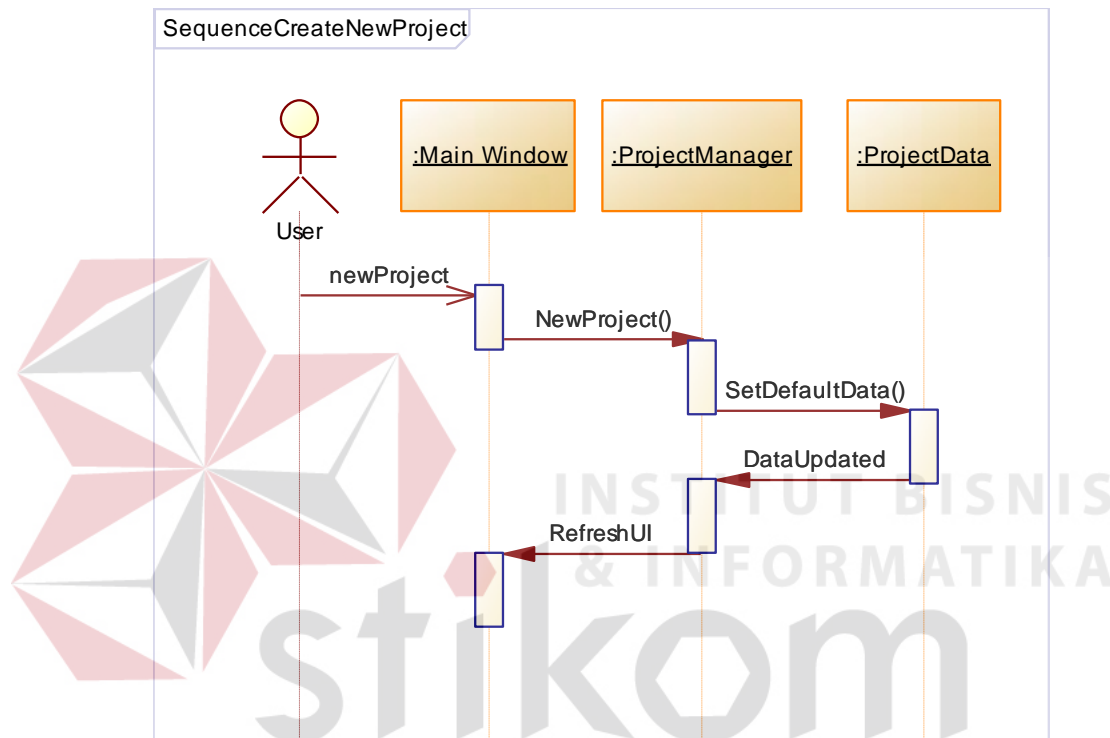
Sistem Informasi ini menerapkan *workspace* dimana tiap *workspace* dapat berisi beberapa *layer* dan pengaturan kerja. Dan setiap *workspace* dapat disimpan kedalam bentuk *file*. Apabila *user* ingin membuka kembali lembar kerjanya. *user* dapat melakukannya dengan cara membuka kembali *file project* yang telah disimpan sebelumnya. *Project* disimpan kedalam format *XML* sehingga memudahkan dalam penggunaannya. *Project* tidak menyimpan semua data *binary* dari *layer*, tetapi *project* hanya menyimpan alamat dari tiap alamat gambar maupun *vector* yang digunakan pada *layer* sebagai sebuah link. Activity Diagram Open Project Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing* dapat dilihat pada Gambar 3.33. di halaman 56.



Gambar 3.33 Activity Diagram Open Project

## H. Sequence Diagram Create New Project

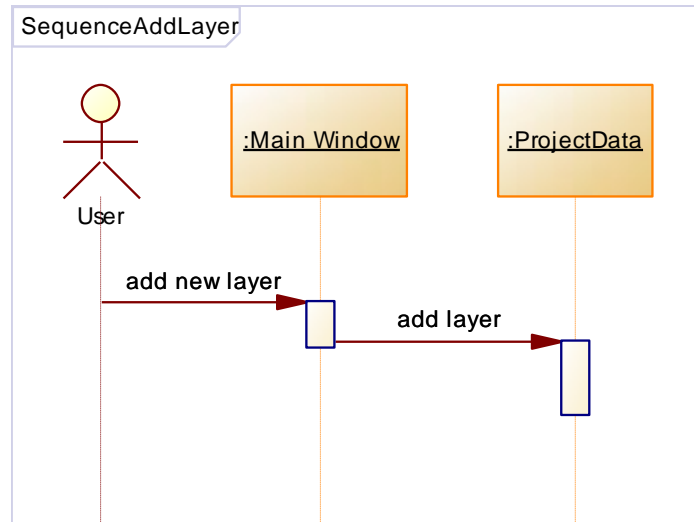
Untuk membuat sebuah *project* baru, Sistem Informasi ini memiliki fitur *Create New Project* yang dapat menutup *project* sebelumnya dan membuat layar kerja menjadi layar kerja yang baru. *Sequence Diagram* Dari *Create New Project* dapat dilihat pada Gambar 3.34.



Gambar 3.34 Sequence Diagram Create New Project

## I. Sequence Diagram Add Layer

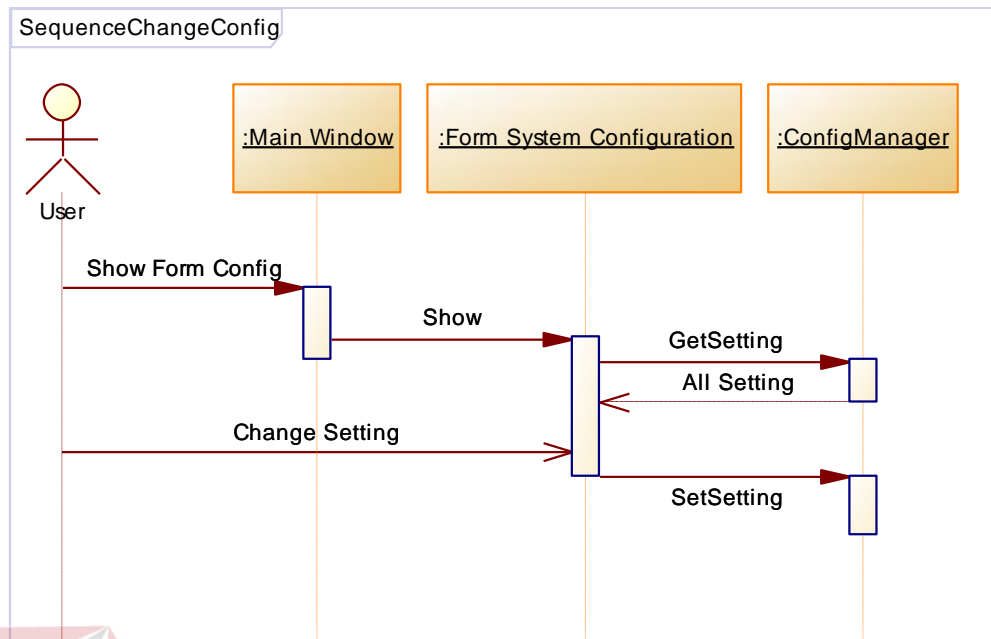
*Layer* merupakan kumpulan dari data peta yang ditumpuk menyerupai lapisan atau *layer*. Pada sistem informasi Geografis Penentuan Lahan potensial Menggunakan *Image Processing* ini, *layer* dapat berupa data *vector*, dan data *raster*. Untuk dapat menambah *layer* baru, *user* dapat melakukannya pada menu yang telah disediakan. *Sequence Diagram Add New layer* dapat dilihat pada Gambar 3.35 di halaman 58.



Gambar 3.35 *Sequence Diagram Add New Layer*

#### J. *Sequence Diagram Change Config*

Konfigurasi merupakan hal yang cukup penting pada Sistem Informasi Geografis ini. Banyaknya penanganan *variable* dari tiap *library* yang digunakan mengharuskan tiap *library* diatur dalam sebuah konfigurasi yang disimpan dalam tabel. Untuk dapat merubah konfigurasi dari Sistem informasi ini, *user* dapat melakukannya melalui fitur *Change Config* yang telah disediakan. *Sequence Diagram Change Config* Sistem Informasi Geografis Penentuan Lahan Potensial dapat dilihat pada Gambar 3.36. di halaman 59.

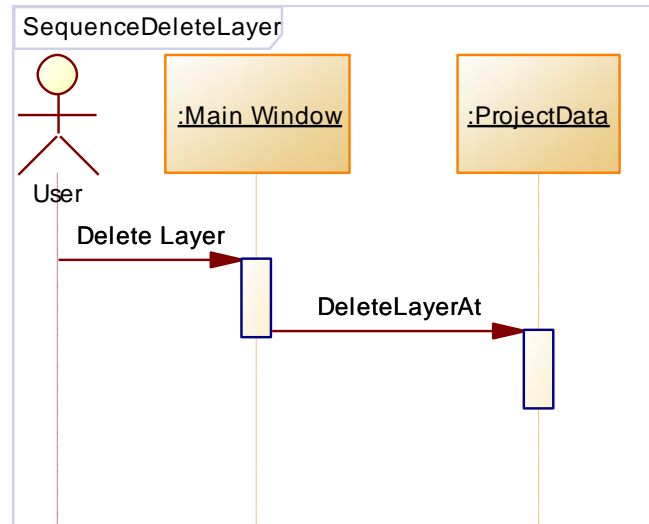


Gambar 3.36 Sequence Diagram Change Config

#### K. Sequence Diagram Delete Layer

Beberapa dari *layer* yang digunakan pada aplikasi mungkin suatu saat tidak akan digunakan kembali oleh *user*. Tentunya *user* harus dapat menghapus *layer* tersebut agar tidak mengganggu dan menghemat penggunaan *memory*. Sistem Informasi Geografis ini memungkinkan *user* untuk dapat melakukan penghapusan pada *layer* yang ada pada layar kerja. *Sequence Diagram Delete layer* dapat dilihat pada Gambar 3.37. di halaman 60.

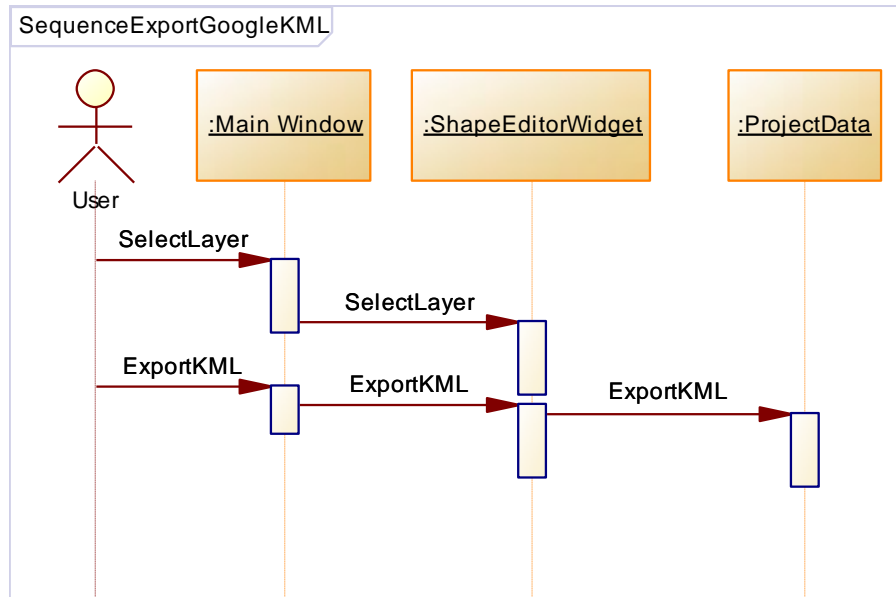




Gambar 3.37 *Sequence Diagram Delete Layer*

#### L. *Sequence Diagram Export Google KML*

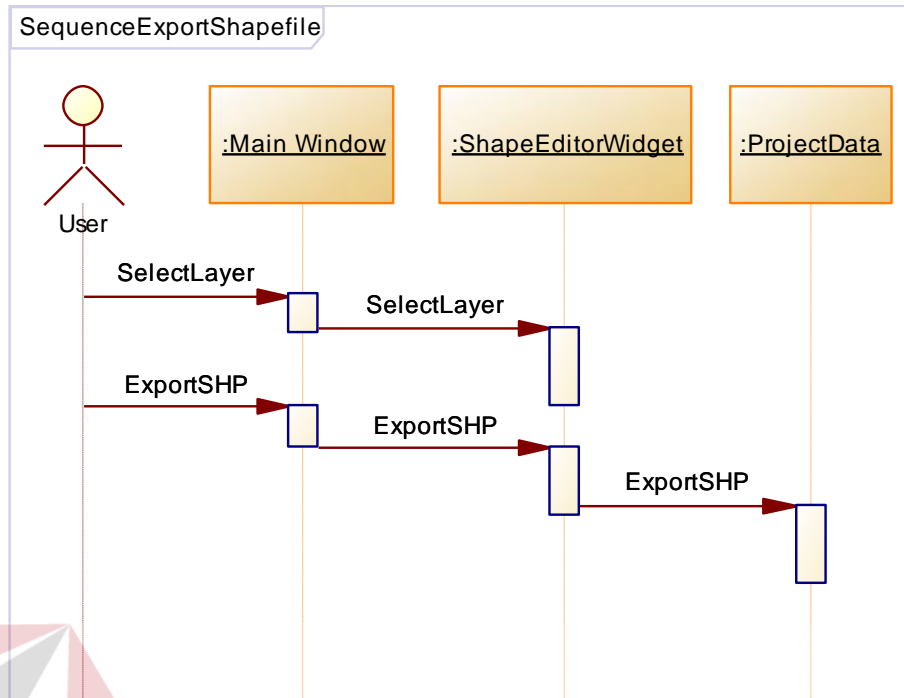
Hasil analisa dari Sistem Informasi Geografis ini adalah berupa data *vector* bukan berupa data *raster*. Selain dapat dilihat dengan menggunakan Sistem Informasi Ini hasil analisa juga dapat disimpan ke dalam format yang berbeda. Salah satu format yang dapat disimpan adalah *Google KML*. Sistem akan dapat menyimpan hasil analisa ke dalam format *Google KML*, sehingga nantinya setelah hasil analisa selesai, hasilnya tidak berhenti sampai disitu saja. *User* dapat memanfaatkan hasil tersebut kedalam program lain yaitu *Google Earth*. Dengan hasil yang dapat disimpan dengan format lain, hasil sistem informasi geografis ini bisa dimanfaatkan untuk kepentingan yang lain. *Sequence Diagram Export Google KML* dapat dilihat pada Gambar 3.38 di halaman 61.



