

**APLIKASI WEB BROWSER ANIMATIF
MENGUNAKAN DELPHI DAN OPENGL**



Oleh :

TEDDY GIGIH PRABOWO

M3304035

Teknik Komputer

TUGAS AKHIR

Ditulis dan diajukan untuk memenuhi sebagai persyaratan

Memperoleh gelar Ahli Madya Ilmu Komputer

**PROGRAM DIPLOMA III ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SEBELAS MARET
SURAKARTA**

2007

HALAMAN PENGESAHAN

APLIKASI WEB BROWSER ANIMATIF
MENGGUNAKAN DELPHI DAN OPENGL

Yang disusun oleh
TEDDY GIGIH PRABOWO
M 3304035

dibimbing oleh

Pembimbing I

Pembimbing II

Drs. YS. Palgunadi, M.Sc.

Hartono, S.Si

NIP. 131 285 866

NIP. 500 127 069

telah dipertahankan didepan Dewan Penguji
Pada hari Kamis, tanggal 12 Juli 2007
dan dinyatakan telah memenuhi syarat.

Anggota Tim Penguji

Tanda Tangan

1. Drs. Y.S. Palgunadi, M.Sc

1.

2. Hartono, S.Si

2.

3. Drs. Siswanto, M.Si

3.

Surakarta, Juni 2007

Disahkan Oleh

Fakultas MIPA

Dekan,

Ketua Program Studi D-III

Ilmu Komputer,

Prof. Dr. Sutarno, M.Sc P.Hd

Irwan Susanto, DEA

NIP. 131 649 948

NIP. 132 134 694

KATA PENGANTAR

Syukur Alhamdulillah dihaturkan kehadiran Allah SWT, berkat, rahmat, hidayah dan inayah-Nya, pada akhirnya tugas akhir ini dapat diselesaikan. Tugas akhir merupakan syarat akademis untuk memperoleh gelar Ahi Madya (A.Md) di Universitas Sebelas Maret Surakarta. Dalam mengerjakan tugas akhir diambil judul :

Aplikasi Web Browser Animatif Menggunakan Delphi dan OpenGL

Berbagai macam kendala telah ditemui selama pembuatan tugas akhir ini. Namun berkat pertolongan Allah SWT dan dukungan dari teman-teman yang senantiasa membantu dan memberikan semangat untuk terus maju Alhamdulillah dapat teratasi walaupun belum sempurna, tak lupa dari bimbingan dan pengarahan dari Bapak Pembimbing sampai terwujudnya laporan Tugas Akhir ini. Maka dengan segala kerendahan hati disampaikan terima kasih kepada :

1. Bapak Drs. Y.S. Palgunadi, M.Sc selaku pembimbing I yang meluangkan waktunya dan memberikan pengarahan dalam penyusunan tugas akhir ini.
2. Bapak Hartono, S.Si selaku pembimbing II yang senantiasa menuntun dan mengarahkan penulisan susunan tugas akhir tanpa pamrih.
3. Ayah dan Ibu tercinta yang telah mendo'akan serta menuntun dalam setiap waktu hingga selesainya tugas akhir ini.
4. Seluruh Staff Dosen D3 Ilmu Komputer yang selalu siap sedia membantu dan menularkan ilmu pengetahuannya.

Kesempurnaan hanyalah milik Allah SWT. Sebagai hamba Allah, penulis sadar bahwa tugas akhir ini jauh dari sempurna, sehingga perbaikan dan kajian lebih lanjut merupakan pertimbangan yang harus diperhatikan.

Hanya satu harapan penulis, semoga karya kecil ini mampu memberikan manfaat bagi para pembaca.

Surakarta, 23 Juni 2007

Penulis

DAFTAR ISI

HALAMAN PENGESAHAN.....	ii
KATA PENGANTAR.....	iii
MOTTO.....	iv
DAFTAR ISI.....	v
DAFTAR GAMBAR.....	vii
DAFTAR TABEL.....	viii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah.....	2
1.3 Tujuan Penelitian.....	3
1.4 Manfaat Penelitian.....	3
BAB II LANDASAN TEORI.....	4
2.1 Grafika Komputer.....	4
2.1.1 Perangkat Interaktif.....	4
2.1.2 Aplikasi Grafika Komputer.....	5
2.2 Animasi.....	5
2.2.1 Pengaruh Animasi.....	6
2.2.2 Fungsi Animasi dalam Pembelajaran.....	6
2.3 OpenGL.....	8
2.3.1 Menggambar di OpenGL.....	9
2.4 Borland Delphi.....	11
2.5 Aplikasi Browser.....	12
2.6 Dunia 3 Dimensi.....	13
2.6.1 Dua Dimensi dan Tiga Dimensi.....	13
2.6.2 Sistem Koordinat.....	13
2.6.3 Representasi Data Lokasi Titik 3 Dimensi.....	13
BAB III METODOLOGI PENELITIAN.....	14
3.1 Metode Pengumpulan Data.....	14
3.2 Analisa.....	14
3.3 Perancangan.....	14
3.4 Implementasi Awal.....	15

3.5	Evaluasi.....	15
3.6	Implementasi Akhir.....	15
3.7	Alat Penelitian.....	15
BAB IV	HASIL PENELITIAN DAN PEMBAHASAN.....	16
4.1	Analisa Sistem.....	16
4.2	Alternatif Sistem yang Diusulkan.....	16
4.2.1	Pembuatan Aplikasi.....	16
4.2.1.1	Perangkat Pendukung.....	16
4.2.1.2	Packaging.....	17
4.2.1	Pembahasan Program Aplikasi Browser.....	17
4.3	Kelayakan Sistem.....	17
4.4	Perancangan Aplikasi Program.....	17
4.4.1	Struktur Hirarki Program.....	18
4.4.1.1	Desain Form Proses.....	20
4.4.1.2	Menghubungkan delphi dengan OpenGL.....	24
4.4.2	Pembahasan Program Aplikasi Browser.....	26
4.5	Keunggulan dan Kelemahan Program.....	29
BAB V	PENUTUP.....	30
5.1	Kesimpulan.....	30
5.2	Saran.....	30
	DAFTAR PUSTAKA.....	31

DAFTAR GAMBAR

Gambar 2.1 Tampilan Aplikasi Swift 3D.....	8
Gambar 2.2 Mode Menghubungkan Verteks.....	10
Gambar 4.3 Diagram Proses Aplikasi Browser.....	19
Gambar 4.4 Tampilan Program Utama.....	27
Gambar 4.5 Tampilan Program Setelah dieksekusi.....	28

DAFTAR TABEL

Tabel 2.1 Mode pada GLBegin.....	10
----------------------------------	----

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Komputer Grafik telah dikembangkan dan diaplikasikan dengan mengesankan selama beberapa dekade. Dimana perkembangan aplikasi-aplikasi yang menggunakan disiplin ilmu ini telah memimpin teknologi di beberapa bidang seperti komunikasi digital dan internet, penyiaran (*broadcasting*), alat kedokteran, sistem multimedia, biologi, ilmu pengetahuan material, robot, dan manufaktur, sistem *intelligent sensing*, *remote sensing*, seni grafik dan proses print.

