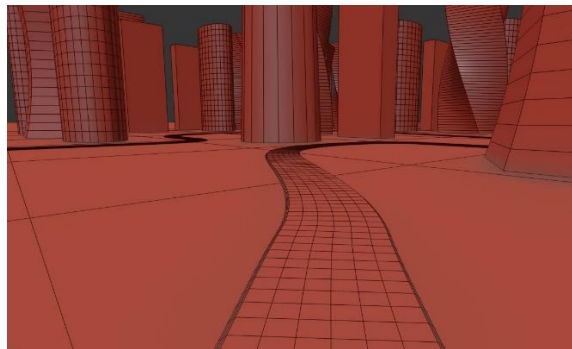
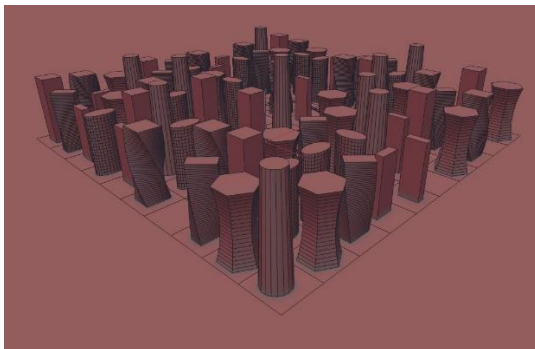
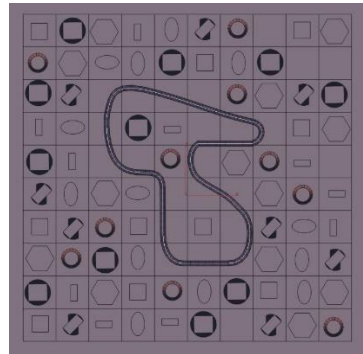
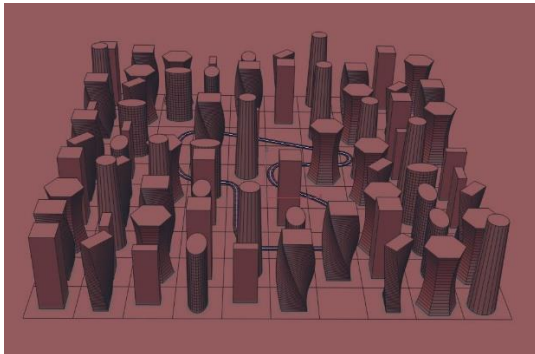


Trabajo Práctico - Sistemas Gráficos

1er. cuatrimestre 2025 – versión 1

Objetivo

Desarrollar una aplicación 3D interactiva utilizando Three.js que represente una escena urbana. La aplicación deberá implementar algoritmos de generación procedural de geometría (superficies de barrido), manejo de curvas, animación basada en física (básica), múltiples cámaras, iluminación y controles interactivos.



