

30245 - Arquitectura software

Información del Plan Docente

Año académico: 2019/20

Asignatura: 30245 - Arquitectura software

Centro académico: 110 - Escuela de Ingeniería y Arquitectura

Titulación: 439 - Graduado en Ingeniería Informática

Créditos: 6.0

Curso: 3

Periodo de impartición: Segundo semestre

Clase de asignatura: ---

Materia: ---

1. Información Básica

1.1. Objetivos de la asignatura

La asignatura y sus resultados previstos responden a los siguientes planteamientos y objetivos:

Tras haber cursado la asignatura de 'Ingeniería del Software', los alumnos saben aplicar técnicas de análisis y diseño orientado a objetos. En esta asignatura los alumnos aprenden a diseñar sistemas de software atendiendo a las estructuras fundamentales del mismo, es decir abstrayéndose de los detalles de sus componentes.

La asignatura presenta un catálogo de técnicas y patrones arquitecturales y los aplica a un conjunto de casos de estudio. Además se desarrolla un trabajo de asignatura que pone en práctica los conocimientos teóricos adquiridos.

1.2. Contexto y sentido de la asignatura en la titulación

La asignatura 'Arquitectura Software' proporciona a los alumnos una introducción a algunas de las propuestas, técnicas y patrones necesarios para la construcción de sistemas **software** de tamaño mediano, grande y muy grande, a partir de los conocimientos adquiridos fundamentalmente en la asignatura de 'Ingeniería del Software'. Es una asignatura obligatoria dentro de la especialidad en 'Ingeniería del Software'.

1.3. Recomendaciones para cursar la asignatura

El alumno que curse esta asignatura ha de contar con una formación como la que se recibe en la asignatura de 'Ingeniería del Software'.

2. Competencias y resultados de aprendizaje

2.1. Competencias

Al superar la asignatura, el estudiante será más competente para...

Concebir, diseñar y desarrollar proyectos de Ingeniería.

Resolver problemas y tomar decisiones con iniciativa, creatividad y razonamiento crítico.

Comunicar y transmitir conocimientos, habilidades y destrezas en castellano y en inglés.

Usar las técnicas, habilidades y herramientas de la Ingeniería necesarias para la práctica de la misma.

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.

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.

2.2. Resultados de aprendizaje

El estudiante, para superar esta asignatura, deberá demostrar los siguientes resultados...

Sabe diseñar y documentar la arquitectura de un sistema de software, desde distintos puntos de vista y niveles de

abstracción.

Conoce diferentes patrones arquitecturales y es capaz de aplicarlos para diseñar sistemas software.

Es capaz de proponer patrones arquitecturales originales aplicables al diseño de sistemas software.

Sabe aplicar técnicas aprendidas en "Ingeniería de Software" (casos de uso, modelado estructural y modelado dinámico) al diseño de arquitecturas software

Conoce y utiliza herramientas CASE para diseñar y documentar arquitecturas software.

2.3.Importancia de los resultados de aprendizaje

Conocer técnicas específicas de diseño de arquitecturas software facilita concebir y llevar a cabo sistemas medianos, grandes o muy grandes. Este tipo de técnicas ayudan a diseñar y documentar sistemas software mantenibles, eficientes y escalables, aplicando los patrones arquitecturales más adecuados para cada caso. Los conocimientos de arquitectura de software permitirán a los alumnos crecer como ingenieros, siendo éstos clave para enlazar sus conocimientos previos sobre programación e ingeniería del software con el desarrollo profesional de sistemas de software grandes.

3.Evaluación

3.1.Tipo de pruebas y su valor sobre la nota final y criterios de evaluación para cada prueba

El estudiante deberá demostrar que ha alcanzado los resultados de aprendizaje previstos mediante las siguientes actividades de evaluación

Convocatoria de Junio. La evaluación de la asignatura se realiza en base a dos pruebas:

P1. Ejercicios del curso. Se requiere una nota mínima de 5.0 puntos en esta prueba para aprobar la asignatura. Si se obtiene esta nota mínima, entonces la prueba pondera un 40% en la nota de la asignatura.

P2. Trabajo de asignatura. Es obligatorio presentar este trabajo y obtener una nota mínima de 5.0 puntos para aprobar la asignatura. Si se obtiene esta nota mínima, entonces esta prueba pondera un 60% en la nota de la asignatura. Cada grupo deberá entregar y defender un trabajo focalizado en las diferentes vistas de la arquitectura de un software realizado durante el curso.

