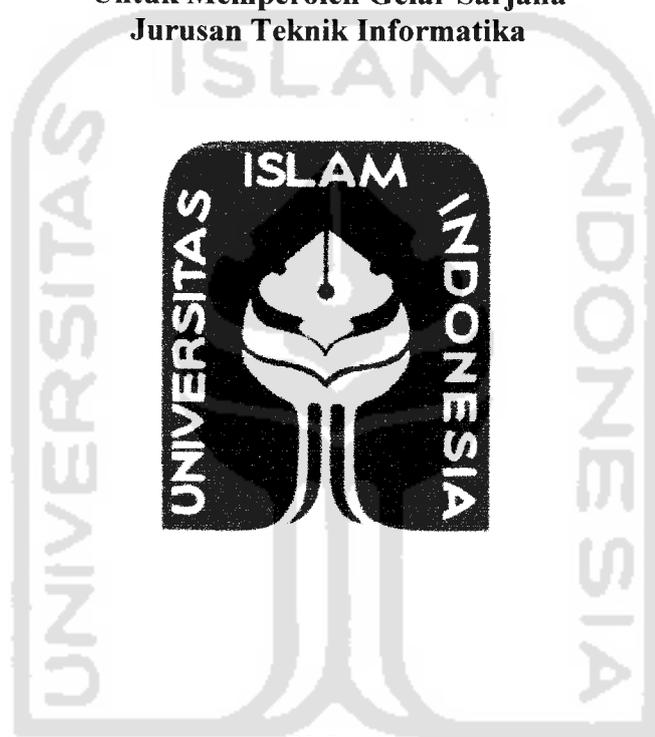


**ALAT BANTU AJAR OPERASI–OPERASI DASAR
GRAFIS**

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana
Jurusan Teknik Informatika**



Oleh :

Nama : David Syahril

No. Mahasiswa : 99 523 237

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA
2007**

LEMBAR PERNYATAAN KEASLIAN
HASIL TUGAS AKHIR

Saya yang bertanda tangan di bawah ini,

Nama : David Syahril

No. Mahasiswa : 99 523 237

Menyatakan bahwa seluruh komponen dan isi dalam Tugas Akhir ini adalah hasil karya saya sendiri.

Apabila di kemudian hari terbukti bahwa ada beberapa bagian dari karya ini adalah bukan hasil karya saya sendiri, maka saya siap menanggung resiko dan konsekuensi apapun.

Demikian pernyataan ini saya buat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 28 Mei 2007

(David Syahril)

LEMBAR PENGESAHAN PEMBIMBING

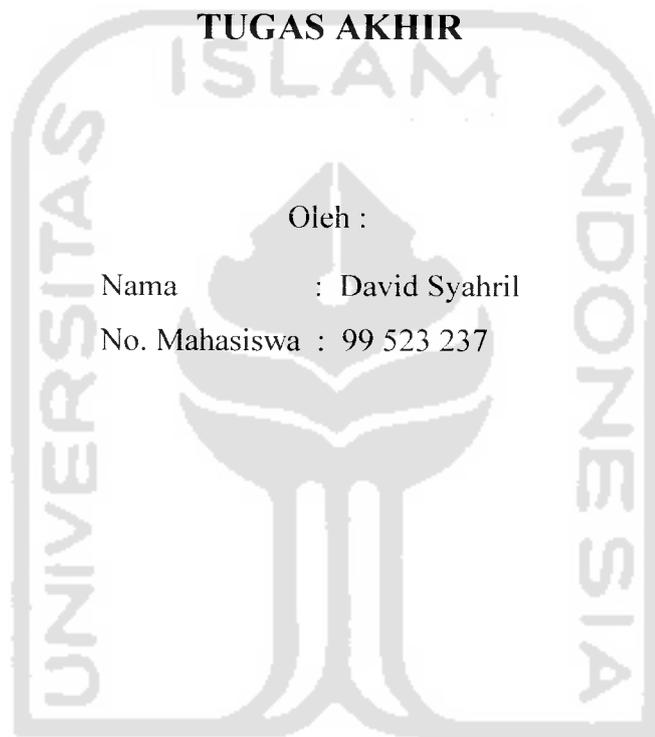
ALAT BANTU AJAR OPERASI-OPERASI DASAR GRAFIS

TUGAS AKHIR

Oleh :

Nama : David Syahril

No. Mahasiswa : 99 523 237



Yogyakarta, 28 Mei 2007

Pembimbing

A handwritten signature in black ink, appearing to read 'Sri Kusumadewi', written over a faint, larger version of the UII logo.

Sri Kusumadewi, Ssi., MT.

LEMBAR PENGESAHAN PENGUJI
ALAT BANTU AJAR OPERASI-OPERASI DASAR GRAFIS
TUGAS AKHIR

Oleh :

Nama : David Syahril

No. Mahasiswa : 99 523 237

Telah Dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat
untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika
Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 28 Juni 2007

Tim Penguji

Sri Kusumadewi, SSi., MT.

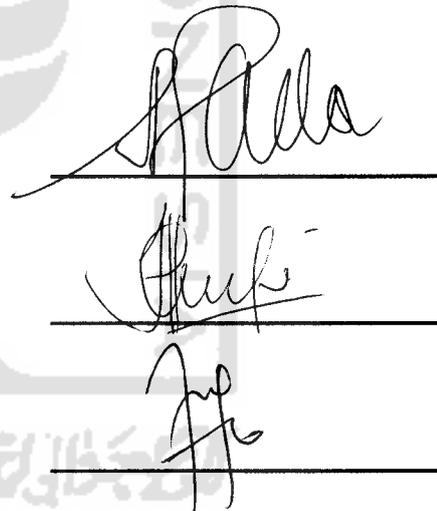
Ketua

Taufiq Hidayat, ST., MCS.

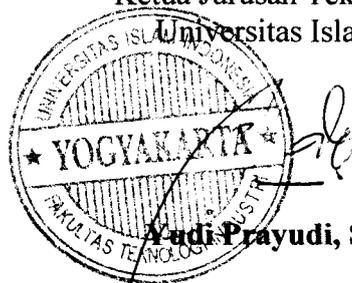
Anggota I

Nur Wijayaning R., S.Kom.

Anggota II



Mengetahui,
Ketua Jurusan Teknik Informatika
Universitas Islam Indonesia



Yudi Prayudi, SSi., M.Kom

Persembahan

“Alhamdulillah ya Allah, berkat-Mu akhirnya selesai juga tugasku”

*Disepanjang perjalanan hidupku yang penuh perjuangan, ketertatihan, misteri,
dan konflik, kupersembahkan karya ini kepada :*

*Mama dan Papa tercinta, atas ketulusan doa, cinta, kasih sayang, cucuran
airmata, dan kesabaran menunggu sarjanaku...
percayaalah bahwa anakmu ini bisa...*

*Semua Kakakku dan Adikku yang kusayang,
Dan seseorang yang selalu mendampingi dan menyananyi aku semoga
kebahagiaan selalu menyertaimu...*

Motto :

“ Sesungguhnya sesudah kesulitan itu ada kemudahan ; Maka apabila kamu telah selesai (dari suatu urusan), kerjakanlah dengan sungguh-sungguh (urusan) yang lain ”.

(Q.S. Alam Nasyrah ayat 6 dan 7)

“ Jadilah sabar dan sholat sebagai penolongmu, sesungguhnya Allah beserta orang-orang yang sabar “.

(Q.S. Al Baqarah ayat 153)

“ Dunia hanya berjalan tiga hari, yaitu : Kemarin, yang kita tidak berpengharapan apa-apa lagi darinya. Hari ini, yang harus kita peroleh kebaikan dan kesuksesannya. Dan esok hari, yang tidak kita ketahui apakah kita termasuk yang masih hidup atau yang tergolong sudah meninggal “.

(Al Hasan Al Bashri)

KATA PENGANTAR



Assalamu'alaikum Wr. Wb

Alhamdulillah, puji syukur kepada Allah SWT atas rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan penulisan tugas akhir yang berjudul *“Alat Bantu Ajar Operasi-Operasi Dasar Grafis”* ini dengan baik.

Telah banyak hal dan rintangan yang dihadapi dalam pembuatan tugas akhir ini, tidak sedikit pihak baik langsung maupun tidak langsung membantu dan mendorong serta memberi masukan berharga dalam pelaksanaan penyusunan tugas akhir ini. Dalam kesempatan kali ini penulis mengucapkan terima kasih kepada :

1. Bapak Fathul Wahid, ST, Msc, selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
2. Bapak Yudi Prayudi, Ssi, M.Kom, selaku Ketua Jurusan Teknik Informatika Universitas Islam Indonesia.
3. Ibu Sri Kusumadewi, S.Si, MT, selaku Dosen Pembimbing Tunggal, yang telah memberikan bimbingan, saran, dan masukan dalam penyusunan tugas akhir ini.

4. Ayahanda ST. Syahril Hasanuddin, BE dan Ibunda Yusmaini tercinta, terimakasih atas motivasi dan kesabaran yang selalu diberikan serta do'a yang tiada henti-hentinya.
5. Kakak-kakak dan adikku, yang selalu memberikan motivasi dan dukungan materi maupun spiritualnya.
6. Teman-teman mahasiswa Teknik Informatika 1999 - 2002.
7. Teman-teman gank sawit.
8. Dan semua pihak yang telah membantu dalam penyusunan tugas akhir ini yang tidak dapat penyusun sebutkan satu persatu karena keterbatasan yang ada.

Dalam penyelesaian Tugas Akhir ini penulis menyadari bahwa masih banyak terdapat kelemahan dan kekurangan. Oleh karena itu penulis mengharapkan kritik dan saran yang bersifat membangun agar pada masa mendatang menjadi lebih baik.

Semoga apa yang penulis sampaikan dalam laporan ini dapat berguna bagi para penuntut ilmu, rekan-rekan mahasiswa dan siapa saja yang membutuhkannya.

Wassalamu'alaikum Wr. Wb.

Yogyakarta, 28 Mei 2007

Penyusun

SARI

Mempelajari sesuatu dalam rangka memahami suatu ilmu pengetahuan ada kalanya membutuhkan alat bantu untuk memudahkannya. Dengan semakin luasnya penggunaan komputer, penulis mencoba memanfaatkan komputer tersebut dalam membantu proses belajar mengajar. Dalam hal ini pembelajaran dilakukan untuk mengetahui bentuk obyek-obyek grafis dengan proses operasi-operasi dasarnya. Obyek grafis yang dibentuk dapat dilakukan perubahan baik bentuk maupun letaknya.

Dengan dibuatnya aplikasi ini, diharapkan dapat sedikit membantu dalam memahami proses-proses operasi dasar pada obyek grafis. Visualisasi perangkat alat bantu ajar operasi-operasi dasar grafis ini dibuat dan diimplementasikan dengan menggunakan bahasa pemrograman visual yakni *Borland Delphi 7*.