Gambar 3.38 *Sequence Diagram Export Google KML*

### M. *Sequence Diagram Export Shapefile*

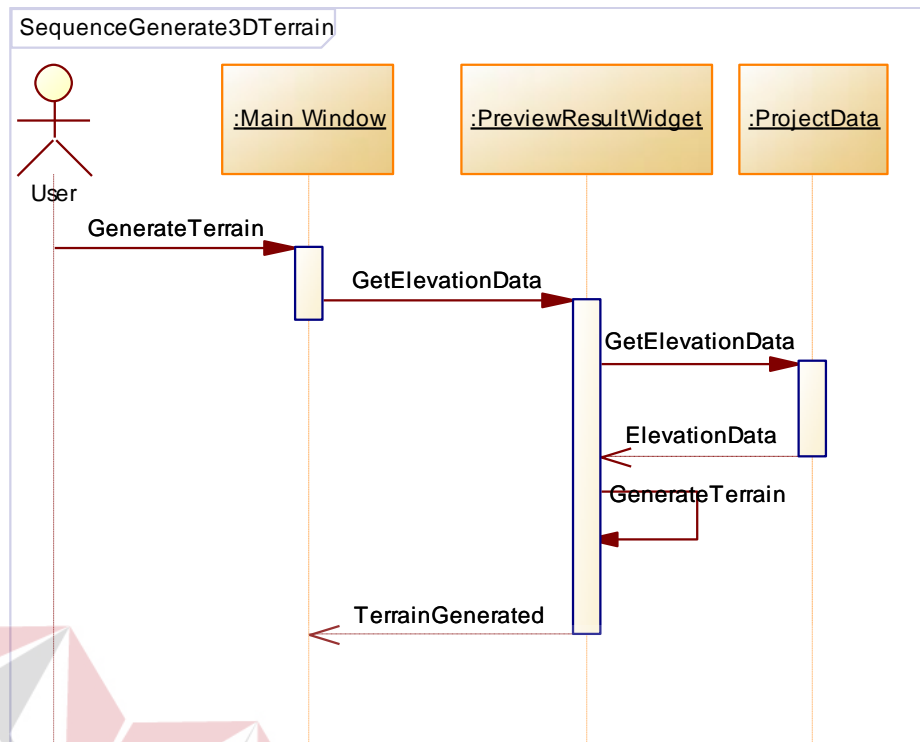
Hasil analisa dari Sistem Informasi Geografis ini adalah berupa data *vector* bukan berupa data *raster*. Selain dapat dilihat dengan menggunakan Sistem Informasi Ini hasil analisa juga dapat disimpan ke dalam format yang berbeda. Salah satu format yang dapat disimpan adalah *ESRI Shapefile*. *ESRI Shapefile* adalah format yang paling sering digunakan dalam aplikasi Sistem Informasi Geografis. Dengan hasil yang dapat disimpan dengan format *Shapefile*, hasil sistem informasi geografis ini bisa dimanfaatkan untuk kepentingan yang lain. *Sequence Diagram Export Google KML* dapat dilihat pada Gambar 3.39. di halaman 62.



Gambar 3.39 *Sequence Diagram Export Shapefile*

#### N. *Sequence Diagram Generate Terrain 3D*

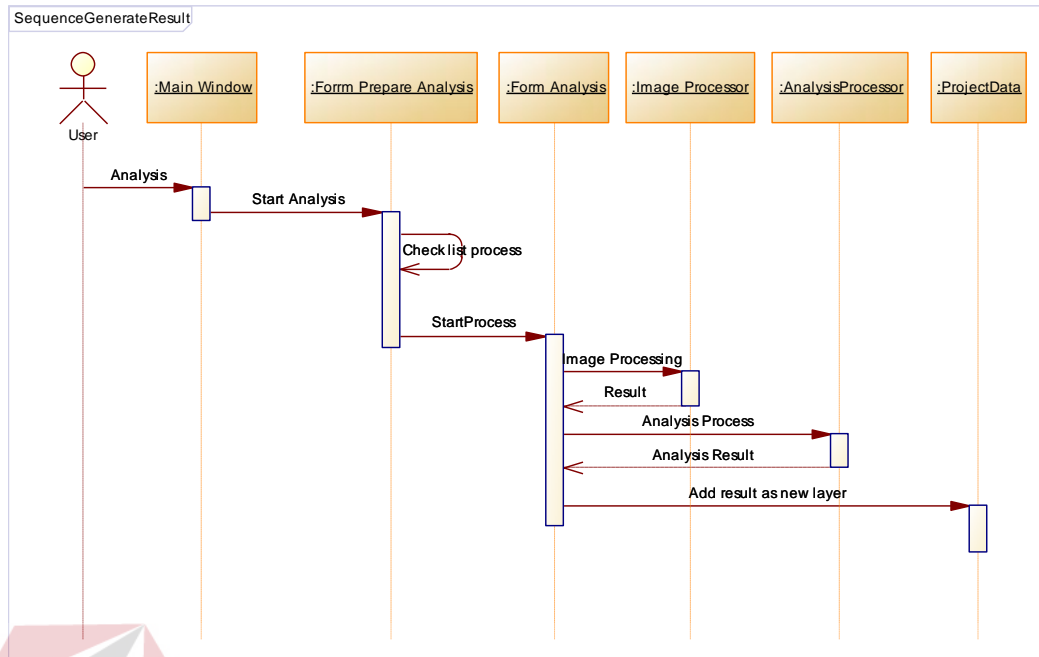
Sistem Informasi Geografis ini mampu memberikan visualisasi tiga dimensi dari peta ketinggian yang telah diinputkan oleh *user* sebelumnya. Dengan visualisasi tiga dimensi, *user* dapat melihat dalam perspektif yang lebih luas dan dapat melihat daerah seperti di dunia nyata. Untuk dapat melakukan visualisasi tiga dimensi diperlukan sebuah proses yaitu melakukan *Generate Terrain* tiga dimensi. Proses ini merubah data *raster* dua dimensi menjadi sebuah *terrain* tiga dimensi yang dapat dilihat dari sisi manapun. *Sequence Diagram Generate Terrain 3D* dapat dilihat pada Gambar 3.40. di halaman 63.



Gambar 3.40 Sequence Diagram Generate Terrain 3D

#### O. Sequence Diagram Generate Result

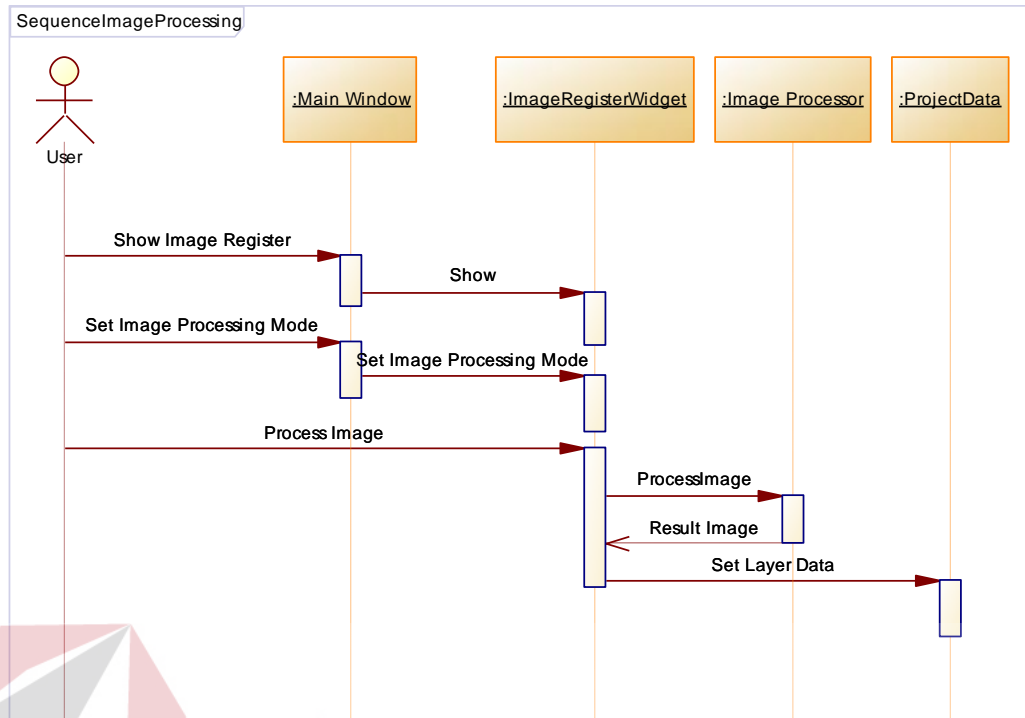
Proses terpenting dalam Sistem Informasi Geografis ini adalah proses *Generate Result* yang menghasilkan daerah potensial dengan menerapkan *Image Processing*. Proses ini diawali dengan memecah peta ketinggian kemudian digabungkan dengan hasil *Image Processing* dari peta permukaan kemudian kedua hasil digabungkan dengan metode *binary image intersection*. Setelah itu hasil dirubah menjadi data *vector*. *Sequence Diagram Generate Result* dari Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing* dapat dilihat pada Gambar 3.41. di halaman 64.



Gambar 3.41 Sequence Diagram Generate Result

#### P. Sequence Diagram Image Processing

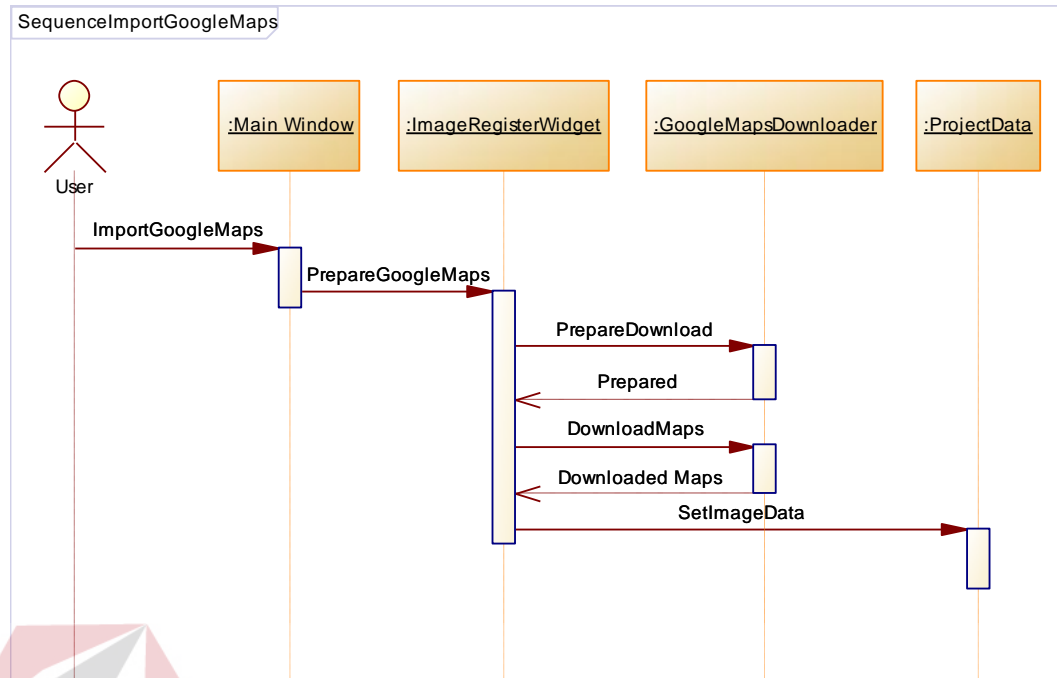
*Image Processing* digunakan pada beberapa proses pada Sistem Informasi Geografis ini. Diantaranya adalah pada proses mencari daerah yang berwarna biru dan proses merubah format dari *raster* ke data *vector*. *User* juga dapat melakukan percobaan *Image Processing* pada menu *image register* yang telah disiapkan. *Image Processing* yang dapat digunakan pada aplikasi ini meliputi *color seletion*, *smooth*, *dilate*, *erode*, dan *find contour*. Semua fitur tersebut diambil dari *library OpenCV*. *Sequence Diagram Image Processing* sistem informasi geografis ini dapat dilihat pada Gambar 3.42 di halaman 65.



