



UNA EXPERIÈNCIA D'APRENENTATGE BASAT EN PROBLEMES EN ENGINYERIA DEL SOFTWARE

Autor: Joan Serrat Gual

Departament: Ciències de la Computació

Centre: Escola Tècnica Superior d'Enginyeries

E-mail: joan.serrat@uab.es

Coautor: Antonio M. López Peña

RESUM DE L'EXPERIÈNCIA:

El departament de Ciències de la Computació imparteix des de fa força anys l'assignatura d'enginyeria del software a les titulacions superior i tècnica d'enginyeria informàtica. Fins el curs passat, la metodologia docent era la "tradicional": classes magistrals de teoria, classes de problemes i pràctiques de laboratori, amb una avaluació en que un examen escrit té el major pes. El nostre convenciment que l'estudiant aprenia molt menys del que era possible i necessari, més les idees aportades per un dels cursos de formació del IDES, ens ha portat a experimentar durant el curs 2005-06 una adaptació de l'enfocament d'aprenentatge basat en resolució de problemes. En aquesta comunicació exposem breument els continguts de la assignatura, els motius d'aquest canvi, el nou mètode docent emprat i la valoració (globalment molt positiva) que en fem tant els professors de la assignatura com els estudiants.

Especificar l'àmbit d'aplicació :

A totes dues titulacions d'informàtica de la UAB, tècnica i superior, als campus de Sabadell i Bellaterra, hi ha dues assignatures trimestrals d'enginyeria del software (ES1 i ES2). Aquesta comunicació es refereix únicament a la segona, ES2.

PARAULES CLAU:

Aprenentatge basat en problemes i projectes, enginyeria del software.



1. INTRODUCCIÓ

Fins que l'estudiant d'enginyeria informàtica no arriba a les assignatures d'enginyeria del software, la idea que s'ha construït del desenvolupament de software és que simplement cal conèixer bé un llenguatge de programació i començar a codificar. En realitat, la programació només n'és una petita part, necessària però d'una importància relativa. L'enginyeria del software és la disciplina que estudia com aconseguir desenvolupar software de qualitat en un temps i a un cost raonables. El mot *qualitat* té moltes acepcions en software: manca d'errors, acompliment de les especificacions, mantenibilitat, facilitat d'us i d'aprenentatge... El *com* fa referència tant al procés de desenvolupament —quines fases té, què fer en cada una, com ordenar-les temporalment— com a les tècniques emprades en cada una: com especificar els requeriments del software ("què ha de fer"), com dissenyar-ne l'arquitectura ("fer-ne el planell"), programar en un llenguatge ("construir segons el planell"), gestionar els continus canvis que es produeixen durant el desenvolupament i de quina manera fer proves que detectin diferents tipus d'errors. Al nostre parer doncs, és evident que aquesta és una part fonamental en la formació d'un enginyer informàtic.

2. MOTIUS DEL CANVI

Fins el curs passat, la metodologia docent era, podríem dir, la clàssica. Hi havia classes de teoria, problemes i pràctiques de laboratori amb el professor (2, 1 i 1.5 hores per setmana, respectivament). Durant les classes de teoria, exposàvem un tema diferent per setmana. La pràctica consistia en acabar una aplicació que es donava començada als estudiants, atenent al poc temps de pràctiques de laboratori disponible (que calia complementar amb 10 o 20 hores més sense professor per poder fer quelcom de mida mínimament raonable). Els estudiants feien les pràctiques en grups de 3 o 4. Alguns problemes detectats amb aquest enfocament eren que :

