

Manual Técnico

HOUSE Phineas and Ferb



INDICE

Objetivo del Proyecto	3
Alcance del Proyecto.....	3
Costos del Proyecto	4
LIMITANTES	5
TABLA DE VARIABLES Y DESCRIPCIONES.....	6
Escenarios Recreados	7
.....	7
Metodología del Software Aplicada	8
Conclusión	9

Objetivo del Proyecto

El propósito de este proyecto es poner en práctica los conocimientos adquiridos durante el curso de Computación Gráfica, mediante la creación de un entorno tridimensional inspirado en la casa de la serie Phineas y Ferb, el cual incluye dos habitaciones: una sala y una cocina.

Para lograrlo, se emplearán técnicas de modelado geométrico usando primitivas básicas, a las que se aplicarán texturas mediante el uso de librerías gráficas. El objetivo es generar una representación visual precisa, cuidando los detalles y respetando el diseño del material original.

Asimismo, se implementarán principios de animación y ambientación gráfica para simular un espacio dinámico e interactivo. Se desarrollarán tanto animaciones simples como secuencias más complejas que reflejen condiciones reales, como el paso del día a la noche o el humo saliendo de la chimenea. Estas animaciones responderán al entorno o se ejecutarán automáticamente para dar mayor realismo.

Para enriquecer la experiencia visual, se integrarán modelos de iluminación avanzados que aporten profundidad y definición, además de efectos como transparencias y sombras. Estas características contribuirán a una atmósfera más inmersiva y creíble para el usuario.

En resumen, este proyecto busca reunir técnicas, herramientas y conceptos aprendidos durante el curso para construir un entorno visualmente atractivo, basado en una caricatura, que permita al usuario interactuar con una representación fiel y detallada de la casa.

Alcance del Proyecto

La realización del proyecto se basa en el uso de la API gráfica OpenGL, que permitirá manejar gráficos en tres dimensiones y aplicar efectos visuales realistas. OpenGL se combinará con técnicas avanzadas de renderizado para ofrecer una experiencia fluida y de alta calidad.

El lenguaje utilizado será C++, lo que facilitará un control más eficiente sobre la memoria y los procesos gráficos de bajo nivel. Esto permitirá optimizar el rendimiento del programa, especialmente en la gestión de los recursos visuales.

El desarrollo se llevará a cabo en Visual Studio Community 2022, aprovechando sus herramientas de depuración, optimización y control de versiones para mantener un flujo de trabajo organizado y estable.

Entre los elementos animados que se incluirán desde el arranque del programa destacan el humo de la chimenea y los ciclos de iluminación diurna y nocturna. Estas animaciones no requerirán intervención del usuario, lo que contribuirá a una ambientación continua y realista desde el inicio de la experiencia.

Costos del Proyecto

1. Costos Directos

1.1 Mano de obra especializada

- **Diseño de la fachada**
 - Tiempo estimado: 36 horas (9 semanas, 4 horas por semana).
 - Tarifa por hora: **\$180 MXN.**
 - **Subtotal:** \$6,480 MXN.
- **Modelado de objetos**
 - Total de objetos: 12.
 - Tiempo estimado: 48 horas (4 horas por objeto).
 - Tarifa por hora: **\$200 MXN.**
 - **Subtotal:** \$9,600 MXN.
- **Texturizado y pruebas en OpenGL**
 - Tiempo estimado: 24 horas.
 - Tarifa por hora: **\$190 MXN.**
 - **Subtotal:** \$4,560 MXN.
- **Configuración de iluminación**
 - *Spotlight* (foco de luz dirigido):
 - Requiere conocimientos técnicos específicos.
 - **Subtotal:** \$1,200 MXN.
 - *Point lights* (luces puntuales):
 - **Subtotal:** \$1,000 MXN.
- **Animaciones**
 - Animaciones complejas (2 unidades):
 - Tiempo estimado: 16 horas (8 horas cada una).
 - Tarifa por hora: **\$200 MXN.**
 - **Subtotal:** \$3,200 MXN.
 - Animaciones sencillas (3 unidades):
 - Tiempo estimado: 9 horas (3 horas cada una).
 - Tarifa por hora: **\$180 MXN.**
 - **Subtotal:** \$1,620 MXN.

