**PEMBANGUNAN PROGRAM MINIATUR**

**WAHANA BERMAIN 3D**

**LAPORAN TUGAS GRAFIK KOMPUTER**



1. Erlangga Ganjar Prayoga (10108459)
2. Tri Gunawan (10108444)
3. Afdhal Zikri (10108425)
4. Ivaldhy Graha (10107445)

**PROGRAM STUDI S1**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK DAN ILMU KOMPUTER**

**UNIVERSITAS KOMPUTER INDONESIA**

**2012**

**BAB I**

**PENDAHULUAN**

* 1. **Latar Belakang**

Pada pembuatan animasi atau miniatur 3D harus sesuai dengan model dan ketentuan yang ada, untuk itu dalam tugas besar grafik komputer ini kami membuat salah satu wahana tempat bermain seperti dufan yang berbentuk tiga dimensi, dimana wahana ini bergerak seperti wahana yang ada di dufan pada umumnya.

Wahana bermain ini dibangun dengan arsitektur yang biasa dan memiliki bentuk yang terbilang sederhana walaupun bentuk wahana ini tidak terlalu menyerupai seperti bentuk aslinya.

Wahana bermain yang ingin kami buat yaitu wahana bermain bianglala, histeria, dan rajawali. Wahana bianglala berbentuk lingkaran dan bergerak berputar secara vertikal serta memiliki tempat bagi para pengunjung atau penumpang untuk duduk. Untuk wahana bermain histeria, wahana ini bergerak naik turun dan memilki penyangga seperti tiang yang berdiri di tengah secara vertikal. Sedangkan wahana rajawali seperti penggabungan antara wahana bianglala dan histeria, wahana rajawali bergerak berputar secara horizontal dan bergerak naik turun seperti halnya wahana bermain rajawali yang ada di dufan.

* 1. **Identifikasi Masalah**

Wahana bermain 3D yang akan dibuat sangat bergantung pada ketepatan posisi, putaran, dan penempatan lokasi sehingga tidak terlihat mencolok atau aneh. Serta pergerakan atau perputaran dari wahana bermain 3D ini diharapkan memberi kesan yang hampir sama dengan wahana bermain yang ada di dufan pada umumnya.

* 1. **Maksud dan Tujuan**

Maksud dari pembuatan wahana bermain 3D ini yaitu untuk memenuhi salah satu tugas besar grafik komputer.

Tujuan yang akan di capai dari pembuatan wahana bermain 3D ini adalah:

1. Memberikan ketertarikan kepada pengunjung yang melihat.
2. Memberikan kesan sama dengan wahana yang ada di dufan mulai dari bentuk dan gerakan walaupun tidak 100% sama.
   1. **Metodologi Penelitian**
3. Tahap pengumpulan data dalam pembuatan program ini sebagai berikut :
4. Studi literatur

Mengumpulkan data-data yang diperlukan dalam pembuatan program ini.

1. Tahap pembuatan peramgkat lunak
2. *System / Information Engineering*

Menyediakan hal-hal yang dibutuhkan untuk mendukung pembuatan wahana bermain 3D.

1. *Analisis*

Tahap awal yang dilakukan dalam pembuatan program ini memperhatikan dan menyediakan hal-hal yang dibutuhkan dalam pembuatan wahana bermain ini.

1. *Design*

Membuat program ini semenarik mungkin agar pengunjung tertarik untuk melihat wahana bermain 3D yang dibuat.

1. *Coding*

Ketika penggunaan coding dalam pembuatan program ini disesuaikan dengan bahasa pemrograman yang digunakan.

1. *Pengujian*

Pengujian dilakukan di komputer atau laptop penulis.

* 1. **Batasan Masalah**

Dalam pembuatan wahana bermain 3D ini terdapat beberapa batasan masalah. Adapun batasan masalahnya adalah :

* + 1. Bahasa pemrograman yang digunakan adalah bahasa C++.
    2. Komponen-komponen OpenGl yang digunakan diantaranya lglut32, lglu32, lopengl32, lwinmm, lgdi32, dan modeling.

Fitur yang tersedia pada miniatur wahana bermain 3D ini adalah interaksi pergerakan kamera dengan menggunakan tombol-tombol yang terdapat pada *keyboard* yaitu panah atas, bawah, kiri, dan kanan.

* 1. **Gambaran Sistem**

Apabila aplikasi ini dibuka maka akan tampak semua fasilitas wahana bermain karena kamera yang ada pada miniatur wahana bermain ini bisa digerakkan sesuai arah yang telah dibuat menggunakan tombol-tombol yang ada di keyboard.

