

**APLIKASI SISTEM INFORMASI GEOGRAFIS UNTUK PENYAJIAN  
INFORMASI PEMASARAN PERUMAHAN SEKUNDER  
DI KOTA MALANG**

**( Studi kasus : ERA Puri Broker Properti Wilayah Malang )**

**TUGAS AKHIR**



**Oleh :**

**MOH. ABDULLAH**

**9 5 2 5 0 5 9**

**JURUSAN TEKNIK GEODESI  
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN  
INSTITUT TEKNOLOGI NASIONAL MALANG  
2005**

MAKING THE BEST USE OF THE AVAILABLE INFORMATION IN THE  
FIELD OF THE INVESTIGATION OF THE SUBJECT'S ACTS

AND THE RESULTS THEREOF

(This is a preliminary report and should not be used as a final report)

THE RESULTS

1945

THE RESULTS

OF THE

INVESTIGATION OF THE SUBJECT'S ACTS  
AND THE RESULTS THEREOF

1945

**APLIKASI SISTEM INFORMASI GEOGRAFIS UNTUK PENYAJIAN  
INFORMASI PEMASARAN PERUMAHAN SEKUNDER  
DI KOTA MALANG**

( Studi kasus : ERA Puri Broker Properti Wilayah Malang )

**TUGAS AKHIR**

Diajukan untuk memenuhi tugas-tugas dan  
Syarat-syarat memperoleh Ijasah Sarjana S1  
Jurusan Teknik Geodesi Institut Teknologi Nasional  
Malang

**Disusun Oleh :**

**MOH. ABDULLAH**

**9 5 2 5 0 5 9**

**Dosen Pembimbing :**

- 1. Ir. Leo Pantimena, Msc.**
- 2. Ir. Agus Darpono, MT.**

**JURUSAN TEKNIK GEODESI  
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN  
INSTITUT TEKNOLOGI NASIONAL MALANG  
2005**

**APLIKASI SISTEM INFORMASI GEOGRAFIS UNTUK PENYAJIAN INFORMASI  
PEMASARAN PERUMAHAN SEKUNDER DI KOTA MALANG**

( Studi kasus : ERA Puri Broker Properti Wilayah Malang )

Oleh : Moh. Abdullah

Hak Cipta © 2005, pada penulis

Hak cipta dilindungi Undang-Undang. Dilarang memperbanyak atau memindahkan sebagian atau seluruh isi buku ini dalam bentuk apapun, baik secara elektronik maupun mekanis, termasuk memfotocopy, merekam atau dengan sistem penyimpanan lainnya, tanpa izin tertulis dari Penulis

Penerbit dan Percetakan

Joeragan

Jl. Permata Hijau B.34

Telp. [0341] 572250 Malang 65144



MILIK  
PERPUSTAKAAN  
ITN MALANG

**APLIKASI SISTEM INFORMASI GEOGRAFIS UNTUK PENYAJIAN  
INFORMASI PEMASARAN PERUMAHAN SEKUNDER  
DI KOTA MALANG**

( Studi kasus : ERA Puri Broker Properti Wilayah Malang )

**TUGAS AKHIR**

Diajukan untuk memenuhi tugas-tugas dan  
Syarat-syarat memperoleh Ijasah Sarjana S1  
Jurusan Teknik Geodesi Institut Teknologi Nasional  
Malang

Oleh :

**MOH. ABDULLAH**  
**9 5 2 5 0 5 9**

Menyetujui,

Dosen Pembimbing I

Dosen Pembimbing II



[ Ir. Leo Pantimena, MSc. ]



[ Ir. Agus Darpono, MT. ]

Mengetahui,

Plh. Ketua Jurusan Teknik Geodesi



[ Ir. Leo Pantimena, MSc. ]

Dipertahankan didepan panitia penguji tugas akhir Jurusan Teknik Geodesi, Fakultas Teknik Sipil Dan Perencanaan, Institut Teknologi Nasional Malang dan diterima untuk memenuhi sebagian dari syarat-syarat untuk memperoleh gelar Sarjana S1 Teknik Geodesi.

Pada hari/tanggal : Jumat, 08 Oktober 2005

### **Panitia Ujian Tugas Akhir**

Ketua

Sekretaris



Ir. Agustina Nurul Hidayati, MT.  
Dekan FTSP



Ir. Leo Pantimena, MSc.  
Plh. Ka.Jur Teknik Geodesi

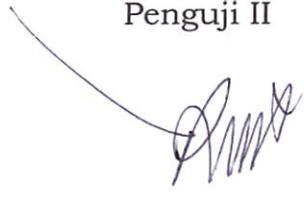
### **Anggota Penguji,**

Penguji I

Penguji II



[ Ir. Leo Pantimena, MSc. ]

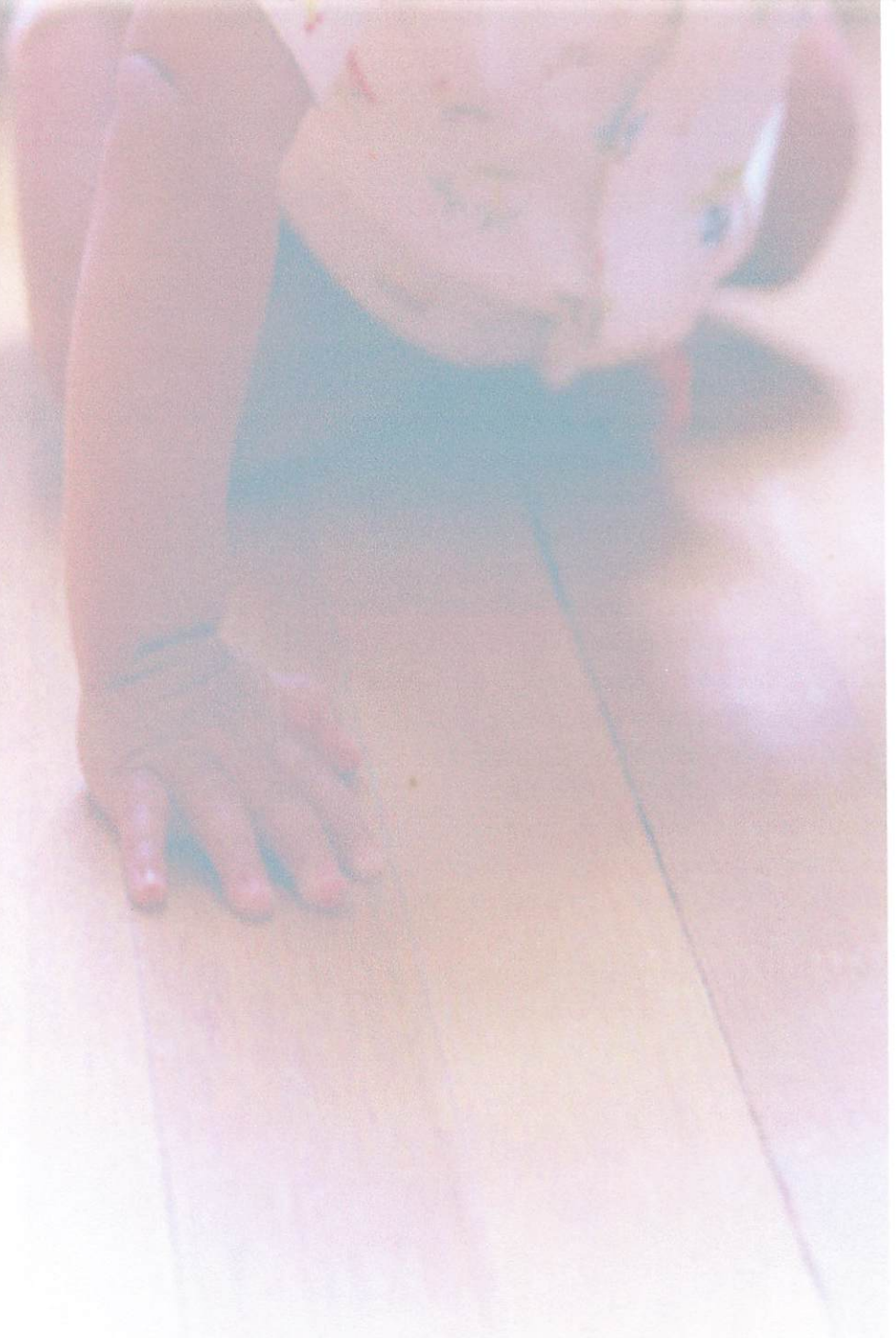


[ Ir. Rinto Sasongko, MT. ]

Penguji III



[ Ir. M. Ruslin Anwar, MSi. ]



*To everyone...  
who give me everything...*

*Special for 'emak, kulo nyusun agunge pangapunten...  
Mbak Ning n mas Edi, mbak Is n mas Is matur suwan sing akeh,  
sepurane ngrepoti terus adikmu iki...*

*iDea 'my wife' I Love forever...*

*Thank  
you*

*I Love  
you*



## KATA PENGANTAR



Alhamdulillahirobilalamin...Puji syukur kehadlirat ALLAH SWT. yang telah memberikan rahmad dan hidayahNYA sehingga terselesaikannya tugas akhir “Aplikasi Sistem Informasi Geografi untuk Penyajian Informasi Pemasaran Perumahan Sekunder di Kota Malang” (Studi kasus : ERA Puri Broker Properti Wilayah Malang). Walau harus dikerjakan dengan rentang waktu yang cukup lama dan dengan segala pengorbanan.

Dengan kemajuan teknologi khususnya dibidang komputer, terdapat banyak hal yang dapat dimanfaatkan untuk kemajuan properti di Indonesia (khususnya di Malang) yang sangat pesat perkembangannya. Salah satu teknologi yang memungkinkan untuk mengumpulkan, menganalisa dan menyajikan data-data lokasi secara visual dan akurat adalah *Geographic Information System* (Sistem Informasi Geografis atau SIG). Dengan kemampuan *Microsoft Visual Basic* yang dipadukan dengan SIG merupakan alat bantu yang tepat dalam studi pasar properti, karena pada dasarnya studi pasar adalah perpaduan antara analisa data bangunan, karakter ekonomi-demografi dan tren pasar yang selalu berkaitan dengan orientasi geografis.

Dalam teknologi SIG semua dapat digambar atau divisualkan atau lebih tepatnya dipetakan, karena teknologi SIG memang merupakan *tools* atau alat untuk mengelola dan menganalisa data dan informasi secara spatial. Dengan mengaplikasikan fungsionalitas SIG yang ada maka analisis lokasi yang diharapkan dapat diwujudkan.

Tugas akhir ini dimaksudkan untuk membangun kemampuan mahasiswa secara runtut hasil dari suatu pengkajian yang berkaitan dengan satu bidang atau lintas bidang keahlian dalam lingkup bidang Geodesi dan Geoinformatika. Sedangkan laporan ini merupakan bukti telah menyelesaikan tugas akhir yang harus ditempuh untuk menyelesaikan studi dan mendapat gelar sarjana S1 Teknik Geodesi. Muatan-muatan yang ada

didalamnya tidaklah jauh dari pelaksanaan penelitian dan berdasarkan teori-teori yang ada. Besar harapan agar karya-karya semacam ini bukan hanya sebagai tuntutan tetapi sebagai wujud tanggung jawab setiap mahasiswa dalam mengembangkan wawasan keilmuannya yang akhirnya bermanfaat bagi yang lainnya dimasa datang.

Ucapan Terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu penulis belajar, berkarya dan meyelesaikan laporan tugas akhir ini. Diantaranya kepada yang terhormat :

1. Dr. Ir. Abraham Lomi, MSEE. selaku Rektor ITN Malang
2. Ir. Agustina Nurul Hidayati, MT. Dekan FTSP ITN Malang
3. Ir. Joanes Pradono De Deo, MSc. Bapak Geodesi ITN Malang
4. Ir. Leo Pantimena, MSc. Plh Ketua Jurusan Teknik Geodesi dan dosen Pembimbing I
5. Ir. Agus Darpono, MT. Dosen Pembimbing II
6. Ir. Rinto Sasongko, MT. dan Ir. M. Ruslin Anwar, MSi. Selaku dosen penguji dan yang memberikan perbaikan pada hasil akhir tugas ini.
7. Mr. Hendrono, Direktur ERA Puri Properti Malang
8. Mr. and Brother 'dosen' Mas Herry Purwanto, ST. MSc.
9. Seluruh staf pengajar, administrasi dan karyawan khususnya dilingkungan Teknik Geodesi dan ITN Malang secara keseluruhan.
10. The Gank 'YPS' (abLeh, Pithok, Nippon, Chimot, Ndok'teloor', n Vanda) aduh sepurane rek.. aku telah menjadi *bau tak sedap* dalam gank kita [lulus mburi dewe..]
11. Temen-temen sekaligus saudara di PH B34 spesial Jo'chandra'pan suwuuuuun banget yo...
12. Temen-temen Geodesi '95 dan seluruh angkatan terima kasih.

Semoga ilmu dan amalan yang telah diberikan kepada penulis dapat diterima oleh ALLAH SWT. dan berguna bagi kehidupan dimasa mendatang. Amiin....

Permata Hijau, Oktober 2005

Moh. Abdullah

## **DAFTAR ISI**

Lembar Persetujuan .....	i
Lembar Pengesahan.....	ii
Lembar Persembahan .....	iii
Kata Pengantar.....	iv
Daftar Isi .....	vi
Daftar Gambar .....	viii
Daftar Tabel .....	ix

### **BAB I PENDAHULUAN**

1.1. Latar Belakang.....	1
1.2. Maksud dan Tujuan Penelitian.....	2
1.3. Manfaat Penelitian .....	2
1.4. Batasan Penelitian .....	3
1.5. Tinjauan Pustaka.....	3
1.6. Dasar Teori .....	5
1.6.1. SIG sebagai Basis Data .....	5
1.6.2. Pengertian Sistem Informasi Geografis .....	6
1.6.3. Komponen Sistem Informasi Geografis .....	6
1.6.3.1. Data Input (Data Masukan).....	7
1.6.3.2. Penyimpanan dan Pemanggilan Data.....	10
1.6.3.3. Data Manipulasi dan Analisa.....	11
1.6.3.4. Data Output atau Visualisasi Hasil .....	11
1.6.4. Pengertian Basis Data.....	11
1.6.4.1. Sistem Manajemen Basis Data (DBMS).....	13
1.6.4.2. Merancang Basis Data SIG.....	14
1.6.4.3. Struktur Basis Data .....	16
1.6.4.5. Model Data dari Sistem Basis Data.....	20
1.6.4.6. Konsep Hubungan Antar Entity (E - R).....	20
1.6.5. Transaksi Properti.....	23



1.6.5.1. Macam Cara Transaksi Properti.....	24
1.6.5.2. Rumah Sekunder.....	27
1.6.6. Perangkat Lunak Visual Basic 6.0.....	28
1.6.6.1. IDE Visual Basic.....	29
1.6.6.2. Obyek Linking and Embedding (OLE) .....	31
1.6.6.3 Perangkat Lunak Map Object 2.1.....	32

## **BAB II PELAKSANAAN PENELITIAN**

2.1. Deskripsi Lokasi Penelitian.....	34
2.2. Bahan dan Alat Penelitian .....	35
2.2.1. Bahan penelitian .....	35
2.2.2. Peralatan Penelitian .....	35
2.3. Pelaksanaan Penelitian .....	37
2.3.1. Persiapan Pelaksanaan Penelitian .....	43
2.3.2. Pemasukan Data Spasial .....	43
2.3.3. Editing Di AutoCAD.....	49
2.3.4. Export Data (Format .DXF).....	53
2.3.5. Import Data.....	54
2.3.6. Editing dengan ArcInfo .....	55
2.3.7. Pembuatan Topologi .....	57
2.3.8. Pengkodean Data Spasial .....	59
2.3.9. Pembuatan Basis Data .....	61
2.3.9.1. Pembuatan Tabel .....	63
2.3.9.2. Pembuatan Relationship Tabel .....	65
2.3.9.3. Pembuatan Query .....	68
2.3.10. Export Basis Data .....	71
2.3.11. Joint Item atau Penggabungan Data .....	72
2.3.12. Convert File .....	73
2.3.13. Perancangan Sistem dengan Visual Basic 6.0.....	74
2.3.13.1. Pembuatan Atribut-Atribut Form .....	74
2.3.13.2. Pembuatan Module pada Visual Basic .....	78

2.3.13.3. Distribusi Sistem .....	78
2.3.13.4. Implementasi dan Evaluasi Sistem .....	80

### **BAB III HASIL DAN PEMBAHASAN**

3.1. Penyajian Peta Informasi Obyek Pemasaran Properti .....	83
3.2. Penyajian Data Atribut Obyek Properti .....	85
3.3. Form Pencarian .....	87
3.4. Form Updating .....	90
3.5. Pembahasan Hasil .....	91
3.5.1. Pembahasan Data Spasial.....	91
3.5.2. Pembahasan Data Non Spasial.....	92
3.5.3. Pembahasan Pembuatan Program.....	93
3.5.4. Hal-hal yang Dapat Ditampilkan pada Sistem Informasi.....	97
3.5.5. Kemudahan dan Kekurangan Program .....	100

### **BAB IV KESIMPULAN DAN SARAN**

4.1. Kesimpulan .....	101
4.2. Saran .....	103

#### **Daftar Pustaka**

#### **Lampiran – Lampiran**

## DAFTAR GAMBAR

Gambar 2.1.	Struktur Komponen SIG.....	7
Gambar 2.2.	Struktur model data raster dan data vektor .....	10
Gambar 2.3.	Diagram Tahap Konseptual .....	15
Gambar 2.4.	Diagram Tahap Internal .....	15
Gambar 2.5.	Struktur Database Hirarki .....	17
Gambar 2.6.	Struktur Database Network.....	18
Gambar 2.7.	Struktur Database Relational.....	19
Gambar 2.8.	Tingkat Relasi Satu ke Satu .....	21
Gambar 2.9.	Diagram ER dengan Relasi Satu ke Satu .....	21
Gambar 2.10.	Tingkat Relasi Satu ke Banyak.....	22
Gambar 2.11.	Diagram ER dengan Relasi Satu ke Banyak.....	22
Gambar 2.12.	Tingkat Relasi Banyak ke Banyak .....	23
Gambar 2.13.	Diagram ER dengan Relasi Banyak ke Banyak .....	23
Gambar 2.14.	Ilustrasi Cara dalam Transaksi Properti .....	27
Gambar 2.15.	IDE Visual Basic dengan Jendela-Jendela yang Terbuka ..	30
Gambar 3.1.	Lokasi Penelitian .....	34
Gambar 3.2.	Diagram Alur Pelaksanaan Penelitian .....	40
Gambar 3.3.	Diagram Alur Pembuatan Program .....	42
Gambar 3.4.	Tampilan Software Auto Cad R14 .....	44
Gambar 3.5.	Tampilan Konfigurasi Menu Digital .....	45
Gambar 3.6.	Tampilan Konfigurasi Pemilihan Jenis Digital.....	45
Gambar 3.7.	Tampilan Konfigurasi Pemilihan Jumlah Tombol Mouse...	46
Gambar 3.8.	Tampilan Proses Kalibrasi pada Digitizer .....	48
Gambar 3.9.	Tampilan Cara Membuat Layer .....	49
Gambar 3.10.	Tampilan Penggunaan Perintah Trim.....	50
Gambar 3.11.	Tampilan Penggunaan Perintah Extend .....	51
Gambar 3.12.	Tampilan Penggunaan Perintah Pedit (Joint) .....	51
Gambar 3.13.	Tampilan Penggunaan Perintah Move .....	52

Gambar 3.14.	Tampilan Penggunaan Perintah Fillet .....	53
Gambar 3.15.	Tampilan Coverage yang akan di Edit.....	56
Gambar 3.16.	Tampilan Kesalahan pada Coverage .....	56
Gambar 3.17.	Hubungan Antar Entitas .....	62
Gambar 3.18.	Skema Kerangka Entiti.....	63
Gambar 3.19.	Tampilan Pembuatan Basis Data.....	64
Gambar 3.20.	Tampilan Kotak Dialog Show Tabel.....	65
Gambar 3.21.	Tampilan Jendela Kerja Relationship.....	66
Gambar 3.22.	Tampilan Relationship antar Tabel .....	67
Gambar 3.23.	Tampilan Jendela Kerja Query .....	69
Gambar 3.24.	Tampilan Jendela Kerja Select Query .....	69
Gambar 3.25.	Tampilan Jendela Kerja Select Tabel dan Field .....	70
Gambar 3.26.	Tampilan Jendela Kerja Select siap di Jalankan .....	71
Gambar 3.27.	Tampilan Cara Mengexport Database .....	72
Gambar 3.28.	Tampilan Penggunaan Perintah Joint pada ArcView .....	73
Gambar 3.29.	Tampilan Perintah Convert File pada ArcView .....	74
Gambar 3.30.	Tampilan Form Main dan Penulisan Koding .....	75
Gambar 3.31.	Tampilan Form Data dan Penulisan Koding.....	76
Gambar 3.32.	Tampilan Form Identifikasi dan Penulisan Koding.....	76
Gambar 3.33.	Tampilan Form Detail dan Penulisan Koding.....	77
Gambar 3.34.	Tampilan Form Pencarian dan Penulisan Koding.....	77
Gambar 3.35.	Tampilan Form Add Point dan Penulisan Koding .....	77
Gambar 3.36.	Tampilan Perintah Module dan Jendela Module .....	78
Gambar 3.37.	Program Package & Development Wizard.....	79
Gambar 3.38.	Package Toolbars .....	80
Gambar 3.39.	Tampilan Program Informasi .....	81
Gambar 4.1.	Menu Utama Dalam Sistem Informasi Pemasaran .....	82
Gambar 4.2.	Tampilan Form untuk Memasukan Koordinat .....	83
Gambar 4.3.	Tampilan Menu Legenda .....	83
Gambar 4.4.	Penyajian Peta Informasi Obyek Pemasaran Properti .....	82

Gambar 4.5.	Tampilan Informasi Individual Properti.....	85
Gambar 4.6.	Tampilan Informasi Detail Properti.....	85
Gambar 4.7.	Tampilan Data Atribut Obyek Properti.....	87
Gambar 4.8.	Tampilan Form Pencarian dari Menu Cari Properti.....	89
Gambar 4.9.	Hasil Pencarian dalam bentuk Tabel dan Simbol Lokasi ...	89
Gambar 4.10.	Tampilan Form Pencarian dari Menu Data .....	89
Gambar 4.11.	Tampilan Form Updating.....	91
Gambar 4.12.	Form MDI.....	94

## **DAFTAR TABEL**

Tabel	1.1.	Perbandingan Untung Rugi Cara Transaksi Properti .....	26
Tabel	3.1.	Pengkodean Data Spasial [Feature Titik] .....	59
Tabel	3.2.	Pengkodean Data Spasial [Feature Poligon] .....	60
Tabel	3.3.	Pengkodean Data Spasial [Feature Garis] .....	60
Tabel	3.4.	Tabel Entitas .....	61
Tabel	4.1.	Data Koordinat Posisi Obyek Properti .....	92



# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Sebagian besar hal-hal yang berkaitan dengan sumber daya alam, sosial dan ekonomi keberadaannya terkait erat dengan kriteria spasial atau geografis, yakni lokasinya di permukaan bumi. Mengeksplorasi dimensi spasial pada hakekatnya mencari lokasi terbaik sumberdaya tersebut berada, sehingga keberhasilan menemukan lokasi yang tepat dapat merupakan langkah penting bagi awal keberhasilan usaha, termasuk salah satunya adalah pemilihan atau pembelian properti.

Bagi orang yang awam soal jual beli, termasuk cara memasarkan rumah, negoisasi hingga proses transaksi, bantuan seorang broker properti sangat diperlukan. Seorang broker properti profesional akan memberikan *advis* terbaik bagi pelanggannya. Seperti membantu menetapkan harga jual yang pantas atau mengecek kelengkapan dan keabsahan dokumen dari properti yang akan dibeli. Selain itu seorang broker properti juga akan memberikan gambaran mengenai nilai investasi bila properti tersebut dijual kembali di masa mendatang.

Bagi seorang broker properti, pemberian informasi atau data-data kelengkapan sebuah properti yang akan dijual atau ditawarkan pada

pelanggan akan sangat membantu tercapainya sebuah transaksi, dalam artian pelanggan akan merasa puas.

## **1.2. Maksud dan Tujuan Penelitian**

Maksud dari penelitian ini adalah memanfaatkan penggunaan basisdata sistem informasi geografis untuk inventarisasi perumahan sekunder sekaligus kemudahan sebagai alat bantu presentasi pemasarannya. Sedangkan tujuan yang akan dicapai dalam penelitian ini adalah :

1. Menyajikan informasi dalam bentuk digital tentang data properti yang ditawarkan dalam bentuk basis data peta lokasi, dan visualisasi foto kenampakan obyek.
2. Pembuatan program berorientasi obyek dengan memanfaatkan Visual Basic sebagai piranti lunaknya.

## **1.3. Manfaat Penelitian**

Manfaat dari hasil penelitian yang dapat di sosialisasikan adalah :

1. Perusahaan broker properti pada khususnya dapat melakukan transformasi data-data spasial dan non spasial kedalam data digital yang sudah terkomputerisasi dengan tingkat aksesibilitas pemanfaatan data yang optimal dan dapat ditampilkan kembali secara mudah, cepat dan tepat.

2. Kemudahan dalam pencarian data dan informasi sesuai dengan keinginan konsumen.
3. Membantu seorang broker properti dalam kemudahan pemasaran.
4. Memudahkan konsumen mendapatkan informasi properti sekunder.

#### **1.4. Batasan Masalah**

Batasan masalah dari penelitian ini adalah memfokuskan penyajian informasi pemasaran perumahan sekunder di Kota Malang, dalam hal ini pemasaran di perusahaan broker properti ERA Puri Properti di wilayah Malang. Sedangkan pembuatan program untuk penampilan hasil akan menggunakan bahasa komputer Visual Basic.

#### **1.5. Tinjauan Pustaka**

Sistem Informasi Geografis adalah suatu sistem berkomputer yang mempunyai kemampuan untuk membangun, menyimpan, memanipulasi, dan menayangkan informasi yang bereferensi geografis, yaitu data yang diidentifikasi sesuai dengan lokasinya. Sementara pengertian lain menyertakan unsur operator (sumberdaya manusia) dan data masukan sebagai bagian dari SIG secara keseluruhan. [Sri Handoyo,1996].

Ilustrasi SIG menurut PA. Burrough menunjukkan dua faktor penting yang mempengaruhi definisi dari SIG. Pertama, terdapat beberapa bidang ilmu dan kegiatan yang menggunakan SIG seperti

perencanaan tata kota, pembangunan real estat, kegiatan pemasaran dan lainnya. Kedua, sistem informasi geografis mempunyai kemampuan untuk membantu meningkatkan proses kerja kegiatan-kegiatan tersebut. SIG menyediakan beberapa kemampuan untuk meningkatkan efisiensi dan efektifitas bekerja dengan informasi visual maupun tabel. [Arief N Rahardjo].<sup>~1</sup>

SIG yang digunakan dan dikembangkan saat ini, banyak menggunakan sistem-sistem manajemen basisdata (DBMS) yang telah lahir sebelumnya, hal ini karena DBMS telah demikian banyak memiliki dan menangani fungsi-fungsi (dan prosedur) yang sangat diperlukan oleh SIG. Dengan demikian, sebagian besar fungsi dan prosedur dasar yang ada pada SIG sudah disediakan oleh DBMS-nya. Jika tidak, fungsi-fungsi atau prosedur tersebut harus diprogram khusus untuk SIG. [Prahasta,2001].

Visual Basic adalah suatu *development tools* untuk membangun aplikasi dalam lingkungan windows. Dalam pengembangan aplikasi, Visual Basic menggunakan pendekatan visual untuk merancang *user interface* dalam bentuk *form*, sedangkan kodingnya menggunakan dialek bahasa basic. [Daryanto,2003].

---

<sup>~1</sup> Makalah Seminar SIG untuk Perkotaan dan Real Estat, Lembaga Geomatika Perdana Internusa, 1997

## **1.6. Landasan Teori**

### **1.6.1. SIG sebagai Basisdata**

Pengembangan SIG dimulai dari awal sekali, dengan menggunakan *tools* yang sangat terbatas baik jumlah maupun kemampuannya. Tetapi pada saat ini, SIG dikembangkan dengan menggunakan sistem-sistem manajemen basisdata (DBMS) yang telah lahir sebelumnya. Beberapa DBMS menggunakan beberapa asumsi mengenai data-data yang ditanganinya. Untuk mengefektifkan penggunaan DBMS, maka asumsi ini harus dipenuhi. Beberapa tipe DBMS lebih sesuai untuk SIG daripada tipe yang lain, karena DBMS ini memiliki asumsi yang sesuai untuk data spasial SIG yang bersangkutan. Pada umumnya terdapat dua pendekatan untuk menggunakan DBMS di dalam SIG.<sup>~2</sup>

Pertama adalah pendekatan solusi DBMS total, yaitu semua data spasial dan non spasial diakses melalui DBMS sehingga data tersebut harus memenuhi asumsi-asumsi yang telah ditentukan oleh perancang DBMS-nya.

Yang kedua adalah pendekatan solusi kombinasi. Pada pendekatan ini beberapa data (pada umumnya berupa tabel atribut berikut relasinya) diakses melalui DBMS karena data-data tersebut telah sesuai dengan modelnya.

---

<sup>~2</sup> Konsep-Konsep Dasar SIG, Eddy Prahasta 2001 hal.180.

### **1.6.2. Pengertian Sistem Informasi Geografis ( SIG )**

Konsep dasar Sistem Informasi Geografis (SIG) adalah merupakan suatu sistem yang mengorganisir perangkat keras dan lunak serta data sehingga dapat mendayagunakan sistem penyimpanan, pengolahan maupun analisa data secara simultan, sehingga dapat diperoleh informasi yang berkaitan dengan aspek keruangan. SIG juga merupakan manajemen data spasial dan non spasial yang berbasis komputer dan mampu menyajikan secara bersama-sama. [Purwadhi, 1999].

Istilah sistem informasi geografis memiliki tiga unsur pokok : sistem, informasi, dan geografis. SIG merupakan salah satu sistem informasi yang menekankan pada unsur informasi geografis, oleh karena itu sistem informasi geografis merupakan suatu kesatuan formal yang terdiri dari berbagai sumberdaya fisik dan logika yang berkenaan dengan obyek-obyek yang terdapat dipermukaan bumi.[Yeyep Yousman,2004]<sup>~3</sup>

Teknologi SIG memungkinkan untuk penggunaan basis data dengan informasi keruangan yang ditampilkan dalam bentuk grafis pada layar komputer atau disebut peta digital.

### **1.6.3. Komponen Sistem Informasi Geografis (SIG)**

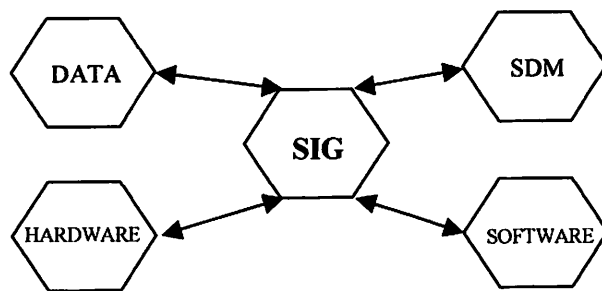
Sistem Informasi Geospasial terdiri dari empat komponen dasar, yaitu data, perangkat lunak, perangkat keras, dan sumberdaya manusia

---

<sup>~3</sup> SIG dengan MapInfo Propesional,2004 hal 09



atau pengguna SIG. Komponen tersebut saling berhubungan dan berkomunikasi seperti pada gambar 1.1. dimana data merupakan komponen utama yang akan diproses. Perangkat lunak merupakan komponen untuk mengintegrasikan berbagai macam data masukan yang akan diproses. Perangkat keras dapat berupa komputer yang dilengkapi dengan peralatan dijital, *scanner*, *plotter*, monitor, dan printer. Sedangkan sumberdaya manusia merupakan pengguna sistem dan yang mengoperasikan perangkat lunak maupun perangkat keras, serta data yang diolah, maupun dianalisa sesuai dengan kebutuhan.



Gambar 1.1. Struktur Komponen SIG

SIG merupakan suatu siklus mulai pengumpulan data dari permukaan bumi dan dilakukan input data dalam suatu *database* sehingga dapat dilakukan manipulasi dan analisa untuk menghasilkan informasi yang diimplementasikan ke permukaan bumi. Dari siklus tersebut secara garis besar ada empat komponen utama SIG adalah sebagai berikut :

#### 1.6.3.1. Data Input (Data Masukan)

Komponen ini bertugas untuk mengumpulkan dan mempersiapkan data spasial dan atribut dari berbagai sumber. Komponen ini pula yang bertanggung jawab dalam mengkonversi atau mentransformasikan format data aslinya kedalam format yang digunakan oleh SIG. Data input SIG dapat berupa :

- ~ Data Photo Udara
- ~ Data Penginderaan Jauh (Citra)
- ~ Data Peta
- ~ Data Survei Lapangan
- ~ Data GPS
- ~ Data Tabular (Non Spasial)
- ~ Data Base Digital yang lain.

Tipe-tipe data masukan yang dapat digunakan oleh SIG meliputi :

1. Jaringan Titik Geodesi, merupakan tingkat ketelitian jaringan titik kontrol.
2. Unsur-unsur Topografi, misalnya : jalan, jembatan, bangunan, danau, tambak, sungai, hutan, dan lain-lain.
3. Unsur-unsur Kadastral, misalnya : persih tanah dan atributnya.
4. Unsur-unsur Batas Luasan, misalnya: batas kota, batas kecamatan, batas kelurahan, batas desa, dan lainnya.
5. Unsur-unsur Utilitas, misalnya: jaringan telepon, air minum, listrik dan lain-lain.
6. Zona Sosial Ekonomi, misalnya: tingkat kepadatan penduduk, tingkat kesejahteraan, jumlah penduduk, dan lain-lain.