1.2 Recursos Técnicos y Software

- **Hardware:** Se utilizó una computadora de alto rendimiento durante todo el proceso.
- **Energía eléctrica:** Costo estimado por horas de uso.

2. Costos Indirectos

2.1 Activos

- **Software empleado:**
 - Autodesk 3Ds Max (licencia educativa): Sin costo.
 - OpenGL: Gratuito.

2.2 Pasivos

- **Gastos asociados:**
 - Energía eléctrica.

3. Impuesto al Valor Agregado (IVA)

- El IVA se calcula sobre el subtotal de costos directos:
 - **Subtotal:** \$27,660 MXN.
 - Tasa de IVA: 16%.
 - **IVA:** \$4,425.60 MXN.

4. Costo Total del Proyecto

- **Subtotal de costos directos:** \$27,660 MXN.
- **IVA (16%):** \$4,425.60 MXN.
- **Total general: \$32,085.60 MXN.**

Resumen de Costos

CONCEPTO	SUBTOTAL (MXN)	IVA (16%)	TOTAL (MXN)
DISEÑO DE FACHADA	\$6,480	\$1,036.80	\$7,516.80
MODELADO DE OBJETOS	\$9,600	\$1,536.00	\$11,136.00
TEXTURIZADO Y PRUEBAS OPENGL	\$4,560	\$729.60	\$5,289.60
CONFIGURACIÓN DE ILUMINACIÓN	\$2,200	\$352.00	\$2,552.00
ANIMACIONES	\$4,820	\$771.20	\$5,591.20
TOTALES	\$27,660	\$4,425.60	\$32,085.60

LIMITANTES

Estas limitantes establecen las restricciones necesarias para el correcto funcionamiento y rendimiento del programa:

- **Tamaño Máximo de los Objetos:** Los objetos 3D utilizados en la escena tienen una limitación de tamaño para asegurar un rendimiento adecuado y evitar sobrecargar la memoria gráfica.
- **Texturas Escaladas a Potencias de Dos:** Las texturas deben estar escaladas para garantizar una lectura y procesamiento adecuados en OpenGL, evitando problemas de visualización y asegurando compatibilidad con las especificaciones de la API

TABLA DE VARIABLES Y DESCRIPCIONES

Variable	Tipo / Constante	Descripción
toRadians	const float	Constante para convertir grados a radianes.
solDirZ	float	Dirección del sol en el eje Z (para ciclo día/noche).
solDirY	float	Dirección del sol en el eje Y.
solDirYOffset	float	Desplazamiento Y adicional para el sol.
invierteCiclo	bool	Indica si se está invirtiendo el ciclo día/noche.
esDeDia	bool	Bandera que indica si actualmente es de día.
humoTime	float	Tiempo acumulado para animación del humo.
humoDuration	float	Duración total del ciclo de humo.
humoScale	float	Escala inicial del humo.
humoMaxScale	float	Escala máxima que puede alcanzar el humo.
humoHeight	float	Altura inicial del humo.
humoMaxHeight	float	Altura máxima del humo durante animación.
humoOpacity	float	Nivel de opacidad del humo.
puertaAbierta	bool	Indica si la puerta está abierta.
anguloPuerta	float	Ángulo actual de rotación de la puerta.
velocidadPuerta	const float	Velocidad a la que se abre/cierra la puerta.
animDurationDados	float	Duración de la animación de los dados.
velocidadRotacion	float	Velocidad de rotación de los dados.
alturaMaxima	float	Altura máxima que alcanzan los dados al "saltar".
posDado1	const glm::vec3	Posición fija del dado 1 sobre la mesa.
posDado2	const glm::vec3	Posición fija del dado 2 sobre la mesa.
rotFinalDado1	static glm::vec3	Rotación final del dado 1.
rotFinalDado2	static glm::vec3	Rotación final del dado 2.
animTimeDados	float	Tiempo acumulado de animación de dados.