Gambar 3.42 *Sequence Diagram Image Processing*

#### Q. *Sequence Diagram Import Google Maps*

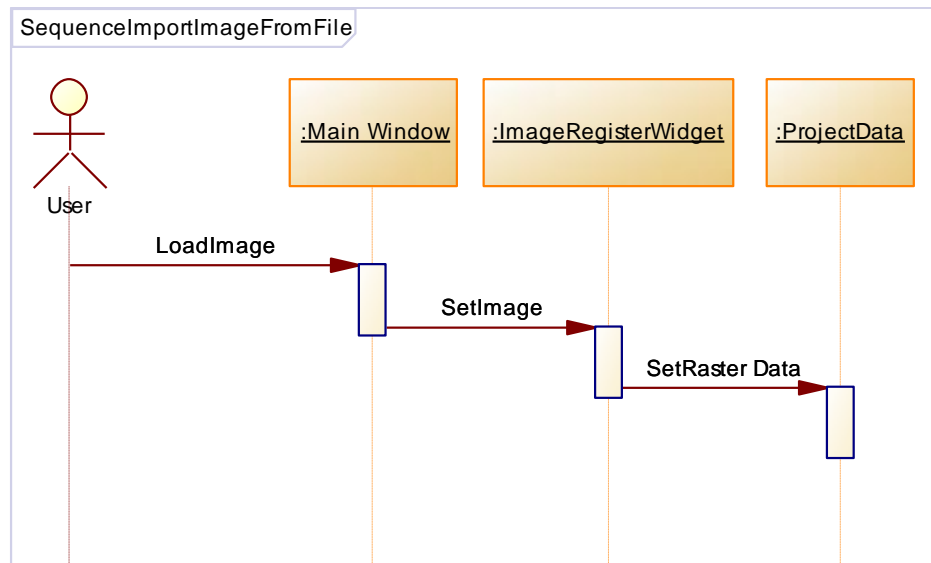
Sistem informasi geografis penentuan lahan potensial menggunakan *Image Processing* ini memiliki fitur untuk menggunakan peta milik *Google Maps*. Selain dari gambar biasa, sistem informasi geografis ini memungkinkan *user* untuk mendownload peta dari *Google Maps* dengan tujuan mengurangi kesalahan pemasukan gambar peta dari gambar. Dengan menggunakan *Google Maps*, *user* hanya perlu menginputkan lokasi daerah berupa titik koordinat geografis, kemudian sistem secara otomatis akan mengunduh daerah yang *user* maksudkan. *Sequence Diagram* untuk *Import Google Maps* dapat dilihat pada Gambar 3.43. di halaman 66.



Gambar 3.43 Sequence Diagram Import Google Maps

#### R. Sequence Diagram Import From File

Untuk menggunakan gambar atau data DTED, *user* harus melakukan proses *import data from file* ke dalam *layer*. Sistem ini memiliki fungsi untuk meng-*import* gambar dari *file* ke dalam *layer*. Sequence Diagram untuk *import from file* dapat dilihat pada Gambar 3.44. di halaman 67.

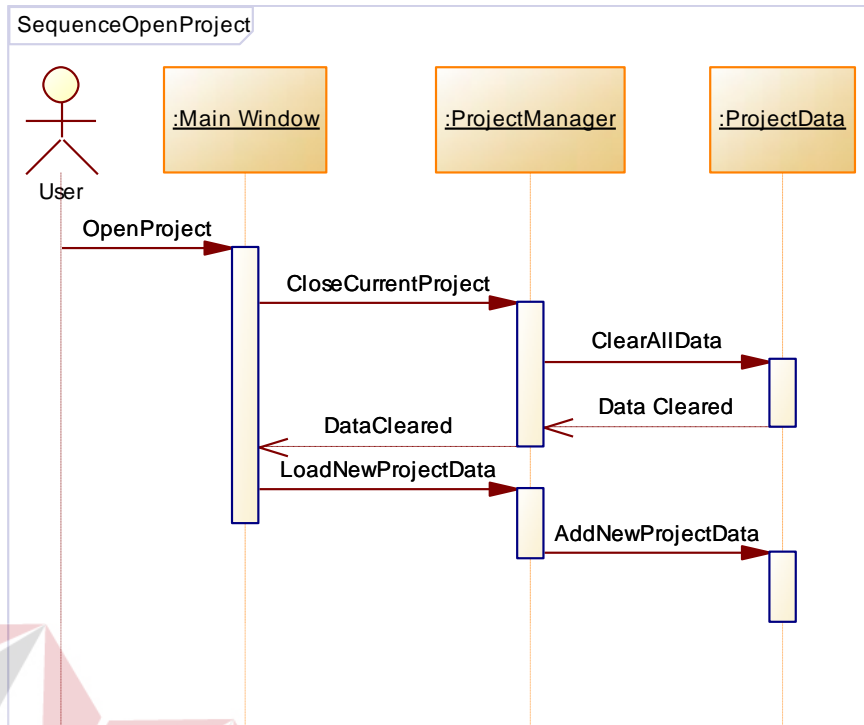


Gambar 3.44 *Sequence Diagram Import From File*

### S. *Sequence Diagram Open Project*

Sistem Informasi Geografis penentuan lahan potensial menggunakan *Image Processing* ini menggunakan konsep *single document interface*, dimana hanya satu *project* yang dapat dikerjakan dalam satu waktu. *Project* yang sedang dikerjakan oleh *user* dapat disimpan ke dalam *file* dengan format *xml* dan dapat dibuka kembali. Ketika *user* membuka *project*, maka semua *layer* dan data yang telah *user* masukkan ke dalam *project* akan terbuka kembali sama seperti sebelumnya. *Project* tersimpan dalam format *XML* seperti dijelaskan pada Gambar 3.45. di halaman 68.

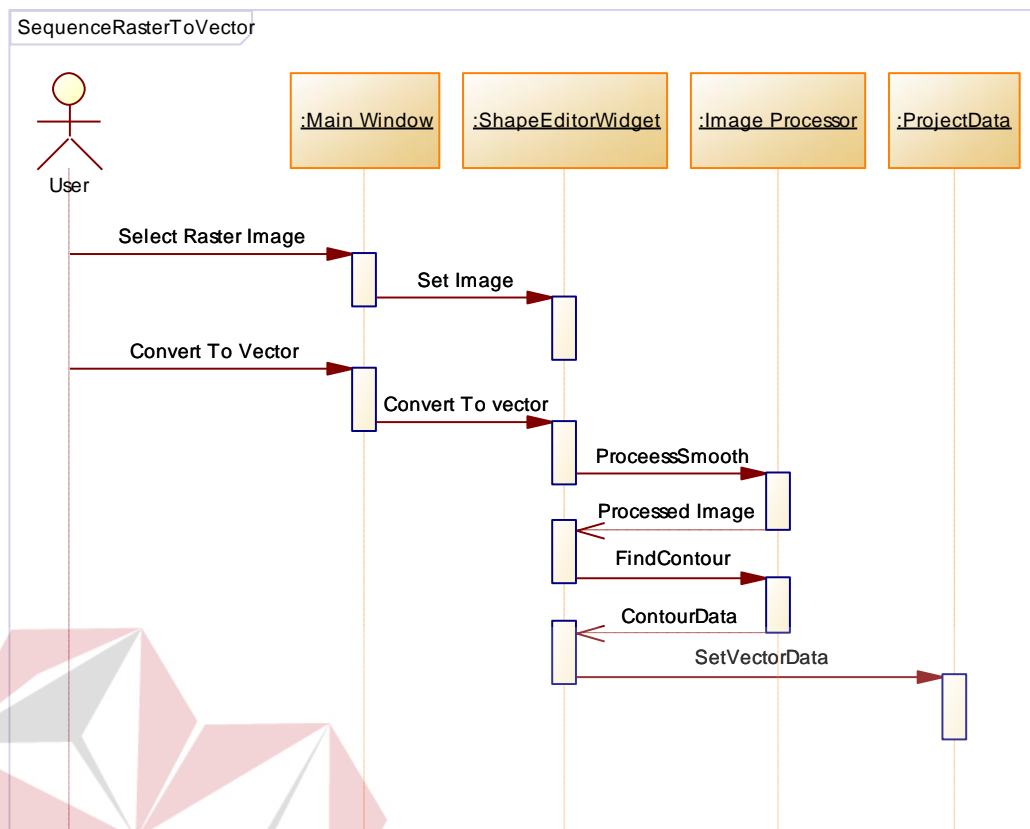




Gambar 3.45 *Sequence Diagram Open Project*

#### T. *Sequence Diagram Convert Raster To Vector*

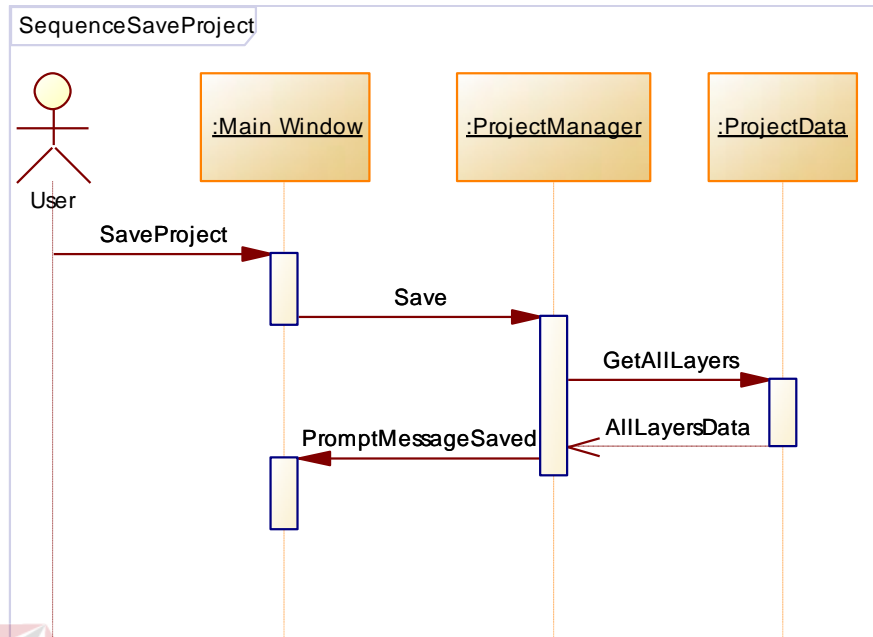
Proses perubahan data dari *raster* menjadi *vector* dilakukan oleh beberapa *class* pada sistem. *User* pertama berinteraksi dengan *main window* kemudian *main window* akan berinteraksi pada *shape editor*. Proses akan memanggil *image processor class* yang memiliki beberapa fitur yang dapat merubah data *raster* menjadi data *vector*. Setelah data dirubah hasil *vector* kemudian akan dimasukkan ke dalam *layer*. *Sequence Diagram Convert Raster To Vector* dapat dilihat pada Gambar 3.46. di halaman 69.



