



Centro adscrito



GRADO EN DISEÑO Y DESARROLLO DE VIDEOJUEGOS

PLANIFICACIÓN DE LA DOCENCIA UNIVERSITARIA

GUÍA DOCENTE

PROGRAMACIÓN ORIENTADA A OBJETOS

CURSO ACADÉMICO 2017 – 2018

1. DATOS DE IDENTIFICACIÓN DE LA ASIGNATURA

Título:	Grado en Diseño y Desarrollo de Videojuegos
Módulo:	Módulo Ciencias Aplicadas y Tecnología
Denominación de la asignatura:	Programación Orientada a Objetos
Código:	40013
Curso:	Segundo
Semestre:	Primero
Tipo de asignatura (básica, obligatoria u optativa):	Obligatoria
Créditos ECTS:	5
Modalidad/es de enseñanza:	Presencial
Profesor:	Pablo Nogueira Iglesias, Elena Gómez Martínez
Lengua vehicular:	Español
Página web: www.esne.es	

2. PROFESORADO DE LA ASIGNATURA

Profesor:

Pablo Nogueira Iglesias

Datos de contacto:

pablo.nogueira@esne.es

TUTORÍAS ACADÉMICAS: consultar en el Campus Virtual el documento "horarios de tutorías - Grado en Diseño y Desarrollo de Videojuegos"

Profesor:

Sandra Garrido

Datos de contacto:

sandra.garrido@esne.es

TUTORÍAS ACADÉMICAS: consultar en el Campus Virtual el documento "horarios de tutorías - Grado en Diseño y Desarrollo de Videojuegos"

3. REQUISITOS PREVIOS.

Esenciales:

Conocimientos básicos de programación.

Aconsejables:

Los propios del título.

4. SENTIDO Y APORTACIONES DE LA ASIGNATURA AL PLAN DE ESTUDIOS

Campo de conocimiento al pertenece la asignatura.

Esta asignatura pertenece a la rama de Ingeniería y Arquitectura y es una materia de Informática.

Relación de interdisciplinariedad con otras asignaturas del currículum.

La asignatura de Programación Orientada a Objetos hará que los alumnos aprendan y trabajen con el paradigma de la programación orientada a objetos. Ya que la asignatura cuenta con un fuerte componente práctico y que la programación será un punto fundamental en el desarrollo de los proyectos, esta asignatura se relaciona de manera directa con las asignaturas de primero: Fundamentos de la Programación Estructurada, Programación II y será de gran utilidad para el alumno en las asignaturas de tercero de Desarrollo para Dispositivos Móviles e Ingeniería del Conocimiento: IA.

El hecho de usar C# como lenguaje de programación, hace que esta asignatura tenga como natural continuación la segunda parte de la asignatura de Animaciones y Scripting, donde se usará la herramienta Unity, con la que se desarrollará en el mismo lenguaje.

Aportaciones al plan de estudios e interés profesional de la asignatura.

Gracias a esta asignatura el alumno será capaz de plantear eficazmente el desarrollo de un proyecto en equipo, anticipando las distintas fases del mismo y pudiendo utilizar metodologías, normas y estándares de la industria.

5. RESULTADOS DE APRENDIZAJE EN RELACIÓN CON LAS COMPETENCIAS QUE DESARROLLA LA MATERIA

COMPETENCIAS GENERALES

CG7. Conocerá metodologías, normas y estándares, técnicas y programas de uso específico, y será capaz de utilizarlos en el desarrollo de videojuegos.

COMPETENCIAS ESPECÍFICAS

CE7. Entenderá el proceso de diseño y desarrollo de un videojuego como un proyecto multidisciplinar, y será capaz de plantear dicho proyecto en términos de trabajo en equipo. Asimismo, será capaz de idear proyectos de videojuegos, plasmando dichas ideas gráficamente y por escrito de forma estructurada, ordenada y comprensible.

CE8. Será capaz de prototipar un sistema a partir de un diseño.

CE13. Comprenderá en qué momento es adecuado utilizar un paradigma u otro de programación y sabrá aplicarlo correctamente mediante los elementos de desarrollo que manejará (frameworks, kits de desarrollo, etc.).

RESULTADOS DE APRENDIZAJE RELACIONADOS CON LA ASIGNATURA

El alumno podrá aplicar las metodologías y técnicas estándares a sus proyectos, independientemente de cuál sea su naturaleza, ya que conocerá los aspectos básicos del desarrollo.

