

30226 - Proyecto Software

Información del Plan Docente

Año académico: 2019/20

Asignatura: 30226 - Proyecto Software

Centro académico: 110 - Escuela de Ingeniería y Arquitectura

326 - Escuela Universitaria Politécnica de Teruel

Titulación: 439 - Graduado en Ingeniería Informática

443 - Graduado en Ingeniería Informática

Créditos: 6.0

Curso: 3

Periodo de impartición: 439 - Segundo semestre

439 - Segundo semestre

439 - Segundo semestre

439 - Segundo semestre

439 - Segundo semestre

439 - Segundo semestre

439 - Segundo semestre

439 - Segundo semestre

439 - Segundo semestre

439 - Segundo semestre

443 - Segundo semestre

443 - Segundo semestre

443 - Segundo semestre

443 - Segundo semestre

Clase de asignatura: Obligatoria

Materia: ---

1. Información Básica

1.1. Objetivos de la asignatura

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

Al finalizar el curso el estudiante habrá realizado un proyecto software completo, desde la fase de determinación de requerimientos hasta la entrega al cliente, evaluando y tomando decisiones técnicas que condicionarán su trabajo, y elaborando la documentación que lo acompaña tanto desde el punto de vista del producto como del proceso.

1.2. Contexto y sentido de la asignatura en la titulación

En el marco de las asignaturas cursadas en los dos primeros cursos del grado, el estudiante adquiere competencias y conocimientos que le van a posibilitar desarrollar aplicaciones informáticas de pequeño y mediano tamaño. La asignatura de Ingeniería del Software dota a los alumnos de los fundamentos de ingeniería que requiere el desarrollo profesional de sistemas de software de tamaño mediano o grande, mantenibles y escalables. La asignatura de Proyecto Software aporta el elemento integrador y de soporte que permite que el proceso de construcción de un sistema software se realice de una manera eficaz y eficiente. Para ello se destaca la sistematización de actividades y procesos como un elemento clave para el desarrollo de sistemas de calidad, sobre la base de la mejora continua.

1.3. Recomendaciones para cursar la asignatura

Dada la configuración del plan de estudios de la titulación, en los dos primeros cursos del grado el estudiante adquiere competencias y conocimientos que le van a posibilitar desarrollar aplicaciones informáticas de pequeño y mediano tamaño. Dado que la adquisición de conocimientos y competencias en esta asignatura se efectúa alrededor del desarrollo de una aplicación software de pequeño tamaño, se considera necesario haber superado estos dos cursos para cursar la asignatura.

Adicionalmente, la asignatura es la continuación natural de la asignatura de Ingeniería del Software presentada en el cuatrimestre anterior. Resulta por ello recomendable que se curse una vez se haya superado esta asignatura.

2. Competencias y resultados de aprendizaje

2.1.Competencias

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

Afrontar con éxito los siguientes desempeños transversales:

1. Concebir, diseñar y desarrollar proyectos de Ingeniería.
2. Planificar, presupuestar, organizar, dirigir y controlar tareas, personas y recursos.
3. Combinar los conocimientos generalistas y los especializados de Ingeniería para generar propuestas innovadoras y competitivas en la actividad profesional.
4. Resolver problemas y tomar decisiones con iniciativa, creatividad y razonamiento crítico.
5. Comunicar y transmitir conocimientos, habilidades y destrezas en castellano y en inglés.
6. Usar las técnicas, habilidades y herramientas de la Ingeniería necesarias para la práctica de la misma.
7. Analizar y valorar el impacto social y medioambiental de las soluciones técnicas actuando con ética, responsabilidad profesional y compromiso social.
8. Trabajar en un grupo multidisciplinar y en un entorno multilingüe.
9. Aprender de forma continuada y desarrollar estrategias de aprendizaje autónomo.

Afrontar con éxito los siguientes desempeños relacionados con la Ingeniería del Software:

1. 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.
2. Comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.
3. Conocer la normativa y la regulación de la informática en los ámbitos nacional, europeo e internacional.

2.2.Resultados de aprendizaje

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

- Conoce cómo 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.
- Es capaz de 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.
- Comprende la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.
- Conoce cómo elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.
- Conoce cómo llevar a cabo el mantenimiento de sistemas, servicios y aplicaciones informáticas.
- Conoce los fundamentos básicos de la normativa y la regulación de la informática en los ámbitos nacional, europeo e internacional.
- Aprecia la necesidad del dialogo permanente y colaborativo

2.3.Importancia de los resultados de aprendizaje

La realidad de la industria muestra que el desarrollo de sistemas software se efectúa casi siempre dentro de un marco presupuestario, de acuerdo a un calendario, y cumpliendo una serie de requisitos negociados con un cliente. Esto no puede llevarse a cabo sin la necesaria capacidad para organizar y estructurar el trabajo, optimizar el uso de los recursos con los que se cuenta, sistematizar las tareas que puedan ser repetitivas (con el fin de poder dedicar mayor tiempo a las creativas), y perseguir una mejora continua que redunde en un mayor nivel de calidad.

