	Laboratorio de Computación Gráfica e Interacción Humano - Computadora	Cortez Ibarra Derek
		314005539 grupo 04
		Fecha: 20 de enero del 2010

MANUAL TECNICO

PROYECTO FINAL

Objetivo:

El alumno deberá aplicar y demostrar los conocimientos adquiridos durante todo el curso.

Descripción del proyecto:

El trabajo consiste en realizar un cuarto, el cual contenga 7 elementos diferentes. Dichos elementos van a ser recreados en Maya y 1 tiene que ser modelado en OpenGL usando primitivas. De estos 7 objetos, 3 deben tener una animación sencilla y 2 deben tener una animación más compleja.

El cuarto debe tener estructura tanto interna como externa, por lo que fuera del cuarto debe representarse la fachada de una casa entendiendo que es la parte exterior del cuarto

Organización:

A continuación, se muestra el diagrama de Gantt de la planeación y elaboración que tuvo el proyecto

Tarea	Diciembre								Enero					
	Primera Semana	Segunda Semana	Tercera Semana	Cuarta Semana	Primera Semana	Segunda Semana	Tercera Semana							
Inicio														
Leer especificaciones del proyecto														
Primer avance de modelos														
Segundo avance de modelos														
Implementación en OpenGL														
Finalización de modelos														
Fachada														
Resto del cuarto														
Texturas														
Fase final														
Animaciones														
Documentación														

Para la subida a GitHub hubo varios problemas, ya que normalmente utilizo el modo gráfico para hacer repositorios, pero como el proyecto abarcaba mucho más de 100 archivos, tuve que aprender algunos comandos para poder subir el repositorio vía la línea de comandos en GitBash

Alcance del proyecto:

Este informe contiene la información sobre la entrega que corresponde al funcionamiento completo del proyecto. Su principal objetivo es el uso de lo aprendido en el semestre

Para lograr el objetivo se necesita modelar objetos en 3D, texturizar los objetos y realizar animaciones

Modelado de Objetos 3D:

Para este paso es necesario utilizar un software a nuestro gusto (Maya) para poder recrear un objeto lo más parecido posible a su imagen de referencia

Texturización de objetos:

Aquí lo complicado es elegir las imágenes correctas para el texturizado, ya que algunas pueden salir borrosas, no cargan los canales RGB en OpenGL o simplemente no cargan y mandan una excepción en el código

Animación:

En la parte final, se debe elegir que elementos pueden realizar una animación y cuales es mejor dejar de forma estática. Después de esto, tratar de ver a cual objeto se le puede hacer una animación un poco más compleja que solamente caminar o girar

Código:

En el proyecto se utilizó como código base, el correspondiente a la Práctica 11 del curso. Ya que a mi parecer es el más completo comparándolo con prácticas anteriores, antes de tener dicho código, se utilizaba el código de la Práctica 9, con éste se hicieron los avances pedidos en días anteriores. Lo que se agregó fue el dibujo de los modelos, sus transformaciones y sus animaciones

```

//Modelos de objetos
Model Computadora1((char*)"Models/Proyecto/Computadora.obj");
Model Computadora2((char*)"Models/Proyecto/Computadora1.obj");
Model Libro((char*)"Models/Proyecto/Libro.obj");
Model Mueble1((char*)"Models/Proyecto/MuebleTv.obj");
Model Mueble2((char*)"Models/Proyecto/muebleTV1.obj");
Model Mueble3((char*)"Models/Proyecto/MuebleTV2.obj");
Model Pikachu((char*)"Models/Proyecto/Pikachu.obj");
Model Reloj1((char*)"Models/Proyecto/Reloj.obj");
Model Reloj2((char*)"Models/Proyecto/Reloj1.obj");
Model Reloj3((char*)"Models/Proyecto/Reloj2.obj");
Model Silla1((char*)"Models/Proyecto/Silla.obj");
Model Silla2((char*)"Models/Proyecto/Silla1.obj");
Model Switch((char*)"Models/Proyecto/Switch.obj");
Model Television((char*)"Models/Proyecto/Television.obj");

//Modelo del resto del cuarto
Model Cuarto((char*)"Models/Cuarto/Cuarto.obj");
//Modelo de fachada
Model Fachada((char*)"Models/Proyecto/Fachada.obj");