Dari tipe-tipe data masukan diatas, secara garis besar data input dapat dibedakan menjadi dua yaitu data spasial dan data non spasial.

#### **A. Data Spasial**

Data merupakan bahan dasar yang diolah atau diproses sehingga menjadi informasi yang dapat digunakan untuk keperluan tertentu. Data spasial adalah data mengenai obyek-obyek atau unsur geografis (baik dibawah, diatas, dan dipermukaan bumi) yang dapat diidentifikasi dan mempunyai acuan lokasi berdasarkan sistem koordinat tertentu atau bergeoreferensi. Tipe data spasial yang paling umum digunakan adalah :

##### **1. Model Data Vektor**

Model data vektor menampilkan, menempatkan, dan menyimpan data spasial dengan menggunakan titik, garis, atau poligon. Bentuk-bentuk dasar representasi data spasial ini, didalam sistem model data vektor didefinisikan oleh sistem koordinat kartesian dua dimensi: (X,Y). Model data ini biasanya diperoleh dari hasil digitasi.

##### **2. Model Data Raster**

Model data raster menampilkan, menempatkan, dan menyimpan data spasial dengan menggunakan struktur matriks atau piksel-piksel yang membentuk grid. Struktur data dinyatakan dalam bentuk sel yang terbentuk atas baris dan kolom dari kiri atas, setiap



data tergantung dari bagaimana data diorganisasi atau diatur didalam media penyimpanan data. Penyimpanan dan pengorganisasian data berdasarkan jenis dan struktur data, data spasial disimpan dalam *file* grafis sedangkan data *non spasial* disimpan dalam bentuk tabel.

#### **1.6.3.3. Data Manipulasi dan Analisa**

Manipulasi data dalam SIG meliputi data spasial dan non spasial. Manipulasi data spasial dapat berupa perbaikan karena adanya kesalahan digitasi, penambahan dan pengurangan *arc*, label, adanya *Undeshoot* dan *Overshoot*, *Edgemacth* peta dan lainnya. Sedangkan manipulasi data non spasial berupa menampilkan, penambahan, pengurangan dari label, perbaikan data dan manipulasi lainnya. Peranan operator sangat penting dalam menentukan model analisa dan fungsi-fungsi SIG untuk penentuan persyaratan informasi yang akan ditampilkan.

#### **1.6.3.4. Data Output atau Visualisasi Hasil**

Komponen ini berfungsi menampilkan atau menghasilkan keluaran seluruh atau sebagian basidata, baik dalam bentuk *softcopy* atau *hardcopy*. dalam format tabel, grafik, peta atau format lainnya.

#### **1.6.4. Pengertian Basisdata**

Basisdata merupakan kumpulan data *non-redundant* yang dapat digunakan bersama oleh sistem-sistem aplikasi yang berbeda. Dengan

kata lain, basisdata adalah kumpulan data-data yang terkait satu dengan lainnya (dinyatakan oleh atribut-atribut kunci dari tabel-tabelnya / struktur data dan relasi-relasi) dalam membangun informasi yang penting. [Prahasta 2001].

Definisi lain basisdata adalah penyajian suatu aspek dari dunia nyata yang merupakan kumpulan data dari berbagai sumber yang secara logika mempunyai arti implisit dan perlu dirancang, dibangun dan dikumpulkan untuk suatu tujuan. Pengelolaan basisdata dapat dilakukan secara manual maupun dengan komputer. [Waljiyanto,2000].

Sedangkan pengertian sistem basisdata menurut [Fathan99], secara umum, sistem basisdata merupakan sistem yang terdiri dari kumpulan file (tabel) yang saling berhubungan (dalam sebuah basisdata di sebuah sistem komputer) dan sekumpulan program (DBMS) yang memungkinkan beberapa pemakai dan atau program lain untuk mengakses dan memanipulasi file-file tersebut.

Komponen pengelolaan basisdata secara fisik tidak dilakukan secara langsung oleh pengguna, tetapi ditangani oleh sistem perangkat lunak khusus (DBMS) yang akan menentukan bagaimana data diorganisasikan, disimpan, diubah, dan dipanggil. Perangkat lunak ini pula yang menerapkan mekanisme pengamanan data, penggunaan data bersama (data *sharing*), konsistensi data, dan sebagainya.

#### 1.6.4.1. Sistem Manajemen Basisdata (DBMS)

Sistem Manajemen Basisdata adalah kumpulan (gabungan) dari data yang saling berelasi (yang biasanya dirujuk sebagai basisdata) dengan sekumpulan program-program yang mengakses data-data tersebut. [Korth 91]. Menurut [Ade20a] DBMS adalah tempat penyimpanan data beserta *user interface*-nya yang dipersiapkan untuk memanipulasi dan administrasi basisdata. Dengan demikian, DBMS juga dapat dianggap sebagai sistem perangkat lunak.

Definisi lain dari DBMS adalah merupakan suatu program komputer yang digunakan untuk memasukkan, mengubah, menghapus, memanipulasi, dan memperoleh data dan informasi dengan praktis dan efisien. [Kadir 99].

Sedangkan komponen-komponen yang terdapat pada Sistem Manajemen Basisdata, menurut [Hkbu 20]<sup>4</sup>, sistem manajemen basisdata (DBMS) dapat dibentuk dari komponen-komponen sebagai berikut :

1. **Data** yang disimpan didalam basisdata
2. **Operasi standar** yang disediakan oleh hampir semua DBMS. Operasi ini melengkapi pengguna dengan kemampuan dasar untuk memanipulasi data.
3. **DDL** (*Data Desinition Language*) yang merupakan bahasa yang digunakan untuk mendeskripsikan isi (dan struktur) basisdata.

---

<sup>4</sup> Konsep-Konsep Dasar SIG, Eddy Prahasta 2001 hal.192

4. DML (*Data Manipulation Language*) atau bahasa *query*, untuk membentuk perintah-perintah untuk masukan, keluaran, *editing*, analisis basisdata atau dikenal dengan SQL.
5. Bahasa pemrograman, disamping oleh perintah-perintah dan *queries*, basisdata harus bisa diakses secara langsung oleh program aplikasi melalui bahasa pemrograman ini.
6. Struktur file, digunakan untuk mengorganisasikan data.

#### **1.6.4.2. Merancang Basis Data SIG**

Ada 3 tahapan yang harus dilakukan dalam merancang suatu basis data yaitu [ menurut Schema dan Preuner01]<sup>~5</sup> :

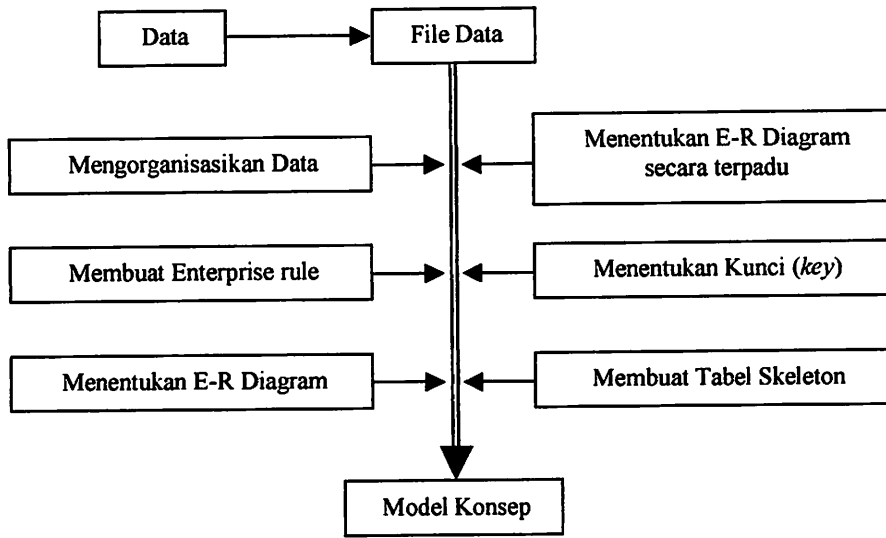
1. Tahap Eksternal, yaitu tahap mengidentifikasi kebutuhan pengguna, yaitu dilakukan identifikasi semua kebutuhan ( berikut tujuan yang hendak dicapai) pengguna yang berhubungan dengan basisdatanya. Aktivitas identifikasi kebutuhan pengguna ini pada umumnya dapat ditempuh dengan cara melakukan studi pustaka, wawancara (interview) dengan calon pengguna, dan penyebaran angket/kuesioner. Setelah itu hasilnya diinventarisasikan, diklasifikasikan, dievaluasi, dan diformulasikan menjadi "kebutuhan pengguna".

---

<sup>~5</sup> Konsep-Konsep Dasar SIG, Eddy Prahasta 2001 hal.283



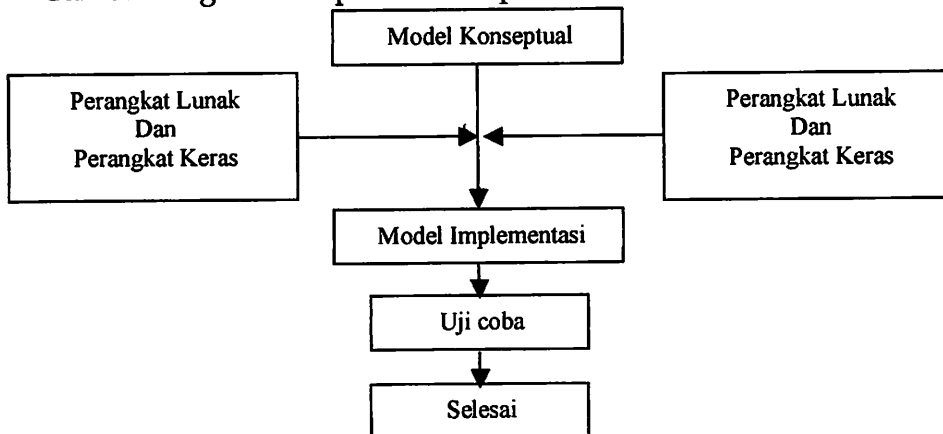
2. Tahap Konseptual, yaitu tahap mengorganisasi data, memilih, mengelompokkan, menyederhanakan data, menetapkan *enterprise rule*, membuat *entity relationship* ( E-R ) diagram, menetapkan kunci ( *key* ) dan membuat tabel skeleton secara terstruktur.



Gambar 1.3. Diagram Tahap Konseptual

3. Tahap Internal, yaitu tahap mengimplementasikan tabel yang telah dirancang ke dalam perangkat lunak kemudian dilakukan uji coba.

Gambar diagram tahap internal seperti berikut :

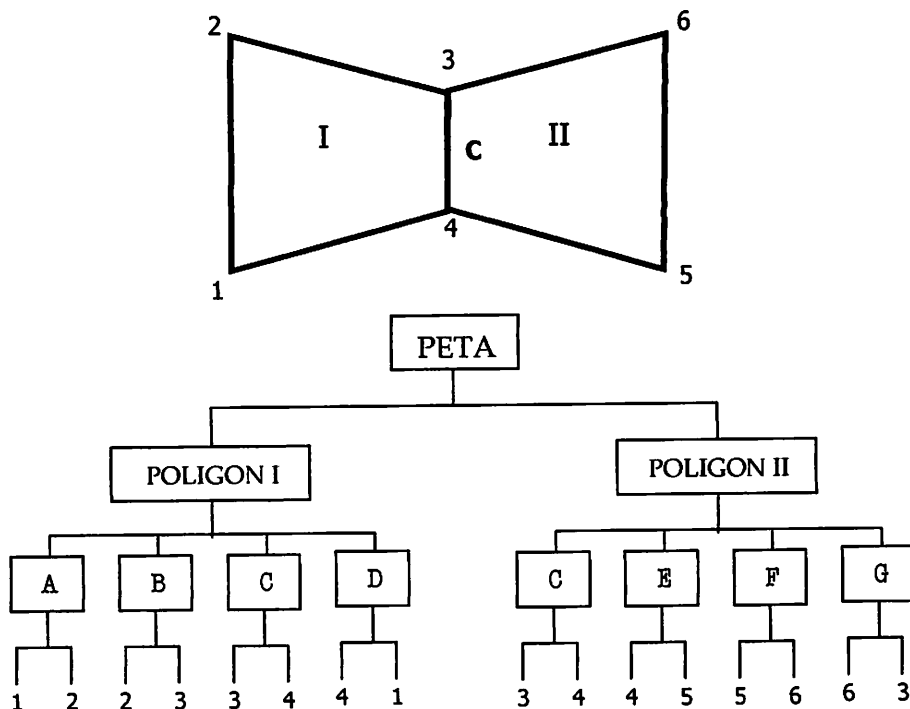


Gambar 1.4. Diagram Tahap Internal

### 1.6.4.3. Struktur Basis Data

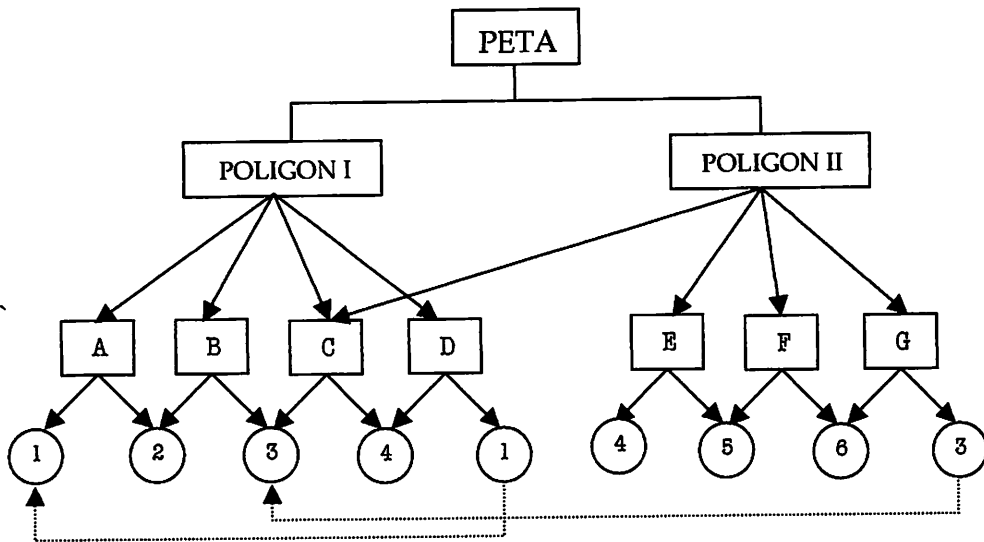
Dalam penyusunan suatu sistem basis data, maka yang perlu ditinjau pertama kali adalah pembuatan struktur sistem basis data. Adapun struktur sebuah sistem basis data adalah sebagai berikut :

1. Struktur *Database Herarki*, dibuat pada tahun 1970-1980 mempunyai beberapa karakteristik, yaitu :
  - a. Struktur databasenya seperti pohon ( satu anak hanya mempunyai satu orang tua ).
  - b. Sangat cepat dan mudah dalam mendapatkan suatu data.
  - c. Pembentukan kembali struktur dari sebuah database adalah kompleks.
  - d. Tidak fleksibel dalam query data ( pola hanya keatas dan kebawah , tidak bisa akses perpotongan dari kumpulan data).
  - e. Hubungan data *one to one* ( 1:1 ) atau *one to many* ( 1:M ) dapat dikerjakan.
  - f. Untuk mengambil data *many to many* yang redanden harus ada.



Gambar 1.5. Struktur Database Hirarki

2. Struktur *database Network*, dibuat pada tahun 1970-1980 mempunyai beberapa karakteristik, yaitu :
  - a. Struktur databasenya berupa pohon ( seorang anak dapat mempunyai lebih dari satu orang tua ).
  - b. Tidak ada redanden tetapi dibutuhkan banyak pointer (perpotongan kumpulan data ).
  - c. Mudah dan cepat dalam mendapatkan sebuah data.
  - d. Lebih fleksibel didalam query data, tetapi lebih sedikit kompleks.
  - e. Semua databasenya *one to one* ( 1:1 ), *one to many* ( 1:M ), *many to many* ( M:N ) dapat dikuasai atau dihandel.

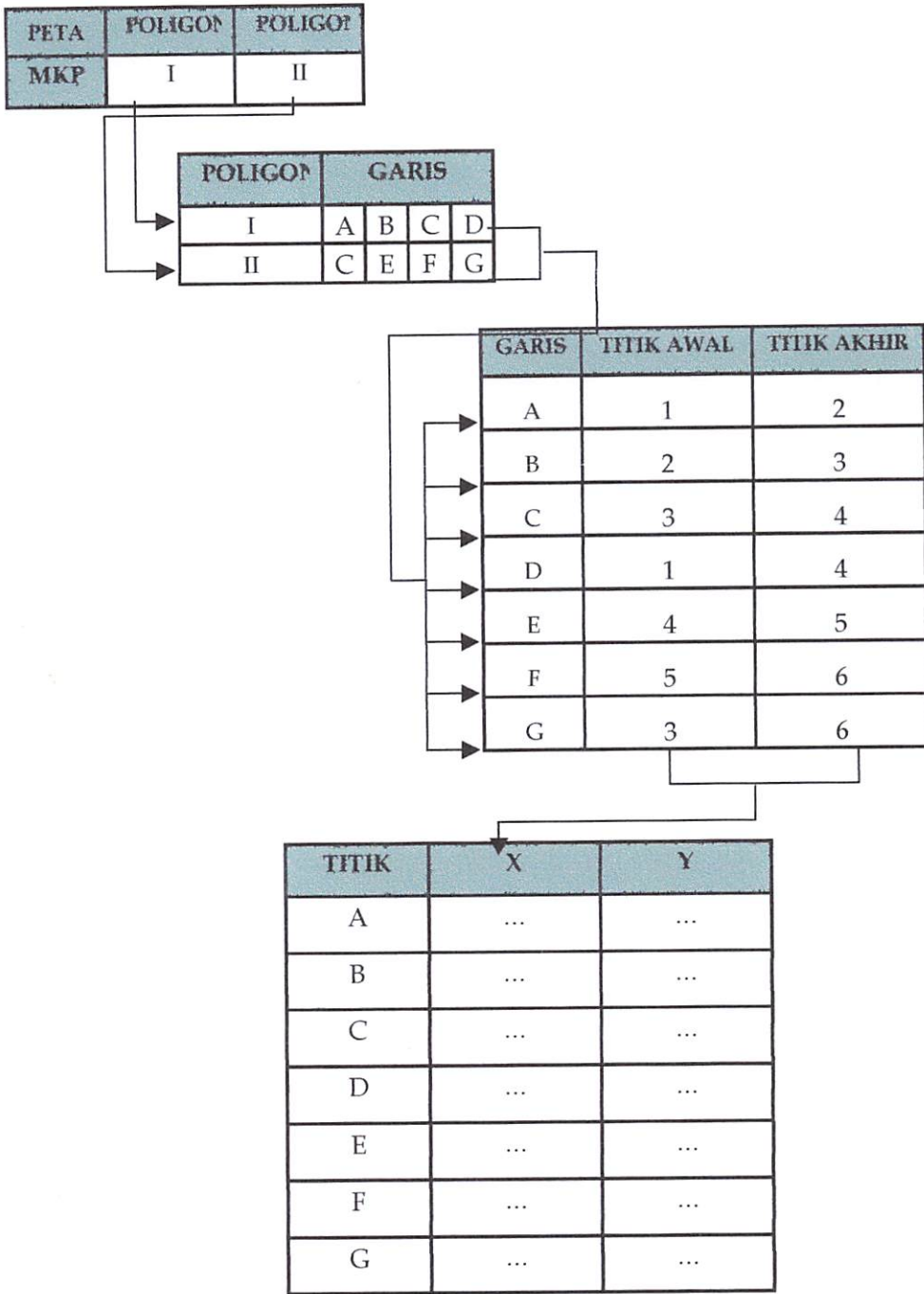


Gambar 1.6. Struktur Database Network

3. Struktur *Database Relational*, merupakan model yang paling sederhana, sehingga mudah digunakan dan di pahami oleh pengguna serta yang paling populer pada saat ini. Model ini menggunakan sekumpulan tabel berdimensi dua ( yang disebut relasi atau tabel ), dengan masing-masing relasi tersusun atas baris dan atribut.

Struktur *Database Relational* mempunyai karakteristik, yaitu :

- a. Penggunaan desain metodologi.
- b. Struktur databasenya yang simpel dan sederhana ( semua data disimpan didalam dua dimensional tabel ).
- c. Semua databasenya *one to one* ( 1:1 ), *one to many* ( 1:M ), *many to many* ( M:N ) dapat dihandel.
- d. Tidak ada data redandent ( normalisasi tabel ).
- e. Sangat baik dan *standart query* ( SQL ).



Gambar 1.7. Struktur Database Relational.

#### **1.6.4.5. Model Data dari Sistem Basis Data**

Dalam model data konseptual digunakan konsep entiti ( *entity* ), atribut ( *attribut* ), dan hubungan ( *relationship* ). Pengertian ketiga istilah tersebut adalah sebagai berikut :

1. Entity ( *Entitas* ), merupakan penyajian obyek, kejadian atau konsep dari dunia nyata yang keberadaannya secara eksplisit didefinisikan dan disimpan dalam basisdata, contoh : OBYEK, PEMBELI
2. Atribut ( *attribut* ), merupakan keterangan - keterangan yang dimiliki oleh suatu entity, seperti *Nama\_Obyek, Lokasi, Luas, Status, Harga* pada entity OBYEK
3. Hubungan ( *relationship* ), merupakan interaksi antar entity satu dengan lainnya, misalnya JUAL menyatakan hubungan antara entity OBYEK dan PEMBELI.

#### **1.6.4.6. Konsep Hubungan Antar Entity ( E - R )**

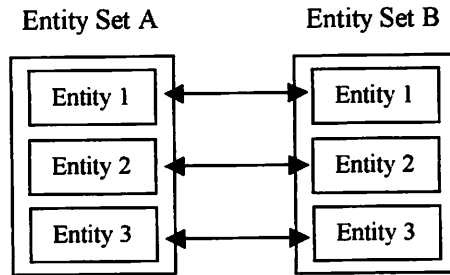
Hubungan antar dua file atau dua tabel dapat dikategorikan menjadi tiga macam. Demikian untuk membantu gambaran hubungan secara lengkap terdapat juga tiga macam relasi dalam hubungan atribut dalam suatu file.<sup>~6</sup>

##### **1. Derajat hubungan *one to one* dengan 2 file**

---

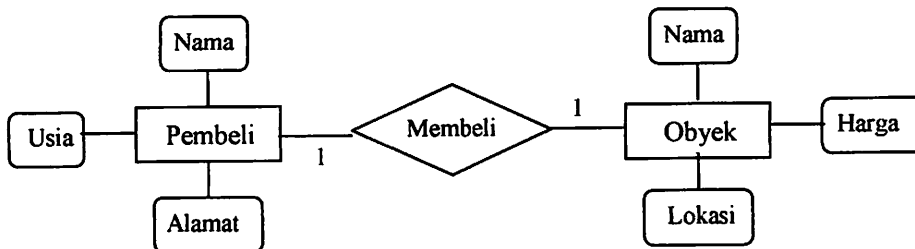
<sup>~6</sup> Konsep-Konsep Dasar SIG, Eddy Prahasta 2001 hal.99-103

Dengan relasi seperti ini, maka setiap entity pada *entity set A* berhubungan satu (paling banyak) entity pada *entity set B*. Demikian pula sebaliknya.



Gambar 1.8. Tingkat relasi satu ke satu

Sedangkan untuk diagram entity-relationship (ER) dicontohkan seperti berikut : Dimana pada diagram ini memperlihatkan adanya relasi antara *entity set* pembeli dengan *entity set* obyek pada sebuah transaksi. *Relationship set*-nya dinamai "Membeli". Dengan relasi ini, setiap pembeli akan membeli satu obyek, dan setiap obyek akan dibeli satu pembeli.

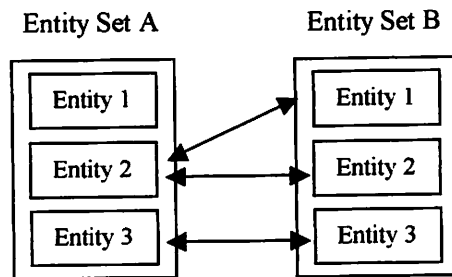


Gambar 1.9. Diagram ER dengan relasi satu ke satu

## 2. Derajat hubungan *one to many* dengan 2 file

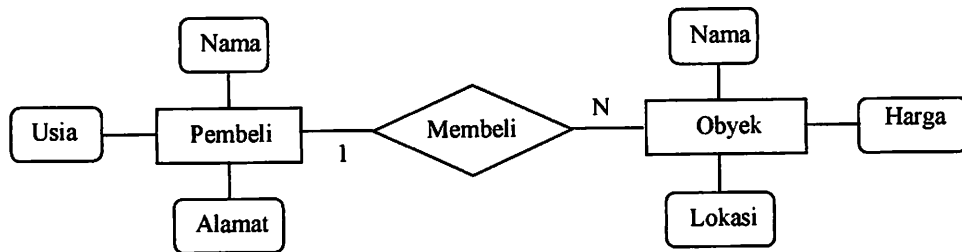
Dengan relasi seperti ini, maka setiap *entity* pada *entity set A* dapat berhubungan dengan lebih dari satu (banyak) *entity* pada *entity set B*, dan

setiap *entity* pada *entity set B* hanya dapat berhubungan dengan satu *entity* pada *entity set A*.



Gambar 1.10. Tingkat relasi satu ke banyak

Diagram ER dapat dicontohkan seperti pada pameran perumahan dimana satu pembeli dapat membeli lebih dari satu (banyak) obyek, dan setiap obyek hanya dibeli oleh satu pembeli saja. Pada diagram ER ini terdapat adanya relasi antara *entity set* Pembeli dengan *entity set* Obyek, dan *relationship set*-nya dinamai "Membeli".

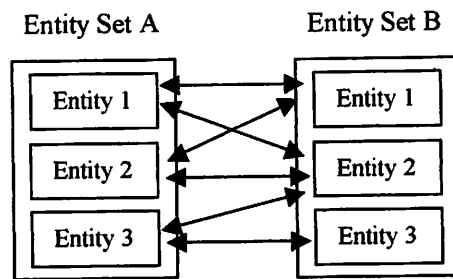


Gambar 1.11. Diagram ER dengan relasi satu ke banyak

### 3. Derajat hubungan *many to many* dengan 2 file

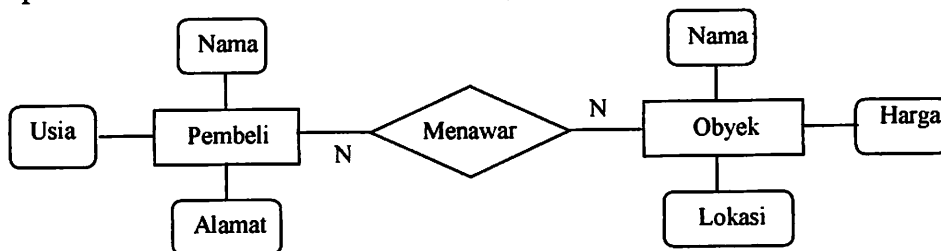
Dengan relasi ini, setiap *entity* pada *entity set A* boleh berhubungan dengan banyak *entity* dari *entity set B*, demikian pula sebaliknya, setiap *entity* pada *entity set B* boleh berhubungan dengan banyak *entity* pada *entity set A*.





Gambar 1.12. Tingkat relasi banyak ke banyak

Diagram ER untuk relasi *many to many* dapat dicontohkan adanya relasi *entity set* Obyek dengan *entity set* Pembeli, *relationship set*-nya dinamai "Menawar". Dengan relasi ini, setiap pembeli (banyak) dapat menawarkan lebih dari satu (banyak) obyek. Begitu pula dengan setiap obyek yang dapat ditawarkan oleh lebih dari satu (banyak) pembeli.



Gambar 1.13. Diagram ER untuk relasi banyak ke banyak

### 1.6.5. Transaksi Properti

Secara umum transaksi properti dapat dibagi dalam 3 (tiga) kategori, yaitu jual beli, sewa menyewa, dan kerjasama yang dapat dibagi dalam beberapa variasi bentuk kerjasama, antara lain kerja sama operasional (KSO), bagi hasil, bangunan guna serah (BOT - *Built, Operate, and Transfer*), dan banyak lagi bentuk-bentuk lain yang merupakan kombinasi dari kerjasama.

Untuk kepentingan konsumen properti secara langsung, jual beli properti yang banyak mengandung konflik antara pengembang (penjual) dengan konsumen (pembeli), terlebih lagi bila transaksi dilakukan dengan sistem *pre-project selling*, dimana jual beli dengan penandatanganan Akta Jual Beli (AJB) dihadapan PPAT belum dapat dilaksanakan karena barang/properti-nya (obyek jual beli) belum dapat diserahkan. Sedangkan dalam hukum pertanahan menganut azas tunai dan tuntas, maksudnya pembayaran dilakukan secara tunai/lunas dan pengalihan hak atas tanah/properti dilakukan saat itu juga. Karena itu, sebaiknya transaksi dilakukan dengan terlebih dahulu para pihak (penjual dan pembeli) menandatangani Perjanjian Pengikatan untuk Jual Beli (PPJB), yang berisi segala ketentuan-ketentuan termasuk hak dan kewajiban para pihak yang wajib dijalankan sebelum dan sesudah ditandatanganinya AJB.

#### **1.6.5.1. Macam Cara Transaksi Properti**

Secara umum ada 3 (tiga) cara dalam menjalankan transaksi properti, khususnya dalam jual, beli dan atau sewa properti, yaitu pertama bisa melalui developer, kedua dengan cara meminta bantuan agen properti, dan ketiga transaksi secara langsung.

Developer sebetulnya setara pihak pertama, dimana biasanya developer menawarkan dan menjual rumah dikawasan yang dikembangkan, cara penawarannya beragam. Bisa lewat iklan dimedia

massa atau lewat brosur, dan lewat ajang pameran properti. Developer biasanya menjual rumah baru. Beberapa menjual rumah *Ready Stock* (stok jadi) yang lain secara *Indent* (pemesanan). Perantara lain dalam transaksi properti ada agen properti. Di Indonesia sepak terjang agen properti semakin kencang. Tidak hanya agen-agen *franchise*, tetapi juga agen lokal sampai makelar. Agen lokal atau *franchise* biasanya lebih profesional daripada makelar. Kalau agen lebih tahu soal hukum dan legalitas, makelar lebih mengutamakan "dagangannya" cepat laku.

Agen properti merupakan perusahaan jasa yang membantu klien menjual, membeli, menyewakan, dan atau melelang properti. Agen bisa menjadi mediator antara penjual dan pembeli dalam menyelesaikan transaksi properti, juga ikut membantu semua proses pengalihan tersebut secara sah dan sesuai prosedur. Dalam proses kerjanya, pelayanan yang diberikan agen properti terdiri dari pelayanan utama dan pelayanan tambahan. Pelayanan utama berupa program pemasaran, sedangkan pelayanan tambahan berkaitan dengan hal penyampaian informasi mengenai harga properti, pengecekan surat-surat, dan pelayanan untuk proses KPR. Agen properti juga melakukan kerjasama dengan beberapa bank yang memberikan fasilitas KPR.

Cara lain dalam transaksi properti adalah beli langsung tanpa lewat perantara. Keuntungan dari membeli secara langsung adalah bisa menawar harga rumah langsung dengan penjual rumah, meski tak jarang

harga enggan turun. Walau demikian, tetap merasa lebih puas karena mengerjakan sendiri. Mulai dari mencari rumah yang sesuai, tawar menawar harga dan kondisi rumah, mengurus surat-surat administrasi dan jual beli, serta mencari sumber pembiayaan.

