



## Grado en Ingeniería Informática itinerario Ingeniería del Software

### DATOS DE LA ASIGNATURA

**Nombre:**

Calidad, Medición y Estimación de Productos y Procesos Software

**Denominación en inglés:**

Quality, Measurement and Estimation of Software Products and Processes

**Código:**

606010222

**Carácter:**

Obligatorio

**Horas:**

	Totales	Presenciales	No presenciales
Trabajo estimado:	150	60	90

**Créditos:**

Grupos grandes	Grupos reducidos			
	Aula estándar	Laboratorio	Prácticas de campo	Aula de informática
3	0	0	0	3

**Departamentos:**

Tecnologías de la Información

**Áreas de Conocimiento:**

Lenguaje y Sistemas Informáticos

**Curso:**

4º - Cuarto

**Cuatrimestre:**

Primer cuatrimestre

### DATOS DE LOS PROFESORES

**Nombre:**

\*De la Villa Cordero, Manuel

**E-Mail:**

manuel.villa@dti.uhu.es

**Teléfono:**

959217639

**Despacho:**

Edif. Torreumbria, Dcho. 41

\*Profesor coordinador de la asignatura

## 1. Descripción de contenidos

### 1.1. Breve descripción (en castellano):

Para poder asegurar que un proceso (o un producto) presentan un determinado grado de calidad y mejorarlo es necesario disponer y aplicar un conjunto de indicadores cuantitativos y cualitativos. La medición del software nos permite controlar qué es lo que ocurre en los proyectos y predecir su esfuerzo y duración; así como mejorar la calidad de los productos software. La medición cuenta con una larga tradición y constituye una disciplina fundamental en cualquier ingeniería, y la Ingeniería del Software no debe ser una excepción, si bien hay que tener siempre presente las peculiaridades que diferencian al software de otros productos. La necesidad y motivación por medir se ha incrementado notablemente con la preocupación de las organizaciones por alcanzar mayores niveles de madurez y las consiguientes certificaciones basadas en modelos y normas como ISO 9000, ISO 15504 o CMMI.

También las metodologías Ágiles, de creciente difusión, necesitan de indicadores y métricas de productividad y progreso del proyecto específicos que se abordarán convenientemente.

### 1.2. Breve descripción (en inglés):

In order to ensure that a process (or product) have a certain degree of quality and improve it, is necessary to have and implement a set of quantitative and qualitative indicators. The software measurement allows us to control what happens in projects and predict their effort and duration, as well as improving the software products.

The measurement has a long tradition and is essential in any engineering discipline. Software Engineering should not be an exception, although we must bear in mind the peculiarities that differentiate software from other products. The need and motivation to gain predictability has increased significantly with the concern of organizations to achieve higher levels of maturity and subsequent model-based certifications such as ISO 9000, ISO 15504 and CMMI.

Agile methodologies, that are increasingly widespread, also need specific indicators, productivity metrics and project progress metrics that will be addressed accordingly.

## 2. Situación de la asignatura

### 2.1. Contexto dentro de la titulación:

Una vez el alumno ha recibido una formación básica de la Ingeniería del Software como disciplina, en especial de la importancia del modelo de proceso seguido, ahora se hará hincapié en la importancia de la predictibilidad, de ser capaces de estimar y predecir costes, duración y calidad tanto del proceso de desarrollo como del producto software elaborado así como de establecer herramientas que nos caractericen y permitan controlar un proyecto.

### 2.2. Recomendaciones:

Se hace más que necesaria la adquisición de las competencias y destrezas impartidas en la asignatura de 'Principios y Fundamentos de la Ingeniería del Software'

## 3. Objetivos (Expresados como resultados del aprendizaje):

Podemos considerar los siguientes objetivos de carácter general:

- Introducir al alumno en el conocimiento de las métricas
- Hacerle comprender su importancia dentro de una ingeniería
- Medir la capacidad de los procesos de desarrollo de una compañía
- Estimar de manera fiable costes, duraciones y esfuerzos en proyectos concretos
- Medir objetivamente la calidad de los productos generados.
- Definir procesos de mejora basados en indicadores

## 4. Competencias a adquirir por los estudiantes

### 4.1. Competencias específicas:

- **CE3-IS:** Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles
- **CE5-IS:** Capacidad de identificar, evaluar y gestionar los riesgos potenciales asociados que pudieran presentarse
- **CE6-IS:** 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.

#### 4.2. Competencias básicas, generales o transversales:

- **CB2:** Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio
- **CG0:** Capacidad de análisis y síntesis: Encontrar, analizar, criticar (razonamiento crítico), relacionar, estructurar y sintetizar información proveniente de diversas fuentes, así como integrar ideas y conocimientos.
- **G03:** Capacidad para la resolución de problemas
- **G04:** Capacidad para tomar decisiones basadas en criterios objetivos (datos experimentales, científicos o de simulación disponibles) así como capacidad de argumentar y justificar lógicamente dichas decisiones, sabiendo aceptar otros puntos de vista
- **G05:** Capacidad de trabajo en equipo.
- **G07:** Motivación por la calidad y la mejora continua, actuando con rigor, responsabilidad y ética profesional.
- **T01:** Uso y dominio de una segunda lengua
- **T02:** Conocimiento y perfeccionamiento en el ámbito de las TIC's

## 5. Actividades Formativas y Metodologías Docentes

### 5.1. Actividades formativas:

- Sesiones de Teoría sobre los contenidos del Programa.
- Sesiones de Resolución de Problemas.
- Sesiones Prácticas en Laboratorios Especializados o en Aulas de Informática.
- Actividades Académicamente Dirigidas por el Profesorado: seminarios, conferencias, desarrollo de trabajos, debates, tutorías colectivas, actividades de evaluación y autoevaluación.

