



Guía docente				
Datos Identificativos				2012/13
Asignatura (*)	Proceso Software	Código	614G01019	
Titulación	Grao en Enxeñaría Informática			
Descritores				
Ciclo	Periodo	Curso	Tipo	Créditos
Grado	2º cuatrimestre	Segundo	Obligatoria	6
Idioma	Castellano			
Prerrequisitos				
Departamento	Tecnoloxías da Información e as Comunicaciós			
Coordinador/a	Garcia Vazquez, Rafael Carlos	Correo electrónico	rafael.garcia@udc.es	
Profesorado	Ares Casal, Juan Manuel Garcia Vazquez, Rafael Carlos Rodriguez Yañez, Santiago	Correo electrónico	juan.ares@udc.es rafael.garcia@udc.es santiago.rodriguez@udc.es	
Web				
Descripción general	Esta materia destaca a importancia de seguir un proceso de desenvolvemento software de calidade, diferenciando o proceso de desenvolvemento artesán do enxeñeiro. Concretamente, a materia centrase nas tarefas incluídas nas distintas etapas do ciclo de vida do proceso software: planificación, análise, deseño, codificación, verificación e validación. Finalmente, presétase ao alumno os conceptos básicos da calidade do software.			

Competencias de la titulación	
Código	Competencias de la titulación
A7	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.
A14	Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
A22	Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.
A25	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.
A26	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.
A28	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.
A51	Capacidad para comprender y aplicar los principios y las técnicas de gestión de la calidad y de la innovación tecnológica en las organizaciones.
B6	Valorar críticamente el conocimiento, la tecnología y la información disponible para resolver los problemas con los que deben enfrentarse.
B7	Asumir como profesional y ciudadano la importancia del aprendizaje a lo largo de la vida.
B9	Capacidad de resolución de problemas
B10	Trabajo en equipo
B11	Capacidad de análisis y síntesis
B12	Capacidad para organizar y planificar
B15	Preocupación por la calidad
C6	Valorar críticamente el conocimiento, la tecnología y la información disponible para resolver los problemas con los que deben enfrentarse.
C7	Asumir como profesional y ciudadano la importancia del aprendizaje a lo largo de la vida.

Resultados de aprendizaje	
Competencias de materia (Resultados de aprendizaje)	Competencias de la titulación



Capacidad para comprender la importancia del Proceso Software	A7 A14 A22 A25 A26 A28	B7 B9 B11 B15	C7
Capacidad para comprender la necesidad de gestionar y controlar el Proceso Software	A25 A26 A51	B7 B9 B12 B15	C7
Capacidad para aprender de forma autónoma		B7	C7
Capacidad para trabajar en equipo		B6 B9 B10 B11 B12	C6
Capacidad para adaptarse a nuevas situaciones y contextos		B6	C6

Contenidos	
Tema	Subtema
Introducción a la Ingeniería del Software	Antecedentes históricos Definiciones e importancia de la Ingeniería del Software
Ciclos de Vida	Definiciones Ciclo de Vida Vs Ciclo de Desarrollo Características de los principales Ciclos de Vida
Planificación de proyectos	Definiciones Calendarización del proyecto
Desarrollo Software	Análisis Diseño Codificación Pruebas
Metodologías de desarrollo	Definiciones e importancia Características deseables Principales metodologías de desarrollo
Introducción a la calidad del Software	Calidad del Software Control de calidad Aseguramiento de la calidad

Planificación			
Metodologías / pruebas	Horas presenciales	Horas no presenciales / trabajo autónomo	Horas totales
Sesión magistral	30	30	60
Prácticas de laboratorio	25	25	50
Prueba objetiva	2	33	35
Atención personalizada	5	0	5

(*) Los datos que aparecen en la tabla de planificación són de carácter orientativo, considerando la heterogeneidad de los alumnos

Metodologías	
Metodologías	Descripción



Sesión magistral	Clases en las que se presentan los conceptos básicos de la asignatura y en las que se muestran ejemplos de su aplicación.
Prácticas de laboratorio	Trabajos en los que se aplican los conocimientos adquiridos a lo largo del curso a casos prácticos.
Prueba objetiva	Examen escrito para valorar los conocimientos teóricos y prácticos adquiridos durante el curso.

