



Universitat de Lleida

GUÍA DOCENTE **INGENIERÍA DE SOFTWARE LIBRE**

Coordinación: GIMENO ILLA, JUAN MANUEL

Año académico 2023-24

Información general de la asignatura

Denominación	INGENIERÍA DE SOFTWARE LIBRE			
Código	102056			
Semestre de impartición	1R Q(SEMESTRE) EVALUACIÓN CONTINUADA			
Carácter	Grado/Máster	Curso	Carácter	Modalidad
	Grado en Ingeniería Informática	4	OBLIGATORIA	Presencial
	Grado en Ingeniería Informática	4	OPTATIVA	Presencial
	Máster Universitario en Ingeniería Informática		COMPLEMENTOS DE FORMACIÓN	Presencial
Número de créditos de la asignatura (ECTS)	6			
Tipo de actividad, créditos y grupos	Tipo de actividad	PRALAB		TEORIA
	Número de créditos	3		3
	Número de grupos	1		1
Coordinación	GIMENO ILLA, JUAN MANUEL			
Departamento/s	INGENIERÍA INFORMÁTICA Y DISEÑO DIGITAL			
Distribución carga docente entre la clase presencial y el trabajo autónomo del estudiante	20% Presencial 20% Virtual 60% Trabajo autónomo			
Información importante sobre tratamiento de datos	Consulte este enlace para obtener más información.			
Idioma/es de impartición	Parte oral: 40%-> Preferentemente en Catalán Material escrito y audiovisual (apuntes, vídeos, etc.): 60% -> Inglés			
Distribución de créditos	Juan Manuel Gimeno (3) Montserrat Sendín (3)			

Profesor/a (es/as)	Dirección electrónica\nprofesor/a (es/as)	Créditos impartidos por el profesorado	Horario de tutoría/lugar
GIMENO ILLA, JUAN MANUEL	juanmanuel.gimeno@udl.cat	3	Con cita previa.
SENDIN VELOSO, MONTSERRAT	montse.sendin@udl.cat	3	

Información complementaria de la asignatura

Se asumen conocimientos mínimos de Ingeniería de Software para seguir la asignatura de forma adecuada.

Objetivos académicos de la asignatura

- Conocer el concepto de Software Libre y las principales consecuencias que de él se derivan.
- Conocer cómo se organizan los proyectos libres.
- Emplear las herramientas más usadas tanto en la familia GNU como en Java.
- Disponer de una perspectiva del software libre tanto desde un punto de vista histórico como de proyectos actuales.
- Conocer los principales modelos de negocio y métodos de financiación que se han creado alrededor del software libre.
- Conocer el uso del software libre en la administración pública.
- Conocer la infraestructura de soporte al desarrollo del software libre.

Competencias

Competencias estratégicas de la UdL

- **CT2:** Adquirir un dominio significativo de una lengua extranjera, especialmente del inglés.
- **CT3:** Adquirir capacitación en el uso de las nuevas tecnologías y de las tecnologías de la información y la comunicación.

Competencias transversales

- **EPS-11:** Capacidad de comprender las necesidades del usuario/a expresadas en un lenguaje no técnico.

Competencias específicas de la titulación

- **GII-IS1:** Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del Ingeniería de Computadores usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software.
- **GII-IS3:** Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.
- **GII-IS4:** Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.
- **GII-IS6:** 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.

Contenidos fundamentales de la asignatura

Parte teórica

1. Introducción al software libre
2. Un poco de historia
3. Ingeniería del software libre: la catedral y el bazar
4. Los lenguajes de programación como proyectos de Software Libre
5. Unicode e Internacionalización de aplicaciones
6. Documentación en el Software Libre
 - 6.1 Licencias para la creación de documentación libre
 - 6.2 Herramientas de creación de documentación libre
7. Aspectos económicos
 - 7.1 Métodos de financiación
 - 7.2 Modelos de negocio
8. Proyectos de Software Libre
 - 8.1 Modelos de desarrollo de proyectos de Software Libre
 - 8.2 Estudio de casos
9. Infraestructura adicional de soporte al desarrollo de Software Libre
 - 9.1 Herramientas de comunicación
 - 9.2 Repositorios de proyectos
 - 8.3 Otros
10. El software libre y las administraciones públicas
 - 10.1 Aspectos generales
 - 10.2 Estudio de casos

Paralelamente, en las sesiones de laboratorio se presentaran:

- Herramientas de construcción de proyectos GNU (make i autotools)
- Herramientas de construcción de proyectos Java (maven)
- Herramientas de construcción de proyectos JavaScript (npm)
- Herramientas de despliegue de aplicaciones usando contenedores (docker)
- Herramientas de internacionalización de proyectos GNU (gettext, Java resource bundles)
- Herramientas de gestión de proyectos (forjas)

