



Universidad Nacional Autónoma de México  
Facultad de ingeniería



# Manual técnico

Computación grafica e interacción  
humano computadora

Profesor: Jose Roque Roman  
Guadarrama

Semestre 2021-2

Grupo: 08

Alumno: Vázquez Zaragoza Jesús  
Arturo

Fecha de entrega: 11 de agosto del  
2021

Contenido

Objetivo.....3

Objetivo del proyecto.....3

Alcance del proyecto.....3

Limitantes.....4

Manual:.....4

# Objetivo

El presente documento tiene como objetivo dar a conocer la documentación del proyecto realizado para la materia de Computación gráfica e interacción humano computadora, además se presentara los objetivos del proyecto realizado, presentar el diagrama de Gantt seguido para el desarrollo del mismo, cual es el alcance, así como sus limitantes y la documentación pertinente del código utilizado.

## Objetivo del proyecto

El proyecto desarrollado durante el semestre tiene como objetivo poner a prueba la creatividad, los conocimientos adquiridos durante el semestre, así como la capacidad de resolver los problemas que se presenten durante el desarrollo de este, para cumplir con dicho objetivo deberá entregarse un espacio virtual, ya sea real o ficticio, donde se recrearan ciertos objetos lo más parecidos a como son en realidad.

## Alcance del proyecto

Con el desarrollo de este proyecto se busca cubrir todos los rubros o temas vistos en clase tales como: animaciones, manejo de modelos, iluminación, sombreado, etc.

Se busca desarrollar una fachada, así como su ambientación. Para el caso particular se creo la fachada de una casa de dos pisos en la cual se agregaran elementos únicamente en el primero de ellos o la planta baja, tendrá 5 habitaciones donde cada habitación corresponderá a lo siguiente: sala, comedor, cocina, recibidor y recamara.

La sala se busca que tenga sillones, tele, lampara de mesa, algunos muebles decorativos, así como un ventilador con su respectiva luz, este último tendrá que poder prenderse y apagarse.

El comedor deberá tener una mesa y al menos 4 sillas igual, al igual que la sala un ventilador con su luz.

En la cocina habrá una estufa y un refrigerador, una luz en el techo y algún alimento que sirva para simular su “vuelo” al comedor, la luz debe poder prenderse y apagarse.

Para el recibidor no habrá objetos especiales.

En la habitación habrá una luz que se pueda encender y apagar, además de una cama.

Cada una de las habitaciones tendrá que tener un papel tapiz diferente en sus paredes, además algunas de las paredes tendrán ventanas funcionales, es decir que se podrá ver a través de ellas.

Para fuera de la casa habrá un auto y una escoba voladora, la escoba deberá tener una animación que simule el vuelo de la misma.

# Limitantes

Las principales limitantes que me encontré al momento de desarrollar el proyecto es que no pude usar todos los modelos que me gustaría, bien porque no existían o bien porque eran modelos de paga.

Una limitante también fue no saber como desarrollar modelos 3D por cuenta propia, ya que esos modelos que no podía encontrar o usar podría haberlos desarrollado, cuando intente desarrollar algunos de mis modelos me di cuenta que es un proceso demasiado tardado si es que los quería dejar con el nivel de detalle que me gustaría.

Algunos de los modelos que encontraba y quería usar no siempre estaban texturizados, cuando estaban separados en elementos era muy sencillo texturizarlo, sin embargo cuando era un solo elemento era difícil hacer mis texturas, ya que tenía que trabajarlas por separado y hacer parte por parte para que se viera como deseaba.

En mi proyecto no pude usar modelos de personas ya que los modelos que encontré tenían un costo para poder ser descargados.