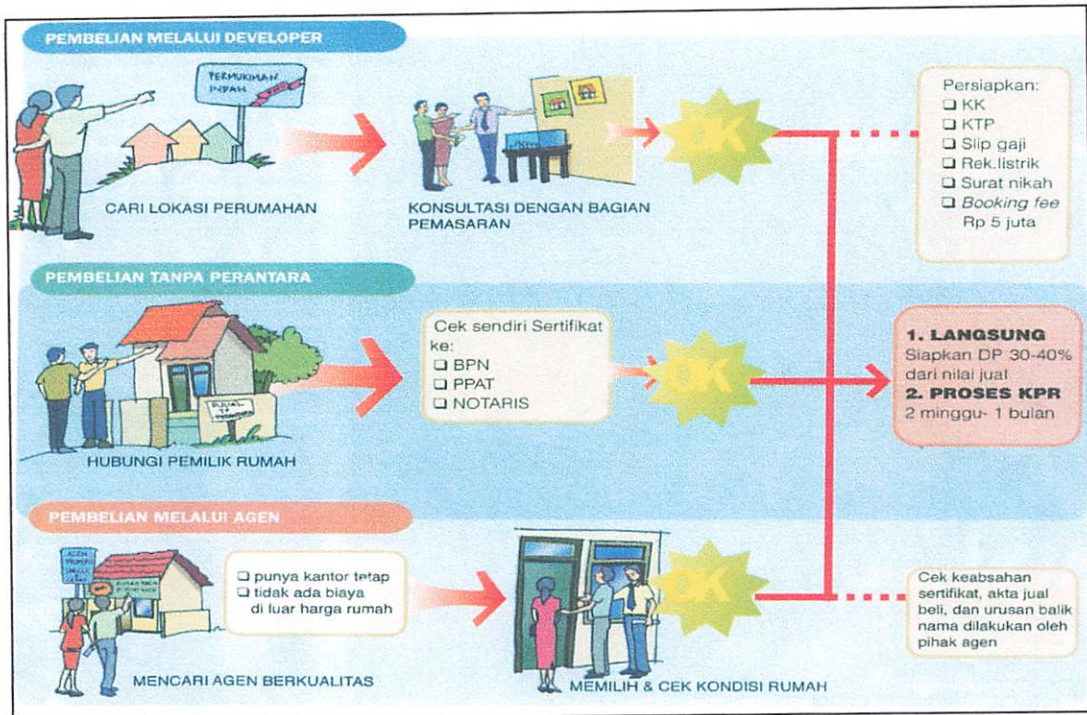
Berikut adalah tabel perbandingan untung rugi 3 (tiga) cara dalam transaksi properti :

Tabel 1.1. Perbandingan untung rugi cara transaksi properti

	Transaksi Properti		
	Secara langsung	Melalui Developer	Agen Properti
Menentukan pilihan	<ul style="list-style-type: none"> <li>▪ Mencari dan menentukan sendiri</li> </ul>	<ul style="list-style-type: none"> <li>▪ Menawarkan tipe dan lokasi kavling rumah</li> <li>▪ Banyak pilihan</li> <li>▪ Pembeli bisa memilih kavling, model, dan harga</li> </ul>	<ul style="list-style-type: none"> <li>▪ Menentukan sendiri</li> <li>▪ Agen membantu mencari</li> <li>▪ Cukup menyebut harga dan lokasi</li> </ul>
Kondisi rumah	<ul style="list-style-type: none"> <li>▪ Apa adanya</li> <li>▪ Kondisi bisa dibicarakan</li> </ul>	<ul style="list-style-type: none"> <li>▪ Kondisi baik</li> <li>▪ Ada yang bertanggungjawab</li> <li>▪ Garansi</li> </ul>	<ul style="list-style-type: none"> <li>▪ Apa adanya</li> </ul>
Mengurus surat - surat	<ul style="list-style-type: none"> <li>▪ Sendiri</li> <li>▪ Mencari alternatif pembiayaan sendiri</li> </ul>	<ul style="list-style-type: none"> <li>▪ .Developer</li> <li>▪ Ada tawaran solusi pembiayaan dan KPR</li> </ul>	<ul style="list-style-type: none"> <li>▪ Agen</li> <li>▪ Ada tawaran solusi pembiayaan</li> </ul>
Biaya-biaya	<ul style="list-style-type: none"> <li>▪ Bisa lebih murah atau mahal</li> <li>▪ Biaya notaris bisa ditawar</li> <li>▪ KPR lebih kecil dari taksiran <i>appraisal</i></li> <li>▪ Perlu dana tambahan untuk pelunasan harga rumah</li> </ul>	<ul style="list-style-type: none"> <li>▪ Sesuai ketentuan developer</li> <li>▪ Ada <i>booking fee</i></li> <li>▪ Uang muka KPR sekitar 30%</li> </ul>	<ul style="list-style-type: none"> <li>▪ Sesua ketentuan agen</li> <li>▪ Tak ada <i>booking fee</i></li> <li>▪ Uang muka kredit 10% dari harga jual rumah, meliputi biaya administrasi, pajak, dan cicilan pertama.</li> </ul>
Harga rumah	<ul style="list-style-type: none"> <li>▪ Tawar menawar langsung dengan penjual</li> </ul>	<ul style="list-style-type: none"> <li>▪ Harga resmi, sesuai ketentuan developer</li> <li>▪ Biasanya ada diskon</li> <li>▪ Besarnya tergantung cara pembayaran</li> </ul>	<ul style="list-style-type: none"> <li>▪ Tawar menawar dengan agen</li> <li>▪ Masih mungkin turun harga</li> </ul>
Pembayaran	<ul style="list-style-type: none"> <li>▪ Tunai atau KPR</li> </ul>	<ul style="list-style-type: none"> <li>▪ Tunai atau KPR</li> </ul>	<ul style="list-style-type: none"> <li>▪ Tunai atau KPR</li> </ul>

Rata-rata lama proses [memilih hingga memiliki]	<ul style="list-style-type: none"> <li>▪ Dua minggu hingga satu bulan</li> <li>▪ Bisa lebih lama tergantung kesanggupan</li> </ul>	Dua minggu sampai satu bulan	<ul style="list-style-type: none"> <li>▪ Dua minggu sampai satu bulan</li> </ul>
---	--	------------------------------	--

Sumber: Majalah IDEA edisi Juli 2004



Gambar 1.14. Ilustrasi cara dalam transaksi properti

### 1.6.5.2. Rumah Sekunder

Rumah sekunder yang dalam bisnis properti dikenal dengan *secondary class* atau *primary market* dapat diartikan sebagai rumah - rumah yang telah ditempati untuk dijual kembali. Menurut beberapa analisa, rumah sekunder diperkirakan masih mendominasi pasar properti sekunder. Tahun 2005, investasi dibisnis ini masih tetap menjanjikan, sejauh pintar-pintar mencari peluang untuk mengembangkannya.

Seperti dikatakan ketua Asosiasi Real Estat Broker Indonesia (AREBI) kepada *media Sinar harapan* (7/6), perkembangan ini terjadi karena adanya keinginan masyarakat untuk menjual. Masyarakat pun merespon sejauh harga murah. Jadi, meski dipasar primer semakin banyak suplai, tapi pasar sekunder tetap bertumbuh dengan baik.

Di Malang, lokasi strategis bagi properti sekunder diantara adalah daerah Araya, Pondok Blimbing Indah (PBI), Sawojajar, Tidar, dan masih banyak lagi, apalagi di Malang banyak dibangunnya pusat pertokoan dan ruko-ruko.

#### **1.6.6. Perangkat Lunak Visual Basic 6.0.**

Sistem pemrograman Visual Basic merupakan bahasa pemrograman yang mengkombinasikan kemampuan bahasa basic dan piranti desain visual, bahasa ini menyediakan kesederhanaan dan kemudahan penggunaan tanpa mengorbankan kinerja atau fasilitas grafik yang menyebabkan windows menjadi lingkungan kerja yang menyenangkan.

Visual Basic adalah jenis bahasa pemrograman yang didukung konsep SQL ( *Structured Query Language* ). Didalam penggunaanya, aplikasi yang didukung oleh konsep SQL tersebut akan menjadi sebuah aplikasi yang efisien dalam proses pencarian data, terutama bila dilihat dari segi waktu dan penggunaan memori komputer.[Bagus K.2002;81]

Visual basic adalah salah satu bahasa komputer yang mendukung pemrograman *even-driven*, yaitu gaya pemrograman yang sangat cocok untuk antar muka pengguna. Pada pemrograman *even-driven* sasarannya adalah menjadikan pemakai sebagai orang yang berkuasa [ Nelson,1996 ].

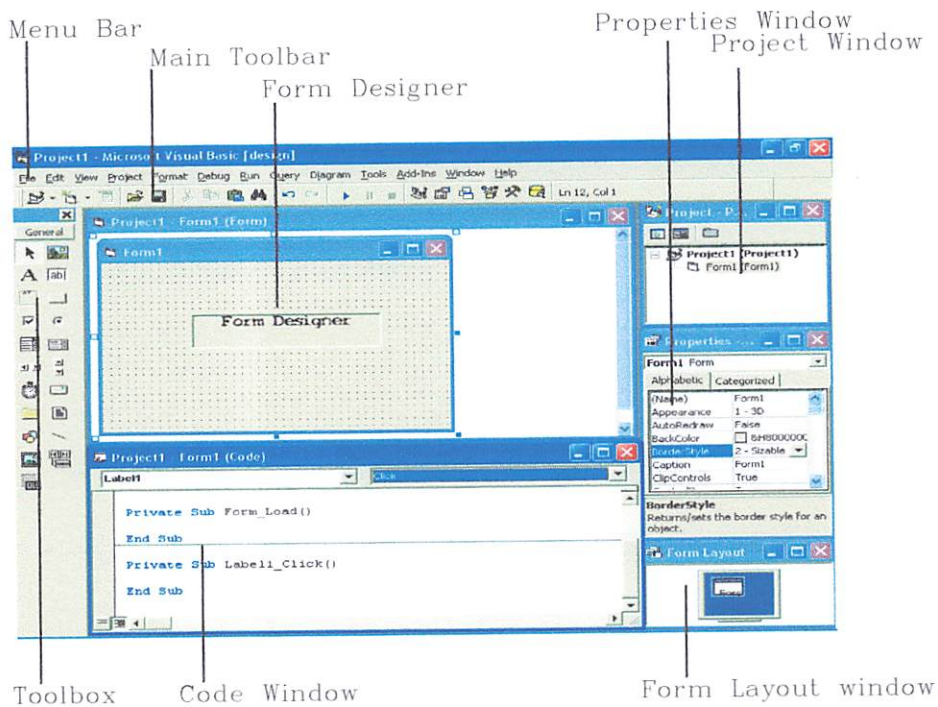
#### **1.6.6.1. IDE Visual Basic**

Langkah awal dari penggunaan pemrograman Visual Basic adalah mengenal IDE (*Integrated Development Environment*), Visual Basic yang merupakan Lingkungan Pengembangan Terpadu bagi programmer dalam mengembangkan aplikasinya. Dengan menggunakan IDE programmer dapat membuat *user interface*, melakukan koding, *testing* dan *debuging* serta mengkompilasi program menjadi *executable*. Penguasaan yang baik akan IDE akan sangat membantu programmer dalam mengefektifkan tugas-tugasnya sehingga dapat bekerja dengan efisien.<sup>~7</sup>

IDE Visual Basic 6 menggunakan model MDI (*Multiple Document Interface*). Berikut ini adalah gambar yang menunjukkan bagian-bagian dan nama-nama jendela yang ditampilkan pada IDE Visual Basic.

---

<sup>~7</sup> Belajar Komputer Visual Basic, Daryanto 2003, hal 13.



Gambar 1.15. IDE Visual Basic dengan jendela-jendela yang terbuka

Dengan menggunakan model MDI (*Multiple Document Interface*), IDE Visual Basic 6 dengan jendela terbuka hampir setiap waktu, maka akan membuat programmer senantiasa berinteraksi dengan *User Interface* tanpa menyadari bahwa dibelakangnya berjalan intruksi-intruksi program yang mendukung tampilan dan proses yang dilakukan.

Berikut adalah kelima jendela dilingkungan Visual Basic yang merupakan elemen utama dari pemrograman Visual Basic :

1. *Form*, merupakan *Area* kerja yang digunakan untuk merancang suatu program aplikasi Visual Basic. Pada jendela *form* ini dapat diletakkan kontrol seperti *command button*, *textbox*, *label*, *checkbox*, dan lainnya.



2. *Tollbox*, merupakan sebuah jendela dimana setiap alat-alat mempunyai fungsi atau kontrol dalam membentuk suatu program, dengan cara dipasang pada *form*.
3. *Project Explorer*, merupakan *area* yang berisi semua *file* program aplikasi Visual Basic. Suatu aplikasi Visual Basic disebut dengan *project* (proyek), dan setiap proyek terdiri dari satu atau lebih *file*.
4. *Properties*, Jendela ini menguraikan setiap elemen individual pada setiap aplikasi, dan bertugas menyiapkan segala *property* dari kontrol yang diperlukan dalam perancangan pemrograman.
5. *Code*, merupakan *area*, yang dapat digunakan untuk menuliskan kode-kode program Visual Basic. Suatu kode-kode program merupakan kumpulan dari intruksi untuk menjalankan obyek yang berupa kontrol, *form* serta logika program.

#### **1.6.6.2. Obyek Linking and Embedding (OLE)**

Kelebihan yang dimiliki oleh Visual Basic adalah fasilitas OLE (*Obyek Linking and Embedding*) yang memungkinkan untuk membuat suatu obyek dalam suatu aplikasi yang berisi data dari aplikasi lain, yaitu yang dapat ditempatkan didalam program Visual Basic.

1. **Embedding**, Obyek dari aplikasi lain yang dihubungkan dengan aplikasi Visual Basic, sewaktu pemakai keluar dari aplikasi, secara

otomatis obyek diperbaharui dan disimpan dalam aplikasi Visual Basic. Tidak ada aplikasi yang mempunyai akses data yang di-embed.

2. **Linking**, Obyek dari aplikasi lain yang dihubungkan dengan aplikasi Visual Basic, sewaktu pemakai keluar dari aplikasi, obyek yang diperbaharui disimpan dalam aplikasi sumbernya. Obyek yang sama dapat dihubungkan dengan beberapa aplikasi yang lain.

#### **1.6.6.3. Perangkat Lunak Map Object 2.1.**

Map Object 2.1. merupakan salah satu komponen yang digunakan untuk menyajikan peta, yang dikombinasikan dari object basis data dalam SIG. Aplikasi yang digunakan dispesifikasikan terhadap keperluan atau perintah yang dijalankan dalam pemrograman Visual Basic 6.0.

Dalam Map Object 2.1. didukung aktif oleh *Active X Data Object* [ADO] yang merupakan model basis data dari *Microsoft* dengan fungsi sebagai alat untuk menjalankan *Universal Data Access* [UDA]. Tujuan dari UDA agar pemrograman Visual Basic dapat mengikuti standart *OLEDB* dalam membuat suatu aplikasi. *OLEDB* adalah suatu standar basis data yang digunakan sebagai alat menghubungkan data dari berbagai sumber data, misalnya penyimpanan data dalam sebuah basis data, sebuah file teks, atau lembar kerja. Dalam Map Object dapat dibuat tabel object dengan menghubungkan data dari *Microsoft Acces 2000* menggunakan

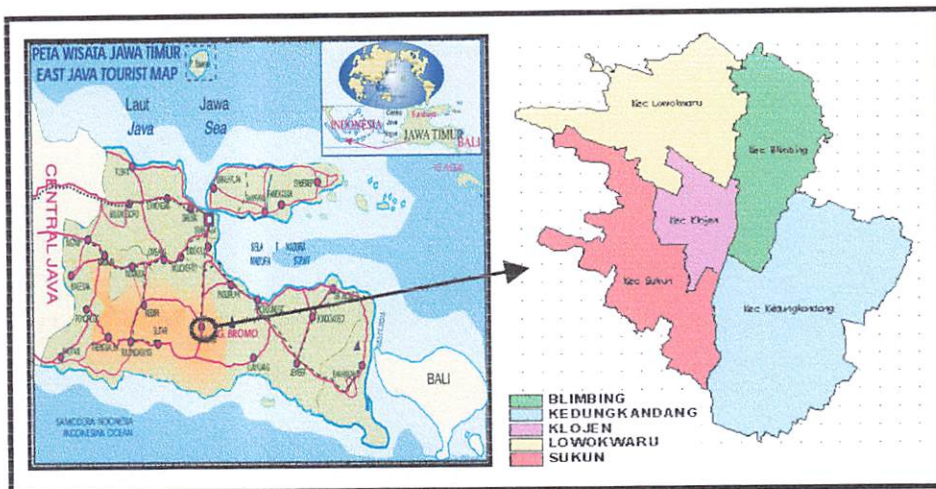
*Microsoft Jet 4.0 OLEDB Provider* untuk menggabungkan informasi ke dalam *Shapefile*. [ESRI, MapObject Online Reference, 2001].

## BAB II

### PELAKSANAAN PENELITIAN

#### 2.1. Deskripsi Lokasi Penelitian

Lokasi Penelitian berada di Propinsi Jawa Timur tepatnya dikota Malang, Kota Malang memiliki luas 124.456 Km. persegi, dihuni oleh 700.000 warganya. Kepadatan penduduk mencapai 5.000 - 12.000 jiwa per kilometer persegi dengan tingkat pertumbuhan 3.9 % per tahun. Tersebar di 5 Kecamatan ( Klojen, Blimbing, Kedungkandang, Sukun dan Lowokwaru ), 47 Kelurahan, 10 desa, 442 RW dan 3.208 RT berada pada ketinggian 399 - 662,5 m, terletak pada koordinat 112,34'09" - 11,41'34" BT 7,54'52", 22 - 8,03'05", 11 LS dengan Suhu rata-rata 23 - 25 derajat serta Curah Hujan mencapai 1.833 mm per tahun. Adapun tampilan lokasi penelitian seperti pada gambar 2.1



Gambar 2.1. Lokasi Penelitian

## **2.2. Bahan dan Alat Penelitian**

### **2.2.1. Bahan Penelitian**

Beberapa data yang diperlukan dalam penelitian adalah sebagai berikut :

#### **A. Data Spasial**

Peta Wilayah Administrasi Kota Malang Skala 1 : 5.000 tahun 2004, sistem proyeksi UTM.

#### **B. Data Non Spasial**

##### **1. Data Atribut Obyek Properti**

Meliputi data :

- Nama properti
- Lokasi properti
- Luas bangunan
- Luas tanah
- Status hak tanah / bangunan
- Harga properti
- Fasilitas properti

##### **2. Data Atribut Wilayah Administrasi**

##### **3. Data Atribut Fasilitas Umum**

##### **4. Foto-foto Kenampakan Obyek**

### **2.2.2. Peralatan Penelitian**

#### **A. Perangkat Lunak ( *software* )**

1. AutoCAD 2000

Digunakan dalam penggambaran grafis, atau digunakan untuk mengubah data analog menjadi data digital dengan cara digitasi

2. Arc info 3.5

Digunakan input data, digitasi, editing, menyimpan dan menyajikan informasi ( mengolah data spasial ).

3. Arc View 3.1

Digunakan untuk menggabungkan data dan mengvisualisasikan basisdata non spasial

4. Microsoft Acces 2000

Digunakan untuk penyusunan data atribut dalam basis data (*Database*).

5. Map Object 2.1.

Digunakan untuk menyajikan peta, dispesifikasikan terhadap perintah yang dijalankan dalam Visual Basic.

6. Visual Basic 6.0.

Digunakan untuk pembuatan program visualisasi gambar dari obyek transaksi, mulai dari lokasi secara geografisnya, data dan informasi, sampai foto obyeknya.

**B. Perangkat Keras ( *hardware* )**

1. Satu set Personal Computer Intel Pentium 4 dengan kecepatan 256 MB.
2. Digitizer
3. Meja Digitizer Kurta XLC 36 x 48
4. Plotter / printer

### **2.3. Pelaksanaan Penelitian**

Pelaksanaan penelitian ini akan dilakukan dengan menggunakan langkah-langkah sebagai berikut :

#### **1. Persiapan Penelitian**

Melakukan persiapan semua data-data spasial dan data non spasial sesuai dengan kebutuhan penelitian, dan juga melakukan persiapan terhadap perangkat keras dan perangkat lunak untuk pengoperasiannya.

#### **2. Pemasukan dan Pengelompokan Data**

Pemasukan dan pengelompokan meliputi pemasukan data spasial dan data non spasial, data-data spasial dibuat dalam bentuk digital dengan cara merubah peta dengan menggunakan digitizer dan AutoCad 2000 untuk mengolahnya termasuk mendigit data grafisnya. Setelah data dalam bentuk digital kemudian diexport ke perangkat lunak Arc Info ( pembentukan layer/tabel ), dan dilakukan pengeditan kembali dengan ArcEdit hingga semua data

yang disajikan kepada *user* telah bebas dari kesalahan. Untuk data non spasial dilakukan pemilihan dan pengelompokan terlebih dahulu sesuai dengan kebutuhan sebelum dimasukkan ke DBMS.

### 3. Penggabungan Data

Penggabungan data ini dimaksudkan untuk mengelompokkan data spasial dan data non spasial yang telah disusun dengan kebutuhan dan merupakan suatu informasi terpadu dalam suatu sistem, sehingga dapat dilakukan suatu analisa terhadap data-data tersebut, yaitu data spasial dan data non spasial yang digabungkan.

### 4. Pembuatan Program

Pembuatan program untuk visualisasi gambar dari obyek transaksi, mulai dari lokasi secara geografisnya, data dan informasi, sampai foto obyeknya dengan menggunakan Visual Basic.

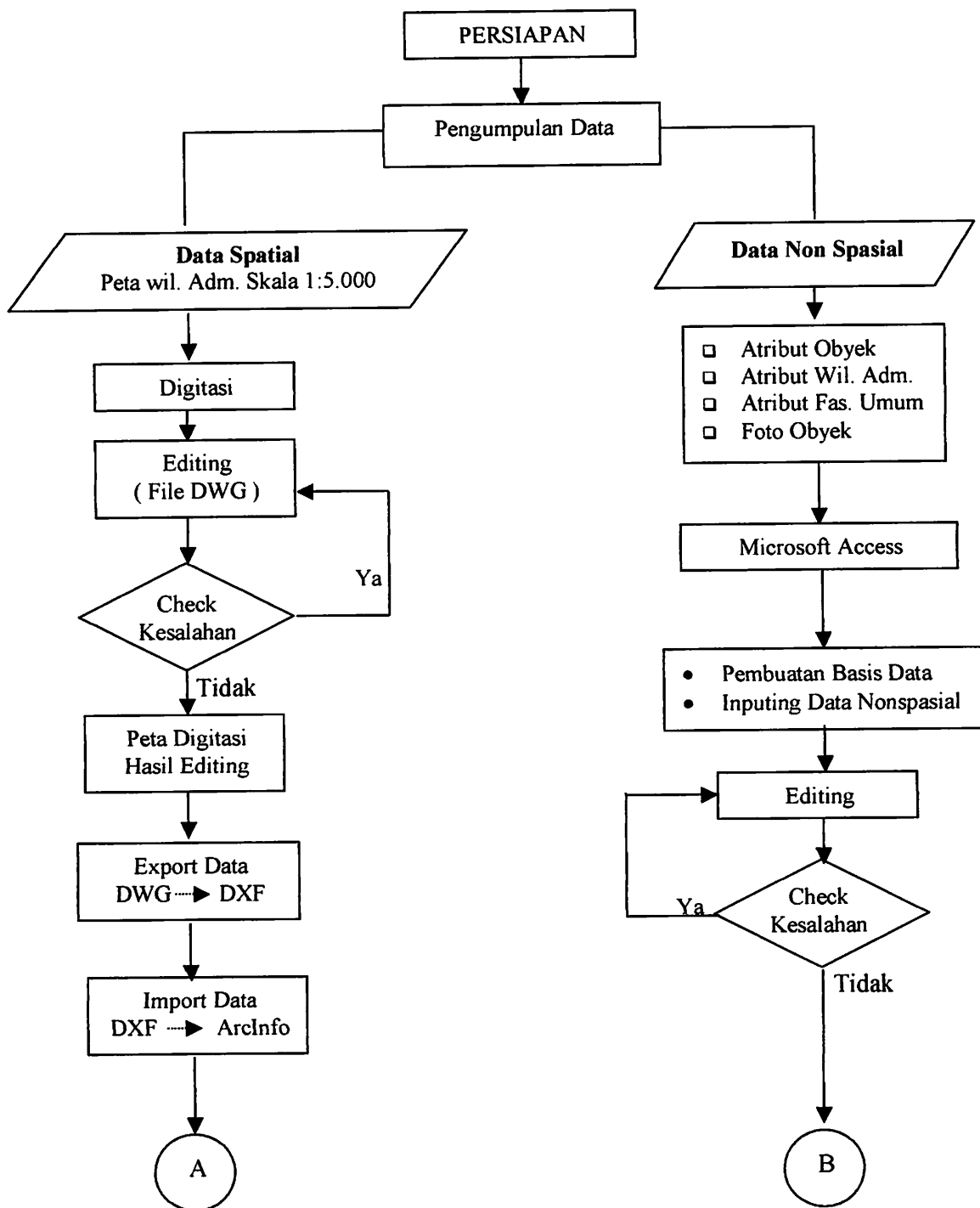
### 5. Penyajian Hasil

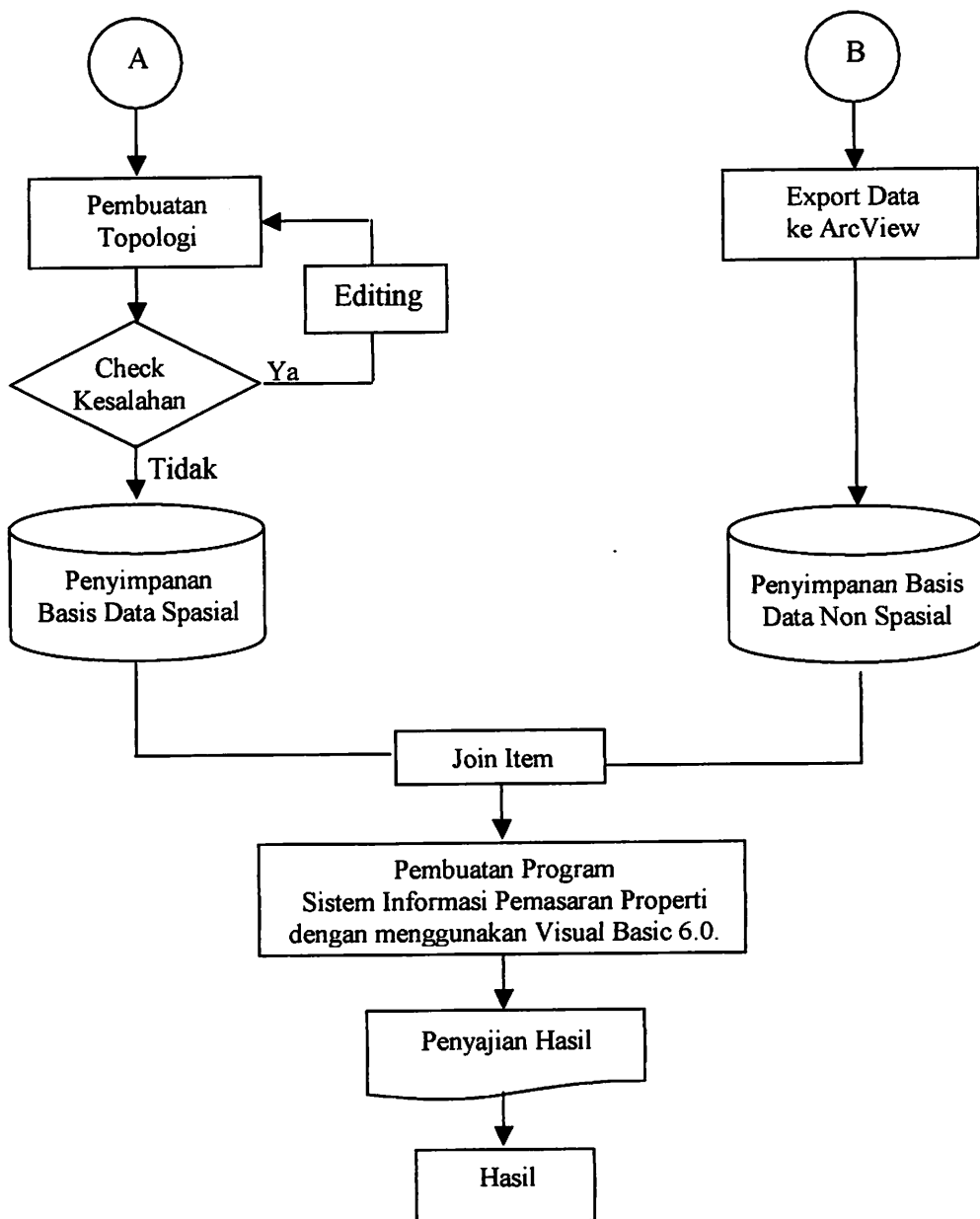
Penyajian hasil bertujuan untuk menampilkan keseluruhan hasil yang telah selesai dikerjakan.

Untuk keterangan lebih lanjut dan lebih rinci terdapat pada diagram alir (*flowchart*) penelitian sebagai berikut :



### Bagan Alir Pelaksanaan Penelitian



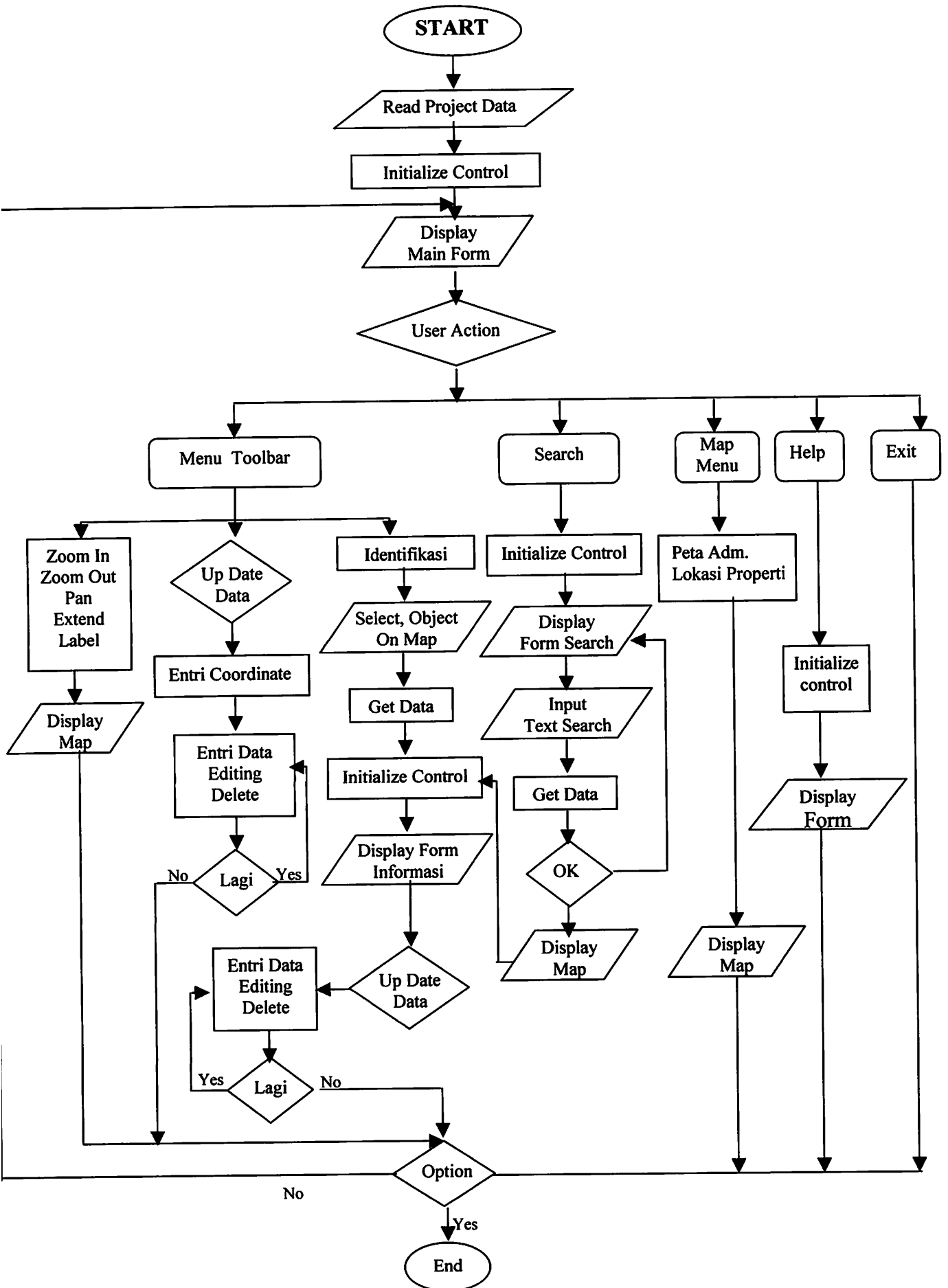


Gambar 2.2. Diagram Alur Pelaksanaan Penelitian

Pengertian istilah-istilah dalam bagan alir penelitian :

1. **Persiapan dan pengumpulan data** yaitu mempersiapkan peralatan penelitian dan personil serta data-data yang akan digunakan baik data atribut maupun data spasial, dan dikelompokkan menurut jenisnya.

2. **Penyusunan basisdata** yaitu menyusun basisdata menurut jenis dan macamnya dengan cara membuat tabel, sehingga mempermudah membuat hubungan antar atribut dengan data spasial atau atribut dengan atribut lainnya.
3. **Digitasi** yaitu proses merubah peta (data spasial) analog menjadi peta digital. Proses pendigitasian ini menggunakan bantuan meja digitizer.
4. **Editing** yaitu proses perbaikan data spasial ataupun data atribut dengan perangkat lunak tertentu.
5. **Checking** adalah memeriksa kelengkapan dan kebenaran data, jika terjadi kesalahan atau kekurangan data dilakukan proses ulang.
6. **Export data** adalah proses merubah *extension file* dari Auto Cad (dalam bentuk DWG) ke ArcInfo (dalam bentuk DXF)
7. **Join Item** yaitu proses penggabungan antara data atribut dengan data spasial.
8. **Pembuatan program**, dilakukan pembuatan program dengan menggunakan Visual Basic 6.0. untuk menghasilkan alat bantu proses visualisasi. Mulai dari menampilkan data dan informasinya serta menampilkan gambar-gambar atau foto-foto obyek secara visual.
9. **Penyajian hasil**, berupa visualisasi hasil, tabel dan laporan.



