

**PERANCANGAN APLIKASI PROYEKSI OBJEK 3 DIMENSI
TERHADAP BIDANG PROYEKSI 2 DIMENSI**

**SKRIPSI
KONSENTRASI REKAYASA KOMPUTER**

**Diajukan untuk memenuhi sebagian persyaratan
Memperoleh gelar Sarjana Teknik**



**Disusun oleh:
MUHAMMAD NUR LANTA ADENA
NIM. 0510630071 - 63**

**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
MALANG
2012**



PENGANTAR

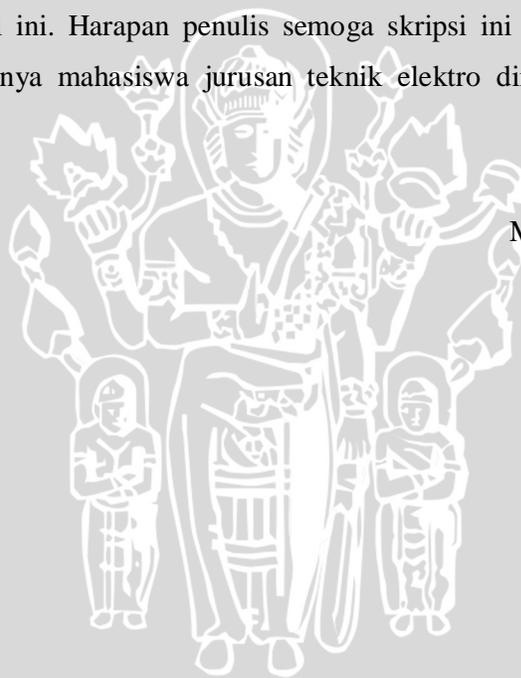
Puji syukur kehadiran Tuhan Yang Maha Esa, Sang Maha Pencipta yang telah memberikan limpahan rahmat dan hidayah sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul “Perancangan Aplikasi Proyeksi 3 Dimensi Terhadap Bidang 2 Dimensi” sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik di Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang.

Tidak banyak yang bisa penulis sampaikan kecuali ungkapan terima kasih kepada berbagai pihak yang telah dengan tulus ikhlas memberikan bimbingan, arahan, dan dukungan hingga penulisan tugas akhir ini dapat terselesaikan. Pada kesempatan kali ini, penulis mengucapkan banyak terima kasih kepada:

1. Bapak Dr. Ir. Sholeh Hadi Pramono, MS. selaku Ketua Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya.
2. Bapak M. Azis Muslim, S.T., M.T., Ph.D selaku Sekretaris Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya.
3. Bapak Moch. Rif'an. ST., MT. selaku Ketua Program Studi Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya.
4. Bapak Muhammad Aswin, Ir. selaku dosen pembimbing I yang telah banyak memberikan bimbingan, masukan dan arahan dalam penyusunan tugas akhir ini.
5. Bapak Waru Djuriatno, ST., MT. selaku dosen pembimbing II yang telah banyak memberikan bimbingan, masukan dan arahan dalam penyusunan tugas akhir ini.
6. Bapak Mahfudz Shiddiq, ST., MT. selaku dosen pendamping akademik.
7. Bapak dan Ibu Dosen serta karyawan jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya.
8. Ayahanda Abdul Gani, Ibunda Mahari Sriati, dan semua orang-orang terdekat yang telah memberikan dukungan serta doa dan semangat untuk terus maju hingga terselesaikan skripsi ini.

9. Fahmanda Idyanto dan Alfian Yudha P. yang telah memberikan saran, arahan serta dukungan hingga terselesaikan skripsi ini.
10. Teman-teman Streamline 2005, teman-teman RisTIE-HME kepengurusan 09/10 dan juga teman paket E angkatan 2005 yang selalu setia dengan kebersamaan dan kepedulian.
11. Serta semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat baik secara langsung maupun tidak langsung demi terselesainya tugas akhir ini.

Penulis menyadari sepenuhnya bahwa Tugas Akhir ini masih banyak kekurangan. Segala kritik dan saran yang membangun akan penulis terima demi kesempurnaan skripsi ini. Harapan penulis semoga skripsi ini bermanfaat bagi pembaca dan khususnya mahasiswa jurusan teknik elektro dimasa yang akan datang.



Malang, Juli 2012

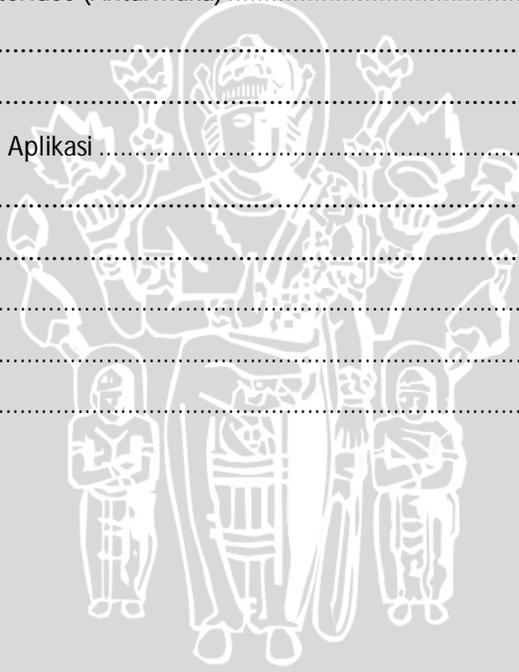
Penyusun

DAFTAR ISI

PENGANTAR.....	i
DAFTAR ISI.....	iii
DAFTAR GAMBAR.....	v
DAFTAR TABEL.....	vii
ABSTRAK.....	viii
BAB I.....	1
PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	1
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	2
1.5 Manfaat.....	3
1.6 Sistematika Penulisan.....	4
BAB II.....	5
TINJAUAN PUSTAKA.....	5
2.1 Grafika Komputer.....	5
2.2 Konsep Dasar 3 Dimensi (3D).....	6
2.2.2 Proyeksi.....	7
2.2.2.1 Proyeksi Paralel.....	9
2.3 Delphi.....	12
BAB III.....	13
METODOLOGI PENELITIAN.....	13
3.1 Studi Literatur.....	13
3.2 Perancangan.....	13
3.3 Implementasi.....	14
3.4 Pengujian.....	14
3.5 Pengambilan Kesimpulan dan Saran.....	14
BAB IV.....	15
PERANCANGAN DAN IMPLEMENTASI.....	15
4.1 Perancangan Sistem.....	15



4.1.1 Diagram Konteks.....	15
4.2 Cara Kerja Aplikasi.....	16
4.2.1 Input Data	16
4.2.2 Proyeksi Objek	17
4.3 Implementasi Sistem	21
4.3.1 Lingkungan Implementasi	21
4.3.2 Implementasi Proses Proyeksi Objek 3 Dimensi.....	21
4.3.2.1 Implementasi Proyeksi Objek	21
4.3.2.2 Implementasi Proses Penyimpanan Data.....	22
4.3.2.3 Implementasi Proses Deteksi Letak Permukaan Objek.....	22
4.3.2.4 Implementasi Proses Menampilkan Hasil Proyeksi Objek.....	23
4.4 Implementasi Interface (Antarmuka).....	24
BAB V.....	25
PENGUJIAN	25
5.1 Pengujian Pada Aplikasi	25
BAB VI	45
PENUTUP	45
6.1 Kesimpulan.....	45
6.2 Saran.....	45
DAFTAR PUSTAKA	46



DAFTAR GAMBAR

Gambar 2.1	(a) Sistem koordinat tangan kiri (b) Sistem kordinat tangan kanan	7
Gambar 2.2	Proyeksi titik pada koordinat 2D.....	7
Gambar 2.3	Proyeksi paralel.....	9
Gambar 2.4	(a) Proyeksi <i>orthographic</i> (b) proyeksi <i>oblique</i>	10
Gambar 2.5	Proyeksi <i>oblique</i> posisi koordinat (x,y,z) ke posisi koordinat (xp,yp) pada bidang proyeksi	10
Gambar 4.1	Diagram konteks aplikasi proyeksi	15
Gambar 4.2	Diagram alir proyeksi titik (xp,yp).....	18
Gambar 4.3	Diagram alir penentuan zmax	19
Gambar 4.4	Diagram alir pengurutan zmax tiap permukaan.....	20
Gambar 4.5	Antarmuka aplikasi proyeksi	24
Gambar 5.1	Objek 3 dimensi yang diujikan	25
Gambar 5.2	Tampilan aplikasi saat memproses permukaan 1 ($\alpha=45$)	29
Gambar 5.3	Tampilan aplikasi saat memproses permukaan 2 ($\alpha=45$)	30
Gambar 5.4	Tampilan aplikasi saat memproses permukaan 3 ($\alpha=45$)	30
Gambar 5.5	Tampilan aplikasi saat memproses permukaan 4 ($\alpha=45$)	31
Gambar 5.6	Tampilan aplikasi saat memproses permukaan 5 ($\alpha=45$)	31
Gambar 5.7	Tampilan aplikasi saat memproses permukaan 6 ($\alpha=45$)	32
Gambar 5.8	Tampilan aplikasi saat memproyeksikan objek 3 dimensi ($\alpha=45$)	32
Gambar 5.9	Tampilan aplikasi saat memproses permukaan 1 ($\alpha=60$)	33
Gambar 5.10	Tampilan aplikasi saat memproses permukaan 2 ($\alpha=60$)	33
Gambar 5.11	Tampilan aplikasi saat memproses permukaan 3 ($\alpha=60$)	34
Gambar 5.12	Tampilan aplikasi saat memproses permukaan 4 ($\alpha=60$)	34
Gambar 5.13	Tampilan aplikasi saat memproses permukaan 5 ($\alpha=60$)	35
Gambar 5.14	Tampilan aplikasi saat memproses permukaan 6 ($\alpha=60$)	35
Gambar 5.15	Tampilan aplikasi saat memproyeksikan objek 3 dimensi ($\alpha=60$).....	36
Gambar 5.16	Tampilan aplikasi saat memproses permukaan 1 ($\alpha=90$)	36
Gambar 5.17	Tampilan aplikasi saat memproses permukaan 2 ($\alpha=90$)	37
Gambar 5.18	Tampilan aplikasi saat memproses permukaan 3 ($\alpha=90$)	37
Gambar 5.19	Tampilan aplikasi saat memproses permukaan 4 ($\alpha=90$)	38
Gambar 5.20	Tampilan aplikasi saat memproses permukaan 5 ($\alpha=90$)	38
Gambar 5.21	Tampilan aplikasi saat memproses permukaan 6 ($\alpha=90$)	39
Gambar 5.22	Tampilan aplikasi saat memproyeksikan objek 3 dimensi ($\alpha=90$).....	39
Gambar 5.23	Gambar 5.23 Objek 3 dimensi yang diujikan (limas segiempat)	40

- Gambar 5.24 Tampilan aplikasi saat memproyeksikan objek 3 dimensi (limas) ($\alpha=45$) ..43
- Gambar 5.25 Tampilan aplikasi saat memproyeksikan objek 3 dimensi (limas) ($\alpha=60$) ..43
- Gambar 5.26 Tampilan aplikasi saat memproyeksikan objek 3 dimensi (limas) ($\alpha=90$) ..44



DAFTAR TABEL

Tabel 5.1	Informasi titik objek 3 dimensi permukaan 1.....	26
Tabel 5.2	Informasi titik objek 3 dimensi permukaan 2.....	26
Tabel 5.3	Informasi titik objek 3 dimensi permukaan 3.....	27
Tabel 5.4	Informasi titik objek 3 dimensi permukaan 4.....	27
Tabel 5.5	Informasi titik objek 3 dimensi permukaan 5.....	28
Tabel 5.6	Informasi titik objek 3 dimensi permukaan 6.....	28
Tabel 5.7	Informasi warna tiap permukaan	28
Tabel 5.8	Informasi titik objek 3 dimensi (limas) permukaan 1	40
Tabel 5.9	Informasi titik objek 3 dimensi (limas) permukaan 2	41
Tabel 5.10	Informasi titik objek 3 dimensi (limas) permukaan 3	41
Tabel 5.11	Informasi titik objek 3 dimensi (limas) permukaan 4	41
Tabel 5.12	Informasi titik objek 3 dimensi (limas) permukaan 5	42
Tabel 5.13	Informasi warna tiap permukaan limas.....	42



ABSTRAK

Proyeksi adalah cara untuk menampilkan atau menyajikan suatu objek 3D ke dalam suatu bidang 2D. Beberapa contoh bidang 2D adalah kertas dan monitor. Terdapat dua jenis metode proyeksi yang sering digunakan, yaitu Proyeksi Paralel (paralel projection) dan Proyeksi Perspektif (perspective projection). Skripsi ini membahas mengenai proyeksi objek yang terletak pada ruang 3 dimensi terhadap bidang proyeksi yaitu bidang 2 dimensi dan proyeksi yang digunakan adalah proyeksi paralel yaitu paralel *oblique*. Sistem ini bekerja dengan cara memproyeksikan titik-titik yang telah ditentukan pada koordinat 3 dimensi pada bidang proyeksi 2 dimensi yang berupa screen atau monitor. Berdasarkan hasil pengujian dan analisis sistem, perangkat lunak ini berhasil menjalankan semua fungsinya dengan benar pada kondisi-kondisi dimana permukaan objek yang digunakan adalah bidang datar. Proyeksi paralel *oblique* menghasilkan proyeksi objek 3 dimensi yang dapat dilihat ketiga bagiannya yaitu bagian atas, depan, dan samping.

kata kunci – grafika komputer, proyeksi 3 dimensi, paralel *oblique*.

BAB I PENDAHULUAN

1.1 Latar Belakang

Grafika komputer merupakan salah satu bidang dari ilmu komputer yang perkembangannya sangat cepat. Penggunaan grafika komputer sangat terasa manfaatnya di hampir seluruh kegiatan, terutama yang berhubungan dengan komputer. Menampilkan objek gambar secara 3D dapat mempermudah visualisasi dan penyampaian makna gambar dibandingkan dengan visualisasi secara dua dimensi (2D) karena gambar 3D lebih memudahkan dalam hal proses penyampaian informasi.