# Manual

Para el manejo de las luces use lo siguiente:

```
//Bandera que permite activar o desactivar las luces
bool activeLuz1 = false;
bool activeLuz2 = false;
bool activeLuz3 = false;
bool activeLuz4 = false;
```

```
// Posición de los puntos de luz
glm::vec3 pointLightPositions[] = {
    glm::vec3(7.0f, 4.0f, 5.0f),
    glm::vec3(-6.0f, 4.0f, 5.0f),
    glm::vec3(-9.0f, 4.8f, -6.0f),
    glm::vec3(11.5f, 4.8f, -7.0f)
};

//Asigna la luz de cada punto
glm::vec3 LightP1;
glm::vec3 LightP2;
glm::vec3 LightP3;
glm::vec3 LightP4;
```

```
if (keys[GLFW_KEY_1])
{
    activeLuz1 = !activeLuz1;
    if (activeLuz1) {
        LightP1 = glm::vec3(0.984f, 0.992f, 0.427f);
    }
    else {
        LightP1 = glm::vec3(0.0f, 0.0f, 0.0f);
    }
}
```

```
model = glm::mat4(1);
model = glm::translate(model, glm::vec3(11.5f, 4.8f, -7.0f));
glUniformMatrix4fv(modelLoc, 1, GL_FALSE, glm::value_ptr(model));
LamparaTecho.Draw(lightningShader);
```

```
model = glm::mat4(1);
model = glm::translate(model, glm::vec3(-6.0f, 4.0f, 5.0f));
model = glm::rotate(model, rotVent2, glm::vec3(0.0f, 1.0f, 0.0f));
glUniformMatrix4fv(modelLoc, 1, GL_FALSE, glm::value_ptr(model));
Ventilador.Draw(lightningShader);
```

```
//Rotación de los ventiladores
float rotVent1 = 0.0;
float rotVent2 = 0.0;

//Bandera que controla la velocidad del ventilador
bool velocidad1 = false;
bool velocidad2 = false;
```

```
//Sube o baja la velocidad del ventilador 1
if (activeVent1) {
    if (velocidad1)
        rotVent1 += 0.05;
    else
        rotVent1 += 0.1;
}
```

```
//Activa o desactiva los ventiladores
if (keys[GLFW_KEY_5])
    activeVent1 = !activeVent1;

if (keys[GLFW_KEY_6])
    activeVent2 = !activeVent2;
//Sube o baja la velocidad del ventilador
if (keys[GLFW_KEY_L]) {
    velocidad1 = !velocidad1;
}

if (keys[GLFW_KEY_R]) {
    velocidad2 = !velocidad2;
}
```

Como podemos ver cada una de las luces tiene una posición diferente, ya que se ubican en cuartos diferentes, asimismo todas cuentan con una bandera para activar o desactivar su luz. Por último todas las luces emiten el mismo color cuando son activadas.

Para los ventiladores de igual forma se utilizaron banderas para prenderlos y apagarlos, así como para controlar su velocidad, ya sea rápida o lenta.

```
//Activa o desactiva el movimiento de la pizza o la escoba
if (keys[GLFW_KEY_Q])
    circuitoP = !circuitoP;
if (keys[GLFW_KEY_E])
    circuitoC = !circuitoC;
```

```
//Recorrido a realizar por la pizza
if (circuitoP)
{
    if (recorridoP1)
    {
        movPizzaX += 0.01f;
        if (movPizzaX > 5)
        {
            recorridoP1 = false;
            recorridoP2 = true;
        }
    }

    if (recorridoP2)
    {
        movPizzaY += 0.01f;
        if (movPizzaY > 1)
        {
            recorridoP2 = false;
            recorridoP3 = true;
        }
    }
}
```

```
//Recorrido a realizar por la escoba
if (circuitoC)
{
    if (recorridoC1)
    {
        movCZ -= 0.1f;
        movCY = 5 * glm::sin(0.5 * movCZ);
        if (movCZ < -20.0f)
        {
            recorridoC1 = false;
            recorridoC2 = true;
        }
    }
    if (recorridoC2)
    {
        rotC = 90;
        movCX -= 0.1f;
        movCY = 5 * glm::sin(0.5 * movCX);
        if (movCX < -20.0f)
        {
            recorridoC2 = false;
            recorridoC3 = true;
        }
    }
}
```

Para manejar el movimiento de la escoba y la pizza hice uso de las banderas para controlar los movimientos y las direcciones, así como en caso de ser necesario rotar los modelos. También es necesaria una bandera para activar o desactivar la animación.