Gambar 2.3. Diagram Alur Pembuatan Program

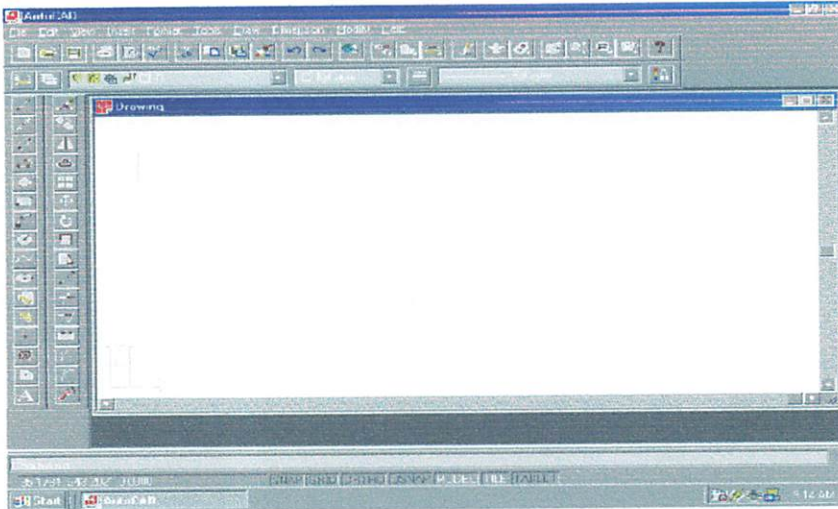
### **2.3.1. Persiapan Pelaksanaan Penelitian**

Pada persiapan ini meliputi peralatan baik perangkat lunak maupun perangkat keras sebagai sarana untuk tercapainya dari tujuan penelitian. Selain itu juga disusun daftar data-data apa saja yang diperlukan dari penelitian, kemudian melakukan survei lapangan. Adapun kegiatan persiapan yang dilakukan sebelum memulai penelitian, yaitu :

1. Menyiapkan perangkat keras (*hardware*) dengan kapasitas yang disesuaikan dengan kebutuhan penelitian.
2. Menginstal perangkat lunak (*software*) yang dibutuhkan seperti ArcInfo, ArcView, MsAcces dan software lainnya kedalam komputer.
3. Mengumpulkan data-data yang diperlukan sebagai bahan penelitian berupa peta-peta seperti peta administrasi, peta jaringan jalan dan selain berupa peta bisa juga berupa tabel-tabel. Data-data ini kemudian dikelompokkan menjadi data spasial dan data non spasial (data atribut).
4. Setelah semua peralatan dan data lengkap, maka penelitian telah siap untuk dikerjakan.

### **2.3.2. Pemasukkan Data Spasial**

Pemasukan data spasial adalah merubah dari analog menjadi digital dengan cara mendigitasi peta tersebut yang menggunakan software AutoCAD R14. Adapun tahapan-tahapan pendigitasian, yaitu :



Gambar 2.4. Tampilan software AutoCAD R14

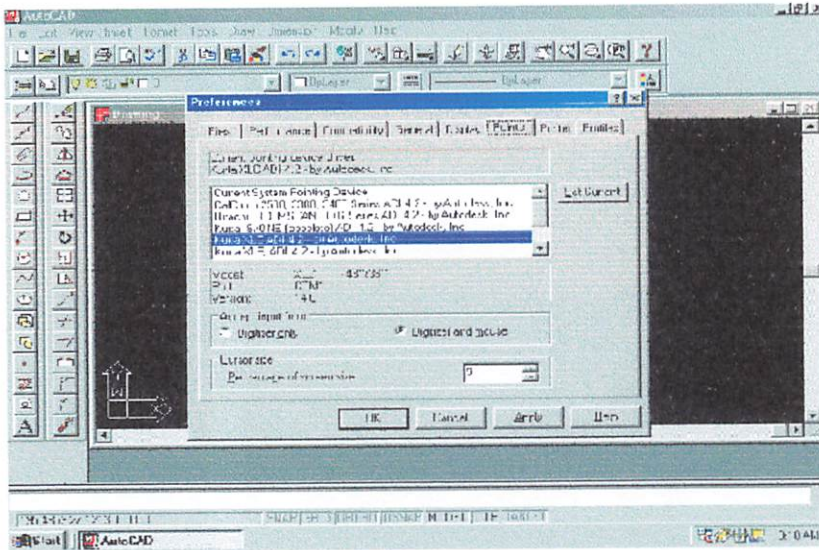
### A. Konfigurasi Peralatan

Setelah peralatan digitasi disiapkan, langkah awal yang dilakukan adalah dengan pengaturan konfigurasi alat digitasi yang digunakan. Tahapan pelaksanaannya sebagai berikut :

1. Mengaktifkan program AutoCAD R14
2. Selanjutnya mengatur konfigurasi peralatan digitasi dengan mengetikkan perintah Config pada Command line :

*Command : Config ( tekan enter )*

Maka pada layar monitor akan tampil konfigurasi menu :



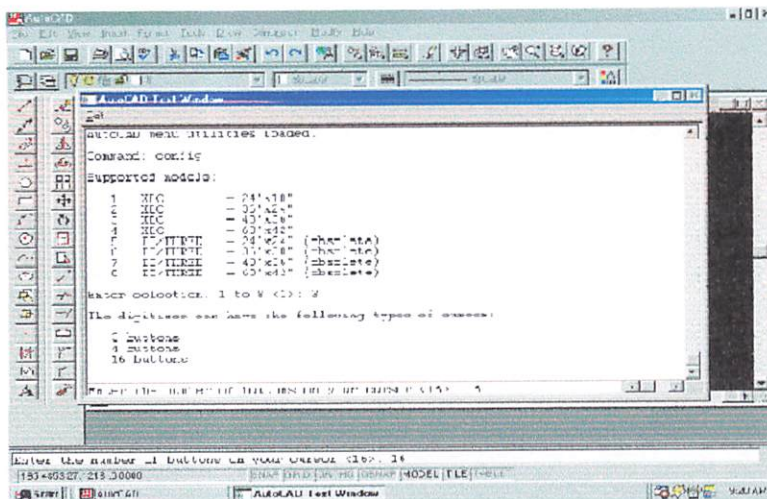
Gambar 2.5. Tampilan Konfigurasi Menu Digital

Klik menu *pointer*

Kemudian pilih *Kurta XLC ADI 4.2 Autodesk, Inc*

selanjutnya klik menu *Set Current*

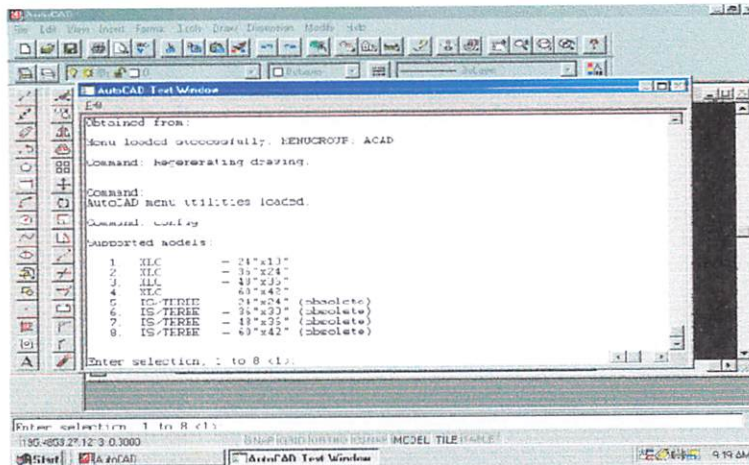
Maka pada layar monitor akan tampil



Gambar 2.6. Tampilan Konfigurasi Pemilihan Jenis Digitizer

Ketik angka 3 pada perintah *enter selection, 1 to 1 : 3 ( enter )*

Maka akan tampil :



Gambar 3.7. Tampilan konfigurasi pemilihan jumlah tombol mouse digitizer

Mengetikkan angka 16 pada perintah *enter the number of buttons on your cursor [16] ( tekan enter )*

*Enter serial port name for digitizer or, for none <COM1 : COM1 ( tekan enter )*.

Kemudian tutup kotak dialog konfigurasi dengan mengklik tanda silang pojok kanan atasnya.

## B. Proses Kalibrasi

Prosedur pelaksanaan kalibrasi adalah sebagai berikut :

Menyiapkan atau memasang peta pada meja digitizer, dengan posisi baik dan kuat agar tidak bergeser saat pendigitasian.

*Command : Tablet ( tekan enter )*

*Option (ON/OFF/CAL) : CAL ( tekan enter )*



*Digitize point # 1* : Menempatkan benang silang digitizer pada pojok kiri bawah dari bingkai peta lalu tekan tombol 1 pada digitizer, maka akan muncul perintah dilayar seperti berikut ini :

*Enter coordinates for point first* : masukkan koordinat ( tekan enter )

*Digitize point # 2 (or return to end)* : menempatkan benang silang digitizer pada pojok kiri atas lalu tekan tombol 1 pada digitizer, maka akan muncul perintah dilayar seperti berikut ini :

*Enter coordinates for point # 2* : masukkan koordinat ( tekan enter )

*Digitize point # 3 (or return to end)* : Menempatkan benang silang digitizer pada pojok kiri bawah dari bingkai peta lalu tekan tombol 1 pada digitizer, maka akan muncul perintah dilayar seperti berikut ini :

*Enter coordinates for point # 3* : masukkan koordinat ( tekan enter )

*Digitize point # 4 (or return to end)* : Menempatkan benang silang digitizer pada pojok kiri bawah dari bingkai peta lalu tekan tombol 1 pada digitizer, maka akan muncul perintah dilayar seperti berikut ini :

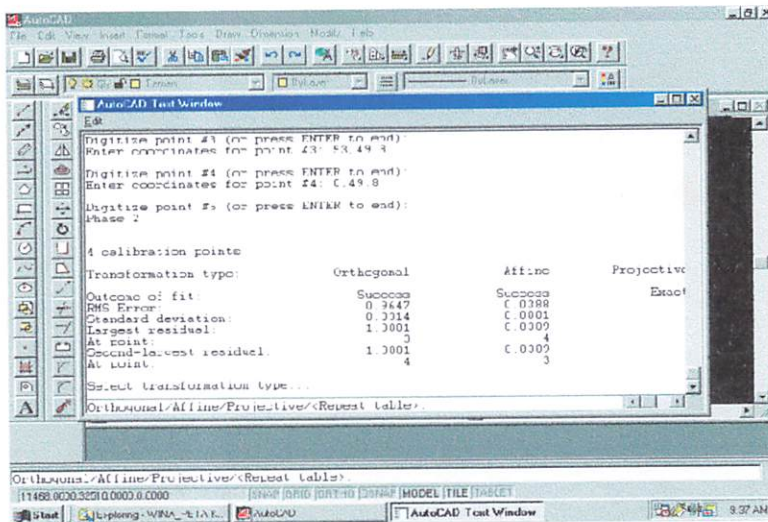
*Enter coordinates for point # 4* : masukkan koordinat ( tekan enter )

*Digitize point # 5 (or return to end)* : (tekan enter)

Pada layar monitor akan muncul hasil dari kalibrasi tersebut

4 calibration points			
Transformation	: Orthogonal	Affine	Projektive
Outcome Of Fit	: Succes	Success	Exact
RMS Error	: 0.9647	0.0013	
Standart Deviation	: 0.0314	0.0000	
Largest Residual	: 0.0001	0.0023	
At point	: 3	4	
Secent-Largest Residual	: 0.0153	0.0023	
At Point	: 4	3	
Select tranformation type .....			
Orthogonal ? Affine/projective/<repeat table>	: A		

Kemudian tutup kotak tablet dengan mengklik tanda silang pada pojok kanan atasnya. Adapun tampilan kalibrasi alat pada digitizer.



Gambar 2.8. Tampilan Proses Kalibrasi pada Digitizer

### C. Pelaksanaan Digitasi

Adapun langkah-langkah pendigitasian adalah sebagai berikut :

1. Membuat bingkai dari peta yang akan digitasi, dengan mengetikkan *rectang* pada perintah command.

*Command* : *Rectang* ( tekan enter )

*Chamfer/elevation/fillet/Thickness/Width/< First corner >* :

Menempatkan benang silang digitizer pada pojok kiri bawah bingkai peta yang akan digitasi lalu menekan tombol 1 pada digitizer.

*Other Corner* : Menempatkan benang silang digitizer pada pojok kanan atas bingkai peta yang akan di digitasi lalu tekan tombol 1 pada digitizer.

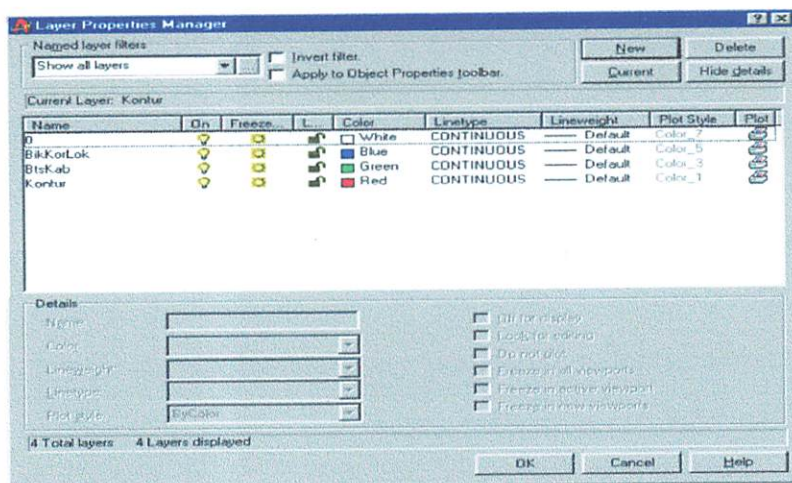
2. Membuat layer untuk masing-masing unsur yang akan di digitasi :

Memilih menu format, lalu klik layer. Kemudian dilayar akan muncul kotak dialog layer.

Klik NEW, kemudian mengetik nama layer yang akan dibuat.

Untuk layer yang lain dilakukan dengan cara yang sama.

- Melakukan pendigitasian, contohnya untuk garis menggunakan perintah polyline ( PL ).
- Setelah seluruh objek digitasi, disimpan dengan memilih menu file. Kemudian klik Save, maka akan muncul kotak dialog Save.
- Mengetik nama file yang dikehendaki, kemudian klik Save
- Maka data akan tersimpan pada file yang dikehendaki tersebut
- Menutup program AutoCAD.



Gambar 2.9. Tampilan Cara Membuat Layer


### 2.3.3. Editing di AutoCAD

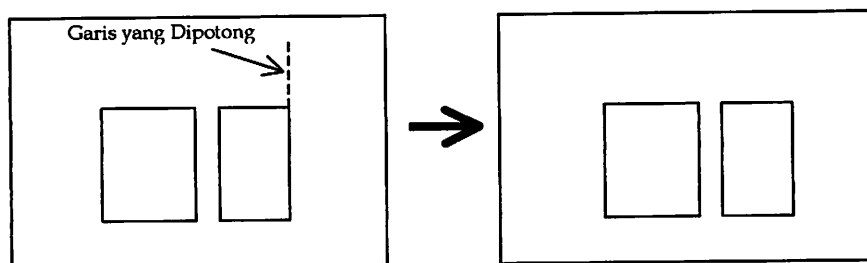
Dalam hal ini editing dilakukan adalah untuk memperbaiki atau menyempurnakan hasil digitasi dengan menggunakan perintah-perintah editing yang tersedia pada program AutoCAD.

Dalam pelaksanaannya perintah-perintah yang sering dipergunakan antara lain seperti : *Trim, Extend dan Pedit*

#### A. Penggunaan perintah TRIM

Perintah Trim digunakan untuk memotong akhir objek atau garis sehingga berakhir pada batas yang telah ditentukan.


1. Memilih perintah Trim pada menu yang telah tersedia diprogram AutoCAD atau klik yang berlambang 
2. Mengklik garis batasnya sebagai pemotong
3. Lalu diklik garis yang akan dipotong tersebut
4. Maka garis yang akan dipotong akan terpotong tepat pada batas garis pemotongan, seperti pada gambar 2.10.



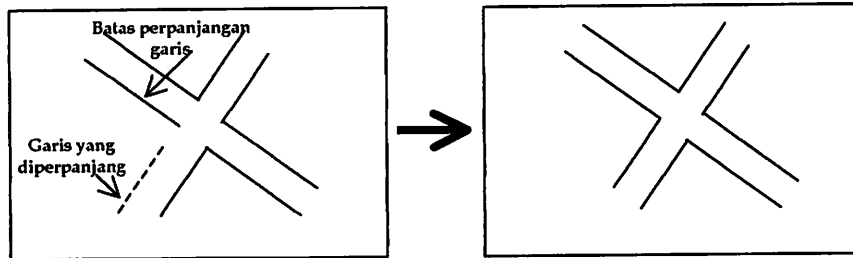
Gambar 2.10. Tampilan penggunaan perintah TRIM

#### B. Penggunaan perintah Extend

Perintah Extend digunakan untuk menyambung objek atau garis yang kurang pada batas yang telah ditentukan.

1. Memilih perintah Extend pada menu yang telah tersedia di program AutoCAD atau klik yang berlambang 
2. Mengklik garis pada batas perpanjangan.

3. Lalu diklik garis yang akan diperpanjang tersebut.
4. Maka garis yang akan diperpanjang akan sampai pada garis batas perpanjangan.

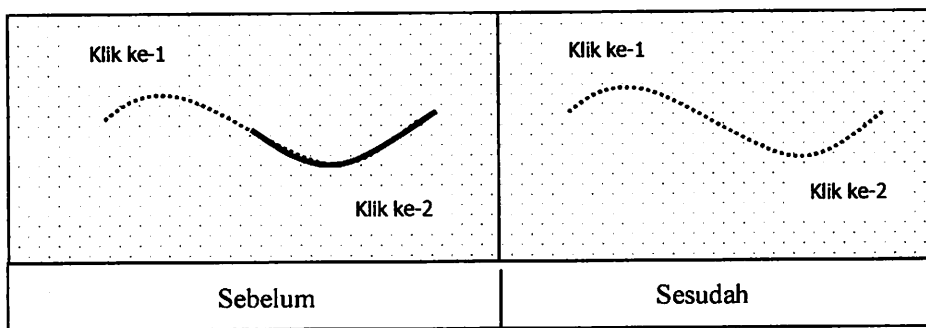


Gambar 2.11. Tampilan penggunaan perintah Extend

### C. Penggunaan perintah Pedit

Perintah Pedit digunakan untuk menjadikan beberapa garis menjadi satu kesatuan utuh.


1. Mengetikkan perintah Pedit
2. Memilih garis sebagai penyambung lalu diklik, layer akan tampil :  
*Close/Join/EditVertex/Fit/Spline/Decurve/Ltype/Undo/Exit <X> :J ( tekan enter )*
3. Klik Garis yang akan dijadikan satu kesatuan kemudian tekan enter, maka garis tersebut akan menjadi satu kesatuan.

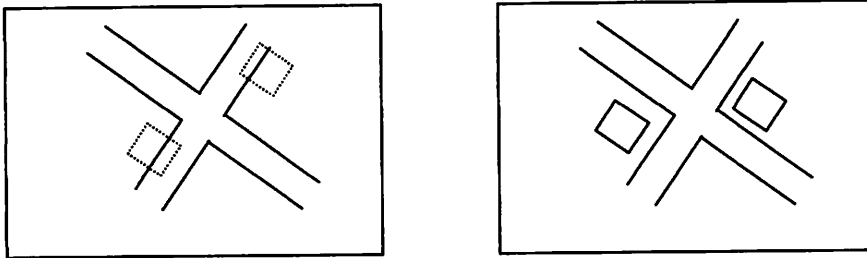


Gambar 2.12. Tampilan penggunaan perintah Pedit (joint)

#### D. Penggunaan Perintah Move

Untuk memindahkan posisi suatu obyek dapat dijalankan dengan menggunakan perintah Move. Perintah Move dapat diketikan langsung pada *Command Line* atau pilih menu *Modify*, klik sub menu move


1. Ketikan perintah Move pada *command* atau klik ikon berlambang 
2. Pilih obyek yang akan dipindah dengan cara mengeklik obyek tersebut
3. Pilih obyek yang lain yang dipindah kemudian tekan Enter
4. Klik bagian obyek yang dijadikan titik basis
5. Kemudian memindah obyek ditempat yang diinginkan.

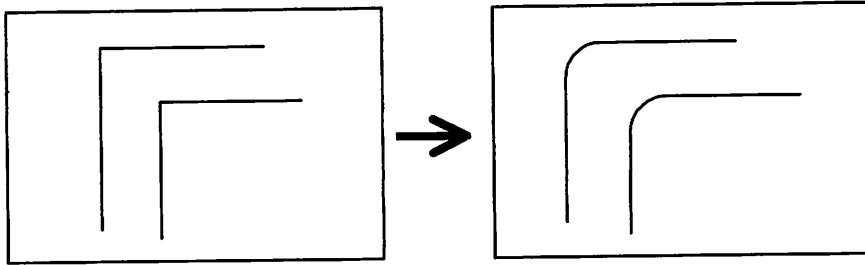


Gambar 2.13. Tampilan penggunaan perintah Move

#### E. Penggunaan perintah Fillet

Perintah Fillet digunakan untuk memperhalus pertemuan antara garis pada radius tertentu.

1. Ketikan perintah Fillet pada *command* atau klik ikon berlambang 
2. Pilih obyek pertama yang akan diperhalus
3. Pilih obyek kedua yang diperhalus
4. Secara otomatis obyek pertemuan garis akan diperhalus



Gambar 2.14. Tampilan penggunaan perintah Fillet

Data dari hasil editing disimpan dengan menekan CTRL+S atau pilih menu File, kemudian klik Save, maka data akan disimpan dengan format (extension) DWG.


#### **2.3.4. Export Data ( Format .DXF )**

Data yang disimpan dalam format .DWG tersebut harus dirubah dalam bentuk format .DXF. Adapun caranya yaitu dengan meng-*export* data tersebut, sedangkan langkah-langkahnya adalah sebagai berikut :

1. Memilih menu File
2. Klik Export (jika menggunakan AutoCAD R14) atau Klik Save as (jika menggunakan AutoCAD 2000).
3. Akan muncul kotak dialog Export, lalu pilih extension DXF.
4. Ketikan nama filenya.
5. Klik Save

### 2.3.5. Import Data

Data yang sudah di export dalam format DXF kemudian di *import* kedalam ArcInfo, pelaksanaannya adalah sebagai berikut :

1. Membuka program ArcInfo 3.5 for DOS dengan terlebih dahulu komputer masuk pada program MS DOS. 

2. Mengetikkan perintah ARC pada *prompt* F (program ArcInfo berada)

F:\>ARC [enter]

3. Setelah muncul logo ArcInfo dan sudah berada dalam lingkungan program , kemudian mengeditkan perintah untuk import data :

```
KEL.SBX      [SHP]  KELURA~1.DXF
7 file(s)      590,313  bytes
3 dir(s)    6,756,532,224  bytes free
```

```
(F:\Abdullah\Spasial)[ARC]dxdfarc kelura~1 klrhn
[PC ARC/INFO 3.5 DXFARC - 09/12/04]
```

```
Enter layer names and options (type END or $REST when done)
```

```
=====
```

```
Enter the 1 st layer and options: kelurahan (Enter)
```

```
Enter the 2nd layer and options:
```

```
Character string expected
```

```
Done entering layer names and option (Y/N)? Y
```

```
Do you wish to use the above layers and options (Y/N)? Y (Enter)
```

```
Processing KELURA~1.DXF...
```

```
No labels, killing XCODE...
```

```
16 Arcs written.
```

```
0 Labels written.
```

```
0 Annotations written.
```

```
0 Annotations levels.
```

```
(F:\Abdullah\Spasial)[ARC]
```



### **2.3.6. Editing dengan ArcInfo**

Editing merupakan tahap yang sangat penting hal ini dilakukan untuk memperbaiki kekurangan data atau penghapusan data. Proses editing ini dilakukan di ArcInfo yang meliputi :

- a. Menghilangkan Undershoot dan Overshoot
- b. Menghilangkan Sliver
- c. Pemberian label (ID)
- d. Memperbesar dan memperkecil feature

Adapun langkah-langkah editingnya adalah sebagai berikut :

1. Menuliskan perintah arcedit

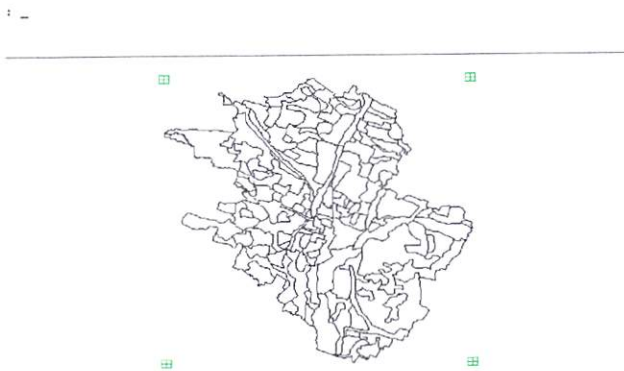
```
(F:\Abdullah\Spasial)[ARC]arcedit  
[PC ARC/INFO 3.5 ARCEDIT - 09/12/04]  
Serial Communications Driver – Version 5.0  
COM1 (IRQ04 Level – I/O Port 3F8)
```

```
ARCEDIT Ver 3.5.1  
Copyright © 1996 by  
Environmental Systems Research Institute  
380 New York Street  
Redlands, CA 92373  
All Rights Reserved Worldwide
```

```
: disp 4  
(F:\)[ARC] ARCEDIT (Enter)
```

2. Memanggil coverage yang akan diedit :

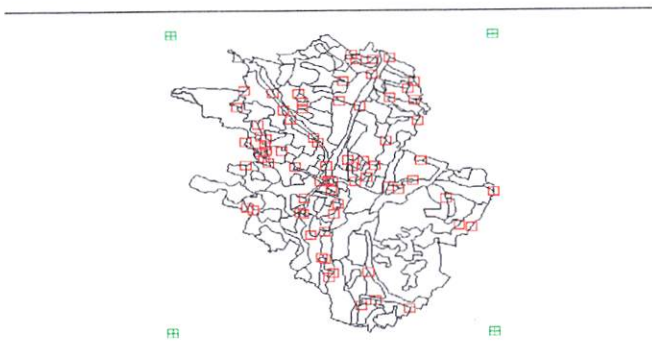
```
: editcov klrhn  
The edit coverage is now F:\Abdullah\Spasial\klrhn  
The Map extent is not defined  
Defaulting the map extent to BND of  
F:\Abdullah\Spasial\klrhn  
: drawn all  
: draw
```



Gambar 2.15. Tampilan coverage yang akan diedit

3. Mendeteksi dan menampilkan kesalahan pada coverage

: *drawn node dangle*  
 : *draw (Enter)*  
 : \_



Gambar 2.16. Tampilan kesalahan pada coverage

4. Memperbaiki kesalahan pada coverage

A. Overshoot ( menghilangkan kelebihan garis )

: *ef arc (Enter)*  
 : *select box (Enter)*  
 : *delete (Enter)*  
 : *draw (Enter)*

B. Undershoot ( menyambung garis atau memindahkan node ke node lain)

: *ef node (Enter)*  
 : *move (Enter)*

C. Memilih node yang akan dipindahkan lalu klik ke node tujuan kemudian

tekan angka 2

: *draw (Enter)*

5. Pemberian *User-ID* atau nilai label

: *ef label (Enter)*

: *add (Enter)*

Pilih dan tekan angka 8 lalu 1 untuk mengisikan ID atau label yang

diinginkan kemudian klik di polygon mana ID tersebut berada pilih

dan tekan angka 9 untuk mengakhiri perintah tersebut.

6. Merubah nilai label

: *ef label (Enter)*

: *select (Enter)*

: *calculate <nama cover\_id> = nilai yang benar (Enter)*

: *draw (Enter)*

7. Menghapus nilai label yang lebih dari satu

: *ef label (Enter)*

: *select many (Enter)*

: *delete (Enter)*

: *draw (Enter)*

8. Setelah semua *editing* selesai, dilanjutkan dengan menyimpan hasil

*editing* tersebut dan keluar dari *Arccedit* dengan mengetikkan perintah

: *Quit (Enter)*

### **2.3.7. Pembuatan Topologi**

Topologi merupakan hubungan eksplisit (hubungan spasial) diantara *feature* geospasial (*polygon, arc, point*) yang digunakan untuk

mempresentasikan keterkaitan antara *feature* yang terdapat dalam suatu *coverage* (peta), meliputi *connectivity*, *contiguity*, dan *definisi area* (tata letak, batas, luasan). [Sunaryo, 2000].

Peta atau *coverarage* yang telah dibuat topologinya akan terbentuk tabel, dimana tabel tersebut menyimpan atribut standart yang menerangkan seluruh elemen / *feature* dari *coverage* secara geomatik.

Setelah editing *coverage* selesai dan sudah disimpan, maka selanjutnya keluar dari Arcedit untuk membuat topologi dari hasil editing tersebut . Membuat topologi dapat dilakukan dengan dua cara, yaitu dengan perintah CLEAN dan BUILD. Adapun langkah pembuatan topologi adalah sebagai berikut :

(F:/)[ARC]CLEAN Nama\_file(Poly/Line) (Enter)

atau dengan perintah :

(F:/)[ARC]BUILD Nama\_file(Poly/Line) (Enter)

```
(F:\Abdullah\Spasial)[ARC]build klhm
[PC ARC/INFO 3.5 BUILD - 09/12/04]
Building polygons...
  Sorting input file...
  Sorting label file...
  Processing...
  Assigning final lds...
  Writing arc file...
  Generating polygon report...
  Creating attribute file for klrhn.
  Sorting User-lds...
  Marging record    6
```

(F:\Abdullah\Spasial)[ARC]

(F:\Abdullah\Spasial)[ARC]clean klhm  
 [PC ARC/INFO 3.5 CLEAN - 09/12/04]  
 Cleaning klrhn.  
 Sorting...

CLNSRT Ver 3.5.1  
 Copyright © 1996 by  
 Environmental Systems Research Institute  
 380 New York Street  
 Redlands, CA 92373  
 All Rights Reserved Worldwide

Intersecting...  
 Assembling Polygons...  
 Sorting input file...  
 Sorting label file...  
 Processing...  
 Assigning final Ids...  
 Writing arc file...  
 Generating polygon report...  
 Creating PAT...  
 Sorting User-Ids...  
 Marging record 6

(F:\Abdullah\Spasial)[ARC]

### 2.3.8. Pengkodean Data Spasial

Setiap *coverage* yang telah dibuat topologinya akan memiliki tabel dengan item-item standart dengan urutan sebagai berikut :

↳ Untuk *feature* titik :

Item	Keterangan Item
AREA	Informasi luas dari setiap poligon dalam satuan <i>coverage</i> selalu bernilai nol (0).
PERIMETER	Informasi panjang setiap batas poligon dalam satuan <i>coverage</i> selalu bernilai nol (0).
Cover_	Informasi nomor poligon atau titik internal (ditentukan program <i>ArcInfo</i> )
Cover_ID	Informasi penggunaan ID setiap poligon atau titik (ditentukan pemakai)

Tabel 2.1. Pengkodean Data Spasial [Feature Titik]

↳ Untuk *feature* poligon :

Item	Keterangan Item
AREA	Informasi luas dari setiap poligon dalam satuan <i>coverage</i>
PERIMETER	Informasi panjang setiap batas poligon dalam satuan <i>coverage</i>
Cover_	Informasi nomor poligon atau titik internal (ditentukan program <i>ArcInfo</i> )
Cover_ID	Informasi penggunaan ID setiap poligon atau titik (ditentukan pemakai)

Tabel 2.2. Pengkodean Data Spasial [Feature Poligon]

↳ Untuk *feature* Garis :

Item	Keterangan Item
FNODE	Informasi nomor <i>node</i> dari setiap <i>feature</i> garis yang dimulai dari posisi <i>node</i> ke...
TNODE	Informasi nomor <i>node</i> dari setiap <i>feature</i> garis yang diakhiri oleh posisi <i>node</i> ke...
LPOLY	Informasi nomor posisi <i>polygon</i> kiri terhadap posisi setiap garis yang dibatasi oleh TNODE ke... dan FNODE ke..
RPOLY	Informasi nomor <i>polygon</i> kanan terhadap posisi setiap garis yang dibatasi oleh TNODE ke... dan FNODE ke..
LENGTH	Panjang setiap garis yang dibatasi oleh TNODE ke.. dan FNODE ke.. dalam satuan <i>coverage</i>
COVER_	informasi nomor garis internal (ditentukan program <i>ArcInfo</i> )
COVER_ID	Informasi penggunaan ID setiap garis (ditentukan pemakai)

Tabel 2.3. Pengkodean Data Spasial [Feature Garis]

Pemberian *identifier* (ID) pada setiap *feature* oleh pemakai merupakan tahap pengkodean secara unik pada setiap elemen peta (poligon, garis, titik). Pemberian ID ini dilakukan dalam sistem *Arcedit* dengan perangkat lunak *ArcInfo*. [Sunaryo, 2000]. Pada *coverage* poligon

dan titik, setiap *feature* harus diberi *label* terlebih dahulu, selanjutnya pemberian ID dapat dilakukan untuk memberi identitas unik pada setiap *feature* titik dan atau poligon. Identitas unik tersebut akan tersimpan dalam tabel atribut standar yang dimiliki suatu *coverage*. Tabel tersebut memiliki extension *PAT*.

Pada *coverage* garis setiap *feature* dapat langsung di-*select*, selanjutnya langsung diberi ID / identitas unik pada setiap *feature* garis yang ada dalam *coverage*. Tabel atribut standart *feature* garis secara otomatis akan menyimpan ID tersebut. Dalam *ArclInfo*, tabel tersebut memiliki extension *AAT*. ID ini selanjutnya dapat digunakan untuk menghubungkan setiap *feature* di dalam *coverage* dengan atribut baru yang akan di tentukan oleh pemakai.

### 2.3.9 Pembuatan Basis Data

Dalam pembuatan basis data untuk inventarisasi obyek properti yang akan dipasarkan harus ditentukan entitas terlebih dahulu. Adapun entitas dalam penelitian ini dapat dilihat pada tabel 2.4.