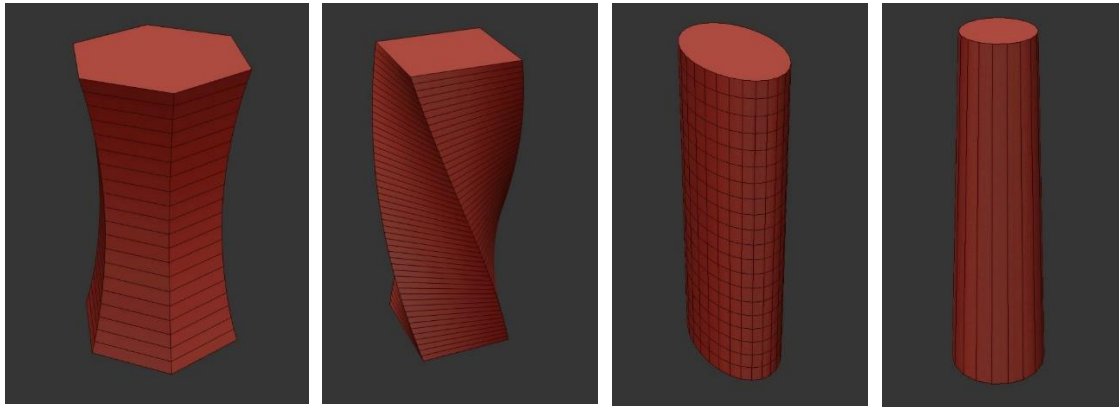
Descripción general de la escena

La escena consistirá en un plano base sobre el cual se generará una ciudad moderna. La ciudad estará organizada en una grilla de manzanas, cada una potencialmente ocupada por un edificio o torre. Una ruta definida por una curva cerrada atravesará la ciudad, y sobre esta ruta circulará un vehículo. La ruta también contendrá elementos como postes de iluminación a los lados, arcos o carteles que cruzan de lado a lado, túneles y rampas.

Entorno urbano y grilla

La ciudad debe organizarse conceptualmente en una grilla de al menos 10x10 manzanas. Se debe implementar un algoritmo que distribuya edificios en las manzanas de la grilla, saltando aquellas que estén total o parcialmente sobre la ruta. El diseño puede ser mas complejo si se desea (por ejemplo, pueden haber más de 1 edificio por manzana).

Edificios modernos (superficies de barrido)

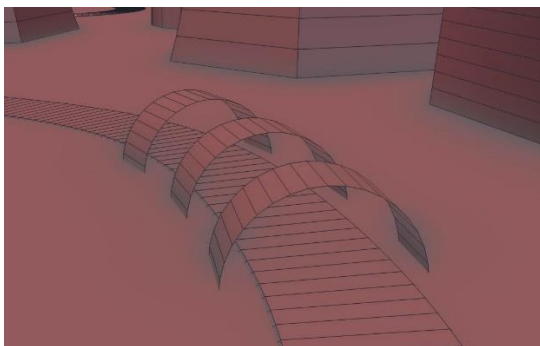


Cada edificio/torre se generará utilizando el algoritmo de **superficie de barrido**, pudiéndose utilizar la clase ParametricGeometry. El recorrido del barrido será una línea recta vertical. El perfil base (forma 2D) deberá ser seleccionado de al menos 3 variantes predefinidas (ej. rectángulo, triángulo, óvalo, u otra forma definida por una curva cerrada simple). Además, durante el barrido a lo largo del recorrido vertical, se deben poder aplicar variaciones paramétricas como:

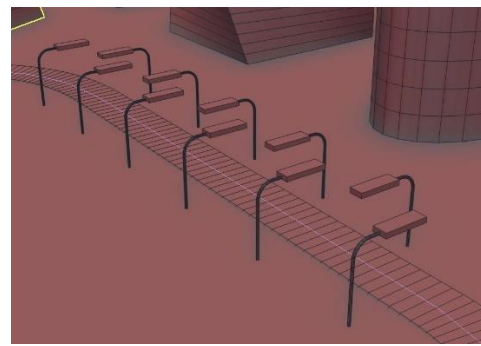
- Torsión: La forma puede girar sobre su eje a medida que se eleva.
- Escalado: La forma puede cambiar de tamaño a lo largo del recorrido.

Deben definirse al menos 3 formas diferentes de variación del ángulo de torsión y del factor de escalado. Los diferentes tipos de torres surgirán de combinar la forma, el perfil de torsión, de escalado, y una altura final variable. Las manzanas deben completarse con una distribución aleatoria de estas variantes para dar diversidad a la ciudad.