Tujuan dari grafik 3D adalah untuk merepresentasikan alam tiga dimensi ke alam dua dimensi. Representasi dilakukan dalam dua dimensi karena media untuk menampilkan grafik tersebut, yaitu layar komputer adalah dua dimensi. Merepresentasikan objek dalam tiga dimensi dapat dilakukan dengan menggunakan sistem koordinat yang menyediakan tiga sumbu koordinat. Tiga sumbu ini biasanya dinamakan X,Y,Z. Sedangkan sumbu koordinat pada bidang proyeksi adalah X dan Y.

Proyeksi adalah cara untuk menampilkan atau menyajikan suatu objek 3D ke dalam suatu bidang 2D. Beberapa contoh bidang 2D adalah kertas dan monitor. Terdapat dua jenis metode proyeksi yang sering digunakan, yaitu Proyeksi Paralel (*parallel projection*) dan Proyeksi Perspektif (*perspective projection*).

Skripsi membahas mengenai proyeksi objek yang terletak pada ruang 3 dimensi terhadap bidang proyeksi yaitu bidang 2 dimensi dan proyeksi yang digunakan adalah proyeksi *oblique*.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan secara jelas diatas, maka rumusan masalah ditekankan pada:

1. Memproyeksikan objek 3 dimensi yang direpresentasikan dengan titik x , y , dan z terhadap bidang proyeksi 2 dimensi yaitu layar komputer.
2. Pengaruh sudut α pada hasil proyeksi objek 3 dimensi

1.3 Batasan Masalah

Batasan perumusan masalah dalam penulisan studi kasus pada tugas akhir ini, meliputi:

1. Bidang proyeksi 2 dimensi sejajar dengan sumbu z pada koordinat 3 dimensi.
2. Objek 3 dimensi yang digunakan adalah bidang geometri dengan beberapa titik yang dihubungkan dengan garis lurus dengan permukaan berupa bidang datar dan menggunakan warna *monochrome*.
3. Proyeksi yang digunakan adalah proyeksi paralel *oblique*
4. Aplikasi berjalan di desktop.

1.4 Tujuan

Tujuan dari tugas akhir (skripsi) ini adalah:

1. Menerapkan proses proyeksi pada bangun geometri yang dibentuk dari permukaan bidang datar dengan beberapa titik.
2. Membangun sebuah aplikasi yang dapat memproyeksikan objek 3 dimensi.

1.5 Manfaat

Diharapkan manfaat yang dapat diperoleh melalui pengerjaan skripsi ini adalah:

a) Bagi Penyusun

1. Mampu mengembangkan sistem proyeksi objek 3 dimensi terhadap bidang proyeksi 2 dimensi.
2. Memperoleh pemahaman mengenai kelebihan serta kekurangan perangkat lunak yang telah dibuat.
3. Memahami bahwa sistem yang telah dibuat adalah hasil pengembangan daya pikir manusia, sebagai sumber daya terpenting dalam pembangunan sistem.
4. Menambah wawasan ilmu pengetahuan yang telah dipelajari sebelumnya, dan serta sebagai pelatihan berpikir kritis dalam menyelesaikan suatu masalah yang dihadapi.
5. Menerapkan ilmu yang telah didapat dalam perkuliahan.
6. Memberikan sumbangan bagi perkembangan ilmu pengetahuan khususnya dalam bidang *Information Technology (IT)*.

b) Bagi Pengguna dan Peneliti Lain

1. Dapat dikembangkan dan diintegrasikan dengan sistem yang ada untuk keperluan mereka, misalnya untuk memproyeksikan sebuah objek 3 dimensi yang diinginkan.

1.6 Sistematika Penulisan

Sistematika penulisan laporan skripsi ini adalah sebagai berikut :

BAB I Pendahuluan

Dalam bab ini akan dijelaskan latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat, metode penelitian dan sistematika penulisan laporan tugas akhir.

BAB II Tinjauan Pustaka

Dalam bab ini akan dibahas dan dijelaskan mengenai dasar teoritis yang menjadi landasan dan mendukung pelaksanaan penulisan tugas akhir.

BAB III Metodologi Penelitian

Dalam bab ini akan membahas tentang metode yang dipakai penulis untuk menyelesaikan laporan tugas akhir.

BAB IV Perancangan dan Implementasi

Dalam bab ini menjelaskan langkah langkah perancangan dan implementasi proyeksi 3 dimensi terhadap bidang proyeksi 2 dimensi.

BAB V Pengujian

Dalam bab ini akan disampaikan hasil pengujian dari aplikasi yang telah dibuat.

BAB VI Penutup

Memuat kesimpulan yang diperoleh dari analisis algoritma dan pengembangan aplikasi, serta saran-saran untuk pengembangan lebih lanjut.

BAB II TINJAUAN PUSTAKA

2.1 Grafika Komputer

Grafika komputer merupakan salah satu bidang dari ilmu komputer yang perkembangannya sangat cepat. Penggunaan grafika komputer sangat terasa manfaatnya di hampir seluruh kegiatan, terutama yang berhubungan dengan komputer. Pada kenyataannya, sebagian besar kegiatan manusia memanfaatkan grafika komputer. Industri film, televisi, desain grafis dan arsitektur adalah beberapa contoh kegiatan yang banyak sekali memanfaatkan grafika komputer. Bidang ilmu murni seperti Fisika, Matematika, Kimia dan Biologi pun merasakan manfaat dari grafika komputer. Bidang-bidang tersebut memanfaatkan grafika komputer untuk visualisasi model-model objek yang secara kasat mata mustahil terlihat seperti; atom, sel dan bakteri. Bahkan perkembangan bidang-bidang tersebut menjadi semakin cepat, karena para ilmuwan semakin berani melakukan eksperimen tanpa takut melakukan kesalahan yang mengakibatkan kerugian besar, sehingga mereka dapat menghasilkan penemuan-penemuan baru.

Berkat bantuan grafika komputer, Benoit Mandelbort, seorang ilmuwan matematika berhasil memodelkan dan memvisualisasikan fraktal, suatu bentuk geometri yang sangat rumit. Padahal selama berpuluh-puluh tahun fraktal hanya dikenal sebagai bentuk teori matematika saja dan masih abstrak. Merupakan hal yang mustahil menghasilkan suatu citra fraktal tanpa bantuan komputer, karena pada bagian-bagiannya memiliki bentuk yang sangat kecil dan tidak dapat terlihat oleh mata manusia yang paling normal sekalipun. Selain itu, membuat citra fraktal adalah pekerjaan yang sangat membosankan dan melelahkan. Dengan bantuan grafika komputer, hal tersebut menjadi mungkin dan bukan hal yang mustahil lagi.

Industri film dan *game* adalah yang paling merasakan manfaat grafika komputer. Saat ini, film yang digemari bukan lagi film-film kartun, tetapi film animasi yang menggunakan teknologi tiga dimensi (3D). Dengan grafik 3D, dapat dihasilkan suatu objek yang menyerupai bentuk aslinya. Hal ini sangat menguntungkan, karena pembuat film dapat mengurangi biaya produksi yang

biasanya digunakan untuk menyewa aktor dalam film tersebut, karena aktor-aktor tersebut digantikan perannya oleh objek-objek 3D buatan komputer. Hal yang sama terjadi pada industri game. Jika dulu game-game yang digemari masih dalam bentuk dua dimensi (2D), dengan teknologi yang ada sekarang dapat dihasilkan suatu game yang lebih realistis karena sudah dalam bentuk 3D.

Di Indonesia sendiri bidang-bidang yang berhubungan dengan grafika komputer sudah menjadi alternatif pilihan pekerjaan yang banyak diminati, seperti; desain grafis, pengolahan citra dan digital fotografi. Selain itu, pada bidang ini, bangsa Indonesia tidak terlalu tertinggal jika dibandingkan dengan bidang ilmu komputer yang lain seperti jaringan komputer dan pemrograman. Bahkan banyak ditemui orang-orang yang sudah sangat ahli pada bidang-bidang yang digelutinya, meskipun belum ditemui orang yang mampu menghasilkan suatu game yang spektakuler.

2.2 Konsep Dasar 3 Dimensi (3D)

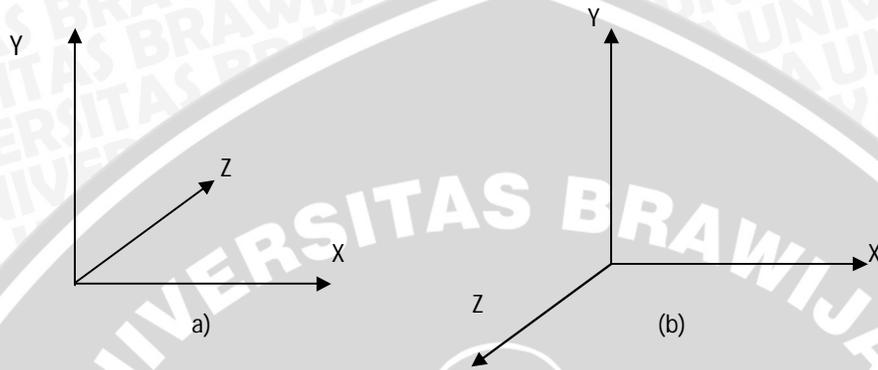
2.2.1 Sistem Koordinat

Menampilkan objek gambar secara 3D dapat mempermudah visualisasi dan penyampaian makna gambar dibandingkan dengan visualisasi secara dua dimensi (2D). Dari dahulu gambar 3D lebih dekat dengan orang awam, karena kemudahan proses penyampaian informasi.

Tujuan dari grafik 3D adalah untuk merepresentasikan alam tiga dimensi ke alam dua dimensi. Representasi dilakukan dalam dua dimensi karena media untuk menampilkan grafik tersebut, yaitu layar komputer adalah dua dimensi. Merepresentasikan objek dalam tiga dimensi dapat dilakukan dengan menggunakan sistem koordinat yang menyediakan tiga sumbu koordinat. Tiga sumbu ini biasanya dinamakan X,Y,Z.

Terdapat dua macam sistem koordinat 3D yaitu sistem tangan kiri (*Left-hand*) dan sistem tangan kanan (*Right-hand*). Perbedaan dari kedua sistem ini terletak pada arah sumbu z. Pada sistem tangan kiri koordinat yang lebih jauh mempunyai nilai z yang lebih besar, sedangkan koordinat yang lebih dekat mempunyai nilai z yang lebih kecil. Pada layar komputer, arah sumbu-z terlihat

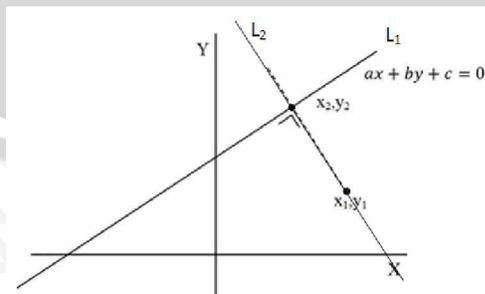
seolah-olah menjauhi pengguna komputer. Pada sistem tangan kanan, arah sumbu-z berkebalikan dengan tangan kiri, sehingga pada layar komputer terlihat seolah-olah mendekati pengguna komputer. Pada skripsi ini digunakan sistem koordinat tangan kiri.



Gambar 2.1 (a) Sistem koordinat tangan kiri (b) Sistem kordinat tangan kanan

2.2.2 Proyeksi

Proyeksi adalah cara untuk menampilkan atau menyajikan suatu objek 3D ke dalam suatu bidang 2D. Proyeksi dari sebuah obyek 3 dimensi didefinisikan dengan garis proyeksi yang bersumber dari sebuah titik proyeksi, kemudian melewati setiap titik pada obyek dan memotong sebuah bidang proyeksi guna membentuk proyeksi tersebut. Beberapa contoh bidang 2D adalah kertas dan monitor. Gambar 2.2 merupakan posisi titik yang berada di koordinat 2 dimensi dari koordinat 3 dimensi, dimana sumbu-z berhadapan dengan kita sehingga searah dengan titik pusat koordinat. Bidang proyeksi sejajar dengan sumbu-z sehingga membentuk sebuah garis.



Gambar 2.2 Proyeksi titik pada koordinat 2D

Dengan menarik garis lurus melewati titik (x_1, y_1) , maka akan didapatkan persamaan garis L_2 yang tegak lurus dengan garis L_1 dan berpotongan di titik (x_2, y_2) yang merupakan titik proyeksi.

Persamaan garis lurus L_1 : $ax + by + c = 0$(1)

Dimana

$$y = \frac{-ax - c}{b} \dots\dots\dots(2)$$

dan

$$x = \frac{-by - c}{a} \dots\dots\dots(3)$$

Persamaan garis L_2 yang melewati titik (x_1, y_1) dan tegak lurus dengan garis L_1 adalah:

$$y - y_1 = \frac{b}{a}(x - x_1)$$

Atau

$$bx - ay - (bx_1 - ay_1) = 0 \dots\dots\dots(4)$$

Kedua garis tersebut berpotongan di titik (x_2, y_2) dan dengan memasukkannya ke dalam kedua persamaan garis di atas, didapatkan (x_2, y_2) sebagai fungsi (x_1, y_1) :

$$bx_2 - ay_2 - (bx_1 - ay_1) = 0 \dots\dots\dots(5)$$

Karena titik (x_2, y_2) menurut persamaan (2) maka,

$$y_2 = \frac{-ax_2 - c}{b} \dots\dots\dots(6)$$

Lalu dimasukan ke dalam persamaan (5) menjadi,

$$bx_2 - a\left(\frac{-ax_2 - c}{b}\right) - (bx_1 - ay_1) = 0$$

$$bx_2 + \frac{a^2x_2 + ac}{b} - (bx_1 - ay_1) = 0$$

$$bx_2 + \frac{a^2x_2 + ac}{b} = (bx_1 - ay_1)$$

$$\frac{b^2x_2 + a^2x_2 + ac}{b} = (bx_1 - ay_1)$$

$$b^2x_2 + a^2x_2 + ac = (b^2x_1 - aby_1)$$



$$(a^2 + b^2)x_2 = b^2x_1 - aby_1 - ac$$

$$x_2 = \frac{b^2x_1 - aby_1 - ac}{(a^2 + b^2)} \dots\dots\dots(7)$$

Dengan memasukan persamaan (7) ke dalam persamaan (6), maka diperoleh,

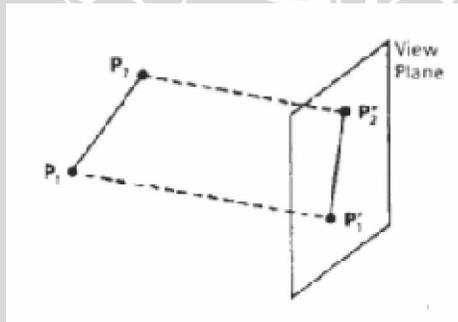
$$y_2 = \frac{-a}{b} \left(\frac{b^2x_1 - aby_1 - ac}{(a^2 + b^2)} \right) - \frac{c}{b} \dots\dots\dots(8)$$

Nilai titik (x_2, y_2) dapat diperoleh dari persamaan (7) dan persamaan (8).