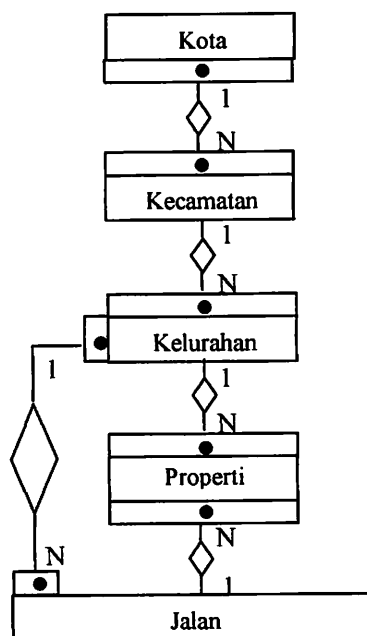
No	Nama Entitas
1	Kota
2	Kecamatan
3	Kelurahan
4	Jalan
5	Properti

Tabel 2.4 Tabel Entitas

Sesuai dengan hubungan antar entitas yang diperoleh, selanjutnya ditentukan derajat keanggotaan dan kelas masing-masing hubungan entitas. Berikut *Enterprise Rule* dari ER diagram Lokasi Properti adalah :

- Sebuah Kota terdiri paling sedikit satu Kecamatan, Satu Kecamatan harus terletak dalam sebuah Kota.
- Satu Kecamatan terdiri paling sedikit satu Kelurahan, satu Kelurahan harus terletak pada satu Kecamatan.
- Satu Kelurahan paling sedikit mempunyai sebuah Properti, sebuah Properti harus terletak pada satu Kelurahan.
- Satu Kelurahan mempunyai paling sedikit sebuah Jalan, sebuah Jalan harus berada pada satu Kelurahan.
- Satu Jalan mungkin terdapat sebuah Properti, sebuah Properti terletak pada satu Jalan.

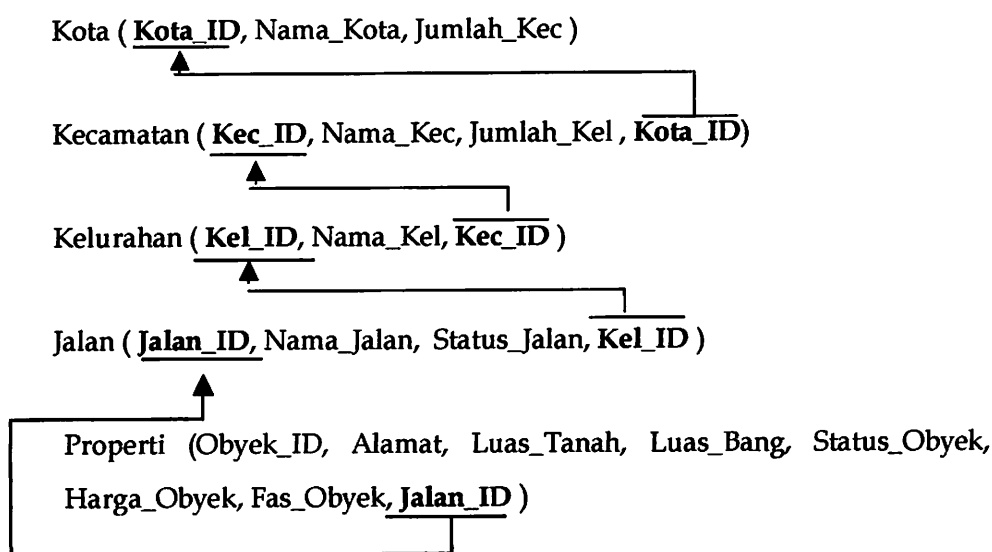
Model ER diagram Properti dapat digambarkan seperti berikut :



Gambar 2.17. Hubungan antar Entitas



Setelah hubungan antar entitas telah diketahui, maka dalam basis data relational disimpan dalam tabel-tabel saling berkaitan. Hubungan antar tabel ditandai dengan nilai data-data yang sama untuk dijadikan elemen kunci atau identitas tabel entiti. Skema kerangka tabel basis data relational untuk inventarisasi properti disajikan pada gambar 2.18.




Gambar 2.18. Skema kerangka tabel entiti

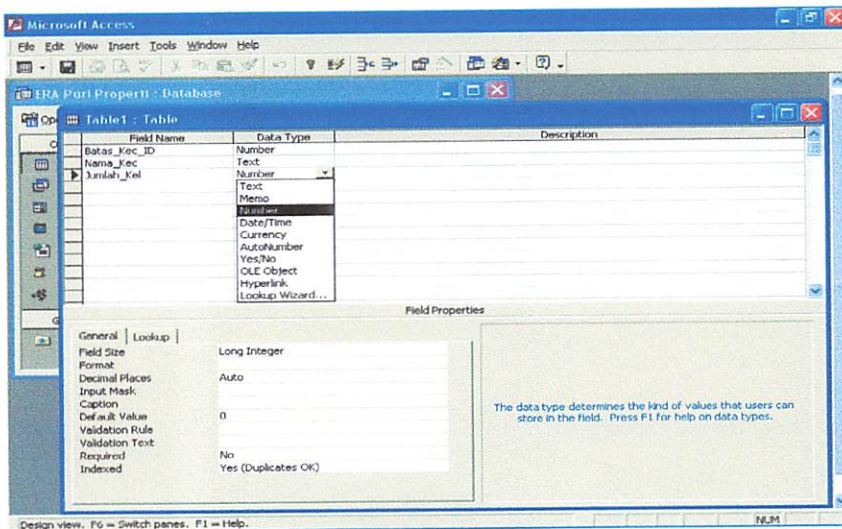
### 2.3.9.1 Pembuatan Tabel

Data atribut disusun dalam bentuk tabel-tabel yang merupakan kumpulan data yang tersusun menurut aturan tertentu. Secara fisik tabel berupa grid yang terdiri atas baris (*record data*) dan kolom (*field data*). Data atribut yang diperlukan untuk penelitian ini terdiri dari berupa tabel diantaranya adalah :

- Tabel Kecamatan dan Tabel Kelurahan
- Tabel Jalan dan Tabel Properti

Sebagai contoh pembuatan tabel, untuk tabel kecamatan yaitu :


1. Mengaktifkan software Microsoft Access dengan mengklik shortcut Microsoft Acces ( berlambang  ).
2. Dari kotak dialog Microsoft Acces, pilih Blank Database, lalu klik Ok
3. Dari kotak file name, ketikkan nama file, kemudian klik create
4. Kemudian akan muncul jendela database, lalu klik New
5. Klik design View, kemudian klik Ok.
6. Muncul kotak dialog Table, kemudian masukan ID atau keterangan yang lainnya pada kolom Description, Field Name dan pada kolom Data Type pilih namber atau text.
7. Setelah selesai diisi, maka klik Save untuk menyimpan data tersebut.
8. Untuk mengisinya tampilkan tabelnya, aktifkan tabel tersebut, dapat dilihat pada gambar 2.19.

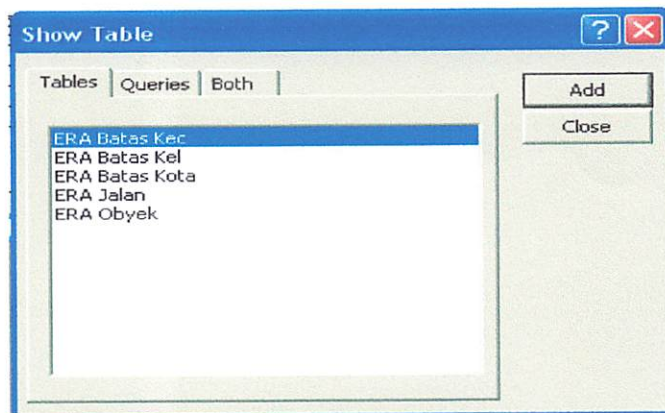


Gambar 2.19. Gambar Tampilan Pembuatan Basis Data dengan Microsof Access

### 2.3.9.2 Pembuatan Relationships Tabel

Setelah diagram dari beberapa entitas telah dibuat, selanjutnya hubungan antar entitas tersebut dapat diimplementasikan pada software Microsoft Acces. Adapun langkah kerjanya sebagai berikut :

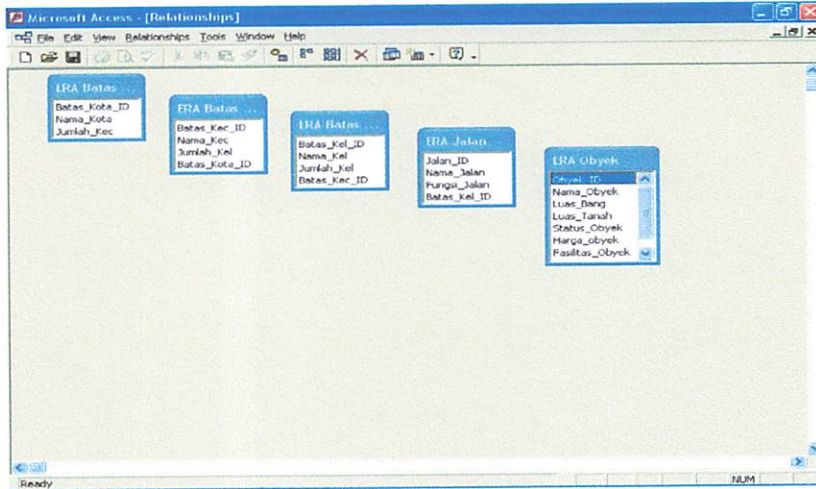
1. Mengaktifkan software Microsoft Acces dengan mengklik shortcut Microsoft Acces ( berlambang  Microsoft Access ).
2. Dari kotak dialog Microsoft Acces, pilih Blank Database, lalu klik Ok
3. Dari kotak file name dan buka file yang akan dihubungkan.
4. Pilih dan klik menu **T**ools, **R**elationships atau klik tombol toolbar Relationships. Kotak dialog Show Table akan ditampilkan pada gambar 2.20.



Gambar 2.20. Tampilan Kotak dialog Show Table

5. Pilih dan klik nama tabel yang akan dihubungkan, kemudian klik tombol perintah **A**dd atau klik dua kali paada nama tabel yang diinginkan. Apabila ingin menghubungkan seluruh tabel, maka select ( blok ) seluruh nama tabel, kemudian klik tombol **A**dd.

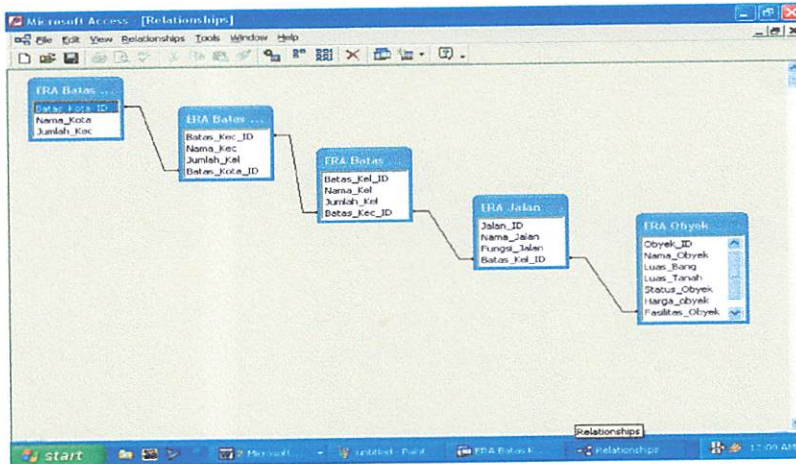
6. Untuk kotak dialog Show Table, Pilih dan klik tombol perintah Close. Dengan langkah ini jendela kerja Relationship yang berisi tabel/queries yang dipilih akan ditampilkan pada gambar 2.21.



Gambar. 2.21. Gambar Jendela Kerja Relationship

7. Untuk membuat hubungan antar tabel, klik nama field kunci yang diinginkan, kemudian digeser (*drag*) ke posisi nama field yang sama tabel lain. Atau dengan kata lain hubungan nama field yang sama antar tiap-tiap tabel dengan bantuan kursor dan akan akan tampilan pada gambar 2.22.
8. Dilakukan penghubungan antar tabel hingga membentuk Lop/kring/tertutup.
9. Untuk menyimpan hasil rancangan hubungan antar tabel tersebut, Pilih dan klik menu **File, Save (Ctrl+S)** atau tombol toolbar **Save**.
10. Untuk menutup jendela kerja Relationship, Pilih dan klik menu **File, Close** atau klik tombol **Close (X)**.





Gambar 2.22. Gambar Tampilan Relationship antar tabel

Untuk menggambarkan terjadinya hubungan antar entity digunakan diagram hubungan antar entity (Entity Relationship Diagram) yang bisa disingkat dengan E-R diagram. Notasi yang digunakan untuk menggambarkan E-R diagram adalah :

- a. Segiempat menggambarkan entity
- b. Diamon menggambarkan hubungan
- c. Elip atau lingkaran menggambarkan atribut

Pada E-R diagram cukup dilukiskan dengan jenis entiti dan hubungan yang terjadi. Sedangkan data atribut dituliskan dalam kerangka tabel entiti yang didalamnya berisi identitas dari tabel tersebut. Untuk penelitian ini digunakan derajat antar entiti yaitu hubungan satu ke banyak ( 1 : N ), untuk hubungan antara :

- Kecamatan dengan kelurahan
- Kelurahan dengan Jalan
- Jalan dengan Obyek

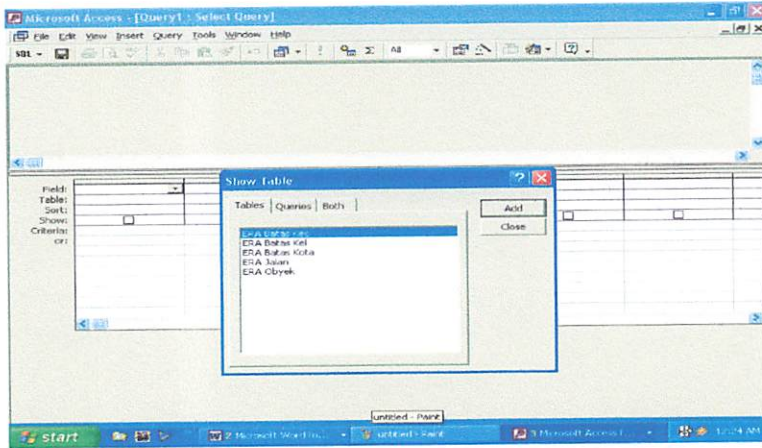
### 2.3.9.3 Pembuatan Query

Query merupakan pertanyaan yang diajukan dan diinginkan jawabannya dari Microsoft Access. Dengan menggunakan Microsoft Access, bisa untuk membuat tipe-tipe query sebagai berikut :

- *Select query*, yang dipergunakan untuk memilih dan menampilkan data dari satu atau lebih. Juga bisa dilakukan perhitungan pada data-data yang ditampilkan.
- *Parameter query*, untuk memunculkan pertanyaan tentang data yang akan ditampilkan query.
- *Crosstab query*, untuk membuat tabulasi silang pada data yang akan ditampilkan query.

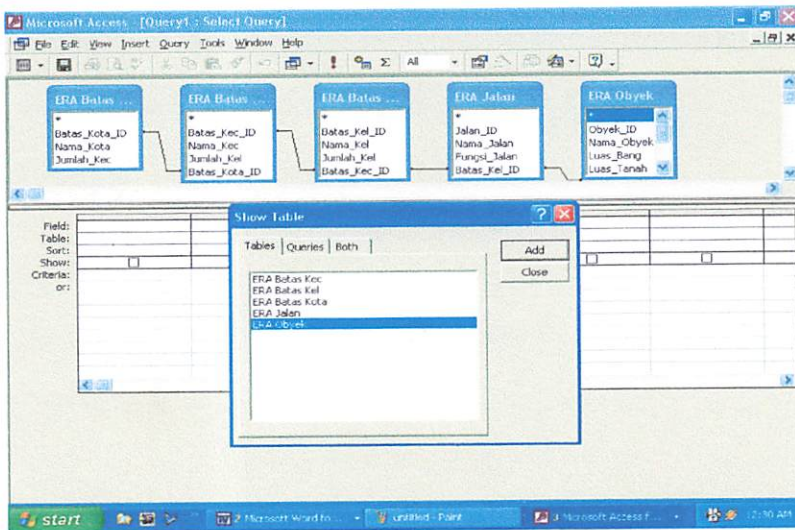
Untuk membuat query dari tabel yang ada dengan menggunakan fasilitas *Create Query in Design View* , dapat dilakukan dengan langkah sebagai berikut :

1. Buka file database yang telah dibuat, kemudian pada jendela database, klik *Query* yang ada dibawah *Objects* atau pilih menu *View*, database *Objects, Query*.
2. Pada jendela kerja database dengan pilihan objek queries tersebut, klik dua kali *Create Query in Design View*. Jendela kerja Select Query dengan kotak dialog Show Table akan ditampilkan pada gambar 2.23.



Gambar 2.24. Gambar Jendela Kerja Query

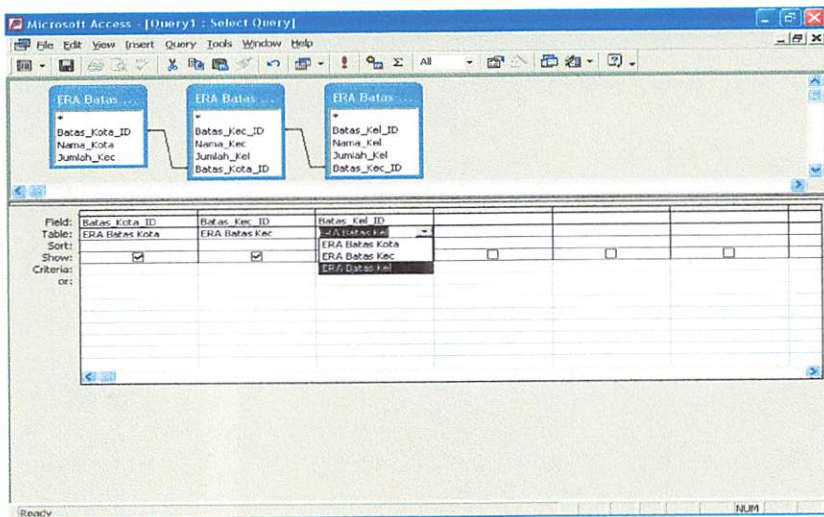
3. Pada kotak dialog Show Table, Pilih dan klik tabel atau query lain yang akan dilibatkan. Kemudian pilih dan klik tombol perintah Add.
4. Untuk menutup kotak dialog Show Table, Pilih dan klik tombol perintah Close. Dengan langkah ini jendela kerja Select Query yang siap digunakan akan ditampilkan pada gambar 2.24.



Gambar 2.24. Gambar Tampilan Jendela Kerja Query

5. Pada Jendela kerja Select query tersebut, tempatkan field-field yang diinginkan pada kolom yang disediakan. Untuk itu digunakan salah satu cara sebagai berikut :


- Mulai dengan kolom pertama, dengan menggunakan tombol daftar pilihan yang ada pada baris Tabel, Pilih dan klik tabel yang diinginkan, kemudian dengan menggunakan tombol daftar pilihan yang ada pada baris Field, Pilih dan klik field yang akan ditampilkan. Dengan menggunakan langkah yang sama, lanjutkan untuk pengisian kolom-kolom berikutnya yang akan ditampilkan pada gambar 2.25 dan gambar 2.26.

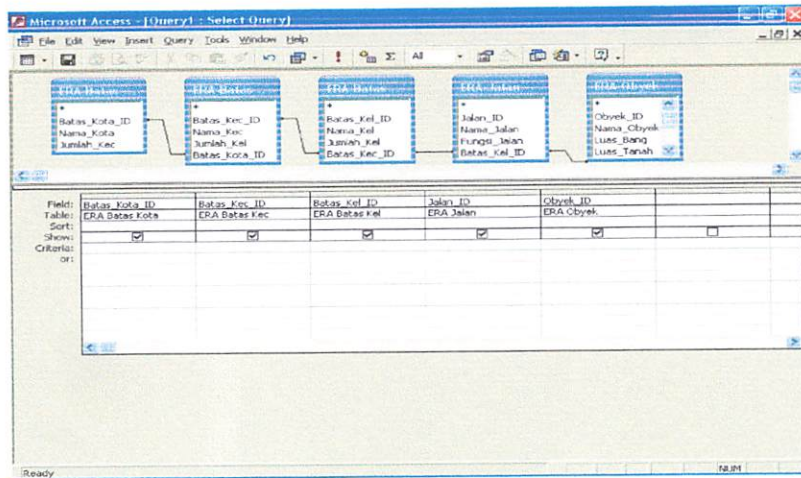


Gambar 2.25 : Tampilan jendela kerja Select Table dan Field

- Pada kotak daftar pilihan field yang ada dibagian atas jendela kerja Select Query, pilih dan klik nama field yang ingin ditampilkan kemudian digeser (drag) keposisi kolom yang diinginkan. Lakukan hal tersebut untuk baris field dan baris Table.
- Pada kotak daftar pilihan yang ada dibagian atas jendela kerja Select Query, Pilih dan klik dua kali nama field yang akan ditampilkan. Dengan langkah ini secara otomatis field yang akan dipilih akan ditempatkan pada posisi kolom berikutnya yang masih kosong.



6. Untuk melihat hasil Query yang baru, Pilih dan klik menu *Query, Run* atau Tombol toolbar  **Run**.



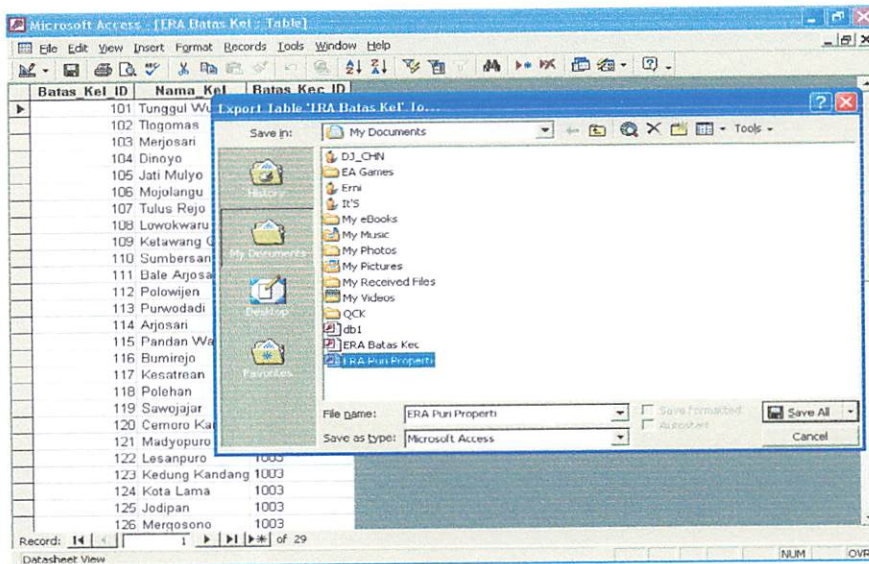
Gambar 2.26. Tampilan jendela kerja Select yang sudah siap dijalankan.

### 2.3.10 Export Basis Data

Basis data dapat ditampilkan dan digabungkan dengan data spasial di software Arc View, maka data tersebut harus diexport sehingga menjadi DBASE.

Adapun cara mengexportnya adalah sebagai berikut :

1. Di jendela Database, klik *tab Tables* (daftar tabel akan muncul)
2. Memilih tabel yang akan diexport (klik dua kali atau klik Open)
3. Memilih menu file, kemudian klik *Save As / Export*
4. Akan muncul kotak dialog *Save As / Export*
5. Memilih *Dbase 5* pada *Save As Type*, kemudian klik *Export*, dapat dilihat pada gambar 2.27.



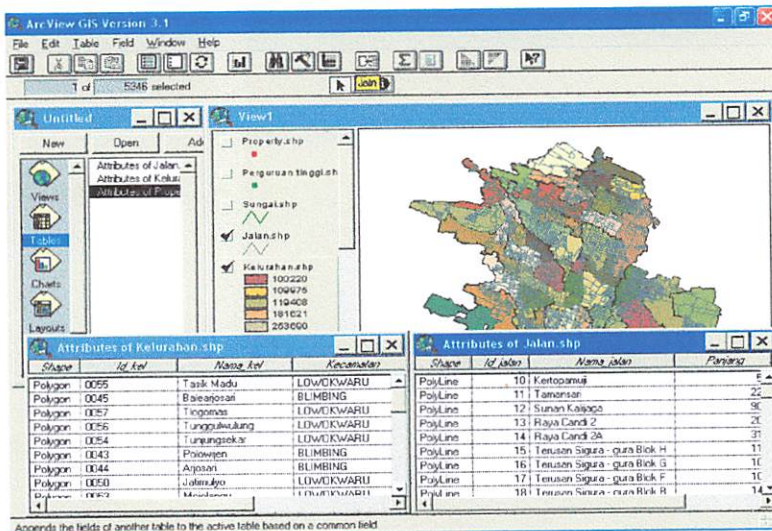
Gambar 2.27. Tampilan cara mengexport Database

### 2.3.11. Join Item atau Penggabungan Data

Join Item dilakukan di software Arc View, maksud dari *join item* yaitu untuk menggabungkan Database dari Microsoft Access dengan data spasial. Yang dijoinkan adalah Id masing-masing. Adapun caranya adalah sebagai berikut :

1. Mengaktifkan software Arc View.
2. Muncul kotak dialog Untitled, kemudian klik *new* pada kotak dialog Untitled dan muncul *View 1*, lalu klik *Add Theme*.
3. Memilih *file* yang akan ditampilkan pada kotak View 1, kemudian diklik Ok.
4. Mengklik *theme Table*, sehingga akan muncul atribut dari file (*coverage*) yang dipilih.
5. Kemudian mengklik *tables* pada *Untitled*, lalu klik *add*.
6. Mengklik ID dari Tables, kemudian *ID Attribute Of* (nama file).

7. Kemudian mengklik **toolbar join**, untuk menggabungkan dua ID tersebut. Adapun tampilan hasil join Item pada gambar 2.28.



Gambar 2.28. Tampilan Penggunaan Perintah *Join* pada ArcView

### 2.3.12. Convert File

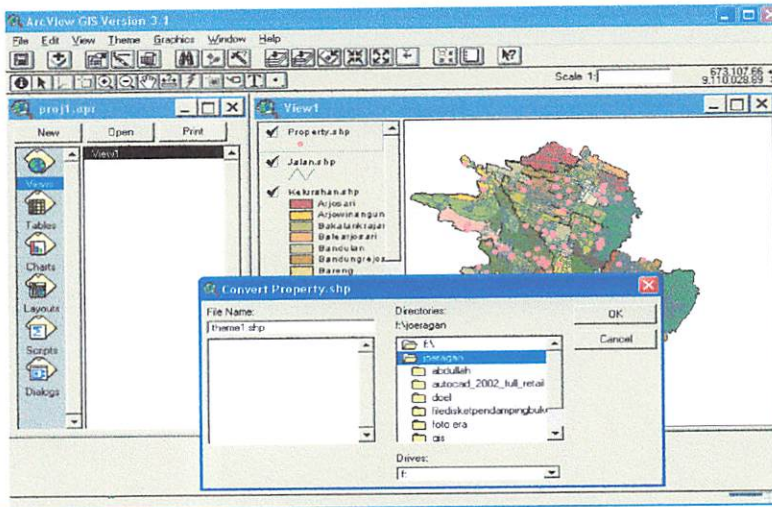
Setelah data spasial dan data atribut telah selesai dilakukan penggabungan, kemudian data hasil penggabungan tersebut harus di *Convert* kedalam bentuk ekstension ( \*.shp ) dengan tujuan supaya *file* tersebut dapat terbaca pada program Map Objects.

Langkah - langkah untuk meng-*Convert file* tersebut adalah sebagai berikut :

1. Mengaktifkan Arc View
2. Membuka *file* yang akan di-*convert*
3. Setelah *file* dibuka, kemudian pilih perintah *theme* yang terdapat pada menu, selanjutnya pilih *convert to shapefile* untuk membuka *form*-nya.



4. Isikan nama *file* pada *File Name*, dimana *file* tersebut akan diletakkan dengan mengganti *directories*.



Gambar 2.29. Tampilan Perintah *Convert File* pada ArcView

### 2.3.13. Perancangan Sistem dalam Visual Basic 6.0.

Pada Perencanaan sistem ini dilakukan tahapan perancangan interface dari program, yaitu merancang bentuk dari program sistem informasi yang mudah diakses, cepat dan efisien. Merancang output sistem yang dapat berupa tampilan dilayar monitor atau dalam bentuk laporan tercetak. Dari kedua perancangan tersebut, yang lebih utama adalah proses pendesainan prosedur/alur dari input data.

#### 2.3.13.1. Pembuatan Atribut-Atribut Form

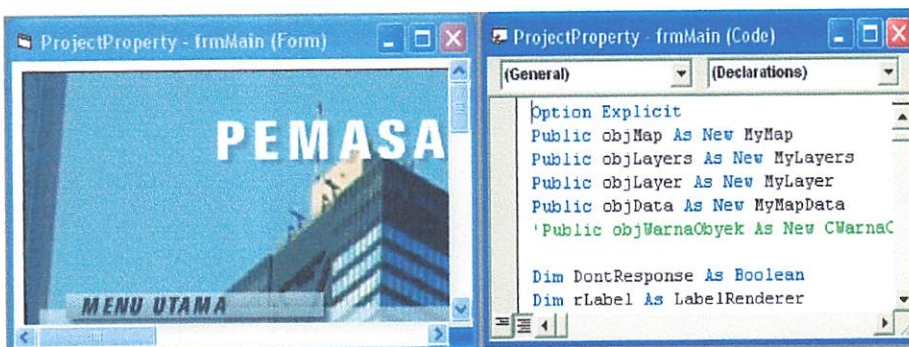
Form atau project yang dibuat, komponen-komponen pendukungnya tidak lepas dari struktur tabel yang telah disiapkan

sebelumnya. Fungsi-fungsi *Build in* program juga tidak terlepas dari *event list* yang telah ada. Adapun langkah pembuatan *form* sebagai berikut :

1. Mengaktifkan IDE Visual Basic dan pilih **Standar EXE**
2. Pada form1 membuat kontrol label dan kontrol text, serta Command Button.
3. Mengatur properti-properti dari masing-masing kontrol sesuai kebutuhan dan design yang dibuat.
4. Melakukan pengkodean pada masing-masing form yaitu menuliskan listing program yang akan dibuat, dengan cara menekan F7 untuk menampilkan jendela koding atau menekan double klik pada form tersebut.
5. Langkah yang sama dilakukan pada form-form berikutnya.

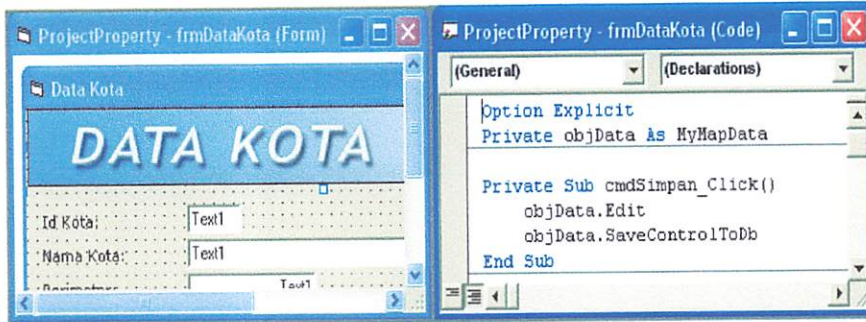
Pada penelitian ini form-form yang dibuat sebagai berikut :

1. Form main, yang merupakan tampilan utama dari visualisasi hasil dari program yang akan dibuat.



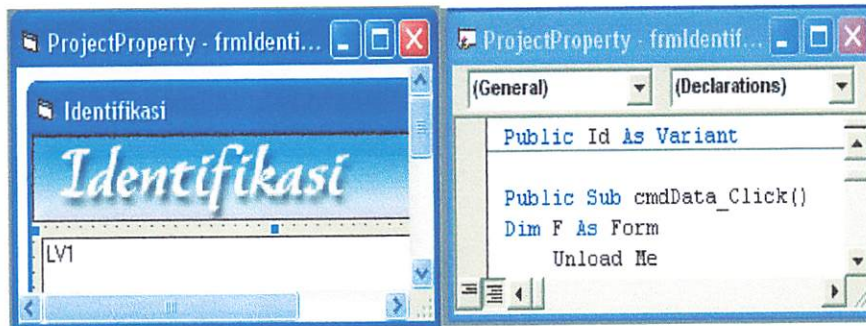
Gambar 2.30. Tampilan Form Main dan Penulisan Koding

2. Form Data, merupakan form yang dibuat untuk menampilkan data-data yang telah dibuat sebelumnya dalam database. Form data ini terdiri dari data kota, kecamatan, kelurahan, jalan, sungai, perguruan tinggi, dan data properti.



Gambar 2.31. Tampilan Form Data dan Penulisan Koding

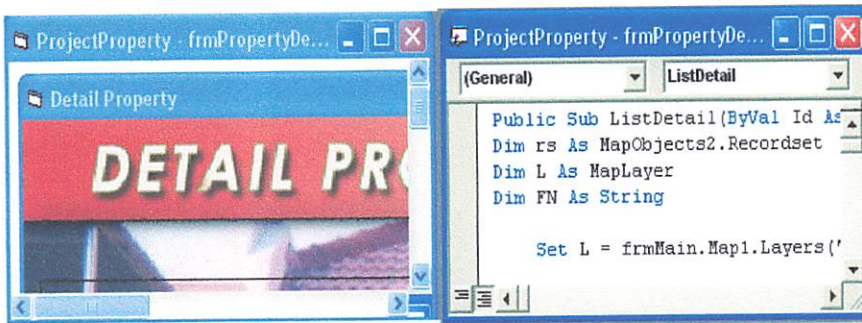
3. Form Identifikasi, form ini berisikan tampilan identifikasi atau keterangan yang akan ditampilkan dari data-data tersebut.