Penglihatan manusia adalah salah satu alat yang penting dan sangat kompleks dalam hal mekanismenya. Mata memberikan informasi yang dibutuhkan untuk tugas-tugas yang sederhana secara relatif seperti pengenalan objek dan juga tugas-tugas yang cukup kompleks seperti perencanaan, pembuatan keputusan, riset penelitian, pengembangan kecerdasan manusia. Begitu juga *Computer Graphics and Animation* yang memainkan peran penting dalam pengolahan data berbasis grafik. Dari *image/graphics* manusia dapat melakukan segala aktifitasnya. Sehingga dapat dikatakan bahwa mata merupakan sensor manusia dalam pengambilan *image* dari lingkungan sekitarnya secara *on-line*. Pengembangan pengolahan *image* masih memerlukan waktu yang cukup lama. Para ilmuwan masih berusaha mencari metode yang praktis dan efisien dalam pengolahan *image*. Dan para ilmuwan ingin menciptakan duplikat dari mata yang notabene mata merupakan sensor ciptaan Tuhan yang sangat sempurna dan jauh dari *error*.

Dengan adanya teknologi saat ini, maka banyak jenis pemrograman grafik yang digunakan dalam kehidupan sehari-hari. Kontribusi dalam penelitian ini adalah mencoba merubah sebuah sistem konvensional tersebut menjadi modern yang secara otomatis didukung oleh sebuah komputer. Melalui komputer manusia dapat mengolah grafik menggunakan software yang ada (borland delphi).

Computer Graphics merupakan teknik pemrograman yang dapat digunakan untuk memproses gambar dengan jalan memanipulasinya menjadi data gambar yang diinginkan untuk mendapat informasi tertentu dan telah dapat diaplikasikan dalam berbagai bidang. Keberhasilan aplikasi pengolahan gambar pada beberapa bidang telah banyak kita ketahui. Pada tugas akhir ini kami mensimulasikan grafik (animasi) menggunakan teknik pemrograman dengan delphi dan *library OpenGL*. Maka dengan ini kami dituntut untuk mengembangkan gambar tersebut. Karena pengolahan gambar merupakan salah satu proses *intelligent* dengan fleksibilitas yang sangat tinggi dibidang aplikasi apapun, maka berdasarkan atas hal tersebut kami mencoba untuk melakukan pengenalan *OpenGL*.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah diuraikan maka dapat diangkat perumusan masalah sebagai berikut :

- a. bagaimana cara kita membuat suatu simulasi grafik dengan bahasa pemrograman delphi menggunakan *library OpenGL* yang kemudian dapat dijadikan sebagai pustaka yang menambah ilmu pengetahuan dan simulasi *software* animasi web *portal* yang dapat memanggil alamat URL di *browser*

Dalam penyelesaian tugas akhir ini diberikan beberapa batasan antara lain :

- a. menggunakan bahasa pemrograman IDE (*Integrated Development Environment*)
- b. menggunakan pustaka *library OpenGL*
- c. URL atau alamat *browser* sudah ditentukan

1.3 Tujuan Penelitian

Tujuan dari tugas akhir ini adalah membuat simulasi grafis dengan bahasa pemrograman delphi dan *library opengl* kemudian memperkenalkan *OpenGL* sebagai pustaka program (*program library*) yang menyediakan sejumlah perintah yang berhubungan dengan grafika selain itu juga mampu untuk

mendesain dan mengimplementasikan grafika komputer berbasis delphi dan *library* dari *OpenGL*.

1.4 Manfaat Penelitian

Manfaat penelitian yang diperoleh dari pengerjaan tugas akhir ini antara lain :

- a. dapat mengolah grafik komputer 3D dengan menggunakan Delphi dan tambahan *library OpenGL*.
- b. meningkatkan produktivitas sumber daya manusia khususnya dalam bidang ilmu pengetahuan mengenai grafika komputer.
- c. dapat digunakan sebagai aplikasi tambahan dalam mengolah grafik (dua dimensi dan tiga dimensi).

BAB II

LANDASAN TEORI

2.1 Grafika Komputer

Menurut Hill (1990:2) dalam (Teori dan Praktek Grafika Komputer halaman 1) Grafika Komputer merupakan sekumpulan alat yang digunak untuk membuat gamabr (*to create pictures*) dan berinteraksi dengan gambar dengan cara-cara seperti yang biasa digunakan (*and to interact with them in natural ways*). Peralatan tersebut dapat berupa perangkat keras (*hardware*) ataupun piranti lunak (*software*). Semua peralatan tesebut memberi fasilitas terhadap *programmer* untuk membuat program-program grafis yang baik. Secara umum program-program komputer grafis ditujukan untuk memudahkan interaksi manusi antar komputer. Dengan memvisualkan kata-kat menjadi gambar maka manusia akan lebih mudah berinteraksi dengan komputer, perintah-perintah tidak lagi diberikan dengan cara m,enulis tetapi dengan cara menunjuk area atau gambar tertentu.

2.1.1 Perangkat Interaktif

Perangkat interaktif adalah perangkat yang digunakan untuk memberi perintah kekomputer. Perangkat-perangkat tersebut mengirim data kooridinat atau lokasi ke pemakai ke program grafis. Dengan perangkat interaktif maka pemakai dapat memberikan perintah penggeseran, penskalaan, ataupun perintah-perintah lain.

Perangkat interaktif biasanay menggunakan ruang semu (*virtual space*). Ruang semu adalah sebuah ruang maya yang didefinisikan sebagai ruang kerja (*workspace*) bagi pemakai. Ruang semu pada kenyataannya tidak ada, tetapi pemakai seolah-olah berada dan bergerak dalam batas-batas bidang tersebut. Ruang semu dapat berdimensi dua dan atau tiga.

Perangkat interaktif yang beroperasi dalam bidang dua dimensi misalnya : *Light Pen, Mouse, Trackball dan Digitzer*. sedangkan

perangkat interaktif yang beroperasi dalam ruang tiga dimensi misalnya : *laser scan, joystick*.

2.1.2 Aplikasi Grafika Komputer

Aplikasi grafika komputer adalah program komputer yang dirancang khusus untuk menghasilkan grafis atau representasi visual dari data kedalam bentuk gambar, slide dan sebagainya (Hill, 1990).

Ruang lingkup aplikasi grafika komputer sangat luas mencakup dari sekedar memvisualkan data sampai dengan proses interaktif. Selain itu, ruang lingkup aplikasi grafika komputer mulai dari bidang dua dimensi sampai tiga dimensi, mulai dari pemodelan benda sederhana sampai ke representasi data geografis.

2.2 Animasi

Animasi pada dasarnya adalah suatu objek yang seolah-olah bergerak sehingga terjadi pergerakan yang membuat mata kita memandangnya sebagai sebuah kesatuan utuh. Padahal di balik itu, animasi dibuat berdasarkan pada bagian-bagian tertentu (terpisah) dan baru kemudian akan disatukan untuk mendapatkan pergerakan yang sempurna.

Untuk membuat animasi, terlebih dahulu kita harus membuat satu per satu bagian-bagian tertentu tadi. Hal ini merupakan hal yang sangat berat dalam membuat animasi mengingat kita harus memikirkan bagaimana desain atau bagian dari animasi yang kita buat tersebut supaya benar-benar mendapatkan hasil yang sempurna jika disatukan.

Perkembangan teknologi komputer demikian pesatnya, yang memiliki fungsi awal sebagai alat bantu dalam menyelesaikan persoalan dan masalah dalam segala bidang kemudian memasuki fungsi sebagai penghibur. Animasi merupakan aplikasi yang banyak digunakan dalam berbagai bidang, diantaranya dalam dunia perfilman dan periklanan. Dengan menggunakan aplikasi animasi, seorang *creator* dapat bergerak dalam dunia nyata. Seseorang dapat mencurahkan

pikirannya dengan membuat berbagai macam aplikasi. Contoh aplikasi, animasi antara lain adalah *Extreme3D*, *AnimatorPro*, *Swift MX*, dll.

