

Titulació	Tipus	Curs	Semestre
2502441 Enginyeria Informàtica	OB	3	1
2502441 Enginyeria Informàtica	OT	4	1

Professor de contacte

Nom: Xavier Otazu Porter

Correu electrònic: Xavier.Otazu@uab.cat

Utilització d'idiomes a l'assignatura

Llengua vehicular majoritària: català (cat)

Grup íntegre en anglès: No

Grup íntegre en català: Sí

Grup íntegre en espanyol: No

Prerequisits

No és obligatori, però es recomana cursar en paral·lel l'assignatura "Requisits del Software"

Objectius

En aquesta assignatura s'introdueixen els conceptes bàsics per planificar i realitzar proves del software així com estimacions de la seva qualitat.

Competències

Enginyeria Informàtica

- Adquirir hàbits de pensament
- Capacitat per a desenvolupar, mantenir i avaluar serveis i sistemes de software que satisfacin tots els requisits de l'usuari i es comportin de manera fiable i eficient, siguin assequibles de desenvolupar i mantenir i compleixin les normes de qualitat, aplicant les teories, principis, mètodes i pràctiques de la Enginyeria del software
- Capacitat per concebre, desenvolupar i mantenir sistemes, serveis i aplicacions informàtiques emprant els mètodes de l'enginyeria del software com a instrument per a assegurar-ne la qualitat.

Resultats d'aprenentatge

1. Aplicar les tècniques formals de verificació del compliment de requisits de les aplicacions software.
2. Aplicar mètodes formals de revisió del disseny del software.
3. Avaluar els sistemes software assegurant que satisfacin tots els requisits de l'usuari i es comporten de manera fiable i eficient, siguin assequibles de desenvolupar i mantenir i compleixin normes de qualitat, aplicant les teories, principis, mètodes i pràctiques de la Enginyeria del Software.
4. Desenvolupar la capacitat d'anàlisi, síntesi i prospectiva.
5. Planificar i documentar les proves del disseny del software en casos pràctics.

Continguts

L'ordre temporal dels següents continguts al llarg del curs no ha de seguir necessàriament aquest ordre.

1. Introducció a la prova del software

Objectiu de la prova. Tipus de proves: caixa blanca, caixa negra, de validació, regressió, càrrega etc. Nivells de prova: unitat, integració, sistema, acceptació. Planificació i documentació de les proves. Prova envers qualitat del software.

Donar una visió general de la part de test de la assignatura. Què és qualitat? Les proves només en són una part.

2. Proves de caixa negra

Classes d'equivalència. Anàlisi de valors límit. Taules de decisió. Pairwise testing. Transició d'estats. Disseny de casos de prova a partir de casos d'us. Exemple ATM. Prova a partir de casos d'us.

3. Proves de caixa blanca

Control de flux (complexitat ciclomàtica). Cadenes definició-us. Prova de cobertura envers prova de control de flux.

4. Prova exploratòria

Proves senzilles i ràpides. Interès pràctic i efectivitat.

5. Proves d'unitat

Objectiu i metodologia. Perquè fer proves d'unitat. Proves d'unitat i test-driven development. Junit. "Mock" objects.

6. Planificació i documentació de la prova

Estàndard de documentació de la prova IEEE Std 82-2008. Anàlisi del risc. Mastertestplan. Exemples de pla de proves i de documentació.

7. Automatització de les proves

Creació de proves "automàtiques" (eines de record and replay de la interacció de l'usuari per afegir-hi tot tipus de controls o tests).

8. Garantiment de la qualitat del software

Introducció. Revisions formals, mètriques i estàndards de desenvolupament.

9. Revisions tècniques formals

Motivació. Funcionament. Video CMU i discussió. Exercici de RTF a classe: inspecció de codi font.

10. Mètriques del software

Metodologia

Aquesta assignatura té un equilibri entre la part teòrica i la part pràctica. No és una assignatura totalment ABP ja que els conceptes teòrics són importants, i la part pràctica es realitzarà sobretot en laboratoris tancats sobre un projecte de software proporcionat pel professor.

