



I. INFORMACIÓN GENERAL

CURSO	:	Fundamentos De La Teoría De Juegos
CÓDIGO	:	CC92
CICLO	:	201602
CUERPO ACADÉMICO	:	Shiguihara Juarez, Pedro Nelson
CRÉDITOS	:	3
SEMANAS	:	16
HORAS	:	2 H (Práctica) Semanal /2 H (Teoría) Semanal
ÁREA O CARRERA	:	Ciencias de la Computacion

II. MISIÓN Y VISIÓN DE LA UPC

Misión: Formar líderes íntegros e innovadores con visión global para que transformen el Perú.

Visión: Ser líder en la educación superior por su excelencia académica y su capacidad de innovación.

III. INTRODUCCIÓN

Curso electivo que se dicta en la carrera de Ciencias de la Computación, de carácter teórico-práctico, dirigido a estudiantes del séptimo ciclo, que busca desarrollar la competencia general de pensamiento crítico y la competencia específica de diseñar sistemas, componentes o procesos para encontrar soluciones en la atención de necesidades teniendo en cuenta restricciones económicas, sociales, políticas, éticas, de salud y seguridad y otras propias del entorno empresarial acorde al ABET- Student Outcome (c).

Este curso el estudiante implementa un subsistema en el marco de la arquitectura de motores de videojuegos. Al comprender la arquitectura de motores de videojuegos, los estudiantes reconocen la importancia de implementar aspectos de un videojuegos tales como renderizado, input, animaciones, audio, entre otros. Y al reconocer la importancia de su implementación, podrán buscar alcanzar mejores resultados en el desarrollo de juegos en comparación con los juegos desarrollados a partir de motores previamente existentes.

IV. LOGRO (S) DEL CURSO

Al finalizar el curso, el estudiante puede implementar un subsistema de un motor de videojuego a partir de una arquitectura establecida, acompañada de su documentación respectiva, con el objetivo de desarrollar sistemas de videojuegos con características particulares.

V. UNIDADES DE APRENDIZAJE

UNIDAD N°: 1 Arquitectura de un Motor de Videojuegos

LOGRO

Al finalizar la unidad, el estudiante comprende la arquitectura base de un motor de videojuegos en un entorno de Windows.

TEMARIO

Win32 API. Ventanas y Procesamiento de mensajes. Bucle de juego. Metodos de actualización. Capa visual, capa de bajo nivel y capa de datos. Partes generales de un motor de videojuego.

HORA(S) / SEMANA(S)

8 hrs / Semanas 1 y 2

UNIDAD N°: 2 Sistemas de Soporte y de Recursos**LOGRO**

Al finalizar la unidad, el estudiante comprende los fundamentos de sistemas de soporte y de recursos de un motor de videojuegos.

TEMARIO

Manipulación de Bits. Archivos Binarios. Multi-threading. Sistemas data-driven. Sistemas de soporte: Manejador de Memoria. Sistemas de Recursos: Empaquetado de Archivos y Manejador de Recursos.

HORA(S) / SEMANA(S)

12 hrs / Semanas 3, 4 y 5

UNIDAD N°: 3 Sistemas de Input y Debug/Testing**LOGRO**

Al finalizar la unidad, el estudiante crea un sistema de eventos para componer sistemas de input y debugging de un motor de videojuegos.

TEMARIO

Patrón de Diseño Singleton. Manejador de eventos. Manejador de inputs. Sistema de debug y testing.

HORA(S) / SEMANA(S)

8 hrs / Semanas 6 y 7

UNIDAD N°: 4 Renderizado en Tiempo real**LOGRO**

Al finalizar la unidad, el estudiante comprende los fundamentos de computación gráfica aplicada a videojuegos.

TEMARIO

Rasterización. Rendering Pipeline en DirectX. Shaders en HLSL.
Archivos FBX.

HORA(S) / SEMANA(S)

12 hrs / Semanas 9, 10, 11

UNIDAD N°: 5 Arquitectura Actor-Componente**LOGRO**

Al finalizar la unidad, el estudiante comprende las características de la arquitectura orientada a componentes y sus beneficios en un motor de videojuegos.

TEMARIO

Sistema manejador de Entidades. Componentes. Transforms. Aplanamiento de jerarquías.