* 1. **Sistematika Penulisan**

Isi dari penulisan ini berhubungan dengan data-data yang telah diambil serta referensi dari buku dan makalah, disini setiap bab terbagi dalam beberapa sub bab, sebagai berikut :

**BAB I PENDAHULUAN**

Pada bab ini merupakan bab pendahuluan yang terdiri dari latar belakang, identifikasi masalah, maksud dan tujuan, metedologi penelitian, batasan masalah, gambaran sistem, dan sistematika penulisan.

**BAB II LANDASAN TEORI**

Pada bab ini berisi materi tentang pengertian dasar dan jenis-jenis tentang grafik komputer, serta pengertian dari OpenGL.

**BAB III ANALISA DAN PEMBAHASAN MASALAH**

Bab ini menjelaskan tentang analisa dari langkah-langkah pembuatan wahana bermain 3D yang mencakup source code dengan OpenGL, dan penjelasan fungsi-fungsi source code.

**BAB IV PENUTUP**

Bab ini merupakan penutup yang berisi kesimpulan dari semua analisa dan pembahasan dalam pembuatan program ini, dan saran-saran dari penulis.

**BAB II**

**LANDASAN TEORI**

* 1. **Pengertian Grafik komputer**

Grafik Komputer adalah suatu proses pembuatan, penyimpanan dan manipulasi model dan citra. Model berasal dari beberapa bidang seperti fisik, matematik, artistik dan bahkan struktur abstrak.

Istilah ”Grafik Komputer”ditemukan tahun 1960 oleh William Fetter : pembentukan disain model cockpit (Boeing) dengan menggunakan pen plotter dan referensi model tubuh manusia 3 Dimensi.

Pengguna mengendalikan isi, struktur dan kemunculan objek serta menampilkan citra melalui suatu komponen dasar visual feedback.

Komponen Dasar Sistem Grafik Interaktif :

* + Masukan : mouse, tablet dan stylus, peralatan force feedback, scanner, live video stream, dll
  + Proses dan Penyimpanan
  + Keluaran : layar, printer berbasis kertas, perekam video, non-linear editor, dll.
  1. **Pengertian OpenGL**

OpenGL (Open Graphics Library) adalah spesifikasi standar yang mendefinisikan sebuah [cross-bahasa](http://id.wikipedia.org/w/index.php?title=Cross-bahasa&action=edit&redlink=1), cross-platform [API](http://id.wikipedia.org/wiki/API) untuk menulis aplikasi yang menghasilkan komputer [2D](http://id.wikipedia.org/wiki/2D) dan [3D](http://id.wikipedia.org/wiki/3D) grafis. Antarmuka terdiri dari lebih dari 250 panggilan fungsi yang berbeda yang dapat digunakan untuk menggambar tiga dimensi yang kompleks adegan-adegan dari primitif sederhana. OpenGL dikembangkan oleh [Silicon Graphics Inc](http://id.wikipedia.org/w/index.php?title=Silicon_Graphics_Inc&action=edit&redlink=1) (SGI) pada tahun 1992 [2] dan secara luas digunakan dalam CAD, virtual reality, visualisasi ilmiah, visualisasi informasi, dan simulasi penerbangan. Hal ini juga digunakan dalam video game, di mana bersaing dengan Direct3D on Microsoft Windows platform (lihat vs OpenGL Direct3D). OpenGL dikelola oleh sebuah teknologi nirlaba konsorsium, yang Khronos Group.

OpenGL adalah suatu graphic library yang sebagian bersifat open source, dipakai pada banyak platform (windows, linux) dan dapat digunakan pada berbagai jenis compiler seperti C++ atau Delphi.

OpenGL bukanlah bahasa pemrograman tetapi merupakan suatu Application Programming Interface (API).

Sintaks perintah OpenGL mengikuti aturan penulisan dari library dimana fungsi tersebut berasal, format penulisan fungsi OpenGL adalah : **<awalan library><perintah><optional jumlah argumen><optional tipe argumen>** Semua perintah OpenGL menggunakan awalan gl diikuti dengan huruf kapital pada setiap kata membentuk nama perintah (sebagai contoh glClearColor). Untuk mendefinisikan konstanta diawali dengan GL\_, dengan menggunakan huruf kapital dan garis bawah untuk memisahkan kata (seperti GL\_POLY\_STIPPLE). Terkadang beberapa huruf dan angka ditambahkan pada akhir perintah (seperti 3f pada glVertex3f). Dalam hal ini angka 3 menunjukkan berapa banyak argumen yang harus ada pada perintah tersebut dan akhiran huruf f menunjukkan jenis datanya yaitu floating. Sebagai contoh pada dua perintah berikut ini :

*glVertex3i(1,0,-2);*

*glVertex3f(1.0, 0.0, -2.0);*

adalah sama yaitu meletakkan titik di layar pada koordinat x = 1, y = 0 dan z = -2, perbedaannya yaitu pada perintah pertama menspesifikasikan titik dengan tipe data integer 32-bit, sedangkan yang kedua dengan tipe data single precision floating point.

