Documentación

Programación Multimedia

Programa: Videojuego 3D

Autor: Cristina Fuster García

Índice de contenidos

# **1. Documentación de usuario final:**

## **1.1. Guía de usuario**

Este proyecto es una visualización 3d que utiliza la librería Three.js para renderizar objetos en una escena 3D.La camara la hemos modificado para que tenga la forma esférica, dentro de ella también va a encontrar un suelo con textura, una cara de mono en formato GLB, y también podemos observar el cubo y la esfera a parte de poder moverse tanto en recto como en diagonal con la camara.

- Navegador compatible con WebGl (chrome, Firefox, Safari)

- Sistema Operativo: Windows, macOS, Linux

Interfaz de usuario

Se muestra en un lienzo en pantalla completa, los objetos y las luces se renderizan dentro del lienzo.

## **1.2. FAQ (Preguntas frecuentes)**

* ¿Cómo se puede cambiar el color de los objetos?  
     
  Está definido en el código, se puede modificar el valor hexadecimal en las líneas donde definimos el material de los objetos.
* ¿Podemos agregar más objetos?  
     
  Se pueden crear más objetos, utilizando las clases proporcionadas por Three.js como THREE.BoxGeometry o THREE.SphereGeometry..
* ¿Que es Three.js y cómo se usa?  
     
  Es una librería de JS para renderizar gráficos 3d en el navegador usando WebGl.
* ¿Cómo funciona la librería Three.js?  
     
  Generalmente sigue el flujo de crear una escena(Scene), creamos una camara que determinara la perspectiva desde donde se vera la escena.

El bucle de animación se encarga de actualizar las propiedaes de los objetos y redibujar la escena constantemenet.

## 

## **1.3. Solución de problemas comunes**

1. El proyecto no carga o muestra una pantalla en blanco.

◦ Asegúrate de que tu navegador soporte WebGL. Si no es compatible, intenta actualizar tu navegador prueba en uno diferente.

◦ Revisa la consola de desarrollo del navegador (F12) para ver si hay errores de carga de recursos.

2. Los objetos no se mueven al presionar las teclas de flecha.

◦ Verifica que el navegador esté permitiendo la interacción del teclado y que el script JavaScript esté correctamente enlazado en el archivo HTML.

3. El proyecto corre de manera lenta o no responde bien.

◦ Asegúrate de que tu dispositivo tenga una tarjeta gráfica compatible con WebGL y que los controladores estén actualizados.

# **2. Documentación técnica para desarrolladores**

## **2.1. Introducción al software**

Objetivos Principales

Tiene como objetivo demostrar los conceptos básicos de three.js para crear una escena 3D interactiva donde los usuarios pueden mover los objetos utilizando las teclas.

Tecnologías Utilizadas

Three.js: librería para renderizar 3D en el navegador utilizando WebGL.

HTML5: para la estructura principal y la integración de los elementos.

CSS: para le diseño básico de la págin web (pantalla completa)

JavaScript: para la lógica de la escena,objetos y la iteración.

## **2.2. Arquitectura del sistema**

El sistema consta de varios componentes clave para crear y manipular una escena 3D en tiempo real.

Componentes Principales

1. Escena: THREE.Scene() contiene todos los objetos, cámaras y luces.

2. Cámara: THREE.PerspectiveCamera() establece la perspectiva de visualización de la escena.

Para crear camara esferica utilizamos THREE.SphereGeometry(30, 32, 16)

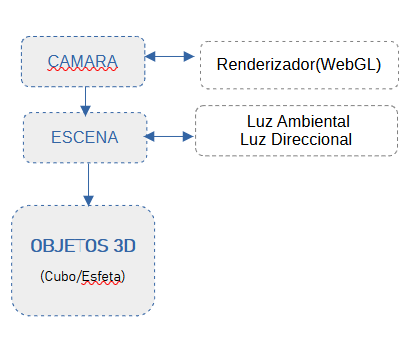
3. Renderizador: THREE.WebGLRenderer() se encarga de renderizar la escena en el navegador.

4. Objetos 3D: Se utilizan geometrías y materiales para crear objetos visuales como cubos y esferas.

Para cargar las imagenes glb usaremos GTFLoader

5. Luces: THREE.PointLight() y THREE.AmbientLight() se usan para iluminar los objetos en la escena.

Diagrama de Arquitectura



## **2.3. Estructura del código**

El archivo principal contiene las importaciones de la biblioteca.

JS: en el script se definen las escenas, los objetos y la animación de movimiento.

Relación entre los Componentes

El renderizador se encarga de visualizar la escena completa que contiene la cámara, luces y objetos.

Los objetos están conectados en la escena y la animación mueve la esfera

## **2.5. Dependencias y configuraciones**

Dependencias:

• Three.js: Se incluye la librería desde un CDN (https://cdnjs.cloudflare.com/ajax/libs/three.js/r128/three.min.js).

Configuraciones:

• Tamaño de la ventana: El tamaño del renderizador se ajusta dinámicamente a las dimensiones de la ventana del navegador.

• Movimiento de la esfera: Las coordenadas de la esfera se actualizan a través de la interacción del teclado.

# **3. Guías de instalación y despliegue**

## **3.1. Requisitos previos**

Software:

• Navegador web moderno con soporte para WebGL (por ejemplo, Google Chrome, Firefox, Safari).

Hardware:

• Una tarjeta gráfica con soporte para WebGL (la mayoría de las tarjetas gráficas modernas son compatibles).

## **3.2. Instrucciones de instalación**

1. Descarga o clona el repositorio.

2. Abre el archivo index.html en un navegador web.

3. Verás una escena 3D con un cubo y una esfera. Usa las teclas de dirección para mover la esfera

# **4. Mantenimiento y Soporte**

## **4.1. Manejo de errores**

El navegador no soporta WebGL, asegúrate de que tu navegador esté actualizado a la última versión.

No se muestra nada en la pantalla, hay que verificar la consola de errores del navegador para detectar problemas con la cargar Three.js o errores de Javascript.

## **4.2. Registros de Cambios**

Versión 1: Conocimos como importar la librería y las carpetas correspondientes.

Versión 2: Conocimos como poner el material y los objetos.

Versión 3: estuvimos trabajando con la iluminación.

Versión 4: implementar el movimiento, también implementemos el movimiento lateral.

Versión 5: Implementos de las imágenes en formato glb.

## **4.2. Registros de Cambios**

Para soporte técnico o preguntas adicionales, puedes contactar a los desarrolladores en:

• Correo electrónico: soporte@threejsdemo.com

• Página de soporte: www.threejsdemo.com