Gambar 2.32. Tampilan Form Data dan Penulisan Koding

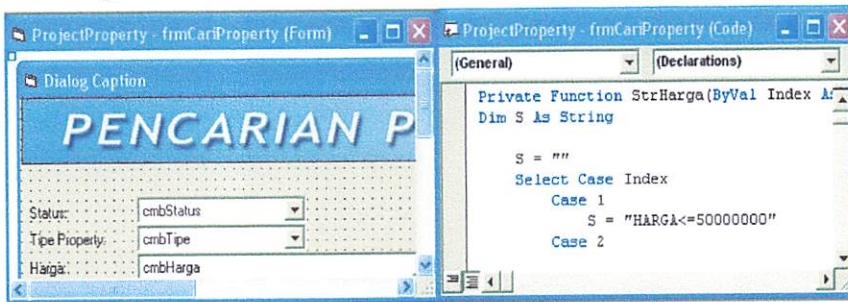
4. Form Properti Detail, form ini bertujuan untuk menampilkan data atau keterangan dari properti secara lebih detail.





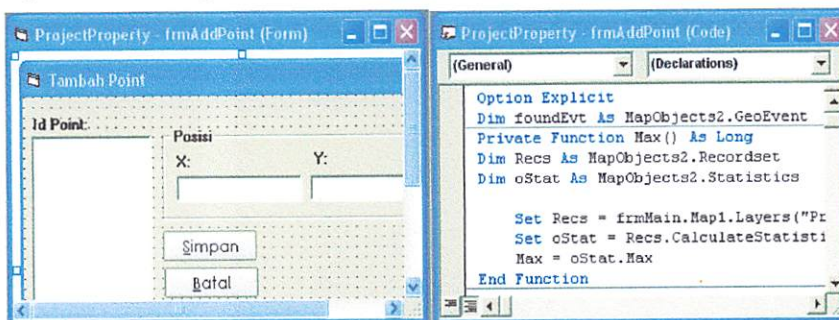
Gambar 2.33. Tampilan Form Detail dan Penulisan Koding

5. Form Pencarian, form ini berfungsi untuk mencari lokasi database program yang telah dibuat dan akan menampilkannya. Form pencarian ini terdiri dari pencarian data properti, perguruan tinggi, jalan, sungai, dan Kelurahan.



Gambar 2.34. Tampilan Form Pencarian dan Penulisan Koding

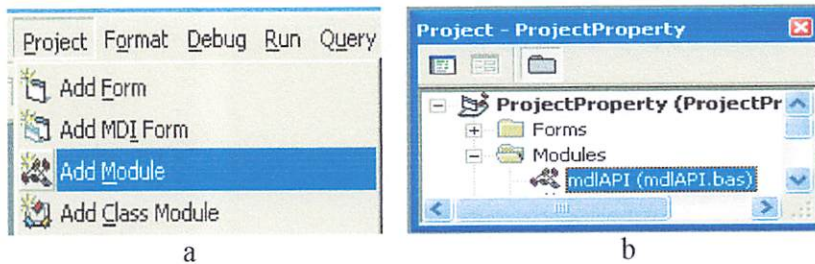
6. Form Add Point, form ini berfungsi untuk *update point* data properti dengan cara mengisi koordinat titiknya.



Gambar 2.35. Tampilan Form Add Point dan Penulisan Koding

### 2.3.13.2. Pembuatan Module pada Visual Basic

Visual Basic menyediakan module yang dapat digunakan untuk memuat fungsi, subrutin, konstanta, variabel, dan type ciptaan yang akan disediakan bagi keseluruhan project. Untuk membuat module dalam project menggunakan perintah *Project, Add Module* (gambar 2.36a.). Kemudian pada Jendela Project Explorer akan nampak module yang telah dibuat (gambar 2.36b.). Untuk melakukan koding, dengan cara menekan double klik pada Module tersebut. Deklarasi variabel pada module menggunakan awalan DIM, PRIVATE maupun PUBLIC, dimana awalan DIM dan PRIVATE akan membentuk variabel modul level, sedangkan awalan PUBLIC menghasilkan variabel global yang akan berfungsi bagi keseluruhan program.



Gambar 2.36.  
Tampilan Perintah Module dan Jendela Module Explorer

### 2.3.13.3. Distribusi Sistem

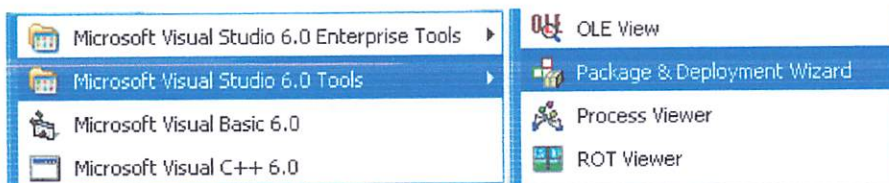
Distribusi sistem dimaksudkan untuk mengkompilasikan program kedalam bentuk *executable* (file \*.EXE) agar program tersebut dapat dijalankan diluar sistem Visual Basic. Program yang telah dibuat



kemudian didistribusikan kedalam bentuk program *SETUP* karena Microsoft Visual Basic telah menyediakan fasilitas tersebut, yaitu dengan menggunakan program *Package & Deployment Wizard* yang terintegrasi didalam *Microsoft Visual Basic 6.0 Tools*.

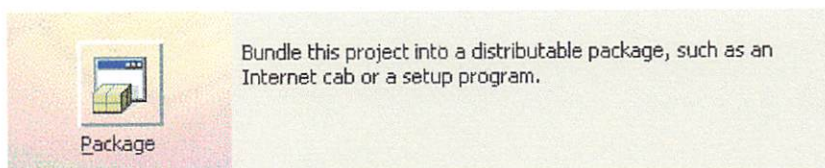
Penggunaan program *Package & Deployment Wizard* langkah – langkahnya sebagai berikut :

1. Membuka program *Package & Deployment Wizard*



Gambar 2.37. Program *Package & Deployment Wizard* pada Start Menu

2. Memilih file project yang akan dibuat program *SETUP* nya, kemudian tekan *OPEN*.
3. Memilih *Package ToolBar* untuk memulai pembuatan program *SETUP*



Gambar 2.38. *Package ToolBar*

4. Menunggu beberapa saat sementara *processing* dilaksanakan.
5. Pada kotak dialog *Package & Deployment Wizard – Package Type*, pilih *Standard Setup Package* pada kotak pilihan *Package Type*, lalu tekan *Next* untuk melanjutkan proses berikutnya.

6. Menentukan lokasi atau folder tempat menyimpan file SETUP pada kotak dialog *Package Folder*.
7. Selanjutnya akan muncul kotak dialog *Included File*. Kotak ini berfungsi untuk memasukan file-file pendukung program. Menekan tombol *Add* dan menentukan lokasi filenya untuk memasukan file yang dibutuhkan.
8. Setelah menekan *Next* pada kotak dialog *Included File*, akan muncul kotak dialog *Cab Option*. Fungsinya adalah untuk mengkompres kedalam file-file kabinet. Selanjutnya tekan *Next*.
9. Tahapan berikut adalah penentuan judul program instalasi. Pada kotak dialog *Installation Title*, diisi nama program yang dibuat dan diakhiri dengan menekan tombol *Next*.
10. Menentukan *gruping program* pada start menu pada kotak dialog *Start Menu Items*.
11. Setelah menekan tombol *Next*, kemudian mengakhiri semua proses dengan menekan tombol *Finish* pada kotak dialog *Finished!*.

#### **2.3.13.4. Implementasi dan Evaluasi Sistem**

Tahapan ini merupakan proses sosialisasi program bagi pengguna, atau dengan kata lain mengoperasikan program secara keseluruhan. Dengan demikian sekaligus dapat dievaluasi kekurangan-kekurangan yang terjadi dalam program.

Pada penelitian ini, hasil dari program berupa visualisasi informasi dari sebuah pemasaran perumahan sekunder, dengan sistem desain yang didukung dengan desain antarmuka-pengguna (*User Interface*) akan membantu memudahkan pemahaman *user* dalam mengoperasikan. Desain tampilan dibuat lebih menarik secara visual yang menunjukkan jati diri sebuah pemasaran perumahan untuk mendukung aplikasi bisnis. Beberapa fasilitas ditampilkan untuk mendukung aplikasi sistem informasi. Tools semacam *Zoom In*, *Zoom Out*, *Info*, *Label*, dapat dimanfaatkan untuk mengeksplorasi informasi dibalik simbol-simbol peta.



Gambar 2.39. Tampilan Program Informasi

### BAB III




#### HASIL DAN PEMBAHASAN

Hasil dari penelitian ini adalah berupa visualisasi peta tematik untuk informasi pemasaran perumahan sekunder di kota Malang dari sebuah broker properti yaitu Era Puri Properti. Peta tematik yang dihasilkan ini berupa peta posisi beserta basis datanya dari setiap properti yang dipasarkan. Ada 3 menu berisikan fasilitas-fasilitas yang telah disediakan dalam sistem informasi ini, diantaranya adalah :







1. Menu Utama, berisikan fasilitas untuk mengeksplorasi data-data yang tersedia, fasilitas untuk mencetak hasil, fasilitas aplikasi untuk menghubungkan dengan program-program lain, menu help, dan menu untuk keluar dari sistem informasi ini.



Gambar 3.1. Menu Utama dalam Sistem Informasi Pemasaran.

2. Menu Map, berisikan perintah-perintah yang dapat digunakan untuk mengeksplorasi peta dan informasinya, berikut adalah *toolbar* yang tersedia dalam menu map beserta keterangannya.
  - ~ Zoom In  untuk memperbesar gambar tampilan pada view.
  - ~ Zoom Out  untuk memperkecil gambar tampilan pada view.
  - ~ Pan  untuk mengeser tampilan gambar pada view.

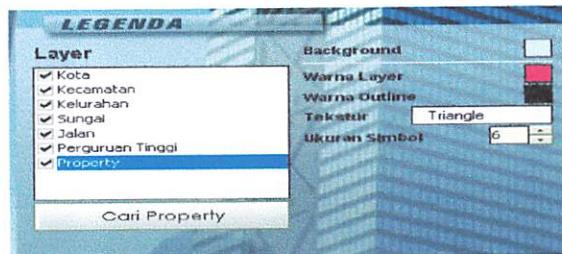


- ~ Zoom Extent  untuk mengembalikan ke ukuran normal tampilan peta
- ~ Identifikasi  untuk menampilkan informasi data individual.
- ~ Pointer Normal  Penunjuk obyek aktif.
- ~ Label  berfungsi untuk menampilkan label pada peta.
- ~ Add Point  untuk menambahkan point dari obyek properti yang dapat dilakukan dengan memasukan koordinat pada *form* yang tersedia (gambar 3.2.) atau langsung meng-klik ditempat yang diinginkan.
- ~ Pilih Point  untuk mengetahui besar koordinat khusus pada obyek properti.

Id Baru: 95	X: <input type="text"/>	Y: <input type="text"/>	<input type="checkbox"/> Tampilkan Data Property baru
-------------	-------------------------	-------------------------	---

Gambar 3.2. Tampilan *Form* untuk memasukan koordinat

3. Legenda, berisikan *layer* peta dan fasilitas pencarian properti serta fasilitas *editing* untuk warna dan simbol tampilan peta.



Gambar 3.3. Tampilan Menu Legenda

### 3.1. Penyajian Peta Informasi Obyek Pemasaran Properti

Peta informasi obyek pemasaran properti ini merupakan peta lokasi dari obyek rumah sekunder yang akan dipasarkan. Pada peta ini akan dapat diperoleh informasi penuh dari data-data obyek yang







Gambar 3.5. Tampilan Informasi Individual Properti



Gambar 3.6. Tampilan Informasi Detail Properti

### 3.2. Penyajian Data Atribut Objek Properti

Sistem informasi ini mempunyai fasilitas untuk menampilkan data-data secara menyeluruh dari tiap layer yang aktif. Karena program ini merupakan sistem informasi pemasaran perumahan sekunder, maka data tentang properti lebih mendetail. Data atribut obyek properti yang ditampilkan diantaranya adalah lokasi obyek, tipe obyek, status pemasaran, luas bangunan, luas tanah, harga obyek, besarnya daya listrik,

penggunaan daya air, jumlah kamar tidur dan kamar mandi, nama penghubung (*broker*), dan fasilitas-fasilitas lainnya yang terdapat pada obyek properti tersebut yang disertai dengan gambar atau foto obyek.

Fasilitas lain yang dapat digunakan pada tampilan data atribut ini adalah *editing*, pencarian data, dan penyajian lokasi. Untuk melakukan *editing* data bisa dilakukan dengan cara langsung mengganti data yang dimaksud dan menyimpannya dengan meng-klik menu *Simpan* yang tersedia pada program ini. Selain menyimpan, program ini juga menyediakan fasilitas untuk menghapus data properti. Untuk melakukan pencarian data yang diinginkan, sistem informasi ini dilengkapi fasilitas pencarian data. Dengan menekan menu *Cari* kemudian akan ditampilkan *form* Pencarian Properti, dalam *form* ini terdapat 5 (lima) pilihan kriteria untuk pencarian properti, yaitu berdasarkan status pemasaran, tipe properti, harga properti, jalan dan kelurahan dimana lokasi properti berada. Dan untuk menampilkan lokasi properti pada *form* data ini, cukup dengan meng-klik menu *Lokasi* maka akan ditampilkan lokasi pada *view* sesuai dengan data properti yang ingin ditampilkan lokasinya.







Gambar 3.7. Tampilan Data Atribut Obyek Properti

### 3.3. Form Pencarian

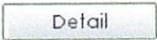
Sistem informasi ini juga memberi kemampuan pencarian informasi atas obyek-obyek sesuai *theme* atau *layer* yang diinginkan. Ada 2(dua) cara untuk penggunaan *form* pencarian, yaitu :

1. Menggunakan menu **Cari Properti** yang terdapat pada menu **Legenda**.

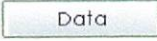
Penggunaan *form* pencarian ini hanya dapat digunakan pada *theme* atau *layer* properti saja (gambar 3.8.). Berikut langkah penggunaannya :


- ~ Pilih *theme Properti*
- ~ Klik ikon *Cari Properti* 
- ~ Pilih kriteria - kriteria yang telah tersedia di *form* pencarian, banyak pilihan dari masing-masing kriteria, berdasarkan status pemasaran, tipe properti, dan harga properti yang diinginkan. Dilanjutkan meng-klik icon *Cari* 

- ~ Secara otomatis hasil data akan ditampilkan sesuai dengan kriteria diatas.
- ~ Masih dalam *form* pencarian, selanjutnya dapat dilakukan eksplorasi sesuai dengan menu yang tersedia.

 : Untuk menampilkan informasi data individual secara detail dari obyek properti yang dipilih.



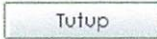
 : Untuk menampilkan lokasi pada view.

 : Untuk menampilkan basis data obyek properti

- ~ Untuk mengakhiri *form* pencarian klik pada tombol 

## 2. Menggunakan menu **Cari** yang terdapat pada *form* **Data**

Penggunaan menu **Cari** pada *form* **Data** ini dapat digunakan pada semua *layer* atau *theme*. Langkah penggunaannya seperti berikut :

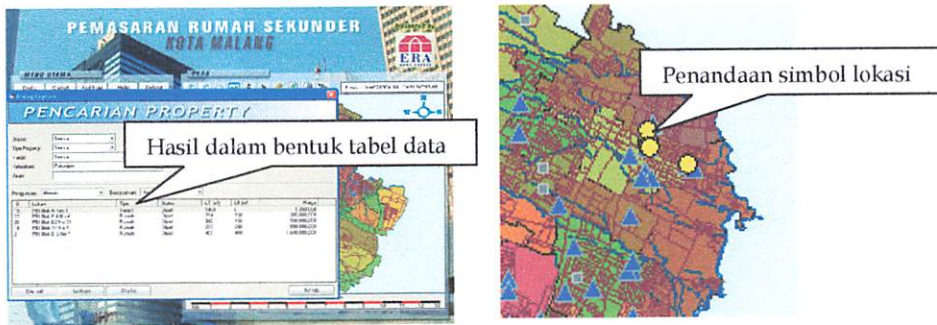
- ~ Klik *theme* atau *layer* yang akan dicari informasinya
- ~ Pilih menu **Data** pada menu utama, kemudian akan ditampilkan *form* data.
- ~ Dari *form* data pilih dan klik ikon **Cari** 
- ~ Sistem informasi akan memunculkan jendela *form* pencarian
- ~ Pilih kriteria - kriteria yang telah tersedia di *form* pencarian, banyak pilihan dari masing-masing kriteria, berdasarkan **status pemasaran, tipe properti, harga properti, Kelurahan** dan atau **Jalan** yang diinginkan. Dilanjutkan meng-klik icon **Cari** 
- ~ Secara otomatis hasil data akan ditampilkan sesuai dengan kriteria diatas dalam bentuk tabel dan penandaan posisi lokasi obyek dalam bentuk simbol dan pewarnaan yang berbeda (gambar 3.9.)
- ~ Masih dalam *form* pencarian, selanjutnya dapat dilakukan eksplorasi sesuai dengan menu yang tersedia.
- ~ Untuk mengakhiri *form* pencarian klik pada tombol 



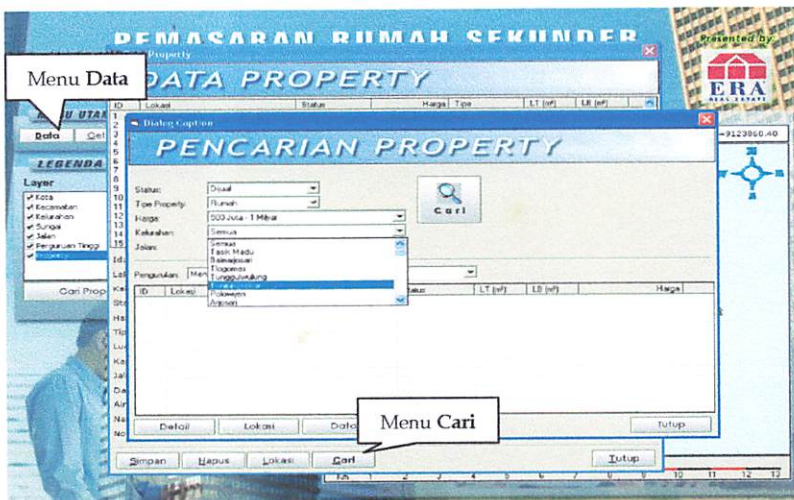
~ Untuk kembali ke tampilan utama klik tombol **Tutup** pada *form* data. ( gambar 3.10).



Gambar 3.8. Tampilan *form* Pencarian dari Menu Cari Properti



Gambar 3.9. Hasil Pencarian dalam bentuk tabel dan simbol lokasi



Gambar 3.10. Tampilan *form* Pencarian dari Menu Data

### 3.4. Form Updating

Fasilitas yang lain dalam sistem informasi ini adalah kemampuannya untuk meng-*update point* atau titik dimana lokasi obyek properti itu berada, hal ini penting karena sebagai sistem informasi pemasaaran, datanya selalu *update* dan silih berganti. Fasilitas ini juga dapat dilakukan untuk mengisi data atribut baru.

Berikut adalah langkah untuk meng-*update point* obyek properti :

~ Pilih *theme* properti

~ Klik tombol *Tambah Point* 

~ Sistem Informasi akan jendela dialog seperti berikut :

Id Baru: 95 X: <input type="text"/>	Y: <input type="text"/>	<input type="checkbox"/> Tampilkan Data Property baru
-------------------------------------	-------------------------	---

~ Klik pada kotak tampilkan data baru, fungsinya adalah untuk menampilkan isian data properti baru.

~ Masukkan angka besaran koordinat X dan Y [tekan Enter].

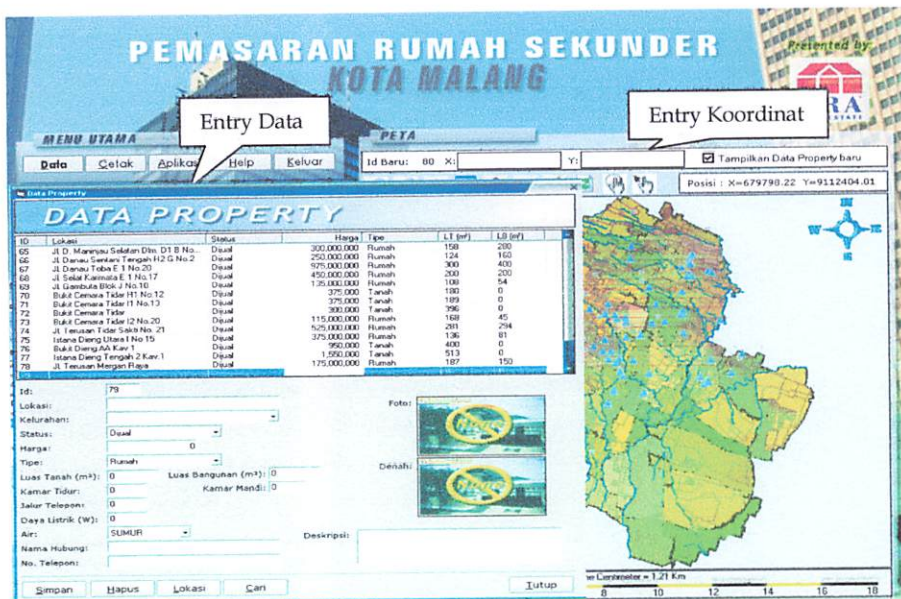
~ Kemudian akan ditampilkan *form* data dan isi data-data tersebut.

~ Setelah isian data selesai lalu klik menu *Simpan* pada *form* data.

~ Untuk mengecek point yang telah dimasukkan klik menu *Lokasi*.

~ Untuk menutup *form* data tekan menu *Tutup*.

~ Langkah yang sama dapat dilakukan untuk menambah point properti selanjutnya.



Gambar 3.11. Tampilan form Updating

### 3.5. Pembahasan Hasil

#### 3.5.1. Pembahasan Data Spasial

Dalam pelaksanaan penelitian ini menggunakan data spasial dan non spasial yang disusun dalam beberapa layer untuk menginformasikan obyek properti yang ditawarkan. Untuk menampilkan masing-masing layer digunakan *software* Map Basic 2.1. yang diintegrasikan dengan *software* Visual Basic 6.0. untuk penyajian. Ada tujuh *theme* atau layer yang ditampilkan dalam sistem informasi ini, diantaranya adalah *theme* Kota, Kecamatan, Kelurahan, Jalan, Sungai, Perguruan Tinggi, dan *theme* Properti yang merupakan fokus utama dari tujuan pembuatan sistem informasi ini. Pada *theme* properti akan ditampilkan titik (*point*) yang merupakan posisi dari tiap obyek properti yang disimbolkan berupa segitiga (*triangle*).



Penentuan posisi dari tiap obyek properti dilakukan dengan menggunakan alat bantu GPS genggam (*hendheld*) berdasarkan data alamat dari masing-masing obyek properti. Hasil dari alat ini adalah data koordinat ( X ; Y ) yang kemudian diproses pada software AutoCAD 2000 sehingga didapatkan peta posisi objek properti. Berikut adalah contoh tabel koordinat posisi obyek properti :

Tabel 3.1. Data Koordinat Posisi Obyek Properti

No	Lokasi Properti	Tipe Properti	Status Properti	Harga	Koordinat	
					X	Y
1	P B I Blok A 4/ No 1	RUMAH	JUAL	500 000 000	682146.96	9122694.24
2	P B I Blok A Kav 1	TANAH	JUAL	1 250 000/M	682133.99	9122535.58
3	P B I Blok B 2/ No 31	RUMAH	JUAL	500 000 000	682130.91	9122435.82
4	P B I Blok B 6/ No 7	RUMAH	JUAL	1 600 000 000	682155.03	9122397.27
5	P B I Blok F 4/ No 41	RUMAH	JUAL	385 000 000	682794.62	9122099.49
6	P B I Blok G 3/ No 4	RUMAH	JUAL	425 000 000	682922.11	9122051.13
7	P B I Blok K 2/ No 2	RUMAH	SEWA	4 500 000/Th	682104.61	9121995.25
8	P B I Blok K 7/ No 88	RUMAH	JUAL	350 000 000	682198.42	9121895.82
9	P B I Blok K 3/ No 09	RUMAH	JUAL	235 000 000	682048.96	9122017.94
10	TELUK ETNA Kav. 123	RUMAH	JUAL	350 000 000	682212.91	9122769.83

### 3.5.2. Pembahasan Data Non Spasial

Dalam penyusunan basis data sistem informasi pemasaran perumahan sekunder ini menggunakan struktur data relational dimana semua data disimpan dalam tabel dua dimensi, dengan *Enterprise Rules* (ER) modelling menggunakan pendekatan *Top Down*. Pada ER diagram ini terdapat beberapa entitas, yaitu : relasi antara Kota dengan Kecamatan, Kecamatan dengan Kelurahan, Kelurahan dengan Jalan, Kelurahan dengan Properti, Jalan dengan Properti, dimana relasi-relasi tersebut memiliki hubungan relasi *One to Many* dengan *obligatory* pada dua sisi,

relasi *Many to One* dengan *obligatory* pada satu sisi, dan relasi *One to One obligatory* pada satu sisi.

Penyusunan data non spasial menggunakan *software* Microsoft Access 2000. Dalam penyajiannya, data-data yang disusun sesuai dengan atribut ditampilkan pada *software* Visual Basic 6.0. (*link data*) sebagai informasi basis data pada masing-masing layer dimana data-data tersebut dapat dilakukan proses *editing* dan proses *up to date* data masukan baru.

Informasi non spasial yang ditampilkan adalah data atribut dari masing-masing *theme* peta, untuk informasi dari masing-masing obyek properti yang ditampilkan adalah foto obyek dan data pendukung berupa:

1. Alamat Lokasi Obyek
2. Tipe Properti
3. Status Pemasaran
4. Harga Obyek Properti
5. Luas Tanah dan Luas Bangunan
6. Jumlah Kamar Tidur dan Kamar Mandi
7. Jumlah Jalur Telepon
8. Besar Daya Listrik
9. Sumber Penggunaan Air
10. Nama dan No Telepon *Broker*
11. Fasilitas-Fasilitas lainnya yang terdapat pada obyek properti.

### **3.5.3. Pembahasan Pembuatan Program**

Informasi pemasaran perumahan sekunder yang disusun dalam basis data dan peta tematik semuanya digabungkan dalam sebuah program yang dibuat dengan menggunakan *software* Visual Basic 6.0. dan Map Basic 2.1. Hasil dari pembuatan program ini adalah membuat suatu

penyajian dalam sebuah pemasaran perumahan sekunder lebih menarik, efektif dan efisien bagi pemakai dalam hal ini *marketing, broker properti* atau *klien* dalam mencari informasi spasial dan atributnya dari obyek properti yang dipasarkan.

Penggunaan *software* Visual Basic 6.0. dimaksudkan karena Visual Basic 6.0. merupakan bahasa program *event-driven* berbasiskan *windows* sehingga perancangan *user interface* dibuat untuk memudahkan pemakai dalam penggunaan sistem informasi ini.

Dalam pembuatan program ini terdiri atas tiga bagian *form* besar yaitu *form* MDI (sebagai *Main form*), *form* Peta dan *form* Database.

A. *Form* MDI, sebagai *main form* (form utama) yaitu *form* sebagai tempat induk untuk mengatur *form-form* lainnya.



Gambar 3.12. *Form* MDI

B. *Form* Peta, *form* ini berfungsi untuk menampilkan peta tematik baik itu peta administrasi maupun peta lokasi obyek properti. Untuk memanggil peta, Visual Basic harus dikoneksi dengan Map Object yang berisi deklarasi pernyataan-pernyataan yang dapat menjalankan perintah untuk menampilkan data spasial.



C. *Form Database*, *form* ini berfungsi untuk menampilkan *database* (tabel) yang dibuat pada program Microsoft Access. Pada *form* ini telah dilengkapi bahasa *query* (SQL) yang diwakili *command button* dan *text* yang merupakan fasilitas pada Visual Basic untuk melakukan pencarian berdasarkan *query*.

Selain pembuatan *form-form* seperti diatas, pelaksanaan pembuatan program yang terpenting adalah mengimplementasikan fungsi atau fasilitas-fasilitas kedalam bahasa basic yang dapat dijalankan dalam program. Diantaranya adalah menampilkan peta, dalam hal ini Visual Basic harus dikoneksi dengan Map Object sebagai program penunjang dalam menampilkan peta. Adapun prosedur (listing program) sebagai berikut :

```
'Private Attribute
Private Conn_ As ADODB.Connection
Private DC_ As DataConnection
Private Map_ As MapObjects2.Map

'Private Property
'Public Property
Public Property Let Connection(pConn As ADODB.Connection)
    Set Conn_ = pConn
End Property
Public Property Get Connection() As ADODB.Connection
    Set Connection = Conn_
End Property
Public Property Let DataConnection(DC As Object)
    Set DC_ = DC
End Property
Public Property Get DataConnection() As Object
    Set DataConnection = DC_
End Property
Public Property Let Map(pMap As Object)
    Set Map_ = pMap
End Property
Public Property Get Map() As Object
    Set Map = Map_
End Property
```

Fasilitas lain yang harus diimplementasikan dalam bahasa pemrograman adalah fasilitas pencarian. Listing program sebagai berikut:

```

Private Sub cmdCari_Click()
Dim L As MapObjects2.MapLayer
Dim rs As MapObjects2.Recordset
Dim Exp As String
Dim li As ListItem
Dim shp As Object

    Set L = frmMain.Map1.Layers("property")
    Exp = ""
' Cek Combo Status
    If cmbStatus.ListIndex <> 0 Then
        Exp = " STATUS='" & (Me.cmbStatus.Text) & "'"
    End If
' Cek Combo Tipe
    If cmbTipe.ListIndex <> 0 Then
        Exp = Exp & IIf(Exp <> "", " AND ", "") & " TIPE='" &
            (Me.cmbTipe.Text) & "'"
    End If
' Cek Combo Harga
    If cmbHarga.ListIndex <> 0 Then
        Exp = Exp & IIf(Exp <> "", " AND ", " ") &
            StrHarga(Me.cmbHarga.ListIndex)
    End If
' Cek Combo Kel
    If cmbKel.ListIndex <> 0 Then
        Exp = Exp & IIf(Exp <> "", " AND ", " ") & " NAMA_KEL='" &
            Me.cmbKel.Text & "'"
    End If
' Cek Jalan
    If txtJalan.Text <> "" Then
        Exp = Exp & IIf(Exp <> "", " AND ", " ") & " LOKASI like '%" &
            Me.txtJalan.Text & "%'"
    End If

Private Sub cmdData_Click()
    If LV1.ListItems.Count = 0 Then Exit Sub
    If frmDataProperty.Visible = False Then Load frmDataProperty
    frmDataProperty.objData.FindItemInListView LV1.SelectedItem.Text, 0, 1, 0
    frmDataProperty.objData.SelectFromListView
    If frmDataProperty.Visible Then
        Unload Me
        frmDataProperty.SetFocus
    Else
        frmDataProperty.Show 1
    End If
End Sub
Private Sub cmdDetail_Click()
    frmPropertyDetail.ListDetail Me.LV1.SelectedItem.Text
End Sub

```

Listing program lainnya akan ditampilkan pada lampiran.

Untuk efisiensi bahasa basic yang telah dibuat, selanjutnya adalah mengumpulkan prosedur (listing program) dalam suatu tempat yang telah disediakan dalam Visual Basic yaitu Module. Fungsi Module adalah untuk memuat fungsi, subrutin, konstanta, variabel dan type yang telah dibuat bagi keseluruhan *project*. Adapun module yang dibuat dalam program informasi ini adalah :

1. Module API, merupakan kumpulan prosedur (listing program) yang digunakan untuk memanggil fungsi-fungsi windows Aplikasi Program Interface (API)
2. Module Main, merupakan kumpulan prosedur (listing program) fungsi utama dalam penyajian
3. Module Public, merupakan kumpulan prosedur (listing program) secara umum pada pemrograman Visual Basic.
4. Class Module Cregsetting, merupakan kumpulan prosedur (listing program) yang berfungsi membaca dan atau menulis data dari *registry* windows.

#### **3.5.4. Hal-hal yang Dapat Ditampilkan pada Sistem Informasi**

Dengan bantuan software Arc View, Sistem Informasi ini memberikan kemudahan dalam pengolahan data, beberapa informasi diperoleh dengan mengklik pada tempat-tempat yang diinginkan. Keterangan yang diperoleh dapat berupa atribut atau tabel dan gambar

obyek properti. Beberapa hal yang dapat ditampilkan pada sistem informasi pemasaran properti ini adalah :

1. Mengidentifikasi Feature Individual dengan Mouse

Informasi pada sebuah objek individual dalam peta dapat diperoleh dengan cara mengklik pada obyek individual tersebut. Informasi yang ditampilkan berdasarkan pada data atribut yang ada pada tabel.

2. Memperoleh Informasi Detail Properti dengan Mouse

Informasi detail properti dapat ditampilkan dengan cara mengklik pada obyek individual yang diinginkan. Informasi yang ditampilkan berupa foto obyek dan keterangan yang dimiliki obyek tersebut secara mendetail.

3. Memperoleh Informasi pada Feature Terpilih secara spasial.

Informasi feature terpilih secara spasial akan ditampilkan dalam bentuk simbol yang berbeda. Misalnya untuk mencari obyek rumah dengan status jual maka akan ditampilkan simbol berbeda pada simbol obyek properti yang ada, dalam hal ini ditampilkan dengan simbol lingkaran (*circle*) dengan pewarnaan yang lain sehingga dapat diketahui secara spasial hasil dari pencarian feature terpilih tersebut.