```

```

//Carga de modelo
//Television
view = camera.GetViewMatrix();
glm::mat4 model(1);
//tmp = model = glm::translate(model, glm::vec3(0, 1, 0));
model = glm::translate(model, glm::vec3(-10.54f, 1.0f, -1.30f));
model = glm::scale(model, glm::vec3(1.0f, 1.83f, 1.57f));
glUniformMatrix4fv(modelLoc, 1, GL_FALSE, glm::value_ptr(model));
Television.Draw(lightningShader);
//Mueble TV (1)
view = camera.GetViewMatrix();
model = glm::mat4(1);
tmp = model = glm::translate(model, glm::vec3(-10.48f, 0.0f, -1.37f));
model = glm::scale(model, glm::vec3(1.2f, 1.5f, 2.0f));
glUniformMatrix4fv(modelLoc, 1, GL_FALSE, glm::value_ptr(model));
Mueble1.Draw(lightningShader);
//Mueble (2)
view = camera.GetViewMatrix();
model = glm::translate(model, glm::vec3(movMueble, 0.0f, 0.0f));
glUniformMatrix4fv(modelLoc, 1, GL_FALSE, glm::value_ptr(model));
Mueble2.Draw(lightningShader);
//Mueble (3)
view = camera.GetViewMatrix();
glUniformMatrix4fv(modelLoc, 1, GL_FALSE, glm::value_ptr(model));
Mueble3.Draw(lightningShader);
//Libro
view = camera.GetViewMatrix();
model = glm::mat4(1);
tmp = model = glm::translate(model, glm::vec3(1.5f, -0.2f, 8.0f));
glUniformMatrix4fv(modelLoc, 1, GL_FALSE, glm::value_ptr(model));
Libro.Draw(lightningShader);

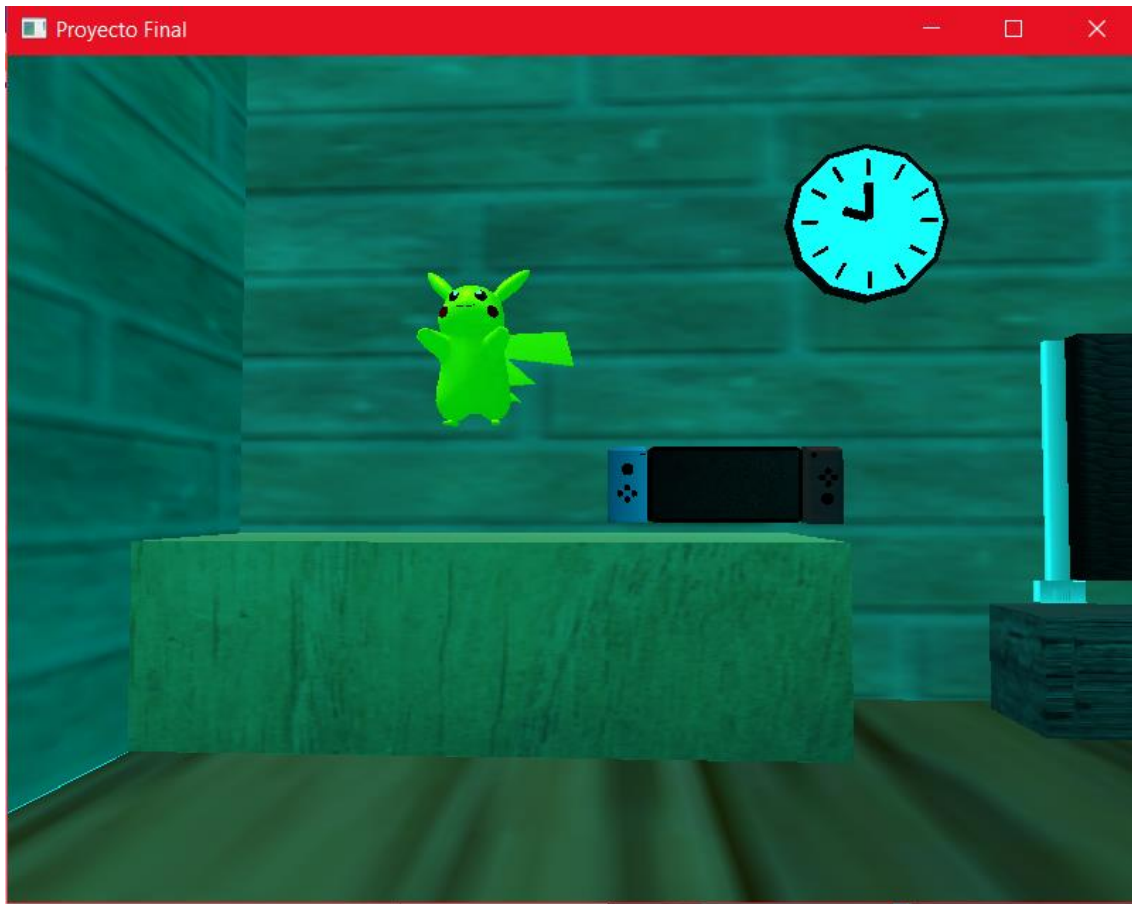
```