- l'assistència dels estudiants a les classes de teoria i problemes anava minvant durant el semestre. Hem de dir que no calia "prendre apunts" perquè tot el material era accessible a la pàgina web de l'assignatura. També, molts estudiants treballen o estan matriculats d'un gran nombre d'assignatures. Però el fet és que un 30% dels matriculats ja no assistia a la primera classe de presentació. I a les darreres potser només hi acabava assistint un 20%.
- Durant les classes de teoria l'actitud dels estudiants era molt passiva. A més, durant tot el trimestre, quasi cap estudiant consultava als professors durant les hores de tutories, si bé esporàdicament aprofitaven altres moments.
- La majoria d'estudiants estudiaven a darrera hora, dos dies abans de l'examen, segons ells mateixos reconeixien. Únicament assimilaven realment aquelles parts necessàries per poder fer la pràctica.
- Els exàmens demostraven que els estudiants entenien poc els conceptes de teoria, o bé no els sabien aplicar o fins i tot explicar. Per això els exàmens de tipus test eren els preferits: era més fàcil reconèixer la solució (a preguntes simples) que sintetitzar-la i raonar-la en un qüestionari no test. Resoldre petits problemes també suposava una dificultat gran.
- La pràctica, per tal que els estudiants la poguessin acabar en les 10 o 15 hores del laboratori més unes 10 hores addicionals, va arribar a ser molt dirigida : els



III Jornada d'Innovació Docent 2006

enunciats explicaven com s'havien de resoldre. Tot i les facilitats afegides curs darrera curs, aquest problema no es resolvia sinó que, al contrari, s'agreujava : a més facilitats, menys esforç per part dels estudiants !

En resum, l'estudiant no estudiava (regularment), no anava a classes (de teoria i problemes), segurament no entenia (però no preguntava), cada cop demanava més facilitats a les pràctiques (que cada cop havien de ser més senzilles). En conseqüència, aprenia poc. Per la nostra part, els professors havíem d'anar adaptant el nivell d'exigència dels exàmens i pràctiques a aquesta realitat.

Els detonants del canvi radical en la metodologia docent per trencar aquesta situació, decebedora per a nosaltres i un mal servei envers l'estudiant, han estat diversos. La aplicació del pla pilot DURSI a la titulació d'enginyeria informàtica a la ETSE ha fet que els estudiants fossin tolerants a canvis importants en les assignatures. També el suport institucional en qüestions com la supressió dels exàmens de setembre. L'assistència durant l'estiu del 2005 a un curs d'aprenentatge basat en problemes organitzat per l'IDES i, finalment, el coneixement d'experiències similars en altres assignatures de la titulació (Lògica, Visió per Computador, Gràfics 2) ens van decidir a fer un canvi.

3. NOUS OBJECTIUS

El primer pas ha estat la redefinició dels objectius de la assignatura en termes de coneixements i habilitats. L'enginyeria del software, per la seva mateixa definició, inclou un gran nombre de temes. Per nosaltres docents, molts d'ells són imprescindibles i els voldríem incloure en el temari. No només això, sinó que aquesta disciplina evoluciona amb una enorme rapidesa: el que fa cinc o deu anys eren temes de recerca emergent ara són part de qualsevol curs d'enginyeria del software en totes les universitats (per exemple, els patrons de disseny orientat a objecte).

No obstant, com a mínim en aquesta assignatura, l'estudiant no aprèn allò que escolta en una classe magistral, sinó el que aplica per a resoldre un problema, el que l'obliga a pensar, a raonar. D'aquí la nostra decisió de reduir de manera dràstica el nombre de temes, seleccionant els 4 o 5 més importants segons la seva aplicabilitat professional.

Pel que fa a les habilitats, n'hem considerat tres. La primera és saber aplicar els coneixements anteriors. Concretament, donada una breu especificació del que ha de fer la aplicació, l'estudiant ha de saber:

- fer-ne un disseny orientat a objecte fàcilment extensible i resistent a futurs canvis previsibles en les especificacions
- implementar (programar) aquest disseny i fer-ne proves d'unitat segons les especificacions a nivell de disseny per contracte de cada classe
- dissenyar i implementar una interfície gràfica d'usuari d'acord amb uns principis generals de disseny
- dissenyar i fer proves de validació de la aplicació sencera
- justificar la solució a la que han arribat i comparar-la amb altres d'alternatives



III Jornada d'Innovació Docent 2006

- desenvolupar-la en equip, treballat en paral·lel amb els altres membres del seu grup, cosa que implica repartir-se tasques sinó també gestionar els canvis en les parts (arxius) que desenvolupen (modifiquen) en comú

La segona, la capacitat de treball en equip. Això és quelcom molt important en enginyeria del software, donat que els projectes de software quasi mai es desenvolupen isoladament. Per això, hem fet que els estudiants s'organitzin en grups de 3 o 4 membres i se'ls demana que, al llarg del curs, confeccionin ells mateixos llistes de tasques, se les distribueixin. Les tasques poden ser tant cercar informació que no es troba en els apunts subministrats, estudiar temes de "teoria" per a resoldre algun problema, dissenyar o implementar parts de la aplicació, confeccionar un informe etc.

