

Ingeniería del Software

Programa 2014/2015

**Grado en Ingeniería Informática de Gestión y
Sistemas de Información**

Grupo: 01 Curso: 2º Cuatrimestre: 2º

La asignatura

➤ *Objetivo*

Diseñar e implementar aplicaciones, cuyos requisitos han sido previamente capturados, aplicando arquitecturas software de varios niveles y utilizando el lenguaje de modelado UML con el fin de desarrollar proyectos software siguiendo un proceso sistemático y apoyándose en herramientas que permiten mejorar la calidad del software.

➤ **6 créditos ECTS**

- ❖ Clases magistrales: 45 horas
 - ❖ 3h semanales
- ❖ Laboratorios (prácticas de ordenador): 20 horas
 - ❖ Laboratorios de 2h cada 2 semanas (semanas pares)
- ❖ Trabajo no presencial: 90 horas

El Horario y las tutorías

Lunes	Martes	Miércoles	Jueves	Viernes
			ISoft (AI1.2) 8:00-10:00	
		ISoft (AI1.2) 10:00-11:00		ISoft GL2 (AI0.1) 10:00 – 11:00
				ISoft GL2 (AI1.1) 12:00 – 14:00

➤ Mikel Larrañaga (*e-mail*: mikel.larranaga@ehu.eus)

❖ Tutorías

- Miércoles: 08:00 – 10:00h
- Jueves: 10:00 – 12:00h
- Viernes: 08:00 – 10:00 (Semanas pares)
10:00 – 12:00 (Semanas impares)

Las competencias

- ❖ Saber distinguir las diversas etapas que componen todo proceso de ingeniería del software.
- ❖ Saber entender un sistema software con orientación a objetos en el lenguaje UML.
- ❖ Saber diseñar un sistema software en una arquitectura de varios niveles a partir del análisis.
- ❖ Saber implementar un sistema a partir del diseño de la aplicación.

Transversal:

- ❖ Saber comunicar y transmitir conocimientos, habilidades y destrezas de la profesión de Ingeniero en Informática

El programa (<http://egela.ehu.eus>)

■ Unidad 1. Introducción a la Ingeniería del Software

Tras revisar la evolución de los productos hardware y software a lo largo de su historia, justifica y describe el concepto “Ingeniería del Software” presentando:

- Su motivación y el ciclo de vida del software
- Objetivos, propiedades y tecnologías de programación asociados.

■ Unidad 2. Especificación de artefactos UML

Estudio de los diferentes artefactos existentes en UML

■ Unidad 3. Arquitecturas software de varios niveles: Presentación, Lógica de Negocio y Datos

Diseño de las diferentes capas que componen un sistema software

■ Unidad 4. Diseño y programación orientados a objetos

Realización del diseño de la funcionalidad de un sistema software

■ Unidad 5. Implementación de algún producto concreto

Implementación de un sistema de software utilizando un conjunto de lenguajes y herramientas actuales

El proyecto

- En la segunda o tercera semana del cuatrimestre, se presentarán los requisitos funcionales del sistema que se deberá desarrollar a lo largo del cuatrimestre.
- El Diseño e implementación de dicho sistema requerirá la puesta en práctica de los conceptos que se abordan en la asignatura.
- El proceso de desarrollo constará de diferentes iteraciones en las que se construirán productos parciales **correctos** que se ampliarán incrementalmente hasta obtener el sistema final
 - Cubrir los requisitos funcionales establecidos
 - Adecuadamente diseñado
 - Operativo
 - Robusto

Cronograma

	Magistral	Laboratorio
1	Introducción + Arquitecturas de varios niveles	
2	Metodología de trabajo + Ejemplo	Interfaces gráficas
3	Diseño	
4		
5		
6		
7	Examen	
8	Diseño (Incluyendo herencia simple y múltiple)	GUI + Revisión estado proyecto
9		
10		
11		
12		
13		
14	Java 8	Revisión estado proyecto
15		

Evaluación

- **Convocatorias**
 - ❖ Ordinaria
 - ❖ Examen 7 semana (aprox.) (30%)
 - ❖ Examen de mayo (30%)
 - ❖ Extraordinaria (**Junio-Julio**)
- **Sistema de evaluación:**
 - ❖ Proyecto (40%)
 - ❖ Exámenes (60%)

Evaluación

➤ *Calificación final:*

- ❖ Si el alumno NO se presenta al examen, obtendrá un “No presentado”
- ❖ Si el alumno se presenta al **examen final**, la nota de prácticas en ese curso se añadirá a la que obtenga en el examen escrito, siempre y cuando en el total de la nota de examen escrito alcance un mínimo de **3,5** puntos sobre 10.
- ❖ Para calificar el proyecto se tendrá en cuenta tanto el trabajo individual como el funcionamiento global del grupo. Si algún miembro del grupo no colabora o trabaja de manera autónoma sin contar con el resto, a la nota del grupo se le aplicará un factor de corrección de 0,5. La nota definitiva, en este caso, sería **nota*0,5**
- ❖ **ACLARACIÓN!!** La nota de prácticas se guarda durante todo el curso, pero no para años siguientes.
- ❖ Existe la posibilidad de renunciar a la evaluación mixta en la **convocatoria extraordinaria** y optar al 100% de la nota de la asignatura mediante un examen final.

Evaluación

➤ *Casos excepcionales:*

- ❖ Solamente podrán obtener el 100% de la nota mediante el examen escrito los casos excepcionales justificados al comienzo del curso.
- ❖ Deben comunicarse al profesor al comienzo de la asignatura o cuando la circunstancia excepcional ocurra, si se da a lo largo de la asignatura.
- ❖ Justificación documentada adecuadamente.

Bibliografía

- *Ingeniería del Software. Un enfoque práctico. Roger S. Pressman. MacGraw-Hill, 2001. 5a Edición.*
- *Ingeniería El Proceso Unificado de Desarrollo de Software Jacobson, Booch, Rumbaugh. Editorial Addison Wesley, 1999*
- *Construcción de Software Orientado a Objetos. Bertrand Meyer. Prentice-Hall. 1998.*
- *Java 8 in Action. Raoul-Gabriel Urma, Mario Fusco, Alan Mycroft. Manning. 2014*