Gambar 3.46 Sequence Diagram Convert Raster To Vector

#### U. Sequence Diagram Save Project

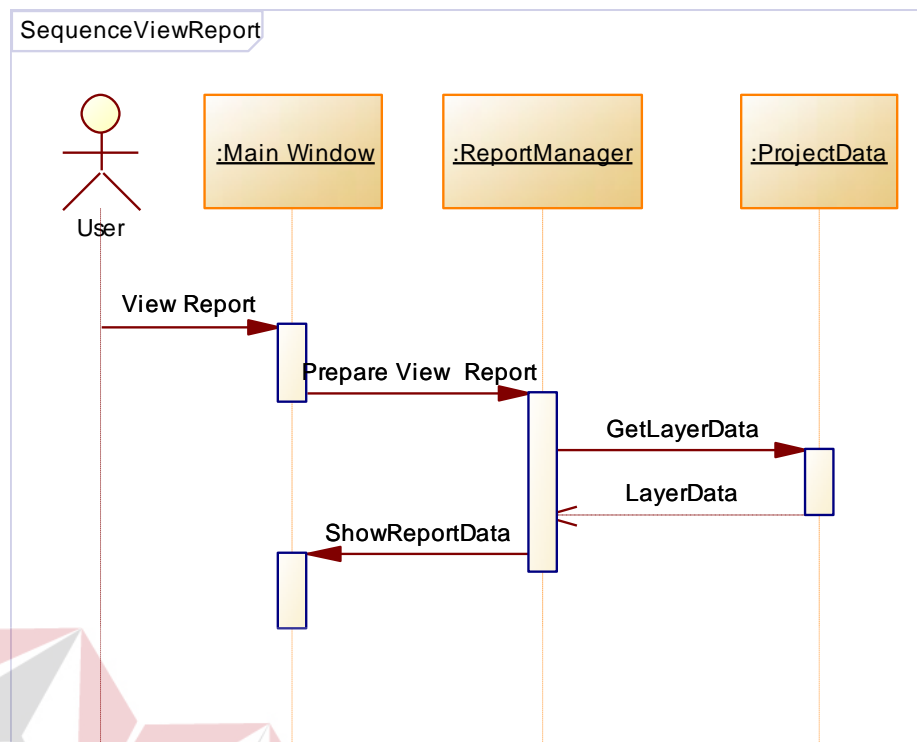
Menyimpan sebuah *project* dalam Sistem Informasi Geografis memerlukan beberapa proses dan melibatkan beberapa *class*. Pada saat *user* ingin melakukan *save project* melalui *main window*, maka *main window* memanggil fungsi *save* pada *class project manager*. *Class* tersebut akan menyimpan semua data yang ada pada *project data*. Mulai dari data seluruh *layer* yang ada pada *workspace*, *data working area* dan data seluruh peta. Seluruh data tersebut akan disimpan kedalam sebuah *file* dengan format *XML*. *Sequence Diagram Save Project* dapat dilihat pada Gambar 3.47. di halaman 70.



Gambar 3.47 Sequence Diagram Save Project

## V. Sequence Diagram View Report

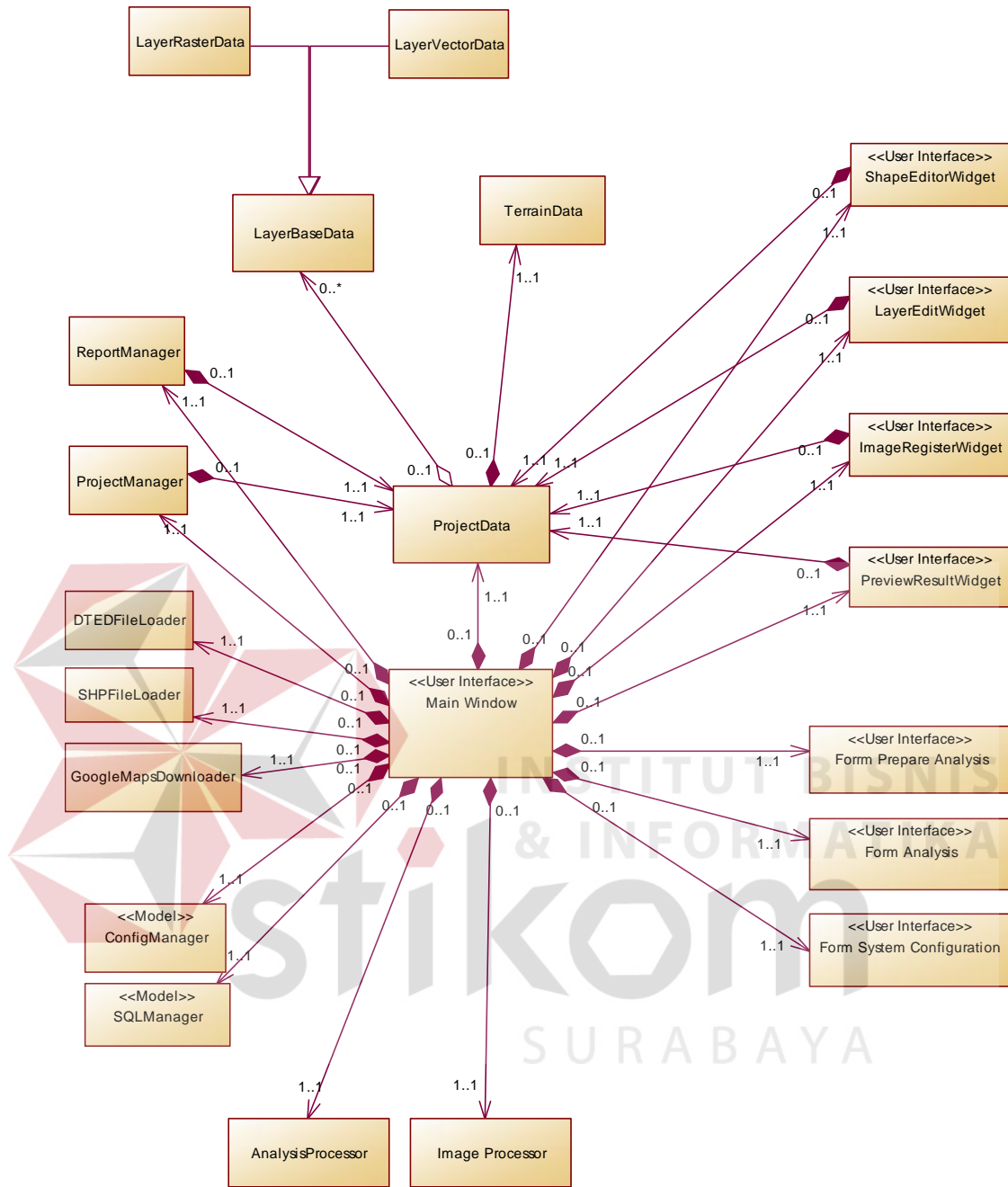
Pembuatan laporan atau *report* sangat penting bagi pihak *managerial*. Karena dengan adanya *report*, pihak *managerial* dapat mengambil keputusan yang tepat. *Report* dalam Sistem Informasi Geografis ini mengambil seluruh data dari *Project Data*. Data penting yang diambil adalah data peta ketinggian atau *elevation data*, data permukaan bumi atau *surface map* dan hasil analisa. Apabila *user* menginputkan *query*, maka *report* akan menambah satu halaman yang berisi hasil perhitungan *geoprocessing query* yang telah diinputkan oleh *user* sebelumnya dengan data hasil analisa penentuan lahan potensial oleh sistem. Proses pembuatan *report* dapat dilihat pada *sequence diagram* pembuatan *report*. *Sequence Diagram View Report* dapat dilihat pada Gambar 3.48. di halaman 71.



Gambar 3.48 Sequence Diagram View Report

### W. Class Diagram

*Class diagram* digunakan untuk menampilkan kelas-kelas atau paket-paket dalam sistem dan relasi antar mereka. *Class* digunakan pada perancangan pembuatan sistem informasi geografis ini karena aplikasi ini akan menerapkan *object oriented programming*. *Qt* sebagai salah satu *framework* bahasa pemrograman *C++* menerapkan *class* dalam pembuatan *form*, *modul*, *database engine* dan lain-lain, sehingga untuk dapat membuat aplikasi yang menggunakan *framework Qt*, perancangan dari aplikasi ini juga menerapkan konsep *class*. *Class Diagram* Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing* dapat dilihat pada Gambar 3.49 di halaman 72.



Gambar 3.49 *Class Diagram* Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing*

### 1.2.3 Struktur Tabel

Struktur tabel merupakan penjabaran dan penjelasan dari suatu *database*. Dalam struktur tabel dijelaskan fungsi dari masing-masing tabel hingga fungsi masing-masing *field* yang ada di dalam tabel. Selain itu juga terdapat tipe data dari masing-masing field beserta konstrainnya.

#### 1. Tabel Tanaman

Primary Key : ID\_Tanaman  
 Foreign Key : -  
 Fungsi : Menyimpan data tanaman

Tabel 3.1 Struktur Tabel Tanaman

Field	Tipe	Ukuran	Keterangan
ID_Tanaman	Varchar	50	Primary Key
Nama	Varchar	100	-
ID_Katagori	Varchar	100	-

#### 2. Tabel KarakterLahan

Primary Key : ID\_Karakter  
 Foreign Key : ID\_Tanaman  
 Fungsi : Menyimpan data karakteristik lahan yang cocok untuk tiap jenis tanaman

Tabel 3.2 Struktur Tabel Karakter Lahan

Field	Tipe	Ukuran	Keterangan
ID_Karakter	Varchar	100	Primary Key
ID_Tanaman	Varchar	50	Foreign Key
Temperatur	Varchar	100	
CurahHujan	Varchar	100	
Kelembaban	Varchar	100	

#### 3. Tabel Kategori

Primary Key : ID\_Kategori  
 Foreign Key : -  
 Fungsi : Menyimpan data kategori yang dimiliki tiap tanaman

Tabel 3.3 Struktur Tabel Kategori

Field	Tipe	Ukuran	Keterangan
ID_Kategori	Varchar	100	Primary Key
NamaKategori	Varchar	50	

#### 4. Tabel Configuration

Primary Key : -  
 Foreign Key : -  
 Fungsi : Menyimpan konfigurasi untuk menjalankan aplikasi

Tabel 3.4 Struktur Tabel Configuration

Field	Tipe	Ukuran	Keterangan
ConfigName	Varchar	100	
ConfigValue	Varchar	500	
ConfigType	Varchar	50	

#### 5. Tabel Sector

Primary Key : SectorID  
 Foreign Key : -  
 Fungsi : Menyimpan data sector berupa alamat *file*.

Tabel 3.5 Struktur Tabel Sector

Field	Tipe	Ukuran	Keterangan
SectorID	Integer		Primary Key
SectorName	Varchar	20	
Filepath	Varchar	300	

#### 6. Tabel SectorTanaman

Primary Key : ID\_SectorTanaman  
 Foreign Key : SectorID, ID\_Tanaman  
 Fungsi : Menyimpan tanaman yang terdapat di tiap sector

Tabel 3.6 Struktur Tabel SectorTanaman

Field	Tipe	Ukuran	Keterangan
ID_SectorTanaman	Integer		Primary Key
SectorID	Integer		Foreign Key
ID_Tanaman	Varchar	50	Foreign Key

## 7. Tabel Kabupaten

Primary Key : KabupatenID  
 Foreign Key :  
 Fungsi : Menyimpan nama kabupaten

Tabel 3.7 Struktur Tabel Kabupaten

Field	Tipe	Ukuran	Keterangan
KabupatenID	Integer		Primary Key
NamaKabupaten	Varchar	200	

## 8. Tabel Kecamatan

Primary Key : KecamatanID  
 Foreign Key :  
 Fungsi : Menyimpan nama kecamatan

Tabel 3.8 Struktur Tabel Kecamatan

Field	Tipe	Ukuran	Keterangan
KecamatanID	Integer		Primary Key
KabupatenID	Integer		Foreign Key
NamaKabupaten	Varchar	200	

## 9. Tabel DetailSectorTanaman

Primary Key :  
 Foreign Key : ID\_SectorTanaman, KabupatenID, KecamatanID  
 Fungsi : Menyimpan nama kecamatan

Tabel 3.9 Struktur Tabel DetailSectorTanaman

Field	Tipe	Ukuran	Keterangan
ID_SectorTanaman	Integer		Foreign Key
KabupatenID	Integer		Foreign Key
KecamatanID	Integer		Foreign Key