2.2.1 Pengaruh Animasi

Pengaruh animasi dalam kehidupan seseorang dapat terjadi karena dengan melihat suatu gerakan animasi seseorang akan berimajinasi. Dan dengan imajinasi itu seseorang akan dapat membuat hal yang baru. Berikut ini adalah pengaruh animasi terhadap manusia

A. Pengaruh Positif

Pengaruh positif animasi terhadap manusia adalah :

- a. mendidik
- b. menghibur
- c. menjadikan seseorang dapat menciptakan sesuatu hal yang baru. (penemuan)
- d. dapat digunakan sebagai media/alat presentasi kita agar orang dapat lebih memahami. Karena orang akan lebih cepat memahami bila disalurkan dengan berbagai gerakan (animasi)

B. Pengaruh Negatif

Pengaruh/efek negative dari animasi adalah :

- a. cenderung menjadi suatu candu bagi anak-anak (film kartun)
- b. hal-hal yang ada didalamnya adalah semu
- c. diperlukan keahlian khusus untuk membuatnya

2.2.2 Fungsi animasi dalam pembelajaran

Terdapat tiga jenis format animasi.

1. Animasi tanpa sistem kontrol, animasi ini hanya memberikan gambaran kejadian sebenarnya (*behavioural realism*), tanpa ada kontrol sistem. Misal untuk pause, memperlambat kecepatan pergantian frame, Zoom in, Zoom Out dll.
2. Animasi dengan sistem kontrol, animasi ini dilengkapi dengan tombol kontrol. Misal tombol untuk pause, zoom in, zoom out, dll

3. Animasi manipulasi langsung (*Direct-manipulation Animation (DMA)*). DMA menyediakan fasilitas untuk pengguna berinteraksi langsung dengan control navigasi (misal tombol dan slider). Pengguna bebas untuk menentukan arah perhatian. Menekan tombol atau menggeser slider akan menyebabkan perubahan keadaan. Hasilnya dapat langsung dilihat dan kejadiannya dapat diulang-ulang.

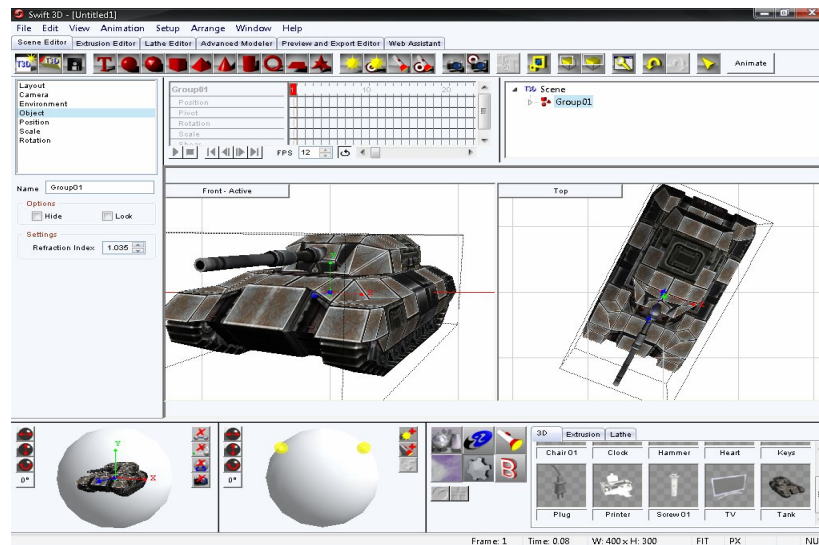
Selama ini animasi digunakan dalam media pembelajaran untuk dua alasan. Pertama, untuk menarik perhatian siswa dan memperkuat motivasi. Animasi jenis ini biasanya berupa tulisan atau gambar yang bergerak-gerak, animasi yang lucu, aneh yang sekiranya akan menarik perhatian siswa. Animasi ini biasanya tidak ada hubungannya dengan materi yang akan diberikan kepada murid. Fungsi yang kedua adalah sebagai sarana untuk memberikan pemahaman kepada murid atas materi yang akan diberikan.

Efektifitas animasi dalam pembelajaran tidak hanya berhubungan dengan bagaimana animasi itu diterima dan dikonsepskan, namun juga bagaimana animasi tersebut dirancang.

Animasi yang tidak dilengkapi sistem kontrol memiliki kelemahan, bisa jadi animasi terlalu cepat, pengguna tidak memiliki waktu yang cukup untuk memperhatikan detail tertentu karena tidak ada fasilitas untuk *pause* dan *zoom in*.

Animasi dengan sistem kontrol memungkinkan pengguna untuk menyesuaikan animasi dengan kapasitas pemrosesan informasi mereka. Namun hal ini pun masih memiliki kekurangan, penelitian menunjukkan bahwa kurangnya pengetahuan awal (*prior knowledge*) atas materi yang dipelajari menyebabkan murid tidak tahu mana bagian yang penting dan harus diperhatikan guna memahami materi dan yang tidak. Seringkali murid lebih memperhatikan bagian yang tampak lebih menonjol secara perseptual

(*Dina Utami : 2007*)



Gambar 2.1 Tampilan Aplikasi Swift 3D

2.3 Opengl

Opengl merupakan kepanjangan dari *Open Graphic Library*. *Opengl* diproduksi oleh Silicon Graphics dan pada awalnya ditujukan hanya untuk sistem komputer mereka, tetapi dalam perkembangannya, *Opengl* diterima menjadi salah satu bakuan (*standart*) dalam bakuan grafika komputer. *Opengl* merupakan pustaka program (*library program*) yang menyediakan sejumlah perintah yang berhubungan dengan grafika. *Opengl* adalah suatu *Graphics Library 3D powerful* yang multiplatform. Artinya bila anda memprogram dengan *Opengl*, anda dapat dengan mudah mengcompile program anda di berbagai platform dengan *source code* yang nyaris tidak berubah. *Opengl* saat ini sudah dikenal dengan luas dan sering dipakai dalam pembuatan game-game tiga dimensi yang canggih-canggih.

Antarmuka antara program yang akan memanfaatkan *Opengl* disediakan melalui sejumlah *Application Program Interface (API)*. Melalui API tersebut pemrogram dapat memerintahkan *Opengl* melakukan sesuatu tindakan tertentu, dan mengingat *Opengl* telah diimplementasikan kedalam berbagai sistem komputer maka pemrogram akan dapat menggunakan API yang sama dengan hasil yang sama untuk berbagai sistem komputer. Dengan demikian pemrogram

dapat berkonsentrasi pada penyelesaian masalahnya dan tidak lagi direpotkan dengan urusan-urusan implementasi teknis.

Didalam sistem operasi Windows, *Opengl* diimplementasikan kedalam 2 buah file, yaitu : *Opengl32.dll* dan *glut32.dll*. *Opengl32.dll* merupakan pustaka program yang menyediakan perintah-perintah utama dari *Opengl* sedangkan *glut32.dll* menyediakan perintah tambahan (*utility*) dari *Opengl*.

