



## DATOS IDENTIFICATIVOS

### Aplicaciones con lenguajes de script

Asignatura	Aplicaciones con lenguajes de script			
Código	O06G150V01941			
Titulación	Grado en Ingeniería Informática			
Descriptores	Creditos ECTS	Seleccione	Curso	Cuatrimestre
	6	OP	4	2c
Lengua Impartición	Castellano Inglés			
Departamento	Informática			
Coordinador/a	García Pérez-Schofield, José Baltasar			
Profesorado	García Pérez-Schofield, José Baltasar			
Correo-e	jbgarcia@uvigo.es			
Web	<a href="http://webs.uvigo.es/jbgarcia/">http://webs.uvigo.es/jbgarcia/</a>			
Descripción general	Desarrollo de aplicaciones mediante lenguajes de script. El idioma inglés se emplea para impartir las clases de prácticas de la asignatura, pero no en las actividades de evaluación.			

## Competencias

Código	
A2	Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio.
A3	Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética.
A4	Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.
A5	Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.
B1	Capacidad para concebir, redactar, organizar, planificar, desarrollar y firmar proyectos en el ámbito de la ingeniería en informática que tengan por objeto, de acuerdo con los conocimientos adquiridos, la concepción, el desarrollo o la explotación de sistemas, servicios y aplicaciones informáticas.
B2	Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la informática de acuerdo con los conocimientos adquiridos.
B3	Capacidad para diseñar, desarrollar, evaluar y asegurar la accesibilidad, ergonomía, usabilidad y seguridad de los sistemas, servicios y aplicaciones informáticas, así como de la información que gestionan.
B4	Capacidad para definir, evaluar y seleccionar plataformas hardware y software para el desarrollo y la ejecución de sistemas, servicios y aplicaciones informáticas, de acuerdo con los conocimientos adquiridos.
B5	Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad, de acuerdo con los conocimientos adquiridos.
B6	Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes de acuerdo con los conocimientos adquiridos.
B7	Capacidad para conocer, comprender y aplicar la legislación necesaria durante el desarrollo de la profesión de Ingeniero Técnico en Informática y manejar especificaciones, reglamentos y normas de obligado cumplimiento.
B9	Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.
B11	Capacidad para analizar y valorar el impacto social y medioambiental de las soluciones técnicas, comprendiendo la responsabilidad ética y profesional de la actividad del Ingeniero Técnico en Informática.
B12	Conocimiento y aplicación de elementos básicos de economía y de gestión de recursos humanos, organización y planificación de proyectos, así como la legislación, regulación y normalización en el ámbito de los proyectos informáticos, de acuerdo con los conocimientos adquiridos.