DAFTAR ISI

JUDUL.....	i
LEMBAR PENGESAHAN PEMBIMBING.....	ii
LEMBAR PENGESAHAN PENGUJI.....	iii
HALAMAN PERSEMBAHAN.....	iv
HALAMAN MOTTO.....	v
KATA PENGANTAR.....	vi
SARI.....	viii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	2
1.5 Manfaat Penelitian.....	2
1.6 Metodologi Penelitian.....	3
1.7 Sistematika Penulisan.....	3
BAB II LANDASAN TEORI.....	5
2.1 Tinjauan Pustaka.....	5
2.2 Obyek Grafis.....	7

2.3 Sistem Koordinat Obyek Grafis.....	11
2.4 Operasi-operasi Dasar.....	14
2.5 Borland Delphi 7.0.....	15
BAB III METODOLOGI.....	16
3.1 Metode Analisis.....	16
3.2 Pembuatan Program Operasi Grafis.....	16
BAB IV HASIL DAN PEMBAHASAN.....	30
4.1 Hasil.....	30
4.2 Pembahasan.....	36
4.2.1 Kebutuhan Masukan.....	37
4.2.2 Kebutuhan Keluaran.....	38
4.2.3 Kebutuhan Proses.....	38
4.2.4 Antarmuka Sistem.....	38
4.2.5 Analisis Perangkat Lunak.....	39
4.2.6 Analisis Perangkat Keras.....	39
4.2.7 Kuisisioner.....	39
BAB V SIMPULAN DAN SARAN.....	43
5.1 Simpulan.....	43
5.2 Saran.....	43
DAFTAR PUSTAKA.....	45

DAFTAR TABEL

Tabel 2.1 Sistem pemetaan koordinat.....	13
Tabel 2.1 Tabel kuisioner.....	40



DAFTAR GAMBAR

Gambar 2.1 Koordinat obyek terhadap layar.....	12
Gambar 3.1 Attribute Width dan Height sebagai pembentuk obyek.....	17
Gambar 3.2 Menggambar segitiga.....	18
Gambar 3.3 Menggambar kotak.....	18
Gambar 3.4 Menggambar lingkaran.....	19
Gambar 3.5 Operasi obyek segitiga.....	20
Gambar 3.6 Operasi obyek kotak.....	23
Gambar 3.7 Operasi obyek lingkaran.....	26
Gambar 3.8 Desain program grafis.....	29
Gambar 4.1 Tampilan awal program.....	31
Gambar 4.2 Tampilan obyek segitiga.....	32
Gambar 4.3 Tampilan obyek kotak.....	32
Gambar 4.4 Tampilan obyek lingkaran.....	33
Gambar 4.5 Tampilan menu about.....	33
Gambar 4.6a Obyek segitiga.....	34
Gambar 4.6b Translasi obyek segitiga.....	34
Gambar 4.7a Obyek segitiga.....	34
Gambar 4.7b Penskalaan obyek segitiga.....	34
Gambar 4.8a Obyek kotak.....	34
Gambar 4.8b Translasi obyek kotak.....	34
Gambar 4.9a Obyek kotak.....	35

Gambar 4.9b Penskalaan obyek kotak.....	35
Gambar 4.10a Obyek lingkaran.....	35
Gambar 4.10b Translasi obyek lingkaran.....	35
Gambar 4.11a Obyek lingkaran.....	36
Gambar 4.11b Penskalaan obyek lingkaran.....	36
Gambar 4.12 Diagram kuisisioner.....	42



BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi komputer terus berjalan dengan sangat cepat seiring dengan pertumbuhan tingkat kecerdasan manusia. Sekarang ini teknologi komputer telah menyentuh hampir seluruh bidang kehidupan manusia. Mulai dari bidang politik, ekonomi, hiburan, kesehatan sampai pendidikan. Pendidikan telah mengambil peran penting dalam proses kemajuan manusia. Melalui pendidikan tercipta teknologi-teknologi baru yang membantu dan mempermudah aktifitas kehidupan manusia sehari-hari. Seiring berkembangnya zaman, teknologi komputer dapat digunakan sebagai alat bantu dalam proses pendidikan atau lebih dikenal dengan istilah pendidikan secara elektronik.

Operasi-operasi grafis terdiri dari beberapa bentuk obyek seperti lingkaran, segitiga, elips, segiempat dan lain-lain. Bentuk obyek grafis tersebut dapat dijadikan model pengamatan untuk proses pembelajaran pendidikan dasar melalui visualisasi secara komputerisasi.

Hal tersebut menjadi pertimbangan penulis untuk membuat alat bantu ajar dengan menggunakan bahasa pemrograman Delphi 7.0 agar dapat memenuhi kebutuhan akan ilmu pengetahuan tanpa dibatasi waktu dan ruang.

1.2 Rumusan Masalah

Permasalahan yang diangkat dalam tugas akhir ini adalah bagaimana membuat program alat bantu ajar operasi-operasi dasar grafis yang dapat digunakan semudah mungkin dengan memberikan tampilan yang baik.

1.3 Batasan Masalah

Batasan masalah dari tugas akhir ini adalah:

1. Operasi-operasi dasar yang digunakan adalah translasi dan penskalaan.
2. Pergerakan arah pada obyek di bidang area grafis menggunakan tombol mouse.
3. Obyek-obyek grafis yang ditampilkan dibatasi dengan hanya menampilkan lingkaran, segiempat, dan segitiga.

1.4 Tujuan Penelitian

Tujuan pembuatan tugas akhir ini adalah membuat alat bantu ajar untuk obyek-obyek grafis yaitu : lingkaran, segiempat, dan segitiga.

1.5 Manfaat Penelitian

Manfaat yang dapat diambil dari penelitian ini adalah:

1. Membantu sistem belajar mengajar berbasis komputer secara praktis, mudah dan cepat dipahami khususnya dalam mempelajari bentuk obyek-obyek grafis.

2. Menambah ketertarikan untuk mempelajari dan memahami materi yang disampaikan dalam bentuk visual.

1.6 Metodologi Penelitian

Metode yang digunakan dalam pembuatan tugas akhir ini adalah:

1. Mengumpulkan data melalui buku-buku referensi yang relevan mengenai pemrograman grafis komputer dan mempelajari cara-cara untuk membentuk dan memanipulasi gambar model suatu obyek menggunakan komputer sesuai dengan bidang permasalahan yang dihadapi.
2. Implementasi perangkat lunak yang digunakan dalam tugas akhir ini yaitu menggunakan bahasa pemrograman Delphi 7.0.

1.7 Sistematika Penulisan

Penulisan laporan tugas akhir ini tersusun atas lima bab. Bab-bab ini disusun sedemikian rupa sehingga diharapkan isi laporan mudah dipahami. Adapun sistematika dari laporan tugas akhir ini adalah sebagai berikut :

BAB I Pendahuluan

Berisi latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian dan metodologi penelitian.

BAB II Landasan Teori

Teori yang digunakan dalam pembuatan tugas akhir ini adalah merupakan teori-teori yang berhubungan dengan obyek-obyek grafis yang akan diterapkan kedalam teori bahasa pemrograman Delphi.

BAB III Metodologi

Metodologi penelitian yang dilakukan adalah dengan mencari buku-buku referensi yang relevan mengenai pemrograman grafis komputer dan mempelajari cara-cara untuk membentuk dan memanipulasi gambar model suatu obyek menggunakan komputer.

BAB IV Hasil dan Pembahasan

Hasil analisis merupakan pemilihan kebutuhan sistem yang diwujudkan dalam perangkat lunak dan pembahasan tugas akhir ini berisi rancangan penulis dalam melakukan pembuatan obyek-obyek grafis dengan dasar pemrograman Delphi.

BAB V Simpulan dan Saran

Berisikan kesimpulan yang didapatkan selama pembuatan tugas akhir terutama penggunaan alat bantu ajar ini dan saran yang dapat dipergunakan oleh pihak yang berkepentingan maupun untuk peneliti terhadap keterbatasan yang ditemukan dalam pembuatan penelitian ini.

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Aplikasi alat bantu ajar operasi-operasi dasar grafis ini diambil dari beberapa pustaka seperti buku-buku tentang bahasa pemrograman dan buku-buku yang berisi perancangan visualisasi grafis. Secara umum grafis dapat diartikan sebagai salah satu bentuk terma-terma *visual* (gambar) dengan menggunakan titik, warna, bentuk, dan garis. Grafis dalam bahasa Inggris *graphic* adalah presentasi visual pada sebuah permukaan kanvas, dinding, layar komputer, yang bertujuan untuk memberi tanda, informasi, ilustrasi, atau untuk hiburan [WIK07].

Komputer sebagai media alternatif mampu menyajikan data atau informasi dalam bentuk tampilan *image*, teks, *audio* (suara) dan animasi dalam suatu program aplikasi. Untuk itu komputer dapat digunakan sebagai alat untuk memvisualisasikan suatu sistem yang telah ada. Sehingga dari aplikasi itu dapat dihasilkan suatu tampilan yang menarik dan berinteraksi dengan *user* (pengguna). Penggunaan gambar dan teks untuk menyampaikan informasi kepada *user* sering disebut informasi multimedia. Kecanggihan aplikasi ini dapat dengan cepat menarik perhatian dan rasa ingin tahu seseorang sehingga dimanfaatkan untuk hal-hal yang sesuai kepentingan.

Grafis mempekerjakan berbagai elemen seperti garis, marka, simbol, uraian verbal yang divisualisasikan lewat gambar. Elemen-elemen gambar

tersebut diartikan sebagai obyek grafis seperti segiempat, lingkaran, segitiga, elips, dan lain-lain.

Dari uraian tersebut, dapat dijelaskan bahwa obyek-obyek grafis dapat dibuat dengan menggunakan bahasa pemrograman. Beberapa daftar pustaka yang diambil oleh penulis ditujukan agar dapat dibuatkan gambaran sebuah program yang dapat melakukan interaksi dengan *user*. Penulis mendapatkan hasil dari beberapa sumber bahwa penggunaan bahasa pemrograman memegang peranan penting dalam membangun alat bantu ajar grafis ini. Dari beberapa bahasa pemrograman, maka dipilihlah bahasa pemrograman yang sudah mendukung *interface* yang baik. Untuk itulah, diperlukan beberapa referensi yang benar agar didapatkan bahasa pemrograman yang dimaksud. Dalam hal ini, pemilihan bahasa pemrograman haruslah disesuaikan dengan segala kebutuhan.

Bahasa pemrograman memegang peranan penting didalam membangun program aplikasi. Beberapa *software* yang beredar memiliki keunggulan masing-masing. Hal yang harus diperhatikan dalam memilih sebuah bahasa pemrograman adalah tersedianya petunjuk yang lengkap mengenai penggunaan dari bahasa pemrograman tersebut, juga dukungan dari beberapa *software* lainnya. Sebuah bahasa pemrograman harus dapat digunakan semudah mungkin agar pengembang tidak cepat beralih kepada bahasa pemrograman yang lain.