2.3.1 Menggambar di OpenGL

Semua proses menggambar didalam OpenGL harus diletakkan diantara perintah `glBegin` dan `glEnd` seperti pada listing dibawah ini.

```
glBegin<mode gambar>;
    <perintah-perintah menggambar>
glEnd;
```

Beberapa perintah yang berhubungan dengan menggambar pixel di OpenGL antara lain :

glBegin (mode)

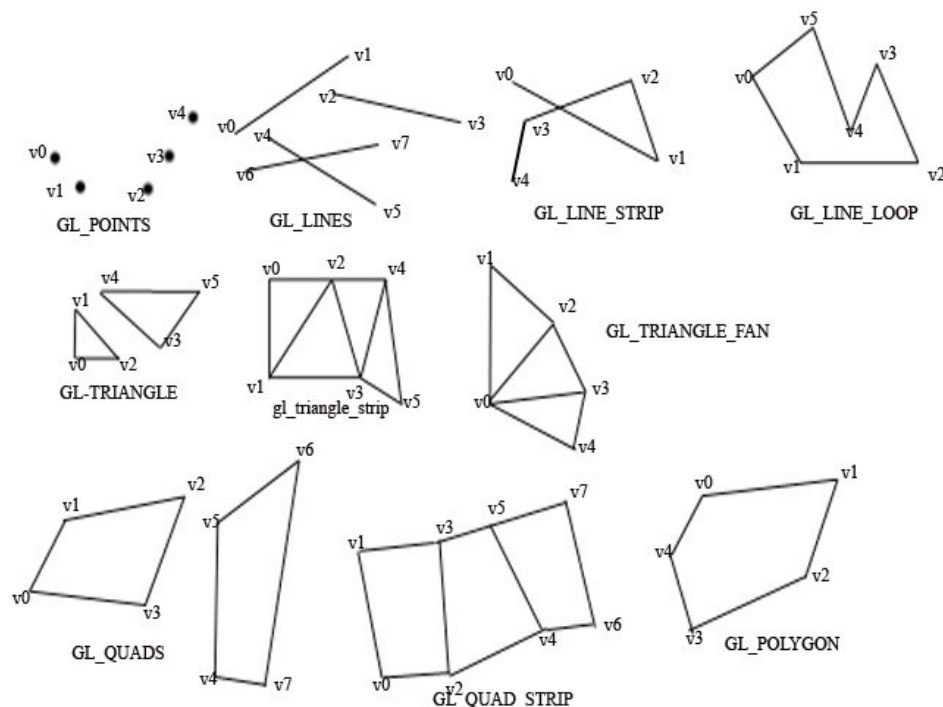
Perintah mengawali perintah untuk menggambar. Mode merupakan konstanta yang menyatakan bagaimana *opengl* harus menghubungkan titik/verteks yang akan digambar.

Tabel 2.1 Mode pada GLBegin

Nilai	Arti
GL_POINTS	Setiap verteks diperlakukan sebagai titik terpisah
GL_LINES	Dua pasang verteks diperlakukan sebagai garis
GL_LINE_STRIP	Sama seperti GL_LINES tetapi setiap garis saling dihubungkan
GL_LINE_LOOP	Sama seperti GL_LINES_STRIP tetapi vertek pertamadan terakhir membentuk garis pula
GL_TRIANGLES	Tiga pasang verteks dianggap sebagai bidang segitiga

GL_TRIANGLES_STRIP	Bidang segitiga yang saling berhungan
GL_TRIANGLES_FAN	Mirip GL_TRIANGLE_STRIP tetapi semua bidang menggunakan satu verteks yang sama
GL_QUADS	Empat verteks dianggap sebagai polygon empat sisi (<i>quadrilaterals</i>)
GL_QUAD_STRIP	Pasangan <i>quadrilaterals</i>
GL_POLYGON	Verteks dianggap sebagai titik sudut polygon

Hubungan antara masing-masing mode diperlihatkan pada gambar 2.1



Gambar 2.2 Mode menghubungkan verteks

glColor3f(Red,Green,Blue:GLFloat)

Perintah untuk menyatakan warna depan/warna objek yang akan digambar. Nilai Red,Green,Blue berkisar 0 sampai dengan 1.

glVertex2f(x,y:GLFloat)

Perintah untuk menggambar sebuah verteks dilayar dengan menggunakan koordinat 2 dimensi. Pada perintah ini koordinat z dianggap sama dengan 0

glVertex3f (x,y,z:GLfloat)

Perintah untuk menggambar sebuah verteks dilayar dengan menggunakan koordinat 3 dimensi. Sebagai contoh, perintah glVertex3f(1.0, 2.0, 3.0) akan menggambar verteks di lokasi $x = 1.0$, $y = 2.0$ dan $z = 3.0$

glVertex4f (x,y,z,w:GLfloat)

Sama seperti glVertex3f tetapi nilai x,y,z yang sebenarnya merupakan hasil pembagian dengan parameter w. Sebagai contoh, perintah glVertex4f(1.0, 2.0, 3.0, 2.0) akan menggambar vertex di $x = 1.0/2.0 = 0.5$, $y = 2.0/2.0 = 1.0$ dan $z = 3.0/2.0 = 1.5$.

glFlush

Pada saat OpenGL menerima perintah menggambar, OpenGL akan menggambar di buffer internalnya dan memindahkan isi buffer ke layar saat menerima perintah glFlush. Hal ini ditujukan agar implementasi OpenGL tidak tergantung kepada system yang digunakan.

(Edhi Nugroho, Teori dan Praktek Grafika Komputer : 2005)

2.4 Borland Delphi

Borland delphi adalah kompiler / penterjemah bahasa delphi (awalnya dari Pascal) yang merupakan bahasa tingkat tinggi sekelas dengan Basic, C. Bahasa Pemrograman di Delphi disebut bahasa prosedural artinya bahasa/sintaknya mengikuti urutan tertentu / prosedur. Ada jenis pemrograman *non*-prosedural seperti pemrograman untuk kecerdasan buatan seperti bahasa prolog. Delphi termasuk keluarga visual sekelas Visual Basic, Visual C, artinya perintah-perintah untuk membuat objek dapat dilakukan secara visual. Pemrogram tinggal memilih objek apa yang ingin dimasukkan kedalam *Form/Window*, lalu tingkah laku objek tersebut saat menerima event/aksi tinggal dibuat programnya.

Delphi merupakan bahasa berorientasi objek, artinya nama objek, properti dan *methode/procedure* dikemas menjadi satu kemasan (*encapsulate*). Sebelum

mempelajari ketiga struktur pemrograman ada baiknya kenali dahulu tampilan IDE, yang merupakan editor dan tools untuk membuat program Delphi. Pada IDE akan ditampilkan *Form* baru yang merupakan aplikasi/program window yang akan dibuat. Aplikasi / program berbasis windows sering disebut dengan jendela (*window*). Bagaimana membuat aplikasi berbasis windows (berbasis grafik dan bukan berbasis teks seperti pada DOS)? Caranya dengan membuat sebuah *form*. Pada pemrograman berbasis windows, kita akan dihadapkan pada satu atau beberapa jendela yang nampak dihadapan kita. Jendela ini dalam Delphi disebut juga dengan *form*. Delphi merupakan perangkat lunak (bahasa pemrograman) untuk membuat program / aplikasi komputer berbasis windows. Ciri sebuah objek adalah memiliki nama, properti dan *method/procedure*. Delphi disebut juga *visual programming* artinya komponen-komponen yang ada tidak hanya berupa teks (yang sebenarnya program kecil) tetapi muncul berupa gambar-gambar.

Borland Delphi memiliki kelebihan antara lain, memiliki VCL (*Visual Component Language*) yang lengkap, baik untuk kebutuhan sistem, hubungan dengan data, hubungan dengan internet dan lain-lainnya.