Classes de teoria: Són classes de tipus magistral on s'introdueixen els conceptes teòrics de l'assignatura. Tot i ser classes de teoria, una bona part de la sessió (aproximadament la meitat) es destinarà a solucionar en grup senzills problemes plantejats pel professor.

Seminaris: Són classes on es plantejaran casos pràctics que requeriran l'aplicació dels conceptes explicats a

les classes de teoria. Al final de la classe de seminaris es plantejarà algun problema que l'alumne haurà de solucionar de forma autònoma i que entregará voluntariament al començar la següent sessió de seminaris.

Laboratoris: Són sessions en les que l'alumne solucionarà una serie de problemes sobre un problema real d'una certa complexitat. Són sessions en laboratori tancat. Abans d'aquestes sessions, l'alumne haurà de realitzar una serie de tasques de forma autònoma que seran entregades al començar la sessió de laboratoris.

Activitats formatives

Títol	Hores	ECTS	Resultats d'aprenentatge
Tipus: Dirigides			
Classes de teoria	22	0,88	1, 2, 3, 4, 5
Tipus: Supervisades			
Laboratoris	12	0,48	1, 2, 3, 4
Seminaris	12	0,48	1, 2, 3, 4, 5
Tipus: Autònomes			
Estudi proves d'avaluació	60	2,4	1, 2, 3, 4, 5
Preparació laboratoris	24	0,96	4, 5
Preparació seminaris	14	0,56	1, 2, 3, 4

Avaluació

La nota final de l'assignatura ve donada per la fórmula:

$$\text{Nota} = 0.5 \cdot \text{NInd} + 0.5 \cdot \text{NGrup},$$

amb

$$\text{NInd} = \text{NTeo}$$

$$\text{NGrup} = 0.8 \cdot \text{NLab} + 0.2 \cdot \text{NSem},$$

on NInd és la nota d'avaluació individual, NGrup és la nota d'avaluació grupal, NTeo és la nota de Teoria, NLab la nota de laboratoris, i NSem és la nota mitja dels exercicis voluntaris entregats al començar cada sessió de seminaris. Cadascuna de les notes NTeo i NLab han de ser superiors a 5, en cas contrari l'assignatura està suspesa i la qualificació final de l'assignatura serà la qualificació mínima de NInd i NGrup.

La nota NTeo és

$$\text{NTeo} = 0.5 \cdot \text{NTeo1} + 0.5 \cdot \text{NTeo2}$$

on NTeo1 i NTeo2 són dos activitats d'avaluació que es realitzaran al llarg del curs en horaris de sessions de teoria. El valor de NTeo només es calcularà en cas que NTeo1 i NTeo2 siguin ambdues superiors a 4. En cas que no sigui així, es considera que el valor de NTeo és el valor mínim de NTeo1 i NTeo2. Si NTeo no és superior a 5 després de realitzar les dues proves NTeo1 i NTeo2, l'alumne es podrà tornar a avaluar de NTeo durant les proves d'avaluació final. En les proves d'avaluació final, la nota màxima de NTeo es saturarà en un valor de 6.

La nota NLab serà la mitja de les diferents sessions de laboratori, és a dir

$$NLab = (NLab1 + NLab2 + NLab3 + NLab4 + NLab5 + NLab6) / 6$$

No hi ha segona convocatòria per a la nota NLab ni per cap de les notes NLab(i). El valor de NLab només es calcularà en cas que totes i cadascuna de NLab(i) siguin superiors a 1. En cas contrari NLab serà el valor mínim d'entre tots els valors NLab(i).

Es convalidarà la nota NLab d'aquells alumnes que hagin cursat l'assignatura en un màxim de dos cursos anteriors. No es convalidarà cap altra nota.

