

**Información del Plan Docente**

<b>Año académico</b>	2017/18
<b>Centro académico</b>	110 - Escuela de Ingeniería y Arquitectura
<b>Titulación</b>	439 - Graduado en Ingeniería Informática
<b>Créditos</b>	6.0
<b>Curso</b>	4
<b>Periodo de impartición</b>	Indeterminado
<b>Clase de asignatura</b>	Obligatoria
<b>Módulo</b>	---

**1. Información Básica****1.1. Introducción**

Un proyecto de software comprende todas las actividades técnicas y de gestión requeridas para entregar al cliente los resultados que han sido acordados en un plazo de tiempo y con un presupuesto. El objetivo de esta asignatura es que los alumnos tengan la ocasión de ampliar su conocimiento sobre la gestión de proyectos de software de un manera eminentemente práctica, trabajando en un proyecto software de su elección gestionado con la metodología ágil Scrum.

**1.2. 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 tamaño. Puesto que la adquisición de conocimientos y competencias en esta asignatura se efectúa alrededor del desarrollo de una aplicación software de pequeño tamaño, pero desarrollada siguiendo técnicas de ingeniería del software y con un proceso de desarrollo definido, es necesario haber superado los dos primeros cursos del plan de estudios y, al menos, las asignaturas de Ingeniería del Software y Proyecto Software del quinto y sexto cuatrimestres respectivamente.

**1.3. Contexto y sentido de la asignatura en 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 tamaño. La asignatura de Ingeniería del Software ha dotado a los alumnos de los fundamentos de ingeniería que requiere el desarrollo profesional de sistemas de software de tamaño mediano o grande, y la asignatura de Proyecto Software les ha aportado el elemento integrador y de soporte que permite que el proceso de construcción de un sistema software sea realizada por un equipo de una manera controlada. Esta asignatura profundiza en las técnicas que permiten sistematizar las actividades y procesos necesarios para el desarrollo eficaz de cualquier sistema software, adaptándose a los requisitos actuales de agilidad, calidad y mejora continua.

**1.4. Actividades y fechas clave de la asignatura**

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.

**2. Resultados de aprendizaje**

## **2.1.Resultados de aprendizaje que definen la asignatura**

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

- Conoce estrategias y aproximaciones para desarrollar y gestionar los procesos vinculados a la obtención de un contrato de un proyecto software. Esto incluye aproximaciones para la definición de objetivos y entregables de un proyecto, estimación del coste del proyecto y la elaboración de un presupuesto para el mismo.
- Conoce las bases para abordar la gestión y optimización del equipo humano que integra el proyecto. Esto incluye estrategias para la formación del equipo, herramientas para optimizar su funcionamiento (basadas principalmente en dinámicas de grupo), y aproximaciones a la identificación, caracterización y asignación de roles dentro de un proyecto.
- Conoce el concepto de riesgo dentro de un proyecto software. Así como mecanismos para la planificación de su gestión. Estos mecanismos comprenden, entre otros elementos, la identificación, valoración, selección y definición de estrategias de mitigación.
- Conoce las bases conceptuales y diversas técnicas para el seguimiento, revisión y evaluación de un proyecto software.
- Conoce procedimientos para llevar a cabo el cierre de un proyecto software, las implicaciones que esto tiene, la medición y evaluación de un proyecto, así como el aprovechamiento de la información generada estos procesos.
- Conoce las problemáticas asociadas al mantenimiento del software.
- Sabe gestionar y organizar las actividades involucradas en el mantenimiento del software.
- Conoce los aspectos éticos, sociales, legales y económicos intrínsecos al desarrollo de un proyecto software de empresa, generales y específicos al ámbito de uno o varios dominios de aplicación.
- Conoce una infraestructura de procesos y herramientas necesarios para desarrollar un proyecto software, basado en las buenas prácticas de ingeniería de software disponible en un entorno empresarial de factoría de software.
- Pone en práctica los conocimientos adquiridos en las asignaturas de la intensificación de Ingeniería de Software en un proyecto concreto desarrollado en equipo: requisitos, análisis, diseño, pruebas (verificación y validación), gestión de proyectos.

## **2.2.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.Objetivos y competencias**

### **3.1.Objetivos**

Al finalizar el curso los estudiantes habrán realizado un proyecto software en equipo, de temática a su elección, gestionado con la metodología ágil Scrum. Durante el trabajo evaluarán y tomarán decisiones técnicas, equilibrarán el esfuerzo de desarrollo con el de la creación de la documentación que aporte mayor valor y analizarán su proceso de gestión con el objetivo de mejorarlo.