Terdapat dua jenis metode proyeksi yang sering digunakan, yaitu proyeksi paralel (*parallel projection*) dan Proyeksi Perspektif (*perspective projection*). Skripsi menggunakan proyeksi paralel.

2.2.2.1 Proyeksi Paralel

Pada proyeksi paralel, suatu objek diproyeksikan terhadap bidang sepanjang garis secara paralel. Dengan menggunakan metode ini ukuran dan bentuk objek tidak berubah. Gambar 2.3 menunjukkan gambar titik yang diproyeksikan paralel terhadap suatu bidang proyeksi.



Gambar 2.3 Proyeksi paralel

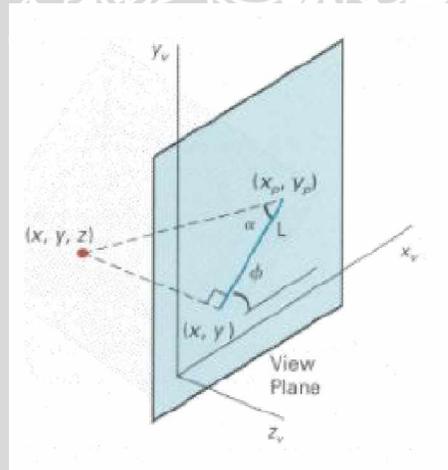
(“Computer Graphics Second Edition”. Prentice-Hall International Inc. 1994)



Gambar 2.4 Proyeksi *orthographic* (a) dan proyeksi *oblique* (b)

(“Computer Graphics Second Edition”. Prentice-Hall International Inc. 1994)

Jenis proyeksi paralel yang digunakan dalam tugas akhir ini adalah proyeksi *oblique*. Proyeksi paralel *oblique* diperoleh dengan memproyeksikan titik sepanjang garis paralel yang tidak tegak lurus dengan bidang proyeksi. Proyeksi *oblique* ditentukan oleh dua buah sudut, α dan Φ , seperti yang ditunjukkan pada gambar 2.5 berikut.



Gambar 2.5 Proyeksi *oblique* posisi koordinat (x,y,z) ke posisi koordinat (x_p,y_p) pada bidang proyeksi

(http://comp575.web.unc.edu/files/2010/10/12_3DViewingPipeline.pdf. 2012)

Sudut α merupakan sudut yang dibentuk antara sinar proyeksi objek dengan garis L atau sudut pandangan pergeseran ke kiri atau ke kanan (arah

horizontal) memandang terhadap objek 3 dimensi. Sudut Φ merupakan sudut yang dibentuk antara garis L dengan garis horizontal bidang proyeksi atau sudut pandangan naik atau turun (arah vertikal) dari memandang terhadap objek 3 dimensi. Titik (x,y,z) diproyeksikan pada posisi (x_p,y_p) pada bidang proyeksi. koordinat proyeksi *orthographic* pada bidang proyeksi adalah (x,y) . Garis proyeksi *oblique* dari (x,y,z) ke (x_p,y_p) membuat sudut α dengan garis pada bidang proyeksi yang menghubungkan (x_p,y_p) dan (x,y) . Garis ini, sepanjang L, membentuk sudut Φ terhadap arah horizontal pada bidang proyeksi. Kita dapat menampilkan koordinat proyeksi dalam persamaan x, y, L, dan Φ sebagai berikut:

$$x_p = x + L \cos \phi$$

$$y_p = y + L \sin \phi$$

Panjang L bergantung pada sudut α dan koordinat z dari titik yang diproyeksikan, sehingga:

$$\tan \alpha = \frac{z}{L}$$

$$L = \frac{z}{\tan \alpha}$$

Sehingga,

$$L = zL_1$$

Dimana L_1 adalah invers dari $\tan \alpha$, yang mana juga merupakan nilai L ketika $z=1$. Kita bisa menuliskan persamaan proyeksi *oblique* sebagai berikut:

$$x_p = x + z(L_1 \cos \phi)$$

$$y_p = y + z(L_1 \sin \phi)$$

Apabila $\alpha = 90^\circ$ maka $L_1 = 0$ sehingga dari rumus di atas kita memperoleh proyeksi *orthographic*, tetapi apabila L_1 tidak sama dengan 0 maka kita akan memperoleh proyeksi *oblique*. Proyeksi *oblique* dengan $\alpha = 45^\circ$ disebut sebagai proyeksi cavalier, apabila $\alpha = 63,4^\circ$ maka kita akan memperoleh proyeksi cabinet.

2.3 Delphi

Delphi adalah sebuah IDE *Compiler* untuk bahasa pemrograman Pascal dan lingkungan pengembangan perangkat lunak. Produk ini dikembangkan oleh CodeGear sebagai divisi pengembangan perangkat lunak milik Embarcadero, divisi tersebut sebelumnya adalah milik Borland. Bahasa Delphi, atau dikenal pula sebagai object pascal (pascal dengan ekstensi pemrograman berorientasi objek (PBO/OOP)) pada mulanya ditujukan hanya untuk Microsoft Windows, namun saat ini telah mampu digunakan untuk mengembangkan aplikasi untuk Linux dan Microsoft .NET framework. Dengan menggunakan *Free Pascal* yang merupakan proyek opensource, bahasa ini dapat pula digunakan untuk membuat program yang berjalan di sistem operasi Mac OS X dan Windows CE.

Umumnya Delphi lebih banyak digunakan untuk pengembangan aplikasi desktop dan enterprise berbasis database, tapi sebagai perangkat pengembangan yang bersifat general-purpose ia juga mampu dan digunakan dalam berbagai jenis proyek pengembangan software. Ia juga yang dikenal sebagai salah satu yang membawa istilah *RAD tool*, kepanjangan dari *Rapid Application Development*, saat dirilis tahun 1995 untuk windows 16-bit. Delphi 2, dirilis setahun kemudian, mendukung lingkungan windows 32-bit, dan versi C++, C++Builder, dirilis beberapa tahun kemudian. Pada tahun 2001 sebuah versi linux yang dikenal sebagai Kylix tersedia. Dengan satu rilis baru setiap tahunnya, pada tahun 2002 dukungan untuk Linux (melalui Kylix dan CLX component library) ditambahkan dan tahun 2003 .NET mulai didukung dengan munculnya Delphi.Net (Delphi 8). Pengerjaan skripsi ini menggunakan Delphi 7.

BAB III

METODOLOGI PENELITIAN

Pada tahap ini dijelaskan mengenai langkah-langkah yang akan dilakukan untuk mengimplementasikan perangkat lunak yang akan dibuat. Adapun langkah-langkah yang akan dilakukan adalah sebagai berikut :

3.1 Studi Literatur

Berupa kajian pustaka yang mendukung pembuayan aplikasi yaitu konsep proyeksi 3 dimensi dan jenis proyeksi yang digunakan.

3.2 Perancangan

Tugas akhir ini bertujuan untuk menghasilkan proyeksi 2 dimensi dari objek 3 dimensi. Sistem ini akan memproyeksikan objek 3 dimensi yang dibentuk dari beberapa permukaan yang dibentuk dari beberapa titik yang ditentukan oleh pengguna. Sistem akan menampilkan output berupa objek 2 dimensi disertai dengan warna objeknya berdasarkan data masukan dari pengguna.

Cara kerja dari aplikasi proyeksi ini adalah sebagai berikut :

1. Pengguna memilih jumlah bidang permukaan dari objek 3 dimensi yang akan diproyeksikan. Jumlah bidang dibatasi sebanyak 10 buah. Selanjutnya pengguna memilih jumlah titik yang akan merepresentasikan titik sudut objek 3 dimensi yang akan diproyeksikan. Jumlah titik dibatasi sebanyak 8 buah titik. Ketika memilih jumlah titik, otomatis form input akan berubah sesuai dengan jumlah titik yang dipilih. Kemudian pengguna menentukan warna dari objek yang akan ditampilkan tersebut.
2. Pengguna memasukan nilai titik pada form input. Satu titik diwakilkan dengan 3 buah input yaitu masing-masing x , y , dan z yang merepresentasikan posisi titik tersebut di dalam bidang 3 dimensi.
3. Ketika semua titik telah diisi nilainya, kemudian pengguna menekan tombol proses untuk memproses koordinat hasil proyeksi yang terletak di bagian bawah. Selanjutnya, pengguna menekan tombol tambah untuk

menambah bidang dan menyimpan nilai dari titik tersebut. Kemudian dilanjutkan dengan proses yang sama untuk bidang berikutnya.

4. Ketika semua bidang telah diproses proyeksi, selanjutnya pengguna dapat menekan tombol proyeksi untuk menampilkan objek tersebut pada bidang proyeksi 2 dimensi.

3.3 Implementasi

Proses selanjutnya adalah implementasi yakni, proses transformasi hasil perancangan perangkat lunak yang telah dibuat ke dalam kode (*coding*) sesuai dengan sintaks dari bahasa pemrograman yang digunakan yaitu menerapkan bahasa pemrograman Pascal dengan menggunakan Borland Delphi 7.

3.4 Pengujian

Pengujian dilakukan dengan memasukan jumlah titik dengan nilai-nilai koordinat yang telah ditentukan sebelumnya. Objek 3 dimensi tersebut telah ditentukan terlebih dahulu kemudian mencocokkan hasil bentuk objeknya apakah sama dengan hasil keluaran objek dengan menggunakan aplikasi proyeksi.

3.5 Pengambilan Kesimpulan dan Saran

Kesimpulan diperoleh dari data-data hasil pengujian aplikasi yang dilakukan, sehingga diperoleh saran juga guna perbaikan aplikasi pada penelitian selanjutnya.

BAB IV PERANCANGAN DAN IMPLEMENTASI

Di dalam bab perancangan ini akan dijelaskan mengenai perancangan perangkat lunak. Perancangan yang dilakukan meliputi perancangan sistem dan cara kerja sistem.

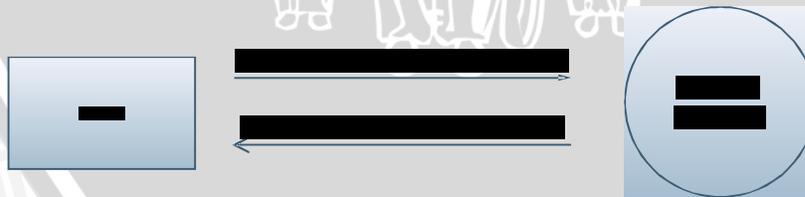
4.1 Perancangan Sistem

Perancangan sistem merupakan tahap awal dari perancangan perangkat lunak. Perancangan ini dilakukan untuk mengetahui aplikasi sistem yang akan dibuat secara umum. Perancangan sistem ini akan didahului dengan pendefinisian pelaku atau user yang akan menggunakan program ini dan juga alat bantu yang digunakan, yaitu:

1. User : Pengguna yang akan memasukan data-data yang diperlukan objek untuk diproyeksikan.
2. Program : Perangkat lunak yang digunakan untuk memproyeksikan objek 3 dimensi ke bidang 2 dimensi.

4.1.1 Diagram Konteks

Diagram konteks adalah sebuah diagram sederhana yang menggambarkan hubungan dengan entitas luar, masukan dan keluaran dari sistem. Diagram konteks aplikasi proyeksi 3 dimensi terhadap bidang proyeksi 2 dimensi ditunjukkan pada gambar 4.1



Gambar 4.1 Diagram konteks aplikasi proyeksi

Berikut adalah penjelasan bagian dalam diagram konteks tersebut:

1. Pengguna memasukan data-data dari objek 3 dimensi. Data tersebut adalah

posisi titik pada koordinat 3 dimensi, warna objek, jumlah permukaan dan nilai α .

2. Proses proyeksi menggunakan proyeksi paralel *oblique* yang akan memproses hasil proyeksi berdasarkan data-data nilai titik dan α yang telah dimasukkan oleh pengguna. Hasilnya merupakan proyeksi objek 3 dimensi.
3. Hasil proyeksi objek 3 dimensi akan ditampilkan ke layar.

4.2 Cara Kerja Aplikasi

Cara kerja dari aplikasi proyeksi adalah sebagai berikut:

1. Program mempunyai masukan data berupa nilai x , y , dan z dari tiap-tiap titik dan nilai α . Untuk nilai Φ merupakan nilai konstanta yang telah ditetapkan terlebih dahulu.
2. Data nilai tersebut akan diproses dengan menggunakan rumus proyeksi paralel *oblique* untuk mendapatkan nilai proyeksi pada sumbu x dan pada sumbu z .
3. Nilai proyeksi x dan y ini adalah nilai posisi titik tersebut pada bidang proyeksi 2 dimensi.
4. Bidang datar diberi warna sesuai dengan input dari pengguna.
5. Data ini kemudian disimpan sementara dan dilanjutkan dengan memasukan data untuk tiap jumlah permukaan yang telah ditentukan.
6. Selanjutnya tiap-tiap titik yang telah diketahui nilai x dan y proyeksinya tersebut akan dihubungkan dengan garis lurus sehingga membentuk sebuah bidang datar.
7. Setelah data-data setiap permukaan terkumpul, kemudian data-data tersebut akan diproses untuk ditampilkan hasil proyeksinya pada layar.

4.2.1 Input Data

Data-data yang dimasukan oleh pengguna adalah data berupa:

1. Jumlah permukaan objek 3 dimensi. Dimulai dari 1 sampai dibatasi 8 buah permukaan. Menentukan jumlah permukaan objek 3 dimensi ini

sesuai dengan jumlah permukaan objek 3 dimensi yang akan dibentuk oleh pengguna.