Atención personalizada

Metodologías	Descripción
Prácticas de laboratorio	Un aspecto importante de la asignatura es saber aplicar los conceptos teóricos presentados en la misma. Para ello se realizarán un conjunto de prácticas a lo largo del curso. Dentro de estas prácticas destacan las supervisadas, planteadas con la intención de detectar posibles desviaciones dentro del proceso de aprendizaje y, de este modo, aplicar las medidas correctivas necesarias para subsanarlas.

Evaluación

Metodologías	Descripción	Calificación
Prácticas de laboratorio	A lo largo del curso se plantearán diferentes tipos de prácticas: supervisadas (que representan el 30% de la nota final) y de trabajo autónomo (computando el 10%).	40
Prueba objetiva	Examen escrito individual sobre los conocimientos teóricos y prácticos de la asignatura. Del peso asignado a esta prueba (60% de la calificación final), el 50% se corresponde con la parte teórica y el 10% con la parte práctica.	60

Observaciones evaluación

La evaluación de la asignatura se articula en base a los siguientes mecanismos:

1. Prácticas supervisadas (30% de la nota final). Estas prácticas se caracterizan por una supervisión constante del trabajo realizado por los alumnos con objeto de realizar una evaluación continua de cada alumno. Para ello se tendrá en cuenta los siguientes aspectos: asistencia, participación y aplicación de los conocimientos adquiridos en la asignatura.
2. Prácticas de trabajo autónomo (10% de la nota final). En estas prácticas los alumnos tendrán que resolver, de manera autónoma, el trabajo propuesto por el profesor. Posteriormente, tendrán que defenderlo en el horario establecido por el profesor.
3. Examen final (60% de la nota final). El examen final se compone de dos partes: una teórica, con un peso del 50% sobre la nota final, y otra práctica, con un peso del 10%. Es importante remarcar que en ambas partes podrá preguntarse cualquier aspecto visto en la asignatura.

Finalmente, cabe indicar que para aprobar la asignatura es necesario cumplir con los siguientes requisitos:

1. Obtener una calificación mínima de 5 puntos sobre 10 entre todas las pruebas según la ponderación establecida.
2. Obtener un mínimo de 5 puntos sobre 10 en la parte teórica del examen final. En caso de no superar este requisito, la nota final máxima que se podrá alcanzar será un 4,5.

Fuentes de información

Básica	
Complementaria	

Recomendaciones

Asignaturas que se recomienda haber cursado previamente

Gestión de Proyectos/614G01021
Aseguramiento de la Calidad/614G01028
Análisis y Desarrollo de los Sistemas de Información/614G01041
Metodologías de Desarrollo/614G01051
Ingeniería de Requisitos/614G01222

Asignaturas que se recomienda cursar simultáneamente



Asignaturas que continúan el temario

Programación I/614G01001
Informática Básica/614G01002
Programación II/614G01006
Bases de Datos/614G01013
Paradigmas de Programación/614G01014
Diseño Software/614G01015

Otros comentarios

La asignatura pretende dar una visión general del proceso software, de manera que el alumno comprenda la importancia de la aplicación rigurosa de dicho proceso tanto para la satisfacción de las necesidades del usuario/cliente como del posterior mantenimiento del producto obtenido. Por ello, a lo largo del curso se introducirán numerosos conceptos que requerirán una posterior profundización en función de la especialidad elegida por el alumno.

(*) La Guía Docente es el documento donde se visualiza la propuesta académica de la UDC. Este documento es público y no se puede modificar, salvo cosas excepcionales bajo la revisión del órgano competente de acuerdo a la normativa vigente que establece el proceso de elaboración de guías