2.5 Aplikasi Browser

Aplikasi browser adalah aplikasi yang digunakan oleh *user* agar dapat membuka alamat web yang ditujunya. Pada program yang dibuat browser akan tergantung pada komputer. Aplikasi hanyalah berfungsi sebagai *link* menuju alamat web yang terlihat pada ikon jendela dari program aplikasi.

(Syafi'e : 2007)

2.6 Dunia 3 Dimensi

2.6.1 Dua Dimensi dan Tiga Dimensi

Perbedaan yang mendasar antara dua dimensi dan tiga dimensi adalah kedalaman. Kedalaman didefinisikan sebagai jarak antara *viewer* terhadap benda yang terlihat. Berbeda dengan dua dimensi yang hanya menggunakan 2 ukuran, yaitu panjang dan lebar, maka tiga dimensi

menggunakan 3 ukuran, yaitu : panjang, lebar dan kedalaman. Secara geometri ketiga ukuran tersebut disimbolkan dengan sumbu x,y, dan z.

2.6.2 Sistem Koordinat

Dalam menyatakan sistem koordinat 3 dimensi, terdapat dua cara yaitu aturan tangan kiri dan aturan tangan kanan. Pada dasarnya kedua aturan tersebut adalah kemana sumbu x^+ menunjuk. atau Aturan tangan kiri, sumbu x^+ ditunjuk oleh ibu jari dan mengarah kekanan sedangkan Aturan tangan kanan, sumbu x^+ menuju arah kiri. Pada kedua aturan tersebut, sumbu y^+ maupun z^+ menuju kearah yang sama.

2.6.3 Representasi Data Lokasi Titik 3 Dimensi

Untuk menyimpan informasi mengenai lokasi sebuah titik 3D menggunakan komputer kita dapat mendefinisikan struktur data seperti pada listing dibawah ini :

Listing struktur data titik 3 dimensi

```
type
TPoint3D = record
    x , y , z : real
end;
```

(Edhi Nugroho, Teori dan Praktek Grafika Komputer : 2005)

BAB III

METODOLOGI PENELITIAN

Metode penelitian dilakukan untuk mendapatkan pemecahan masalah yang dihadapi. Dalam merancang aplikasi ini diambil beberapa langkah yaitu dengan mempelajari dasar-dasar pemrograman program serta hal-hal yang berhubungan dengan program.

Kemudian dirancang beberapa hal dalam membuat program ini yaitu sebagai berikut :

- a. membuat dan mengembangkan aplikasi animasi interaktif dan mencari alternatif pemecahannya
- b. mempelajari sistem aplikasi dan implementasinya
- c. menghubungkan Delphi dengan OpenGL serta bagaimana menggunakannya

Untuk memperoleh hasil yang tepat dan akurat guna mendapatkan hasil program yang baik, digunakan beberapa metode penelitian.

3.1 Metode Pengumpulan Data

Dalam melakukan pembuatan aplikasi ini dilakukan dengan mengembangkan metode *study* pustaka. Sebagai landasan teoritis, dikumpulkan data melalui *study* literatur, yaitu dengan membaca dan mengumpulkan bahan – bahan teori yang diperlukan dari beberapa sumber/literatur yang berkaitan dengan penelitian dalam penyusunan laporan.

3.2 Analisa

Analisa meliputi analisa sistem dan analisa kebutuhan perangkat lunak yang akan dibangun.

3.3 Perancangan

Pada tahap ini dibuat struktur hirarki, desain form proses, dan perancangan aplikasi *browser*.

3.4 Implementasi Awal

Aplikasi akan diimplementasikan ke dalam bentuk program berdasarkan hasil analisa dan perancangan yang telah diperoleh dari tahap sebelumnya.

3.5 Evaluasi

Aplikasi yang telah selesai diimplementasikan akan dievaluasi, dimana akan dilakukan koreksi dan penyempurnaan program apabila diperlukan.

3.6 Implementasi Akhir

Aplikasi telah selesai dibangun dan siap untuk diserahkan kepada end user.

3.7 Alat Penelitian

Dalam melakukan penelitian aplikasi ini digunakan beberapa alat, agar program dapat berjalan dengan baik dan memudahkan *user* dalam membuat dan mengkreasikan program yang akan dibuatnya.

- A. Perangkat lunak yang digunakan.
 - a. Borland Delphi yang mendukung pustaka *library OpenGL*
 - b. Microsoft Windows Xp
 - c. Swift 3D untuk melihat beberapa model 3D guna mendapatkan inspirasi dalam membuat program
- B. Perangkat keras

Perangkat keras yang digunakan dalam pembuatan aplikasi ini adalah komputer dengan spesifikasi sebagai berikut :

 - a. prosesor : Intel Pentium IV 2.66 Ghz
 - b. memori : 256 MB
 - c. *harddisk* : 40 Gb
 - d. VGA : 64 MB
 - e. *display* : 1024 x 768

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

4.1 Analisa Sistem

Analisa sistem adalah penguraian dari suatu informasi yang utuh ke dalam bagian-bagian komponen, dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, hambatan-hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikan

4.2 Alternatif Sistem yang Diusulkan

Alternatif sistem yang diusulkan adalah membuat sebuah program aplikasi perangkat lunak yang berhubungan dengan grafik dalam dunia IT. Penyusunan program aplikasi ini adalah dengan menggunakan Borland Delphi dan *library OpenGL* dengan memanfaatkan pustakanya.

4.2.1 Pembuatan Aplikasi

4.2.1.1 Perangkat Pendukung

Berikut adalah perangkat yang digunakan pada tahap pembuatan aplikasi.

A. Perangkat Keras

Perangkat keras yang digunakan dalam pembuatan aplikasi ini adalah komputer dengan spesifikasi :

prosesor : Intel Pentium IV 2.66 Ghz

memori : 256 Mb

harddisk : 40 Gb

vga : 64 MB

monitor : 17 inch

keybord dan mouse

B. Perangkat Lunak

Perangkat lunak yang digunakan untuk pembuatan aplikasi ini adalah Borland Delphi 7.0 yang telah di-*update* sehingga dapat mendukung *OpenGL*.

4.2.1.2 *Packaging*

Aplikasi pada dibuat menggunakan borland delphi 7.0 harus dikompilasi terlebih dahulu sebelum dapat digunakan oleh *user* pada komputer pribadinya menjadi .exe

4.3 Kelayakan Sistem

Dalam hal ini diberikan sebuah gagasan kepada para pelaku dunia komputer, untuk menggunakan sistem aplikasi ini sebagai media untuk melakukan proses browsing.

Menyusun sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan, tidak lain bertujuan agar sistem mampu berjalan dengan baik. Alasan yang melandasi perlunya pengembangan sistem adalah :

- a. memperkenalkan *OpenGL* sebagai pustaka program (*program library*) yang menyediakan sejumlah perintah yang berhubungan dengan grafika komputer.
- b. mendesain dan mengimplementasikan grafika komputer berbasis delphi dan library dari *OpenGL*.
- c. tuntutan akan hasil yang cepat dan akurat

4.4 Perancangan Aplikasi Program

Pada rancangan aplikasi program, dibuat rancangan struktur hirarki program dan *Flowchart* program tampilan serta bagaimana cara menghubungkan delphi dengan opengl. Karena pada intinya, dalam membuat aplikasi banyak sekali menggunakan perintah-perintah yang berhubungan dengan *opengl*.

