



Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación

30222 - Ingeniería del Software

Guía docente para el curso 2014 - 2015

Curso: 3, Semestre: 1, Créditos: 6.0

Información básica

Profesores

- **José Javier Meseguer Hernaiz** jmerse@unizar.es
- **Francisco Javier Noguerras Iso** jnog@unizar.es
- **Miguel Angel Latre Abadia** latre@unizar.es
- **Félix Serna Fortea** fserna@unizar.es

Recomendaciones para cursar esta asignatura

El alumno que curse esta asignatura ha de contar con una formación en programación orientada a objetos del nivel de la asignatura de Tecnología de Programación.

Actividades y fechas clave de la asignatura

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

Inicio

Resultados de aprendizaje que definen la asignatura

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

- 1:** Comprende la necesidad de abordar el desarrollo de sistemas software desde un punto de vista basado en conceptos de Ingeniería del Software.
- 2:** Conoce y sabe aplicar los principios de análisis, diseño, verificación y validación de software.
- 3:** Conoce y tiene capacidad para utilizar técnicas y herramientas de ayuda al desarrollo de software (herramientas CASE).

- 4:** Comprende qué es una metodología y cómo abordar la construcción de sistemas software desde una perspectiva orientada a objetos.

Introducción

Breve presentación de la asignatura

Esta asignatura persigue que el alumno comprenda que el desarrollo de software es una industria, y que como en cualquier industria se deben aplicar sólidos principios de ingeniería y buenas prácticas de gestión. Es decir, para obtener un producto de calidad con un tiempo y coste determinado se deben aplicar de forma sistemática un conjunto de métodos, técnicas y herramientas. Esta primera asignatura de Ingeniería del Software se centra en el aprendizaje de las tareas vinculadas a la construcción del producto software, dejando para la asignatura de Proyecto Software el aprendizaje sobre los aspectos relacionados con la gestión del proceso de construcción. Para ello el alumno va a aprender a determinar los requisitos de un software a construir, realizar el análisis del sistema, transformar el análisis en el diseño de una solución concreta, y planificar y ejecutar un conjunto de pruebas que permitan verificar y validar el software construido.

Contexto y competencias

Sentido, contexto, relevancia y objetivos generales de la asignatura

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

Tras haber cursado con aprovechamiento un conjunto de asignaturas de programación (Programación I, Programación II, Tecnología de Programación) donde se ha aprendido a diseñar la implementación de programas (y sus estructuras de datos) utilizando un paradigma orientado a objeto, en esta asignatura el alumno va a aprender a abordar la construcción de software de mayor entidad y desde una perspectiva global, no solamente centrada en aspectos de implementación. Tomando como punto de partida la descripción de un software a construir, el alumno aprenderá a determinar sus requisitos, analizarlos, realizar un diseño, y plantear un conjunto de pruebas sobre el software construido. Para ello se utilizará una metodología orientada a objeto, aunque los conceptos presentados son aplicables a cualquier otra metodología de desarrollo de software. Asimismo, se contextualizarán los conocimientos adquiridos en todas las asignaturas de Programación, Estructuras de Datos, Bases de Datos e Interacción Persona Ordenador.

La asignatura tiene un marcado carácter aplicado. El alumno aprenderá los conceptos necesarios sobre análisis, diseño y pruebas aplicándolos sobre un conjunto de ejemplos de sistemas presentados tanto en las clases de problemas, como en las distintas sesiones de las prácticas de laboratorio.

Contexto y sentido de la asignatura en la titulación

La asignatura de Ingeniería del Software constituye el primer peldaño de un estudiante de Ingeniería Informática para formarse en los principios de la Ingeniería del Software, es decir, en la aplicación coordinada de técnicas, metodologías y herramientas para producir un software de alta calidad, con un determinado presupuesto y antes de una determinada fecha. Es una asignatura obligatoria englobada en la materia de formación común en Ingeniería del Software y Sistemas de Información.

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

- 1:** Concebir, diseñar y desarrollar proyectos de Ingeniería.
- 2:** Planificar, presupuestar, organizar, dirigir y controlar tareas, personas y recursos.
- 3:** Usar las técnicas, habilidades y herramientas de la Ingeniería necesarias para la práctica de la misma.

- 4:** Combinar los conocimientos generalistas y los especializados de Ingeniería para generar propuestas innovadoras y competitivas en la actividad profesional.
- 5:** Aplicar los principios, metodologías y ciclos de vida de la ingeniería de software.

Importancia de los resultados de aprendizaje que se obtienen en la asignatura:

Aprender a abordar el desarrollo de sistemas de información aplicando los principios de una ingeniería es esencial para un ingeniero informático que participe en un equipo humano de desarrollo de software de tamaño medio o grande. Lo que aprenda en esta asignatura, que complementa a lo ya aprendido en las asignaturas de programación, estructuras de datos y algoritmos, bases de datos e interacción persona ordenador, será un importante paso adelante en su formación como ingeniero.