2. Jumlah titik dari permukaan objek 3 dimensi. Dimulai dari 3 titik dan dibatasi sampai 8 titik.
3. Warna permukaan objek 3 dimensi, untuk membedakan antara permukaan satu dengan yang lainnya.
4. Nilai x , y , dan z yang merepresentasikan posisi titik tiap titik sudut pada permukaan objek 3 dimensi.
5. Nilai α yang merupakan sudut antara pengamat dengan objek dan bergerak sejajar dengan bidang proyeksi.

4.2.2 Proyeksi Objek

Proyeksi yang digunakan pada aplikasi ini adalah proyeksi paralel *oblique*. Titik x , y , dan z pada ruang 3 dimensi diproyeksikan terhadap bidang proyeksi sehingga memiliki koordinat hasil proyeksi x_p dan y_p . Untuk mendapatkan nilai tersebut digunakan persamaan proyeksi *oblique* yang dapat dituliskan sebagai berikut :

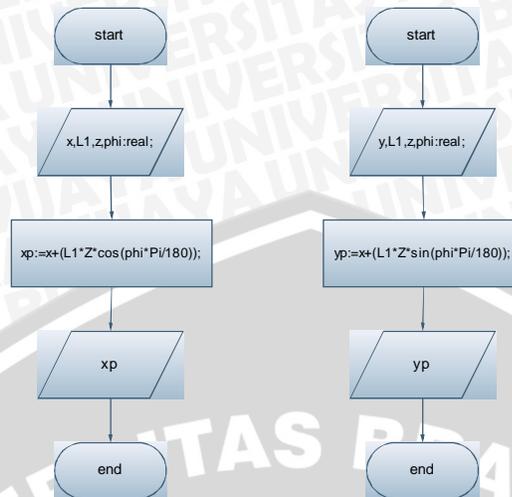
$$x_p = x + z(L_1 \cos \phi)$$

$$y_p = y + z(L_1 \sin \phi)$$

$$L_1 = \frac{1}{\tan \alpha}$$

Pilihan sudut Φ yang biasanya digunakan adalah 30° dan 45° yang menampilkan kombinasi tampilan proyeksi depan, samping, dan atas dari objek. Pengguna memasukan nilai sudut α yang merupakan sudut antara pemandang dengan bidang proyeksi. Dalam aplikasi ini digunakan $\Phi=30^\circ$ yang menjadi variabel konstanta.

Adapun diagram alir dari proyeksi objek adalah sebagai berikut:



Gambar 4.2 Diagram alir proyeksi titik (x_p, y_p)

4.2.3 Menyimpan Data Sementara

Data-data yang diperlukan untuk proyeksi tersebut disimpan sementara di dalam sebuah array. Ketika pengguna selesai mengisi data sebuah permukaan dan akan menambah data permukaan yang lainnya, pengguna akan menyimpan data permukaan sebelumnya sebelum melanjutkan mengisi data untuk permukaan selanjutnya. Setelah jumlah permukaan telah terpenuhi semua sesuai dengan kebutuhan, data-data ini akan diproyeksikan secara bersamaan dan membentuk sebuah permukaan yang akan membentuk sebuah objek 3 dimensi.

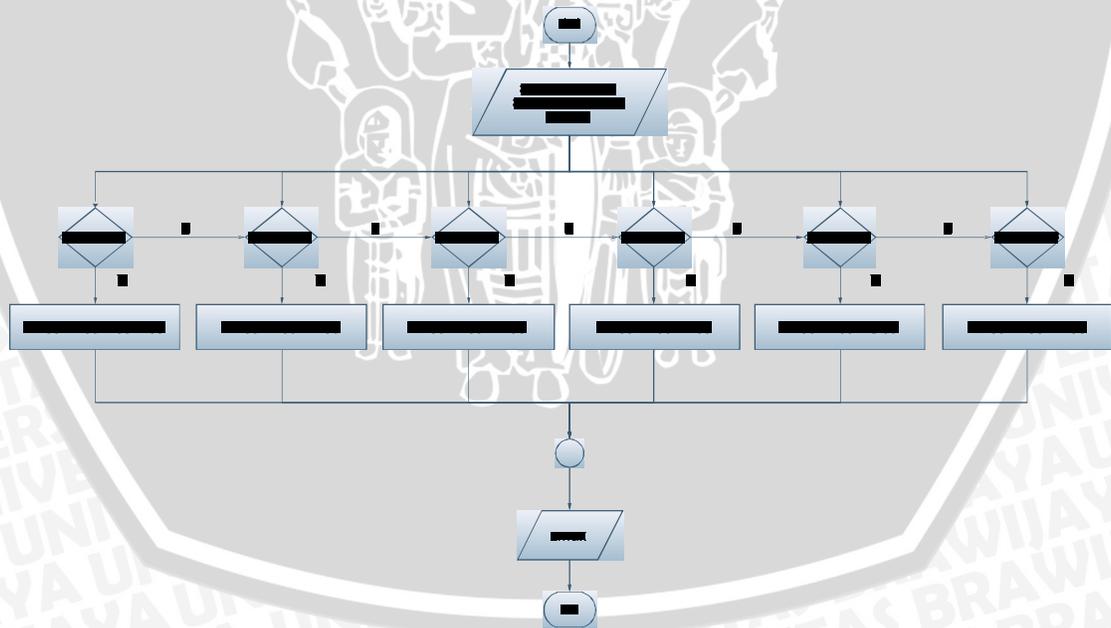
4.2.4 Deteksi Letak Permukaan

Objek 3 dimensi dibentuk dari beberapa permukaan. Permukaan objek yang akan ditampilkan berdasarkan dengan letak permukaan tersebut pada koordinat 3 dimensi. Permukaan yang terletak di depan atau paling dekat jaraknya dengan pemandang akan ditampilkan paling depan dan sebaliknya, permukaan yang letaknya paling jauh dengan pemandang akan diletakkan paling belakang. Untuk menghindari ketimpangan antar permukaan sehingga menghasilkan bentuk objek 3 dimensi yang tidak sesuai keinginan, maka letak permukaan harus ditentukan terlebih dahulu. Cara yang digunakan adalah dengan melihat nilai z

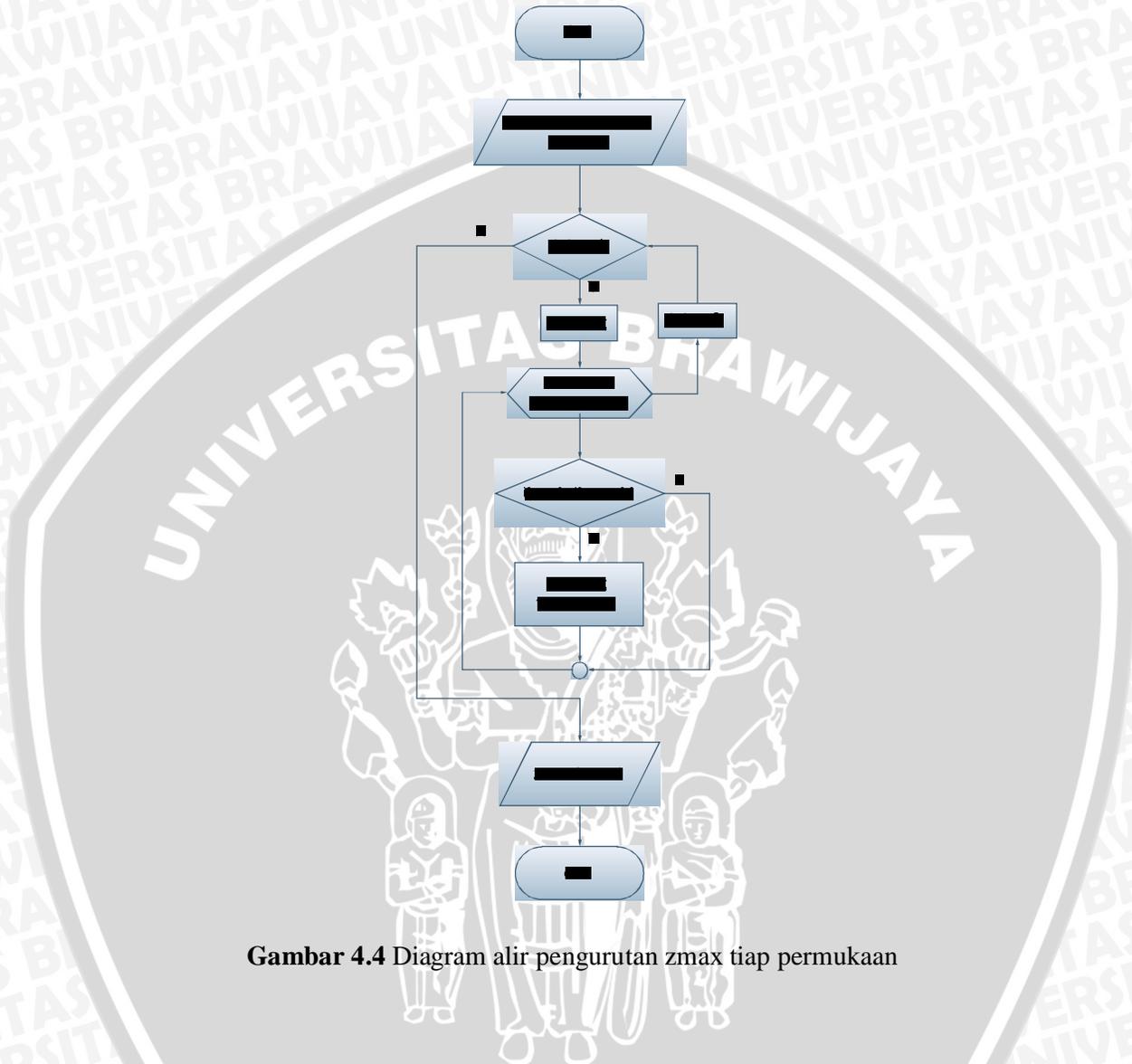
dari permukaan, dimana semakin besar nilai z , maka permukaan tersebut terletak lebih jauh dari pemandang .

Karena objek akan diproyeksikan pada sumbu x dan y , maka sumbu z merupakan sumbu yang menentukan kedalaman objek 3 dimensi tersebut dalam koordinat 3 dimensi. Setelah nilai titik x , y , dan z dari sebuah permukaan terkumpul, akan dilakukan proses penjumlahan nilai z sehingga didapatkan nilai jumlah z yang diasumsikan sebagai z terbesar (z_{max}) dari sebuah permukaan. Ketika nilai z_{max} telah didapatkan, nilai ini akan disimpan sementara untuk kemudian dibandingkan dengan nilai z_{max} permukaan lainnya. Nilai z_{max} permukaan yang paling besar akan meletakkan permukaan di posisi paling belakang dan nilai z_{max} yang paling kecil akan menempatkan posisi permukaan paling depan, sehingga bentuk objek 3 dimensi yang akan diproyeksikan akan sesuai dengan yang diharapkan.

Adapun diagram alir dari penentuan peletakan permukaan adalah sebagai berikut:



Gambar 4.3 Diagram alir penentuan z_{max}



Gambar 4.4 Diagram alir pengurutan zmax tiap permukaan

4.2.5 Menampilkan Hasil Proyeksi Objek

Untuk menampilkan hasil proyeksi objek yang telah dibentuk yaitu dengan menghubungkan titik sudut satu dengan yang lainnya pada permukaan dengan garis lurus. Kemudian tiap-tiap bidang permukaan objek tersebut diberi warna *monochrome* sesuai dengan data masukan dari pengguna.

4.3 Implementasi Sistem

Setelah tahap perancangan dibuat maka untuk selanjutnya adalah tahap implementasi. Pada tahap implementasi ini merupakan proses transformasi dari proses perancangan yang telah dibuat sebelumnya dimana hasil perancangan ditransformasikan ke dalam bentuk bahasa pemrograman (*coding*) yang sesuai dengan sintaks yang ada pada bahasa pemrograman Pascal.

4.3.1 Lingkungan Implementasi

Sistem dibuat dengan menggunakan aplikasi pemrograman Borland Delphi 7. Sistem diimplementasikan dengan menggunakan spesifikasi sebagai berikut:

a) Perangkat Keras :

1) Notebook

CPU : Intel(R) Pentium(R) Dual CPU T2390 (1.86GHz)

Memory : 1 GB RAM

VGA : VIA Chrome9 HC IGP

Harddisk : 160 GB

b) Perangkat Lunak :

1) Windows XP

2) Borland Delphi 7

4.3.2 Implementasi Proses Proyeksi Objek 3 Dimensi

4.3.2.1 Implementasi Proyeksi Objek

Untuk mendapatkan nilai titik hasil proyeksi yaitu x_p dan y_p digunakan dua buah fungsi yaitu *ProyeksiX* dan *ProyeksiY* yang akan memproses data titik-titik masukan x , y , dan z dari permukaan. Untuk implementasinya pada Delphi adalah sebagai berikut:

```
function Proyeksi X (x, L1, Zp, Z, phi : real) : real ;
```

```
var
```

```
xp: real ;
```

```

begin
xp: =x+(L1*(Zp-Z)*cos(phi *Pi /180));
result: =xp;
end;

function ProyeksiY (y, L1, Zp, Z, phi : real) : real;
var
yp: real;
begin
yp: =y+(L1*(Zp-Z)*sin(phi *Pi /180));
result: =yp;
end;

```

Nilai x, y, z , dan α merupakan masukan dari pengguna, sedangkan nilai Z_p dan Φ merupakan nilai konstanta.

4.3.2.2 Implementasi Proses Penyimpanan Data

Data titik x, y , dan z yang telah dimasukan oleh pengguna untuk satu permukaan akan disimpan sementara dengan menggunakan array. Masing-masing array menyimpan nilai x, y , dan z dengan indeks titik masing-masing. Kemudian titik-titik ini akan diproses untuk menghasilkan titik proyeksinya pada bidang proyeksi. Nilai hasil proyeksi titik-titik tersebut kemudian akan disimpan sementara untuk diproyeksikan secara bersamaan nantinya. Nama permukaan dideskripsikan dengan sebuah variabel yaitu N yang bertindak sebagai counter. nilai awal N adalah 0 dan akan bertambah setiap menambah data-data permukaan.