Ejes metodológicos de la asignatura

- **Evaluación continuada**, en la que se integra de manera natural la presentación de contenidos por parte del alumno/a y el intercambio de experiencias con el resto de la clase.
- **Trabajo en grupo.**
- Se combinan las clases teóricas con clases de laboratorio en las que practicar con las diferentes herramientas presentadas.
- El alumno/a deberá profundizar por cuenta propia en el estudio de los diferentes temas presentados, así como en la valoración de las diferentes opciones que se le proponen.
- En el desarrollo de trabajos teóricos se procurará que el alumno/a desarrolle el espíritu crítico para seleccionar y justificar razonadamente la elección realizada.
- Los trabajos culminan con una presentación oral en la que se defienden todos los criterios empleados.
- El sistema de evaluación (detallado en el apartado correspondiente) consta de: **1)** prueba escrita (el examen 1r parcial); y **2)** diversos trabajos prácticos (a desarrollar individualmente o en equipo, dependiendo de cada caso).
- En las actividades formativas intervienen estudios de casos (a desarrollar por parejas), juntamente con el desarrollo de trabajos individuales y la resolución de problemas concretos.

Plan de desarrollo de la asignatura

Semana	Teoría (GG)	Laboratorio (GG)/Sesiones de intercambio de experiencias	Trabajo autónomo
1	Historia	Introducción SL	Estudio
2	Historia	Make + Automake	Estudio y lecturas seleccionadas
3	Historia	Maven	Estudio
4	Historia	Npm	Estudio y lectura capítulo
5	Historia Cat & Baz	Docker	Estudio y lectura capítulo
6	Unicode	Gettext + bundles	Estudio y preparación presentación
7	Lenguajes de Programación	Presentaciones	Estudio y preparación presentación
8	Presentaciones	Presentaciones	Estudio y proyecto i18n
9	Primer parcial		
10	Documentación en PL	Presentación herramienta escogida	Estudio, proyecto i18n y lectura capítulo
11	Aspectos economicos		Estudio y despliegue wiki escogida
12	Proyectos de SL		Estudio, lectura capítulo y desarrollo del caso de estudio sobre soporte wiki escogido
13	Proyectos de SL		Estudio y desarrollo del caso de estudio sobre soporte wiki escogido
14	Infraestructura de soporte al SL	Presentación herramienta escogida	Estudio, lectura capítulo y desarrollo del caso de estudio sobre proyecto PL escogido

Semana	Teoría (GG)	Laboratorio (GG)/Sesiones de intercambio de experiencias	Trabajo autónomo
15	SL y AAPP		Estudio y desarrollo del caso de estudio escogido
16	Presentación y defensa actividades en grupo		
17	Presentación y defensa actividades en grupo		
18	Tutorías		
19	Recuperación		

Sistema de evaluación

Evaluación continua

Bloque Actvd.	Descripción	Ponderación	Nota mínima	En grupo	Recuperable
BloqI-Parc1	Primer parcial Conceptos básicos	20%	No	No	No
BloqII-Actv1	Presentación de un caso de estudio	20%	No	No	No
BloqIII-Actv2	Proyecto de Internacionalización	10%	No	No	No
BloqIV-Actv3	Experiencias con el uso de herramientas de software libre	10%	No	No	No
BloqV-Actv4	Despliegue y uso de una wiki	10%	No	Si	No
BloqV-Actv5	Presentación de un caso de estudio	20%	No	Si	No
BloqVI-Actv6	Aplicación práctica del PL en las AAPP. Estudio de caso	10%	No	Sí	No

Nota final = $0,20 * \text{BloqI-Parc1} + 0,20 * \text{BloqII-Actv1} + 0,10 * \text{BloqIII-Actv2} + 0,10 * \text{BloqIV-Actv3} + 0,10 * \text{BloqV-Actv4} + 0,20 * \text{BloqV-Actv5} + 0,10 * \text{BloqVI-Actv6}$

- La asignatura se aprueba si la nota final es igual o superior a 5

Otras consideraciones:

- Tipología del examen parcial: fijación de conceptos
- Para todas las actividades evaluables: Entregas programadas, fechas no prorrogables.
- En las actividades que incluyen una presentación oral, el estudiante será evaluado individualmente tanto por el profesor como por el resto de sus compañeros (co-evaluación).

Evaluación alternativa

Un único examen sobre todo el contenido de la asignatura, es decir, de ambos parciales.

NOTA: La evaluación se realizará siguiendo la [Normativa de Evaluación y Calificación de la Docencia en los Grados i Másteres de la Udl](#) (pendiente de traducción)

Bibliografía y recursos de información

Bibliografía básica

- Jesús González Barahona, Joaquín Seoane Pascual, Gregorio Robles, [Introducción al Software Libre](#). Grupo de Sistemas y Comunicaciones, ESCET, Universidad Rey Juan Carlos de Madrid. 2ª Ed. (2007)
- Karl Fogel, [Producing Open Source Software](#). Published under creative commons, (2013)
- Sam Williams (Second edition revisions by Richard M. Stallman). [Free as in Freedom \(2.0\): Richard Stallman and the Free Software Revolution](#). Published under GNU free documentation license, (2010)

Bibliografía complementaria

- John Calcote, [AutoTools. A practitioner's guide to GNU Autoconf, automake, and libtool](#). No Starch Press (2010)
- Steven Weber, [The success of open source](#). Harvard University Press (2004).