C4	Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería
C5	Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería
C7	Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente
C8	Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social
C11	Conocimiento, administración y mantenimiento de sistemas, servicios y aplicaciones informáticas
C12	Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos
C13	Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema
C14	Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados
C15	Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman
C16	Conocimiento de las características, funcionalidades y estructura de los Sistemas Operativos y diseñar e implementar aplicaciones basadas en sus servicios
C18	Conocimiento y aplicación de las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño y el análisis e implementación de aplicaciones basadas en ellos
C19	Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web
C20	Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real
C22	Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software
C24	Conocimiento de la normativa y la regulación de la informática en los ámbitos nacional, europeo e internacional
C25	Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software
C26	Capacidad para valorar las necesidades del cliente y especificar los requisitos software para satisfacer estas necesidades, reconciliando objetivos en conflicto mediante la búsqueda de compromisos aceptables dentro de las limitaciones derivadas del coste, del tiempo, de la existencia de sistemas ya desarrollados y de las propias organizaciones
C27	Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles
C28	Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales
C29	Capacidad de identificar, evaluar y gestionar los riesgos potenciales asociados que pudieran presentarse
C30	Capacidad para diseñar soluciones apropiadas en uno o más dominios de aplicación utilizando métodos de la ingeniería del software que integren aspectos éticos, sociales, legales y económicos
C31	Capacidad para comprender el entorno de una organización y sus necesidades en el ámbito de las tecnologías de la información y las comunicaciones
C32	Capacidad para seleccionar, diseñar, desplegar, integrar, evaluar, construir, gestionar, explotar y mantener las tecnologías de hardware, software y redes, dentro de los parámetros de coste y calidad adecuados
C33	Capacidad para emplear metodologías centradas en el usuario y la organización para el desarrollo, evaluación y gestión de aplicaciones y sistemas basados en tecnologías de la información que aseguren la accesibilidad, ergonomía y usabilidad de los sistemas
C35	Capacidad para seleccionar, desplegar, integrar y gestionar sistemas de información que satisfagan las necesidades de la organización, con los criterios de coste y calidad identificados
C36	Capacidad de concebir sistemas, aplicaciones y servicios basados en tecnologías de red, incluyendo Internet, web, comercio electrónico, multimedia, servicios interactivos y computación móvil
C37	Capacidad para comprender, aplicar y gestionar la garantía y seguridad de los sistemas informáticos
D1	I1: Capacidad de análisis, síntesis y evaluación
D2	I2: Capacidad de organización y planificación
D3	I3: Comunicación oral y escrita en la lengua nativa
D5	I5: Capacidad de abstracción: capacidad de crear y utilizar modelos que reflejen situaciones reales
D7	I7: Capacidad de buscar, relacionar y estructurar información proveniente de diversas fuentes y de integrar ideas y conocimientos
D8	I8: Resolución de problemas
D9	I9: Capacidad de tomar decisiones
D10	I10: Capacidad para argumentar y justificar lógicamente las decisiones tomadas y las opiniones
D11	P1: Capacidad de actuar autónomamente
D12	P2: Capacidad de trabajar en situaciones de falta de información y/o bajo presión
D13	P3: Capacidad de integrarse rápidamente y trabajar eficientemente en equipos unidisciplinarios y de colaborar en un entorno multidisciplinar

D15 P5: Capacidad de relación interpersonal

D16 S1: Razonamiento crítico

D18 S3: Aprendizaje autónomo

D19 S4: Adaptación a nuevas situaciones

D20 S5: Creatividad

D21 S6: Liderazgo

D22 S7: Tener iniciativa y ser resolutivo

### Resultados de aprendizaje

Resultados previstos en la materia

Resultados de Formación  
y Aprendizaje

RA1. Desarrollar todo tipo de software de aplicación a través de todas las fases.

A2 B1 C4 D1  
B4 C7 D2  
B6 C12 D5  
B12 C13 D7  
C14 D8  
C16 D9  
C18 D11  
C22 D12  
C25 D13  
C27 D15  
C28 D16  
C29 D19  
C31 D20  
C32 D22  
C33  
C35

RA2. Conocer las herramientas de planificación y control para el desarrollo colaborativo de un proyecto informático.

A2 B1 C8 D1  
A3 B2 C19 D2  
A4 B5 C26 D3  
B9 C27 D7  
B11 C29 D8  
C31 D9  
C35 D10  
C36 D12  
C37 D13  
D15  
D16  
D18  
D19  
D20  
D21  
D22

RA3. Conocer métodos prácticos para la especificación de todos los componentes durante el desarrollo de un paquete software.

A2 B1 C4 D1  
A3 B3 C5 D2  
B5 C11 D5  
B11 C14 D7  
C15 D8  
C16 D9  
C18 D11  
C19 D12  
C20 D16  
C22 D18  
C24 D19  
C25 D20  
C27 D22  
C29  
C30  
C32  
C36  
C37

RA4. Conocer las técnicas disponibles para la integración de software.

A2	B1	C4	D1
	B4	C7	D2
	B5	C27	D5
	B9	C32	D7
	B12		D8
			D9
			D10
			D11
			D12
			D16
			D18
			D19
			D20
			D22

RA5. Conocer métodos y estándares para el desarrollo, verificación y mantenimiento de una aplicación integrada.