Evaluación

Actividades de evaluación

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

1:

En la Escuela de Ingeniería y Arquitectura de Zaragoza

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

- **P1. Prueba escrita** en la que responder cuestiones y resolver ejercicios y problemas. 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 80% en la nota de la asignatura. La fecha de realización de esta prueba será la que la dirección de la EINA determine para la realización de la prueba global de la asignatura.
- **P2. Trabajo de asignatura y resultados de prácticas de laboratorio.** Esta prueba pondera un 20% en la nota de la asignatura. Cada alumno deberá entregar los trabajos que se indiquen en las prácticas de la asignatura. Asimismo, se entregará un trabajo por grupos donde se realice el análisis, diseño y pruebas de una segunda versión de la aplicación informática abordada en las clases prácticas, pero ahora incluyendo nueva funcionalidad. Los trabajos se entregarán de forma electrónica con anterioridad a la fecha que la dirección de la EINA establezca para la realización de la prueba global de la asignatura.

Es obligatorio realizar y entregar ambas pruebas para poder aprobar la asignatura. Si en una de las pruebas, o en las dos, la nota obtenida fuera inferior a 5.0, la calificación final de la asignatura será la media ponderada de las dos calificaciones (80% P1 y 20% 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 Febrero, con las mismas ponderaciones y exigencia de notas mínimas. Las calificaciones del alumno obtenidas en la convocatoria de Febrero en cualquiera de las pruebas (P1 y P2) 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.

2:

En la Escuela Universitaria Politécnica de Teruel

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

- **P1. Examen de problemas** en el que deberá resolverse uno o varios casos prácticos. Se requiere una nota mínima de 4.0 puntos en esta prueba para aprobar la asignatura. En esta convocatoria, esta prueba representará el 30% de la nota final.
- **P2. Trabajos asociados a las prácticas de laboratorio.** Se requiere una nota mínima de 4.0 puntos en

esta prueba para poder aprobar la asignatura. En esta convocatoria esta prueba representará el 70% de la nota final de la asignatura.

Convocatoria de Septiembre. La evaluación de la asignatura se realiza en base a dos pruebas análogas a las de la convocatoria de Febrero, con las mismas notas mínimas pero con distintas ponderaciones. En esta convocatoria, la nota final de la asignatura será $NF = 0.7 \cdot P1 + 0.3 \cdot P2$.

Actividades y recursos

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.
2. El aprendizaje de conceptos y metodologías para el análisis, diseño, verificación y validación de software a través de las clases magistrales, en las que se favorecerá la participación de los alumnos.
3. La aplicación de tales conocimientos al análisis, diseño, verificación y validación de ejemplos de sistemas software en las clases de problemas. En estas clases los alumnos desempeñarán un papel activo en la discusión y resolución de los problemas. En algunas de estas clases el trabajo de cada alumno será evaluado, mediante la realización de pruebas evaluables con carácter voluntario.
4. Las clases de prácticas en laboratorio en las que el alumno aprenderá la tecnología necesaria para realizar el análisis, diseño, verificación y validación de una aplicación informática a lo largo de distintas sesiones.
5. Sólo en EINA: El trabajo en equipo abordando una segunda versión de la aplicación informática que se ha trabajado a lo largo de las prácticas de laboratorio, pero esta vez incluyendo nuevas funcionalidades. El resultado de este trabajo se plasma en la entrega de los documentos correspondientes al análisis, diseño y pruebas.

Actividades de aprendizaje programadas (Se incluye programa)

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

- 1: En las clases impartidas en el aula se desarrollará el temario de la asignatura.
- 2: En las clases de problemas se resolverán problemas de aplicación de los conceptos y técnicas presentadas en el programa de la asignatura.
- 3: Las sesiones de prácticas de desarrollan en un laboratorio informático. A lo largo de sus sesiones cada alumno deberá realizar, individualmente o en equipo, trabajos directamente relacionados con los temas estudiados en la asignatura.
- 4: **Sólo en EINA:** Adicionalmente, se solicitará un trabajo por grupos bajo la tutela de un profesor donde se abordará una segunda versión de la aplicación abordada en las sesiones prácticas, donde se incluirá nueva funcionalidad.

Planificación y calendario

Calendario de sesiones presenciales y presentación de trabajos

La organización docente de la asignatura prevista en la **Escuela de Ingeniería y Arquitectura de Zaragoza** es la siguiente:

- Clases teóricas (2 horas semanales)
- Clases de problemas (1 hora semanal)

- Clases prácticas de laboratorio (seis sesiones de 2 horas cada dos semanas). Son sesiones de trabajo análisis, diseño y pruebas de software en laboratorio, tuteladas por un profesor, en las que participan los alumnos de cada uno de los subgrupos en los que se divide el grupo.
- Tutorías de los trabajos en grupo donde los alumnos plantean una segunda versión de la aplicación abordada en las clases prácticas de laboratorio, pero ahora incluyendo nueva funcionalidad.

