



Universitat de Lleida

GUIA DOCENT
**ENGINYERIA DEL
PROGRAMARI**

Coordinació: GIMENO ILLA, JUAN MANUEL

Any acadèmic 2016-17

Informació general de l'assignatura

Denominació	ENGINYERIA DEL PROGRAMARI			
Codi	102018			
Semestre d'impartició	1R Q(SEMESTRE) AVALUACIÓ CONTINUADA			
Caràcter	Grau/Màster	Curs	Caràcter	Modalitat
	Doble Titulació: Grau en Enginyeria Informàtica i Grau en Administració i Direcció d'Empreses	3	OBLIGATÒRIA	Presencial
	Grau en Enginyeria Informàtica	3	OBLIGATÒRIA	Presencial
	Màster Universitari en Enginyeria Informàtica		COMPLEMENTES DE FORMACIÓ	Presencial
Nombre de crèdits ECTS	6			
Grups	1GG,2GM			
Crèdits teòrics	3			
Crèdits pràctics	3			
Coordinació	GIMENO ILLA, JUAN MANUEL			
Departament/s	INFORMATICA I ENGINYERIA INDUSTRIAL			
Distribució càrrega docent entre la classe presencial i el treball autònom de l'estudiant	6 ECTS = (10 h de Classe presencial + 15 h de Treball autònom de l'estudiant) x 6 = 150 h de treball 40% Presencial (equivalent a 60 h) 60% Treball autònom (equivalent a 90 h)			
Informació important sobre tractament de dades	Consulteu aquest enllaç per a més informació.			
Idioma/es d'impartició	Preferentment en Català (Castellà si algun estudiant mostra dificultats amb el Català).			
Distribució de crèdits	Juan Manuel Gimeno Illa 4.5 Montserrat Sendin Veloso 4.5			
Horari de tutoria/lloc	Juan Manuel Gimeno (3.20 EPS, dimecres a les 13h; altres amb cita prèvia) Montserrat Sendín (3.20 EPS; amb cita prèvia)			

Professor/a (s/es)	Adreça electrònica professor/a (s/es)	Crèdits	Horari de tutoria/lloc
GIMENO ILLA, JUAN MANUEL	jmgimeno@diei.udl.cat	4,5	Amb cita prèvia.
SENDÍN VELOSO, MONTSERRAT	msendin@diei.udl.cat	4,5	Amb cita prèvia.

Informació complementària de l'assignatura

Assignatura obligatòria de 3r curs (1r quadrimestre) pertanyent a la formació comú a la branca d'informàtica.
Matèria: Anàlisi i Disseny d'Aplicacions.

RECOMANACIONS: Es donen per coneguts els coneixements d'Orientació a Objectes de Programació II i Estructures de Dades.

Objectius acadèmics de l'assignatura

- Conèixer les bases conceptuals i els diferents aspectes de la disciplina, entre ells els models de procés del cicle de vida del software
- Aplicar la tècnica dels casos d'ús
- Especificar textualment les necessitats funcionals i no funcionals d'un determinat sistema software plantejat a través d'un enunciat (i/o altres inputs procedents de l'usuari)
- Desenvolupar el diagrama de classes d'un determinat sistema software seguint els principis del Modelat Orientat a Objectes
- Utilitzar una eina de modelat basada en UML
- Comprendre el concepte del codi com alguna cosa que evoluciona en el temps
- Ser capaç de programar proves unitàries bàsiques.
- Comprendre els principis fonamentals del disseny orientat a objectes.
- Reconèixer el concepte de responsabilitat com a fonamental a l'hora de plantejar un disseny orientat a objectes.

Competències

Competències transversals

- **EPS11:** Capacitat de comprendre les necessitats de l'usuari expressades en un llenguatge no tècnic.

Competències específiques

- **GII-CRI2:** Capacitat per planificar, concebre, desplegar i dirigir projectes, serveis i sistemes informàtics en tots els àmbits, liderant la seva engegada i la seva millora contínua i valorant el seu impacte econòmic i social.
- **GII-CRI12:** Coneixement i aplicació de les característiques, funcionalitats i estructura de les bases de dades, que permetin el seu adequat ús, i el disseny i l'anàlisi i implementació d'aplicacions basades en elles.
- **GII-CRI13:** Coneixement i aplicació de les eines necessàries per a l'emmagatzematge, processament i accés als Sistemes d'informació, inclosos els basats en web.
- **GII.CRI16:** Coneixement i aplicació dels principis, metodologies i cicles de vida de l'enginyeria de programari.

- **GII-CRI17:** Capacitat per dissenyar i avaluar interfícies persona computador que garanteixin l'accessibilitat i usabilitat dels sistemes, serveis i aplicacions informàtiques.

Continguts fonamentals de l'assignatura

1. Aspectes introductoris

- 1.1. Qüestions inicials al voltant de la Enginyeria del Software
- 1.2. Una mica d'història
- 1.3. Procés de desenvolupament de software
- 1.4. Models de procés de software
- 1.5. Conclusions

2. Anàlisi de requeriments

- 2.1. Especificació de requeriments
- 2.2. La tècnica dels Casos d'Ús
- 2.3. Un pas més en l'especificació: Els Diagrames de Seqüència del Sistema
- 2.4. Conclusions

3. Anàlisi del Domini

- 3.1. Diagrames de Classes de l'Anàlisi
- 3.2. Un pas més en l'anàlisi del domini: els Contractes de les operacions
- 3.3. Conclusions

4. Introducció al disseny i Proves unitàries

- 4.1. El codi com alguna cosa que varia en el temps.
- 4.2. El framework JUnit per a proves unitàries