Objetos recreados en Maya:

- Una televisión
- Un mueble para apoyar la televisión
- Un reloj



- Un Nintendo Switch
- Un Pikachu de juguete
- Baúl en el que se apoyan los juguetes

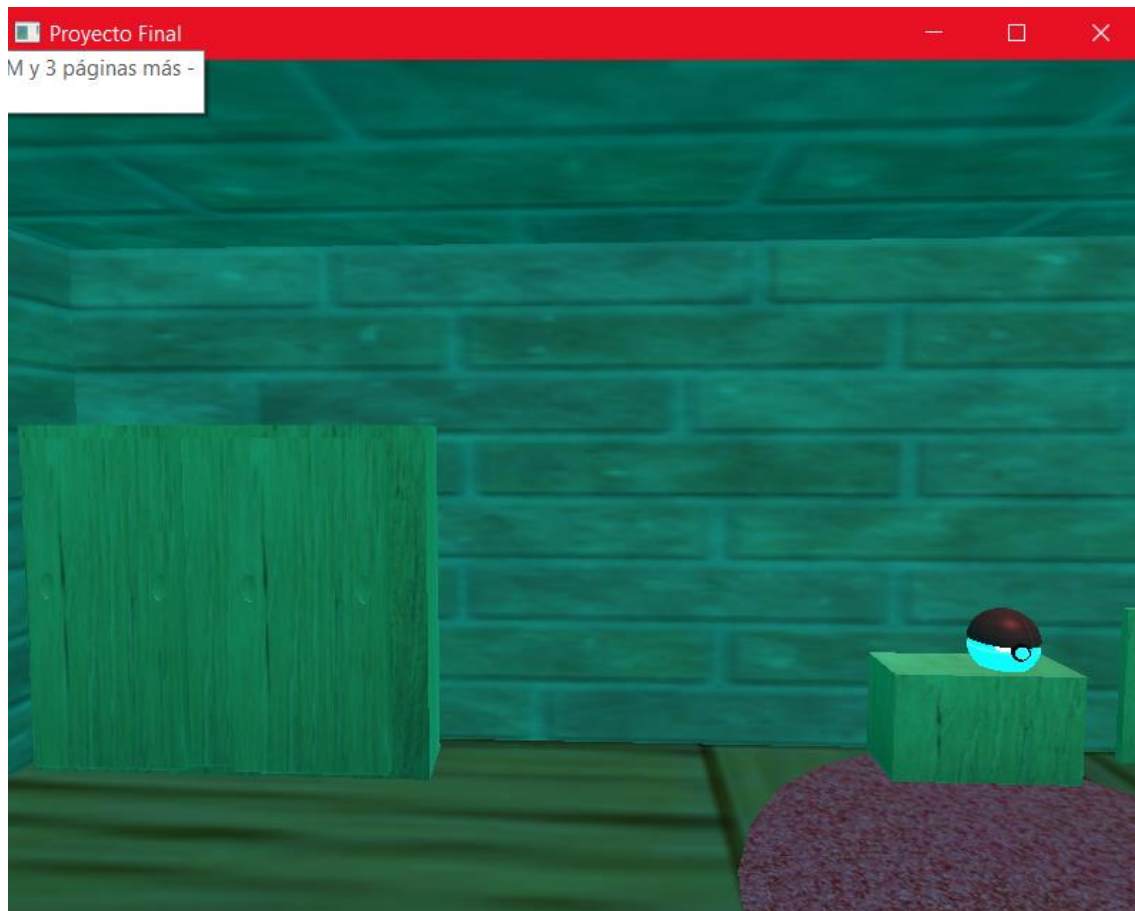


- Una laptop
- Una silla giratoria
- Un libro
- Una mesa

AM y 3 páginas más -



- Un armario
- Una lampara en forma de pokébola
- Un mueble para sostener la lampara
- Una alfombra



- Una cama
- Una almohada



- Estructura exterior de la casa



Objetos recreados en OpenGL:

No se pudo recrear el objeto con primitivas debido a un error al vincular los Shader