A2	B1	C4	D1
	B3	C5	D2
	B4	C11	D5
	B5	C12	D7
	B6	C15	D8
	B11	C16	D9
		C18	D11
		C19	D12
		C20	D16
		C22	D18
		C25	D19
		C27	D20
		C28	D22
		C29	
		C30	
		C32	
		C36	
		C37	

RA6. Ser capaz de aplicar las técnicas de ingeniería del software para obtener aplicaciones de gran calidad y con las funcionalidades solicitadas por el usuario considerando el sistema como un conjunto de aplicaciones.

A2	B1	C4	D1
A3	B2	C5	D2
A4	B3	C7	D3
A5	B4	C8	D7
	B5	C11	D8
	B6	C15	D9
	B7	C16	D10
	B9	C18	D11
	B11	C19	D12
	B12	C20	D13
		C22	D15
		C24	D16
		C25	D18
		C26	D19
		C27	D20
		C29	D21
		C30	D22
		C31	
		C32	
		C33	
		C36	
		C37	

RA7. Trabajar como parte de un equipo que desarrolla proyectos software compuestos de varias fases e hitos de control.	A2	B1	C4	D1
	A3	B2	C5	D2
	A4	B3	C7	D3
		B4	C11	D5
		B5	C15	D7
		B6	C16	D8
		B7	C18	D9
		B9	C19	D10
		B11	C20	D11
		B12	C22	D12
			C24	D13
			C25	D15
			C26	D16
			C27	D18
			C28	D19
			C29	D20
			C30	D21
		C32	D22	
		C33		
		C35		
		C36		
		C37		

RA8. Presentar de forma adecuada la documentación de un proyecto a cada una de las personas implicadas en el desarrollo del mismo: analistas, diseñadores, programadores y clientes.	A2	B1	C4	D1
	A3	B2	C5	D2
	A4	B3	C28	D3
	A5	B9	C29	D7
		B11	C30	D8
				D9
				D10
				D11
				D12
				D13
				D15
				D16
			D18	
			D19	
			D20	
			D21	
			D22	

## Contenidos

Tema	
Introducción	Reseña histórica. Evolución de los lenguajes de script. Tendencias actuales.
Lenguajes de programación web	Encapsulación. Herencia. Polimorfismo. Modelo de objetos. Creación y distribución de aplicaciones.
Persistencia	Serialización simple en formatos JSON y XML.

## Planificación

	Horas en clase	Horas fuera de clase	Horas totales
Lección magistral	18	29	47
Prácticas de laboratorio	12	48	60
Resolución de problemas y/o ejercicios	3	10	13
Práctica de laboratorio	20	10	30

\*Los datos que aparecen en la tabla de planificación son de carácter orientativo, considerando la heterogeneidad de alumnado

## Metodologías

Descripción
-------------

Lección magistral	Presencial: presentación, mediante medios audiovisuales, de los contenidos teóricos de cada tema. Este método se combinará con ejemplos ilustrativos de código y con la realización de preguntas para motivar e incrementar el interés del alumno.  No presencial: revisión, comprensión y afianzamiento de los contenidos.
Prácticas de laboratorio	El objetivo es que el alumno aplique los contenidos teóricos en la solución de problemas simples de programación.  Presencial: resolución de pequeños problemas de programación, que guiarán el proceso de realizar un proyecto completo.  No presencial: resolución de pequeños problemas de programación.