Kelengkapan *feature* dari sebuah program ini akan memberikan efek nyaman bagi *user* ketika berinteraksi dengan sistem, terlebih sistem yang dibangun adalah aplikasi ini. Maka didapatkan pilihan penggunaan bahasa pemrograman Borland Delphi 7.0 yang sudah mendukung GUI (*Graphics User*

Interface). Bahasa pemrograman Borland Delphi 7.0 merupakan *software* yang sudah dikenal luas dikalangan para pemrogram bahasa pemrograman dan memiliki kelayakan untuk dijadikan media pembuatan program aplikasi yang akan digunakan untuk membangun program operasi grafis ini.

Hal penting dalam mengumpulkan referensi adalah agar tidak terjebak dalam memutuskan bagi seseorang untuk memilih penggunaan bahasa pemrograman. Referensi bacaan yang sudah dipaparkan diatas merupakan sumber yang baik bagi terciptanya sebuah program aplikasi, namun bukan berarti semakin banyak referensi akan semakin baik, hal itu sangat tergantung dari kebutuhan masing-masing program.

2.2 Obyek Grafis

Obyek grafis dibuat dari sebuah garis yang terdiri dari kumpulan titik-titik yang digambar secara kontiniu dan linier [GUN05]. Untuk bekerja dalam lingkungan grafis, yang pertama kali perlu diketahui adalah sebagai berikut.

1. Perangkat (*tool*) apa yang diperlukan untuk pemrograman grafis.
2. Perangkat apa yang telah disediakan oleh Delphi maupun Windows.

Dalam pengertian umum, untuk melukis sesuatu diperlukan peralatan seperti alas untuk menggambar, pensil, kuas, dan cat. Alas penggambaran dapat berupa selembar kertas atau kain, dan secara umum disebut kanvas. Untuk memperoleh gambar yang warna-warni maka diperlukan pensil atau cat dengan bermacam warna.

Analogi dengan menggambar pada kertas (kanvas) tersebut, menggambar komputer juga memerlukan perangkat yang sama, yaitu obyek kanvas (Canvas),

pensil (Pen), dan kuas (Brush). Ketiganya merupakan elemen dasar pada pemrograman grafis. Setiap grafis memerlukan kanvas. Dengan Pen dan Brush, kanvas tersebut diisi gambar. Pen untuk corat-coret membuat garis, sedangkan Brush untuk menyapu kanvas dengan warna tertentu. Pen juga menyediakan fasilitas untuk mengatur gaya, warna dan ketebalan garis, sedangkan Brush menyediakan fasilitas untuk mengatur warna dan gaya sapuan pada kanvas.

Delphi 7.0 memberikan komponen-komponen grafis dan methodnya sehingga dapat dipakai dengan mudah. Dibawah ini adalah komponen-komponen yang digunakan dalam program ini.

1. Koordinat

Setiap komponen visual mempunyai property top dan left. Nilai yang tersimpan di dalam property ini menyatakan di mana komponen tersebut diletakkan pada form. Koordinat ini bersifat relatif.

2. Canvas

Canvas adalah property yang dimiliki oleh area gambar atau komponen-komponen grafis lainnya. Dengan property ini kita dapat memanipulasi area gambar. Canvas mempunyai property dan method.

3. Pixel dan Point

Pada dasarnya semua operasi grafis adalah mengubah warna pixel pada area gambar. Selain itu juga dapat memanipulasi pixel secara individual.

4. Palette dan True Color

Palette pada grafis adalah sekumpulan warna yang dapat dipakai di layar. True color adalah warna yang dihasilkan dari kombinasi merah, hijau dan biru. Setiap warna sama dengan sebuah bilangan. Warna ini dinamakan RGB (Red Green Blue). Selain warna RGB, Delphi juga dapat memakai warna yang telah didefinisikan, misalnya *clBlue* untuk menyatakan warna biru dan *clBackGround* untuk menyatakan warna latar belakang.

5. Pen

Pen dipakai untuk menggambar garis atau bentuk. Pen mempunyai atribut, misalnya warna dan lebar. Pen juga mempunyai method untuk menggambar aneka bentuk, seperti lingkaran, segitiga, garis, kotak dan lain-lain.

6. Brush

Brush dipakai untuk menentukan corak pengisian obyek. Ada 3 property dari Brush yaitu Color, Style, Bitmap. Dengan property Color dan Style, warna dari pengisian tergantung dari isi property Color. Property Style mendefinisikan corak pengisian.

Untuk membuat obyek grafis digunakan dua istilah, yaitu menggambar dan melukis. Menggambar (*drawing*) adalah membuat sebuah elemen grafis seperti garis atau bentuk lainnya, menggunakan baris-baris perintah. Contoh fasilitas penggambaran yang disediakan antara lain *LineTo()*, *Rectangle()*, *Ellipse()*, dan lain-lain. Melukis (*painting*) adalah membuat tampilan menyeluruh

sebuah obyek. Melukis biasanya melibatkan proses menggambar. Contohnya adalah menampilkan gambar pada komponen citra yaitu Image, sedangkan untuk melukis antara lain Draw(), StretchDraw(), BrushCopy(), dan lain-lain.

Grafis selalu digambar pada obyek kanvas. Dengan konsep kanvasnya, pemrograman grafis menjadi lebih mudah. Grafis dapat dibuat, digambar, disimpan, dan dihapus tanpa harus menggambarnya langsung ke layar. Pada umumnya grafis digambar dulu di kanvas maya, setelah proses penggambarannya selesai, kemudian hasil gambar tersebut ditampilkan ke kanvas layar melalui komponen visual. Sebuah obyek grafis yang tampil pada aplikasi sangat tergantung dari jenis obyek yang mempunyai kanvas tersebut. Jika menggambar langsung ke kanvas komponen visual, hasilnya akan ditampilkan segera. Tetapi jika menggambar di kanvas maya seperti pada kanvas Bitmap maka gambar tersebut tidak akan ditampilkan sampai gambar tersebut dikopikan ke kanvas penampilnya. Untuk menampilkan grafis disediakan beberapa kanvas melalui komponen PaintBox, Image dan obyek Bitmap.

Komponen PaintBox menyediakan kanvas untuk menggambar elemen grafis secara bebas dan langsung pada kanvas yang tersedia. Komponen Image digunakan untuk menampilkan berkas citra grafis yang berupa bitmap dan icon. Obyek Bitmap adalah kanvas maya yang tidak tampak, yang biasanya digunakan untuk menyembunyikan proses penggambaran. Setelah penggambaran grafis selesai, hasilnya dapat ditampilkan pada kanvas komponen lainnya, seperti PaintBox, Image, dan lain-lain. Obyek Bitmap ini juga mempunyai kemampuan untuk memanggil (*load*) citra untuk mengisi kanvasnya dan juga mempunyai

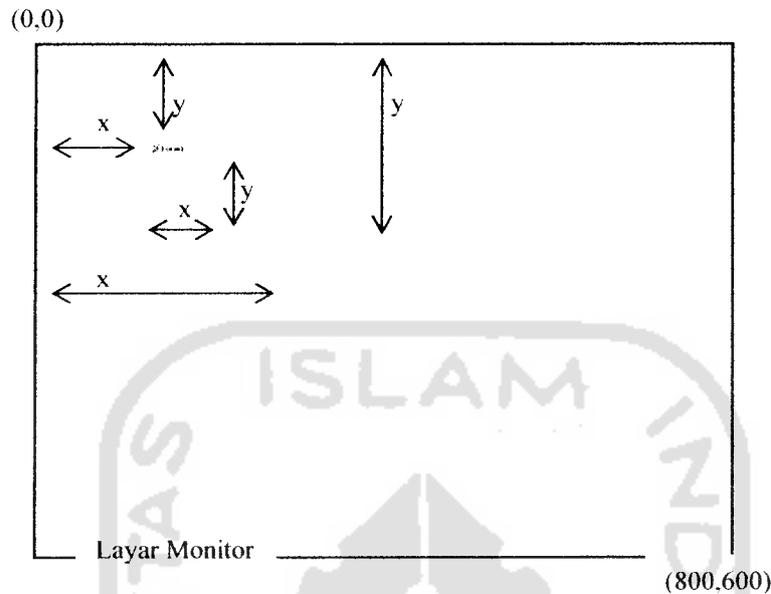
kemampuan untuk menyimpan (*save*) citra pada kanvasnya ke dalam sebuah berkas terpisah.

2.3 Sistem Koordinat Obyek Grafis

Obyek-obyek grafis yang membentuk tampilan tidak digambar langsung pada layar monitor, tetapi melalui konsep visual, yaitu melalui kanvas pada komponen visual seperti Form, Panel, dan lain-lain. Tampilan utama aplikasi adalah sebuah Form. Semua obyek yang membentuk tampilan diletakkan pada Form tersebut.

Koordinat tiap obyek diukur terhadap Form. Satuan ukur koordinat ini adalah piksel (*pixel*), dihitung dari kiri ke kanan (sumbu X) dan dari atas ke bawah (sumbu Y). Sumbu mendatar positif ke arah kanan dan sumbu tegak positif ke arah bawah.

Kebanyakan GDI (*Graphics Device Interface*) yang disediakan oleh Windows membutuhkan koordinat-koordinat yang menentukan lokasi untuk meletakkan gambarnya. GDI menganggap semua masukan berorientasi pada sumbu vertikal dan horisontal mengikuti pertambahan atau pengurangan dari nilai-nilai koordinat X,Y. Koordinat-koordinat ini berlandaskan pada suatu satuan pengukuran, misalnya *pixel*. Gambar 2.1 memperlihatkan koordinat terhadap layar.



Gambar 2.1 Koordinat obyek terhadap layar

Dalam Windows menggunakan dua sistem koordinat untuk memplot koordinat gambar pada area gambar yaitu koordinat perangkat (*device*) dan koordinat logikal (*logical*). Sistem koordinat perangkat (*device*) menunjuk pada perangkat yang sedang dijalankan oleh Windows. Ukurannya dalam *pixel* sedangkan orientasinya sedemikian rupa sehingga sumbu horisontal dan vertikalnya bertambah dari kiri ke kanan dan dari atas ke bawah. Sebagai contoh, jika Windows dengan layar 800×600 *pixel* dijalankan maka koordinat pada pojok kiri atas perangkat yang digunakan adalah $(0,0)$ sedangkan koordinat pada pojok kanan bawah $(800,600)$. Koordinat perangkat jarang sekali digunakan karena gambar tidak langsung dibuat pada perangkat. Sedangkan koordinat logis (*logical*) menunjuk pada sistem koordinat yang digunakan berdasarkan area tertentu pada layar. Area ini disebut jendela (*window*) dan direpresentasikan oleh Form, Panel, dan lain-lain. Koordinat tersebut diukur terhadap bidang area dari ujung kiri-atas

Form adalah koordinat (0,0) dan ujung kanan-bawah ditentukan oleh lebar dan tinggi Form. Setiap posisi dihitung relatif terhadap kanvas tempat obyek tersebut ditampilkan.

