

Laboratori Integrat de Software

Codi: 102788

Crèdits: 9

Titulació	Tipus	Curs	Semestre
2502441 Enginyeria Informàtica	OB	3	2
2502441 Enginyeria Informàtica	OT	4	2

Professor de contacte

Nom: Antonio Manuel López Peña

Correu electrònic: AntonioManuel.Lopez@uab.cat

Utilització d'idiomes a l'assignatura

Llengua vehicular majoritària: espanyol (spa)

Grup íntegre en anglès: No

Grup íntegre en català: No

Grup íntegre en espanyol: No

Prerequisits

Aquesta assignatura té un caràcter molt pràctic, però es basa en els fonaments teòrics impartits a d'altres assignatures de la menció en Enginyeria del Software. Per tant, si es vol cursar aquesta assignatura amb èxit, és necessari haver cursat prèviament:

- (1) Requisits del software,
- (2) Disseny de software,
- (3) Gestió i Administració de Bases de Dades,
- (4) Test i Qualitat del Software,
- (5) Gestió del Desenvolupament de Software (aquesta es pot cursar paral·lelament, no és necessari haver-la cursat prèviament).

Objectius

Aquesta assignatura vol proporcionar una experiència útil als estudiants respecte el que es poden trobar a l'hora de desenvolupar projectes software professionalment. Per tant, des del punt de vista de l'estudiant, es tracta de desenvolupar un projecte software complet i relativament llarg. A més a més, el procés de desenvolupament serà tan o més important que el resultat final (producte software). Per tant, el software s'haurà de desenvolupar de la forma més professional possible, en particular, aplicant les millors pràctiques i treballant en equip.

Així, els objectius de l'assignatura són:

- 1. Treballar en un equip relativament gran, compost de sub-equips amb diferents responsabilitats.
- 2. Aplicar els coneixements teòrics que constitueixen les millors pràctiques del desenvolupament de software.
- 3. Desenvolupar un software complet, és a dir, anant des del paper en blanc fins a una aplicació, amb la seva documentació interna i externa, que satisfà els requisits d'un client.
- 4. Que el desenvolupament d'aquest software també serveixi per adquirir experiència en entorns actuals.

L'aplicació feta, i la manera en que s'ha fet, hauria de poder servir com a "carta de presentació" curricular.

Competències

Enginyeria Informàtica

- Adquirir hàbits de pensament.
- Adquirir hàbits de treball personal.
- Capacitat de solucionar problemes d'integració en funció de les estratègies, estàndards i tecnologies disponibles.
- Capacitat d'identificar, avaluar i gestionar els riscos potencials que puguin presentar-se.
- Capacitat per dissenyar, desenvolupar, avaluar i assegurar l'accessibilitat, l'ergonomia, la usabilitat i la seguretat dels sistemes, serveis i aplicacions informàtiques, així com de la informació que gestionen.
- Capacitat per resoldre problemes amb iniciativa, presa de decisions, autonomia i creativitat. Capacitat per saber comunicar i transmetre els coneixements, habilitats i destreses de la professió d'enginyer tècnic en informàtica.
- Treballar en equip.

Resultats d'aprenentatge

1. Adaptar-se a situacions imprevistes.
2. Assumir i respectar el rol dels diversos membres de l'equip, així com els diferents nivells de dependència de l'equip.
3. Avaluar de manera crítica el treball dut a terme.
4. Definir i gestionar la documentació que es genera durant el desenvolupament d'una aplicació software.
5. Desenvolupar el pensament sistèmic.
6. Dissenyar una arquitectura d'un sistema informàtic basat en components.
7. Dissenyar una arquitectura que permeti solucionar de manera òptima el problema especificat, tenint en compte els riscos associats.
8. Gestionar el temps i els recursos disponibles. Treballar de manera organitzada.
9. Identificar, gestionar i resoldre conflictes.
10. Planificar la integració de les diferents components desenvolupades en el procés de codificació.
11. Prendre decisions pròpies.
12. Resoldre problemes d'integració del software amb iniciativa i autonomia.
13. Saber comunicar i transmetre coneixements, habilitats i destreses relatives a la integració del software.
14. Seleccionar i utilitzar les eines CASE adequades a cada fase de desenvolupament del software.
15. Treballar cooperativament.