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

En la **Escuela de Ingeniería y Arquitectura de Zaragoza**:

La evaluación consistirá en una prueba con dos ejercicios:

1. **Realización y defensa de trabajos/proyectos prácticos en grupo (80%)**: Durante esta actividad se les planteará

a los alumnos el desarrollo de un trabajo que deberán realizar en grupo. El proyecto comprenderá la construcción de una aplicación informática de pequeñas dimensiones en la que deberán abordar todas las tareas propias de la ingeniería del software, con especial atención a las vinculadas a la gestión y soporte de las diferentes actividades de desarrollo. El equipo de profesores evaluará la labor de gestión desarrollada sobre la base de unos entregables proporcionados por el grupo y que reflejen esta labor, y sobre la defensa que cada grupo haga de su trabajo (resultados de aprendizaje 1, 2, 3, 7)

2. **Preguntas escritas sobre conceptos aprendidos en teoría y problemas** (20%) (resultados de aprendizaje 1, 4, 5, 6)

Para superar la asignatura será necesario que la suma ponderada de ambos ejercicios sea al menos de 5 sobre 10.

En la **Escuela Universitaria Politécnica de Teruel**:

Realización y defensa de un trabajo/proyecto práctico en grupo. Durante esta actividad se les planteará a los alumnos el desarrollo de un trabajo que deberán realizar en grupo. El proyecto comprenderá la construcción de una aplicación informática de pequeñas dimensiones en la que deberán abordar todas las tareas propias de la ingeniería del software, con especial atención a las vinculadas a la gestión y soporte de las diferentes actividades de desarrollo. El profesor/a evaluará la labor desarrollada por cada alumno/a sobre la base de unos entregables proporcionados por el grupo y que reflejen su actividad, y sobre la defensa que cada grupo haga de su trabajo (resultados de aprendizaje 1, 2, 3, 7).

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.
2. El aprendizaje de conceptos vinculados a las diferentes actividades que integra la gestión de un proyecto software, así como el marco normativo que regula este trabajo, a través de las clases magistrales, en las que se favorecerá la participación de los alumnos.
3. La aplicación de tales conocimientos a casos prácticos en las clases de problemas. En estas clases los alumnos desempeñarán un papel activo en la discusión y resolución de los problemas. Algunos de estos problemas se encontrarán vinculados a actividades específicas del proyecto que han de desarrollar como trabajo en equipo que se ha de afrontar a lo largo del cuatrimestre.
4. Las clases de prácticas en laboratorio en las que el alumno aprenderá el uso de herramientas que den soporte a los conocimientos teóricos desarrollados en las clases magistrales. En algunas de las prácticas se llevará el uso de estas herramientas al contexto específico del trabajo en equipo que se ha de afrontar a lo largo del cuatrimestre.
5. El trabajo en equipo abordando el desarrollo de un proyecto que comprenderá la construcción de una aplicación informática de pequeñas dimensiones en la que deberán abordar todas las tareas propias de la ingeniería del software, con especial atención a las vinculadas a la gestión y soporte de las diferentes actividades de desarrollo. Este trabajo permitirá aplicar de forma conjunta todos los aspectos teóricos desarrollados en las clases magistrales, así como enfatizar el desarrollo de todas las competencias vinculadas al trabajo en equipo colaborativo.
6. Eventualmente, la participación de profesionales externos que permitan mostrar a los alumnos la realidad de la industria, y cómo los conceptos que se plantean en clase son aplicados en esta realidad.

4.2. Actividades de aprendizaje

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

En la **Escuela de Ingeniería y Arquitectura de Zaragoza**:

- 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.
- Las sesiones de prácticas se desarrollan en seminarios y laboratorios informáticos. A lo largo de sus sesiones cada alumno deberá realizar, individualmente o en equipo, trabajos directamente relacionados con los temas estudiados en la asignatura y con el trabajo en equipo.
- Adicionalmente, se solicitará un trabajo en equipo, bajo la tutela de los profesores, donde se abordará un proyecto de software de pequeño tamaño.

En la **Escuela Universitaria Politécnica de Teruel**:

- 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.
- Las sesiones de prácticas de desarrollan en un laboratorio informático. A lo largo de sus sesiones los alumnos/as deberán realizar el trabajo en equipo solicitado al comienzo del curso, bajo la tutela de un profesor, donde se abordará un proyecto de software de pequeño tamaño.

4.3. Programa

El programa de la asignatura se compone de dos bloques:

1. Diseño de Proyectos de Software. Arquitectura, documentación y construcción automática.
2. Gestión de Proyectos de Software. Visión general, gestión de configuraciones, métricas y estimaciones, planificación, riesgos, equipo humano, calidad y entorno.

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

Calendario de sesiones presenciales y presentación de trabajos

Escuela de Ingeniería y Arquitectura de Zaragoza. Calendario de sesiones presenciales y presentación de trabajos.