- Asimismo, será capaz de idear proyectos de videojuegos, plasmando dichas ideas gráficamente y por escrito de forma estructurada, ordenada y comprensible.
- El alumno podrá implementar *software* mediante el paradigma de programación orientada a objetos en un entorno gráfico.
- El alumno podrá aplicar con coherencia en el desarrollo de los videojuegos principios matemáticos y físicos, tanto en el diseño 2D como 3D, animación y programación.
- El alumno podrá decidir entre las distintas técnicas de desarrollo de *software* que conoce en función de las necesidades actuales de su proyecto.
- El alumno podrá identificar los distintos pasos del ciclo de un proyecto, empleando las técnicas adecuadas a cada uno de ellos, mejorando así la calidad final del *software*.

6. CONTENIDOS / TEMARIO / UNIDADES DIDÁCTICAS

Breve descripción de los contenidos

- Introducción a la programación orientada a objetos: La complejidad inherente al *software*. Crisis del *software*. Factores en la calidad del *software*. El papel de la abstracción. Orientación a objetos. Características. Modularidad. Diseño de módulos. Acoplamiento y cohesión. Tipos abstractos de datos
- Conceptos Fundamentales de Programación Orientada a Objetos (POO). Clases y objetos. Estructura interna de un objeto. Mensajes. Herencia. Ligadura dinámica y polimorfismo.
- Modelado de objetos: Relaciones: Relación de generalización y especialización. Relación de agregación. Relación de asociación. Análisis y diseño orientado a objetos.
- Lenguajes específicos del paradigma orientado a objetos. Sintaxis. Entornos de desarrollo. Librerías gráficas del lenguaje.

Temario detallado

Tema 1. Introducción a la programación orientada a objetos

- La complejidad inherente al *software*.
- Crisis del *software*. Factores en la calidad del *software*.
- El papel de la abstracción. Orientación a objetos.
- Características. Modularidad. Diseño de módulos. Acoplamiento y cohesión.
- Tipos abstractos de datos.

Tema 2. Conceptos fundamentales de Programación Orientada a Objetos (POO).

- Clases y objetos. Estructura interna de un objeto.
- Mensajes. Herencia. Relaciones de agregación.

Tema 3. Modelado de objetos

- Ligadura dinámica.
- Polimorfismo.
- Clases abstractas. Interfaces.

Tema 4. Análisis y diseño orientado a objetos: Identificación de clases.

- Identificación de responsabilidades. Establecimiento de la jerarquía de herencia.

Tema 5. Lenguajes específicos del paradigma orientado a objetos.

- Sintaxis. Entornos de desarrollo.
- Librerías gráficas del lenguaje.

Tema 6. Control de errores y excepciones en C#.

- Utilidad de las excepciones.
- Empleo de las excepciones. Lanzamiento de excepciones.
- Captura y tratamiento de excepciones.

Tema 7: Estructuras de datos en C#

- Listas enlazadas.
- Pilas. Colas.

7. CRONOGRAMA

UNIDADES DIDÁCTICAS / TEMAS	PERÍODO TEMPORAL
Tema 1. Introducción a la programación orientada a objetos	Septiembre - Octubre
Tema 2. Conceptos fundamentales de Programación Orientada a Objetos (POO)	Octubre
Tema 3. Modelado de objetos	Noviembre
Tema 4. Análisis y diseño orientado a objetos: Identificación de clases	Noviembre
Tema 5. Lenguajes específicos del paradigma orientado a objetos	Noviembre
Tema 6. Control de errores y excepciones en C#	Diciembre
Tema 7: Estructuras de datos en C#	Diciembre - Enero

8. MODALIDADES ORGANIZATIVAS Y MÉTODOS DE ENSEÑANZA

MODALIDAD ORGANIZATIVA	MÉTODO DE ENSEÑANZA	COMPETENCIAS RELACIONADAS	HORAS PRESEN C.	HORAS DE TRABAJO AUTÓNOMO	TOTAL DE HORAS
Clases teóricas	Exposición de los temas. Explicar planificación de la asignatura: programa, apuntes y bibliografía. Repasos al inicio de la clase. Resolución de dudas: temas y lecturas. Pruebas de evaluación.	CG7, CE7 CE8, CE13	30	5	35
Clases prácticas	Resolución de ejercicios. Debates sobre los temas y especialmente sobre ejercicios y lecturas. Presentaciones. Pruebas de evaluación.	CG7, CE7 CE8, CE13	20	15	35
Tutorías	Preparación de clase mediante lectura de los temas. Planificación de debates y comentarios mediante la preparación de las lecturas. Resolución de ejercicios. Comentarios y resolución de dudas presencialmente o por correo electrónico.	CG7, CE7 CE8, CE13	10	-	10
Trabajo personal del alumno	Lecturas: preparación y búsqueda de información complementaria. Estudio personal. Preparación de comentarios y debates. Tutorías libres y voluntarias.	CG7, CE7 CE8, CE13	-	50	50

