

## GUÍA DOCENTE 2019-2020

### DATOS GENERALES DE LA ASIGNATURA

<b>ASIGNATURA:</b>	Programación web I
<b>PLAN DE ESTUDIOS:</b>	Grado de Comunicación Audiovisual
<b>FACULTAD:</b>	Facultad de Ciencias Sociales y Humanidades
<b>CARÁCTER DE LA ASIGNATURA:</b>	Optativa
<b>ECTS:</b>	6
<b>CURSO:</b>	Cuarto
<b>SEMESTRE:</b>	Primero
<b>IDIOMA EN QUE SE IMPARTE:</b>	Castellano
<b>PROFESORADO:</b>	Sandra Tejerina
<b>DIRECCIÓN DE CORREO ELECTRÓNICO:</b>	<a href="mailto:sandra.tejerina@uneatlantico.es">sandra.tejerina@uneatlantico.es</a>

### DATOS ESPECÍFICOS DE LA ASIGNATURA

<b>REQUISITOS PREVIOS:</b>
No aplica.
<b>CONTENIDOS:</b>
<ul style="list-style-type: none"> <li>• Tema 1. Historia y estandarización</li> <li>• Tema 2. Arquitectura de apps web             <ul style="list-style-type: none"> <li>2.1 Páginas estáticas</li> <li>2.2 Aplicaciones simples en PHP</li> <li>2.3 Una aplicación en PHP + Javascript</li> <li>2.4 Script &amp; API</li> </ul> </li> </ul>

- Tema 3. HTML
  - 3.1 Historia del HTML.
  - 3.2 Mejores prácticas en creación de HTML.
  - 3.3 Capa de presentación.
  - 3.4 Editores de HTML.
  - 3.5 Errores comunes al momento de crear HTML.
  - 3.6 Depurar HTML.
  
- Tema 4. CSS
  - 4.1 Historia de CSS.
  - 4.2 Mejores prácticas en creación de CSS.
  - 4.3 Capa de presentación.
  - 4.4 Editores de CSS.
  - 4.5 Errores comunes al momento de crear CSS.
  - 4.5 Selectores.
  - 4.6 Depurar CSS.
  - 4.7 Bootstrap
  
- Tema 5. Javascript
  - 5.1 Historia de Javascript.
  - 5.2 Mejores prácticas en codificación en JS.
  - 5.3 Editores de Javascript.
  - 5.4 Depurar Javascript.
  - 5.5 Usos más comunes de JS: Validación de formularios, Animaciones.
  - 5.6 Clases y objetos en Javascript. Prototipado rápido.
  - 5.7 JQuery
  - 5.8 AJAX. Asíncrono
  - 5.9 Eventos de servidor
  
- Tema 6. Frameworks
  - 6.1 Crear una app web single page
  - 6.2 Angular
  - 6.3 React.
  
- Tema 7. PHP
  - 7.1 Historia de PHP.
  - 7.2 Mejores prácticas en codificación de PHP.

- 7.3 Cómo depurar en PHP.
  - 7.4 Hackings más comunes (validación de datos).
  - 7.5 ¿Qué significa ser un lenguaje dinámico?.
  - 7.6 Crear una api rest
  - 7.7 Librerías más comunes
  - 7.8 Plugins y herramientas para desarrollar en PHP
- Tema 8. NODE.JS javascript de lado servidor
- 8.1 API en Node.JS
- Tema 9. CMS
- 9.1 Qué es y para qué sirve un CMS.
  - 9.2 Crear un *ecommerce* en 20 horas con Wordpress
  - 9.3 Crear un tema hijo y extender el padre
  - 9.4 Conceptos básicos
  - 9.5 Woocommerce
  - 9.6 Cómo crear un plugin
  - 9.7 Cómo hacer un CMS extensible: sistema de eventos
  - 9.8 AJAX en Wordpress
  - 9.9 Cómo mantener un servidor *stage* y un producción sincronizados

## COMPETENCIAS

### COMPETENCIAS GENERALES:

Que los estudiantes sean capaces de:

- CG1. Capacidad para reconocer, analizar e interpretar diversos procesos de comunicación.
- CG3. Capacidad para crear y comunicar mensajes a públicos diversos.
- CG4. Capacidad para aplicar sus conocimientos en propuestas innovadoras.