### 5.2. Metodologías docentes:

- Clase Magistral Participativa.
- Desarrollo de Prácticas en Laboratorios Especializados o Aulas de Informática en grupos reducidos.
- Resolución de Problemas y Ejercicios Prácticos.
- Tutorías Individuales o Colectivas. Interacción directa profesorado-estudiantes.
- Planteamiento, Realización, Tutorización y Presentación de Trabajos.
- Evaluaciones y Exámenes.

### 5.3. Desarrollo y justificación:

#### **Clase magistral participativa.**

Consisten en clases magistrales donde se impartirá la base teórica de la asignatura y se expondrán ejemplos aclaratorios de la misma al grupo. Las sesiones serán de una hora y se irán intercalando con las sesiones académicas de problemas a lo largo del curso.

La metodología usada para impartir la teoría y los ejemplos aclaratorios será la exposición mediante transparencias en pizarra digital y uso de pizarra. El profesor podrá solicitar la participación activa del alumno mediante preguntas rápidas, teniendo en cuenta los alumnos que más participen a la hora de evaluar.

En la página web de la asignatura se encontrarán las transparencias y otros materiales de referencia necesarios para el seguimiento de las sesiones.

#### **Resolución de problemas y ejercicios prácticos**

Resolución en grupo de problemas y casos prácticos en los que se valorará de forma expresa, la originalidad de la solución, la calidad y la premura en la solución.

#### **Desarrollo de Prácticas en Laboratorios Especializados o Aulas de Informática en grupos reducidos**

Consisten en el estudio y aplicación de herramientas de medición en el desarrollo de sistemas software. Los alumnos dispondrán con antelación del problema a resolver y la metodología de trabajo. El trabajo podrá realizarse de forma individual o por grupos. Las sesiones serán de dos horas. La asistencia a las sesiones de laboratorio es obligatoria.

#### **Planteamiento, Realización, Tutorización y Presentación de Trabajos.**

A lo largo del curso se propondrán diversas actividades que fomentarán el trabajo autónomo del alumno así como su destreza en la toma de medidas y de decisiones en base a esas medidas. De manera extraordinaria se podrán organizar conferencias y actividades extracurriculares.

## 6. Temario desarrollado:

### Tema 1. Introducción a la medición. ¿Por qué medir y para qué?

#### Tema 2. Gestión de versiones

1. Conceptos y definiciones previas.
2. Estructura y funcionamiento de un CVS
3. Introducción a Git.

#### Tema 3. Aseguramiento de la calidad. Pruebas.

1. Introducción
2. 1. Definiciones  
2. La calidad en el software
3. Pruebas
4. 1. Estrategias pruebas de defectos: Pruebas de caja negra: partición equivalente y análisis de valores límite  
2. Pruebas de caja blanca: complejidad ciclomática y caminos independientes
5. Pruebas de Bajo nivel (Unitarias, de integración, de regresión)
6. Pruebas de Alto nivel (de Stress, de Aceptación)
7. Pruebas unitarias con JUnitTDD en breve

#### Tema 4. Metodologías y estándares de medición.

1. Goal-Question Metric (GQM)
2. Practical Software Measurement (PSM)
3. Normas IEEE 1061-1998 e ISO 15939

#### Tema 5. Métricas software en el proceso, proyecto y producto.

1. Medición del proceso. Madurez.
2. Métricas 'clásicas' de software
3. Métricas para sistemas OO

#### Tema 6. Estimación basada en tamaño y función.

1. Métodos heurísticos. Métodos paramétricos
2. Estimación con LOC (Líneas de Código)
3. Estimación con Punto-Función. FP Lite.
4. Estimación con Puntos Casos de Uso

#### Tema 7. Gestión ágil de proyectos.

1. Historias de usuario. Estimación.
2. Scrum, Lean y Kanban
3. Métricas e indicadores para seguimiento de proyectos ágiles

#### Tema 8. Implantación de un programa de medición y herramientas.

1. Obstáculos y factores de éxito.
2. Herramientas.

## 7. Bibliografía

### 7.1. Bibliografía básica:

MEDICIÓN Y ESTIMACIÓN DEL SOFTWARE: TÉCNICAS Y MÉTODOS PARA MEJORAR LA CALIDAD Y LA PRODUCTIVIDAD  
Piattini Velthuis, Mario G., (aut.)  
Ra-Ma Editorial, S.A.