4.4.1 Struktur Hirarki Program

Aplikasi yang digunakan adalah menampilkan simulasi Aplikasi web *browser* dengan menggunakan program Delphi yang menggunakan pustaka *library OpenGL* secara sederhana. Dengan Borland Delphi program dapat ditampilkan dalam bentuk screen tampilan yang baik. Dalam membuat aplikasi ini menggunakan Delphi 7.0 karena didalam Delphi tersebut sudah tersedia dan mendukung *library OpenGL*

Tampilan program yang dibuat terdiri dari 2 buah jendela. Jendela pertama adalah tampilan utama dari program aplikasi *browser* animatif. Sedangkan jendela yang kedua adalah tampilan *browser* Internet Explorer maupun Mozilla Firefox (tergantung *browser* yang diinstall pada komputer) yang secara langsung membuka dengan sendirinya.

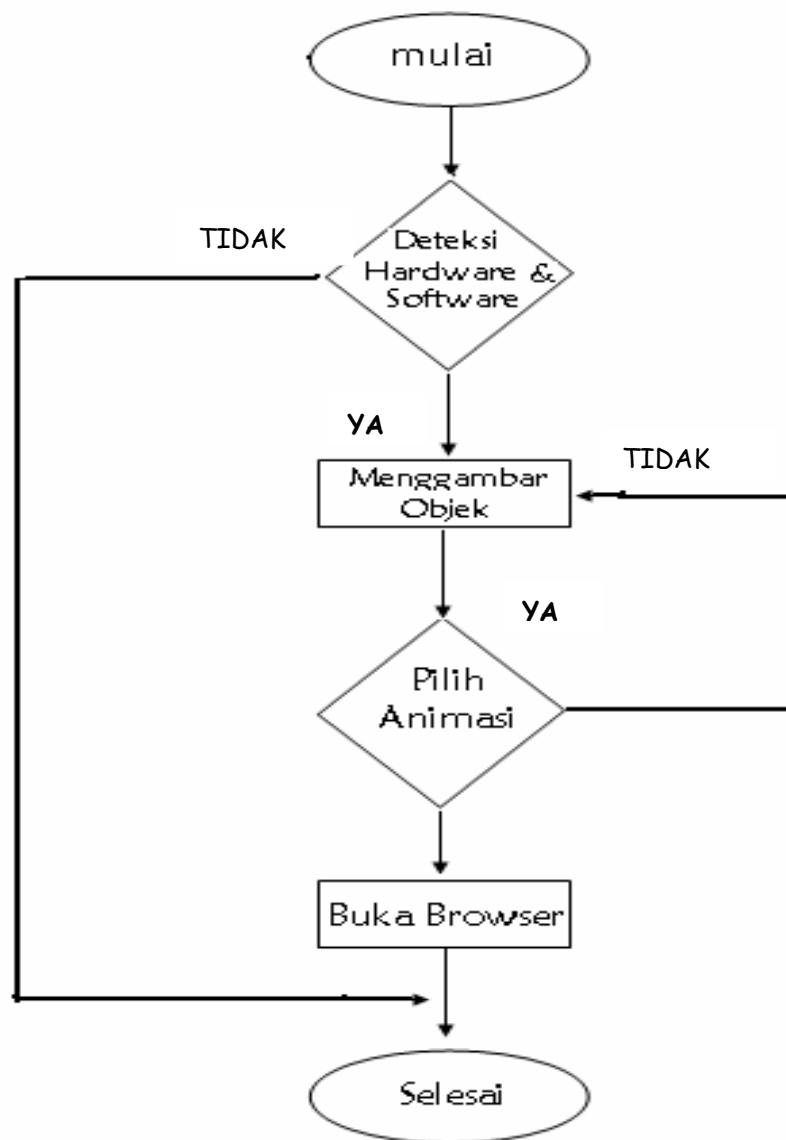
Untuk menjalankan program aplikasi yang dibuat masih diperlukan program lain. Program tersebut adalah internet explorer atau semacamnya yang sudah menjadi bawaan dari sistem operasi yang digunakan.

Rancangan *sistem* :

Proses dari program yang dibuat antara lain :

- a. mulai menjalankan program
- b. program secara otomatis mendeteksi *hardware* dan *software* yang digunakan
- c. menggambar objek kedalam kanvas (rendering animasi)
- d. *user* memilih animasi (salah satu ikon web yang sudah ditentukan)
- e. jika Ya, membuka *browser* yang sudah ditentukan jika Tidak, animasi masih akan tetap berjalan,
- f. selesai.

Berikut adalah diagram proses program yang telah dibuat :



Gambar 4.3

Diagram Proses Aplikasi Arowser

4.4.1.1 Desain Form Proses

Proses dalam pembuatan aplikasi *browser* animatif ini menggunakan bahasa pemrograman delphi dengan pustaka library *OpenGL*. Berikut ini adalah prosedur-prosedur yang digunakan

1. glDraw

Digunakan untuk menggambarkan obyek menggunakan *OpenGL*.

```
procedure glDraw();
var z : GLfloat;
    I : Integer;
begin
    glClear(GL_COLOR_BUFFER_BIT or GL_DEPTH_BUFFER_
    BIT);
    glLoadIdentity();
    glTranslatef(21,0.0,-39);
    glRotatef(30, 0.0,-1.0, 0.0);
    Angle := Angle + ElapsedTime/250;
    Angle :=Angle + Speed;
    if Angle < 30 then
    begin
        Angle :=30;
        Speed :=0;
    end;
end;
```

2. glInit

Inisialisasi terhadap *OpenGL*

```
procedure glInit();
begin
    glClearColor(0.0, 0.0, 0.0, 0.0);
    glShadeModel(GL_SMOOTH);
    glClearDepth(1.0);
    glEnable(GL_DEPTH_TEST);
    glDepthFunc(GL_LESS);
    Test To Do
    glHint(GL_PERSPECTIVE_CORRECTION_HINT,
    GL_NICEST);
```

```

    glEnable(GL_TEXTURE_2D);
    LoadTexture('tunnel21.bmp', TunnelTex, FALSE);
    LoadTexture('windows.jpg', Banners[1], FALSE);
    LoadTexture('mipa.jpg', Banners[2], FALSE);
    LoadTexture('ilkom.jpg', Banners[3], FALSE);
    LoadTexture('google.jpg', Banners[4], FALSE);
    LoadTexture('yahoo.jpg', Banners[5], FALSE);
    LoadTexture('fast.jpg', Banners[6], FALSE);
    LoadTexture('plasa.jpg', Banners[7], FALSE);
    LoadTexture('friend.jpg', Banners[8], FALSE);
    LoadTexture('klik.jpg', Banners[9], FALSE);
    Angle :=100;
    Speed :=0;
    MyQuadratic := gluNewQuadric();
    gluQuadricNormals(MyQuadratic, GLU_SMOOTH);
    gluQuadricTexture(MyQuadratic, GL_TRUE);
end;

```

3. glResizeWnd

Menampilkan ukuran kanvas

```

procedure glResizeWnd(Width, Height : Integer);
begin
    if (Height = 0) then
        Height := 1;
    glViewport(0, 0, Width, Height);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluPerspective(45.0, Width/Height, 1.0, 100.0);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
end;

```

4. ProcessKey

Pengaturan navigasi gerakan animasi dan kecepatannya

```

procedure ProcessKeys;
begin

```

```

    if (keys[VK_UP])      then speed := speed + 0.005;
    if (keys[VK_DOWN])    then speed := speed - 0.005;
    if (keys[VK_RIGHT])   then speed := speed - 0.005;
    if (keys[VK_LEFT])    then speed := speed + 0.005;
end;