La organización docente prevista de las sesiones presenciales es la siguiente:

- Clases magistrales y resolución de problemas y casos (promedio de 2 horas por semana)
- Prácticas de laboratorio (1 sesión de 2 horas programadas según el calendario que ofrezca el Centro)

Los horarios de todas las clases y fechas de las sesiones de prácticas se anunciarán con suficiente antelación a través de las webs del centro y de la asignatura.

Los proyectos propuestos serán entregados al finalizar el cuatrimestre, en las fechas que se señalen.

Escuela Universitaria Politécnica de Teruel. Calendario de sesiones presenciales y presentación de trabajos.

La organización docente prevista de las sesiones presenciales en el campus de Teruel es la siguiente:

- Clases magistrales (2 horas por semana)
- Resolución de problemas y casos (1 hora por semana)
- Prácticas de laboratorio (1 hora por semana)
- Tutorías de trabajos dirigidos (1 hora por semana, los alumnos/as deberán concertar cita con el profesor/a)

Los horarios de todas las clases serán los fijados por el centro.

Los trabajos realizados por los equipos de alumnos/as, deberán ser entregados (y defendidos) en la fecha que el centro establezca en el calendario de exámenes de la titulación.

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:

En la Escuela de Ingeniería y Arquitectura de Zaragoza:

- 32 horas, aproximadamente, de actividades presenciales (sesiones en el aula teóricas y de problemas y sesiones en el laboratorio)
- 103 horas de trabajo en grupo
- 10 horas de trabajo y estudio individual efectivo (estudio de apuntes y textos, resolución de problemas, preparación de clases y prácticas, desarrollo de programas, etc.)
- 5 horas dedicadas a distintas pruebas de evaluación

En la Escuela Universitaria Politécnica de Teruel:

- 60 horas de actividades presenciales (30 horas de sesiones de teoría y problemas, y 30 horas de sesiones de prácticas en el laboratorio)
- 42 horas de trabajo en prácticas T6
- 43 horas de trabajo y estudio individual efectivo
- 5 horas dedicadas a distintas pruebas de evaluación

El calendario detallado de las diversas actividades a desarrollar se establecerá una vez que la Universidad haya aprobado el calendario académico del curso correspondiente. En cualquier caso, las fechas importantes serán anunciadas con la

suficiente antelación.

4.5. Bibliografía y recursos recomendados

[BB: Bibliografía básica / BC: Bibliografía complementaria]

- Zaragoza:

<http://psfunizar7.unizar.es/br13/egAsignaturas.php?codigo=30226&Identificador=14676>

- [BB] Chrissis, Mary Beth. CMMI for development : guidelines for process integration and product improvement / Mary Beth Chrissis, Mike Konrad, Sandy Shrum . 3rd ed. Upper Saddle River, New Jersey : Addison-Wesley, cop. 2011
- [BB] Humphrey, W.S. Introduction to the Team Software Process / W.S. Humphrey. Addison-Wesley. 1999
- [BB] Pressman, Roger S.. Ingeniería del Software : un enfoque práctico / Roger S. Pressman . - 7ª ed. México D. F. : McGraw-Hill Interamericana, cop. 2010
- [BB] Sommerville, Ian. Software engineering / Ian Sommerville . 10th ed. Boston [etc.] : Pearson, cop. 2016
- [BC] Brooks, Frederick Phillips, Jr.. The mythical man-month : essays on software engineering / Frederick P. Brooks, Jr. . - Anniversary ed., [repr. with corr.] Reading, Massachusetts : Addison-Wesley, cop. 1995
- Teruel:

<http://psfunizar7.unizar.es/br13/egAsignaturas.php?codigo=30226&Identificador=13597>

- [BB] Brooks, Frederick Phillips, Jr.. The mythical man-month : essays on software engineering / Frederick P. Brooks, Jr. . - Anniversary ed., [repr. with corr.] Reading, Massachusetts : Addison-Wesley, cop. 1995
- [BB] Chrissis, Mary Beth. CMMI for development : guidelines for process integration and product improvement / Mary Beth Chrissis, Mike Konrad, Sandy Shrum . 3rd ed. Upper Saddle River, New Jersey : Addison-Wesley, cop. 2011
- [BB] Humphrey, Watts S.. Introduction to the Team Software Process / Watts S. Humphrey.. Reading, Mass. [etc.] : Addison-Wesley, 1999.
- [BB] Pressman, Roger S.. Ingeniería del Software : un enfoque práctico / Roger S. Pressman ; traducción Rafael Ojeda Martín...[et al.] ; coordinación general y revisión técnica, Luis Joyanes Aguilar . - 4a. ed. en español Madrid [etc.] : McGraw-Hill, cop. 1998
- [BB] Software engineering guides / [C. Mazza...(et al.)] ; edited by Jon Fairclough . - [1st. ed.] London [etc.] : Prentice Hall, 1996