Un alumne obtindrà un "No Avaluable" com a qualificació final de l'assignatura únicament en el cas que no obtingui cap qualificació en cap de les variables que apareixen en les fórmules matemàtiques anteriors. Si un alumne presenta algun treball per avaluar o realitza alguna de les activitats d'avaluació que es contemplen en les fórmules anteriors, ja no podrà obtenir un "No Avaluable".

Les dates d'avaluació continuada i lliurament de treballs es publicaran al campus virtual (CV) i poden estar subjectes a canvis de programació per motius d'adaptació a possibles incidències. Sempre s'informarà al CV sobre aquests canvis ja que s'entén que el CV és el mecanisme habitual d'intercanvi d'informació entre professor i estudiants.

Sense perjudici d'altres mesures disciplinàries que s'estimin oportunes, i d'acord amb la normativa acadèmica vigent, les irregularitats comeses per un estudiant que puguin conduir a una variació de la qualificació es qualificaran amb un zero (0). Per exemple, plagiar, copiar, deixar copiar, ..., una activitat d'avaluació, implicarà suspendre aquesta activitat d'avaluació amb un zero (0). Les activitats d'avaluació qualificades d'aquesta forma i per aquest procediment no seran recuperables. Si és necessari superar qualsevol d'aquestes activitats d'avaluació per aprovar l'assignatura, aquesta assignatura quedarà suspesa directament amb una qualificació de zero (0), sense oportunitat de recuperar-la en el mateix curs.

Activitats d'avaluació

Títol	Pes	Hores	ECTS	Resultats d'aprenentatge
Avaluació continuada grupal Seminaris 1	0.1	1	0,04	1, 3, 4
Avaluació continuada grupal Seminaris 2	0.1	1	0,04	2, 4, 5
Avaluació continuada individual Teoria 1	0.15	2	0,08	3, 4
Avaluació continuada individual Teoria 2	0.15	2	0,08	1, 2, 4, 5
Avaluació Laboratoris	0.4	0	0	1, 2, 3, 4, 5
Exercicis de Seminaris	0.1	0	0	1, 2, 3, 4, 5

Bibliografia

Entre [] els temes que cobreixen.

[1] A Software Testing Primer: an Introduction to Software Testing. Nick Jenkins. <http://www.nickjenkins.net>

Lectura obligada!

[1,2,3,5,6] Manage Software Testing. Peter Farrell-Vinay. Auerbach Publications, 2008.
[Esta bé la part ded ocumentació i planning, però la de tècniques de prova no gaire potser.](#)

[1,3,4] A practitioner's guide to software testing. Lee Copeland, 2004.

[Llibret molt senzill i breu \(potser massa\), enfoc pràctic.](#)

[4] How to break software: a practical guide to testing. James A. Whittaker. Addison Wesley, 2003.
[El llibre sobre prova exploratòria.](#)

[5] Pragmatic Unit Testing in Java with Junit. Andy Hunt, Dave Thomas. The Pragmatic Programmer, 2003.

[6] IEEE Standard for Software and System Test Documentation IEEE Std82-2008. IEEE Computer Society, 2008.

[6] Systematic Software Testing by. Rick D. Craig, Stefan P. Jaskiel. ArtechHouse, 2002.
[El millor llibre que he trobat sobre planificació de proves. Inclou una explicació de l'estandard IEEE de documentació.](#)

[7] Llibre EUIS, veure a dalt.

[8] Why the Vasa sank: 10 problems and some antidotes for software projects. Richard E. Fairley, Mary Jane Willshire. IEEE Software, març-abril 2003.

[Lectura molt interessant i divertida.](#)

[9] Video "Scenes of software inspections. Video dramatizations for the classroom." L. Deimel. Software Engineering Institute, CMU/SEI-91-EM-5-0, 1991.

Thinking in Java, 3rd edition. B. Eckel. Versió electrònica a <http://www.planetpdf.com/> . Versió impresa en castellà de Pearson Educación, 2002.

[Llibre de referència per programació en Java \(pràctica\).](#)