### 1.2.4 Desain *Input Output*

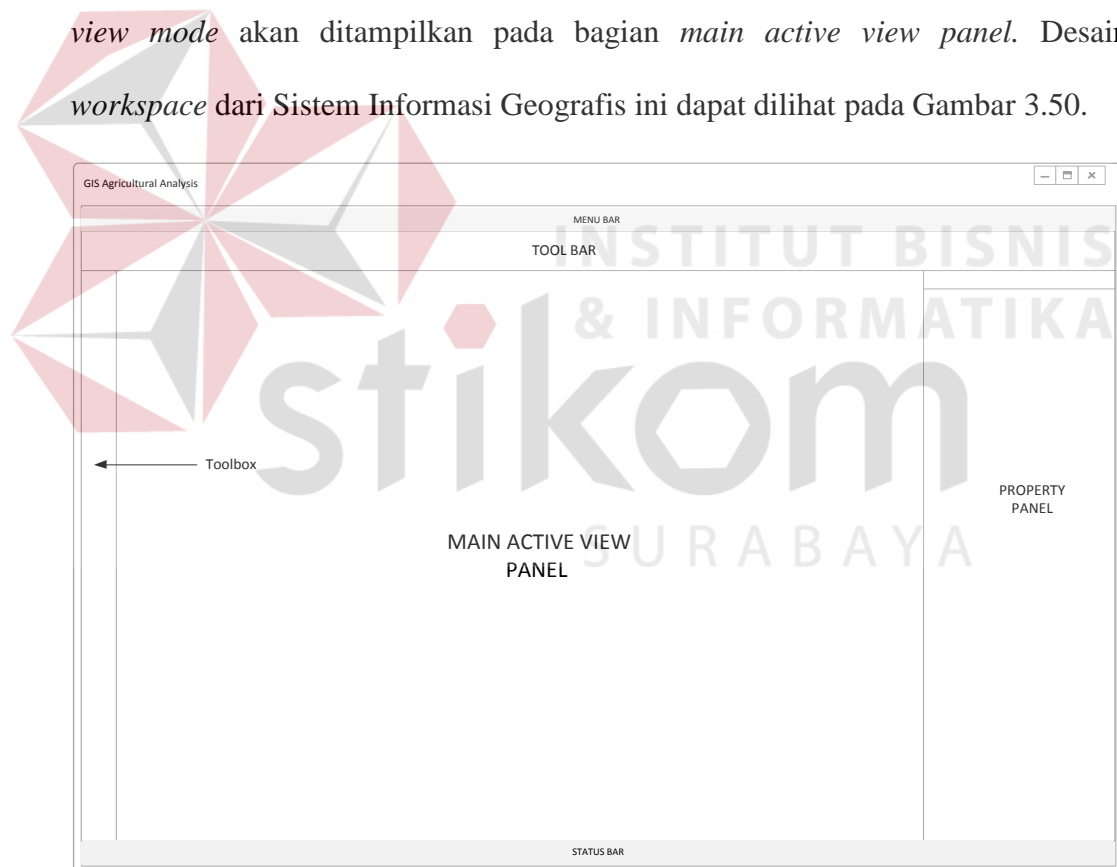
Setelah melakukan perancangan basis data, tahap selanjutnya adalah membuat desain *input/output*. Desain ini berguna untuk menggambarkan interaksi antara pengguna dengan sistem yang dibuat. Secara keseluruhan sistem informasi geografis ini menerapkan konsep *single document interface*, dimana hanya satu



proyek yang dapat dibuka dalam satu waktu. Desain *input output* dilakukan dengan menggunakan program *Microsoft Visio 2010* karena gambar yang dihasilkan oleh aplikasi ini mendekati gambar *input output* aplikasi sesungguhnya.

### 1. Desain *Workspace Form*

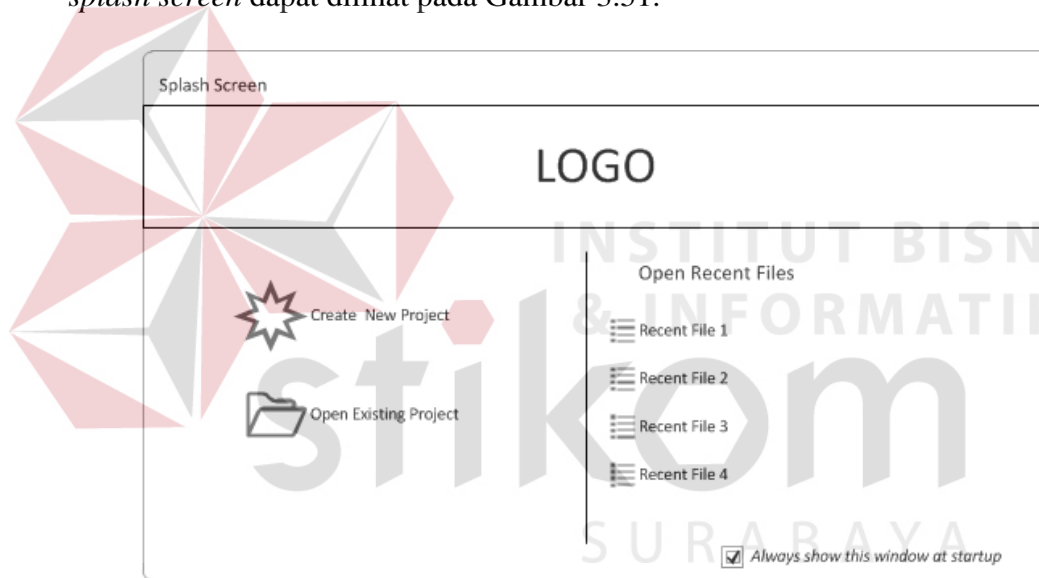
Sistem informasi geografis ini memiliki 6 bagian utama yaitu: *menubar*, *toolbar*, *toolbox*, *status bar*, *property panel*, dan *main active view panel*. Dalam aplikasi ini terdapat 4 buah *view* yang memiliki fungsi tersendiri yaitu: *layer edit mode*, *image register mode*, *shape editor mode*, dan *preview result mode*. Tiap *view mode* akan ditampilkan pada bagian *main active view panel*. Desain *workspace* dari Sistem Informasi Geografis ini dapat dilihat pada Gambar 3.50.



Gambar 3.50 Desain workspace

## 2. Desain *Splash Screen Form*

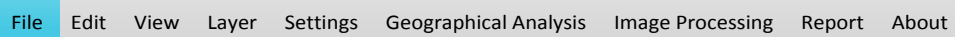
*Splash screen* merupakan *form* yang pertama kali muncul pada saat aplikasi pertama kali dibuka, dan *splashscreen* akan muncul seperti *popup window*. Pada *splashscreen* terdapat beberapa menu yang memudahkan *user* dalam menggunakan aplikasi. Pada sisi kiri *user* dapat membuat *project* baru atau membuka *file project* pada *harddrive*. Sedangkan pada sisi kanan terdapat list *file-file project* yang sudah pernah dibuka oleh *user* sebelumnya. Sehingga *user* dapat membuka *project* yang telah dibuka sebelumnya dengan cepat. Desain *Form splash screen* dapat dilihat pada Gambar 3.51.



Gambar 3.51 Desain *Form Splashscreen*

## 3. Desain *Menu Bar*

*Menu bar* adalah deretan menu yang berada paling atas. Dalam *menu bar* ini terdapat sekumpulan menu yang dikelompokkan pada menu bar yang paling atas. Berikut ini adalah menu bar dari aplikasi. Desain dari menu bar dapat dilihat pada Gambar 3.52 di halaman 78.

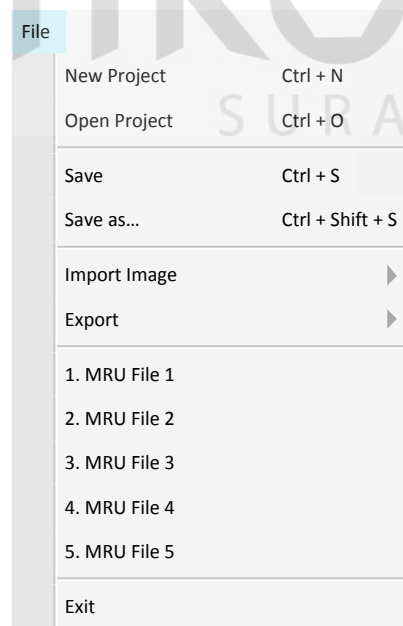


Gambar 3.52 Desain *Menu Bar*

Berikut ini akan dijelaskan isi masing–masing dari kesembilan menu yang terdapat pada menu bar sesuai dengan yang dapat dilihat pada Gambar 3.52.

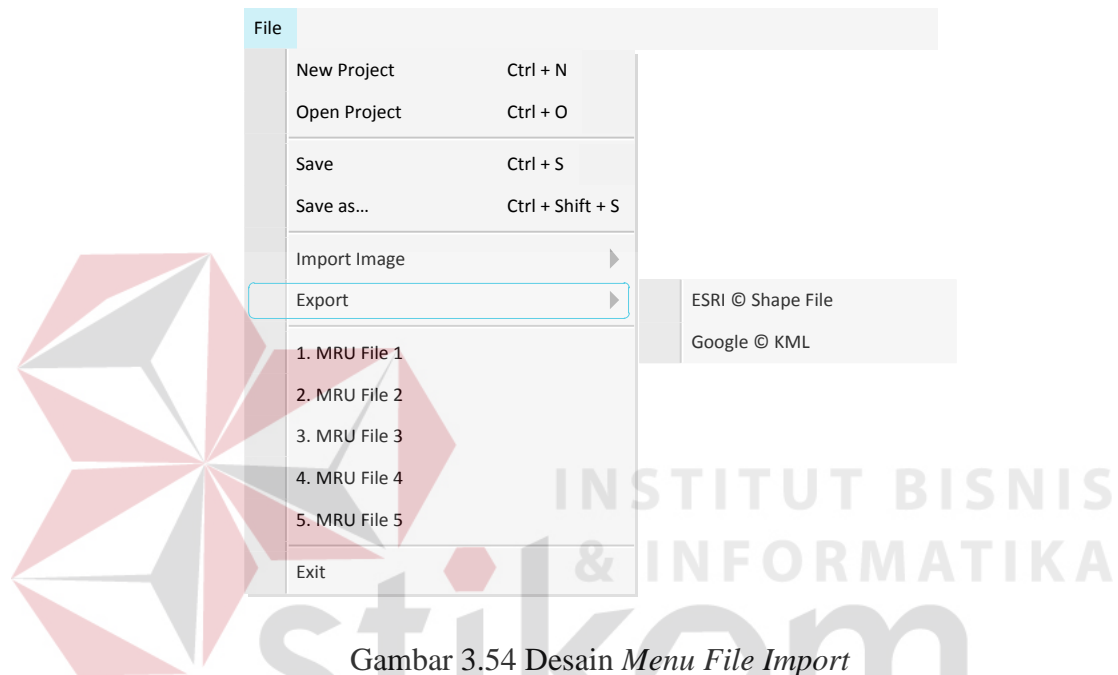
#### 4. Desain *Menu File*

Pada aplikasi terdapat menu, salah satunya adalah menu *file*. Menu *file* disini menu yang memiliki fungsi yang berhubungan dengan *file*. *User* dapat membuat *project* baru, membuka *project* dan menyimpan. Fitur *import* berfungsi untuk mengambil gambar dari *file* di hardisk untuk ditampilkan di *layer* atau dapat langsung mendownload dari Google Maps. Paling bawah terdapat *MRU (Most recent used file)* yaitu kumpulan *project* yang terakhir kali di buka oleh *user*. Sehingga *user* dapat membuka *project* yang telah dibuka sebelumnya dengan cepat. Desain Menu *file* dapat dilihat pada Gambar 3.53.

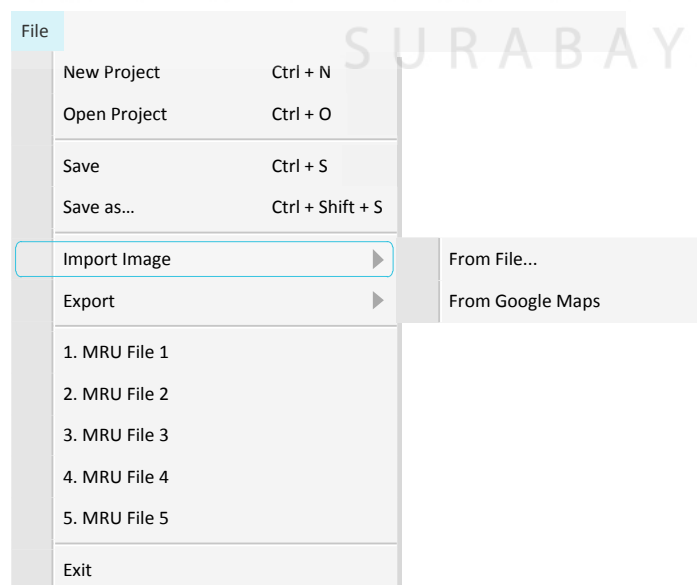


Gambar 3.53 Desain *Menu File*

*File* menu memiliki 2 (dua) sub menu yaitu pada menu *Import image* dan menu *export*. Sub menu tersebut memiliki menu tersendiri. Sub menu *import image* dapat dilihat pada Gambar 3.57 dan sub menu *export* dapat dilihat pada Gambar 3.54. Sub menu dari *import file* pada menu *file* dapat dilihat pada Gambar 3.55.



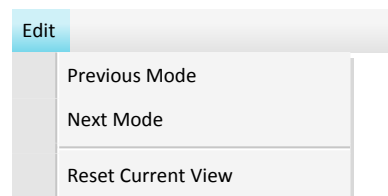
Gambar 3.54 Desain Menu File Import



Gambar 3.55 Desain Menu File Export

## 5. Desain Menu Edit

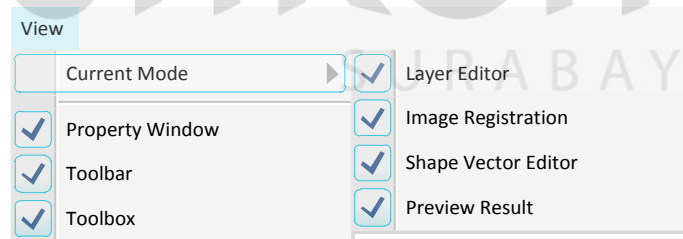
Menu *edit* memiliki fungsi untuk mengembalikan *panel view* yang sedang aktif menjadi ke posisi semula. Selain itu dapat berpindah dari *view* saat ini ke *view* sebelumnya. Menu *Edit* dapat dilihat pada Gambar 3.56.



