



GUÍA DOCENTE 2019-2020
DESARROLLO AVANZADO DE SISTEMAS SOFTWARE

1. Denominación de la asignatura:

DESARROLLO AVANZADO DE SISTEMAS SOFTWARE

Titulación

Grado en Ingeniería Informática

Código

6389

2. Materia o módulo a la que pertenece la asignatura:

Ingeniería del Software

3. Departamento(s) responsable(s) de la asignatura:

Ingeniería Civil

4.a Profesor que imparte la docencia (Si fuese impartida por mas de uno/a incluir todos/as) :

Carlos López Nozal y Jesús Alonso Abad

4.b Coordinador de la asignatura

Carlos López Nozal

5. Curso y semestre en el que se imparte la asignatura:

4º curso, 8º semestre

6. Tipo de la asignatura: (Básica, obligatoria u optativa)

Optativa



7. Requisitos de formación previos para cursar la asignatura:

Preferiblemente haber cursado asignaturas optativas de la mención de ingeniería del software

8. Número de créditos ECTS de la asignatura:

6

9. Competencias que debe adquirir el alumno/a al cursar la asignatura

Competencias Generales de Grado:

CG5 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.

CG8 Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.

CG9 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.

Competencias Específicas de la materia:

IS1 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.

IS3 Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.

IS4 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

Competencias Transversales:

CT1 Capacidad de análisis y síntesis. Instrumental

CT7 Resolución de problemas. Instrumental

CT9 Trabajo en equipo. Personal



CT14 Razonamiento crítico. Sistémica

CT16 Aprendizaje autónomo. Sistémica

CT22 Motivación por la calidad.

10. Programa de la asignatura

10.1- Objetivos docentes
- Conocer y aplicar las actividades relacionadas con el proceso de gestión de calidad del software. - Conocer estándares internacionales de gestión de calidad. - Definir métricas que sirvan como base a la evaluación, comparación o evaluación de productos o procesos. - Conocer y aplicar de varios conjuntos de métricas software. - Conocer el proceso de reingeniería y las actividades de reingeniería inversa y reestructuración de programas. - Reestructurar de programas mediante operaciones de refactorización.
10.2- Unidades docentes (Bloques de contenidos)
Conceptos: Medición y Calidad Administración de Calidad Medición y métricas del software Defectos de diseño Refactorización del software Patrones de reingeniería
10.3- Bibliografía
BIBLIOGRAFÍA BÁSICA Dolado, J.J., y L. Fernández., (2000) Medición para la Gestión en la Ingeniería del Software, Rama, Fowler, M.; Beck, K.; Brant, J.; Opdyke, W. y Roberts D., Refactoring: Improving the Design of Existing Code, Addison-Wesley, Nierstrasz, Oscar, Stéphane Ducasse y Serge Demeyer. , (2009) Object-Oriented Reengineering Patterns, Square Bracket Associates, Piattini, Mario y García, Félix, (2003) Calidad en el desarrollo y mantenimiento del software, Ra-Ma,



BIBLIOGRAFÍA COMPLEMENTARIA

Sommerville, I., (2002) Ingeniería del Software, 6ª Edición, Addison Wesley Pearson,
Wake William C., (2004) Refactoring workbook, Addison-Wesley,
978-0-321-10929-3 0-321-10929-5,

11. Metodología de enseñanza y aprendizaje y su relación con las competencias que debe adquirir el estudiante:

Metodología	Competencia relacionada	Horas presenciales	Horas de trabajo	Total de horas
Clases Teóricas	CG5, CG8, CG9, CT1, CT7, CT9, CT10, CT14, CT22, IS1, IS3, IS4	24	0	24
Clases prácticas (pequeño grupo)	CG5, CG8, CG9, CT1, CT7, CT9, CT10, CT14, CT22, IS1, IS3, IS4	24	0	24
Estudio personal	CG5, CG8, CG9, CT1, CT7, CT9, CT10, CT14, CT22, IS1, IS3, IS4	0	48	48
Realización de trabajos, informes, memorias y pruebas de evaluación	CG5, CG8, CG9, CT1, CT7, CT9, CT10, CT14, CT22, IS1, IS3, IS4	6	48	54
Total		54	96	150

12. Sistemas de evaluación:

La evaluación será continua atendiendo al Reglamento de evaluación de la Universidad de Burgos. A tal efecto se presentan dos opciones de evaluación: Modelo normal y Modelo excepcional.

Modelo normal. El artículo 10 del Reglamento de evaluación en su punto 10.2 establece que en la primera convocatoria se calificará al estudiante de acuerdo a los resultados de la evaluación continua.

Para superar la asignatura es requisito haber superado los siguientes procedimientos de evaluación

- Prueba de evaluación de conocimientos teóricos 30% (nota de corte 4 sobre 10)
- Aplicación de un plan de calidad con métricas software en un proyecto software disponible en un repositorio de código abierto 40% (nota de corte 4 sobre 10)
- Aplicación de un proceso de mantenimiento usando refactorizaciones sobre un caso de estudio en Java 30% (nota de corte 4 sobre 10)



Respecto de la segunda convocatoria, el artículo 10 del Reglamento de evaluación en su punto 10.2, indica que todas las pruebas podrán recuperarse en la segunda convocatoria y en esta convocatoria no podrán exigirse aquellas pruebas superadas ya en la primera.

Procedimiento	Peso
Prueba de evaluación de conocimientos teóricos	30 %
Aplicación de un plan de calidad con métricas software en proyectos software disponible en un repositorio de código abierto	40 %
Aplicación de un proceso de mantenimiento usando refactorizaciones sobre un caso de estudio en Java	30 %
Total	100 %

Evaluación excepcional:

Modelo excepcional. Atendiendo a lo expuesto en el artículo 9 del Reglamento de Evaluación en su punto 9.1, los estudiantes que, por razones excepcionales, no puedan seguir los procedimientos habituales de evaluación continua deberán solicitar por escrito al Decano o Director de Centro acogerse a una «evaluación excepcional». Dicho escrito con las razones que justifiquen la imposibilidad de seguir la evaluación continua deberá presentarse antes del inicio del semestre lectivo o durante las dos primeras semanas de impartición de la asignatura.

Esta forma de evaluación implica los mismos requisitos que el modelo normal con la salvedad de la presencialidad, que deberá sustituirse con la asistencia a tutorías programadas con el profesor.

Para superar la asignatura es requisito haber superado los siguientes procedimientos de evaluación

- Prueba de evaluación de conocimientos teóricos 30% (nota de corte 4 sobre 10)
- Aplicación de un plan de calidad con métricas software en un proyecto software disponible en un repositorio de código abierto 40% (nota de corte 4 sobre 10)
- Aplicación de un proceso de mantenimiento usando refactorizaciones sobre un caso de estudio en Java 30% (nota de corte 4 sobre 10)

13. Recursos de aprendizaje y apoyo tutorial:

Pizarra y Proyector
Páginas Webs relacionadas
Bibliografía disponible en la Biblioteca
Aplicaciones interactivas en la Plataforma UBUvirtual
Tutorías individualizadas o en grupo a demanda de los alumnos



14. Calendarios y horarios:

El calendario aprobado por la Junta de Escuela de la Escuela Politécnica Superior y los horarios publicados en los tabloneros oficiales de la E.P.S.

15. Idioma en que se imparte:

Español. Bibliografía en inglés.