### 7.2. Bibliografía complementaria:

ESTIMACION DE COSTOS Y ADMINISTRACION DE PROYECTOS DE SOFTWARE. DANDO REALISMO A LA ESTIMACION.  
Jones, Capers  
McGraw-Hill  
SOFTWARE METRICS: A RIGUROUS & PRACTICAL APPROACH. Fenton NE and Pfleeger SL, (2nd Edn), PWS, 1998  
GESTIÓN ÁGIL DE PROYECTOS SOFTWARE  
Garzás J. y otros  
Kybele Consulting

## 8. Sistemas y criterios de evaluación.

### 8.1. Sistemas de evaluación:

- Examen de teoría/problemas
- Defensa de Prácticas
- Defensa de Trabajos e Informes Escritos

### 8.2. Criterios de evaluación y calificación:

La evaluación de la asignatura consta de las siguientes partes:

**A) Examen de teoría/problemas.**

En esta sección se consideran los fundamentos teóricos junto con la realización de casos y supuestos prácticos. Con esta actividad se evalúan las competencias CE6-IS, CB2, CG0, CG7. (40% de la calificación total)

**B) Defensa de Prácticas.**

Los alumnos realizarán, en grupos reducidos o de manera individual, ejercicios planificación, estimación, medición y desarrollo de un supuesto práctico, siguiendo una metodología basada en procesos (PSP y/o TSP). Estos ejercicios se explicarán, supervisarán y se comenzarán a trabajar en las clases destinadas para ello. La asistencia a las sesiones de los ejercicios de aplicación será obligatoria. Los alumnos realizarán un conjunto de prácticas guiadas, con el fin de asimilar conceptos vistos en las clases teórico-prácticas. Con esta actividad se evalúan las competencias T01, CB2, CG3, CG4, CG5, CE3-IS y CE5-IS. (50% de la calificación total)

**C) Defensa de Trabajos e Informes Escritos.** (10% de la calificación total) Actividades de trabajo autónomo que podrán consistir en:

- Actividad de exposición y debate. Consistirá en la realización de un trabajo por parte de los alumnos que deberán exponer y debatir en clase.
- Resolución de actividades prácticas. Consistirán en la realización y entrega de ejercicios prácticos, en los que se valorará de forma expresa, la originalidad de la solución, la calidad y la premura en la solución, dichos ejercicios serán corregidos y expuestos en clase por el profesor.
- Asistencia a conferencias. Se valorará la asistencia a las conferencias planificadas durante el curso y la realización de una memoria de la misma, de acuerdo al criterio previamente establecido por el profesor. Con esta actividad se evalúan las competencias T02.

**CRITERIOS DE EVALUACIÓN Y CALIFICACIÓN**

- Examen Teórico Escrito 40%
- Evaluación de las prácticas entregadas/defendidas 50%
- Evaluación de trabajos 10%

La nota en acta del alumno se obtendrá del siguiente modo, teniendo en cuenta que las evaluaciones de los conocimientos teóricos (Parte A), prácticos (Parte B) y de los trabajos se realizarán sobre 10:

Si  $\text{Nota\_Parte\_A} \geq 5$  y  $\text{Nota\_Parte\_B} \geq 5$  Entonces  $\text{Nota\_Final} = \text{Nota\_Parte\_A} \times 0,4 + \text{Nota\_Parte\_B} \times 0,5 + \text{Nota\_trabajos} \times 0,1$

En caso contrario  $\text{Nota\_Final} = \text{Mínimo}(\text{Nota\_Parte\_A}, \text{Nota\_Parte\_B})$

De manera que estará aprobada la asignatura sólo si la  $\text{Nota\_Final}$  es  $\geq 5$ .

A los alumnos que no alcancen el aprobado en la asignatura, se les mantendrá la nota de la Parte\_A, Parte\_B hasta la última convocatoria anual, siempre y cuando su puntuación en las mismas sea  $\geq 5$  y la nota de los trabajos.

Tanto el apartado A como el B se han de superar (calificación  $>5$  sobre 10) de manera individual. Se reservará la nota de los apartados superados para posteriores convocatorias.

**9. Organización docente semanal orientativa:**

	Semanas	Grupos Grandes	Grupos Reducidos Aula Estándar	Grupos Reducidos Aula de Informática	Grupos Reducidos Laboratorio	Grupos Reducidos prácticas de campo	Pruebas y/o actividades evaluables	Contenido desarrollado
#1	2	0	2	0	0		Tema 1	
#2	2	0	2	0	0		Tema 1	
#3	2	0	2	0	0		Tema 2	
#4	2	0	2	0	0		Tema 2	
#5	2	0	2	0	0		Tema 3	
#6	2	0	2	0	0		Tema 3	
#7	2	0	2	0	0		Tema 4	
#8	2	0	2	0	0		Tema 4	
#9	2	0	2	0	0		Tema 4	
#10	2	0	2	0	0		Tema 5	
#11	2	0	2	0	0		Tema 5	
#12	2	0	2	0	0		Tema 5, 6	
#13	2	0	2	0	0		Tema 6	
#14	2	0	2	0	0		Tema 7	
#15	2	0	2	0	0		Tema 7	
	30	0	30	0	0			