```

5. MouseClick

Pemilihan ikon untuk menuju url

```

procedure MouseClick(X, Y : Integer);
var selectBuff : Array[0..63] of GLuint;
    viewport : Array[0..3] of GLuint;
    hits, SelectedName : GLuint;
    I : Integer;
    URL : String;
    Hndl : HWnd;
begin
    glGetIntegerv(GL_VIEWPORT, @viewport);
    glSelectBuffer(23, @selectBuff);
    glMatrixMode(GL_PROJECTION);
    glPushMatrix();
    glRenderMode(GL_SELECT);
    Selection := TRUE;
    glInitNames();
    glPushName(0);
    glLoadIdentity();
    gluPickMatrix(x, viewport[3]-y-27, 1, 1,
@viewport);
    gluPerspective(45.0, viewport[2]/viewport[3],
0.0, 100.0);
    glMatrixMode(GL_MODELVIEW);
    glDraw();
    hits := glRenderMode(GL_RENDER);
    if (hits > 1) then
    begin
        URL := BannerURL[selectBuff[hits*3+1]];
    end;

```

```

        Hndl:=ShellExecute(h_Wnd, 'open', PChar(URL), nil, nil,
        SW_SHOWNOACTIVATE);
        ShowWindow(Hndl, SW_SHOWNOACTIVATE);
    end;
    glMatrixMode(GL_PROJECTION);
    glPopMatrix();
    Selection :=FALSE;
    glMatrixMode(GL_MODELVIEW);
end;

```

6. glKillWnd

Menutup *OpenGL* dengan Delphi

```

procedure glKillWnd(Fullscreen : Boolean);
begin
    if Fullscreen then
    begin
        ChangeDisplaySettings(devmode(nil^), 0);
        ShowCursor(True);
    end;
    if (not wglMakeCurrent(h_DC, 0)) then
        MessageBox(0, 'Release of DC and RC failed!',
        'Error', MB_OK or MB_ICONERROR);
    if (not wglDeleteContext(h_RC)) then
    begin
        MessageBox(0, 'Release of rendering context
        failed!', 'Error',
        MB_OK or MB_ICONERROR);
        h_RC := 0;
    end;
    if ((h_DC = 1) and (ReleaseDC(h_Wnd, h_DC) <>
    0)) then
    begin
        MessageBox(0, 'Release of device context
        failed!', 'Error', MB_OK
        or MB_ICONERROR);
    end;
end;

```

```

        h_DC := 0;
    end;
    if ((h_Wnd <> 0) and (not DestroyWindow(h_Wnd)))
    then
    begin
        MessageBox(0, 'Unable to destroy window!',
        'Error', MB_OK or
        MB_ICONERROR);
        h_Wnd := 0;
    end;
    if (not UnRegisterClass('OpenGL', hInstance))
    then
    begin
        MessageBox(0, 'Unable to unregister window
        class!', 'Error', MB_OK or MB_ICONERROR);
    end;
end;

```

4.4.1.2 Menghubungkan Delphi dengan *OpenGL*

Untuk menghubungkan Delphi dengan *OpenGL* maka sebelumnya kita harus mengetahui apakah delphi kita sudah menyediakan file *OpenGL.dcu* di ...\\lib. Jika sudah tersedia maka cukup dengan menambahkan klausa *OpenGL* di perintah *uses* seperti terlihat pada listing dibawah ini :

```

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls,
    Forms, Dialogs, OpenGL

```

Apabila versi Delphi yang digunakan tidak menyediakan file *OpenGL.dcu* maka terpaksa harus mendownload sendiri file tersebut sesuai dengan versi delphi yang digunakan.

Karena *OpenGL* bukan merupakan dari bagian *sistem* operasi maka kita perlu memberi tahu windows bahwa program yang

bakan dibuat akan memanfaatkan API dari *OpenGL*. Informasi yang perlu disampaikan adalah informasi mengenai

- a. Format pixel yang disampaikan melalui struktur data *TPixelFormatDescriptor*. Format pixel memberikan informasi mengenai jumlah warna dalam bit per pixel, kedalam sumbu z, serta tipe pixel
- b. *Device Context* yang menyatakan perangkat (*device*) yang akan digunakan oleh *OpenGL* untuk menggambar dilayar. Antara perangkat yang disediakan oleh windows dan perangkat yang disediakan oleh *OpenGL* harus dihubungkan sehingga apa yang digambar *OpenGL* akan ditampilkan Windows.

Tahapan untuk menggunakan *OpenGL* dapat dirumuskan dalam empat langkah berikut ini antara lain :

- a. menentukan format pixel
- b. membuat *device* context untuk me-render / menggambar
- c. mengaktifkan *device* context yang dibuat pada langkah 2
- d. gambar pixel

Menghubungkan Delphi dengan *OpenGL* :

type

```
Tform1 = class (TForm)
  Procedure FormCreate(Sender : TObject);
  Procedure FormPaint(Sender : TObject);
  Private
    { Private declaration }
    Pfd:TPixelFormatDescriptor;
    glDC : HDC;
    glContext:HGLRC;
    procedure InitOpenGL(glDC:HDC);
```

public

```
{ Public declaration }
```

end ;

Atribut *pfid* merupakan atribut yang digunakan untuk menyimpan informasi pixel, sedangkan *glDC* merupakan atribut untuk menyimpan *device context* yang dibuat oleh windows dan atribut *glContext* digunakan untuk menyimpan *device context* yang dibuat oleh *OpenGL*. Antara *device context* yang dibuat melalui windows (*glDC*) dan *device context* yang dibuat melalui *OpenGL* nantinya akan dihubungkan melalui perintah *wglMakeCurrent*.

4.4.2 Pembahasan program aplikasi *browser*

Program aplikasi *browser* ini dibuat untuk menambah ilmu kita sebagai study lanjut sebagai orang awam tentang grafika komputer. Program aplikasi ini dibuat dengan menggunakan bahas pemrograman yang berorientasi objek yaitu Borland delphi dengan pustaka library *OpenGL* sehingga orang akan mudah mengenalinya.

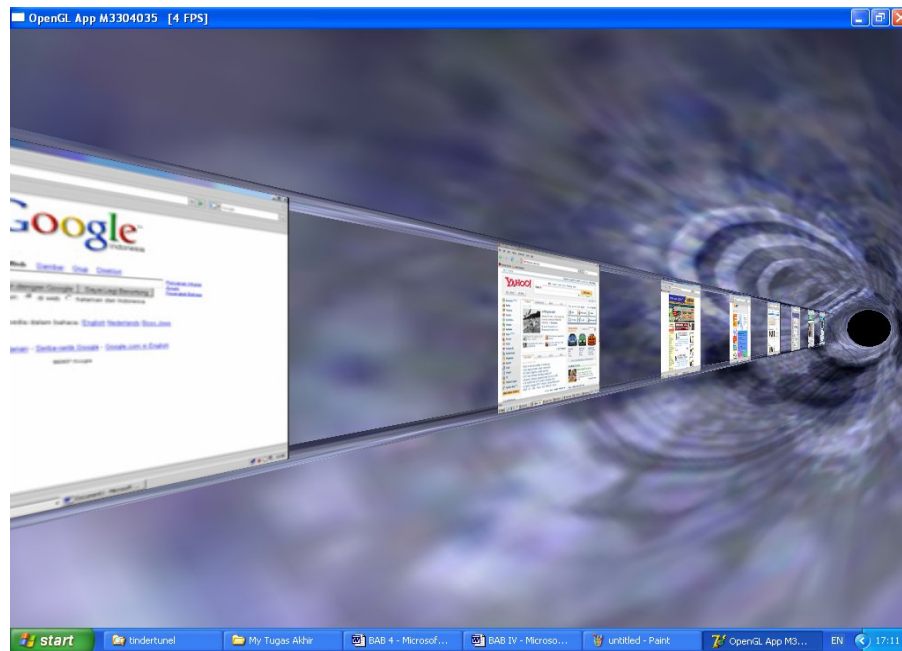
Untuk menjalankan keseluruhan program maka diperlukan koneksi internet dan url dalam program aplikasi ini sudah ditentukan.