4.3.2.3 Implementasi Proses Deteksi Letak Permukaan Objek

Untuk mendeteksi letak permukaan objek, digunakan nilai hasil penjumlahan antara nilai z tiap titik untuk mencari z yang terbesar (z_{max}). Z terbesar (z_{max}) ini menjadi identifikasi letak untuk tiap permukaan pada koordinat 3 dimensi. Kemudian z terbesar (z_{max}) dari tiap permukaan akan

dibandingkan kembali untuk menentukan letak permukaan untuk ditampilkan paling depan atau paling belakang jaraknya dari pemandang.

4.3.2.4 Implementasi Proses Menampilkan Hasil Proyeksi Objek

Untuk menampilkan hasil proyeksi, digunakan sintaks polygon dalam Delphi. Sintaks ini membutuhkan nilai x dan y yang dalam hal ini adalah xp dan yp atau titik hasil proyeksi yang telah didapatkan untuk masing-masing permukaan. Sedangkan untuk menentukan warna permukaannya, digunakan sintaks `ImgUtama.Canvas.Brush.Color` dengan nilai warna berdasarkan masukan dari pengguna. Untuk menampilkan bentuk permukaannya digunakan sintaks `ImgUtama.Canvas.Polygon`. Untuk implementasinya pada Delphi adalah sebagai berikut:

```

for u:=1 to jumlahbidang do
begin
  case warna[u] of
    0: ImgUtama.Canvas.Brush.Color:=clRed;
    1: ImgUtama.Canvas.Brush.Color:=clYellow;
    2: ImgUtama.Canvas.Brush.Color:=clGreen;
    3: ImgUtama.Canvas.Brush.Color:=clBlue;
    4: ImgUtama.Canvas.Brush.Color:=clPurple;
    5: ImgUtama.Canvas.Brush.Color:=clGray;
  end;

  ImgUtama.Canvas.Polygon([Point(round(pxa[u]),ypus(round(pya[u]))),
  Point(round(pxb[u]),ypus(round(pyb[u]))), Point(round(pxc[u]),
  ypus(round(pyc[u])))]);

end;

ypus merupakan fungsi untuk memindahkan titik asal (titik 0) dari sumbu y. Pada
TImage, titik asal (0,0) berada pada sebelah kiri atas. Untuk mengubah posisinya
menjadi di kiri bawah, digunakan fungsi ypus tersebut.

```

4.4 Implementasi Interface (Antarmuka)

Antarmuka aplikasi dibagi menjadi dua bagian, bagian yang pertama adalah bagian input yang memproses data-data objek 3 dimensi, yang kedua adalah bagian output yaitu bidang proyeksi 2 dimensi yang akan menampilkan hasil proyeksi dari objek 3 dimensi dan nilai titik-titik hasil proyeksi. Pada bagian input, terdapat tiga buah dropdown yang berisikan masing-masing jumlah bidang permukaan dari objek 3 dimensi, warna permukaan, dan jumlah titik. Form-form untuk memasukan nilai x , y , dan z untuk masing-masing titik. Form untuk memasukan nilai α (alpha). Tombol-tombol yaitu tombol proyeksi untuk proses proyeksi, tombol tambah untuk menambah permukaan sekaligus menyimpan data permukaan sebelumnya, tombol proyeksi untuk memproyeksikan objek ke bidang proyeksi 2 dimensi, dan tombol tutup untuk menutup aplikasi. Ketika pengguna telah berada pada proses memasukan data permukaan bidang terakhir, ketika tombol proses ditekan maka tombol proses dan tombol tambah akan dinonaktifkan. Pengguna selanjutnya melanjutkan untuk menekan tombol proyeksi untuk memproyeksikan objek 3 dimensi.

Berikut adalah gambaran dari antarmuka aplikasi proyeksi objek 3 dimensi terhadap bidang proyeksi 2 dimensi.

The screenshot shows a software interface for 3D projection. The window title is "Aplikasi Proyeksi 3 Dimensi". The interface is divided into several sections:

- Koordinat Titik Asal (Original Point Coordinates):** Includes dropdown menus for "Jumlah Bidang" (set to 1), "Jumlah Titik" (set to 3), and "Warna" (set to "merah"). Below these are three rows of input fields for points A, B, and C, each with x, y, and z coordinate fields.
- Angle Inputs:** At the bottom left, there are input fields for "phi = 30" and "alpha = 45", along with "Proses" and "Tambah" buttons.
- Bidang Ke: 1 (Plane):** A central label indicating the current plane.
- Koordinat Hasil Proyeksi (Resulting Point Coordinates):** On the right, there are two columns of input fields labeled PA, PB, PC, PD and PE, PF, PG, PH.
- Action Buttons:** At the bottom right, there are "Proyeksi" and "Tutup" buttons.

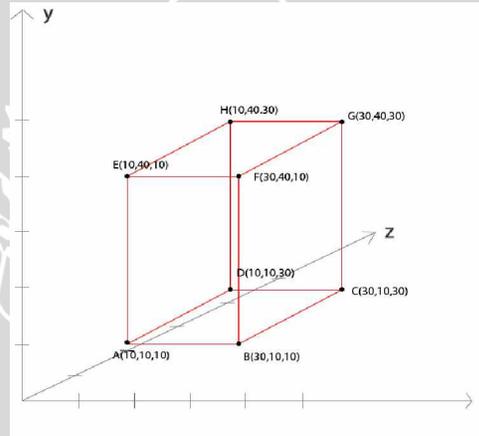
Gambar 4.5 Antarmuka aplikasi proyeksi

BAB V PENGUJIAN

Untuk mengetahui apakah sistem bekerja dengan baik dan sesuai dengan perancangan, maka diperlukan serangkaian pengujian. Pengujian yang dilakukan dalam bab ini adalah membandingkan keluaran proyeksi objek dengan objek 3 dimensi yang diinginkan dan telah ditentukan sebelumnya.

5.1 Pengujian Pada Aplikasi

Objek 3 dimensi yang digunakan dalam pengujian ini adalah balok dan limas segiempat. Untuk objek balok memiliki data-data seperti yang ditunjukkan pada gambar 5.1 berikut ini.



Gambar 5.1 Objek 3 dimensi yang diujikan (balok)

Selanjutnya data titik-titik koordinat 3 dimensi tersebut dimasukan ke dalam aplikasi berdasarkan permukaannya masing-masing. Untuk objek balok memiliki enam buah permukaan, sehingga masukannya adalah 4 titik untuk setiap permukaan. Data masukan nilai titik-titik permukaan balok ditunjukkan pada tabel-tabel berikut ini.

Tabel 5.1 Informasi titik objek 3 dimensi permukaan 1

Titik	X	Y	Z
A	10	10	30
B	30	10	30
C	30	40	30
D	10	40	30

Tabel 5.2 Informasi titik objek 3 dimensi permukaan 2

Titik	X	Y	Z
A	10	10	30
B	30	10	30
C	30	10	10
D	10	10	10

Tabel 5.3 Informasi titik objek 3 dimensi permukaan 3

Titik	X	Y	Z
A	10	10	10
B	30	10	10
C	30	40	10
D	10	40	10

Tabel 5.4 Informasi titik objek 3 dimensi permukaan 4

Titik	X	Y	Z
A	10	10	30
B	10	10	10
C	10	40	10
D	10	40	30

Tabel 5.5 Informasi titik objek 3 dimensi permukaan 5

Titik	X	Y	Z
A	10	40	30
B	30	40	30
C	30	40	10
D	10	40	10

Tabel 5.6 Informasi titik objek 3 dimensi permukaan 6

Titik	X	Y	Z
A	30	10	30
B	30	10	10
C	30	40	10
D	30	40	30

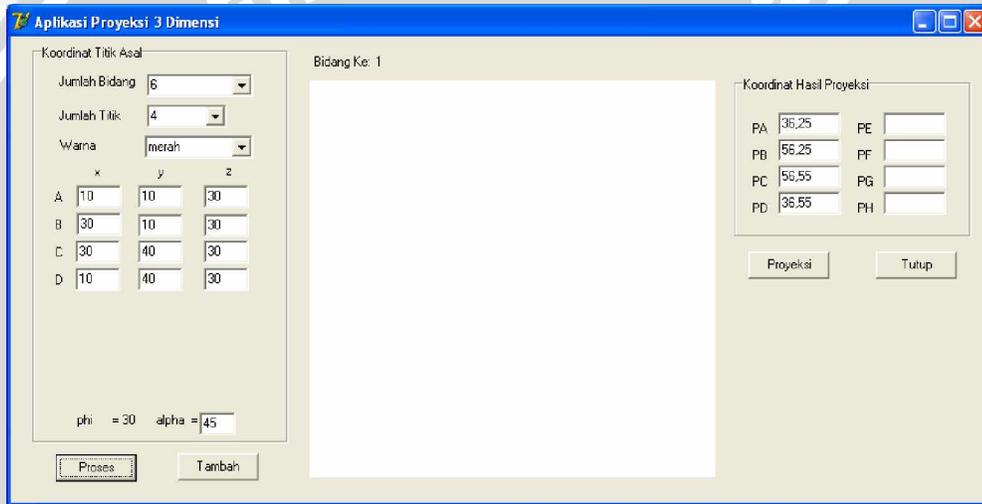
Penentuan warna untuk setiap bidang adalah berdasarkan tabel berikut:

Tabel 5.7 Informasi warna tiap permukaan

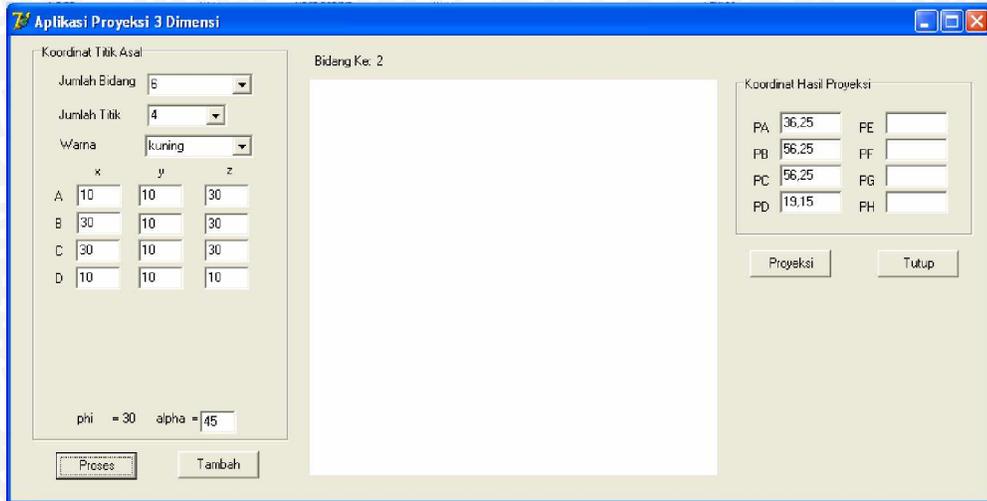
Permukaan	Warna
1	Merah
2	Kuning
3	Hijau
4	Biru
5	Ungu
6	Abu-abu

Selanjutnya data-data titik objek 3 dimensi tersebut bersama data nilai α yang dimasukan oleh pengguna dan $\Phi=30$ yang merupakan nilai konstanta diproses untuk mendapatkan nilai titik hasil proyeksinya (x_p, y_p) dengan menerapkan persamaan proyeksi *oblique*. Data-data yang telah ditentukan tersebut kemudian diimplementasikan ke dalam aplikasi. Untuk pengujian digunakan 3 buah masukan nilai α antara lain 45, 60, dan 90 dan dalam setiap prosesnya adalah sama untuk tiap bidang. Berikut ini adalah penyajian data-data tersebut pada aplikasi proyeksi.

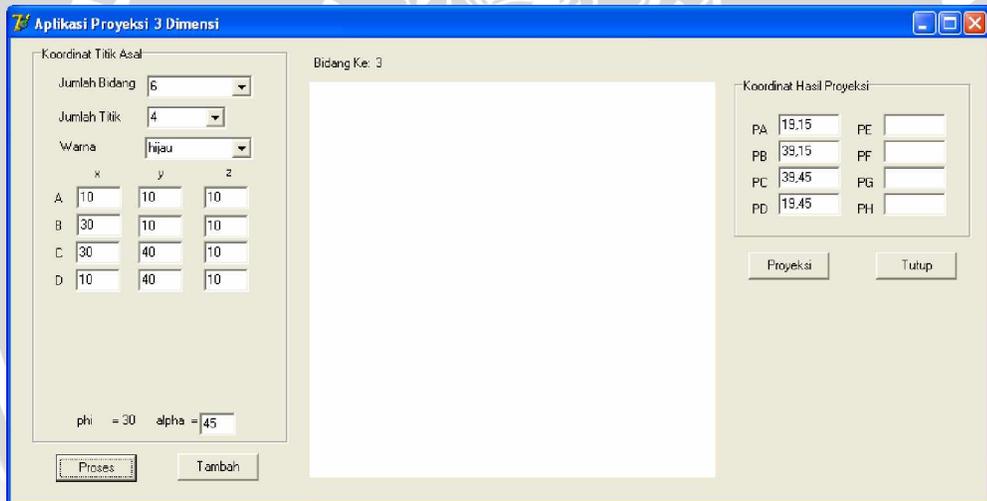
Untuk proyeksi dengan nilai $\alpha=45$:



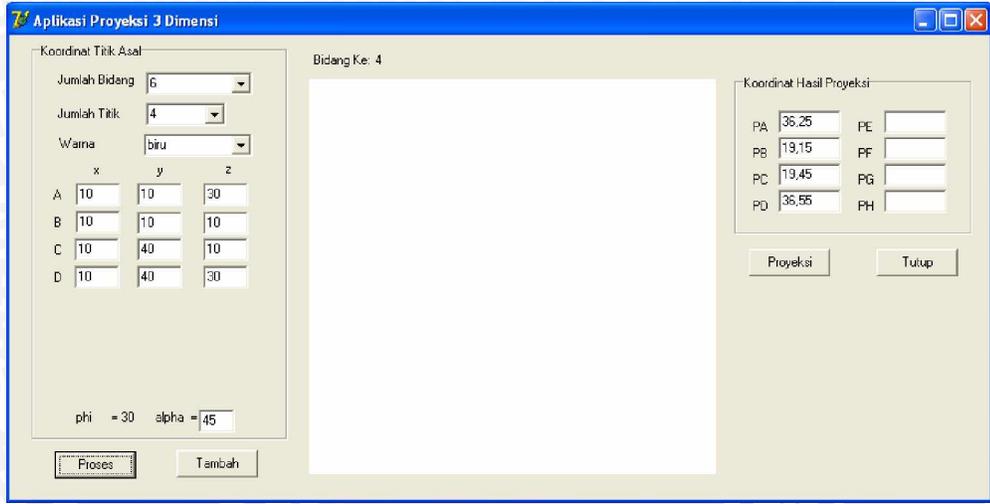
Gambar 5.2 Tampilan aplikasi saat memproses permukaan 1 ($\alpha=45$)