Es obligatorio realizar y entregar las dos pruebas para poder aprobar la asignatura. Si en alguna prueba, la nota obtenida fuera inferior a 5.0, la calificación final de la asignatura será la media ponderada de todas las calificaciones (40% P1 y 60% P2), con un máximo de 4.0.

Convocatoria de Septiembre. La evaluación de la asignatura se realiza en base a dos pruebas análogas a las de la convocatoria de Junio, con las mismas ponderaciones y exigencia de notas mínimas. Las calificaciones del alumno obtenidas en la convocatoria de Junio en cualquier de las dos pruebas se mantienen en Septiembre, salvo que el alumno opte por presentarse a la prueba correspondiente en esta nueva convocatoria, en cuyo caso prevalecerá la nueva calificación.

4.Metodología, actividades de aprendizaje, programa y recursos

4.1.Presentación metodológica general

El proceso de aprendizaje que se ha diseñado para esta asignatura se basa en lo siguiente:

1. El estudio y trabajo continuado desde el primer día de clase. El aprendizaje de conceptos y técnicas para el diseño y documentación de arquitecturas de software a través de las clases magistrales, en las que se favorecerá la participación de los alumnos. En las clases de problemas, la aplicación de los conocimientos adquiridos en las clases magistrales para resolver problemas de diseño y documentación de la arquitectura del software de aplicaciones de ejemplo. En estas clases los alumnos desempeñarán un papel activo en la discusión y resolución de los problemas.
2. El trabajo en equipo, donde los alumnos desarrollan, bajo la supervisión de los profesores, un trabajo focalizado en un software realizado durante el curso.

4.2.Actividades de aprendizaje

El programa que se ofrece al estudiante para ayudarle a lograr los resultados previstos comprende las siguientes actividades...

En las clases impartidas en el aula se desarrollará el temario de la asignatura.

En las clases de problemas se resolverán problemas de aplicación de los conceptos y técnicas presentadas en el programa de la asignatura.

Se realizará un trabajo que consistirá en diseñar y documentar la arquitectura de un sistema software.

4.3.Programa

- Introducción a la arquitectura de software
- Documentación de la arquitectura software
 - Vista de Módulos; Vista de Componente y Conector. Vista de Distribución
 - Documentación de interfaces
- Patrones arquitecturales
 - Cliente/Servidor
 - Broker
 - MVC
 - Publicación-suscripción
 - Microkernel
 - Arquitecturas adaptativas

4.4.Planificación de las actividades de aprendizaje y calendario de fechas clave

Calendario de sesiones presenciales y presentación de trabajos

La organización docente prevista de la asignatura es la siguiente.

- Clases teóricas y de problemas (2 horas semanales).
- Tutorías de los trabajos en grupo. Sesiones semanales de treinta minutos con cada grupo. En ellas los profesores realizan el seguimiento (asesoría y corrección) del trabajo de curso.

Trabajo

Trabajo del estudiante

La dedicación del estudiante para alcanzar los resultados de aprendizaje en esta asignatura se estima en 150 horas distribuidas del siguiente modo:

- 30 horas, aproximadamente, de actividades presenciales (sesiones en el aula teóricas -20- y de problemas -10-)
- 90 horas de trabajo en grupo (incluyen aproximadamente 7 horas de seguimiento y tutoría realizadas con el profesor)
- 25 horas de trabajo y estudio individual efectivo (estudio de apuntes y textos, resolución de problemas y preparación de clases)
- 5 horas dedicadas a las pruebas de evaluación.

El calendario de exámenes y las fechas de entrega de trabajos se anunciará con suficiente antelación.

4.5.Bibliografía y recursos recomendados

<http://psfunizar7.unizar.es/br13/egAsignaturas.php?codigo=30245&Identificador=14710>

[BB: Bibliografía básica / BC: Bibliografía complementaria]

- [BB] Documenting Software Architectures: Views and Beyond / Paul Clements...[et al.]. - 2nd ed. Addison-Wesley Professional, 2010
- [BB] Pattern-Oriented Software Architecture. A System of Patterns / Frank Buschmann...[et al.] John Wiley & Sons, 1996
- [BB] Web Services / Gustavo Alonso ... [et al.] Springer, 2004