<b>Atención personalizada</b>	
<b>Metodologías</b>	<b>Descripción</b>
Lección magistral	

<b>Evaluación</b>			
	Descripción	Calificación	Resultados de Formación y Aprendizaje
Resolución de problemas y/o ejercicios	Se realizarán dos pruebas escritas parciales, una aproximadamente en la mitad de la materia, y otra al final. Estas pruebas son eliminatorias, es decir, en cuanto a la parte teórica, aquellos alumnos que aprueben estas pruebas no necesitarán presentarse a primera opción. Resultados: RA2, RA3, RA4, RA5, RA8.	60	A3 B1 C12 D1 A4 B3 C13 D2 A5 B5 C15 D3 B6 C16 D5 B7 C29 D7 B9 C30 D8 D9 D10 D11 D12 D16 D18 D19 D20 D22
Práctica de laboratorio	Los alumnos realizarán un proyecto a medida que avance la asignatura, aprovechando y aplicando los conocimientos teóricos asimilados en la sesión magistral. Este proyecto será necesario entregarlo al finalizar la asignatura. Resultados: RA1, RA2, RA3, RA4, RA5, RA6, RA7, RA8.	40	A2 B1 C4 D1 A3 B2 C5 D2 A4 B3 C7 D3 B4 C8 D5 B5 C11 D7 B6 C12 D8 B7 C13 D9 B9 C14 D10 B11 C15 D13 B12 C16 D15 C18 D20 C19 D21 C20 C22 C24 C25 C26 C27 C28 C29 C30 C31 C32 C33 C35 C36 C37

**Otros comentarios sobre la Evaluación**

**Crterios de evaluación para asistentes 1ª edición de actas**

Aparte de lo indicado en la tabla anterior, deberá tenerse en cuenta que:

- Todos los estudiantes que se presenten a cualquiera de las pruebas se entiende que siguen la asignatura de forma presencial y por lo tanto deberán de seguir el procedimiento de evaluación descrito anteriormente.
- Si un estudiante no se presenta a alguna de las pruebas se le asignará, como mucho, una calificación de 4 en el total de las mismas, según el resto de calificaciones.

### **Criterios de evaluación para no asistentes, 2ª edición de actas y fin de carrera**

Consiste en:

- una prueba individual del total de la asignatura. Esta prueba se corresponde con la entrada "pruebas de respuesta larga" en la tabla anterior (aunque solo es una), siendo aplicables sus porcentajes en la calificación final, resultados de aprendizaje y competencias.
- un proyecto. Esta prueba se corresponde con la entrada "pruebas prácticas" en la tabla anterior, siendo aplicables sus porcentajes en la calificación final, resultados de aprendizaje y competencias.

### **Proceso de cualificación de actas**

Tanto para alumnos que opten a la primera o segunda opción, se tendrá en cuenta que para aplicar los porcentajes descritos es necesario que en cualquier prueba realizada se obtenga una nota igual o superior a 4. Para considerar la asignatura superada, la calificación final debe ser igual o superior a 5. Si no se ha obtenido al menos un 4 en alguna prueba, a pesar de poder obtener una calificación final igual o superior a 5, la nota final será de 4.

Nótese que los alumnos asistentes o presenciales son aquellos que se han presentado a alguna prueba durante el desarrollo de la asignatura, mientras que los no presenciales son los que tratarán de superar la asignatura presentándose exclusivamente a los exámenes finales, que se relatan a continuación.

### **Fechas de evaluación**

El calendario de pruebas de evaluación aprobado oficialmente por la Junta de Centro de la ESEI se encuentra publicado en la página web <http://www.esei.uvigo.es>.

---

### **Fuentes de información**

#### **Bibliografía Básica**

García Perez-Schofield, Baltasar, **Introducción a la programación con Google Application Engine en Python**, 2, Bubok.es - 978-1326634568, 2016

Sanderson, Dan, **Programming Google App Engine**, 1, O'Reilly Media - 978-1491900208, 2015

#### **Bibliografía Complementaria**

Severance, Charles, **Using Google App Engine**, 1, O'Reilly Media - 978-0596800697, 2009

<https://developers.google.com/appengine/?hl=es>, **Google App Engine**, Google, 2008

<http://es.diveintopython.net/>, **Sumérgete en Python**, 2001

---

### **Recomendaciones**