Yang digunakan pada penelitian ini adalah koordinat logikal karena menggunakan area gambar yaitu *form* yang lokasinya pada perangkat penampil bisa berpindah-pindah. Disamping pemetaan koordinat semacam ini, Windows juga menyediakan beberapa system pemetaan koordinat lain untuk menunjang kebutuhan khusus seperti untuk desain gambar teknik dan aplikasi lain yang membutuhkan kepresisian dan fleksibilitas dalam satuan koordinatnya. Tabel 2.1 adalah daftar selengkapnya sistem pemetaan koordinat yang disediakan oleh Windows.

Tabel 2.1 Sistem pemetaan koordinat

Mode Pemetaan	Ukuran Satuan Logikal	Orientasi (X,Y)
MM_ANISOTROPIC	Skala ($x \neq y$) atau ($x = y$)	Didefinisikan / Didefinisikan
MM_HIENGLISH	0.001 inch	Ke kanan / Ke atas
MM_HIMETRIC	0.01 mm	Ke kanan / Ke atas
MM_ISOTROPIC	Skala ($x = y$)	Didefinisikan / Didefinisikan
MM_LOENGLISH	0.01 inch	Ke kanan / Ke atas
MM_LOMETRIC	0.1 mm	Ke kanan / Ke atas
MM_TEXT	1 piksel	Ke kanan / Ke bawah
MM_TWIPS	1/1440 inch	Ke kanan / Ke atas

Keterangan :

- MM_TEXT adalah default pada Windows.
- MM_ANISOTROPIC dan MM_ISOTROPIC adalah untuk menentukan satuan, orientasi dan skala yang diinginkan.

- MM_HIENGLISH, MM_HIMETRIC, MM_LOENGLISH, MM_LOMETRIC, dan MM_TWIPS adalah untuk aplikasi dalam menggambar yang membutuhkan satuan inci atau millimeter.

2.4 Operasi-operasi Dasar

Seperti diketahui, operasi-operasi grafis dapat berupa translasi, pembesaran, rotasi, penskalaan, shearing dan lain-lainnya. Dalam hal ini penulis mencoba melakukan dua tahapan operasi-operasi dasar yang akan diuraikan dibawah ini :

1. Translasi (*Translation*)

Translasi adalah suatu pergerakan atau perpindahan semua titik dari obyek pada suatu jalur lurus sehingga menempati posisi baru. Dalam operasi ini, setiap titik pada obyek yang di translasi bergerak dalam jarak yang sama. Pergerakan tersebut dapat berlaku dalam arah sumbu X saja atau dalam sumbu Y saja atau keduanya. Koordinat baru titik yang ditranslasi dapat diperoleh dengan menggunakan rumus :

$$X' = X + T_X$$

$$Y' = Y + T_Y$$

dimana (X, Y) adalah koordinat asal suatu obyek dan (X', Y') adalah koordinat baru obyek tersebut setelah ditranslasi [GUN05].

2. Penskalaan (*Scaling*)

Penskalaan adalah melakukan perubahan ukuran baik memperbesar atau memperkecil pada suatu obyek dengan tetap mempertahankan proporsionalitasnya. Proporsionalitas adalah perbandingan tinggi dan lebar obyek. Obyek akan selalu proporsional jika skala mendatar dan

skala tegak selalu sama. Tapi adakalanya obyek tidak proporsional tergantung pada faktor penskalaan yang diberikan. Koordinat baru titik yang diskala dapat diperoleh dengan rumus :

$$X' = X \cdot S_x$$

$$Y' = Y \cdot S_y$$

dimana (X, Y) adalah koordinat asal suatu obyek dan (X', Y') adalah koordinat setelah diskala [GUN05].

2.5 Borland Delphi 7.0

Borland Delphi versi 7.0 adalah bahasa pemrograman berbasis MS-Windows. Sebagai bahasa pemrograman yang mutakhir, Delphi 7.0 didesain untuk dapat memanfaatkan fasilitas MS-Windows.

Delphi 7.0 juga merupakan bahasa pemrograman Object Oriented Programming (OOP), yaitu pemrograman yang berorientasi obyek. Delphi 7.0 menyediakan obyek-obyek yang sangat kuat, berguna dan mudah dipakai. Dengan fasilitas tersebut membuat Delphi 7.0 menjadi begitu diidamkan oleh para programmer.

Kemudahan cara pemakaian Delphi 7.0 membuat pemula tidak akan terlalu sulit untuk mempelajarinya, sedangkan kecanggihannya akan menarik bagi orang yang sudah mulai mempelajarinya untuk lebih mendalami dan memanfaatkannya.

BAB III

METODOLOGI PENELITIAN

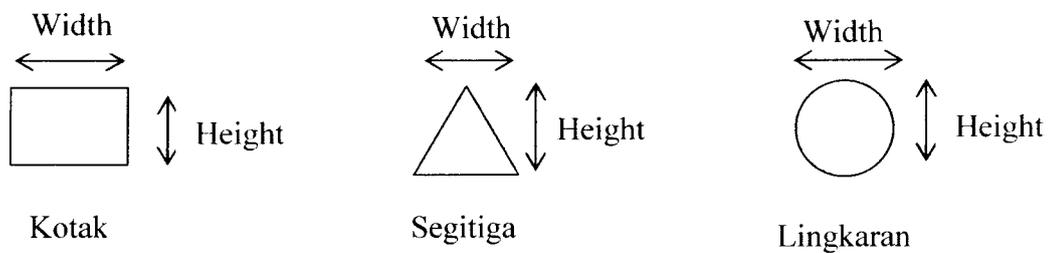
3.1 Metode Analisis

Analisis suatu sistem merupakan salah satu proses yang harus dilakukan dalam perancangan dan implementasi suatu perangkat lunak untuk mengidentifikasi dan mengevaluasi apa saja yang dibutuhkan dalam merancang aplikasi ini. Tahap ini sangat menentukan dalam keberhasilan perangkat lunak yang akan dibuat nanti.

Sistem yang akan dibuat dan digunakan dalam membangun aplikasi program ini dirancang dengan menggunakan metode analisis terstruktur (*structured analysis*), lengkap dengan alat (*tools*) yang berupa jenis komponen yang dibutuhkan dan teknik (*technique*) yaitu metode dan proses-proses yang dibutuhkan dalam pengembangan sistem.

3.2 Pembuatan Program Operasi Grafis

Pada prinsipnya obyek-obyek grafis dibuat melalui kumpulan dari titik-titik yang digambar secara kontiniu dan linier yang menghasilkan sebuah garis. Pada saat dibentuknya garis maka secara bersamaan dicari pula rumusan-rumusan dalam pembentukan obyek-obyek grafis. Hasilnya diperoleh bahwa untuk obyek-obyek kotak, lingkaran atau segitiga diwakili oleh dua *attribute* yaitu *Width* yang dipakai untuk mengubah lebar dan *Height* untuk tingginya seperti terlihat pada Gambar 3.1.



Gambar 3.1 Attribute Width dan Height sebagai pembentuk obyek

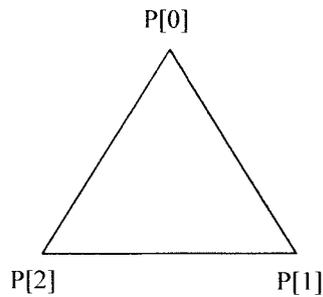
Dari kedua *attribute* ini bisa dibentuk gambar-gambar tersebut dengan fungsi-fungsi yaitu `TFormKomponen.GambarKotak()` untuk membuat gambar dasar kotak, `TFormKomponen.GambarSegitiga()` untuk membuat gambar segitiga, dan `TFormKomponen.GambarLingkaran()` untuk membuat gambar lingkaran seperti berikut ini.

Prosedur untuk menggambar segitiga didapatkan dari tiga titik yaitu titik 0, titik 1, dan titik 2. Ketiga titik tersebut dihubungkan dengan fungsi `Polygon` yang dapat membaca nilai-nilai parameter dari sederetan titik-titik. Sedangkan `TPoint` adalah kumpulan titik-titik koordinat pembentuk poligon. Hasilnya pada Gambar 3.2.

```

procedure TForm1.SegitigaClick(Sender: TObject);
var
    p : array [0..2] of TPoint;
begin
    p[0].x := Round ((Width) /2);
    p[0].y := Round ((Height-100) /2);
    p[1].x := Round ((Width-100) /2);
    p[1].y := Round ((Height-100) /2+100);
    p[2].x := Round ((Width-100) /2+100);
    p[2].y := Round ((Height-100) /2+100);
    Canvas.Polygon (p);
end;

```



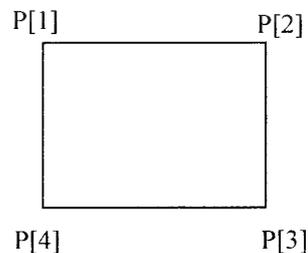
Gambar 3.2 Menggambar segitiga

Prosedur untuk menggambar kotak hampir sama dengan segitiga. Perbedaannya hanya pada penambahan jumlah titiknya. Obyek kotak dibentuk dari empat titik yaitu titik 0, titik 1, titik 2, dan titik 3. Hasilnya pada Gambar 3.3.

```

procedure TForm1.KotakClick(Sender: TObject);
var
  p : array [0..3] of TPoint;
begin
  p[0].x := Round ((Width-100) /2);
  p[0].y := Round ((Height-100) /2);
  p[1].x := Round ((Width-100) /2+100);
  p[1].y := Round ((Height-100) /2);
  p[2].x := Round ((Width-100)/ 2+100);
  p[2].y := Round ((Height-100) /2+100);
  p[3].x := Round ((Width-100) /2);
  p[3].y := Round ((Height-100) /2+100);
  Canvas.Polygon (p);
end;