5. Els principis SOLID

- 5.1. Principi de responsabilitat única
- 5.2. Principi Obert-Tancat
- 5.3. Principi de substitució de Liskov
- 5.4. Principi de segregació d'interfícies
- 5.5. Principi d'inversió de dependències

6. Disseny basat en responsabilitats

- 6.1. El concepte de responsabilitat

6.2. Patrons GRASP d'assignació de responsabilitats

Eixos metodològics de l'assignatura

Grups Grans: Classes Teoria (3 crèdits)

- Part teòrica: classes suportades amb transparències i/o apunts
- Part d'aplicació pràctica: es treballa sempre amb exemples. Es disposa d'una **col.lecció de problemes**, i a classe es treballa la resolució de problemes concrets. Es van proporcionant les solucions al llarg del quadrimestre

Grups Mitjans: Classes Laboratori (3 crèdits)

- Classes dirigides i seguiment personalitzat per grups de pràctiques
- Us de eines de Modelat en UML: **ArgoUML** i/o **Visual Paradigm**
- Treball continuat al voltant d'un **enunciat de pràctica**, que simularà el desenvolupament d'un projecte de software

Treball Autònom (no presencial):

- La pràctica es completarà en hores **No Presencials**
- **Es recomana** que l'alumne resolgui per compte propi els problemes de la **col.lecció de problemes**, a fi de practicar i obtenir feedback per part del professor

El **sistema d'avaluació** (detallat en l'apartat corresponent) consta de: 1) proves escrites (els 2 exàmens parcials); i 2) pràctiques (a desenvolupar individualment o en equip depenent de cada cas).

Pla de desenvolupament de l'assignatura

Setmana	Teoria (GG)	Laboratori (GM)	Treball autònom
1	Presentació Assignatura T1: Aspectes introductoris	T1: Aspectes introductoris	Estudi
2	T1: Aspectes introductoris	T1: Aspectes introductoris	Estudi
3	T2: Anàlisi de requeriments Especificació de requeriments	Aplicació Anàlisi requeriments en enunciat de pràctica	Estudi i resolució problemes (Col.lecció de problemes)
4	T2: Anàlisi de requeriments. La tècnica dels Casos d'Ús. Problemes	Ús eina modelat UML Aplicació tècnica de Casos d'Ús en enunciat de pràctica	Estudi i resolució problemes (Col.lecció de problemes)
5	T2: Anàlisi de requeriments Diagrames de Seqüència del Sistema. Problemes	Aplicació tècnica de Casos d'Ús en enunciat de pràctica	Estudi, resolució problemes (Col.lecció de problemes) i desenvolupament pràctica
6	T3: Anàlisi del Domini Tècnica de Modelat Orientat a Objectes	Ús eina modelat UML Aplicació DSS en enunciat de pràctica	Estudi, resolució problemes (Col.lecció de problemes) i desenvolupament pràctica
7	T3: Anàlisi del Domini Problemes	Ús eina modelat UML Aplicació Model del Domini en enunciat de pràctica	Estudi, resolució problemes (Col.lecció de problemes) i desenvolupament pràctica

8	T3: Anàlisi del Domini Contractes de les operacions	Aplicació Contractes en enunciat de pràctica	Estudi, resolució problemes (Col.lecció de problemes) i desenvolupament pràctica
9	Primer parcial		
10	T4: Introducció al disseny Concepte de proves	Ús de Git	Estudi
11	T4: Junit Objectes substituïts	Ús de Git	Estudi i resolució de problemes
12	T5: Principis SOLID Intro, OCP & LSP	Problemes simples testing	Estudi i pràctica testing
13	T5: Principis SOLID SRP, ISP & DIP	T6: Patrons GRASP Concepte de responsabilitat	Estudi, resolució de problemes i pràctica testing
14	T6: Patrons GRASP Expert, creador, baix acoblament	Testing amb substitucions	Estudi, resolució de problemes i pràctica testing
15	T6: Patrons GRASP Alta cohesió, controlador	Testing amb substitucions	Estudi, resolució de problemes i pràctica testing
16	Setmana per al segon parcial		
17	Setmana per al segon parcial		
18	Tutories		
19	Recuperació		

Sistema d'avaluació

Activd.	Descripció	Ponderació	Nota mínima	En grup	Presencial	Obligatòria	Recuperable
Parc1	Primer parcial Conceptes bàsics	25%	3,0	No	Sí	Sí	Sí
Parc2	Segon parcial Conceptes bàsics	25%	3,0	No	Sí	Sí	Sí
Actv1	Anàlisi de Requeriments	15%	No	Sí	No	Sí	No
Actv2	Model del Domini i Contractes	15%	No	Sí	No	Sí	No
Actv3	Proves unitàries	20%	No	No	No	Sí	No

Nota final = $0,25 * \text{Parc1} + 0,25 * \text{Parc2} + 0,15 * \text{Actv1} + 0,15 * \text{Actv2} + 0,20 * \text{Actv3}$

- L'assignatura s'aprova si la **nota final** és superior a **5** i s'arriba a les notes mínimes als parcials

Altres consideracions:

- Tipologia dels exàmens parcials: fixació de conceptes i resolució de problemes
- Per totes les activitats avaluables: Entregues programades, dates no prorrogables

Bibliografia i recursos d'informació

Bibliografia bàsica

- C. Larman, Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. Prentice-Hall, 2005 (3ª ed.)
- Robert C. Martin: Agile Software Development: Principles, Patterns, and Practices, Prentice-Hall, 2002.

Bibliografia complementària

- G. Kotonya, I. Sommerville: Requirements Engineering: Processes and Techniques. Wiley, 1998
- P.Tahchiev et al.: Junit in Action (2nd edition). Manning, 2011.