Beberapa perintah OpenGL menambahkan perintah huruf akhir v yang menunjukkan bahwa perintah tersebut menggunakan pointer ke array/vektor. Di bawah ini contoh perbedaannya.

*float color\_array[ ]={1.0,0.0,0.0}*

*glColor3f (1.0,0.0,0.0);*

*glColor3fv(color\_array);*

**BAB III**

**ANALISA DAN PEMBAHASAN MASALAH**

**3.1 Penjelasan Fungsi- Fungsi Source Code**

**• #include <math.h>**

**#include <stdlib.h>**

**#include <GL/glut.h>**

**#include <stdio.h>**

**#include <string.h>**

**#include <assert.h>**

**#include <stdarg.h>**

Fungsi ini digunakan untuk menggunakan fungsi GLUT yang ada dalam Windows

**• static float**

Berfungsi untuk mendeklarasikan posisi awal X, Y, Z

**• glColor3f (1,0,0)**

Berfungsi untuk mendefinisikan warna dari objek yang dibuat dengan warna (1,0,0) yaitu warna merah

**• glVertex3d(x1,y1,z2)**

Berfungsi untuk menyatakan sebuah titik koordinat x, y dan z.

**• glClearColor (0.0, 0.0, 0.0, 0.0);**

Berfungsi Mendefinisikan warna dari windows yang dibuat dengan warna (0,0,0) yaitu warna hitam

**• glOrtho(-12,12,-6,6,-35,35);**

Mendefinisikan besarnya sistem koordinat dengan range sumbu x adalah [-12,12], range untuk sumbu y adalah [-6,6], range untuk sumbu z adalah [-35,35]

**• glEnable(GL\_DEPTH\_TEST);**

Berfungsi untuk mengaktifkan atau menampilkan GL\_DEPTH\_TEST

**• glShadeModel (GL\_SMOOTH);**

Berfungsi untuk memberi warna bayangan dengan satu atau banyak warna

**• glMatrixMode (GL\_PROJECTION);**

Berfungsi untuk memberi nilai masukan pada matrix dengan model projection.

**• glLoadIdentity ();**

Fungsi untuk memanggil matriks identitas ke matriks saat ini, agar nilai-nilai awal kembali ke pusat koordinat (0,0,0)

**• gluPerspective(80.0,2.2, 1.0, 20.0);**

Berfungsi untuk mendeklarasikan jarak ke depan dan ke belakang dari bidang pandang (clipping plane)

**• glTranslatef(a,b,c);**

Fungsi ini menggeser seluruh pemandangan ke kiri-kanan, atas-bawah, maju-mundur tergantung nilai x, y, dan z

**• glRotatef(xpoz,1,0,0);**

Fungsi untuk memutar seluruh obyek dengan sudut angle (0–359) dan vektor yang dinyatakan dengan nilai x, y, dan z

**• glBegin(GL\_QUADS);**

Berfungsi untuk membentuk segi empat

**• glBegin(GL\_TRIANGLES);**

Berfungsi untuk membentuk segitiga, dan tiap segitiga saling lepas

**• glBegub(GL\_LINES);**

Berfungsi untuk membentuk garis

**• glutSwapBuffers();**

Digunakan untuk menukar bagian belakan buffer menjadi buffer layar (screen buffer). Dalam modus double-buffered, perintah menggambar pada layar, tetapi menggambar pada buffer belakang layar. Pada saat ingin menampilkan gambar, gambar yang berada di buffer belakang layar dipindahkan ke buffer layar, sehingga menghasilkan animasi yang sempurna.