```



Gambar 3.3 Menggambar kotak

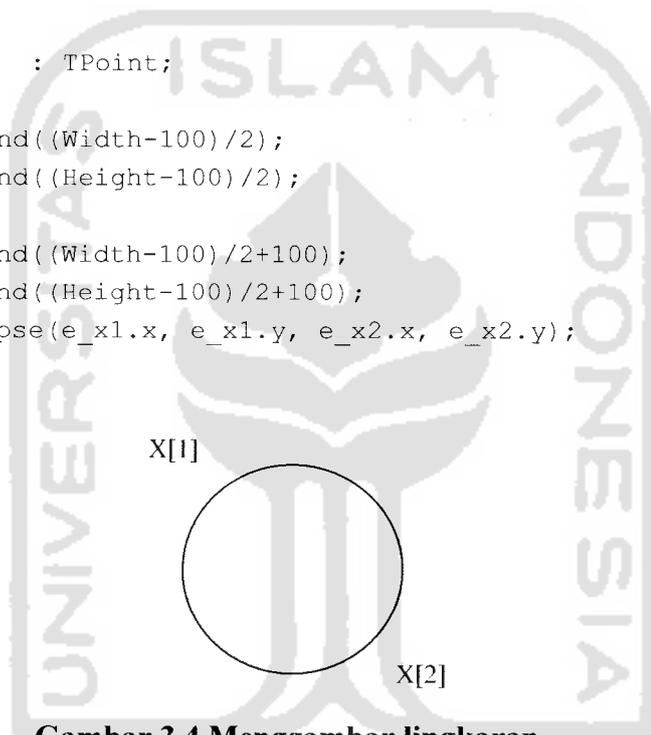
Menggambar lingkaran pada kanvas dibentuk dari batas segiempat yang dinyatakan dengan koordinat kiri-atas dan kanan-bawah. Koordinat X1 merupakan batas lingkaran sebelah kiri-atas sedangkan X2 merupakan batas sebelah kanan-bawah. Hasilnya pada Gambar 3.4.

```

procedure TForm1.LingkaranClick(Sender: TObject);
var
    e_x1, e_x2    : TPoint;
begin
    e_x1.x:=Round((Width-100)/2);
    e_x1.y:=Round((Height-100)/2);

    e_x2.x:=Round((Width-100)/2+100);
    e_x2.y:=Round((Height-100)/2+100);
    Canvas.Ellipse(e_x1.x, e_x1.y, e_x2.x, e_x2.y);
end;

```



Gambar 3.4 Menggambar lingkaran

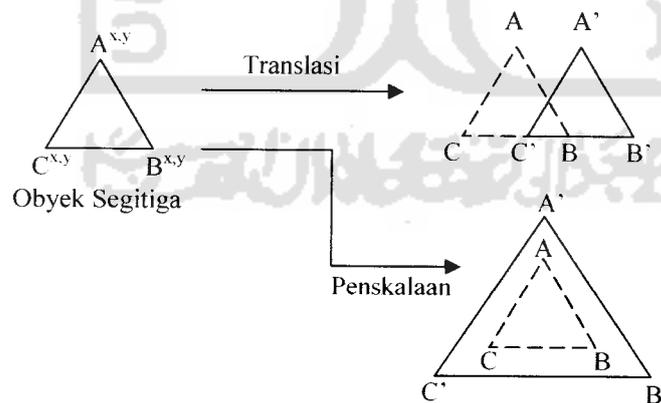
Delphi telah banyak menyediakan fungsi-fungsi untuk menunjang pemrograman grafis ini. Fungsi-fungsi tersebut sedapat mungkin digunakan karena telah teruji dan lebih mudah digunakan. Sebagai pedoman, perintah fungsi-fungsi API hanya jika Delphi belum menyediakannya, atau fungsi yang tersedia tidak sesuai dengan kebutuhan. Berikut ini daftar fungsi-fungsi yang digunakan dalam program ini.

1. Polygon() adalah menggambar bentuk tertutup dengan menghubungkan titik-titik yang telah ditentukan.

2. Ellipse() adalah menggambar bentuk elips berdasarkan area segiempat.
3. FillRect() adalah mengisi area obyek dengan warna tertentu sesuai nilai Brush yang digunakan.
4. Round() adalah menggambar obyek dengan menggunakan sudut-sudut bulat.
5. ReDraw() adalah menampilkan ulang obyek grafis pada koordinat tertentu.

Operasi-operasi yang akan digunakan dalam program ini adalah translasi yang berarti memindahkan suatu obyek sepanjang garis lurus dari suatu lokasi koordinat tertentu ke lokasi yang lain dan skala yang berarti mengubah ukuran suatu obyek. Proses-proses pengoperasian obyek-obyek grafis dijelaskan dibawah ini.

1. Proses pengoperasian pada obyek segitiga :



Gambar 3.5 Operasi obyek segitiga

Persamaan translasi geser ke kanan untuk obyek segitiga :

$$A' = A^x + \text{ZoomLevel}$$

$$B' = B^x + \text{ZoomLevel}$$

$$C' = C^x + \text{ZoomLevel}$$

Persamaan 3.1

Persamaan translasi geser ke kiri untuk obyek segitiga :

$$A' = A^x - \text{ZoomLevel}$$

$$B' = B^x - \text{ZoomLevel}$$

$$C' = C^x - \text{ZoomLevel}$$

Persamaan 3.2

Persamaan translasi geser ke atas untuk obyek segitiga :

$$A' = A^y - \text{ZoomLevel}$$

$$B' = B^y - \text{ZoomLevel}$$

$$C' = C^y - \text{ZoomLevel}$$

Persamaan 3.3

Persamaan translasi geser ke bawah untuk obyek segitiga :

$$A' = A^y + \text{ZoomLevel}$$

$$B' = B^y + \text{ZoomLevel}$$

$$C' = C^y + \text{ZoomLevel}$$

Persamaan 3.4

Persamaan skala perbesar untuk obyek segitiga :

$$A' = A^y - \text{ZoomLevel}$$

$$B' = B^x - \text{ZoomLevel}$$

$$B^y + \text{ZoomLevel}$$

$$C' = C^x + \text{ZoomLevel}$$

$$C^y + \text{ZoomLevel}$$

Persamaan 3.5

Persamaan skala perkecil untuk obyek segitiga :

$$A' = A^y + \text{ZoomLevel}$$

$$B' = B^x + \text{ZoomLevel}$$

$$B^y - \text{ZoomLevel}$$

$$C' = C^x - \text{ZoomLevel}$$

$$C^y - \text{ZoomLevel}$$

Persamaan 3.6

Persamaan skala perbesar lebar untuk obyek segitiga :

$$B' = B^x - \text{ZoomLevel}$$

$$C' = C^x + \text{ZoomLevel}$$

Persamaan 3.7

Persamaan skala perkecil lebar untuk obyek segitiga :

$$B' = B^x + \text{ZoomLevel}$$

$$C' = C^x - \text{ZoomLevel}$$

Persamaan 3.8

Persamaan skala perbesar tinggi untuk obyek segitiga :

$$A' = A^y - \text{ZoomLevel}$$

$$B' = B^y + \text{ZoomLevel}$$

$$C' = C^y + \text{ZoomLevel}$$

Persamaan 3.9

Persamaan skala perkecil tinggi untuk obyek segitiga :

$$A' = A^y + \text{ZoomLevel}$$

$$B' = B^y - \text{ZoomLevel}$$

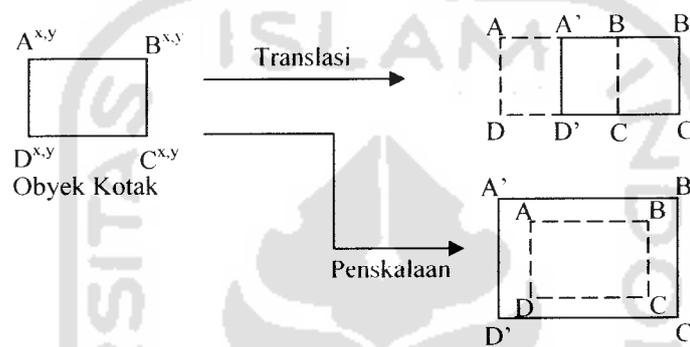
$$C' = C^y - \text{ZoomLevel}$$

Persamaan 3.10

Keterangan :

A, B, C adalah koordinat asal obyek dan A', B', C' adalah koordinat baru obyek setelah dioperasikan. X adalah translasi menurut sumbu X dan Y menurut sumbu Y. ZoomLevel adalah jarak perpindahan obyek.

2. Proses pengoperasian pada obyek kotak :



Gambar 3.6 Operasi obyek kotak

Persamaan translasi geser ke kanan untuk obyek kotak :

$$A' = A^x + \text{ZoomLevel}$$

$$B' = B^x + \text{ZoomLevel}$$

$$C' = C^x + \text{ZoomLevel}$$

$$D' = D^x + \text{ZoomLevel}$$

Persamaan 3.11

Persamaan translasi geser ke kiri untuk obyek kotak :

$$A' = A^x - \text{ZoomLevel}$$

$$B' = B^x - \text{ZoomLevel}$$

$$C' = C^x - \text{ZoomLevel}$$

$$D' = D^x - \text{ZoomLevel}$$

Persamaan 3.12

Persamaan translasi geser ke atas untuk obyek kotak :

$$A' = A^y - \text{ZoomLevel}$$

$$B' = B^y - \text{ZoomLevel}$$

$$C' = C^y - \text{ZoomLevel}$$

$$D' = D^y - \text{ZoomLevel}$$

Persamaan 3.13

Persamaan translasi geser ke bawah untuk obyek kotak :

$$A' = A^y + \text{ZoomLevel}$$

$$B' = B^y + \text{ZoomLevel}$$

$$C' = C^y + \text{ZoomLevel}$$

$$D' = D^y + \text{ZoomLevel}$$

Persamaan 3.14

Persamaan skala perbesar untuk obyek kotak :

$$A' = A^x - \text{ZoomLevel}$$

$$A^y - \text{ZoomLevel}$$

$$B' = B^x + \text{ZoomLevel}$$

$$B^y - \text{ZoomLevel}$$

$$C' = C^x + \text{ZoomLevel}$$

$$C^y + \text{ZoomLevel}$$

$$D' = D^x - \text{ZoomLevel}$$

$$D^y + \text{ZoomLevel}$$

Persamaan 3.15

Persamaan skala perkecil untuk obyek kotak :

$$A' = A^x + \text{ZoomLevel}$$

$$A^y + \text{ZoomLevel}$$

$$B' = B^x - \text{ZoomLevel}$$

$$B^y + \text{ZoomLevel}$$

$$C' = C^x - \text{ZoomLevel}$$

$$C^y - \text{ZoomLevel}$$

$$D' = D^x + \text{ZoomLevel}$$

$$D^y - \text{ZoomLevel}$$

Persamaan 3.16

Persamaan skala perbesar lebar untuk obyek kotak :

$$A' = A^x - \text{ZoomLevel}$$

$$B' = B^x + \text{ZoomLevel}$$

$$C' = C^x + \text{ZoomLevel}$$

$$D' = D^x - \text{ZoomLevel}$$

Persamaan 3.17

Persamaan skala perkecil lebar untuk obyek kotak :

$$A' = A^x + \text{ZoomLevel}$$

$$B' = B^x - \text{ZoomLevel}$$

$$C' = C^x - \text{ZoomLevel}$$

$$D' = D^x + \text{ZoomLevel}$$

Persamaan 3.18

Persamaan skala perbesar tinggi untuk obyek kotak :

$$A' = A^y - \text{ZoomLevel}$$

$$B' = B^y - \text{ZoomLevel}$$

$$C' = C^y + \text{ZoomLevel}$$

$$D' = D^y + \text{ZoomLevel}$$

Persamaan 3.19

Persamaan skala perkecil tinggi untuk obyek kotak :

$$A' = A^y + \text{ZoomLevel}$$

$$B' = B^y + \text{ZoomLevel}$$

$$C' = C^y - \text{ZoomLevel}$$

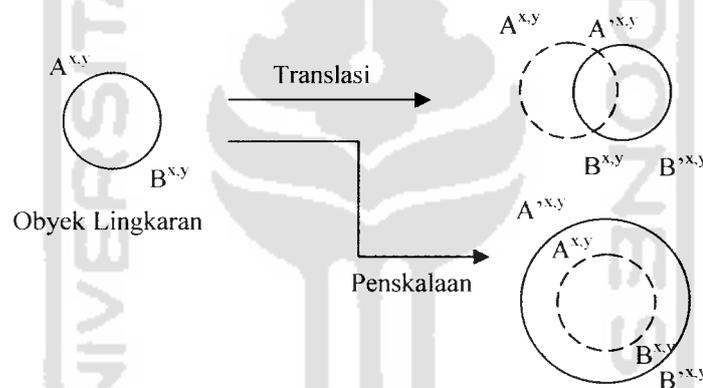
$$D' = D^y - \text{ZoomLevel}$$

Persamaan 3.20

Keterangan :

