

Enginyeria del Software I

Codi	Tipus	Curs/semestre	Crèdits ECTS
21290	<ul style="list-style-type: none"> <li>• Troncal</li> <li>• Semestral</li> </ul>	3er curs / 5è semestre	5

Professors:

<i>Nom</i>	<i>Dpt/Unitat</i>	<i>Despatx</i>	<i>Adreça e-mail</i>	<i>Telèfon</i>
Daniel Ponsa	Ciències de la Computació	S/246	daniel@cvc.uab.es	93 581 2440
Diego Cheda German Ros	Ciències de la Computació	S/247	dcheda@cvc.uab.es german.ros@cvc.uab.es	

Objectius:

Un primer objectiu és donar una visió global i ordenada del procés de desenvolupament del software que no sigui simplement el de la programació --que constitueix, només, una fase dins de tot el procés de la enginyeria del software.

El temari es centra en les dues primeres etapes del cicle de vida del software: anàlisi i disseny. Per cada una de aquestes dues grans parts del curs estudiarem en detall la metodologia orientada a l'objecte. Ens centrarem bàsicament en l'estudi de UML com a llenguatge de modelat i RUP com a paradigma de desenvolupament. Tot això es complementarà amb l'estudi de casos pràctics.

L'objectiu és que l'alumne sigui capaç d'utilitzar aquestes metodologies per tal de realitzar l'anàlisi i el disseny de problemes reals, que li permetrà abordar la creació de software de manera rigorosa.

Coneixements

- Entendre què és el software, quins son els objectius que ens plantegem al desenvolupar un software i la problemàtica del associada al seu desenvolupament.
- Conèixer els principals paradigmes de desenvolupament del software.
- Aprendre què és l'anàlisi de requeriments i perquè és important. Abstracció i partició
- Aprendre què és el disseny. Cóm aconseguir un bon disseny. Modularitat efectiva.
- Introducció a l'UML: notació i construcció dels diagrames.
- Introducció a un procés de desenvolupament del software: RUP.

### Habilitats

- A partir d'un problema obert enunciat en llenguatge natural, saber analitzar que ens esta demanant el client: crear un document d'especificació.
- A partir de la informació obtinguda en el document d'especificació modelar-ho a partir dels diferents diagrames de UML.
- Ser capaç de veure que a l'hora de modelar hi ha punts en que no disposem de tota la informació i per tant s'ha de tornar a analitzar el problema: Iterar.
- Portar a terme aquestes activitats en un context orientat a objecte: els models (diagrames) que obtenim son diferents punts de vista del mateix problema on cada un modela algun característica de les metodologies orientades a objecte.
- Consciència de que un Enginyer en Informàtica no és només un programador i de que desenvolupar software és molt més que programar.
- Saber buscar i seleccionar la informació necessària per assolir els objectius plantejats.

### Competències

- Treball en equip.
- Capacitat d'anàlisi i síntesi.
- Capacitat de resolució de problemes.
- Raonament crític.
- Motivació per la qualitat.
- Capacitat de resolució de problemes.
- Comunicació oral i escrita.
- Capacitat d'organització i planificació.
- Aprenentatge autònom.
- Lideratge

### **Capacitats prèvies:**

Es requereix que l'alumne tingui coneixements i capacitats de:

- Coneixement de Programació.
- Coneixement dels principis bàsics dels llenguatges orientats a objecte.
- Estructures de dades bàsiques.

Aquests conceptes corresponen a continguts de les assignatures:

- Algorismes i programació
- Llenguatges de programació.
- Estructures de dades.

## Continguts

### T0. Presentació de l'assignatura

- Presentació dels continguts, normativa i desenvolupament de l'assignatura

### T1. Principis de l'Enginyeria del Software.

- **Definició i objectius de l'ES.** Definició de software. Característiques del software. Aplicacions del software. Definició d'ES. Objectius de l'ES.
- **Evolució del software.** Etapes. Crisi del software: problemes i causes.
- **Procés, mètode i eina.** Definicions. Activitats en el procés de desenvolupament del software.
- **Paradigmes del desenvolupament del software.** Model lineal seqüencial vs iteratiu.

### T2. Anàlisi de Requeriments del Software.

- **Introducció.** Tipus de requeriments. Tasques a realitzar.
- **Comprensió del problema.** Tècniques de comunicació. Principis de l'anàlisi.
- **Especificació de requeriments.** Propietats desitjables Estàndards. Revisió i validació.

