

**CENTRO UNIVERSITARIO DE TECNOLOGÍA Y ARTE
DIGITAL**



PLANIFICACIÓN DE LA DOCENCIA UNIVERSITARIA

GUÍA DOCENTE

Diseño Web y Aplicaciones

1. DATOS DE IDENTIFICACIÓN DE LA ASIGNATURA.

Título:	Grado en Diseño de Productos interactivos				
Facultad:	Centro Universitario de Tecnología y Arte Digital (U-tad)				
Departamento/Instituto:	Área de Diseño Especializado				
Materia:	Tecnología de productos interactivos				
Denominación de la asignatura:	Diseño web y aplicaciones				
Código:	52047				
Curso:	Cuarto				
Semestre:	Segundo				
Tipo de asignatura (básica, obligatoria u optativa):	Obligatoria				
Créditos ECTS:	3				
Modalidad/es de enseñanza:	Presencial				
Lengua vehicular:	Castellano				
Equipo docente:	Profesor pendiente de contratación. Asignatura no abierta 2017_18				
Profesor/a:	Profesor pendiente de contratación. Asignatura no abierta 2017_18				
Grupos:	4º DPIN				
Despacho:	Sala de Profesores				
Teléfono:	91 640 28 11	Ext.	113	E-mail:	Profesor pendiente de contratación
Página web:	http://www.u-tad.com				

2. REQUISITOS PREVIOS.

Esenciales:
Haber cursado la asignatura "Scripting II".
Aconsejables:
Haber cursado la asignatura "Introducción a la programación" y "Scripting I".

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

Campo de conocimiento al que pertenece la asignatura.
Esta asignatura pertenece al módulo de Diseño avanzado y dentro de éste a la materia de Tecnología de productos interactivos.
Relación de interdisciplinariedad con otras asignaturas del curriculum.
En esta asignatura el alumno desarrollará un conocimiento avanzado de la programación, llevando más allá los objetivos vistos en las asignaturas "Introducción a la programación", "Scripting I" y "Scripting II".
Aportaciones al plan de estudios e interés profesional de la asignatura.
Es una asignatura que a la vez agrupa y ejercita conocimientos ya adquiridos, y prepara al alumno para el trabajo de programación a un nivel mayor que las asignaturas ya mencionadas. Proporcionará al alumno unos conocimientos para ser más autónomo en la programación de productos digitales interactivos.

4. RESULTADOS DE APRENDIZAJE EN RELACIÓN CON LAS COMPETENCIAS QUE DESARROLLA LA ASIGNATURA.

COMPETENCIAS GENÉRICAS	RESULTADOS DE APRENDIZAJE RELACIONADOS CON LAS COMPETENCIAS GENÉRICAS
<ul style="list-style-type: none">• CG2 - Saber adaptarse al cambio y a las nuevas situaciones con flexibilidad y versatilidad.• CG6 - Manifestar motivación por	<ul style="list-style-type: none">• Resolver situaciones y problemas del ámbito profesional de manera versátil y creativa.• Apreciar e implementar las mejoras

<p>la calidad.</p> <ul style="list-style-type: none"> • CG8 - Manifiestar capacidad para trabajar en equipo. • CG18 - Gestionar adecuadamente la información. 	<p>necesarias para dotar a los trabajos de calidad profesional</p> <ul style="list-style-type: none"> • Trabajar en equipo • Gestionar correctamente la información analizando, sintetizando y recogiendo información de diversas fuentes.
COMPETENCIAS ESPECÍFICAS	RESULTADOS DE APRENDIZAJE RELACIONADOS CON LAS COMPETENCIAS ESPECÍFICAS
<ul style="list-style-type: none"> • CE7 - Conocer los fundamentos prácticos del uso y programación de ordenadores y herramientas de desarrollo de productos interactivos. • CE18 - Aplicar los conocimientos teóricos y prácticos de diseño de productos para el desarrollo de contenidos. 	<ul style="list-style-type: none"> • Usar motores de juegos actuales para la creación de videojuegos. • Utilizar programación sencilla para mejorar el diseño de juegos simples • Valorar las técnicas de inteligencia artificial necesarias para un videojuego