Presentación de trabajos objeto de evaluación:

- Los trabajos (P2) se entregarán con anterioridad a la fecha límite para su entrega, que será la misma que la dirección de la EINA establezca para la realización de la prueba P1.

La organización docente de la asignatura prevista **en la Escuela Universitaria Politécnica de Teruel** es la siguiente:

- Clases teóricas (2 horas semanales)
- Clases de problemas (1 hora semanal)
- Clases prácticas de laboratorio (1 hora semanal)
- Tutorías de trabajos dirigidos (1 hora semanal). Los alumnos/as deben concertar cita previa.

Presentación de trabajos objeto de evaluación:

- La práctica general de la asignatura (P2) deberá entregarse antes del comienzo del examen (P1).

Programa

Programa de la asignatura

- Introducción a la ingeniería del software.
- Determinación de requisitos.
- Análisis: modelado de objetos, modelado dinámico.
- Diseño: diseño de sistemas, diseño de objetos.
- Prueba del producto software.

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:

- 60 horas, aproximadamente, de actividades presenciales (sesiones en el aula teóricas -30- y de problemas -15- y sesiones en el laboratorio -15-)
- 50 horas de trabajo en grupo
- 35 horas de trabajo y estudio individual efectivo (estudio de apuntes y textos, resolución de problemas, preparación clases y prácticas, desarrollo de programas)
- 5 horas dedicadas a distintas pruebas de evaluación

Bibliografía

Bibliografía recomendada

- Bruegge B, Dutoit AH (2009). **Object-Oriented Software Engineering Using UML**, Patterns, and Java, 3rd Edition. Prentice Hall.
- Kruchten P (2003). **The Rational Unified Process: An Introduction**, 3rd edition. Addison-Wesley Object Technology Series.
- Rumbaugh J, Jacobson I, Booch G (2007). **El Lenguaje Unificado de Modelado. Manual de Referencia**, 2ª edición. Pearson Educación (Addison Wesley), Madrid.
- Gamma E, Helm R, Johnson R, Vlissides J (1996). **Design Patterns**. Addison-Wesley.

Referencias bibliográficas de la bibliografía recomendada

Escuela Universitaria Politécnica

- Bruegge, Bernd. Ingeniería de software orientado a objetos / Bernd Bruegge, Allen H. Dutoit ; traducción, Sergio Luis María Ruiz Faudón ; revisión técnica, Rafael Gamboa Hiraes, Martha Rosa Cordero López, Marco Antonio Dorantes González. 1a. ed. Naucalpan de Juárez, México [etc.] : Pearson Educacion, 2002 []
- Design patterns : Elements of reusable object-oriented software / Erich Gamma...[et al.] . - 40th. print. Boston [etc.] : Addison-Wesley, 2012
- Kruchten, Philippe. The rational unified process : an introduction / Philippe Kruchten . - 3rd ed. Boston [etc.] : Addison Wesley, cop. 2004
- Rumbaugh, James. El lenguaje unificado de modelado UML : manual de referencia / James Rumbaugh, Ivar Jacobson, Grady Booch ; traducción Héctor castán Rodríguez, à"scar Sanjuán Martínez , Mariano de la Fuente Alarcón ; coordinación general y revisión técnica Luis Joyanes Aguilar . - 2ª ed. Madrid [etc.] : Pearson Educación, D. L. 2007

Escuela Politécnica Superior

- 1. Bruegge, Bernd. Ingeniería de software orientado a objetos / Bernd Bruegge, Allen H. Dutoit ; traducción, Sergio Luis María Ruiz Faudón ; revisión técnica, Rafael Gamboa Hiraes, Martha Rosa Cordero López, Marco Antonio Dorantes González. - 1a. ed. Naucalpan de Juárez, México [etc.] : Pearson Educacion, 2002
- 2. Kruchten, Philippe. The rational unified process : an introduction / Philippe Kruchten . - 3rd ed. Boston [etc.] : Addison Wesley, cop. 2004
- 3. Rumbaugh, James. El lenguaje unificado de modelado UML : manual de referencia / James Rumbaugh, Ivar Jacobson, Grady Booch ; traducción Héctor castán Rodríguez, à"scar Sanjuán Martínez , Mariano de la Fuente Alarcón ; coordinación general y revisión técnica Luis Joyanes Aguilar . - 2ª ed. Madrid [etc.] : Pearson Educación, D. L. 2007
- 4. Design patterns : Elements of reusable object-oriented software / Erich Gamma...[et al.] . - 37th. print. Boston [etc.] : Addison-Wesley, 2009