



**Universidad Nacional Autónoma de México**  
Facultad de Ingeniería  
División de Ingeniería Eléctrica Electrónica  
**CGeIHC (1590)**



Ing. Carlos Aldair Román Balbuena

Semestre 2022-1

**Proyecto Final.**  
**Manual Técnico**

Grupo: 04

Elaborado por:  
Guzmán Santiago Rubén Vicente

Fecha de entrega:  
09 de diciembre del 2021

## Objetivo.

El alumno aplicara los conocimientos teóricos y prácticos que se han adquirido durante el desarrollo del curso, recreando un espacio virtual elegido por el mismo, así reforzando y aplicando lo aprendido sobre modelado geométrico y jerárquico, iluminación, texturizado, animado y programación en C++.

## Diagrama de Gantt.

Actividad	Inicio	Final	12-oct	19-oct	26-oct	27-oct	12-nov	15-nov	16-nov	20-nov	02-nov	04-nov	06-nov	09-nov	16-nov	17-nov	19-nov	21-nov	22-nov	03-dic	06-dic	07-dic	08-dic	09-dic
Imágenes de referencia	12-oct	16-nov																						
Modelado de los objetos	19-oct	06-dic																						
Modelado de la casa	24-oct	03-dic																						
Animaciones	21-nov	07-dic																						
Documento escrito	22-nov	08-dic																						
Entrega	09-dic	09-dic																						

## Alcance del Proyecto.

En este proyecto se recrearán en 3D un espacio elegido por nosotros, junto con una serie de 7 objetos que también definimos previamente, usando el software de Maya, texturizando, modelando y definiendo los objetos que se animarán posteriormente; se utilizara Visual Studio para la animación y renderizado de los objetos que fueron previamente modelados, al igual se dará iluminación al escenario.

Para la creación y/o edición de las texturas utilizadas en el proyecto utilizaremos GIMP como herramienta.

También se utilizará GitHub para la creación de un repositorio virtual que nos permita almacenar y distribuir el proyecto realizado y así facilitar la entrega final.

## Limitantes.

Dentro de las limitantes que se presentaron en el desarrollo del proyecto se encuentran los problemas de procesamiento, ya que al no tener un procesador con mayor capacidad

en ocasiones había problemas de desempeño, ocasionando ralentización en el equipo; también tuve problemas al modelar algunos objetos al no contar con experiencia previa sobre modelado en 3D, por último, el tiempo que se tardan los archivos en subir al repositorio virtual fue muy extenso.