La tercera, i que ha suposat el canvi més important, ha estat la capacitat de l'estudiant de dirigir el propi aprenentatge. Dit en termes més planers, "saber espavilar-se" sol a l'hora de resoldre els problemes que se li plantegen, sense haver de dependre del professor. Concretament, hem volgut que l'estudiant aprengué a:

- identificar els coneixements que no es tenen i es necessiten
- cercar, analitzar i seleccionar en llibres, revistes, web, o consultant a experts...
- fer-ne una síntesi
- comunicar-los efectivament als altres membres del seu grup

Per tal que l'estudiant adquirís els coneixements i les habilitats associats a ells, hem centrat la assignatura en el desenvolupament d'una aplicació de software, petita però completa. Sense adquirir els coneixements, és impossible construir la aplicació, si més no segons els requeriments, la forma de treball i la qualitat demanats.

Hem buscat doncs un problema en un domini ja conegut pels estudiants perquè s'explica a una assignatura de primer. Es tracta de fer un CAD de circuits lògics, com el de la figura 1. En principi no sembla un problema que hagi de motivar els estudiants més enllà de permetre'ls aprovar la assignatura. En canvi, els ha motivat força, superant les nostres expectatives, per les següents raons:

- El fet d'abordar, potser per primer cop en la carrera, el desenvolupament d'una aplicació completa, no trivial, i simplement a partir d'unes breus especificacions, constitueix per a ells un repte.
- El disseny i implementació de la interfície gràfica d'usuari és un component molt atractiu, també una novetat per a força estudiants. Fins al punt de construir la majoria de grups interfícies gràfiques molt elaborades per ells mateixos, sense cap ajut per part del professor.
- Si bé l'aparença externa de la aplicació és senzilla, quan examinem la seva arquitectura és força complexa. L'estudiant veu llavors que no es tracta d'una aplicació "de joguina", a la mesura d'unes poques hores de pràctiques.
- Es un problema que admet moltes solucions de disseny i implementació, que afavoreix la discussió d'alternatives dins del grup.

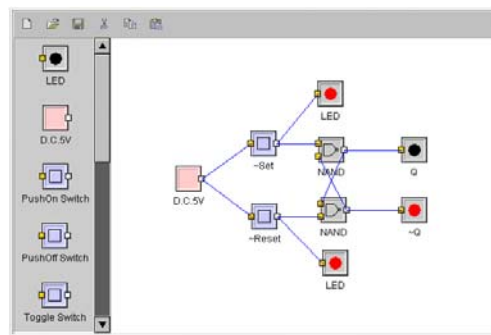


Figura 1: Exemple de CAD de circuits

4. IMPLEMENTACIÓ

Aquest, és clar, ha estat l'aspecte més difícil del canvi de metodologia docent: *com* assolir els objectius de la secció anterior? Hem fet una adaptació de la metodologia d'aprenentatge basat en problemes, perquè és la que millor respon a les necessitats de la secció anterior. Diem adaptació perquè no hem pas eliminat totalment les classes magistrals de teoria, ni tampoc deixem de subministrar als estudiants alguns dels materials que ja teníem preparats de cursos anteriors (bàsicament transparències, capítols seleccionats d'alguns llibres, articles i adreces web). Deixar que l'estudiant hagués de "descobrir" per sí mateix els conceptes teòrics ens va semblar innecessari i inviable.