HORA(S) / SEMANA(S)

4 hrs / Semana 12

UNIDAD N°: 6 Arquitectura Actor-Componente**LOGRO**

Al finalizar la unidad, el estudiante comprende los fundamentos de un Sistema de Renderizado Frontal (Forward Rendering) utilizando DirectX en un arquitectura Actor-Componente. Además el estudiante comprende los fundamentos de un sistema de animación 3D.

TEMARIO

Técnicas de Renderizado. Forward Rendering. Componente de Renderizado. Sistema de iluminación. Sistema de Partículas. Animación por Vértices. Animación Piel-Huesos. Máquinas de Estados Finitos.

HORA(S) / SEMANA(S)

12 hrs / Semanas 13, 14, 15

VI. METODOLOGÍA

El curso se dicta en 2 sesiones semanales teórico-prácticas, una de 2 horas de laboratorio y otra de 2 horas de teoría.

Durante las sesiones teóricas se utilizarán representaciones gráficas y ejemplos prácticos de los sub-sistemas estudiados. Se promueve la participación de los estudiantes en la construcción de la clase con la resolución de ejercicios en la pizarra.

Durante las sesiones de laboratorio se pasará a implementar los sub-sistemas vistos durante la clase teórica. Además se busca que los alumnos diseñen y realicen sus propias implementaciones a través de su propio pensamiento crítico.

Al final de las semanas 3, 5 y 7 los alumnos deben hacer entrega de sus tareas académicas, las cuales deben demostrar lo aprendido en las sesiones anteriores.

A partir de la semana 10, los alumnos podrán empezar a trabajar en su trabajo final. El trabajo final se dividirá en 3 entregables, cada uno posee una parte de documentación y otra de implementación. El primer entregable es en la segunda sesión de la semana 11, el segundo es en la primera sesión de la semana 13 y el último en la última sesión del curso en la semana 15. Durante el desarrollo del trabajo el profesor realizará un seguimiento para que los alumnos realicen con éxito el proyecto. Se usará el lenguaje C++ para el desarrollo de las tareas prácticas.

VII. EVALUACIÓN

FÓRMULA

10% (TA1) + 10% (TA2) + 10% (TA3) + 15% (EA1) + 30% (TF1) + 5% (PA1) + 20% (EB1)

TIPO DE NOTA	PESO %
TA - TAREAS ACADÉMICAS	10
TA - TAREAS ACADÉMICAS	10
TA - TAREAS ACADÉMICAS	10
EA - EVALUACIÓN PARCIAL	15
TF - TRABAJO FINAL	30
PA - PARTICIPACIÓN	5
EB - EVALUACIÓN FINAL	20

VIII. CRONOGRAMA

TIPO DE PRUEBA	DESCRIPCIÓN NOTA	NÚM. DE PRUEBA	FECHA	OBSERVACIÓN	RECUPERABLE
TA	TAREAS ACADÉMICAS	1	Semana 3		NO
TA	TAREAS ACADÉMICAS	2	Semana 5		NO
TA	TAREAS ACADÉMICAS	3	Semana 7		NO
EA	EVALUACIÓN PARCIAL	1	Semana 8		SÍ
TF	TRABAJO FINAL	1	Semana 15		NO
PA	PARTICIPACIÓN	1	Semana 15		NO
EB	EVALUACIÓN FINAL	1	Semana 16		SÍ

IX. BIBLIOGRAFÍA DEL CURSO

BÁSICA

BLACKMAN, Sue (2013) Beginning 3D game development with Unity 4 : all-in-one, multi-platform game development. New York : Apress.

(006.696 BLAC)

CHANDLER, Heather Maxwell (2014) The game production handbook. Burlington, Massachusetts : Jones & Bartlett Learning.

(794.81526 CHAN)

CHANDLER, Heather Maxwell Chandler, Rafael (2011) Fundamentals of game development. Sudbury, Massachusetts : Jones and Bartlett Learning.

(794.8 CHAN)

RECOMENDADA

(No necesariamente disponible en el Centro de Información)

GREGORY, J. (2009) Game Engine Architecture, pp. .. En: Wellesley, No. 1.

NYSTROM, Robert (2014) Game Programming Patterns, pp. .. En: Genever Benning, No. 1.