A, B , C, D adalah koordinat asal obyek dan A', B', C', D' adalah koordinat baru obyek setelah dioperasikan. X adalah translasi menurut sumbu X dan Y menurut sumbu Y. ZoomLevel adalah jarak perpindahan obyek.

3. Proses operasi pada obyek lingkaran :



Gambar 3.7 Operasi obyek lingkaran

Persamaan translasi geser ke kanan untuk obyek lingkaran :

$$A' = A^x + \text{ZoomLevel}$$

$$B' = B^x + \text{ZoomLevel}$$

Persamaan 3.21

Persamaan translasi geser ke kiri untuk obyek lingkaran :

$$A' = A^x - \text{ZoomLevel}$$

$$B' = B^x - \text{ZoomLevel}$$

Persamaan 3.22

Persamaan translasi geser ke atas untuk obyek lingkaran :

$$A' = A^y - \text{ZoomLevel}$$

$$B' = B^y - \text{ZoomLevel}$$

Persamaan 3.23

Persamaan translasi geser ke bawah untuk obyek lingkaran :

$$A' = A^y + \text{ZoomLevel}$$

$$B' = B^y + \text{ZoomLevel}$$

Persamaan 3.24

Persamaan skala perbesar untuk obyek lingkaran :

$$A' = A^x - \text{ZoomLevel}$$

$$A^y - \text{ZoomLevel}$$

$$B' = B^x + \text{ZoomLevel}$$

$$B^y + \text{ZoomLevel}$$

Persamaan 3.25

Persamaan skala perkecil untuk obyek lingkaran :

$$A' = A^x + \text{ZoomLevel}$$

$$A^y + \text{ZoomLevel}$$

$$B' = B^x - \text{ZoomLevel}$$

$$B^y - \text{ZoomLevel}$$

Persamaan 3.26

Persamaan skala perbesar lebar untuk obyek lingkaran :

$$A' = A^x - \text{ZoomLevel}$$

$$B' = B^x + \text{ZoomLevel}$$

Persamaan 3.27

Persamaan skala perkecil lebar untuk obyek lingkaran :

$$A' = A^x + \text{ZoomLevel}$$

$$B' = B^x - \text{ZoomLevel}$$

Persamaan 3.28

Persamaan skala pebesar tinggi untuk obyek lingkaran :

$$A' = A^y - \text{ZoomLevel}$$

$$B' = B^y + \text{ZoomLevel}$$

Persamaan 3.29

Persamaan skala perkecil tinggi untuk obyek lingkaran :

$$A' = A^y + \text{ZoomLevel}$$

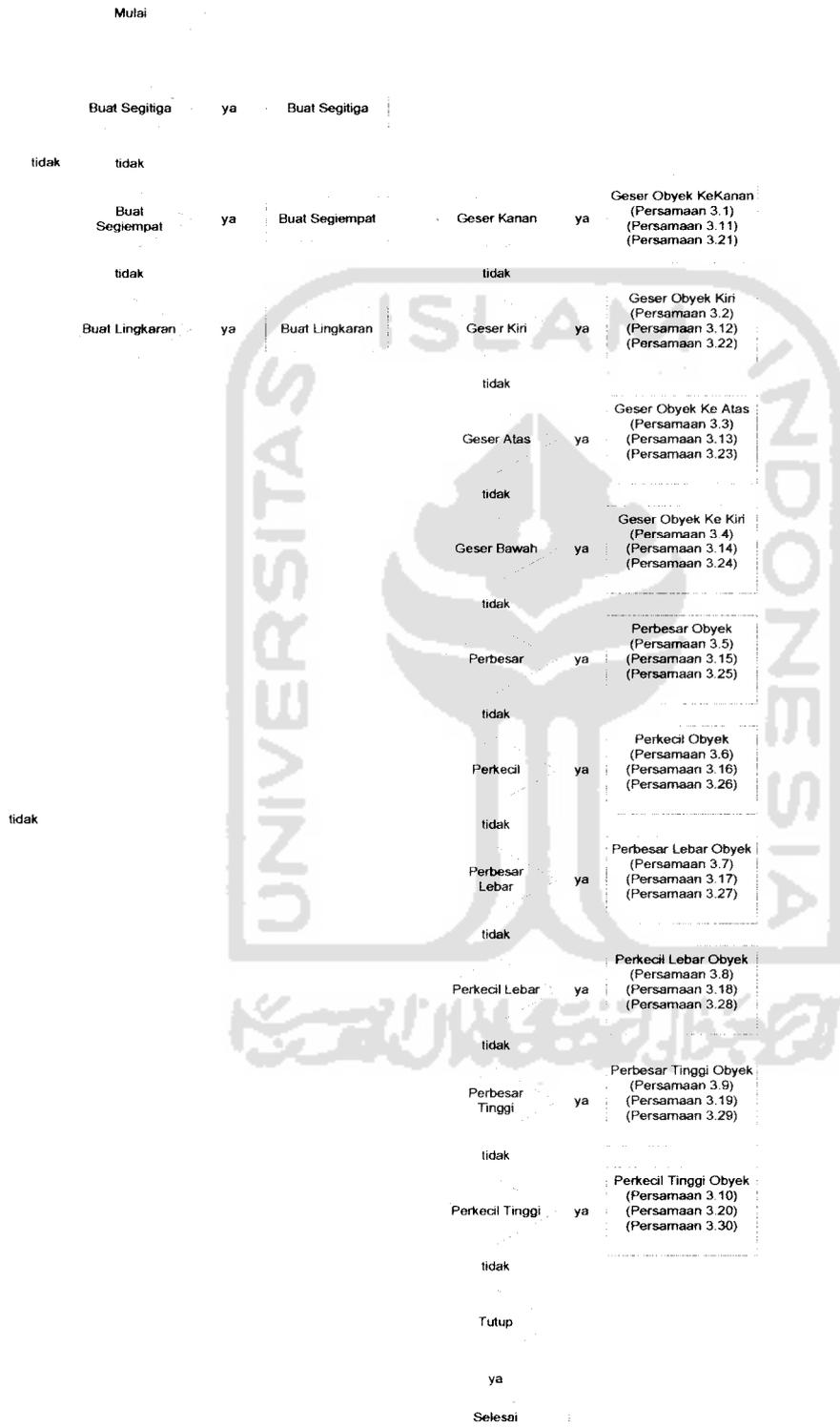
$$B' = B^y - \text{ZoomLevel}$$

Persamaan 3.30

Keterangan :

A, B adalah koordinat asal obyek dan A', B' adalah koordinat baru obyek setelah dioperasikan. ZoomLevel adalah jarak perpindahan obyek.

Setelah melakukan analisa, maka dapat dibuatkan rancangan desain program seperti terlihat pada Gambar 3.8.



Gambar 3.8 Desain program grafis

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil

Pada tahap ini pembuatan program alat bantu ajar operasi-operasi dasar grafis telah dapat dioperasikan dengan benar sesuai dengan prosedurnya dan juga sesuai dengan fungsi dan tujuan program ini dibuat. Sebelum program diterapkan dan diimplementasikan, sebuah program harus sudah terbebas dari kesalahan-kesalahan. Oleh sebab itu, program harus diuji untuk menemukan kesalahan-kesalahan yang mungkin dapat terjadi. Kesalahan dari program yang mungkin terjadi dapat diklasifikasikan dalam tiga bentuk kesalahan, yaitu sebagai berikut:

1. Kesalahan bahasa (*language errors*) atau disebut juga dengan kesalahan penulisan (*syntax errors*) atau kesalahan tata bahasa (*gramatical errors*), adalah kesalahan didalam penulisan kode program yang tidak sesuai dengan yang telah disyaratkan.
2. Kesalahan sewaktu proses (*runtime errors*), adalah kesalahan yang terjadi sewaktu *executable* program dijalankan.
3. Kesalahan logika (*logical errors*), adalah kesalahan dari logika program yang dibuat.

Dalam tahap pengujian ini diharapkan tingkat kesalahan sistem itu sendiri menjadi berkurang bahkan tidak ada supaya program dapat dijalankan dengan benar. Pada saat program pertama kali dijalankan maka akan muncul tampilan awal yaitu *form* utama seperti di bawah ini.

Gambar 4.1 menunjukkan tampilan awal dari program ini.



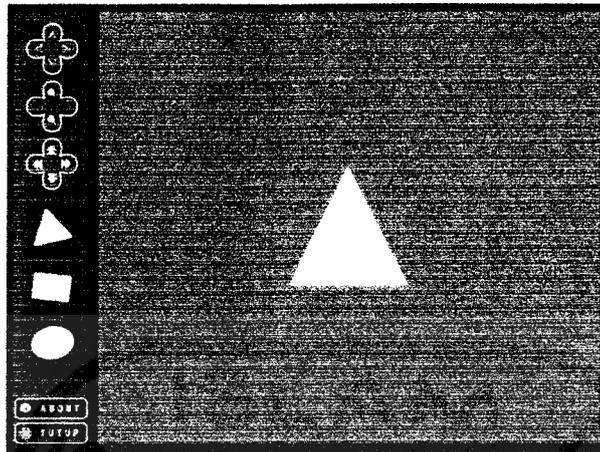
Gambar 4.1 Tampilan awal program

Menu-menu utama program sudah dibuatkan ikon-ikon yaitu :

1. Ikon A adalah ikon translasi yang terdiri dari perintah geser ke kanan, geser ke kiri, geser ke atas dan geser ke bawah.
2. Ikon B dan C adalah ikon skala yang terdiri dari perintah perbesar, perkecil, tambah lebar, kurangi lebar, tambah tinggi dan kurangi tinggi.
3. Ikon D adalah ikon obyek segitiga.
4. Ikon E adalah ikon obyek kotak.
5. Ikon F adalah ikon obyek lingkaran.