Continguts

En aquesta assignatura no busquem afegir més contingut teòric al que ja es desenvolupa a les altres assignatures de la menció d'Enginyeria del Software. És a dir, ja s'assumeixen els coneixements teòrics de la captura de requisits, el disseny i codificació orientat a objecte, la gestió i administració de bases de dades, el test i qualitat del software, i la gestió del desenvolupament. Es tracta de posar tot això en pràctica per desenvolupar una aplicació concreta. Per tant, com a continguts nous hi ha dos aspectes:

1. El treball en equip: rols i responsabilitats d'un equip software, reunions, presentacions, etc.
2. Mètodes àgils de desenvolupament de software.

Metodologia

Aquesta assignatura s'enfoca a desenvolupar una aplicació software de la forma més realista possible. Per tant, es fa un aprenentatge basat en un cas pràctic. De fet hi haurà un cas pràctic per a cada equip de pràctiques. Un equip estarà format per un conjunt d'entre 6 i 8 estudiants. D'una banda això vol dir que hi haurà molt de treball autònom, però, d'altra banda, també hi haurà molt treball d'equip. Així, la tipologia de les classes donarà suport al treball d'equip i al desenvolupament de software basat en un procés àgil. En particular, tindrem:

Classes de teoria. Estan destinades a complementar els aspectes no coberts per les altres assignatures de la menció d'Enginyeria del Software. En particular el treball d'equip i el desenvolupament de software basat en un procés àgil.

Reunions de projecte. Qualsevol equip que desenvolupi una aplicació determinada, estarà dividit en sub-equips amb diferents responsabilitats (p.e., caps de projecte, responsables de proves, responsables de disseny i codificació, etc.). Aquests sub-equips s'han de reunir i discutir l'estat del projecte des dels diferents punts de vista.

Classes de presentació. Cada equip ha de fer presentacions públiques regulars sobre l'estat del seu projecte. També presentacions particulars al professor. Aquestes dues modalitats s'aniran alternant al llarg de les setmanes.

Activitats formatives

Títol	Hores	ECTS	Resultats d'aprenentatge
Tipus: Dirigides			
Classes de teoria	15	0,6	
Presentacions	16	0,64	13
Reunions de projecte	40	1,6	1, 2, 3, 4, 7, 9, 10, 14, 15
Tipus: Autònomes			
Estudi pels exàmens individuals	16	0,64	
Tasques assignades a la pràctica	134	5,36	1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 14, 15

Avaluació

L'avaluació es basarà en les notes següents:

1) Desenvolupament de l'aplicació (DA): 80%. Cal destacar que no només es valora el resultat final, és a dir, l'aplicació en si mateixa, sinó que també es valora, i molt, el procés de desenvolupament d'aquesta. Per tant, tindrà molta importància l'avaluació continuada del treball fet a les reunions de projecte i, especialment, a l'estat del desenvolupament en fites preestablertes. En general, la major part d'aquesta nota serà comuna a tot l'equip, però s'establiran mecanismes per garantir que tots els membres de l'equip realment mereixen la nota. Això es farà a partir d'avaluacions entre companys, que també serviran per acabar de configurar el total de la nota DA. Si es detecta algun cas en que un estudiant no hagi fet la seva feina, l'estudiant involucrat pot rebre una nota inferior, fins i tot, donat que estem davant una avaluació continuada, l'estudiant directament pot suspendre l'assignatura (nota de zero).

2) Presentacions d'equip (PE): 10%. A més a més del material requerit per fer l'avaluació tipus DA, hi haurà presentacions públiques (és clar, acotades als estudiants de l'assignatura) regulars del treball (setmanals o quinzenals). Aquestes presentacions seran avaluades com a tal, i poden ser enteses com a una prova d'equip, per tant, la nota serà per tot l'equip. Respecte a la implicació dels membres de l'equip, s'aplica el mateix criteri que al cas DA.

3) Exàmens individuals (EI): 10%. Es faran exàmens individuals per tal d'avaluar els coneixements teòric-pràctics que cada estudiant assoleix. En particular hi ha dues proves parcials d'aquest tipus, EI1 i EI2.

Per tal d'aprovar l'assignatura, s'han d'aprovar per separat tots aquests apartats. És a dir, $DA \geq 5$ i $PE \geq 5$ i $EI1 \geq 5$ i $EI2 \geq 5$, on DA, PE, EI1 i EI2 són notes sobre 10. Si això es compleix, llavors la nota final, NF, es calcula com:

[Oportunitat 1] $NF = 0.8 DA + 0.1 PE + 0.05 EI1 + 0.05 EI2 + Extra$.

