



UASLP
Universidad Autónoma
de San Luis Potosí



**FACULTAD DE
INGENIERÍA**
Área de Ciencias
de la Computación

2834 – MOTORES GRÁFICOS

Clave de la materia: 2834
Clave Facultad: 2834
Clave U.A.S.L.P.: ----
Nivel del Plan de Estudios: I.S.I.: 7
Horas/Clase/Semana: 4
Horas/Práctica (y/o Laboratorio): 0
Prácticas complementarias: 0
Trabajo extra-clase Horas/Semana: 4
Carrera/Tipo de materia: I.S.I./Electiva de Área de Énfasis
No. de créditos aprobados: ----
Fecha última de Revisión Curricular: 28-junio-2024
Materia y clave de la materia requisito: 2832 – Programación de Videojuegos
Clave CACEI: IA
No. de créditos: 8
Horas totales/Semestre: 64
Modalidad: Presencial y No presencia – en línea

OBJETIVO DEL CURSO

Conocer la arquitectura básica de un motor gráfico para el desarrollo de videojuegos; además de implementar el proceso general para diseñar y programar un videojuego

con un motor gráfico, tomando en consideración temas avanzados que el alumno debe de conocer para realizar dicho videojuego.

CONTENIDO TEMÁTICO

1. GENERALIDADES DE LOS MOTORES GRÁFICOS

Tiempo Estimado: 6 hrs.

Objetivo: Analizar las generalidades de los motores gráficos, usos, aplicaciones y los motores gráficos actuales; así como el uso básico de un motor gráfico específico.

- 1.1. Generalidades
- 1.2. Motores gráficos actuales
- 1.3. Usos y aplicaciones
- 1.4. Descripción de módulos básicos
- 1.5. Diferencias entre tipos de motores gráficos

2. ARQUITECTURA DE LOS MOTORES GRÁFICOS

Tiempo Estimado: 16 hrs.

Objetivo: Conocer y analizar los diferentes módulos básicos que conforman un motor gráfico, así como las diferencias principales de los motores gráficos, la creación de terrenos e implementación de cinemáticas.

- 2.1. Instalación de motor gráfico
- 2.2. Elementos básicos del editor
- 2.3. Creación de entornos
- 2.4. Terrenos
- 2.5. Cinemáticas

3. CONSIDERACIONES DE PROGRAMACIÓN

Tiempo estimado: 6 hrs.

Objetivo: Conocer y aplicar diferentes técnicas de programación que son útiles en la programación de videojuegos; así como los lenguajes y técnicas de programación que usan los motores gráficos.

- 3.1. Técnicas de programación de videojuegos
- 3.2. Lenguajes usados en los motores gráficos
- 3.3. Lógica de juego

4. APLICACIÓN DE UN MOTOR GRÁFICO

Tiempo estimado: 36 hrs.

Objetivo: Ser capaz de utilizar un motor gráfico de forma básica y avanzada, así como de analizar los temas necesarios para la construcción de un videojuego de nivel avanzado.

- 4.1. Personajes y animación
- 4.2. Físicas
- 4.3. Inteligencia artificial
- 4.4. Creación de UI
- 4.5. Optimización
- 4.6. Creación de ejecutable

METODOLOGÍA

Desarrollo del tema por parte del profesor siguiendo el método de aprendizaje basado en problemas. Se espera que el alumno investigue ciertos temas, siguiendo el método de aula invertida. Implementar estrategias de

trabajo en equipo cuando sea conveniente (aprendizaje colaborativo). El alumno deberá realizar un proyecto final que realizará durante el transcurso de la materia. El proyecto se evalúa en cada uno de los tres parciales.

EVALUACIÓN

Se realizarán tres exámenes parciales de forma colegiada en las fechas establecidas por la Facultad, de acuerdo con el Reglamento de Exámenes. La calificación de los exámenes parciales estará compuesta por la

evaluación del examen parcial y por otras actividades, como: tareas, investigaciones, resolución de problemas, ejercicios, etc. La calificación del examen ordinario es el promedio de los tres parciales.

BIBLIOGRAFÍA

Bibliografía Básica

Gregory, J. *Game engine architecture*. Boca Raton, FL: CRC Press, Taylor & Francis Group, 2019.

Francesco, G. D. *Game engine architecture*. Packt Publishing Limited, 2016.

Bibliografía Complementaria

Eberly, D. H. *3D game engine architecture: engineering real-time applications with Wild Magic*. Amsterdam: Elsevier, 2005.

Lengyel, E. *Foundations of game engine development: Volume 2: Rendering*. Lincoln, CA: Terathon Software LLC, 2019.

Moller Tomas, Haines, E., & Hoffman, N. *Real-time rendering*. Boca Raton: CRC Press, 2019.

Unreal Engine: <https://www.unrealengine.com/>

Unity: <https://unity.com/>

Godot: <https://godotengine.org/>