Berikut penjelasan mengenai program aplikasi *browser*

A. Program Utama

Pada tampilan utama program aplikasi *browser* adalah tampilan animasi ikon-ikon yang bergerak dari dalam menuju keluar. Gerakan tersebut dapat dikontrol (kecepatan gerakan) dengan menggunakan tombol navigasi yang ada pada *keyboard*.

Berikut adalah tampilan utama dari program yang telah dibuat :



Gambar 4.4 Tampilan Program Utama

B. Input Proses

Input proses dijabarkan sebagai proses menjalankan program yang telah dibuat. Program *browser* web yang telah dibuat diasumsikan sebagai berikut :

1. program dijalankan, *user* memilih ikon (*link*) yang tersedia
2. setelah diklik akan langsung membuka internet explorer (tergantung pada komputer) kemudian menuju ke alamat web yang telah dipilih.

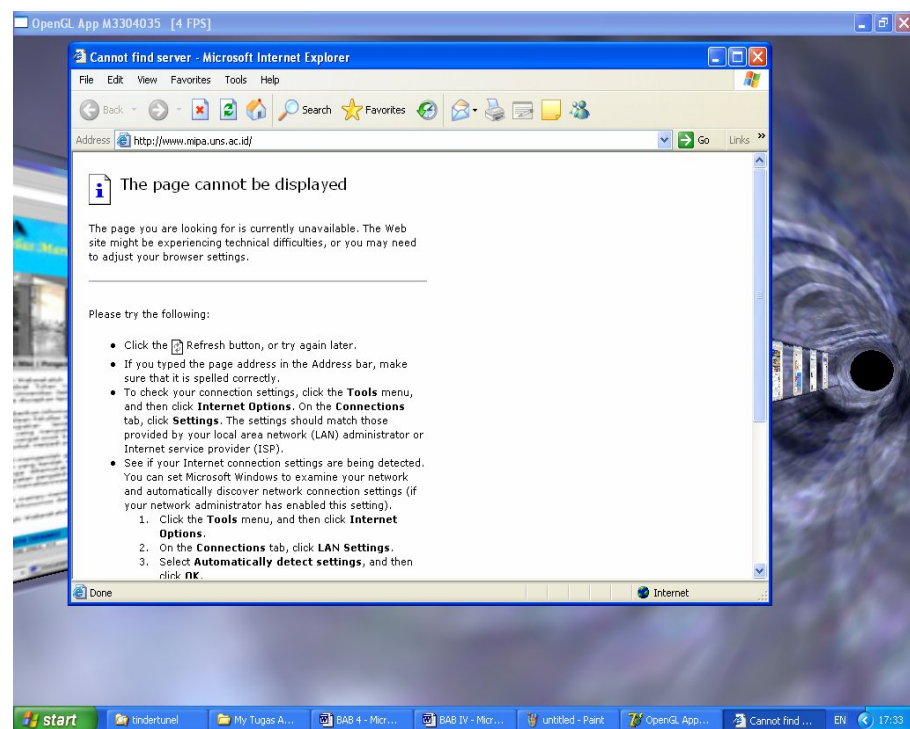
Sebagai catatan, bahwa web yang dituju sudah ditentukan dalam program aplikasi ini. Web-web tersebut adalah web umum yang biasa digunakan oleh *user*. Dan komputer yang digunakan untuk menjalankan program aplikasi ini harus terhubung dengan internet.

Berikut adalah alamat web tersebut :

1. <http://www.yahoo.com>
2. <http://www.friendster.com>
3. <http://www.fastncheap.com>

4. <http://www.ilmukomputer.com>
5. <http://www.plasa.com>
6. <http://www.klik-kanan.com>
7. <http://www.microsoft.com>
8. <http://www.mipa.uns.ac.id>
9. <http://www.google.com>

Berikut adalah tampilan saat program dieksekusi :



Gambar 4.5 Tampilan Program Setelah dieksekusi

Saat program dieksekusi maka tampilan browser akan muncul dan user dapat memilih web favorit yang tersedia. User dapat mengatur gerakan animasi (kecepatan) yaitu dengan cara menggunakan tombol navigasi yang ada pada keyboard. Setelah diklik maka akan langsung membuka alamat web tersebut.

4.5. Keunggulan dan kelemahan program

Keunggulan dan Kelemahan Program

1. Keunggulan

Keunggulan dari program aplikasi ini adalah :

- a. tidak membutuhkan memori yang besar untuk menjalankannya
- b. penggunaan program yang mudah dan bersifat *open source*, dimana seseorang mudah mendapatkannya.
- c. animatif dan interaktif

2. Kelemahan

Adapun kelemahan dari aplikasi ini adalah :

- a. URL tidak dapat ditentukan langsung oleh *browser*
- b. gerakan animasinya terbatas.
- c. desain program dan tampilan belum terlihat sebagai aplikasi grafis komputer

BAB V

PENUTUP

5.1 Kesimpulan

Setelah dilakukan analisis dan pembahasan dapat diambil kesimpulan sebagai berikut :

1. program aplikasi *browser* dapat digunakan oleh kalangan umum dan pelajar sebagai media penambah pengetahuan tentang grafika komputer.
2. simulasi aplikasi animasi web *browser* yang dibuat dengan delphi dan pustaka *library opengl* dapat digunakan untuk membuka alamat url/web favorit.

5.2 Saran

Berdasarkan permasalahan yang ada pada program aplikasi *browser* yang dirancang dan dikembangkan ini, dapat diberikan beberapa saran sebagai berikut:

1. fitur yang digunakan untuk menambahkan alamat web belum ada dan perlu pengembangan program lebih lanjut.
2. mode *full screen* belum ada sehingga tampilan animasi grafis komputer belum bisa dinikmati user secara maksimal.

DAFTAR PUSTAKA

- Horn, Jon. 2001. *Tinder Tunnel Box* . <http://www.sulaco.co.za>
<http://www.google.co.id>
- Nugoroho, Edi. 2005. *Teori dan Praktek Grafikia Komputer Menggunakan Delphi dan OpenGL*. Yogyakarta : Graha Ilmu.
- Purwanto, Eko. *Demo Membuat Animasi Teori dan Praktek*
[.http://www.ilmukomputer.com](http://www.ilmukomputer.com). Copyright 2005.. Jakarta
- Segal, Mark. *The OpenGL Graphics System : A Specification (Version 2.1 – July 30, 2006)*. 22-06-2007. opengl.org. 22-06-2007
- Sudargo, Paulus. 200. *Pemrograman Berorientasi Objek Menggunakan Delphi*. Yogyakarta : Andi
- Utami, Dina, ST. 2007. *Animasi Dalam Pembelajaran*. FKIP Jurusan KTP Prodi TP, NIP 132318129 UNY : Yogyakarta. 22-06-2007
- Zebua, Fajar Yusran. 2005. *Dasar Pembuatan Animasi Sederhana dengan Java*.
<http://www.ilmukomputer.com/Fajarzebua-animasijava.pdf>. 22-06-2007