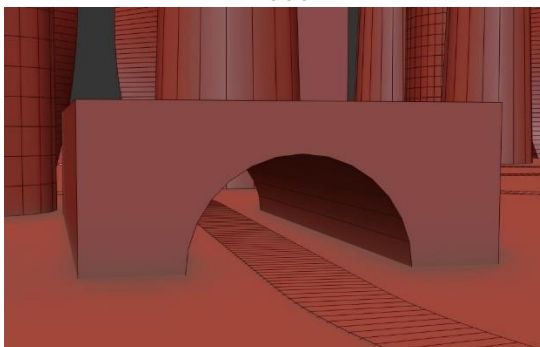
Ruta y elementos asociados



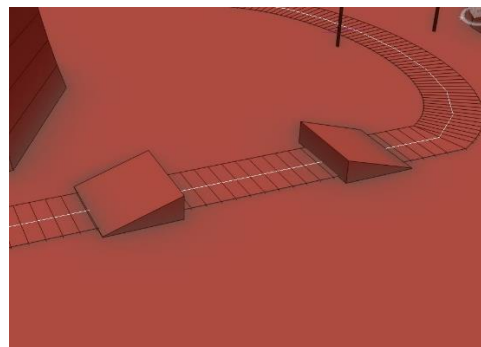
Arcos



Luminarias



Túnel



Rampas

La forma de la ruta debe definirse mediante una curva de Bezier o CatmullRom, cuyos puntos de control estén definidos claramente en el código. La ruta en sí misma se representará como una malla 3D delgada (apta para texturizar como pavimento) que siga la trayectoria de la curva. Utilizando la parametrización de la curva de la ruta (distancia a lo largo de la curva), se deben colocar elementos a intervalos regulares o en puntos específicos:

- Luminarias: Modelos simples (ej. cilindros, combinaciones de primitivas).
- Arcos: Estructuras que crucen la ruta por encima, potencialmente con espacio para carteles.
- Rampas: ubicados sobre la superficie de la ruta para interactuar con el vehículo.
- Túneles: perfil de semicírculo con espesor extruido.

Vehículo

Se debe integrar un modelo de vehículo de 4 ruedas en la escena que recorra la ruta. El movimiento del vehículo será controlado utilizando un motor de física. Se proveerá un ejemplo de código con indicaciones sobre cómo integrarlo. El control del vehículo por parte del usuario se realizará mediante las teclas de flecha del teclado. El modelo 3D puede ser cargado de un archivo o generarse por código. El diseño es de libre elección. Se pueden encontrar modelos gratuitos en estos sitios web:

- sketchfab.com
- cgtrader.com
- turbosquid.com
- free3d.com
- 3dexport.com

Física

El vehículo será el único objeto dinámico de la escena. Las rampas serán obstáculos estáticos contra los cuales podrá colisionar. Además, opcionalmente, se pueden agregar cuerpos rígidos estáticos (colliders) para evitar que el vehículo atravesase otros objetos de la escena.

Iluminación

La escena debe tener 2 modos de iluminación diurna y nocturna. Los detalles de este punto se definirán en la versión 2 de este enunciado.

Cámaras e Interacción

Se deben implementar al menos 4 tipos de cámaras:

- **Cámara Interior Vehículo**: cista desde la cabina del conductor.
- **Cámara Seguimiento Vehículo**: sigue al vehículo desde atrás y/o arriba.
- **Cámara Orbital General**: Permite al usuario orbitar alrededor del centro de la escena y hacer zoom para inspeccionar la ciudad completa usando el mouse.

- **Cámara Primera Persona (Peatón):** Permite al usuario "caminar" por la escena a nivel del suelo. Se controla con teclas (ej. WASD para movimiento, mouse para orientación). Estas teclas no deben interferir con las flechas de control del vehículo.

El usuario debe poder cambiar entre las cámaras activas usando las teclas 1 a 4.

Interfaz de usuario

Se debe incluir una interfaz de usuario simple, que debe contener:

- control para cambiar entre el modo de iluminación día y noche
- control para reiniciar simulación física

Materiales y Texturizado

A definir en versión 2 del enunciado

Ejemplo de física del vehículo

Para poder utilizarlo se requiere utilizar la versión 176.0 de three.js o superior.

Fecha de entrega

27 de junio de 2025