Gambar 3.56 Desain Menu Edit

## 6. Desain Menu View

Menu *View* berhubungan dengan tampilan dari *workspace*. Dengan menu *view user* bisa menampilkan maupun menyembunyikan panel atau *window* yang tersedia pada *workspace*. Selain itu untuk berpindah dari *view* satu ke *view* yang lain dapat dilakukan melalui menu ini. Desain Menu *view* dapat dilihat pada Gambar 3.57.

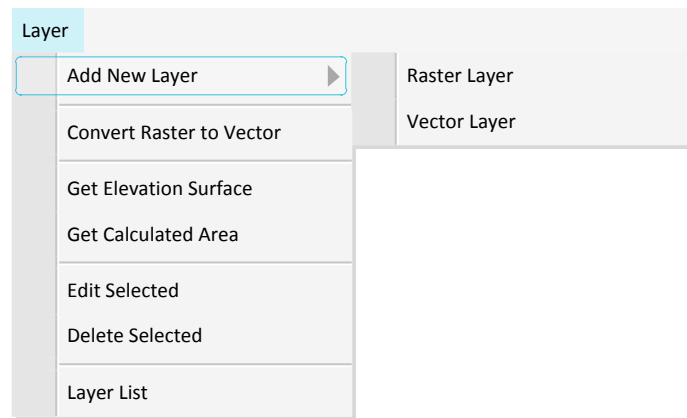


Gambar 3.57 Desain Menu View

## 7. Desain Menu Layer

Menu layer sangat erat kaitannya dengan *layer*. Pada menu ini *user* dapat menambah layer baru, menghapus layer, atau melakukan proses *editing* pada *layer*

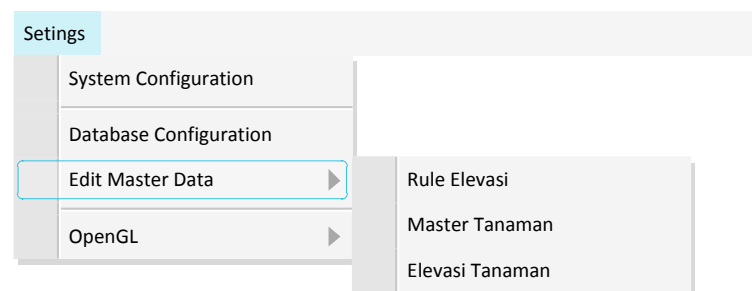
yang sedang dipilih. Selain itu *user* juga dapat membuka *form* layer list melalui *menu* ini. Desain *menu layer* dapat dilihat pada Gambar 3.58.



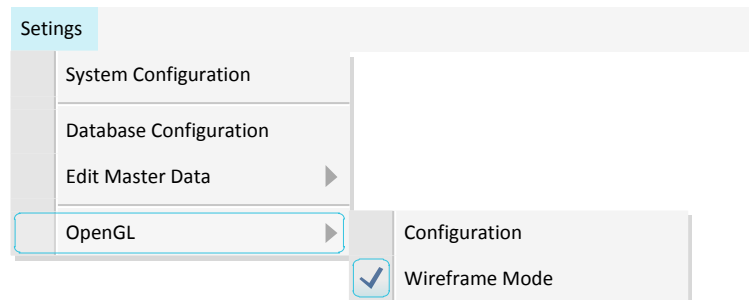
Gambar 3.58 Desain *Menu Layer*

## 8. Desain *Menu Settings*

Menu *settings* merupakan kumpulan menu yang dapat mengatur kinerja dari aplikasi. Terdapat *menu system configuration* dalam menu ini yang menampilkan *form setting*. Selain itu untuk melakukan perubahan pada *database* dapat dilakukan juga pada menu ini. Desain Menu Setting dapat dilihat pada Gambar 3.59. Dan desain submenu OpenGL pada menu setting dapat dilihat pada Gambar 3.60. di halaman 82.



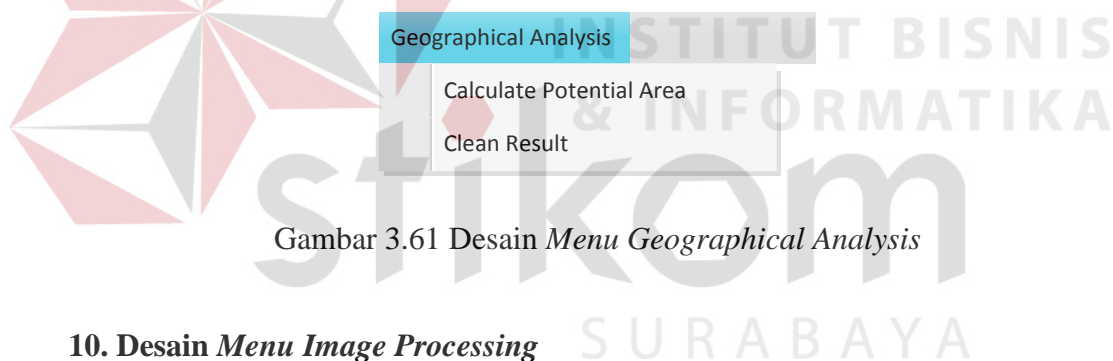
Gambar 3.59 Desain *Menu Settings*



Gambar 3.60 Desain Menu *Settings OpenGL*

### 9. Desain Menu *Geographical Analysis*

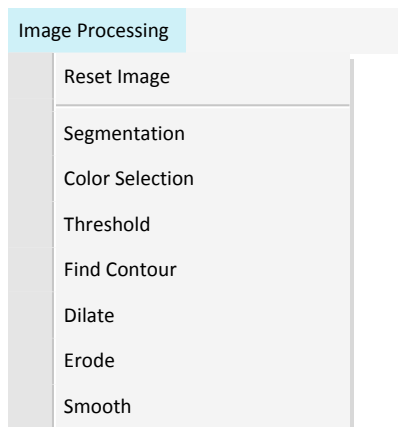
Menu ini berfungsi untuk memanggil *form calculate* potensial area yang akan menghasilkan daerah potensial dari data yang sudah di inputkan sebelumnya. Dengan menu ini *user* dapat menampilkan *form calculate potensial area*. Desain *Menu Geographical Analysis* dapat dilihat pada Gambar 3.61.



Gambar 3.61 Desain Menu *Geographical Analysis*

### 10. Desain Menu *Image Processing*

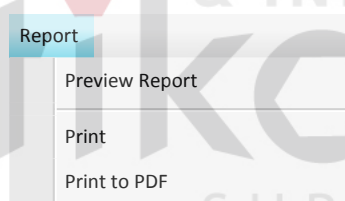
Ada beberapa fungsi *Image Processing* dalam aplikasi ini untuk menentukan daerah potensial. Menu *Image Processing* berisi pilihan jenis *Image Processing* yang digunakan untuk mengolah gambar peta sehingga menjadi sebuah vektor daerah potensial. Desain *Menu Image Processing* Sistem Informasi Geografis Penentuan Lahan Potensial Menggunakan *Image Processing* dapat dilihat pada Gambar 3.62. di halaman 83.



Gambar 3.62 Desain Menu *Image Processing*

### 11. Desain Menu *Report*

Menu *report* berguna untuk menampilkan laporan. *User* dapat melihat *preview* terlebih dahulu sebelum melakukan *print*. Aplikasi ini juga dapat melakukan proses *export report* dalam format PDF. Desain menu *Report* dapat dilihat pada Gambar 3.63.

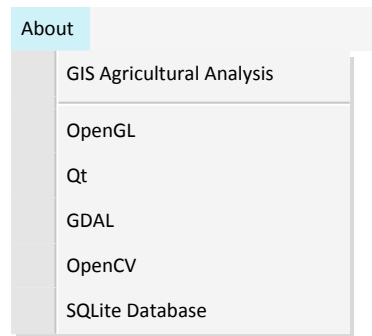


Gambar 3.63 Desain Menu *Report*

### 12. Desain Menu *About*

Pada menu ini *user* dapat melihat about dari aplikasi. *About window* akan menampilkan versi dari aplikasi yang sedang digunakan. Selain itu *about* juga menampilkan versi dari *library* yang digunakan dalam aplikasi ini. Desain Menu *About* pada aplikasi ini dapat dilihat pada Gambar 3.64. di halaman 84.





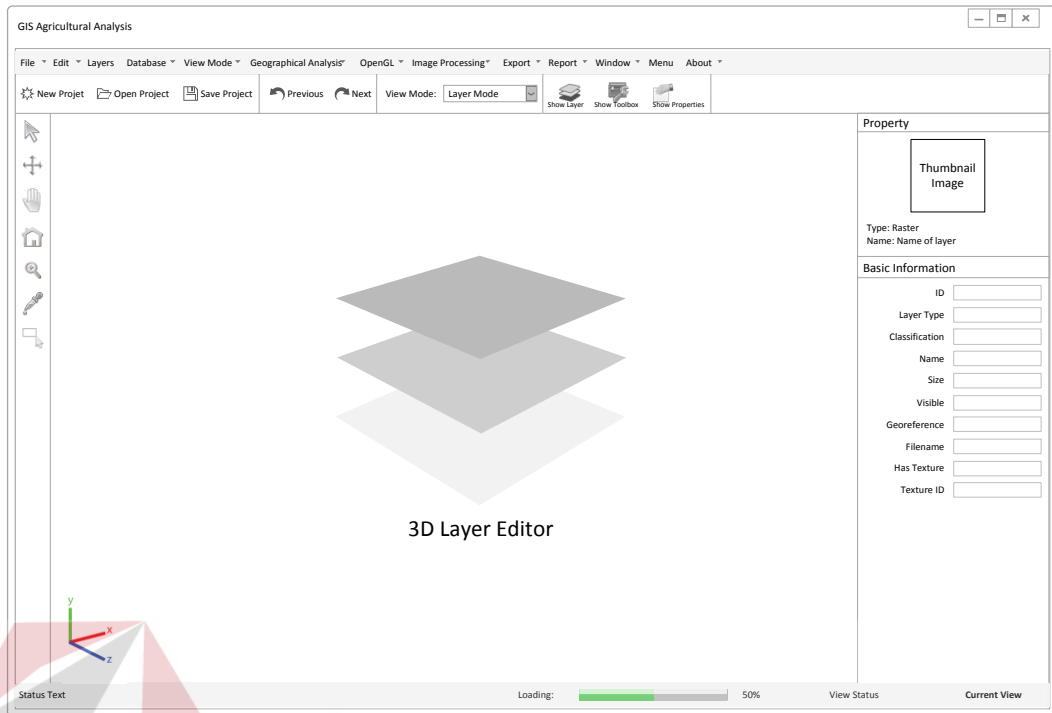
Gambar 3.64 Desain *Menu About*

### 13. Desain *Form Layer Edit*

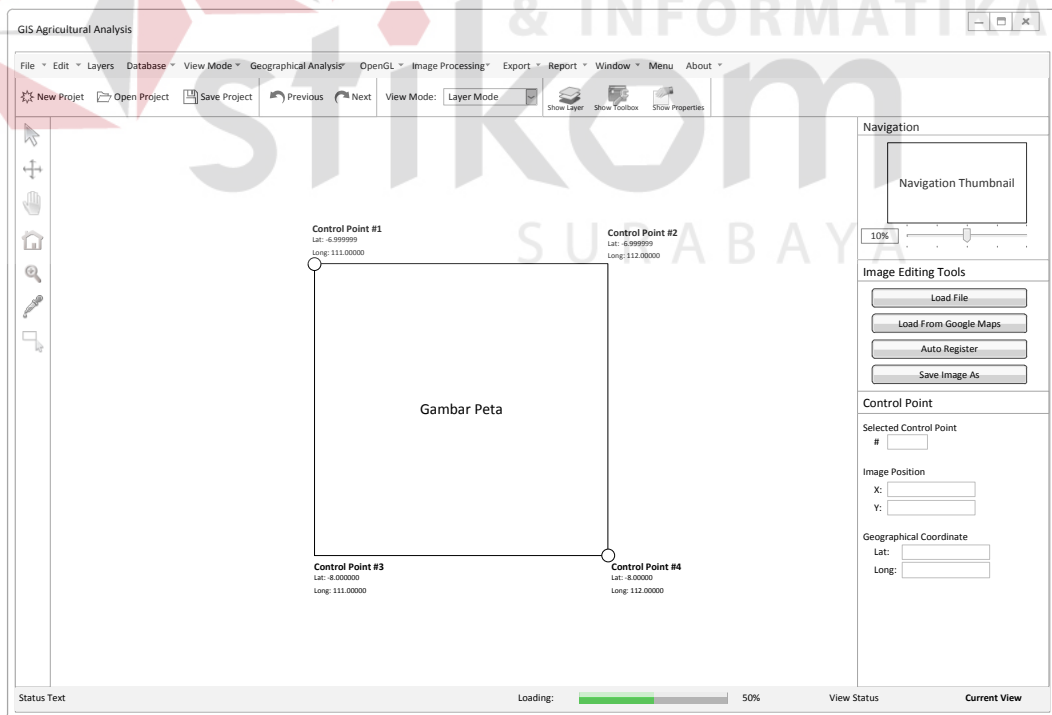
Salah satu dari 4 *view* pada aplikasi ini adalah *Layer Edit View*. Pada *view* ini *user* dapat melakukan proses *edit* tiap-tiap layer dengan data yang berbeda. Pada layar utama menampilkan tumpukan bidang dalam bentuk 3D yang merepresentasikan layer yang sedang digunakan pada aplikasi. Setiap perubahan pada layer maka layer 3D juga akan mengalami perubahan. Disini kanan atau pada *property panel* berisi informasi mengenai *layer* yang sedang dipilih. *User* dapat melihat dan merubah data pada *layer* melalui *property panel* ini. Desain *Form* dari *Layer Edit Mode* ini dapat dilihat pada Gambar 3.65 di halaman 85.