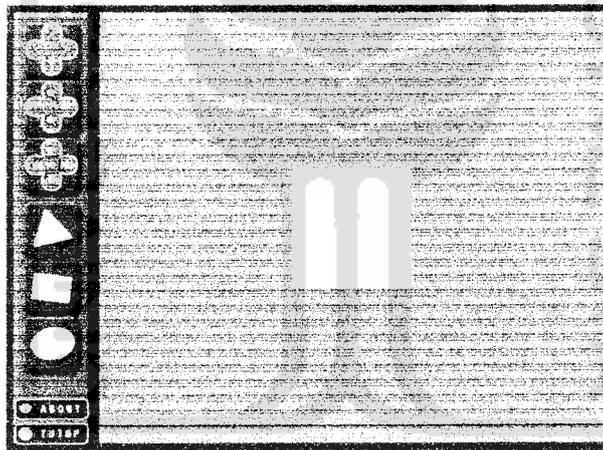
Selanjutnya bisa langsung melihat obyek-obyek grafis dengan memilih ikon-ikon yang ingin ditampilkan seperti gambar dibawah ini.

Gambar 4.2 menunjukkan program yang menampilkan obyek segitiga.



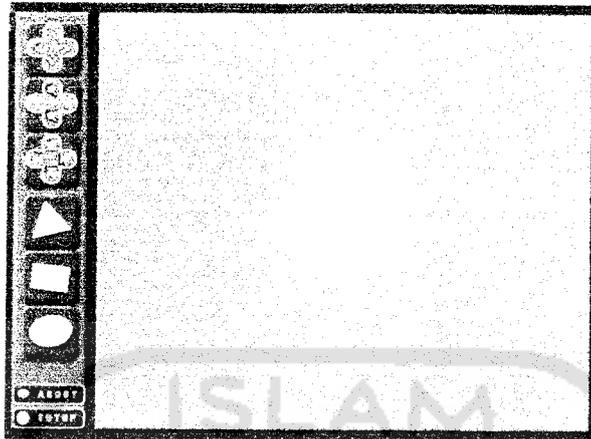
Gambar 4.2 Tampilan obyek segitiga

Gambar 4.3 menunjukkan program yang menampilkan obyek kotak.



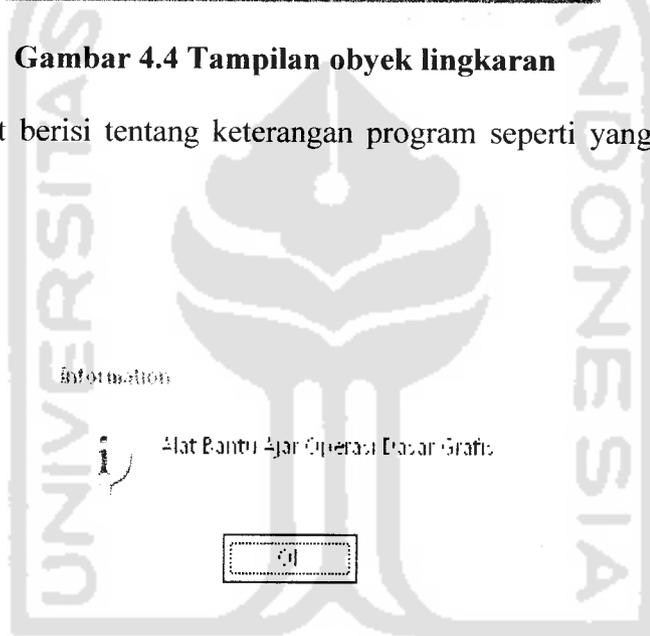
Gambar 4.3 Tampilan obyek kotak

Gambar 4.4 menunjukkan program yang menampilkan obyek lingkaran.



Gambar 4.4 Tampilan obyek lingkaran

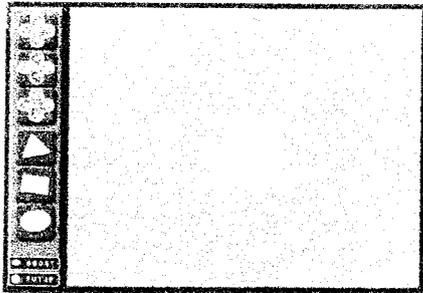
Menu about berisi tentang keterangan program seperti yang ditunjukkan Gambar 4.5.



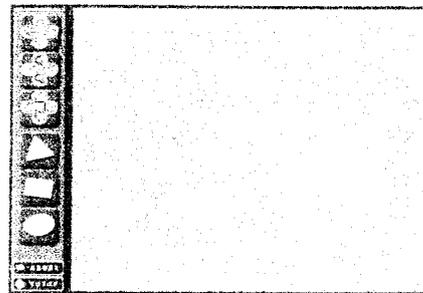
Gambar 4.5 Tampilan menu about

Setelah obyek-obyek grafis dipilih, maka didapatkan bentuk gambar hasil dari pengoperasian seperti dibawah ini.

Gambar 4.6a menunjukkan program yang menampilkan obyek segitiga sebelum di translasi dan Gambar 4.6b menunjukkan obyek segitiga setelah di translasi.



Gambar 4.6a Obyek segitiga



Gambar 4.6b Translasi obyek segitiga

Gambar 4.7a menunjukkan program yang menampilkan obyek segitiga sebelum di skala dan Gambar 4.7b menunjukkan obyek segitiga setelah di skala.

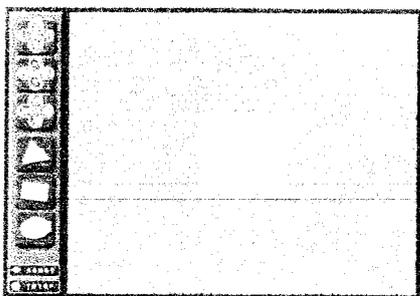


Gambar 4.7a Obyek segitiga

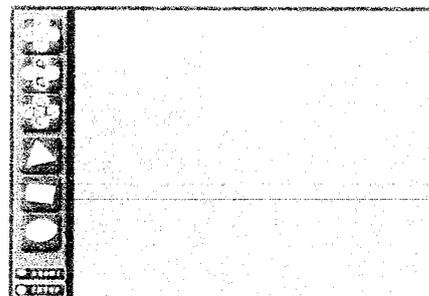


Gambar 4.7b Skala obyek segitiga

Gambar 4.8a menunjukkan program yang menampilkan obyek kotak sebelum di translasi dan Gambar 4.8b menunjukkan obyek kotak setelah di translasi.

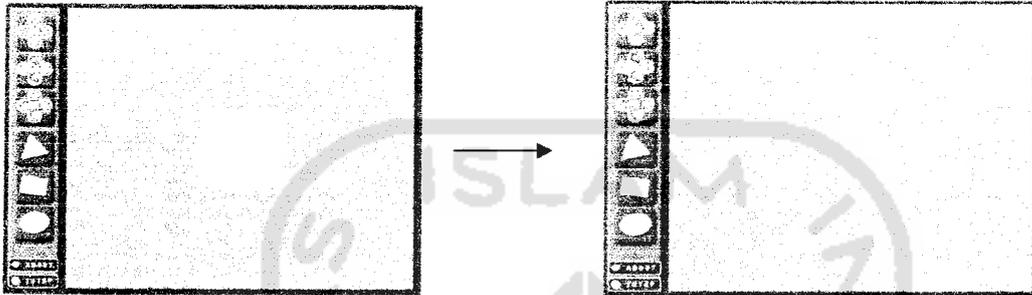


Gambar 4.8a Obyek kotak



Gambar 4.8b Translasi obyek kotak

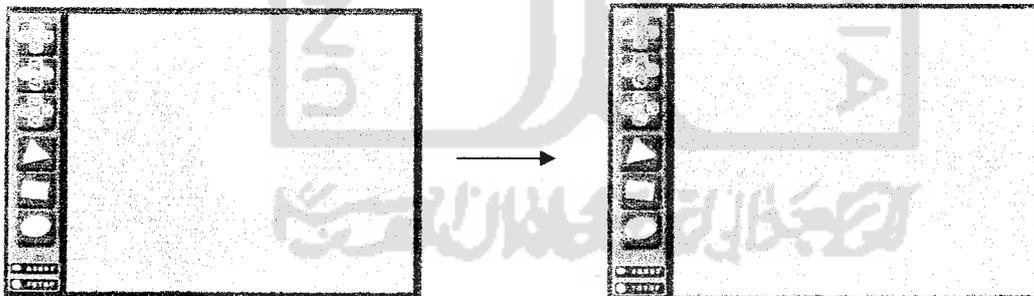
Gambar 4.9a menunjukkan program yang menampilkan obyek kotak sebelum di skala dan Gambar 4.9b menunjukkan obyek kotak setelah di skala.



Gambar 4.9a Obyek kotak

Gambar 4.9b Skala obyek kotak

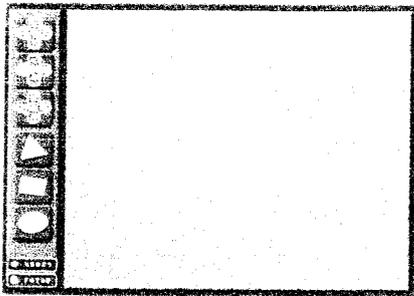
Gambar 4.10a menunjukkan program yang menampilkan obyek lingkaran sebelum di translasi dan Gambar 4.10b menunjukkan obyek lingkaran setelah di translasi.



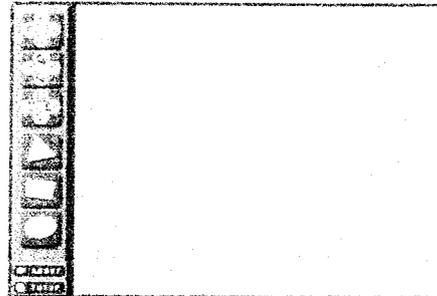
Gambar 4.10a Obyek lingkaran

Gambar 4.10b Translasi obyek lingkaran

Gambar 4.11a menunjukkan program yang menampilkan obyek lingkaran sebelum di skala dan Gambar 4.11b menunjukkan obyek lingkaran setelah di skala.



Gambar 4.11a Obyek lingkaran



Gambar 4.11b Skala obyek lingkaran

4.2 Pembahasan

Pada proses awal, *user* diberikan tampilan menu utama yang sudah dibuatkan ikon-ikon seperti ikon geser ke ke kanan, ikon geser ke kiri, ikon perbesar, ikon tambah lebar, ikon obyek segitiga, ikon obyek lingkaran, menu about, dan menu exit. Berikut akan dijelaskan fungsi dari kcon-ikon operasi grafis sesuai dengan ikon-ikon yang dipilih.