### COMPETENCIAS PROPIAS DE LA ASIGNATURA:

Que los alumnos sean capaces de:

- CEO10. Identificar los procesos que intervienen en la programación informática y aplicar conocimientos prácticos en la programación Web.

#### RESULTADOS DE APRENDIZAJE:

En esta asignatura se espera que los alumnos alcancen los siguientes resultados de aprendizaje:

- Distinguir los recursos informáticos necesarios para el desarrollo de páginas web.
- Aportar en procesos de desarrollo de aplicaciones informáticas para la creación en sitios web.

## METODOLOGÍAS DOCENTES Y ACTIVIDADES FORMATIVAS

#### METODOLOGÍAS DOCENTES:

En esta asignatura se ponen en práctica diferentes metodologías docentes con el objetivo de que los alumnos puedan obtener los resultados de aprendizaje definidos anteriormente:

- Método expositivo.
- Estudio y análisis de casos.
- Resolución de ejercicios.
- Aprendizaje basado en problemas.
- Aprendizaje orientado a Proyectos.
- Aprendizaje cooperativo / Trabajo en grupos.
- Trabajo autónomo.

#### ACTIVIDADES FORMATIVAS:

A partir de las metodologías docentes especificadas anteriormente, en esta asignatura, el alumno participará en las siguientes actividades formativas:

Actividades formativas	
Actividades dirigidas	Clases expositivas
	Clases prácticas
	Seminarios y talleres
Actividades supervisadas	Supervisión de actividades
	Tutorías (individual / en grupo)
Actividades autónomas	Preparación de clases
	Estudio personal y lecturas
	Elaboración de trabajos
	Trabajo individual en campus virtual

El primer día de clase, la profesora proporcionará información más detallada al respecto.

## SISTEMA DE EVALUACIÓN

### CONVOCATORIA ORDINARIA:

En la convocatoria ordinaria de esta asignatura se aplican los siguientes instrumentos de evaluación:

Actividades de evaluación		Ponderación
Evaluación continua	Examen parcial	25 %
	Entregas ejercicios y portfolios	20 %
	Interés y participación del alumno en la asignatura (individual)	5%
Evaluación final	Examen teórico-práctico	50 %

### CONVOCATORIA EXTRAORDINARIA:

La convocatoria extraordinaria tendrá lugar durante el mes de julio (consúltese el calendario académico fijado por la universidad). Esta consistirá en la realización de examen teórico-práctico del 50 % de la nota final de la asignatura. El resto de la nota se complementará con la calificación obtenida en la evaluación continua de la convocatoria ordinaria.

## BIBLIOGRAFÍA Y RECURSOS DE REFERENCIA GENERALES

### BIBLIOGRAFÍA BÁSICA:

Las siguientes referencias son de consulta obligatoria:

- Ballard, P. & Moncur, M. (2009). *Ajax, JavaScript y PHP*. Madrid: Anaya Multimedia 2009.
- Schmitt, C. (2010). *Curso de CSS*. Madrid: Anaya Multimedia/O'Reilly.
- Sierra, A. y Alfonseca, M. (2014). *Programación en C/C++* (tercera edición). Madrid: Anaya Multimedia.

### BIBLIOGRAFÍA COMPLEMENTARIA:

- McFarland, D. S. (2012). *JavaScript y jQuery*. Madrid: Anaya Multimedia.
- Welling, L. & Thomson, L. (2005). *Desarrollo Web con PHP y MySQL*. Madrid: Anaya Multimedia.