9. SISTEMA DE EVALUACIÓN

ACTIVIDAD DE EVALUACIÓN	CRITERIOS DE EVALUACIÓN	VALORACIÓN RESPECTO A LA CALIFICACIÓN FINAL (%)
Exámenes/Pruebas objetivas	- Contenidos: cumplimiento de los requisitos indicados en el enunciado (40%) y defensa oral de la misma	90%
Trabajos y Proyectos individuales y/o cooperativos	- Aplicación correcta del paradigma OO (45%) - Documentación (5%)	
Asistencia Participativa	- Contenidos: cumplimiento de los requisitos indicados en el enunciado (40%) y defensa oral de la misma - Aplicación correcta del paradigma OO (45%) - Documentación (5%)	10%

CONSIDERACIONES GENERALES ACERCA DE LA EVALUACIÓN:

Asistencia a Clase

- La asistencia a clase es obligatoria. Sin una asistencia demostrada de al menos un 80%, el alumno no podrá presentarse a examen debiendo acudir a la siguiente convocatoria. No es necesario justificar las faltas, y por tanto no se admitirán justificantes de las mismas, por lo que superado el 20% de faltas de asistencia, el alumno deberá presentarse en convocatoria extraordinaria.
- La Dirección/Coordinación de la Titulación podrá considerar situaciones excepcionales, previo informe documental, debiendo ser aprobadas por la Dirección Académica de ESNE.
- Se exigirá puntualidad al alumno en el comienzo de las clases. Una vez transcurridos cinco minutos de cortesía, el profesor podrá denegar la entrada en el aula.

Entregas de Trabajos

- En convocatoria ordinaria, los alumnos deben presentar y aprobar todas las entregas que se les soliciten. La no entrega de un trabajo supondrá suspender la asignatura.
- Los trabajos deben entregarse en las fechas que solicite el profesor, no se admiten entregas posteriores. Si excepcionalmente se acepta un trabajo fuera de plazo, la máxima calificación a obtener será 5.
- En los trabajos en grupo, la calificación será individual por cada alumno, atendiendo a criterios de conocimiento de la materia, esfuerzo, presentación, asistencia a tutorías,

etc. Por tanto, miembros de un mismo grupo pueden tener calificaciones diferentes.

- Los trabajos, una vez calificados, deben ser retirados por los alumnos en el tiempo que se determine. Pasado este plazo, los trabajos podrán ser destruidos.

Evaluación Ordinaria

- Para superar la asignatura es necesario aprobar todas las partes. La evaluación de la asignatura se realizará mediante trabajos prácticos que los alumnos deben entregar y presentar al profesor de forma presencial, para la realización de un examen oral acerca de la misma y de los conceptos de la OO incluidos en la misma.

Se realizaran dos entregas:

- Una individual, para evaluar los conceptos básicos de la programación OO.
- Una en grupo para evaluar los conceptos avanzados de la programación OO.

- Para superar la asignatura es necesario obtener una media de 5 puntos. Para realizar la media, la nota de la parte de Exámenes/Pruebas objetivas debe sumar más de 4 puntos.

Evaluación Extraordinaria

- En la evaluación extraordinaria, los alumnos deben volver a presentar los trabajos que no hayan sido aprobados en convocatoria ordinaria. Además, el profesor de la asignatura podrá solicitar la realización de un trabajo extra en la evaluación extraordinaria.

10. BIBLIOGRAFÍA / WEBGRAFÍA

Bibliografía básica

Joyanes Aguilar L., (1998), *Programación orientada a objetos*. McGraw-Hill / Interamericana de España, S.A.

Albahari J. y Albahari B., (2010), *C# 4.0 Pocket Reference*. O'Reilly.

Bibliografía complementaria

VVAA. Microsoft C#. Curso de programación. Ra – Ma.

Arriola N., Cosío L., *C# Guía Total del Programador*. RedUsers.

Wirfs-Brock R. y col. *Designing Object-Oriented Software*. Prentice Hall.

Taylor D. *Object-Oriented Information Systems: Planning and Implementations*. John Wiley & Sons.