- Ikon geser ke kanan dipilih untuk menggeser obyek ke arah positif dari sumbu x.
- Ikon geser ke kiri dipilih untuk menggeser obyek ke arah negatif dari sumbu x.
- Ikon geser ke atas dipilih untuk menggeser obyek ke arah negatif dari sumbu y.
- Ikon geser ke bawah dipilih untuk menggeser obyek ke arah positif dari sumbu y.
- Ikon perbesar dipilih untuk memperbesar ukuran obyek dengan perbandingan tinggi dan lebar yang sama.

- Ikon perkecil dipilih untuk memperkecil ukuran obyek dengan perbandingan tinggi dan lebar yang sama.
- Ikon tambah lebar dipilih untuk memperbesar ukuran obyek dengan hanya membah lebarnya saja.
- Ikon kurangi lebar dipilih untuk memperkecil ukuran obyek dengan hanya mengurangi lebarnya saja.
- Ikon tambah tinggi dipilih untuk memperbesar ukuran obyek dengan hanya menambah tingginya saja.
- Ikon kurangi tinggi dipilih untuk memperkecil ukuran obyek dengan hanya mengurangi tingginya saja.

4.2.1 Kebutuhan Masukan

Masukan dari sistem aplikasi ini adalah sebagai berikut :

1. Input Obyek

Obyek-obyek yang telah disediakan pada menu awal tampilan yaitu obyek segitiga, obyek kotak, dan obyek lingkaran.

2. Input Operasi

Proses pengoperasian yang dilakukan pada saat obyek dipilih yang terdiri dari proses geser ke kanan, geser ke kiri, geser ke atas, geser ke bawah, perbesar, perkecil, perbesar lebar, perkecil lebar, perbesar tinggi, dan perkecil tinggi.

4.2.2 Kebutuhan Keluaran

Keluaran dari sistem aplikasi operasi grafis ini menampilkan bentuk-bentuk obyek yang telah disediakan. Hasil keluaran diberikan sistem setelah *user* memilih ikon obyek yang akan dioperasikan sesuai dengan keperluan *user*.

4.2.3 Kebutuhan Proses

Kebutuhan proses meliputi beberapa *input* yang dibutuhkan dalam proses yang ada dalam sistem untuk diolah mejadi *output*. Proses yang dibutuhkan pada program operasi grafis ini adalah proses pada saat translasi dan penskalaan agar output yang dihasilkan sesuai dengan kebutuhan pengguna.

4.2.4 Antarmuka Sistem

Antarmuka pemakai atau yang lebih dikenal dengan *user interface* adalah bagian penghubung antara program sistem aplikasi operasi grafis dengan pemakai sistem atau *user*. Pada bagian ini akan terjadi interaksi atau komunikasi antara sistem dengan *user*. Interaksi yang diberikan sistem saat *user* melakukan pemilihan ikon obyek yang diberikan sistem dan sistem akan melakukan pemeriksaan berdasarkan masukan dari *user*.

Setelah proses pemeriksaan selesai, sistem akan mengambil kesimpulan dan memberikan solusi yang akan ditampilkan kepada *user*, apakah input yang diberikan *user* itu benar atau salah.

4.2.5 Analisis Perangkat Lunak

Untuk mengembangkan program alat bantu ajar grafis ini dibutuhkan perangkat lunak, antara lain:

1. Microsoft Windows XP Professional, merupakan sistem operasi yang digunakan dalam pembuatan program.
2. Borland Delphi 7.0, merupakan bahasa pemrograman yang digunakan untuk pengembangan program.
3. Photoshop Cs, perangkat lunak aplikasi yang mengolah data gambar dan grafis.

4.2.6 Analisis Perangkat Keras

Perangkat keras yang digunakan pada saat implementasi dan pembuatan program adalah:

1. Processor Intel Celeron 2,4 GHz
2. Memory DDR 512 MB
3. Harddisk 40 GB
4. VGA Card
5. Monitor
6. Mouse + Keyboard

4.2.7 Kuisisioner

Pada penelitian ini dilakukan kuisisioner untuk mengetahui kelebihan dan kekurangan dari program alat bantu ajar ini. Kuisisioner adalah pengumpulan data melalui serangkaian daftar pertanyaan. Penulis

memberikan beberapa pertanyaan yang dijawab oleh responden. Tabel 4.1 adalah tabel kuisisioner.

Tabel 4.1 Tabel kuisisioner

No.	Pertanyaan	SK	K	B	SB
1.	Tampilan				
2.	Kemudahan				
3.	Kelengkapan				
4.	Manfaat				

Penelitian dilakukan kepada sepuluh responden. Kuisisioner terdiri dari empat pertanyaan yaitu, tampilan, kemudahan, kelengkapan, dan manfaat. R adalah responden (subyek penelitian). Nilai yang diberikan adalah sebagai berikut :

- a. SK (Sangat Kurang) : 1
- b. K (Kurang) : 2
- c. B (Baik) : 3
- d. SB (Sangat Baik) : 4

Dari keseluruhan jawaban responden diperoleh skor dari tiap pertanyaan. Keseluruhan skor tersebut kemudian dijumlahkan sehingga didapatkan skor rata-rata menurut perhitungan berikut ini.

$$\begin{aligned}
 1. \text{ Skor rata-rata tampilan} &= ((0.SK) + (2.K) + (6.B) + (2.SB)) / 10 \\
 &= ((0) + (4) + (18) + (8)) / 10
 \end{aligned}$$

$$= 30 / 10$$

Rata-rata = 3

2. Skor rata-rata kemudahan = $((0.SK) + (1.K) + (5.B) + (4.SB)) / 10$
= $((0) + (2) + (15) + (16)) / 10$
= $33 / 10$

Rata-rata = 3.3

3. Skor rata-rata kelengkapan = $((1.SK) + (2.K) + (6.B) + (1.SB)) / 10$
= $((1) + (4) + (18) + (4)) / 10$
= $27 / 10$

Rata-rata = 2.7

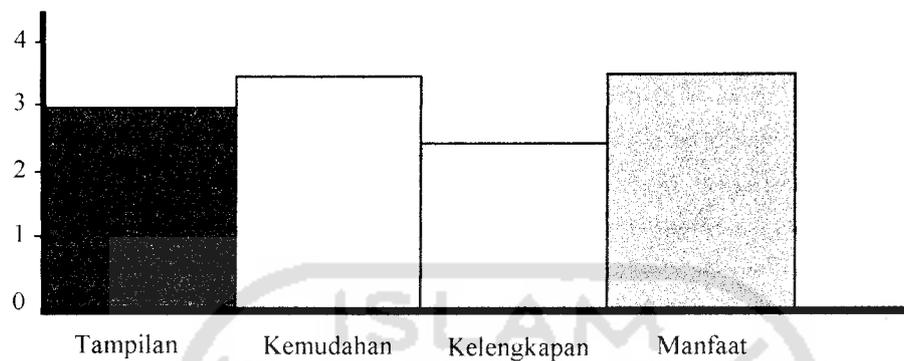
4. Skor rata-rata manfaat = $((0.SK) + (1.K) + (5.B) + (4.SB)) / 10$
= $((0) + (2) + (15) + (16)) / 10$
= $33 / 10$

Rata-rata = 3.3

Hasil dari skor diatas dikelompokkan dalam kategori dibawah ini.

1. Rata-rata ≤ 1 termasuk kategori sangat kurang.
2. $1 < \text{rata-rata} \leq 2$ termasuk kategori kurang.
3. $2 < \text{rata-rata} \leq 3$ termasuk kategori baik.
4. $3 < \text{rata-rata} \leq 4$ termasuk kategori sangat baik.

Gambar 4.12 adalah tampilan diagram kuisisioner.



Gambar 4.12. Diagram kuisisioner

Kesimpulan akhir yang didapat sesuai dengan kategori penilaian adalah :

- Tampilan program ini termasuk dalam kategori baik.
- Kemudahan penggunaan program termasuk dalam kategori sangat baik
- Kelengkapan program termasuk dalam kategori baik.
- Manfaat yang didapatkan dari program ini termasuk dalam kategori sangat baik.

BAB V

SIMPULAN DAN SARAN

5.1 Simpulan

Dari seluruh uraian yang telah dikemukakan dan berdasarkan pada pengamatan, implementasi serta hasil pengujian yang dilakukan pada aplikasi alat bantu ajar operasi-operasi dasar grafis yang telah dibuat, dapat diambil kesimpulan bahwa :

1. Program alat bantu ajar operasi-operasi dasar grafis ini telah selesai dibuat sesuai dengan perencanaan yakni tahap analisis kebutuhan perangkat lunak dengan tambahan beberapa keterbatasan dan asumsi-asumsi yang ditemukan selama tahap implementasi.
2. Program ini dapat dijalankan dan dioperasikan sesuai dengan perintah-perintah yang diinginkan sehingga dapat mempermudah para penggunanya dalam mempelajari dan memahami alat bantu ajar ini.
3. Program ini diharapkan dapat bermanfaat bagi para pengguna yang ingin mempelajari dan mengetahui operasi grafis.

5.2 Saran

Disadari bahwa sistem ini masih banyak kekurangan dan kelemahan, oleh karena itu untuk pengembangan selanjutnya disarankan:

1. Dapat memvisualisasikan program alat bantu ajar operasi-operasi dasar grafis ini dengan lebih memperhalus tampilan grafis yang ada di menu utama.

2. Melengkapi beberapa operasi-operasi grafis lain seperti rotasi, distorsi, shearing dan lain-lainnya.



DAFTAR PUSTAKA

- [GUN05] Guntara, M, *Modul Pratikum Grafika Komputer*, STMIK Akakom, 2005.
- [MAR99] Martina, Inge, *36 Jam Belajar Komputer Delphi 4.0*, Jakarta : Penerbit PT. Elex Media Komputindo, 1999.
- [MUN01] Munir, Rinaldi dan Lidya Leoni. “*Algoritam dan Pemrograman dalam Bahasa Pascal dan C*”, Buku II. Bandung : Jurusan Teknik Informatika ITB, 2001.
- [MUS04] Musalini, Uus, *Membangun Aplikasi Super Cantik dan Full Animasi dengan Delphi*, Jakarta : PT Elex Media Komputindo, 2004.
- [SUS05] Susilo, Djoko, *Grafika Komputer dengan Delphi*, Yogyakarta : Penerbit Graha Ilmu, 2005.
- [WIK07] Wikipedia, *Grafis*, "<http://id.wikipedia.org/wiki/Grafis>", diakses Januari 07.
- [ZUK03] Zuhri, Zainudin, *Dasar-dasar Pemrograman Visual dengan Delphi 6*, Yogyakarta : Penerbit Graha Ilmu, 2003.