5. CONTENIDOS

<ul style="list-style-type: none"> • Metodologías de desarrollo de software aplicadas a la creación de videojuegos • Aspectos avanzados de la Programación Orientada a Objeto: polimorfismo, herencia múltiple, sobrecarga • Aspectos de la programación a bajo nivel • Diseño de algoritmos simples y eficientes • Estructuras de datos

6. CRONOGRAMA

UNIDADES DIDÁCTICAS / TEMAS	PERÍODO TEMPORAL
Metodologías de desarrollo de software	Semanas 1-2-3
Aspectos avanzados de la Programación Orientada a Objeto	Semanas 4-5-6

Aspectos de la programación a bajo nivel	Semanas 7-8-9
Diseño de algoritmos simples y eficientes	Semanas 10-11-12
Estructuras de datos	Semanas 13-14-15

7. MODALIDADES ORGANIZATIVAS Y MÉTODOS DE ENSEÑANZA

MODALIDAD ORGANIZATIVA	MÉTODO DE ENSEÑA	HORAS PRESENCIA	TRABAJO AUTÓNOMO	TOTAL DE HORAS
Clases teóricas	Lección magistral	8	0	8
Seminarios y talleres	Estudio de casos Resolución de ejercicios y problemas	4	0	4
	Aprendizaje basado en problemas Aprendizaje orientado a proyectos	11	0	11
Clases prácticas		0	0	0
Prácticas externas		0	0	0
Tutorías	Aprendizaje orientado a proyectos Aprendizaje basado en problemas	4	0	4
Actividades de evaluación		3	0	3
Estudio y trabajo en grupo	Aprendizaje cooperativo	0	7	7
	Estudio de casos Resolución de ejercicios y problemas Aprendizaje basado en problemas Aprendizaje orientado a proyectos	0	38	38
Estudio y trabajo autónomo, individual				
Total		30	45	75

8. SISTEMA DE EVALUACIÓN

ACTIVIDAD DE EVALUACIÓN	VALORACIÓN RESPECTO A LA CALIFICACIÓN FINAL (%)
Pruebas de autoevaluación, heteroevaluación o coevaluación	40%
Trabajos, proyectos, informes, memorias, portfolios	50%
Prácticas o pruebas de simulación real o ficticia	10%

9. BIBLIOGRAFÍA / WEBGRAFÍA

Bibliografía básica

- Learning JavaScript, 2nd Edition. Add Sparkle and Life to Your Web Pages. Shelley Powers. O'Reilly Media. 978-0-596-52187-5, 2008.
- Getting Started with Processing. Casey Reas and Ben Fry. Published June 2010, O'Reilly Media.

Bibliografía recomendada

- The Ruby Programming Language. David Flanagan, Yukihiro Matsumoto. O'Reilly. 2008. ISBN-10: 0596516177. First Edition.
- *Programming Ruby 1.9 (3rd edition): The Pragmatic Programmers' Guide*. Dave Thomas, with Chad Fowler y Andy Hunt. 2009. ISBN: 978-1-93435-608-1
- *Why's Poignant Guide to Ruby (online)*. Jonathan Gillette. <http://mislav.uniqpath.com/poignant-guide/>
- *Thinking in Python*. Bruce Eckel. <http://www.mindview.net/Books/TIPython>
- Learning Python (Help for Programmers), 4th edition. Mark Lutz y David Ascher. O'Reilly Media.
- *Starting Out with C++*. Tony Gaddis. Addison-Wesley. ISBN-13: 978-0132576253

10.- MATERIAL, SOFTWARE Y HERRAMIENTAS NECESARIAS

TIPOLOGÍA DEL AULA:

- Ordenador del profesor conectado a proyector. Software instalado: Visual Studio.
- Pizarra digital.
- Pizarra blanca.
- Un ordenador para cada alumno en el aula. Software instalado: Visual Studio.

MATERIALES:

No.

SOFTWARE:

Visual Studio.