### T3. Disseny del Software

- **Introducció.** Procés de disseny. Disseny de dades, disseny arquitectònic, disseny de la interfície, disseny procedimental. Principis (objectius) del disseny.
- **Conceptes del disseny.** Abstracció. Modularitat. Refinament.
- **Disseny modular efectiu.** Independència funcional. Cohesió Acoblament. Heurístiques per a un disseny modular efectiu.

### T4. El Llenguatge Unificat de Modelat (UML)

- **Introducció.** Visió general d'UML. Notació.
- **Model estructural.** Classes. Relacions. Mecanismes comuns. Diagrames. Diagrames de classes.
- **Model de comportament.** Interaccions. Casos d'ús. Diagrames de casos d'ús. Diagrames d'interacció. Diagrames d'activitats. Diagrames d'estats.
- **Model arquitectònic.** Diagrames de components. Diagrames de desplegament.
- **Exemples.** Anàlisi de casos d'estudi.

### T5. Processos de Desenvolupament de Software

- **Introducció.** Objectius comuns.
- **Metodologies.** Propostes populars. Esquemes tradicionals vs àgils..

### T6. El Procés Unificat de Rational (RUP)

- **Introducció.** Gestió del desenvolupament del software. Les 6 "best practices".
- **El procés unificat.** Què és RUP?
- **Estructuració del RUP.** Estructura estàtica i dinàmica.
- **Exemples.** Anàlisi de casos d'estudi.

### T7. Tècniques d'Estimació en Projectes SW

- **Introducció.** Planificació de projectes. Dificultats en la planificació de projectes SW.
- **Estimació del treball.** Unitats. Mètodes T-shirt sizing i Planning Poker.

## Metodologia docent:

Les diferents activitats que es duran a terme en l'assignatura s'organitzen de la següent manera:

### Classes de teoria:

S'exposen els conceptes bàsics de l'assignatura i s'indica com aprofundir en aquests continguts.

### Classes de problemes:

S'estenen de manera pràctica temes vistos tangencialment a les classes magistrals. Es resolen problemes i es discuteixen casos pràctics. Amb les activitats plantejades es promou la capacitat d'anàlisi i síntesi, el raonament crític, i s'entrena l'estudiant en la resolució de problemes.

En algunes sessions l'alumne haurà d'entregar els problemes realitzats al llarg de la classe, els quals es consideraran en l'avaluació continua de l'alumne. Es proposaran també problemes addicionals que l'alumne haurà de resoldre autònomament, i que s'hauran d'entregar posteriorment segons s'estableixi.

### Pràctiques:

Durant el curs es realitzarà una pràctica mitjançant el treball en grup de 4 persones (excepcionalment, per causes justificades, es considerarà constituir grups més numerosos). La pràctica consistirà en realitzar l'anàlisi i disseny d'una determinada aplicació informàtica. En el treball realitzat l'alumne haurà de demostrar el seu domini d'UML i de RUP.

El grup realitzarà la pràctica de manera autoorganitzada, i haurà d'assistir a 5 sessions de tutorització al llarg del curs per presentar el treball desenvolupat i comentar-lo amb un professor de l'assignatura. Aquestes sessions tindran una durada màxima d'una hora, i en elles es valorarà la qualitat del treball desenvolupat, la coordinació del grup, i el treball individual de cada membre del grup.

## Activitats Formatives:

Títol	Hores	Crèdits
<b>Tipus: Dirigides</b>		
Classes de Problemes	10	0,40
Classes de Teoria	27	1,08
Proves d'Avaluació	4	0,16
Pràctiques	5	0,20
<b>Tipus: Supervisades</b>		
Preparació de les Sessions de Pràctiques	25	1,00
Ressolució de Problemes Fora de l'Aula	12	0,48
<b>Tipus: Autònomes</b>		
Estudi	40	1,60
Tutories	2	0,08

## **Avaluació:**

L'avaluació de l'assignatura es durà a terme de manera contínua a partir de les evidències d'aprenentatge recollides en els següents processos:

E1. Realització de proves escrites (exàmens).

E2. Resolució i entrega d'exercicis

- a les sessions de problemes.
- autònomament fora de l'aula.

E3. Realització d'una pràctica.

### **E1: Avaluació de coneixements teòrics i pràctics**

Les evidències de l'apartat E1 es recolliran en dues proves parcials, on s'avaluaran amb una nota entre 0 i 10 (ambdós inclosos) els conceptes vistos dins la part del temari especificada a continuació.

- Examen Parcial 1 (ExP1): Avaluació del temes T1, T2, T3 i T4 \*.
- Examen Parcial 2 (ExP2): Avaluació dels temes T5, T6 i T7\*.

\*Temari referenciat segons la notació introduïda a l'apartat Continguts.