**• glutInitDisplayMode (GLUT\_RGB | GLUT\_DEPTH | GLUT\_DOUBLE);**

Untuk inisialisasi modus display. Fungsi tersebut meminta window yang bersifat double buffered, dengan warna RGB

**• glutInitWindowSize (500, 500); glutInitWindowPosition (100, 100);**

Membuat windows dengan ukuran(500,500) dengan titik kiri atas jendela diletakkan pada posisi (100,100) di layar computer

**• glutKeyboardFunc(keyboard);**

Berfungsi untuk mengaktifkan fungsi pada keyboard

**3.2** **Deklarasi variabel.**

**• Variable sudut pada saat animasi**

float angle2=0.0;

**• Variable untuk manipulasi sudut pandang**

static float angle=0.0,ratio; //Sudut perputaran kamera (terhadap sumbu y)

static float x=0.0f,y=10.75f,z=55.0f; //Posisi Kamera

static float lx=0.0f,ly=0.0f,lz=-1.0f; //Vektor sudut pandang

**• Fungsi untuk menyesuaikan perspektif ketika window dirubah ukurannya**

void changeSize(int w, int h) {

**• Menghindari pembagian oleh 0.**

if(h == 0)

h = 1;

**• Rasio antara lebar dengan tinggi layar**

float ratio = 1.0\* w / h;

**• Reset the coordinate system before modifying**

glMatrixMode(GL\_PROJECTION); //Matrik diset : Matrik Proyeksi

glLoadIdentity(); //Meload matrik identitas

**• Set the viewport to be the entire window**

glViewport(0, 0, w, h);

**• Set the correct perspective.**

gluPerspective(45,ratio,1,1000);

//Parameter perspektif (sudut di sumbu yz, rasio lebar/tinggi,

glMatrixMode(GL\_MODELVIEW); //Set matrix ke GL\_MODELVIEW

glLoadIdentity(); //Load identity matrix

**• Fungsi utama**

int main(int argc, char \*\*argv) {

**• argv -> a pointer to the unmodified argv variable from the main function**

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_DEPTH | GLUT\_DOUBLE | GLUT\_RGBA);

**• Parameter posisi window (x,y)**

glutInitWindowPosition(100,100);

**• Parameter ukuran window (panjang,lebar)**

glutInitWindowSize(640,360);

**• Membuat windows Latihan**

glutCreateWindow("3 wahana bermain");

glClearColor( 0.1, 0.5, 1, 0);

**• Fungsi utama untuk menampilkan objek**

glutDisplayFunc(renderScene);

**• Fungsi yang dijalankan dalam keadaan idle**

glutIdleFunc(renderScene);

**• untuk mempertahankan ukuran objek saat jendela dirubah ukurannya**

glutReshapeFunc(changeSize);

glutSpecialFunc(inputKey);

**• Mengaktifkan depth testing**

glEnable(GL\_DEPTH\_TEST);

**• Mengatur Pencahayaan**

glDepthFunc(GL\_LESS);

glEnable(GL\_LIGHT0);

glEnable(GL\_NORMALIZE);

glEnable(GL\_COLOR\_MATERIAL);

glEnable(GL\_LIGHTING);

glLightfv(GL\_LIGHT0, GL\_AMBIENT, light\_ambient);

glLightfv(GL\_LIGHT0, GL\_DIFFUSE, light\_diffuse);

glLightfv(GL\_LIGHT0, GL\_SPECULAR, light\_specular);

glLightfv(GL\_LIGHT0, GL\_POSITION, light\_position);

glMaterialfv(GL\_FRONT, GL\_AMBIENT, mat\_ambient);

glMaterialfv(GL\_FRONT, GL\_DIFFUSE, mat\_diffuse);

glMaterialfv(GL\_FRONT, GL\_SPECULAR, mat\_specular);

glMaterialfv(GL\_FRONT, GL\_SHININESS, high\_shininess);

**• Never ending loop, agar layar tidak langsung tertutup**

glutMainLoop();

**BAB IV**

**PENUTUP**

* 1. **Kesimpulan**

Kesimpulan yang dicapai penulis dari pembuatan wahana bermain 3D ini adalah:

1. Membuat pengunjung yang melihat program wahana bermain 3D ini menjadi tertarik.
2. Membuat wahana bermain 3D ini bergerak seperti halnya wahana bermain yang ada di dufan.

* 1. **Saran**

Saran dari penulis tentang pembuatan program wahana bermain ini adalah banyaknya model yang disediakan untuk pembuatan objek animasi 3D, sehingga dengan kreasi kita bisa membuat objek animasi apa saja yang kita inginkan.

**DAFTAR PUSTAKA**

[**http://karmila.staff.gunadarma.ac.id/Downloads/folder/0.2**](http://karmila.staff.gunadarma.ac.id/Downloads/folder/0.2)

[**http://pareradesi.blogspot.com/2011/02/pengertian-opengl.html**](http://pareradesi.blogspot.com/2011/02/pengertian-opengl.html)

[**http://elib.unikom.ac.id/download.php?id=36531**](http://elib.unikom.ac.id/download.php?id=36531)