## Documentación del código.

Variable o función	Descripción
bool anim = false;	Esta variable la usamos para saber cuando prendera la luz de la lampara
bool tele = false;	Esta variable la usamos para saber cuando prendera la televisión
bool animp1 = false;	Esta variable la usamos para saber cuando se movera la puerta frontal
bool animp2 = false;	Esta variable la usamos para saber cuando se movera la puerta trasera
bool animpC = false;	Esta variable la usamos para saber cuando se movera la puerta de la cocina
bool animm = false;	Esta variable la usamos para saber cuando se movera la mesedora
bool animv1 = false;	Esta variable la usamos para saber cuando se movera la ventana frontal 1
bool animv2 = false;	Esta variable la usamos para saber cuando se movera la ventana frontal 2
bool animv3 = false;	Esta variable la usamos para saber cuando se movera la ventana lateral
bool animsF = false;	Esta variable la usamos para saber cuando se movera la silla frontal de la cocina
bool animsT = false;	Esta variable la usamos para saber cuando se movera la silla trasera de la cocina
bool animsL = false;	Esta variable la usamos para saber cuando se movera la silla lateral de la cocina
bool recorrido1 = true;	Esta variable sirve para saber si la silla frontal está llendo hacia enfrente
bool recorrido2 = false;	Esta variable sirve para saber si la silla frontal está llendo hacia atrás
bool recorrido3 = true;	Esta variable sirve para saber si la silla trasera está llendo hacia enfrente
bool recorrido4 = false;	Esta variable sirve para saber si la silla trasera está llendo hacia atrás
bool recorrido5 = true;	Esta variable sirve para saber si la silla lateral está llendo hacia enfrente
bool recorrido6 = false;	Esta variable sirve para saber si la silla lateral está llendo hacia atrás
float puerta = 0.0f;	Esta variable la usamos para saber el ángulo de movimiento de la puerta frontal
float puerta2 = 0.0f;	Esta variable la usamos para saber el ángulo de movimiento de la puerta trasera
float puertaC = 0.0f;	Esta variable la usamos para saber el ángulo de movimiento de la puerta de la cocina
float mecedora = 0.0f;	Esta variable la usamos para saber el ángulo de movimiento de la mesedora
float v1 = 0.0f;	Esta variable la usamos para saber el ángulo de movimiento de la ventana frontal 1
float v2 = 0.0f;	Esta variable la usamos para saber el ángulo de movimiento de la ventana frontal 2
float v3 = 0.0f;	Esta variable la usamos para saber el ángulo de movimiento de la ventana lateral
int bandera = 0;	Esta variable la usamos para saber cuando detener el movimiento de la puerta frontal
int banderap2 = 0;	Esta variable la usamos para saber cuando detener el movimiento de la puerta trasera
int banderapC = 0;	Esta variable la usamos para saber cuando detener el movimiento de la puerta de la cocina
int banderame = 0;	Esta variable la usamos para saber cuando detener el movimiento de la mesedora
int banderav1 = 0;	Esta variable la usamos para saber cuando detener el movimiento de la ventana frontal 1
int banderav2 = 0;	Esta variable la usamos para saber cuando detener el movimiento de la ventana frontal 2
int banderav3 = 0;	Esta variable la usamos para saber cuando detener el movimiento de la ventana lateral
float sillaF = 0.0f;	Esta variable la usamos para saber las unidades de movimiento de la silla frontal
float sillaT = 0.0f;	Esta variable la usamos para saber las unidades de movimiento de la silla trasera
float sillaL = 0.0f;	Esta variable la usamos para saber las unidades de movimiento de la silla lateral
glm::vec3 pointLightPositions[];	Esta variable almacenará las posiciones de los Point Lights utilizados
Model Casa((char*)"Models/Casa/casacompleta.obj");	Esta variable sirve para cargar el modelo de la casa, que incluye los objetos no animados
Model Puerta1((char*)"Models/Casa/puerta1.obj");	Esta variable sirve para cargar el modelo de la puerta frontal
Model Puerta2((char*)"Models/Casa/puerta2.obj");	Esta variable sirve para cargar el modelo de la puerta trasera
Model Mecedora((char*)"Models/Casa/mecedora.obj");	Esta variable sirve para cargar el modelo de la mesedora
Model Ventana1((char*)"Models/Casa/ventana1.obj");	Esta variable sirve para cargar el modelo de la ventana frontal 1
Model Ventana2((char*)"Models/Casa/ventana2.obj");	Esta variable sirve para cargar el modelo de la ventana frontal 2
Model Ventana3((char*)"Models/Casa/ventana3.obj");	Esta variable sirve para cargar el modelo de la ventana trasera
Model SillaF((char*)"Models/Casa/sillaFrontal.obj");	Esta variable sirve para cargar el modelo de la silla frontal
Model SillaT((char*)"Models/Casa/sillaTrasera.obj");	Esta variable sirve para cargar el modelo de la silla trasera
Model SillaL((char*)"Models/Casa/sillaLateral.obj");	Esta variable sirve para cargar el modelo de la silla lateral
Model PuertaCocina((char*)"Models/Casa/PuertaCocina.obj");	Esta variable sirve para cargar el modelo de la puerta de la cocina
glUniform3f(glGetUniformLocation(lightingShader.Program, "dirLight.direction"), -0.2f, -1.0f, -0.3f);	Controla la dirección de la iluminación
glUniform3f(glGetUniformLocation(lightingShader.Program, "dirLight.ambient"), 0.5f, 0.5f, 0.5f);	Controla la iluminación
glUniform3f(glGetUniformLocation(lightingShader.Program, "dirLight.diffuse"), 0.5f, 0.5f, 0.5f);	Aclara el escenario
glUniform3f(glGetUniformLocation(lightingShader.Program, "dirLight.specular"), 0.3f, 0.3f, 0.3f);	Controla la opacidad de los objetos
PointLights	Iluminaran en un punto del escenario dadas coordenadas, color, brillo y opacidad
model = glm::translate(model, glm::vec3(x, y, z));	Se usa para mover algún objeto que haya sido exportado
model = glm::rotate(model, glm::radians(puerta), glm::vec3(x, y, z));	Se usa para rotar algún objeto que haya sido exportado
void DoMovement()	Se utilizará para saber las acciones que se realizarán en el ambiente 3D
void KeyCallback()	Se utilizará para saber las acciones que se realizarán en el ambiente 3D
if (keys[GLFW_KEY_])	Se utilizará para saber que evento ocurrirá al presionar una tecla definida