La nota final de E1 (NotaE1) es calcularà amb la següent fórmula:

$$\text{NotaE1} = (0,6 * \text{ExP1}) + (0,4 * \text{ExP2})$$

Hi haurà una prova final de l'assignatura en la que es podrà optar a millorar la nota obtinguda a les proves ExP1 i/o ExP2. En aquest cas, la nota final que s'utilitzarà en el càlcul de la nota de E1 serà l'obtinguda en aquesta prova final, amb independència de la nota que s'hagués tret prèviament.

En totes les activitats avaluades per determinar la NotaE1 s'establirà un dia de revisió de la correcció.

### **E2: Avaluació d'exercicis proposats al llarg del curs**

En aquest apartat seran objecte d'avaluació:

- exercicis plantejats en alguna de les sessions de problemes, que s'hauran d'entregar resolts al finalitzar la sessió.
- exercicis proposats al llarg del curs, que s'hauran de resoldre de manera autònoma fora de l'aula i entregar dins un període de temps establert.

La valoració de les diferents entregues es farà amb una nota entre 0 i 10 (ambdós inclosos).

Les entregues no realitzades es puntuaran amb una nota de 0.

La nota final de E2 (NotaE2) es calcularà fent el promig de les notes obtingudes en totes les activitats proposades.

### **E3: Avaluació de la pràctica**

La nota de pràctiques es determinarà a partir de les entregues realitzades pels grups de pràctiques en les sessions de tutorització. Per avaluar la pràctica es valorarà la qualitat del treball desenvolupat pel grup, la coordinació amb que ha treballat el grup, i el treball individual realitzat per cada alumne.

La valoració de la pràctica es farà amb una nota entre 0 i 10 (ambdós inclosos).

Si es detecta que una pràctica es el resultat d'una còpia es suspendrà l'assignatura.

La no entrega de la pràctica es puntuarà amb una nota de 0.

La nota final de E3 (NotaE3) correspondrà a la nota obtinguda a la pràctica.

Els alumnes repetidors amb les pràctiques aprovades l'any passat tenen l'opció de convalidar les pràctiques, mantenint la nota que varen treure.

### **Nota final de l'assignatura**

Per superar l'assignatura, cal obtenir un mínim de 5 a l'aplicar la següent expressió:

$$\text{NotaE1} + (0,1 * \text{NotaE2}).$$

Si no es supera aquest requeriment de mínims, la nota final de l'alumne serà un suspès. En cas contrari, la nota es calcularà segons la següent fórmula:

$$\text{NotaFinal} = (0,6 * \text{NotaE1}) + (0,1 * \text{NotaE2}) + (0,3 * \text{NotaE3})$$

Si la NotaFinal resultant és inferior a 5 es suspendrà l'assignatura.

Atès que l'assignatura es segueix en avaluació contínua, **no es programarà convocatòria de juny.**

### **Bibliografia bàsica:**

- R. Pressman, **Ingeniería del software, un enfoque práctico**. Mc Grah-Hill, 5a ed., 2002.
- I. Somerville, **Software engineering**. 6a Ed., Addison-Wesley, 2002.
- T. Pender, **UML Weekend Crash Course**, Wiley Publishing, Inc., 2002
- T. Quatrani. **Visual modeling with Rational Rose 2000 and UML**. Addison-Wesley, 2000.
- P. Krutchen. **The Rational Unified Process. An introduction**. Addison-Wesley, 2000.
- G. Booch, J. Rumbaugh, I. Jacobson. **El lenguaje unificado de modelado**. Addison-Wesley, 1999.
- I. Jacobson, G. Booch, J. Rumbaugh. **El proceso unificado de desarrollo de software**. Addison-Wesley, 1999.
- J. Rumbaugh, I. Jacobson, G. Booch. **El lenguaje unificado de modelado: manual de referencia**. Addison-Wesley, 1999.

### **Bibliografia complementària:**

- E. Yourdon, **Análisis estructurado moderno**. Prentice-Hall, 1993.
- J. Lladós, X. Roca. **Problemas d'enginyeria del software I**. Servei de Publicacions UAB, 1995.
- M.G. Piattini, J.A. Calvo-Manzano, J. Cervera, L. Fernández. **Análisis y diseño detallado de aplicaciones informáticas de gestión**. Ra-Ma, 1996.
- G. Booch, **Object oriented design with applications**. The Benjamin/Cummings Publishing Company, 1990.
- Yourdon Inc., **Yourdon systems method model-driven systems sevelopment**. Prentice Hall, 1993.
- P. Coad, E. Yourdon, **Object-oriented analysis**. Yourdon Press, 1991.
- P. Coad, E. Yourdon, **Object-oriented design**. Yourdon Press, 1991.