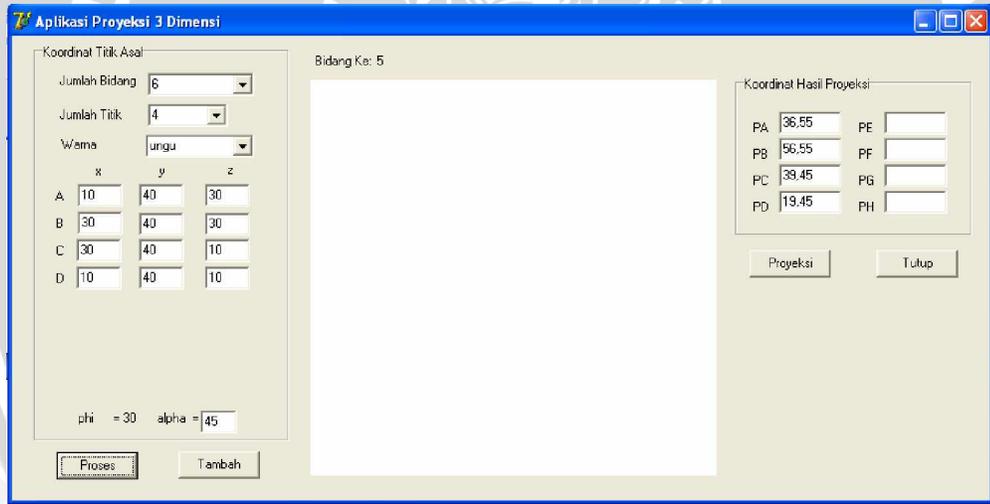
Gambar 5.3 Tampilan aplikasi saat memproses permukaan 2 ($\alpha=45$)



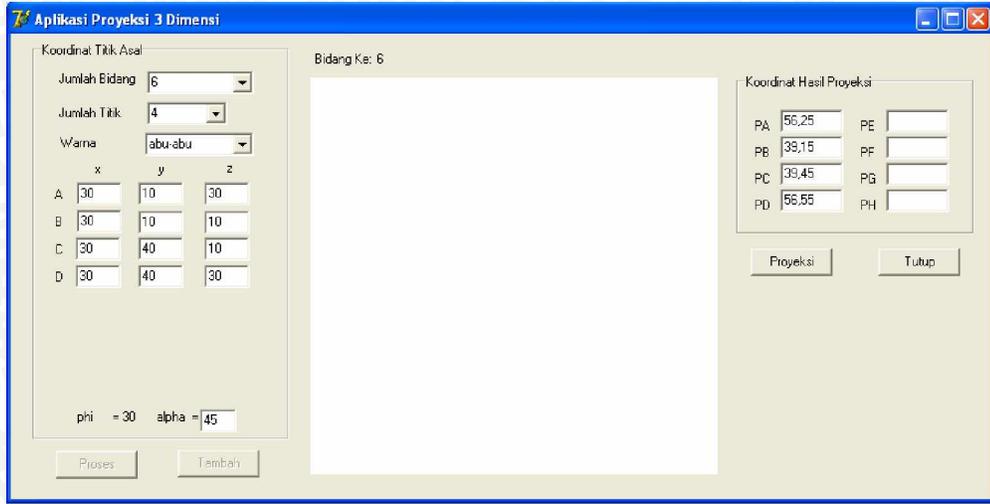
Gambar 5.4 Tampilan aplikasi saat memproses permukaan 3 ($\alpha=45$)



Gambar 5.5 Tampilan aplikasi saat memproses permukaan 4 ($\alpha=45$)

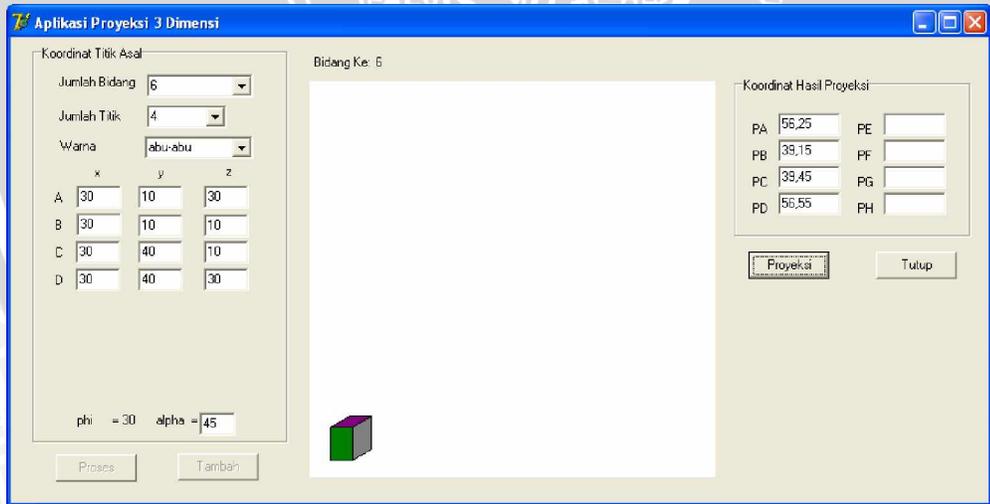


Gambar 5.6 Tampilan aplikasi saat memproses permukaan 5 ($\alpha=45$)



Gambar 5.7 Tampilan aplikasi saat memproses permukaan 6 ($\alpha=45$)

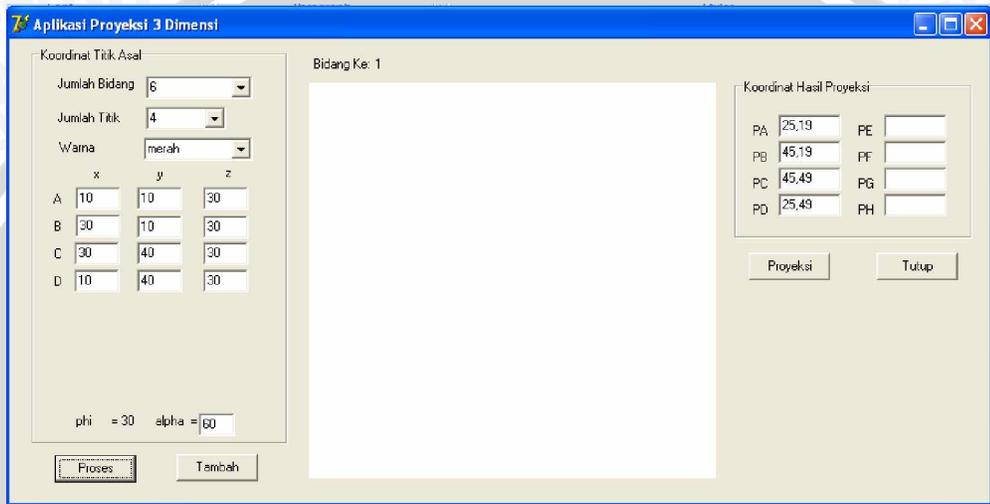
Pada gambar 5.7 tampak bahwa ketika memproses bidang terakhir, tombol proses dan tombol tambah akan nonaktif. Selanjutnya data-data permukaan akan diproses untuk menghasilkan proyeksi objek 3 dimensi. Berikut adalah tampilan saat tombol proyeksi ditekan.



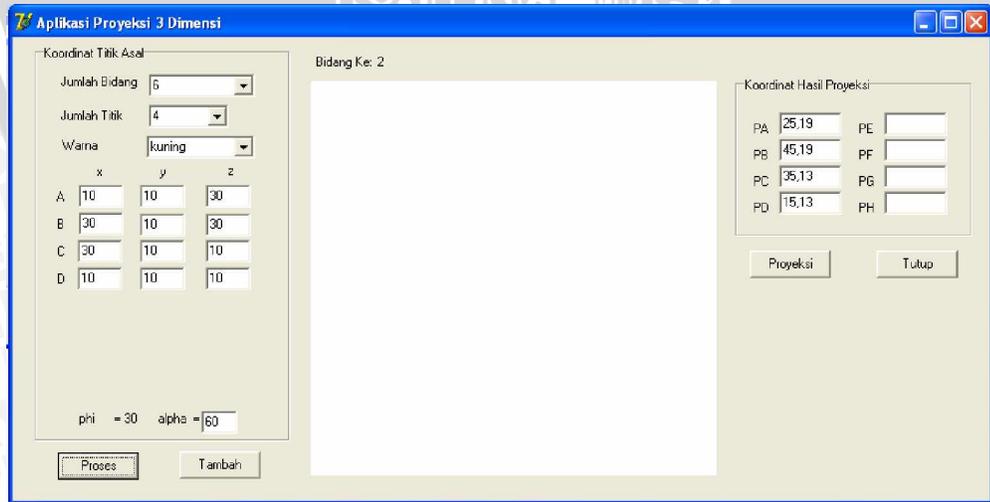
Gambar 5.8 Tampilan aplikasi saat memproyeksikan objek 3 dimensi ($\alpha=45$)

Dari gambar 5.8 dapat terlihat objek 3 dimensi yang telah diproyeksi terhadap bidang proyeksi 2 dimensi sesuai dengan yang diinginkan, yaitu berbentuk balok. Bidang yang tampak pada proyeksi tersebut adalah permukaan 3,

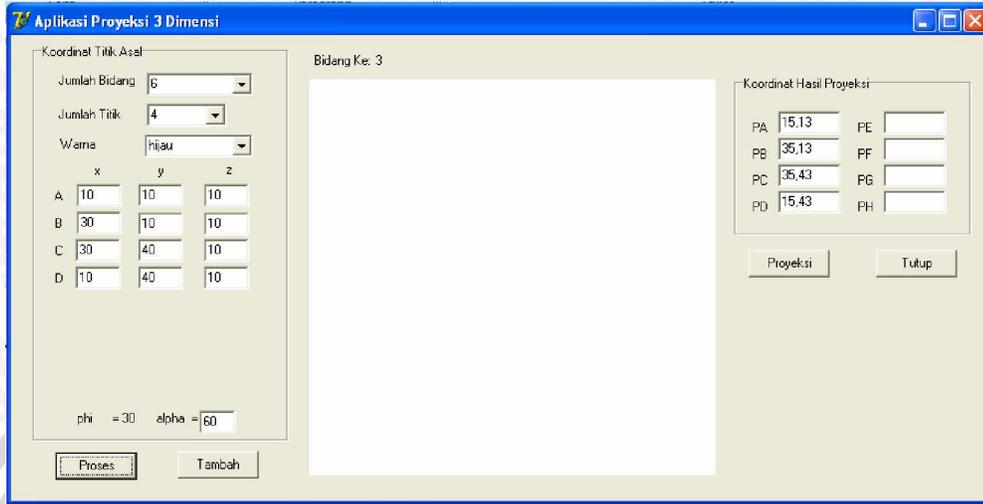
permukaan 6, dan permukaan 5. Permukaan 1, berdasarkan penjumlahan nilai z pada tiap titiknya, memiliki nilai z_{max} paling tinggi yaitu 120, terletak paling belakang dan tidak tampak oleh pemandang. Sedangkan permukaan 3, berdasarkan penjumlahan nilai z pada tiap titiknya, memiliki nilai z_{max} paling kecil yaitu 40, sehingga terletak paling depan. Untuk proyeksi dengan nilai $\alpha=60$:



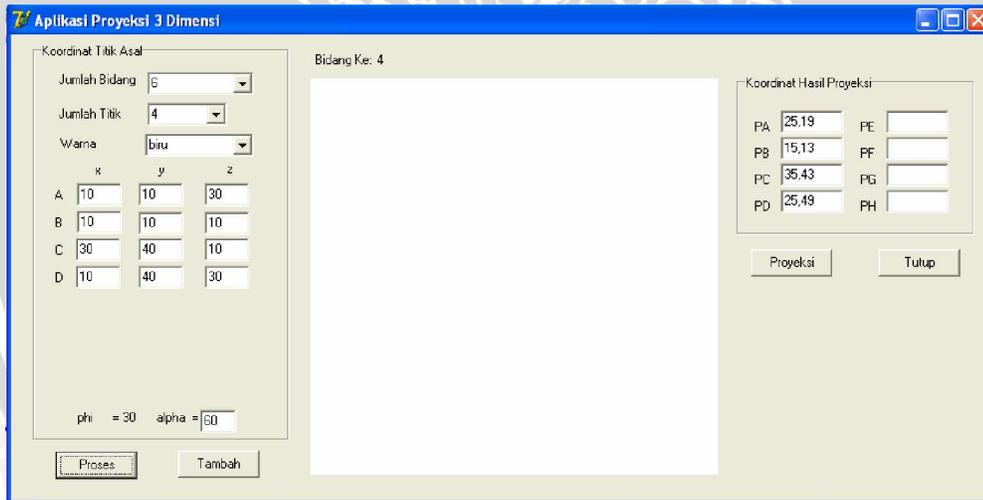
Gambar 5.9 Tampilan aplikasi saat memproses permukaan 1 ($\alpha=60$)



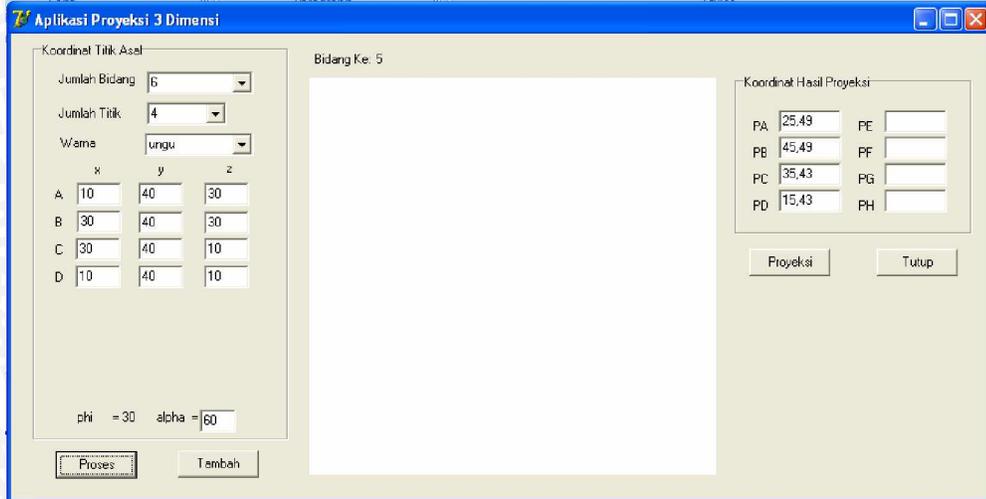
Gambar 5.10 Tampilan aplikasi saat memproses permukaan 2 ($\alpha=60$)