alturaDados	float	Altura actual de los dados durante la animación.
mainWindow	Window	Ventana principal de renderizado.
meshList	std::vector<Mesh*>	Lista de mallas (meshes) cargadas.
shaderList	std::vector<Shader>	Lista de shaders compilados.
camera	Camera	Cámaras utilizadas en la escena.
pisoTexture	Texture	Textura usada para el piso.
HumoTexture	Texture	Textura usada para el humo.
Casa_Phineas	Model	Modelo 3D de la casa de Phineas.
Puerta	Model	Modelo 3D de la puerta.
Dado1, Dado2	Model	Modelos 3D de los dados.
skybox, skybox2	Skybox	Cajas de cielo (una para día y otra para noche).
Material_brillante	Material	Material brillante (mayor especularidad).
Material_opaco	Material	Material opaco (menor especularidad).
deltaTime	GLfloat	Tiempo entre frames.
lastTime	GLfloat	Tiempo del último frame.
limitFPS	static double	Límite de FPS (60).
mainLight	DirectionalLight	Fuente de luz direccional principal.
pointLights	PointLight[MAX_POINT_LIGHTS]	Arreglo de luces puntuales.
spotLights	SpotLight[MAX_SPOT_LIGHTS]	Arreglo de luces tipo spotlight.
vShader	const char*	Ruta al vertex shader.
fShader	const char*	Ruta al fragment shader.

Escenarios Recreados



Casa en OpenGL



Sala



Cocina

Metodología del Software Aplicada

Para el desarrollo del proyecto se implementó la metodología ágil Scrum, una de las más utilizadas actualmente en la ingeniería de software debido a su enfoque iterativo e incremental. Esta metodología permitió organizar el trabajo en sprints, es decir, ciclos de desarrollo cortos y definidos, en los que se plantearon objetivos específicos y medibles. Al finalizar cada sprint, se realizó una revisión del progreso y se ajustaron tanto las tareas como las prioridades según los avances obtenidos y los retos encontrados. Esta dinámica facilitó una mejora continua del diseño, la programación y la implementación de la red simulada, promoviendo además una comunicación constante dentro del equipo. Gracias a Scrum, fue posible detectar errores de forma temprana,

realizar ajustes de manera flexible y garantizar una entrega funcional y alineada con los requerimientos del proyecto.

Conclusión

Este proyecto representó una valiosa oportunidad para reforzar y aplicar los conocimientos adquiridos en Computación Gráfica. A lo largo del desarrollo, no solo puse en práctica las herramientas vistas durante el curso, sino que también integré saberes previos como la planificación estructurada y el uso de diagramas para organizar el proceso.

Enfrentar el reto de construir un entorno 3D realista me impulsó a investigar por cuenta propia y a incorporar técnicas adicionales, como la creación de variables personalizadas para manejar animaciones complejas. Gracias a ello, logré representar efectos visuales como el fuego y el humo, cuidando la precisión en sus movimientos para generar una apariencia convincente.

Cada etapa —desde el modelado geométrico hasta la implementación de animaciones y el diseño de la iluminación— aportó al desarrollo de un espacio atractivo y funcional. Además, el uso de la metodología Scrum facilitó la adaptación a cambios y mejoras continuas, permitiendo perfeccionar detalles en cada sprint e integrar de forma armoniosa los distintos componentes visuales.

En resumen, este proyecto no solo fortaleció mis habilidades técnicas, sino que también me preparó para abordar futuros desarrollos donde la interactividad, el realismo y el rendimiento gráfico sean elementos clave.