### 14. Desain *Form Image Register*

Pada *view image register* ini *user* dapat mengedit *layer* yang memiliki tipe data raster. *User* dapat merubah data *georeference* pada *layer* tersebut. Pada sisi kanan terdapat menu untuk melakukan *load file* dan *edit* koordinat. Selain untuk melakukan registrasi koordinat pada *layer*. Pada menu ini *user* juga dapat melakukan proses *download* gambar dari Google Maps dan langsung memberi informasi georafis pada *layer* tanpa perlu diinputkan oleh *user*. Desain *Form* dapat dilihat pada Gambar 3.66 di halaman 85.



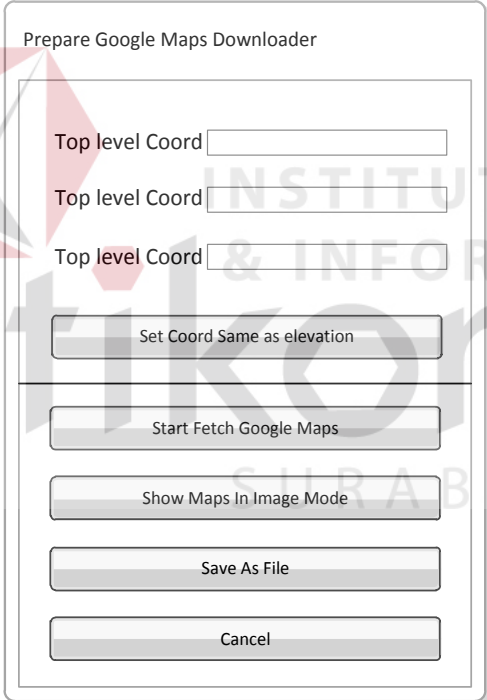
Gambar 3.65 Desain View Layer Editor



Gambar 3.66 Desain View Image Register

### 15. Desain Form Prepare GoogleMaps

*Form prepare google maps* digunakan untuk mempersiapkan *view image* register mendownload gambar dari Google. Beberapa parameter yang perlu disiapkan berupa titik koordinat dan *zoom level*. Apabila *user* ingin mendapatkan peta permukaan dengan koordinat yang sama dengan koordinat peta elevasi, pada *menu* disediakan pilihan untuk melakukannya. Hasil dari peta dapat disimpan ke sebagai gambar atau dapat langsung digunakan pada *layer*. Berikut adalah desain dari *form* tersebut. Desain *Form Prepare GoogleMaps* dapat dilihat pada Gambar 3.67.



Prepare Google Maps Downloader

Top level Coord

Top level Coord

Top level Coord

Set Coord Same as elevation

Start Fetch Google Maps

Show Maps In Image Mode

Save As File

Cancel

Gambar 3.67 Desain Form Prepare Google Maps

### 16. Desain Form Master Data Tanaman

*Form* master data tanaman adalah *form* yang menangani master data tanaman. Master data tanaman merupakan jenis-jenis tanaman yang dapat tumbuh

di daerah yang akan di analisa nanti. Desain *Form Tanaman Rule* aplikasi ini dapat dilihat pada Gambar 3.68.

Master Data Tanaman Rule

ID Tanaman

Nama

Desc

Column 1	Column 2	Column 2	Column 2
Text	Text	Text	Text
Text	Text	Text	Text
Text	Text	Text	Text

Gambar 3.68 Desain *Form Master Data Tanaman*

### 17. Desain *Form Prepare Analysis*

Sebelum melakukan analisa daerah potensial *form* prepare analysis akan tampil terlebih dahulu. *Form* ini dipersiapkan apabila *user* hendak melakukan proses analisa. Proses analisa memerlukan beberapa prasyarat untuk melakukan analisa daerah potensial, salah satunya adalah harus adanya data elevasi pada layer pertama. Pada *form* ini *user* akan diinformasikan prasyarat yang sudah terpenuhi dan yang belum terpenuhi. Apabila ada prasyarat yang belum terpenuhi *form* akan menampilkan *icon* berupa gambar silang dan memberi keterangan prasyarat yang belum terpenuhi dan apa yang harus dilakukan oleh *user*. Sedangkan apabila semua syarat sudah terpenuhi *form* akan menampilkan *icon*

*check* dan tombol proses diaktifkan. Ketika tombol ini di klik sistem akan secara otomatis melakukan prosedur menganalisa daerah potensial. Desain *form* dapat dilihat pada Gambar 3.69.

Gambar 3.69 Desain *Form Prepare Analysis*

### 18. Desain *Form Analysis*

*Form analysis* adalah *form* yang menampilkan progress dari analisa penentuan lahan potensial menggunakan *Image Processing*. Ada *form* ini *user* hanya diberikan informasi mengenai proses apa yang sedang berlangsung dan apabila terjadi error akan ditampilkan. Selama proses berlangsung *user* tidak bisa melakukan apa-apa hanya dapat menunggu hingga proses selesai. Ketika proses selesai tombol *close* akan aktif. Tombol tersebut digunakan untuk menutup *form* tersebut. Desain *Form Analysis* dapat dilihat pada Gambar 3.70 di halaman 89.

Gambar 3.70 Desain *Form Analysis*

### 19. Desain *Form System Configuration*

Aplikasi ini menerapkan beberapa *library* diantara OpenGL untuk visualisasi, GDAL untuk menangani *file* GIS, dan OpenCV untuk melakukan *Image Processing*. Karena banyaknya *library* yang digunakan perlu sebuah konfigurasi untuk menanganinya agar sesuai dengan kebutuhan aplikasi. *Form System Configuration* adalah *form* yang mengatur setting dari *system*. Semua *setting* disimpan kedalam *database*. Yang nantinya data pada *setting* tersebut akan digunakan pada waktu beberapa *library* tersebut digunakan. Desain Input *Form System Configuration* dapat dilihat pada Gambar 3.71. di halaman 90.

Dialog Title

System OpenGL Database

Show Splashscreen at startup  Show

Layer Distance

Pan Position

MRU List

Column 1

Text

Text

Text

Text

Enable Internet Proxy  Show

Internet Proxy Address

Internet Proxy Port

Internet Proxy Username

Internet Proxy Password

Gambar 3.71 Desain *Form System Configuration*

## 20. Desain *Report Analysis Result*

Desain laporan dari Sistem Informasi Geografis ini memiliki beberapa halaman dimana setiap kali proses *printing*, setiap halaman akan tercetak secara bersama-sama. Halaman pertama berisi informasi mengenai daerah yang sedang dilakukan analisa, pada *report* ditampilkan dalam gambar dan pada sisi kiri daerah ditampilkan relatif dari lokasi di Indonesia. Halaman kedua berisi detail daerah analisa. Ditampilkan beberapa jenis peta dengan lokasi yang sama. Jenis peta tersebut antara lain peta permukaan, peta ketinggian, peta suhu, peta kelembaban dan peta curah hujan. Halaman ketiga dan seterusnya adalah halaman

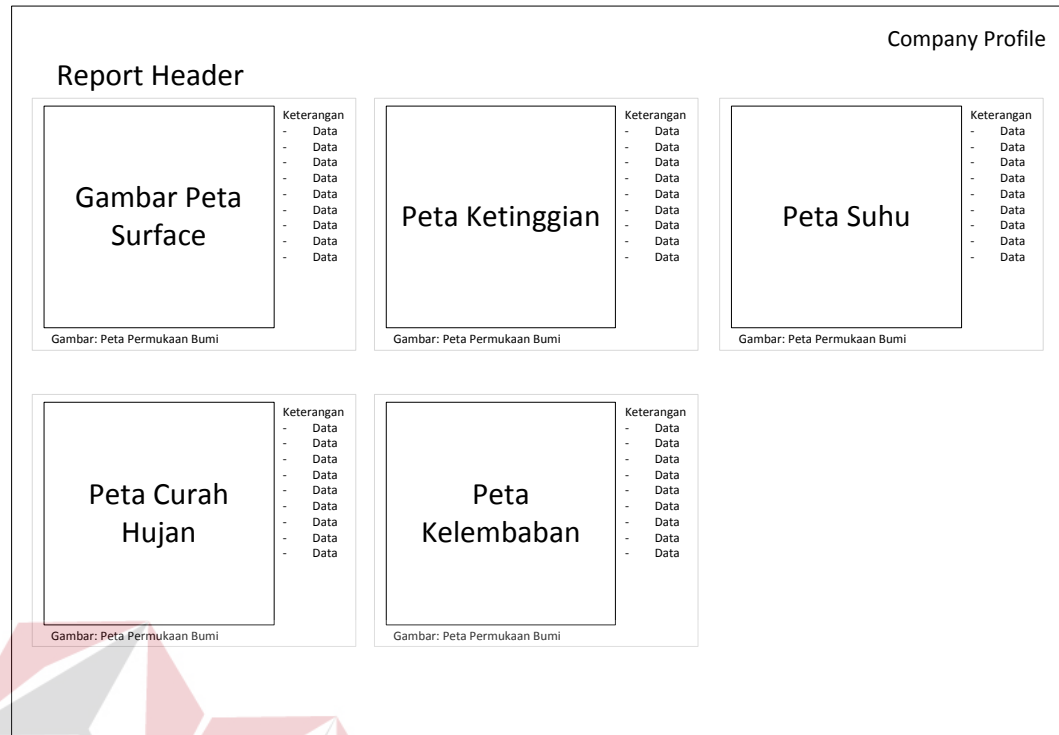
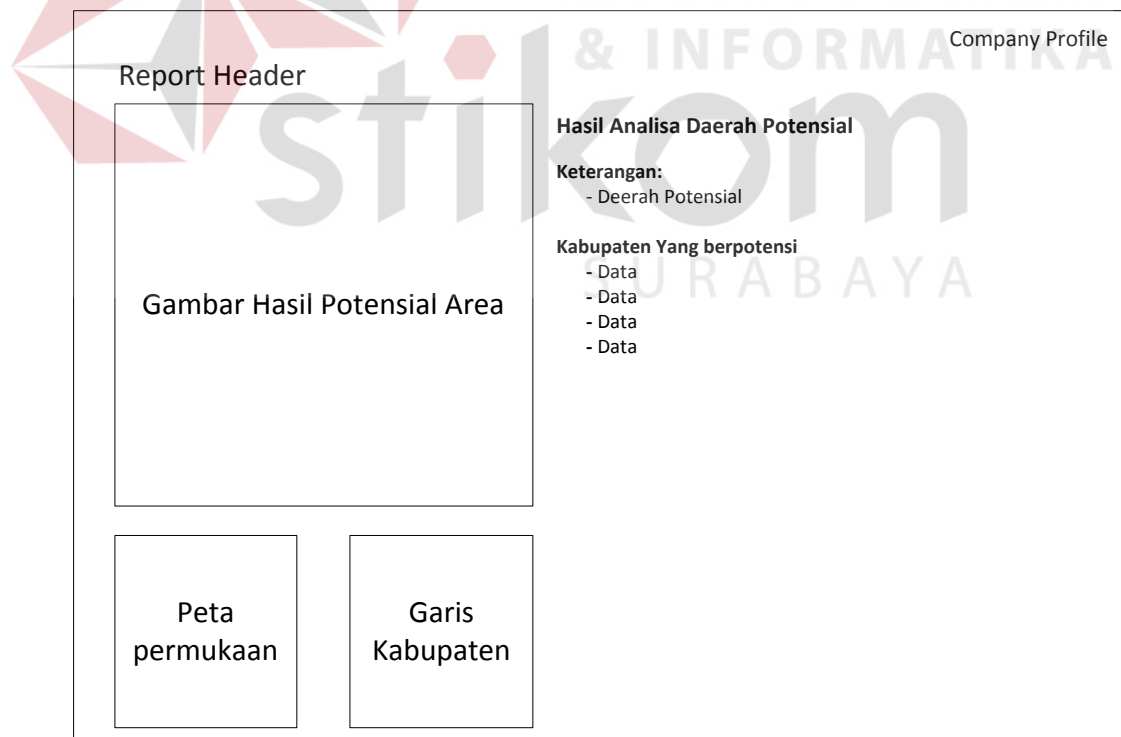
yang berisi hasil analisa daerah potensial berisi tanda berwarna hijau yang merupakan tanda bahwa daerah tersebut adalah daerah potensial. Serta terdapat keterangan yang menjelaskan gambar yang terdapat dalam peta. Desain halaman pertama dapat dilihat pada Gambar 3.72.