### **3.2.Competencias**

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

Afrontar con éxito los siguientes desempeños transversales:

- CT1. Capacidad para concebir, diseñar y desarrollar proyectos de Ingeniería.
- CT2. Capacidad para planificar, presupuestar, organizar, dirigir y controlar tareas, personas y recursos.

- CT4. Capacidad para resolver problemas y tomar decisiones con iniciativa, creatividad y razonamiento crítico.
- CT7. Capacidad para analizar y valorar el impacto social y medioambiental de las soluciones técnicas actuando con ética, responsabilidad profesional y compromiso social.
- CT8. Capacidad para trabajar en un grupo multidisciplinar y en un entorno multilingüe.

Afrontar con éxito los siguientes desempeños relacionados con la Ingeniería Informática en general:

- CGC2. Capacidad para 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.
- CGC3. Capacidad para 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.
- CGC4. Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.
- CGC8. Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

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

- CEIS2. 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.
- CEIS3. Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.
- CEIS5. Capacidad de identificar, evaluar y gestionar los riesgos potenciales asociados que pudieran presentarse.
- CEIS6. Capacidad para diseñar soluciones apropiadas en uno o más dominios de aplicación utilizando métodos de la ingeniería del software que integren aspectos éticos, sociales, legales y económicos.

## 4. Evaluación

### 4.1. Tipo de pruebas, criterios de evaluación y niveles de exigencia

La evaluación tendrá dos ejercicios:

1. Entrega de resultados del proyecto (75%): el ejercicio consiste en la entrega de resultados (memoria técnica, código fuente y otros) que reflejen la labor de los estudiantes en un proyecto de desarrollo de software que siga la metodología ágil Scrum. El proyecto comprenderá la construcción de una aplicación informática de pequeñas dimensiones, de temática libre, en la que se 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 (todos los resultados de aprendizaje).
2. Preguntas escritas sobre conceptos aprendidos en teoría y problemas (25%) (resultados de aprendizaje 1, 2, 3, 4, 5, 10, 16, 17).

Para superar la asignatura será necesario alcanzar al menos 4 sobre 10 en cada uno de los dos ejercicios por separado, y que la suma ponderada de ambos sea al menos de 5 sobre 10. En caso de no alcanzar en alguno de los dos ejercicios una nota de 4 puntos sobre 10, la calificación global en la asignatura será la mínima entre 4 y el resultado de la suma ponderada de ambas partes.

## 5. Metodología, actividades, programa y recursos

### 5.1. 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. 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 clase magistrales, así como enfatizar el desarrollo de todas las competencias vinculadas al trabajo en equipo colaborativo.
5. 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.

### 5.2. Actividades de aprendizaje

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 dentro del marco establecido por el trabajo en grupo
3. Adicionalmente, se solicitará un trabajo en equipo, bajo la tutela de los profesores, donde se abordará un proyecto de software de pequeño tamaño.

### 5.3. Programa

El programa de la asignatura se compone de los siguientes grandes bloques:

1. Gestión ágil con Scrum: principios ágiles, sprints, historias de usuario, la pila del producto, estimación, velocidad, planificación y deuda técnica
2. Técnicas de gestión de proyectos: integración, alcance, tiempo, coste, calidad, personas y riesgos

### 5.4. Planificación y calendario

La organización docente prevista es la siguiente:

- Clases magistrales (2 horas por semana)
- Resolución de problemas y casos (1 hora por semana)

Los horarios de todas las actividades se anunciarán con suficiente antelación a través de las webs del centro y de la asignatura.

Los proyectos propuestos serán entregados en las fechas que se señalen con antelación y la fecha del examen se ajustará al calendario académico siguiendo la normativa.

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

- 45 horas, aproximadamente, de actividades presenciales (sesiones en el aula teóricas, de problemas, de presentación y análisis de casos de estudio y de seguimiento-evaluación)
- 15 horas de estudio y evaluación
- 90 horas de trabajo en grupo

### 5.5. Bibliografía y recursos recomendados

[BB: Bibliografía básica / BC: Bibliografía complementaria]

- [BB] Rubin, Kenneth S. Essential Scrum : a practical guide to the most popular agile process / Kenneth S. Rubin .

## 30248 - Gestión de proyecto software

7th print. Upper Saddle River (New Jersey) : Addison-Wesley, imp. 2015

- [BC] Larman C. Practices for Scaling Lean & Agile Development: Large, Multisite, and Offshore Product Development with Large-Scale Scrum / Larman C, Vodde B. - 1st edition Addison-Wesley Professional, 2010