4. Mencari Informasi Bersyarat.

Mencari informasi dari obyek dapat dilakukan dengan syarat-syarat tertentu yang diberikan oleh *user* berdasarkan kriteria-kriteria yang telah ditentukan program, dalam hal ini berdasarkan status

pemasaran, tipe properti, harga obyek, nama jalan, dan nama kelurahan obyek properti berada. Hasil dari pencarian akan ditampilkan berupa tabel dan penunjukan secara spasial pada view peta. Misalnya , *user* ingin memperoleh informasi properti di jalan tertentu dengan harga sekian, maka akan ditampilkan properti-properti sesuai kriteria dalam bentuk tabel dan lokasinya.

#### 5. Menampilkan Lokasi dari Obyek Terpilih dalam View Peta

Sistem informasi ini dapat menampilkan lokasi dari data properti yang diinginkan, dalam bentuk penandaan simbol yang berkedip-kedip selama kurang lebih 5 detik.

#### 6. Menampilkan Data dalam Bentuk Tabel.

Sistem informasi ini dapat menampilkan data secara menyeluruh dalam bentuk tabel sekaligus dapat dilakukan pengerjaan *editing*.

#### 7. Melakukan Update Data

Update data dapat dilakukan untuk data spasial dalam hal ini *point* atau titik dari lokasi obyek properti sekaligus data non spasialnya.

#### 8. Menampilkan Peta pada View

Penampilan peta pada frame view yang telah disediakan oleh sistem informasi dapat dilakukan *per-layer (theme )* dan dapat juga ditampilkan semua *layer* secara bersama-sama.

### 3.5.5. Kemudahan dan Kekurangan Program

Setelah dilakukan tahap implementasi dan evaluasi sistem dapat disimpulkan adanya kemudahan dan kekurangan dari hasil program tersebut. Kemudahan dari program ini diantaranya adalah :

1. Desain tampilan dibuat lebih menarik secara visual yang mendukung aplikasi bisnis, dengan penempatan *menu* dan *tool* yang efektif dapat memudahkan pemakai (*marketing, broker* atau *klien*) dalam menjalankan dan mengeksplorasikan sistem informasi ini.
2. Data dapat ditampilkan secara menyeluruh dan secara detail tiap obyek properti dengan visualisasi menarik.
3. Proses pencarian data dan menampilkannya secara visual mudah dilakukan dengan berbagai cara.
4. Proses *editing* data dan *Up date* titik obyek dapat dilakukan, hal ini penting karena sistem informasi pemasaran membutuhkan data yang selalu berganti.

Sedangkan kekurangan atau kelemahan dari program ini adalah :

1. Kelemahan dari sebuah program adalah sering kali adanya kerusakan pada *software* sehingga sistem informasi sering *error* dan keluar dari sistem.
2. Kesalahan dalam pemasukan besarnya koordinat (koordinat keluar dari *frame view*) akan mengakibatkan tampilan *error*.

## BAB IV

### KESIMPULAN DAN SARAN

#### 4.1. Kesimpulan

Hasil dari penelitian pembuatan Sistem Informasi Pemasaran Perumahan Sekunder dengan memanfaatkan Sistem Informasi Geografis sebagai basis data spasial dan atribut adalah sebuah sistem informasi sebagai alat bantu presentasi pemasaran sekaligus untuk inventarisasi data. Dari hasil penelitian ini dapat ditarik beberapa kesimpulan :

1. Hasil dari sistem informasi pemasaran perumahan sekunder adalah berupa penyajian informasi spasial dan non spasial yang berhubungan dengan pemasaran properti. Dengan tampilan *design* yang menarik serta penempatan *menu* dan *tool* yang berdekatan sangat efektif dan memudahkan bagi seorang pengguna (*marketing, broker* maupun *klien*) dalam menjalankan program.
2. Sistem informasi ini mampu melakukan pencarian data secara cepat dan menampilkan hasil dalam bentuk tabel dan menyajikan lokasi obyeknya pada *frame view* dengan simbol dan warna yang berbeda.
3. Dengan adanya fasilitas penambahan *point* atau titik yang menunjukkan lokasi obyek dan *editing* pada data atributnya, maka *updating* data dapat dilaksanakan sehingga sistem informasi dapat digunakan berlanjut.

4. Beberapa fungsionalitas GIS yang dikelola dan dimanfaatkan sebagai fasilitas aplikasi meliputi :
  - ~Display Management, dengan fasilitas ini dapat dilihat distribusi obyek properti. *Tool*, semacam *zoom in*, *zoom out*, *info*, dapat dimanfaatkan untuk mengeksplorasikan informasi dibalik simbol-simbol peta tersebut.
  - ~Transaksi Data dan Pemutakhiran Obyek Peta, penambahan (*ploting*) obyek properti dapat dilakukan secara serentak dengan *database* tabularnya melalui *form entry* yang sudah didesain sedemikian rupa.
  - ~Map Query, Searching dan Filtering, mencari dan menyajikan obyek peta dengan kriteria tertentu dapat mudah dilakukan dengan mudah secara visual.
  - ~Peta Tematik, penyajian data dan informasi dalam bentuk peta tematik sehingga memudahkan dalam melihat dan menilai suatu sumberdaya melalui cara pandang spasial.
  - ~Reporting, seluruh aktifitas pengelolaan data dan informasi dapat dicetak dengan format baku sebagai bahan pelaporan untuk manajemen.
5. Fungsi dari Map Object disini adalah sebagai media untuk penyajian data spasial. Karena Map Object berfungsi sebagai alat untuk menjalankan *Universal Data Access (UDA)*, maka pemrograman Visual Basic dapat mengikuti standart *OLEDB* dalam membuat suatu aplikasi.
6. Fungsi Visual Basic dalam pembuatan sistem informasi ini adalah sebagai salah satu *development tools* dalam membangun aplikasi dalam lingkungan *windows*. Dengan adanya IDE (*Integrated Development*



*Environment*) Visual Basic, programmer dapat membuat *user interface*, melakukan koding, melakukan *testing*, dan *debuging* serta mengkompilasi program menjadi *executable*.

#### **4.2. Saran-saran**

1. Untuk menjaga agar basis data tetap terpelihara, perlu dirancang aplikasi khusus untuk pemutakhiran data, yang dikaitkan dengan aplikasi pelayanan masyarakat. Dengan demikian proses pemutakhiran data merupakan bagian yang tidak terpisahkan dari sistem itu sendiri.
2. Bagian terpenting dalam pembuatan sebuah sistem informasi adalah *desain interface* dari sebuah program. Dimana harus disesuaikan dengan maksud dan tujuan serta sasaran yang hendak dicapai dalam pembuatan sistem informasi.
3. Untuk pengembangan dari sistem informasi pemasaran perumahan ini perlu penambahan data spasial dan atribut khususnya data wilayah administrasi sesuai dengan perkembangan kota, sehingga informasi yang ditampilkan lebih komplet.
4. Pengembangan penggunaan *software* perlu dilakukan untuk penyempurnaan dari sistem informasi ini, misalnya dengan penggunaan audio dan video untuk menampilkan obyek propertinya.

## DAFTAR PUSTAKA

- Budiyanto E, 2002 **SIG Menggunakan ARC VIEW GIS**, Penerbit ANDI Yogyakarta.
- Daryanto, 2003, **Belajar komputer Visual Basic**, Penerbit Yrama Widya, Bandung.
- ERA. 2005, **WWW. eraindonesia.com**
- Handoyo Y.S. 1996, **Diktat Sistem Informasi Geografis**, Jurusan Teknik Geodesi FTSP ITN Malang.
- Idea. 2004, **IDEA, Majalah Ide Rumah Kita**, Edisi Juli 2004, Penerbit PT. Samindra Utama Gramedia, Jakarta
- Kurniawan B, 2002 **Sistem Informasi Manajemen dengan Visual Basic 6** Penerbit Andi, Yogyakarta.
- Mangkulo HE. 2004, **Pemrograman Visual Basic 6.0. dan Microsoft Access**, Penerbit Alex Media Komputindo, Jakarta.
- Prahasta E, 2001, **Konsep-konsep Dasar Sistem Informasi Geografis**, penerbit Informatika Bandung.
- Properti. 2004, **Tabloid Transaksi PROPERTI**, Edisi Juli 2004, Penerbit Panangian, Jakarta.
- Properti Era. 2004, **Buletin Bursa Properti Era**, Edisi September-Desember 2004, Penerbit ERA Indonesia.
- Sukmayadi D. 2002, **Prosiding Geo-Informasi, Atlas dan Pemetaan untuk Bisnis**, Copyright pada Bakosurtanal.
- Waluyo Cipto. 1997, **Prosiding Seminar SIG untuk Perkotaan dan Real Estat**, Copyright pada Lembaga Geomatika Perdana Internusa.
- Walijanto, 1998, **Sistem Basis Data ( Analisa dan Pemodelan Data )**, Penerbit Learning, Yogyakarta.
- Yousman Y, 2004 **SIG dengan MapInfo Professional**, Penerbit ANDI Yogyakarta

## **LAMPIRAN - LAMPIRAN**

- 1. Buku Petunjuk Penggunaan Program**
- 2. Listing Program**

## **Petunjuk Penggunaan Program**

### **Sistem Informasi Pemasaran Perumahan Sekunder**








#### **I. PENDAHULUAN**



Sistem informasi pemasaran perumahan sekunder ini dibuat berdasarkan aplikasi dari Sistem Informasi Geografis. Hasil dari program ini adalah berupa visualisasi peta tematik untuk informasi pemasaran perumahan sekunder di kota Malang dari sebuah broker properti yaitu Era Puri Properti. Peta tematik yang dihasilkan ini berupa peta posisi beserta basis datanya dari setiap properti yang dipasarkan. Ada 3 menu berisikan fasilitas-fasilitas yang telah disediakan dalam sistem informasi ini, diantaranya adalah :

1. **Menu Utama**, berisikan fasilitas untuk mengeksplorasikan data-data yang tersedia, fasilitas untuk mencetak hasil, fasilitas aplikasi untuk menghubungkan dengan program-program lain, menu help, dan menu untuk keluar dari sistem informasi ini.



Gambar 1. Menu Utama dalam Sistem Informasi Pemasaran.

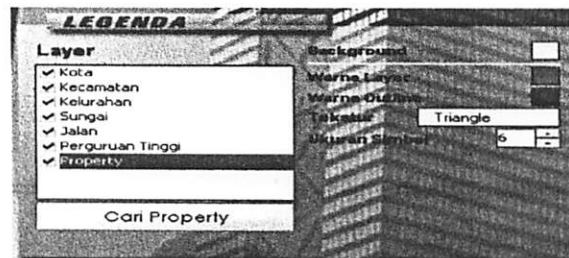
2. **Menu Map**, berisikan perintah-perintah yang dapat digunakan untuk mengeksplorasikan peta dan informasinya, berikut adalah *toolbar* yang tersedia dalam menu map beserta keterangannya.
  - ~ Zoom In  untuk memperbesar gambar tampilan pada view.
  - ~ Zoom Out  untuk memperkecil gambar tampilan pada view.
  - ~ Pan  untuk mengeser tampilan gambar pada view.
  - ~ Zoom Extent  untuk mengembalikan ke ukuran normal tampilan peta
  - ~ Identifikasi  untuk menampilkan informasi data individual.
  - ~ Pointer Normal  Penunjuk obyek aktif.
  - ~ Label  berfungsi untuk menampilkan label pada peta.

- ~ Add Point  untuk menambahkan point dari obyek properti yang dapat dilakukan dengan memasukan koordinat pada *form* yang tersedia (gambar 2.) atau langsung meng-klik ditempat yang diinginkan.
- ~ Pilih Point  untuk mengetahui besar koordinat khusus pada obyek properti.

Id Baru: 95	X: <input type="text"/>	Y: <input type="text"/>	<input type="checkbox"/> Tampilkan Data Property baru
-------------	-------------------------	-------------------------	---


Gambar 2. Tampilan *Form* untuk memasukan koordinat

3. **Legenda**, berisikan *layer* peta dan fasilitas pencarian properti serta faslitas *editing* untuk warna dan simbol tampilan peta.



Gambar 3. Tampilan Menu Legenda

## II. PENGGUNAAN PROGRAM



Untuk memulai penggunaan program sistem informasi ini dengan cara mengklik *icon* project property  secara otomatis akan ditampilkan sistem informasi pemasaran perumahan sekunder. Untuk memudahkan pengoperasian program ini, selanjutnya dalam buku petunjuk ini akan dijelaskan cara pengoperasiannya *perform*.

### A. Form Penyajian Peta Informasi Obyek Pemasaran Properti

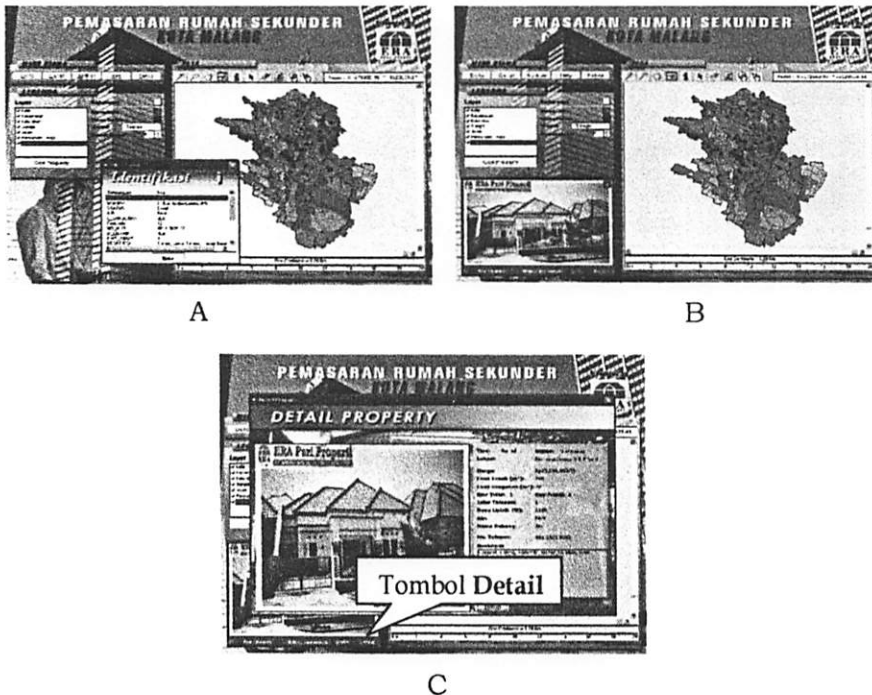
Peta informasi obyek pemasaran properti ini merupakan peta lokasi dari obyek rumah sekunder yang akan dipasarkan. Pada peta ini akan dapat diperoleh informasi penuh dari data-data obyek yang dipasarkan antara lain adalah lokasi obyek, tipe obyek, status pemasaran, luas bangunan, luas tanah, harga obyek, besarnya daya listrik, penggunaan daya air, jumlah kamar tidur dan kamar mandi,

nama penghubung (*broker*), dan fasilitas-fasilitas lainnya yang terdapat pada obyek properti tersebut yang disertai dengan gambar atau foto obyek.

Banyak cara untuk mendapatkan informasi obyek properti dalam peta ini, diantaranya adalah :

- ~ Klik icon  maka akan ditampilkan informasi data atribut dari feature obyek properti terpilih (gambar 4A.),
- ~ Jika menggunakan *pointer* normal  maka akan ditampilkan informasi individual dari feature obyek properti terpilih (gambar 4B.).
- ~ Untuk mendapatkan informasi penuh cukup dengan meng-klik tombol *Detail* pada tampilan informasi individual tersebut(gambar 4C.).
- ~ Untuk mendapatkan informasi data secara menyeluruh dengan meng-klik tombol *Data* pada tampilan informasi individual tersebut


Pada peta ini juga terdapat fasilitas untuk 'pencarian', *editing* data atribut dan *up date point*.



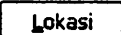



Gambar 4. Form Penyajian Informasi Peta

## B. Form Penyajian Data Atribut

Sistem informasi ini mempunyai fasilitas untuk menampilkan data-data secara menyeluruh dari tiap layer yang aktif. Karena program ini merupakan sistem informasi pemasaran perumahan sekunder, maka data tentang properti lebih mendetail. Data atribut obyek properti yang ditampilkan diantaranya adalah lokasi obyek, tipe obyek, status pemasaran, luas bangunan, luas tanah, harga obyek, besarnya daya listrik, penggunaan daya air, jumlah kamar tidur dan kamar mandi, nama penghubung (*broker*), dan fasilitas-fasilitas lainnya yang terdapat pada obyek properti tersebut yang disertai dengan gambar atau foto obyek.

Penggunaan *form* ini adalah dengan mengklik *icon* Data  maka akan ditampilkan data secara menyeluruh dari layer yang aktif. Dari tampilan data ada beberapa menu yang dapat digunakan antara lain :

-  : Untuk menyimpan data yang diedit atau dimasukan.
-  : Untuk menghapus data per-id
-  : Untuk menampilkan lokasi data pada layar
-  : Untuk mencari data sesuai yang diinginkan berdasarkan kriteria-kriteria yang telah tersedia

## C. Form Pencarian


Sistem informasi ini juga memberi kemampuan pencarian informasi atas obyek-obyek sesuai *theme* atau *layer* yang diinginkan. Ada 2(dua) cara untuk penggunaan *form* pencarian , yaitu :

1. Menggunakan menu **Cari Properti** yang terdapat pada menu **Legenda**.

Penggunaan *form* pencarian ini hanya dapat digunakan pada *theme* atau *layer* properti saja (gambar 5.). Berikut langkah penggunaanya :

- ~ Pilih theme **Properti**
- ~ Klik ikon **Cari Properti** 



- ~ Pilih kriteria – kriteria yang telah tersedia di *form* pencarian, banyak pilihan dari masing-masing kriteria, berdasarkan status pemasaran, tipe properti, dan harga properti yang diinginkan. Dilanjutkan meng-klik icon **Cari** 
- ~ Secara otomatis hasil data akan ditampilkan sesuai dengan kriteria diatas.
- ~ Masih dalam *form* pencarian, selanjutnya dapat dilakukan eksplorasi sesuai dengan menu yang tersedia.

: Untuk menampilkan informasi data individual secara detail dari obyek properti yang dipilih.


: Untuk menampilkan lokasi pada view.

: Untuk menampilkan basis data obyek properti

- ~ Untuk mengakhiri *form* pencarian klik pada tombol

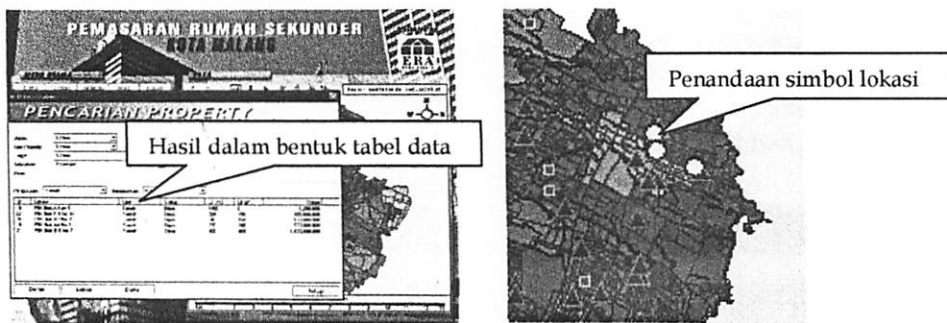
## 2. Menggunakan menu **Cari** yang terdapat pada *form* **Data**

Penggunaan menu **Cari** pada *form* **Data** ini dapat digunakan pada semua *layer* atau *theme*. Langkah penggunaannya seperti berikut :

- ~ Klik *theme* atau *layer* yang akan dicari informasinya
- ~ Pilih menu **Data** pada menu utama, kemudian akan ditampilkan *form* data.
- ~ Dari *form* data pilih dan klik ikon **Cari**
- ~ Sistem informasi akan memunculkan jendela *form* pencarian
- ~ Pilih kriteria – kriteria yang telah tersedia di *form* pencarian, banyak pilihan dari masing-masing kriteria, berdasarkan **status pemasaran, tipe properti, harga properti, Kelurahan** dan atau **Jalan** yang diinginkan. Dilanjutkan meng-klik icon **Cari** 
- ~ Secara otomatis hasil data akan ditampilkan sesuai dengan kriteria diatas dalam bentuk tabel dan penandaan posisi lokasi obyek dalam bentuk simbol dan pewarnaan berbeda (gambar 6.)
- ~ Masih dalam *form* pencarian, selanjutnya dapat dilakukan eksplorasi sesuai dengan menu yang tersedia.
- ~ Untuk mengakhiri *form* pencarian klik pada tombol
- ~ Untuk kembali ke tampilan utama klik tombol **Tutup** pada *form* data. ( gambar 7).



Gambar 5. Tampilan form Pencarian dari Menu Cari Properti



Gambar 6. Hasil Pencarian dalam bentuk tabel dan simbol lokasi




Gambar 7. Tampilan form Pencarian dari Menu Data

#### D. Form Updating

Fasilitas yang lain dalam sistem informasi ini adalah kemampuannya untuk meng-*update point* atau titik dimana lokasi obyek properti itu berada, hal ini penting karena sebagai sistem informasi pemasaran, datanya selalu *update* dan silih berganti. Fasilitas ini juga dapat dilakukan untuk mengisi data atribut baru.

Berikut adalah langkah untuk meng-*update point* obyek properti :

- ~ Pilih *theme* properti
- ~ Klik tombol **Tambah Point** 
- ~ Sistem Informasi akan jendela dialog seperti berikut :

Id Baru: 95	X: <input type="text"/>	Y: <input type="text"/>	<input type="checkbox"/> Tampilkan Data Property baru
-------------	-------------------------	-------------------------	---

- ~ Klik pada kotak tampilkan data baru, fungsinya adalah untuk menampilkan isian data properti baru.
- ~ Masukkan angka besaran koordinat X dan Y [tekan Enter].
- ~ Kemudian akan ditampilkan *form* data dan isi data-data tersebut.
- ~ Setelah isian data selesai lalu klik menu **Simpan** pada *form* data.
- ~ Untuk mengecek point yang telah dimasukkan klik menu **Lokasi**.
- ~ Untuk menutup *form* data tekan menu **Tutup**.
- ~ Langkah yang sama dapat dilakukan untuk menambah point properti selanjutnya.

#### **E. Fasilitas Lain**

Fasilitas lain disini adalah fasilitas tambahan untuk mendukung penggunaan sistem informasi ini diantara adalah :

1. **Menu Cetak**, dapat digunakan untuk mencetak data secara keseluruhan.
2. **Menu Aplikasi**, menu ini disediakan untuk membuka aplikasi lain tanpa harus menutup program dari sistem ini.
3. **Menu Help**, berisi tentang petunjuk penggunaan program ini.
4. **Menu Keluar**, untuk keluar atau mematikan program.

## Listing Program

### A. Form

#### 1. Form Data

```
Public objData As MyMapData
Public Mode As DataStates

Private Sub cmbStatus_Click()
    If cmbStatus.Text = "Disewakan" Then
        lblKet.Caption = "/Tahun"
    Else
        If cmbTipe.Text = "Tanah" Then
            lblKet.Caption = "/Meter"
        Else
            lblKet.Caption = ""
        End If
    End If
End Sub

Private Sub cmbTipe_Click()
    If cmbStatus.Text = "Disewakan" Then
        lblKet.Caption = "/Tahun"
    Else
        If cmbTipe.Text = "Tanah" Then
            lblKet.Caption = "/Meter"
        Else
            lblKet.Caption = ""
        End If
    End If
End Sub

Private Sub cmdCari_Click()
    If frmCariProperty.Visible Then
        Unload Me
        'Me.Show
        frmCariProperty.SetFocus
        'frmCariProperty.Show 1
    Else
        frmCariProperty.Show 1
    End If
End Sub

Private Sub cmdHapus_Click()
    If MsgBox("Anda yakin akan menghapus?", vbQuestion + vbYesNo, "Konfirmasi") = vbYes Then
        objData.Delete
    End If
End Sub

Private Sub cmdLokasi_Click()
    Hide
    Timer1.Enabled = True
End Sub

Private Sub cmdSimpan_Click()
    If Mode = dsEdit Then
        objData.Edit
        objData.SaveControlToDb
    Else
        objData.AddNew
        objData.SaveControlToDb
    End If
End Sub
```

```

Private Sub cmdTutup_Click()
    Unload Me
End Sub

Private Sub Form_Load()
Dim objCombo As New MyCombo
objCombo.FillToCombo Me.cmbKel, frmMain.Map1.Layers("Kelurahan").Records, "Nama_Kel"

Set objCombo = Nothing
Set objData = New MyMapData
objData.SetLayer frmMain.Map1.Layers("Property")
With objData
    .SetListView Me.LV
    .ImagesPath = App.Path & "\Gambar"
    .FillToListView "Id, Lokasi, Status, Harga, Tipe, L_Tanah, L_Bangunan", 0
    .AddBoundControl Me.txtId, "Id", False, False
    .AddBoundControl Me.txtLokasi, "Lokasi", False, False
    .AddBoundControl Me.cmbKel, "Nama_Kel", False, False
    .AddBoundControl Me.cmbStatus, "Status", False, False
    .AddBoundControl Me.txtHarga, "Harga", False, False
    .AddBoundControl Me.cmbTipe, "Tipe", False, False
    .AddBoundControl Me.txtL_Tanah, "L_TANAH", False, False
    .AddBoundControl Me.txtL_Bangunan, "L_BANGUNAN", False, False
    .AddBoundControl Me.txtK_Tidur, "Kmr_Tidur", False, False
    .AddBoundControl Me.txtK_Mandi, "Kmr_Mandi", False, False
    .AddBoundControl Me.txtJ_Telp, "J_Telp", False, False
    .AddBoundControl Me.txtDaya, "D_Listrik", False, False
    .AddBoundControl Me.cmbAir, "Air", False, False
    .AddBoundControl Me.txtDeskripsi, "Deskripsi", True, False
    .AddBoundControl Me.imgFoto, "Gambar", True, False
    .AddBoundControl Me.txtNama, "Nama_Hub", False, False
    .AddBoundControl Me.txtNo_Telp, "No_hub", False, False
    .SelectFromListView
End With
End Sub

Private Sub Form_Unload(Cancel As Integer)
    frmMain.ShowNewLabelPoint
    frmMain.cmdPeta_Click 8
    Set objData = Nothing
End Sub

Private Sub Image1_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    ' cek tombol mouse yang ditekan
    If Button = 1 Then ' 1 = tombol kiri
        Dim ReturnVal As Long
        X = ReleaseCapture()
        ' pindahkan form
        ReturnVal = SendMessage(hwnd, WM_NCLBUTTONDOWN, HTCAPTION, 0)
    End If
End Sub

Private Sub imgFoto_DblClick()
On Error GoTo ErrFoto

    dlgOpen.ShowOpen
    If FileExists(dlgOpen.FileName) Then
        imgFoto.Picture = LoadPicture(dlgOpen.FileName)
        imgFoto.Tag = dlgOpen.FileName
    End If
ErrFoto:

End Sub

Private Sub LV_ItemClick(ByVal Item As MSComctlLib.ListItem)

```

```

objData.SelectFromListView
If UCase(Item.SubItems(2)) = "DISEWAKAN" Then
    lblKet.Caption = "/Tahun"
Else
    If UCase(Item.SubItems(4)) = "TANAH" Then
        lblKet.Caption = "/Meter"
    Else
        lblKet.Caption = ""
    End If
End If
Mode = dsEdit
End Sub

Private Sub Timer1_Timer()
    frmMain.objLayer.FlashAndZoom objData.GetShape, 10, 20.5
    Timer1.Enabled = False
    Me.Show 1
End Sub

```

## 2. Form Identify

```

Option Explicit
Public Id As Variant

Public Sub cmdData_Click()
    Dim F As Form
    Unload Me
    Select Case frmMain.lstLayer.Text
        Case "Kota"
            Set F = frmDataKota
        Case "Kecamatan"
            Set F = frmDataKecamatan
        Case "Kelurahan"
            Set F = frmDataKelurahan
        Case "Sungai"
            Set F = frmDataSungai
        Case "Jalan"
            Set F = frmDataJalan
        Case "Perguruan Tinggi"
            Set F = frmDataPT
        Case "Property"
            Set F = frmDataProperty
    End Select
    If frmMain.lstLayer.Text <> "Kota" Then
        Load F
        F.objData.FindItemInListView Id, 0, 1, 0
        F.objData.SelectFromListView
    End If
    F.Show 1
    Set F = Nothing
End Sub

```

## 3. Form Detail Property

```

Public Sub ListDetail(ByVal Id As String)
    Dim rs As MapObjects2.Recordset
    Dim L As MapLayer
    Dim FN As String
    Dim Ket As String
    Set L = frmMain.Map1.Layers("Property")
    Set rs = L.SearchExpression("Id=" & Id)
    If rs.EOF Then
        Unload Me
    Exit Sub

```

```

End If
Me.lblTipe.Caption = rs.Fields("Tipe").ValueAsString
Me.lblLokasi.Caption = rs.Fields("Lokasi").ValueAsString
Me.lblStatus.Caption = rs.Fields("Status").ValueAsString
Me.lblHarga.Caption = Format(rs.Fields("Harga").ValueAsString, "Rp###,###,###,###")
Me.lbl_Tanah.Caption = rs.Fields("L_Tanah").ValueAsString
Me.lbl_Bangunan.Caption = rs.Fields("L_Bangunan").ValueAsString
Me.lblK_Tidur.Caption = rs.Fields("Kmr_Tidur").ValueAsString
Me.lblK_Mandi.Caption = rs.Fields("Kmr_Mandi").ValueAsString
Me.lblJ_Telp.Caption = rs.Fields("J_Telp").ValueAsString
Me.lblDaya.Caption = rs.Fields("D_Listrik").ValueAsString
Me.lblAir.Caption = rs.Fields("Air").ValueAsString
Me.lblNama_Hub.Caption = rs.Fields("Nama_Hub").ValueAsString
Me.lblTelp_Hub.Caption = rs.Fields("No_Hub").ValueAsString
Me.lblDesk.Caption = rs.Fields("Deskripsi").ValueAsString
Ket = ""
If UCase(Me.lblStatus.Caption) = "DISEWAKAN" Then
    Ket = "/Tahun"
Else
    If UCase(Me.lblTipe.Caption) = "TANAH" Then
        Ket = "/Meter"
    Else
        Ket = ""
    End If
End If
Me.lblHarga.Caption = Me.lblHarga.Caption + Ket
FN = App.Path & "\Gambar" & rs.Fields("Gambar").ValueAsString

'If file doesn't exist use empty images (none.jpg)
If Not (FileExists(FN)) Then FN = App.Path & "\Gambar\none.jpg"

If FileExists(FN) Then
    imgFoto.Picture = LoadPicture(FN)
Else
    imgFoto.Picture = Nothing
End If
Me.Show 1
End Sub

Private Sub cmdTutup_Click()
    Unload Me
End Sub

Private Sub Form_Resize()
    Me.Image1.Width = Me.ScaleWidth
    Me.Image1.Height = Me.ScaleHeight
End Sub

```

#### 4. Form Pencarian

Option Explicit

```

Private Sub RefreshEventSymbol()
    With frmMain.Map1.TrackingLayer.Symbol(1)
        .SymbolType = moPointSymbol
        .Style = frmMain.objLayer.Tekstur
        .Size = frmMain.objLayer.SymbolSize
        .Color = frmMain.objLayer.Color
    End With
End Sub
Private Sub AddColorEvent(shp As Object)
Dim newEvt As MapObjects2.EventRenderer
    ' add a new event
On Error Resume Next