Gambar 3.72 Desain *Report* Halaman Pertama

Sedangkan desain *output* dari laporan halaman kedua dan ketiga dari *report* dapat dilihat pada Gambar 3.73 dan Gambar 3.74. di halaman 92.

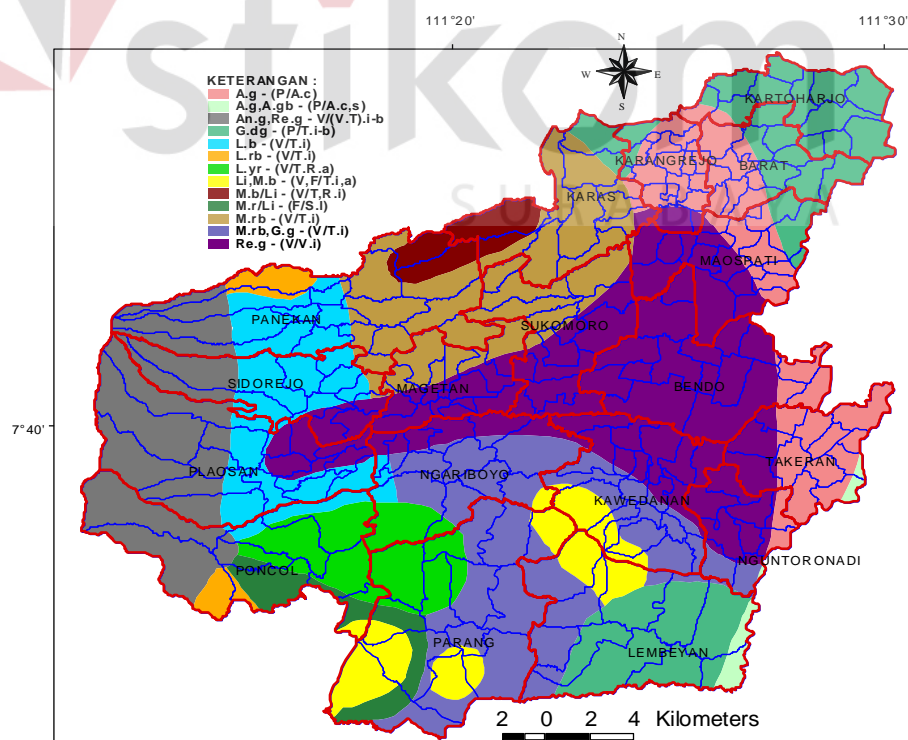


Gambar 3.73 Desain *Report* Halaman KeduaGambar 3.74 Desain *Report* Halaman Ketiga

## 21. Desain Fitur *Experimental*

Sistem Informasi Geografis ini masih dapat dikembangkan lebih jauh lagi dan lebih luas. Beberapa fitur tambahan diuji coba ke dalam sistem untuk menampilkan bahwa fitur ini masih dapat dikembangkan. Untuk dapat mengaktifkan fitur ini, *user* harus mengaktifkan *menu enable experimental feature* pada *form setting*

Kriteria yang sementara digunakan berjumlah tiga buah yaitu suhu, kelembaban dan curah hujan. Namun dalam fitur ini kriteria ditambahkan satu kriteria lagi yaitu jenis tanah dan tidak semua daerah mampu menghasilkan *output* untuk kriteria tambahan ini, hanya satu kabupaten yang dapat memanfaatkan fitur ini yaitu kabupaten magetan. Selain itu pada *report* sistem dapat menampilkan informasi hingga batas kecamatan. Gambar jenis tanah untuk Kabupaten Magetan dapat dilihat pada Gambar 3.75.



Gambar 3.75 Peta Jenis Tanah di Kabupaten Magetan

### 1.2.5 Desain Uji Coba

Desain uji coba bertujuan untuk memastikan bahwa aplikasi telah dibuat sesuai dengan kebutuhan atau tujuan yang diharapkan. Kekurangan atau kelemahan aplikasi pada tahap ini akan dievaluasi sebelum diimplementasikan secara nyata. Uji coba dalam Sistem Informasi Geografis ini terdiri dari tiga tahap pengujian, yaitu: evaluasi *output*, evaluasi visualisasi 3D, dan evaluasi tingkat keakuratan penentuan lahan. Berikut adalah desain masing-masing uji coba.

#### A. Desain Evaluasi Ouput

Uji coba ini dilakukan dengan memberikan suatu input yang sudah disiapkan sebelumnya dan hasil dari sistem sudah ditentukan. Apabila sistem mengeluarkan hasil yang sesuai dengan apa yang diharapkan, maka dapat dikatakan sistem ini lulus untuk uji coba tahap ini.

*Input* yang digunakan adalah sebuah *project* file dari sector 2 yang telah dibuat sebelumnya. Sistem akan mencoba membuka *file project* tersebut dan melakukan proses analisis dengan mencari tanaman padi sawah. *Output* yang seharusnya muncul adalah sistem mampu menampilkan daerah potensial dari bawang dengan beberapa kriteria yang telah ditentukan untuk tanaman padi sawah pada *sector 2*.

#### B. Desain Evaluasi Visualisasi 3D

Uji coba ini dilakukan dengan cara membuka sebuah *file* elevasi berupa *file* DTED. Apabila file selesai dibuka sistem akan melakukan proses *loading*, dan ketika selesai sistem akan menampilkan peta elevasi tersebut dalam bentuk 3D. Evaluasi ini dilakukan untuk menguji apakah sistem dapat menampilkan visualisasi 3D dengan baik. Evaluasi dilakukan dengan cara membuka *project*

yang telah disimpan terlebih dahulu dimana terdapat data peta ketinggian di dalam *project* tersebut. Apabila berhasil maka sistem akan membuka *project* tersebut dan menampilkan peta ketinggian dalam bentuk 3D.

Uji Evaluasi dilakukan dengan cara membuka sebuah *project* yang telah dibuat sebelumnya. *Project* tersebut telah berisi data ketinggian dan peta permukaan bumi. Apabila *project* tersebut dibuka maka seharusnya program dapat menampilkan permukaan bumi dalam bentuk 3D

### **C. Desain Evaluasi Tingkat Keakuratan Penentuan Lahan**

Uji coba ini menguji hasil output dari sistem dengan data yang ada di lapangan. Dalam uji coba ini tidak semua daerah di Jawa Timur hanya beberapa kabupaten yang akan di uji. Uji coba dilakukan dengan membandingkan hasil output dari program dengan data yang diperoleh dari Dinas Pertanian dan BPS.

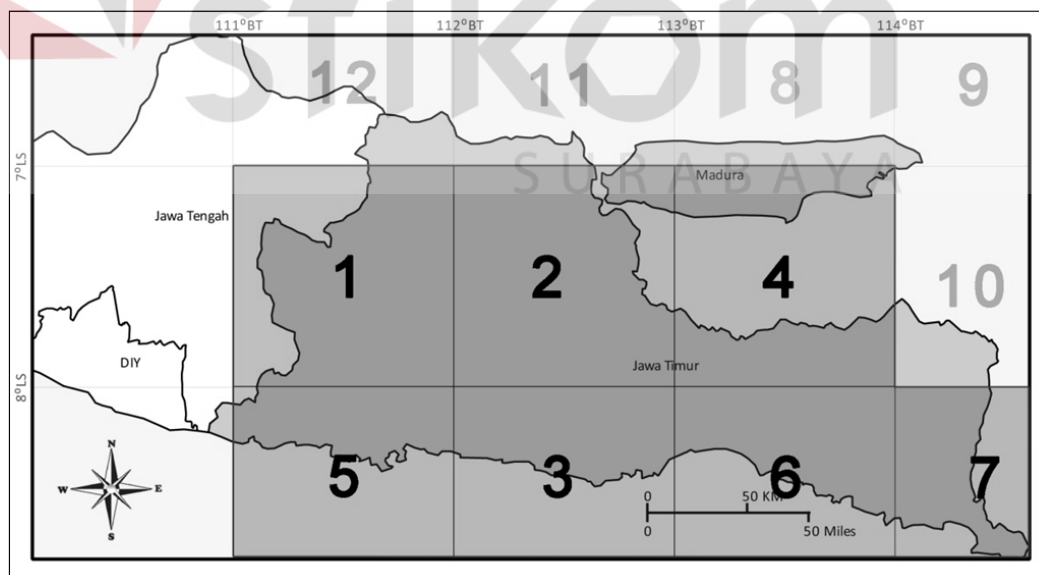
Data dari Dinas Pertanian dan BPS yang digunakan dalam uji coba ini selengkapnya dapat dilihat pada Lampiran 12 – 21 di halaman 162 – 171. Dalam Uji coba tidak seluruh kabupaten yang berada di Jawa Timur dilakukan pengujian, hanya beberapa kabupaten yang berjumlah 28 (dua puluh delapan) karena ketersediaan data dari Dinas Pertanian Propinsi Jawa Timur hanya terbatas sebanyak 28 kabupaten.

Sedangkan untuk sampel tanaman yang diuji, Jenis tanaman yang digunakan adalah komoditi yang unggul di Jawa Timur tetapi tidak seluruhnya, Uji coba dilakukan dengan mengambil tiap kateogoru tanaman untuk dilakukan pengujian. Adapun beberapa sample tanaman yang digunakan dalam uji coba ini berjumlah 10 buah dapat dilihat pada Tabel 3.10 di halaman 96.

Tabel 3.10 Sample Tanaman Uji Coba

No.	Nama Tanaman	Kategori
1	Apel	Buah-buahan
2	Mangga	Buah-buahan
3	Mawar	Bunga
4	Kacang Tanah	Kacang-kacangan
5	Kedelai	Kacang-kacangan
6	Bawang Merah	Sayuran
7	Tomat	Sayuran
8	Jagung	Serelia
9	Padi Sawah	Serelia
10	Jahe	Tanaman Obat

Daerah yang di uji coba adalah sebagian besar Jawa Timur yang merupakan kumpulan dari *file-file* dted yang telah di download sebelumnya. Gambar daerah yang akan di lakukan pengujian dapat dilihat pada Gambar 3.76. Daerah dari nomor 1 sampai dengan nomor 7 adalah daerah yang akan dilakukan pengujian.



Gambar 3.76 Daerah Pengujian

Proses pembandingan dalam uji coba ini adalah, data pada Dinas Pertanian apabila suatu kabupaten menghasilkan suatu tanaman maka akan diberi poin 1,

sedangkan apabila daerah tersebut tidak menghasilkan akan diberi poin 0. Untuk output sistem, apabila sistem mengeluarkan nama kabupaten dalam *report* maka akan diberi poin 1 selain itu akan diberi poin 0. Sedangkan untuk hasil adalah perbandingan antara hasil pada Dinas Pertanian dengan hasil sistem. Apabila pada dinas memiliki point 1 dan hasil sistem memiliki point 1 maka hasil akan bernilai 1. Apabila dinas memiliki point 0 dan hasil sistem memiliki point 0 maka hasil akan bernilai 1 juga. Selain itu akan bernilai 0.

Berikut adalah desain tabel uji coba tiap tanaman. Dalam tabel uji coba terdapat 2 (dua) jenis kriteria tanaman yang di uji cobakan yaitu Kriteria Tanaman S1 dan Kriteria Tanaman S3. Definisi kriteria tanaman S1 dan S3 dapat dilihat kembali pada BAB II Landasan Teori pada sub 2.42 Karakteristik Lahan di halaman 14 – 16.

Tabel 3.11 Desain Uji Coba Tiap Jenis Tanaman

Data Dinas Pertanian	Poin	Kriteria Tanaman S1			Kriteria Tanaman S3		
		Hasil Sistem	Poin	Hasil	Hasil Sistem	Poin	Hasil
Kabupaten-1							
Kabupaten-2							
Kabupaten-3							
.							
.							
.							
.							
.							
Kabupaten- <i>n</i>							
<b>TOTAL</b>							
<b>Prosentase</b>							