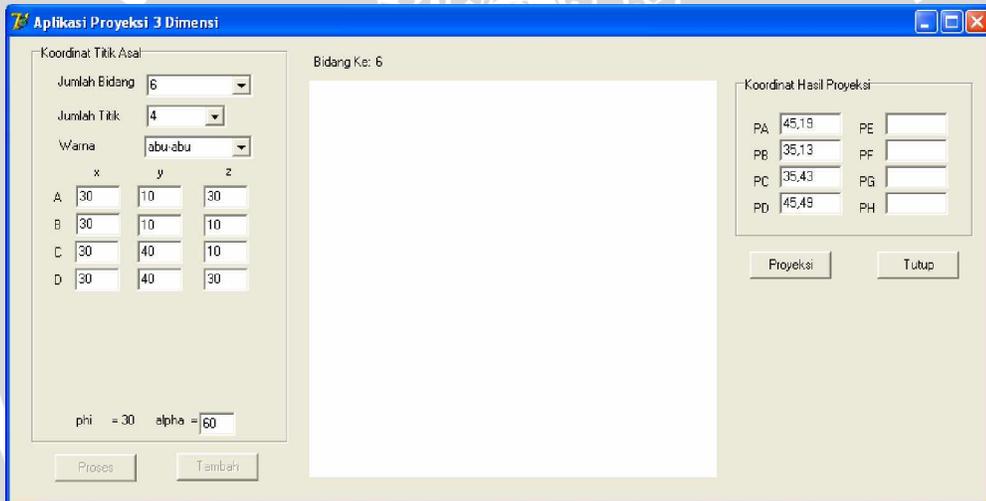
Gambar 5.11 Tampilan aplikasi saat memproses permukaan 3 ($\alpha=60$)



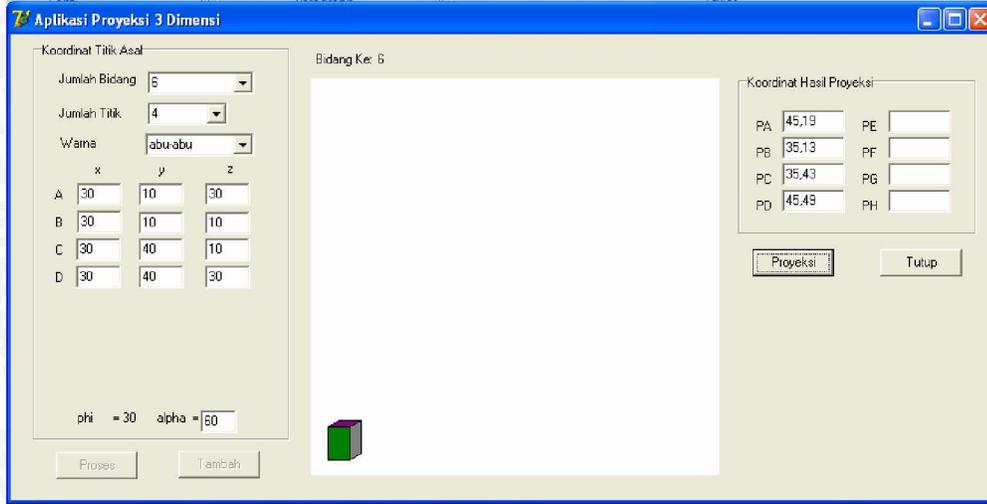
Gambar 5.12 Tampilan aplikasi saat memproses permukaan 4 ($\alpha=60$)



Gambar 5.13 Tampilan aplikasi saat memproses permukaan 5 ($\alpha=60$)



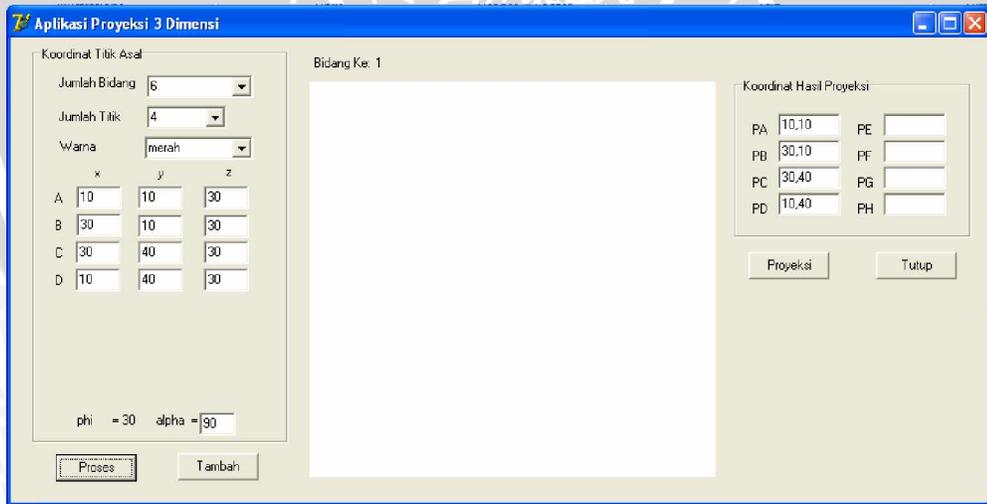
Gambar 5.14 Tampilan aplikasi saat memproses permukaan 6 ($\alpha=60$)



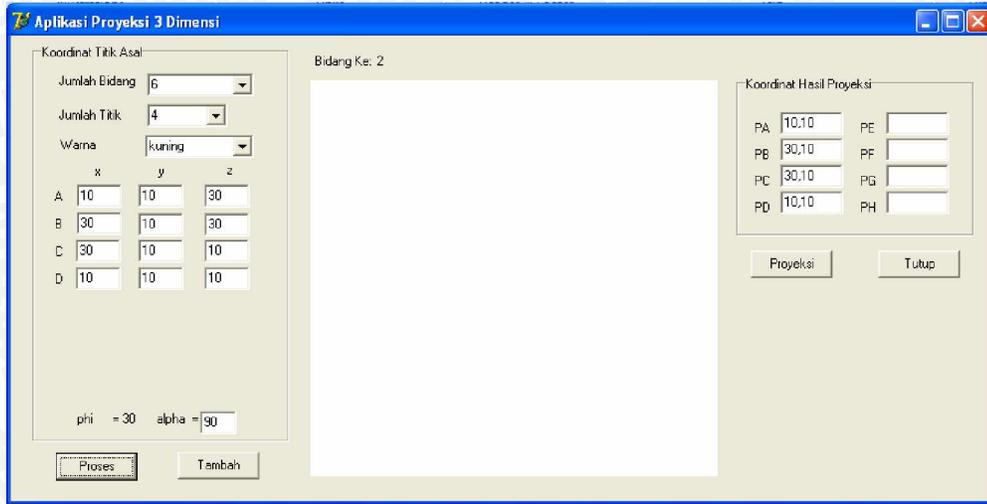
Gambar 5.15 Tampilan aplikasi saat memproyeksikan objek 3 dimensi ($\alpha=60$)

Pada gambar 5.15 terlihat bahwa proyeksi objek 3 dimensi memiliki panjang *shear* sisi lebih pendek dari proyeksi objek dengan nilai $\alpha=45$.

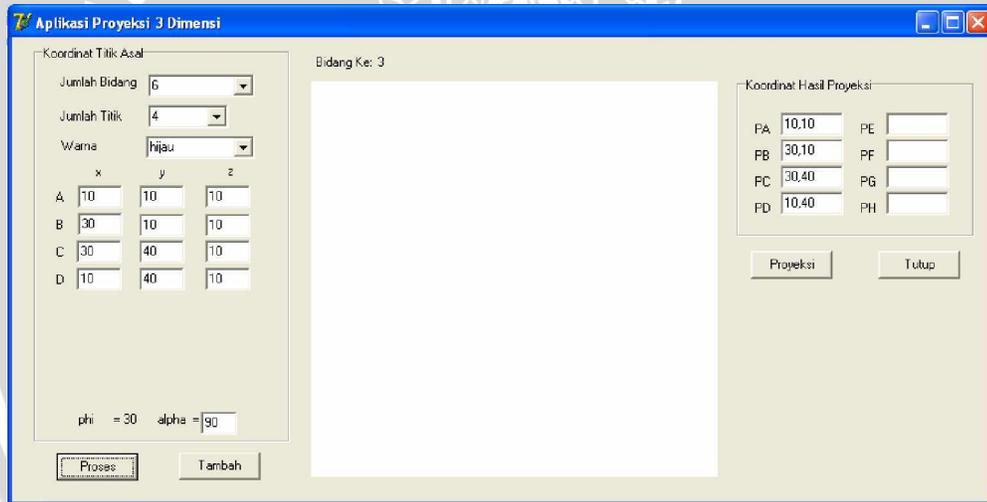
Untuk proyeksi dengan nilai $\alpha=90$



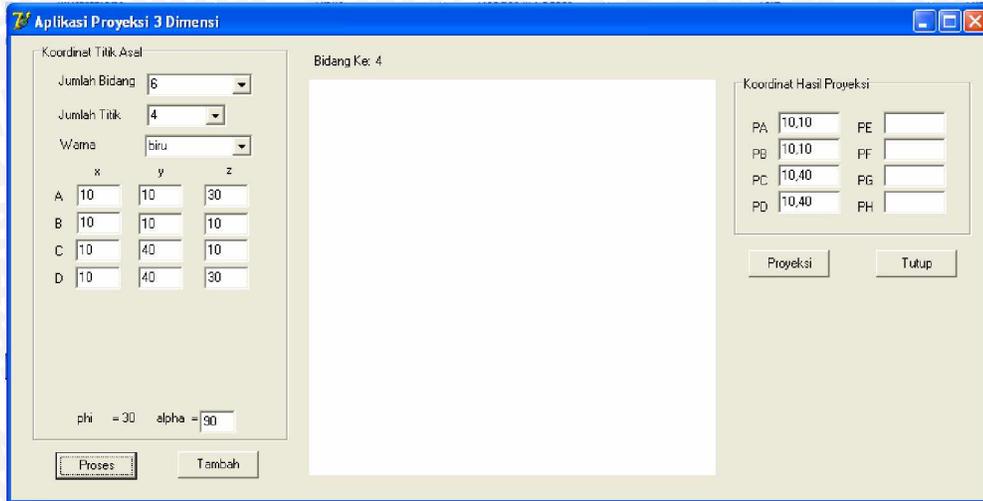
Gambar 5.16 Tampilan aplikasi saat memproses permukaan 1 ($\alpha=90$)



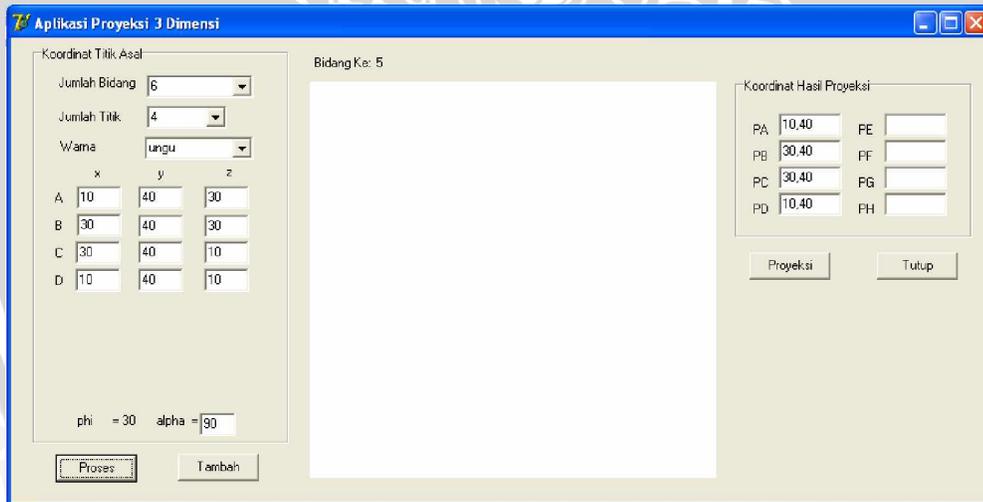
Gambar 5.17 Tampilan aplikasi saat memproses permukaan 2 ($\alpha=90$)



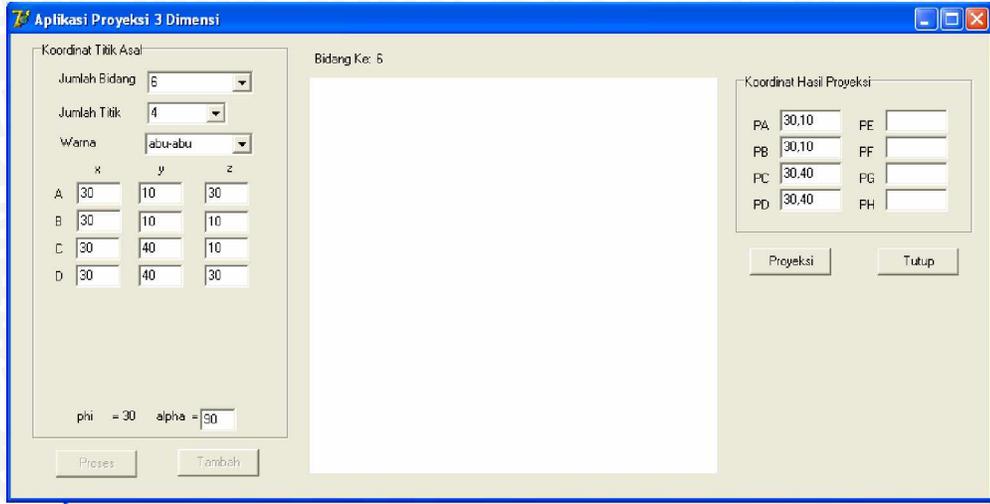
Gambar 5.18 Tampilan aplikasi saat memproses permukaan 3 ($\alpha=90$)