```



```
Set newEvt = frmMain.Map1.TrackingLayer.AddEvent(shp, 2)
End Sub
```

```
Private Function StrHarga(ByVal Index As Integer)
Dim S As String
```

```
S = ""
Select Case Index
Case 1
S = "HARGA<=50000000"
Case 2
S = "HARGA>50000000 AND HARGA<=100000000"
Case 3
S = "HARGA>100000000 AND HARGA<=200000000"
Case 4
S = "HARGA>200000000 AND HARGA<=300000000"
Case 5
S = "HARGA>300000000 AND HARGA<=500000000"
Case 6
S = "HARGA>500000000 AND HARGA<=1000000000"
Case 7
S = "HARGA>1000000000 AND HARGA<=3000000000"
Case 8
S = "HARGA>3000000000 AND HARGA<=5000000000"
Case 9
S = "HARGA>5000000000"
End Select
StrHarga = S
End Function
```

```
Private Function OrderField() As String
Dim S As String
```

```
Select Case cmbKriteria.ListIndex
Case 0
S = "HARGA"
Case 1
S = "TIPE"
Case 2
S = "LOKASI"
Case 3
S = "L_TANAH"
Case 4
S = "L_BANGUNAN"
End Select
OrderField = S
End Function
```

```
Private Sub cmbKriteria_Click()
cmdCari_Click
End Sub
```

```
Private Sub cmbUrut_Click()
cmdCari_Click
End Sub
```

```
Private Sub cmdCari_Click()
Dim L As MapObjects2.MapLayer
Dim rs As MapObjects2.Recordset
Dim Exp As String
Dim li As ListItem
Dim shp As Object
```

```
Set L = frmMain.Map1.Layers("property")
Exp = ""
```

```

'Cek Combo Status
  If cmbStatus.ListIndex <> 0 Then
    Exp = " STATUS=" & (Me.cmbStatus.Text) & ""
  End If
'Cek Combo Tipe
  If cmbTipe.ListIndex <> 0 Then
    Exp = Exp & IIf(Exp <> "", " AND ", "") & " TIPE=" & (Me.cmbTipe.Text) & ""
  End If
'Cek Combo Harga
  If cmbHarga.ListIndex <> 0 Then
    Exp = Exp & IIf(Exp <> "", " AND ", " ") & StrHarga(Me.cmbHarga.ListIndex)
  End If
'Cek Combo Kel
  If cmbKel.ListIndex <> 0 Then
    Exp = Exp & IIf(Exp <> "", " AND ", " ") & " NAMA_KEL=" & Me.cmbKel.Text & ""
  End If
'Cek Jalan
  If txtJalan.Text <> "" Then
    Exp = Exp & IIf(Exp <> "", " AND ", " ") & " LOKASI like %" & Me.txtJalan.Text & "%"
  End If

  If Exp = "" Then
    Exp = "lokasi<>'xxxxxxxx'"
  End If
  Exp = Exp & " ORDER BY " & OrderField & " " & IIf(cmbUrut.ListIndex = 0, "ASC", "DESC")
  MsgBox Exp
  Set rs = L.SearchExpression(Exp)
  Me.LV1.ListItems.Clear
  frmMain.Map1.TrackingLayer.ClearEvents
  While Not rs.EOF
    Set shp = rs.Fields("Shape").Value
    AddColorEvent shp
    Set li = LV1.ListItems.Add(Text:=rs.Fields("Id").ValueAsString)
    li.SubItems(1) = rs.Fields("Lokasi").ValueAsString
    li.SubItems(2) = rs.Fields("Tipe").ValueAsString
    li.SubItems(3) = rs.Fields("Status").ValueAsString
    li.SubItems(4) = rs.Fields("L_TANAH").ValueAsString
    li.SubItems(5) = rs.Fields("L_BANGUNAN").ValueAsString
    li.SubItems(6) = Format(rs.Fields("Harga").ValueAsString, "###,###,###,###")
    rs.MoveNext
  Wend
  frmMain.Map1.Refresh

  Set L = Nothing
  Set rs = Nothing
  Set li = Nothing
End Sub

Private Sub cmdData_Click()
  If LV1.ListItems.Count = 0 Then Exit Sub
  If frmDataProperty.Visible = False Then Load frmDataProperty
  frmDataProperty.objData.FindItemInListView LV1.SelectedItem.Text, 0, 1, 0
  frmDataProperty.objData.SelectFromListView
  If frmDataProperty.Visible Then
    Unload Me
    frmDataProperty.SetFocus
  Else
    frmDataProperty.Show 1
  End If
End Sub

Private Sub cmdDetail_Click()
  frmPropertyDetail.ListDetail Me.LV1.SelectedItem.Text
End Sub

```

```

Private Sub cmdKeluar_Click()
    Unload Me
End Sub

Private Sub Form_Load()
    With frmMain.Map1.TrackingLayer.Symbol(2)
        .SymbolType = moPointSymbol
        .Style = frmMain.objLayer.SymbolType
        .Size = frmMain.objLayer.SymbolSize
        .Color = moYellow
    End With

    Dim objCombo As New MyCombo
    objCombo.FillToCombo Me.cmbKel, frmMain.Map1.Layers("Kelurahan").Records, "Nama_Kel"

    Set objCombo = Nothing
    Me.cmbStatus.ListIndex = 0
    Me.cmbTipe.ListIndex = 0
    Me.cmbHarga.ListIndex = 0
    Me.cmbUrut.ListIndex = 0
    Me.cmbKriteria.ListIndex = 0
    Me.cmbKel.ListIndex = 0
    Me.txtJalan.Text = ""
End Sub

Private Sub cmdLokasi_Click()
    If Me.LV1.SelectedItem Is Nothing Then Exit Sub
    Hide
    Timer1.Enabled = True
End Sub

Private Sub Timer1_Timer()
    Dim rs As MapObjects2.Recordset
    Dim L As MapLayer

    Set L = frmMain.Map1.Layers("property")
    Set rs = L.SearchExpression("ID=" & Me.LV1.SelectedItem.Text)
    If rs.EOF Then
        Timer1.Enabled = False
        Me.Show 1
    End If
    frmMain.objLayer.FlashAndZoom rs.Fields("shape").Value, 10, 20.5
    Timer1.Enabled = False
    Me.Show 1

    Set rs = Nothing
    Set L = Nothing
End Sub

```

## 5. Form Daftar Aplikasi

```

Option Explicit
Dim objData As CData

Private Sub cmdBrowse_Click()
    On Error GoTo Keluar
    dlgOpen.ShowOpen
    Me.lblLokasi.Caption = dlgOpen.FileName
Keluar:
End Sub

Private Sub cmdCancel_Click()
    ClearText
    objData.AddNew
    Me.txtNama_Aplikasi.SetFocus

```

End Sub

Private Sub cmdClose\_Click()

    Unload Me  
    frmMain.FillAppMenu

End Sub

Private Sub cmdDelete\_Click()

    If Trim(Me.txtNama\_Aplikasi.Text) = "" Or objData.Mode = dsNew Then Exit Sub  
    If MsgBox("Anda yakin mau menghapus data?", vbQuestion + vbYesNo, "Konfirmasi") = vbYes Then  
        objData.Delete Trim(Me.txtNama\_Aplikasi.Text)  
        If objData.UpdateSucceed Then  
            LV.ListItems.Remove LV.SelectedItem.Index  
        End If  
    End If

End Sub

Private Sub cmdNew\_Click()

    objData.AddNew  
    ClearText  
    Me.txtNama\_Aplikasi.SetFocus

End Sub

Private Sub ClearText()

    Me.txtNama\_Aplikasi.Text = ""  
    Me.lblLokasi.Caption = ""

End Sub

Private Sub cmdSave\_Click()

Dim li As ListItem

    If Trim(Me.txtNama\_Aplikasi.Text) = "" Then  
        MsgBox "Nama Aplikasi Belum Terisi", vbExclamation, "Peringatan"  
        Me.txtNama\_Aplikasi.SetFocus  
        Exit Sub  
    End If

    If Trim(Me.lblLokasi.Caption) = "" Then  
        MsgBox "Lokasi Belum Terisi", vbExclamation, "Peringatan"  
        Me.cmdBrowse.SetFocus  
        Exit Sub  
    End If

    If objData.Mode = dsEdit Then

        With objData  
            .AddUpdField "Nama\_Aplikasi", Me.txtNama\_Aplikasi.Text  
            .AddUpdField "lokasi", Me.lblLokasi.Caption  
            .UpdateRecord  
            If Not .UpdateSucceed Then Exit Sub  
        End With

        Set li = LV.SelectedItem  
        li.Text = Me.txtNama\_Aplikasi.Text  
        li.SubItems(1) = Me.lblLokasi.Caption

    Else

        With objData  
            .AddInsField Me.txtNama\_Aplikasi.Text  
            .AddInsField Me.lblLokasi.Caption  
            .UpdateRecord  
            If Not .UpdateSucceed Then Exit Sub  
        End With

        Set li = LV.ListItems.Add()  
        li.Text = Me.txtNama\_Aplikasi.Text  
        li.SubItems(1) = Me.lblLokasi.Caption

    End If

```

ClearText
objData.AddNew
Me.txtNama_Aplikasi.SetFocus
End Sub

```

```

Private Sub Form_Load()
Set objData = New CData

objData.Connection = DE1.Conn1
objData.TableName = "Aplikasi"
objData.FillToListView Me.LV, "Nama_Aplikasi, Lokasi"
objData.AddNew
ClearText
End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)
Set objData = Nothing
End Sub

```

```

Private Sub LV_ItemClick(ByVal Item As MSCComctlLib.ListItem)
Dim rs As ADODB.Recordset

Me.txtNama_Aplikasi.Text = Item.Text
Me.lblLokasi.Caption = Item.SubItems(1)
objData.Edit
objData.EditId = Item.Text
End Sub

```

```

Private Sub txtNama_Aplikasi_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
SendKeys "{Tab}"
End If
End Sub

```

```

Sub Find(ByVal Id As String)
Dim li As ListItem

Set li = Me.LV.FindItem(Id, 0, , 0)
If Not (li Is Nothing) Then
Set LV.SelectedItem = li
LV_ItemClick li
End If
End Sub

```

## 6. Form AddPoint

```

Option Explicit
Dim foundEvt As MapObjects2.GeoEvent
Private Function Max() As Long
Dim Recs As MapObjects2.Recordset
Dim oStat As MapObjects2.Statistics

Set Recs = frmMain.Map1.Layers("Property").Records
Set oStat = Recs.CalculateStatistics("id")
Max = oStat.Max
End Function

```

```

Private Sub Reindex()
Dim IMax, i As Long
Dim Evt As MapObjects2.GeoEvent
Screen.MousePointer = vbHourglass
IMax = Max + lstId.ListCount
For i = 0 To lstId.ListCount - 1
Set Evt = frmMain.Map1.TrackingLayer.FindEvent(Me.lstId.List(i))
Me.lstId.List(i) = IMax

```

```
    Evt.Tag = IMax
    IMax = IMax - 1
Next
Screen.MousePointer = vbDefault
End Sub
```

```
Private Sub cmdBatal_Click()
    Unload Me
End Sub
```

```
Private Sub cmdHapus_Click()
    If Not (foundEvt Is Nothing) Then
        frmMain.Map1.TrackingLayer.RemoveEvent foundEvt.Index
        Me.IstId.RemoveItem Me.IstId.ListIndex
        Reindex
    End If
End Sub
```

```
Private Sub cmdSimpan_Click()
    Dim i As Long
    Dim Evt As MapObjects2.GeoEvent
    Dim rs As MapObjects2.Recordset
```

```
    If Me.IstId.ListCount = 0 Then Exit Sub
    Set rs = frmMain.Map1.Layers("Property").Records
    If Not rs.Updatable Then
        MsgBox "Tidak dapat menyimpan point, file sedang dibuka aplikasi lain!", vbExclamation, "Peringatan"
        Exit Sub
    End If
    For i = 0 To Me.IstId.ListCount - 1
        Set Evt = frmMain.Map1.TrackingLayer.FindEvent(IstId.List(i))
        rs.AddNew
        rs.Fields("Shape").Value = Evt.Shape
        rs.Fields("Id").Value = Evt.Tag
        rs.Update
    Next
End Sub
```

```
Private Sub Form_Load()
    'For new points
    Me.IstId.Clear
End Sub
```

```
Private Sub IstId_Click()
    'Find the GeoEvent with the selected Tag
    If IstId.ListIndex < 0 Then Exit Sub
```

```
Private Sub txtX_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then
        UpdateLocation
    Else
        ToNumberOnly KeyAscii
    End If
End Sub
```

```
Private Sub txtY_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then
        UpdateLocation
    Else
        ToNumberOnly KeyAscii
    End If
End Sub
```

## 7. Form Main

```
Option Explicit
Public objMap As New MyMap
Public objLayers As New MyLayers
Public objLayer As New MyLayer
Public objData As New MyMapData
'Public objWarnaObyek As New CWarnaObyek

Dim DontResponse As Boolean
Dim rLabel As LabelRenderer
Dim Id As String

'Variable used for adding some points
Dim pt As New MapObjects2.Point
Dim newEvt As New MapObjects2.GeoEvent
Dim gEvt As New MapObjects2.GeoEvent
Dim maxdist As Double
Dim nEventCount As Integer
Dim iIndex As Integer
Dim PointFound As Point
Dim rsFound As MapObjects2.Recordset

'----- User Defined Procedure -----
Private Sub RefreshEventSymbol()
    With frmMain.Map1.TrackingLayer.Symbol(1)
        .SymbolType = moPointSymbol
        .Style = frmMain.objLayer.Tekstur
        .Size = frmMain.objLayer.SymbolSize
        .Color = frmMain.objLayer.Color
    End With
End Sub
Private Function AutoNumber() As Long
Dim Recs As MapObjects2.Recordset
Dim oStat As MapObjects2.Statistics

    Set Recs = Map1.Layers("Property").Records
    Set oStat = Recs.CalculateStatistics("id")
    AutoNumber = oStat.Max + 1
End Function

Public Sub ShowNewLabelPoint()
    lblIdPoint.Caption = AutoNumber
End Sub
Private Sub AddEvent(ByVal X As Double, ByVal Y As Double, ByVal Manual As Boolean)
Dim rs As MapObjects2.Recordset
Dim pt As Point

    If Not (Manual) Then
        If Me.ScaleMode <> vbTwips Then
            X = Me.ScaleX(X, vbTwips, Me.ScaleMode)
            Y = Me.ScaleY(Y, vbTwips, Me.ScaleMode)
        End If

        ' convert the point to map coordinates
        Set pt = Map1.ToMapPoint(X, Y)
    Else
        'Set pt = New Point
        If Not (rsFound Is Nothing) Then
            PointFound.X = X
            PointFound.Y = Y
            If rsFound.Updatable Then
                rsFound.Edit
                rsFound.Fields("shape") = PointFound
                rsFound.Update
            End If
        End If
    End If
End Sub
```

```

        rsFound.StopEditing
        Me.txtX.Text = ""
        Me.txtY.Text = ""
        Me.lblIdPoint.Caption = AutoNumber
        cmdPeta_Click 8
        Exit Sub
    End If
Else
    MsgBox "Point belum terpilih", vbExclamation, "Peringatan"
    Exit Sub
    Set pt = New Point
    pt.X = X
    pt.Y = Y
End If
End If
' add a new event

Set newEvt = Map1.TrackingLayer.AddEvent(pt, 1)
newEvt.Tag = AutoNumber
lblIdPoint.Caption = AutoNumber
frmAddPoint.lstId.AddItem newEvt.Tag, 0
Set rs = Me.Map1.Layers("Property").Records

If rs.Updatable Then
    rs.AddNew
    rs.Fields("id").Value = newEvt.Tag
    rs.Fields("Shape").Value = pt
    rs.Update
    rs.StopEditing
Else
    Map1.TrackingLayer.RemoveEvent newEvt.Index
    MsgBox "Penambahan data gagal, file masih digunakan", vbExclamation, "Peringatan"
    Exit Sub
End If
Me.txtX.Text = ""
Me.txtY.Text = ""
If chkTampilProperty.Value = vbUnchecked Then Exit Sub
Load frmDataProperty
frmDataProperty.objData.FindItemInListView newEvt.Tag, 0, 1, 0
frmDataProperty.objData.SelectFromListView
frmDataProperty.Show 1
End Sub

Sub UnLoadAppMenu()
Dim i As Byte
For i = 0 To mnuApp.UBound
    If i <> 0 Then
        Unload mnuApp(i)
    Else
        mnuApp(i).Visible = False
    End If
Next
End Sub

Sub FillAppMenu()
Dim strSQL As String
Dim rs As ADODB.Recordset
Dim i As Byte

UnLoadAppMenu
strSQL = "SELECT * FROM Aplikasi ORDER BY Nama_Aplikasi"
Set rs = DE1.Conn1.Execute(strSQL)
i = 0
While Not rs.EOF
    If i <> 0 Then

```



```
    Load mnuApp(i)
Else
    mnuApp(i).Visible = True
End If
```

```
    mnuApp(i).Caption = rs("Nama_Aplikasi")
    mnuApp(i).Tag = rs("Nama_Aplikasi")
    i = i + 1
    rs.MoveNext
Wend
```

```
    sp.Visible = (rs.RecordCount > 0)
End Sub
```

```
Public Sub OpenCrystalReport(crt As CrystalReport, ByVal FileName As String)
    crt.ReportFileName = FileName
    crt.RetrieveDataFiles
    crt.Action = 1
End Sub
```

```
Sub SetOriginalSetting(ByVal Id As Byte)
Dim Refresh As Boolean
```

```
    DontResponse = True
    Refresh = False
    Map1.TrackingLayer.ClearEvents
    objLayer.LayerID = lstLayer.ItemData(Id)
    Set Me.Map1.Layers(objLayer.LayerName).Renderer = Nothing
    If Me.objLayer.ColorType = twMultiColor Then
        Me.picWarnaLayer.Picture = LoadPicture(App.Path & "\Pictures\TipeWarna.jpg")
        Me.objLayer.ShowMultiColor
    Else
        Set Me.picWarnaLayer.Picture = Nothing
        Me.picWarnaLayer.BackColor = objLayer.Color
        With Me.Map1.Layers(objLayer.LayerName).Symbol
            .SymbolType = objLayer.SymbolType
            .Color = objLayer.Color
            If objLayer.shapeType <> moShapeTypeLine Then .OutlineColor = objLayer.Outline
            .Style = objLayer.Tekstur
        End With
    End If
    Me.picOutline.BackColor = objLayer.Outline
    objLayer.ListSymbolType Me.cmbTekstur
    Me.cmbTekstur.ListIndex = objLayer.Tekstur
    Me.txtSymbolSize.Text = objLayer.SymbolSize
```

```
    If Me.Map1.Layers(objLayer.LayerName).Visible <> Me.lstLayer.Selected(Me.lstLayer.ListIndex) Then Refresh = True
```

```
    Me.Map1.Layers(objLayer.LayerName).Visible = Me.lstLayer.Selected(Me.lstLayer.ListIndex)
    ' If lstLayer.ListIndex = 0 Then objWarnaObyek.ApplyColor
```

```
    Set cmdPeta(6).PictureNormal = LoadPicture(App.Path & "\Pictures\label.bmp")
    cmdPeta(6).Tag = 0
```

```
    If Refresh Then
        Map1.Refresh
    End If
    DontResponse = False
End Sub
```

```
Private Sub Command1_Click()
    MsgBox VBNumber(Me.txtX.Text)
End Sub
```

```

Private Sub cmdCariProperty_Click()
    frmCariProperty.Show 1
End Sub

Private Sub cmdClose_Click()
    Me.picShow.Visible = False
End Sub

Private Sub cmdData_Click()
    If frmDataProperty.Visible = False Then Load frmDataProperty
    frmDataProperty.objData.FindItemInListView lblId.Caption, 0, 1, 0
    frmDataProperty.objData.SelectFromListView
    If frmDataProperty.Visible Then
        Unload Me
        frmDataProperty.SetFocus
    Else
        frmDataProperty.Show 1
    End If
End Sub

Private Sub cmdDetail_Click()
    frmPropertyDetail.ListDetail lblId.Caption
End Sub
'-----

```

## B. Modules

### 1. mdlAPI

Option Explicit

'Deklarasi Memindah Form  
 Public Const HTCAPTION = 2  
 Public Const WM\_NCLBUTTONDOWN = &HA1

Public Declare Function ReleaseCapture Lib "user32" () As Long  
 Public Declare Function SendMessage Lib "user32" Alias "SendMessageA" (ByVal hwnd As Long, ByVal wParam As Long, ByVal wParam As Long, lParam As Long) As Long

'Deklarasi ShowOnTop  
 Public Const SWP\_NOMOVE = 2  
 Public Const SWP\_NOSIZE = 1  
 Public Const FLAGS = SWP\_NOMOVE Or SWP\_NOSIZE  
 Public Const HWND\_TOPMOST = -1  
 Public Const HWND\_NOTOPMOST = -2

Declare Function SetWindowPos Lib "user32" \_  
 (ByVal hwnd As Long, \_  
 ByVal hWndInsertAfter As Long, \_  
 ByVal X As Long, \_  
 ByVal Y As Long, \_  
 ByVal cx As Long, \_  
 ByVal cy As Long, \_  
 ByVal wFlags As Long) As Long

'Deklarasi Execute Task  
 Private Type STARTUPINFO  
 cb As Long  
 lpReserved As String  
 lpDesktop As String  
 lpTitle As String  
 dwX As Long  
 dwY As Long  
 dwXSize As Long  
 dwYSize As Long

```

dwXCountChars As Long
dwYCountChars As Long
dwFillAttribute As Long
dwFlags As Long
wShowWindow As Integer
cbReserved2 As Integer
lpReserved2 As Long
hStdInput As Long
hStdOutput As Long
hStdError As Long
End Type

```

```

Private Type PROCESS_INFORMATION
    hProcess As Long
    hThread As Long
    dwProcessID As Long
    dwThreadId As Long
End Type

```

```

Private Declare Function WaitForSingleObject Lib "kernel32.dll" (ByVal hHandle As Long, ByVal dwMilliseconds As Long) As Long
Private Declare Function CloseHandle Lib "kernel32.dll" (ByVal hObject As Long) As Long
Private Declare Function CreateProcessA Lib "kernel32.dll" (ByVal lpApplicationName As Long, ByVal lpCommandLine As String, ByVal lpProcessAttributes As Long, ByVal lpThreadAttributes As Long, ByVal bInheritHandles As Long, ByVal dwCreationFlags As Long, ByVal lpEnvironment As Long, ByVal lpCurrentDirectory As Long, lpStartupInfo As STARTUPINFO, lpProcessInformation As PROCESS_INFORMATION) As Long

```

```

Private Const NORMAL_PRIORITY_CLASS = &H20&
Private Const INFINITE = -1&

```

```

'----- Show On Top -----

```

```

Public Function SetTopMostWindow(hwnd As Long, Topmost As Boolean) _
    As Long

```

```

    If Topmost = True Then 'Make the window topmost
        SetTopMostWindow = SetWindowPos(hwnd, HWND_TOPMOST, 0, 0, 0, 0, FLAGS)
    Else
        SetTopMostWindow = SetWindowPos(hwnd, HWND_NOTOPMOST, 0, 0, 0, 0, FLAGS)
        SetTopMostWindow = False
    End If

```

```

End Function

```

```

'----- Execute Task -----

```

```

Public Function ExecuteTask(Path As String, Optional Parameters As String = "") As Long

```

```

    Dim proc As PROCESS_INFORMATION
    Dim startup As STARTUPINFO
    Dim ret As Long
    Dim cmdline As String

```

```

    cmdline = Path
    If Trim(Parameters) <> "" Then
        cmdline = cmdline & " " & Parameters & ""
    End If

```

```

    ' init struktur STARTUPINFO
    startup.cb = Len(startup)
    ' memulai aplikasi
    ret = CreateProcessA(0&, cmdline, 0&, 0&, 1&, NORMAL_PRIORITY_CLASS, 0&, 0&, startup, proc)
    ExecuteTask = proc.hProcess

```

End Function

```
Public Function IsTaskRunning(taskID As Long) As Boolean
    Dim ret As Long
```

```
    If taskID = 0 Then Exit Function
    ret = WaitForSingleObject(taskID, 0)
    IsTaskRunning = (ret <> 0)
    If Not IsTaskRunning Then
        ret = CloseHandle(taskID)
        taskID = 0
```

```
    End If
End Function
```

## 2. mdlMain

Option Explicit

```
Public DC As New DataConnection
Public DBFileName, DataPath, ImagesPath As String
Public Const SupportedPicture = "JPG, BMP"
Public RegSetting As cRegSetting
```

```
Sub Main()
    Dim StrConn As String
```

```
    LoadSetting
    StrConn = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & DBFileName & ";Persist Security Info=False "
    DE1.Conn1.Open StrConn
    If DE1.Conn1.State = adStateClosed Then
        MsgBox "Koneksi Gagal", vbCritical, "Error"
    End If
    End If
    'Form1.Show
    'frmLogin.Show 1
    frmMain.Show
```

```
End Sub
```

```
Sub LoadRegSetting()
    Dim i As Integer
```

```
    With frmMain
        .picBackground.BackColor = RegSetting.MapBackColor
        .Map1.BackColor = .picBackground.BackColor
    End With
```

```
End Sub
```

```
Sub LoadSetting()
```

```
'Inisialisasi variabel global
    DBFileName = App.Path + "Atribut\Database.mdb"
    DataPath = App.Path + "\Spasial"
    ImagesPath = App.Path + "\Gambar"
```

```
    DC.Database = DataPath
    If Not DC.Connect Then
        MsgBox "Data Peta tidak ditemukan", vbCritical, "Error"
    End If
End If
```

```
End Sub
```

```
'Sub CloseDB()
'    DE1.Conn1.Close
'End Sub
```

### 3. mdiPublic

Option Explicit

Function FileExists(fileName As String) As Boolean

FileExists = False

If Trim(fileName) = "" Then Exit Function

If Dir(fileName) = "" Then Exit Function

If GetAttr(fileName) = vbDirectory Then Exit Function

FileExists = True

End Function

Public Sub ToUpperCase(ByRef KeyAscii As Integer)

KeyAscii = Asc(UCase(Chr(KeyAscii)))

End Sub

Public Sub ToNumberOnly(ByRef KeyAscii As Integer)

If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii = Asc(".") Or KeyAscii = Asc(",") Or KeyAscii = vbKeyBack) Then

KeyAscii = 0

End If

End Sub

Public Sub SelectText(obj As Object)

obj.SelStart = 0

obj.SelLength = Len(obj.Text)

obj.SetFocus

End Sub

Public Function MaxSearchConfirm(ByVal N As Long) As VbMsgBoxResult

MaxSearchConfirm = MsgBox("Pencarian kurang spesifik , ditemukan " & N & " record" & \_  
Chr(13) & "apakah anda mau meneruskan", vbInformation + vbYesNo, "Informasi")

End Function

Public Function EmptyToMin(ByVal S As String) As String

If Trim(S) = "" Then

EmptyToMin = "-"

Else

EmptyToMin = Trim(S)

End If

End Function

Public Function NullToEmpty(ByVal V As Variant)

NullToEmpty = If(IsNull(V), "", V)

End Function

Function RandomRange(ByVal Upperbound As Long, ByVal Lowerbound As Long) As Long

Randomize

RandomRange = Int((Upperbound - Lowerbound + 1) \* Rnd + Lowerbound)

End Function

Function RandomColor() As Long

RandomColor = RGB(RandomRange(100, 255), RandomRange(100, 250), RandomRange(0, 100))

End Function

Function ExtractFileName(ByVal FileName As String, Optional ByVal ShowExt As Boolean)

FileName = StrReverse(FileName)

FileName = StrReverse(Left(FileName, If(InStr(FileName, "\") = 0, Len(FileName), InStr(FileName, "\") - 1)))

If (Not ShowExt) And (InStr(FileName, ".") > 0) Then

FileName = Replace(FileName, "." & ExtractFileExt(FileName), "")

End If

ExtractFileName = FileName

End Function

Function ExtractFileExt(ByVal FileName As String) As String

```

    FileName = StrReverse(FileName)
    If InStr(FileName, ".") > 0 Then
        ExtractFileExt = StrReverse(Left(FileName, InStr(FileName, ".") - 1))
    Else
        ExtractFileExt = ""
    End If
End Function

Function IsLetter(ByVal S As String) As Boolean
    IsLetter = InStr("ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz", S) > 0
End Function

Function IsDrive(ByVal F As String) As Boolean
    Dim DLetter As String
    IsDrive = (Len(F) = 3) And (InStr(F, ":") > 0) And IsLetter(Left(F, 1))
End Function

Function QStr(ByVal S As String) As String
    QStr = "" & S & ""
End Function

Function VBNumber(pValue As String) As Double
    VBNumber = Val(Replace(pValue, ",", ".", , , vbTextCompare))
End Function

```

### C. Class Modules

Option Explicit

```
Private Const APP_NAME = "PROPERTY v1.0"
```

'Warna Default

```
Private Const D_MAP_BACKCOLOR = &H80000005
```

```
Dim p_Background As Long 'Warna Background Peta
```

```
Public Sub CreateRegSetting()
```

```
    SaveSetting APP_NAME, "SettingPeta", "MapBackColor", "&H80000005&"
```

```
    SaveSetting APP_NAME, "SettingPeta", "SimpanOtomatis", "0"
```

```
End Sub
```

```
Public Sub DeleteRegSetting()
```

```
    DeleteSetting APP_NAME
```

```
End Sub
```

```
Property Get MapBackColor() As Long
```

```
Dim S As Variant
```

```
On Error GoTo errHandler
```

```
    S = GetSetting(APP_NAME, "SettingPeta", "MapBackColor", Default:="&H80000005&")
```

```
    MapBackColor = CLng(S)
```

```
On Error GoTo 0
```

```
Exit Property
```

```
errHandler:
```

```
    MapBackColor = vbWhite
```

```
End Property
```

```
Property Let MapBackColor(ByVal Color As Long)
```

```
Dim S As Variant
```

```
On Error GoTo errHandler
```

```
    SaveSetting APP_NAME, "SettingPeta", "MapBackColor", Str(Color)
```

```
On Error GoTo 0
```

```
Exit Property
```

```
errHandler:
  MsgBox "Registry Error", vbCritical, "Error"
End Property

Property Get SimpanOtomatis() As Byte
Dim S As Variant

On Error Resume Next
  S = GetSetting(APP_NAME, "SettingPeta", "SimpanOtomatis", Default:=0)
  SimpanOtomatis = CByte(S)
On Error GoTo 0
End Property

Property Let SimpanOtomatis(ByVal N As Byte)
Dim S As Variant
On Error GoTo errHandler
  SaveSetting APP_NAME, "SettingPeta", "SimpanOtomatis", Str(N)

On Error GoTo 0
Exit Property
errHandler:
  MsgBox "Registry Error", vbCritical, "Error"
End Property
```


**INSTITUT TEKNOLOGI NASIONAL**  
**Jl. Bendungan Sigura-gura No. 02**  
**MALANG**

**Nama : Moh. Abdullah**  
**Nim : 9525059**  
**Dosen Pembimbing I : Ir. Leo Pantimena, Msc.**


**LEMBAR ASSISTENSI**

**Laporan Tugas Akhir**

**Aplikasi Sistem Informasi Geografi untuk Penyajian Informasi**  
**Pemasaran Perumahan Sekunder di Kota Malang**  
**Studi Kasus PT ERA Puri Broker Properti**

<b>NO.</b>	<b>TANGGAL</b>	<b>CATATAN / KETERANGAN</b>	<b>TANDA TANGAN DOSEN PEMBIMBING</b>
	04/05 2014	<ul style="list-style-type: none"><li>- perbaiki penulisan TA</li><li>- daya tarik di perbaiki</li><li>- pembaharuan . lebih detail lagi</li><li>- Kerjasama dan Guru Kiperbil</li><li>- Daftar perantara Klub order</li></ul>	



NO.	TANGGAL	CATATAN / KETERANGAN	TANDA TANGAN DOSEN PEMBIMBING
		<ul style="list-style-type: none"> <li>- Buat buku agenda ora pengoperasian sopkator</li> <li>- tulis dan susun bentuk yg baik dibuat</li> <li>- berikan kesan sopan</li> <li>- Buat ringkasan untuk semua</li> <li>- siapkan buku pembimbing //</li> </ul>	


**INSTITUT TEKNOLOGI NASIONAL**  
**Jl. Bendungan Sigura-gura No. 02**  
**MALANG**

Nama : Moh. Abdullah  
Nim : 9525059  
Dosen Pembimbing I : Ir. Leo Pantimena, Msc.

**LEMBAR ASSISTENSI**

**Makalah Seminar Hasil**

**Aplikasi Sistem Informasi Geografi untuk Penyajian Informasi**  
**Pemasaran Perumahan Sekunder di Kota Malang**  
Studi Kasus PT ERA Puri Broker Properti

<b>NO.</b>	<b>TANGGAL</b>	<b>CATATAN / KETERANGAN</b>	<b>TANDA TANGAN DOSEN PEMBIMBING</b>
	21/06 <sup>05</sup>	Acc seminar hasil.	



INSTITUT TEKNOLOGI NASIONAL  
Jl. Bendungan Sigura-gura No. 02  
MALANG

Nama : Moh. Abdullah  
Nim : 9525059  
Dosen Pembimbing II : Ir. Agus Darpono, MT.

LEMBAR ASSISTENSI

Laporan Tugas Akhir

Aplikasi Sistem Informasi Geografi untuk Penyajian Informasi  
Pemasaran Perumahan Sekunder di Kota Malang  
Studi Kasus PT ERA Puri Broker Properti

NO.	TANGGAL	CATATAN / KETERANGAN	TANDA TANGAN DOSEN PEMBIMBING
	26-3-05	<p>Tun - pustaha. new bab A</p> <p>Bab B. hasil &amp; subkon blm tertulis. yg perlu banya pengumpul pihabara progru. tulsi. kemudaha progru: apa!</p> <p>Dipuler blm dner dy tujien.</p>	
	29-3-05	<p>Revisi dan PBI</p>	



**INSTITUT TEKNOLOGI NASIONAL**  
**Jl. Bendungan Sigura-gura No. 02**  
**MALANG**

**SEMINAR HASIL SKRIPSI JENJANG STRATA I (S1)**  
**JURUSAN TEKNIK GEODESI**  
**FAKULTAS TEKNIK SIPIL DAN PERENCANAAN**

Nama : Moh. Abdullah  
N I M : 9525059  
Dosen Perevisi : Ir. Rinto Sasongko, MT.

**LEMBAR ASSISTENSI**

Revisi Tugas Akhir

No	Tanggal	Catatan/Keterangan	Tanda Tangan
1.	14/10 '05	Sistemnya penuh sudah diselesaikan menjadi 5 bab - Dnsr / INTB belum dimasukkan kem. program	
2.	18/10 '05	Dnsr sdh. dimasukkan sbg. input data 	



**INSTITUT TEKNOLOGI NASIONAL**  
**Jl. Bendungan Sigura-gura No. 02**  
**MALANG**

**SEMINAR HASIL SKRIPSI JENJANG STRATA I (S1)**  
**JURUSAN TEKNIK GEODESI**  
**FAKULTAS TEKNIK SIPIL DAN PERENCANAAN**

Nama : Moh. Abdullah  
N I M : 9525059  
Dosen Perevisi : Ir. M. Ruslin Anwar, MS.

**LEMBAR ASSISTENSI**

Revisi Tugas Akhir

No	Tanggal	Catatan/Keterangan	Tanda Tangan
	18/05/20	- Revisi - ace - Nila Ace	