Si $EI1 < 5$ o $EI2 < 5$, l'estudiant pot tenir una segona oportunitat d'aprovar l'assignatura mitjançant un examen individual de recuperació, però, sempre que es compleixin certes condicions. En particular, per poder fer aquest examen s'exigeix que $DA \geq 5$ i $PE \geq 5$. Llavors, si ER és la nota de l'examen de repesca, la nota final de l'assignatura la calcularem com:

[Oportunitat 2] $NF = \min(6, 0.4 DA + 0.6 ER)$.

Si $DA < 5$ o $PE < 5$ l'estudiant rebrà una nota final (NF) de entre 0 i 4.5 (suspens) depenent del cas. Aquest estudiant no tindria "segona oportunitat".

Noteu, que la nota màxima que pot rebre un estudiant en "segona oportunitat" és un 6.

S'entén que els estudiants "No Avaluables" només són aquells que no s'han sotmès a cap activitat d'avaluació.

El nombre de MH (Matrícula d'Honor) que es pot concedir és proporcional al nombre d'estudiants matriculats. Diguem que es poden atorgar "N" MH. Per poder obtenir MH l'alumne ha de complir $NF > 9.5$ al moment de l'Oportunitat 1. A partir d'aquí, si hi ha més de N candidats, el professor examinarà la trajectòria dels estudiants candidats a MH i seleccionarà els N. Per tant, és el professor qui avalua les evidències que consideri més oportunes (treball continuat, rellevància dins l'equip, NF, etc.) per prendre la decisió final.

Les aplicacions que es desenvolupen són proposades pels propis estudiants de forma individual i surten d'un procés de selecció basat en presentacions i votacions. Per tant, durant aquesta primera fase, les competències transversals T01 i T02 són molt importants. De fet, l'alumne que proposa una aplicació tal que passa tot el procés (és a dir, esdevé una aplicació que es portarà a terme per un equip), és nomenat cap d'equip i rep 1 punt "Extra" (veure NF en Oportunitat 1). Parlant de competències transversals, noteu que tota l'assignatura està inherentment lligada a T03. Però, DA i PE quantifiquen aquesta competència en particular. A més a més, el cap d'equip pot distribuir 1 punt "Extra" (veure NF en Oportunitat 1) entre els membres de l'equip amb una contribució més significativa vista des de dins de l'equip.

Cal destacar també que els repetidors no rebran cap tracte especial, han de fer l'assignatura com la resta d'estudiants.

Sense perjudici d'altres mesures disciplinàries que s'estimin oportunes, i d'acord amb la normativa acadèmica vigent, es qualificaran amb un zero les irregularitats comeses per l'estudiant que puguin conduir a una variació de la qualificació d'un acte d'avaluació. Per tant, plagiar, copiar o deixar copiar una pràctica o qualsevol altra activitat d'avaluació implicarà suspendre-la amb un zero i no es podrà recuperar en el mateix curs acadèmic. Si aquesta activitat té una nota mínima associada, aleshores l'assignatura quedarà suspesa.

Les dates d'avaluació continuada i lliurament de treballs es publicaran al campus virtual i poden estar subjectes a canvis de programació per motius d'adaptació a possibles incidències. Sempre s'informarà al campus virtual sobre aquests possibles canvis ja que aquesta és la plataforma d'intercanvi d'informació entre professors i estudiants.

Activitats d'avaluació

Títol	Pes	Hores	ECTS	Resultats d'aprenentatge
Desenvolupament de l'aplicació (DA)	0.8	0	0	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
Exàmens individuals (EI1 i EI2)	0.1	4	0,16	4, 6, 7, 10, 12, 13, 14
Presentacions d'Equip (PE)	0.1	0	0	13

Bibliografia

A la bibliografia pròpia de la resta de les assignatures de la menció d'Enginyeria del Software, s'ha d'afegir:

- [1] Practical tips for Software-intensive Student Projects, 3rd Edition, D.C. Rajapakse.
- [2] Agile Software Development: Principles, Patterns and Practices, R.C. Martin.
- [3] Agile Software Development: The Cooperative Game, 2nd Edition, A. Cockburn.
- [4] Crystal Clear: A Human-Powered Methodology for Small Teams, A. Cockburn.
- [5] Agile Software Development with Scrum, K. Schwaber, M. Beedle.
- [6] Agile Project Management with Scrum, K. Schwaber.