Gambar 5.19 Tampilan aplikasi saat memproses permukaan 4 ($\alpha=90$)

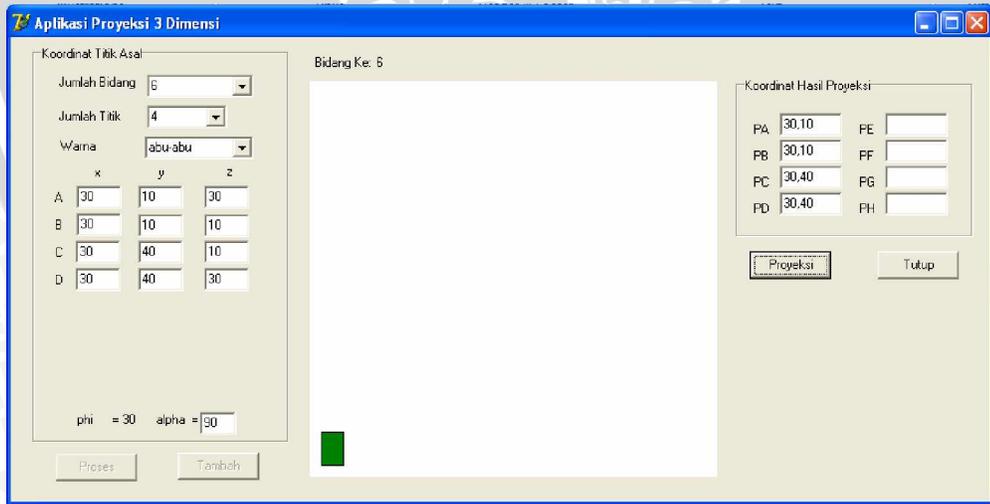


Gambar 5.20 Tampilan aplikasi saat memproses permukaan 5 ($\alpha=90$)



Gambar 5.21 Tampilan aplikasi saat memproses permukaan 6 ($\alpha=90$)

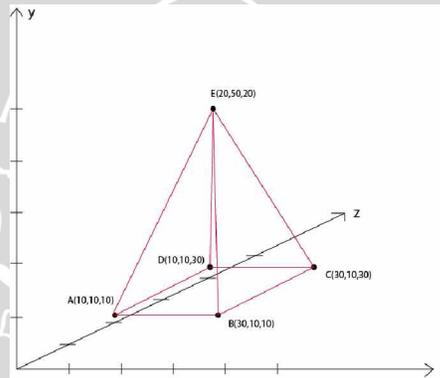
Dari gambar-gambar diatas, terlihat bahwa nilai titik hasil proyeksi (x_p, y_p) untuk tiap-tiap titik adalah sama dengan nilai x, y dari input. Hal ini dikarenakan nilai $L_1=0$ ($1/\tan 90=0$), sehingga diperoleh proyeksi *orthographic*. Berikut adalah hasil proyeksinya.



Gambar 5.22 Tampilan aplikasi saat memproyeksikan objek 3 dimensi ($\alpha=90$)

Pada gambar 5.22, permukaan yang nampak hanyalah permukaan 3 karena terletak paling dekat dengan pemandang. Proyeksi ini menghasilkan proyeksi objek tampak depan. Pada sudut ini, posisi pemandang dengan objek adalah saling berhadapan.

Untuk pengujian dengan objek limas segiempat, memiliki 5 permukaan dengan 4 permukaan (sisi-sisi limas) berupa segitiga sehingga masing-masing memiliki 3 titik, sedangkan alasnya berupa segiempat sehingga memiliki 4 titik. Data titik-titik dari limas segiempat dapat ditunjukkan sebagai berikut:



Gambar 5.23 Objek 3 dimensi yang diujikan (limas segiempat)

Berikut adalah tabel data titik untuk tiap permukaan limas segiempat:

Tabel 5.8 Informasi titik objek 3 dimensi (limas) permukaan 1

Titik	X	Y	Z
A	10	10	30
B	30	10	30
C	20	50	20

Tabel 5.9 Informasi titik objek 3 dimensi (limas) permukaan 2

Titik	X	Y	Z
A	10	10	10
B	10	10	30
C	20	50	20

Tabel 5.10 Informasi titik objek 3 dimensi (limas) permukaan 3

Titik	X	Y	Z
A	10	10	10
B	30	10	10
C	20	50	20

Tabel 5.11 Informasi titik objek 3 dimensi (limas) permukaan 4

Titik	X	Y	Z
A	30	10	10
B	30	10	30
C	20	50	20

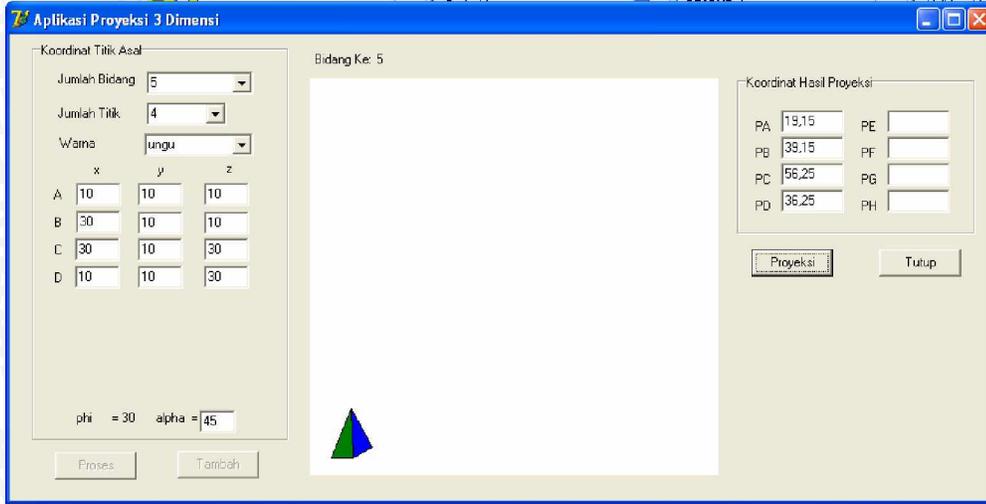
Tabel 5.12 Informasi titik objek 3 dimensi (limas) permukaan 5

Titik	X	Y	Z
A	10	10	10
B	30	10	10
C	30	10	30
D	20	50	20

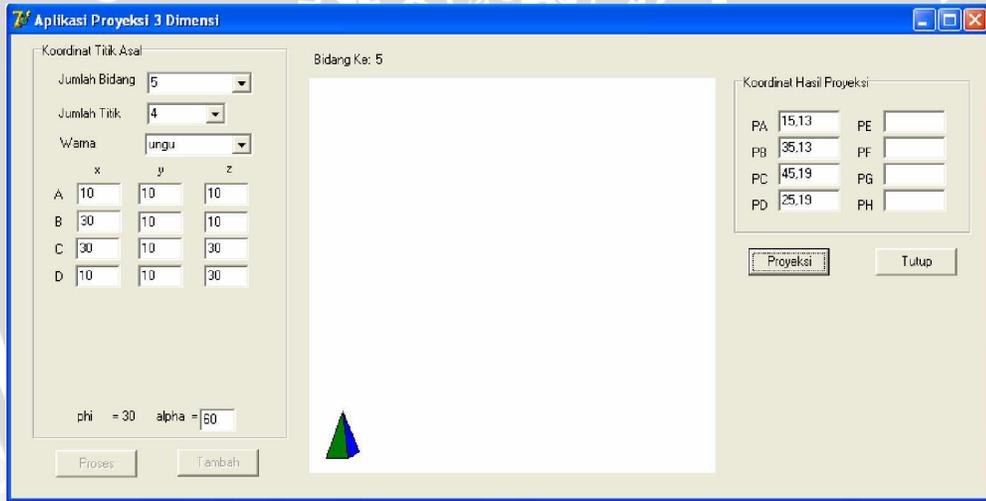
Tabel 5.13 Informasi warna tiap permukaan limas

Permukaan	Warna
1	Merah
2	Kuning
3	Hijau
4	Biru
5	Ungu

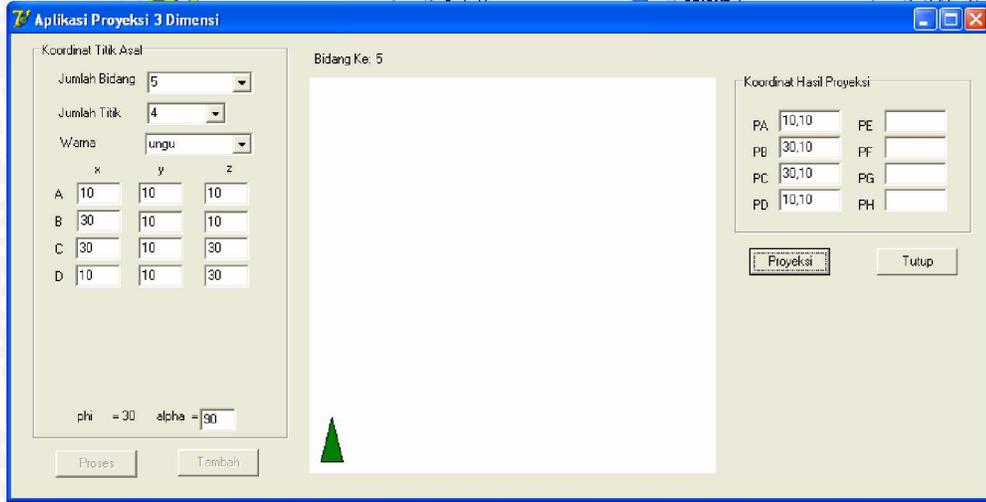
Dari data-data titik permukaan objek tersebut, objek akan diproyeksikan dengan nilai α yaitu 45, 60, dan 90. Berikut adalah hasil proyeksi dari objek 3 dimensi tersebut untuk tiap-tiap nilai α yang berbeda.



Gambar 5.24 Tampilan aplikasi saat memproyeksikan objek 3 dimensi (limas) ($\alpha=45$)

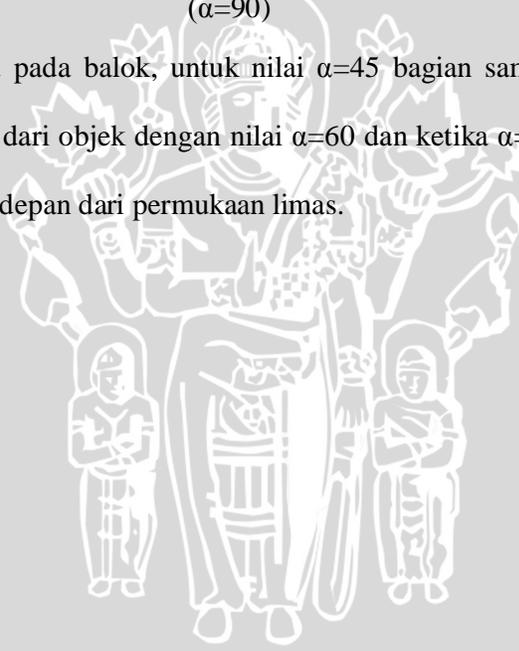


Gambar 5.25 Tampilan aplikasi saat memproyeksikan objek 3 dimensi (limas) ($\alpha=60$)



Gambar 5.26 Tampilan aplikasi saat memproyeksikan objek 3 dimensi (limas) ($\alpha=90$)

Seperti halnya pada balok, untuk nilai $\alpha=45$ bagian samping dari limas tampak lebih panjang dari objek dengan nilai $\alpha=60$ dan ketika $\alpha=90$, bagian yang terlihat adalah bagian depan dari permukaan limas.



BAB VI PENUTUP

6.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, pengujian dan analisis sistem maka dapat diambil kesimpulan sebagai berikut:

1. Proyeksi paralel *oblique* menghasilkan proyeksi objek 3 dimensi dimana posisi benda terhadap bidang proyeksi tetap, tapi arah pandang mata berpindah ke samping atau miring. Sehingga menghasilkan proyeksi objek 3 dimensi yang dapat dilihat ketiga bagiannya yaitu bagian atas, depan, dan samping.
2. Sudut α merupakan sudut pandangan pergeseran ke kiri atau ke kanan (pergerakan horizontal) memandang terhadap objek 3 dimensi. Semakin besar nilai sudut α , maka kemiringan dari sisi objek (*shear*) yang terproyeksi semakin mengecil dan ketika nilai sudut $\alpha=90$ maka akan menghasilkan proyeksi *orthographic* yang mana posisi memandang dengan objek adalah saling berhadapan.

6.2 Saran

Dalam perancangan dan pembuatan aplikasi proyeksi ini masih terdapat kekurangan dan kelemahan, oleh karena itu masih diperlukan adanya penyempurnaan dalam rangka pengembangan ke depan. Adapun hal yang dapat disempurnakan antara lain:

1. Penyempurnaan algoritma dalam menentukan letak proyeksi untuk bidang dengan nilai z_{max} yang sama.
2. Pengembangan untuk bidang permukaan dengan sisi tidak tegak lurus dan bidang tidak datar.

DAFTAR PUSTAKA

- [1] __. http://elib.unikom.ac.id/files/disk1/37/jbptunikompp-gdl-s1-2005-donnyreza1-1827-bab_ii.doc. 5 Maret 2012.
- [2] Guy, Stephen J. 2010. "3D Viewing Pipeline - Introduction to Computer Graphics".
http://comp575.web.unc.edu/files/2010/10/12_3DViewingPipeline.pdf. 15 Juli 2012.
- [3] Harrington, Steven. "Computer Graphics A Programming Approach Second Edition". McGraw-Hill Book Company. 1987.
- [4] Hearn, Donald. Baker, M. Pauline. "Computer Graphics Second Edition". Prentice-Hall International Inc. 1994.
- [5] Wikipedia. 2012. "Paralel Projection".
http://en.wikipedia.org/wiki/Paralel_projection. 8 Maret 2012.
- [6] Wikipedia. 2012. "Borland Delphi".
http://id.wikipedia.org/wiki/Borland_Delphi. 8 Maret 2012.