Ara bé, les classes de teoria serveixen únicament per a fer una breu introducció, una per tema, presentant els conceptes bàsics i algun exemple. Així doncs, l'estudiant segueix havent de cercar i triar els coneixements que necessita per resoldre el problema plantejat, però ja dins d'un context. Per altra cantó, l'entorn de desenvolupament de software que han d'utilitzar els estudiants és completament nou per a ells i s'hi han de familiaritzar pel seu compte, per que els professors no els en expliquem el funcionament. No és una dificultat petita, donat que es tracta d'un ampli conjunt d'eines (Eclipse per Java, CVS, JUnit, la llibreria SWT i els *plugins* Omondo, Checkstyle, Jigloo i Coverlipse).

La teoria doncs s'ha reduït de 40 hores a 9. La resta les hem dedicat a *reunions tutoritzades* i *fites*. Les reunions tutoritzades són reunions d'una hora amb dos o tres grups simultàniament (cada un format per 3 o 4 estudiants). En tractar-se d'assignatures amb més de 150 estudiants matriculats, cada grup només té una reunió per setmana amb el seu professor. En elles, els estudiants exposen el treball fet, el discuteixen amb els professors, resolen problemes i també es coordinen (confeccionen i es distribueixen la llista de tasques). A més d'aquestes reunions, al principi de curs demanem a cada grup estableixi un dia i hora per fer una reunió no tutoritzada (és a dir, pel seu compte) a la setmana. L'experiència aviat els demostra que sense una reunió addicional com a mínim, la feina no surt endavant.

Les fites són reunions on el professor avalua el treball fet fins aleshores. L'enunciat del problema ja estableix quines són les característiques que ha de tenir la aplicació desenvolupada a cada fita, i el barem de puntuació (hi ha requeriments obligatoris, als que se n'hi afegeixen d'opcionals per tenir millor nota). El professor s'assegura que tots els membres del grup han treballat, qüestionant-los directament sobre les seves



III Jornada d'Innovació Docent 2006

aportacions. Un altre mecanisme és un formulari individual d'avaluació de cada company de grup. Consta de quatre qüestions molt simples, puntuables com a A=totalment, B=bastant, C=poc, D=gens d'acord, que cal després raonar:

- Assisteix a totes les reunions de grup (tutoritzades i no) ?
- Compleix puntualment les tasques que té assignades, arriba amb el material llegit (fa la feina) ?
- Escolta els companys i contribueix a les discussions de grup ?
- Les seves aportacions han estat importants per resoldre el problema ?
- Mira de fer entendre als companys els coneixements que té, però no d'imposar el seu punt de vista ?

Aquest qüestionari, que cal lliurar en cada fita, també serveix per detectar els grups en que hi ha conflictes, com ara persones massa dominants, que van per lliure o que no col·laboren suficientment.

Quelcom molt important per l'estudiant és conèixer la planificació temporal de la assignatura en termes de fites, reunions i classes de teoria, per que els demanem que s'organitzin per aconseguir uns objectius en cada fita i perquè l'assistència a fites i reunions és estrictament obligatòria: la no assistència a una fita o a dues reunions tutoritzades implica que la qualificació final és no presentat. Una planificació completa també permet al professor evitar lliuraments de treballs fora de termini i aplicar sense variacions els barems de notes prèviament establerts.

La qualificació final s'obté de la següent manera : 60% notes de les quatre fites, 20% notes de dos qüestionaris de teoria, 20% avaluació dels companys (sigui quina sigui, tot i que en general coincideix amb la impressió del professor). Tot això si han assistit a les quatre fites i han mancat com a màxim a una de les sis reunions tutoritzades.

Amb aquesta metodologia docent doncs, la presencialitat mínima obligatòria de l'estudiant queda reduïda a nou hores durant tot el trimestre. A més, aquesta hora la poden escollir dins la franja contínua de 9h del matí a 8 de la tarda, per tal de minimitzar les incompatibilitats amb el seu horari laboral i altres assignatures.

5. VALORACIÓ

Parlarem aquí de la valoració que han fet els estudiants. La dels docents l'exposarem a la secció de conclusions. Per conèixer la primera, a la darrera fita (l'últim dia, doncs) hem demanat als estudiants que responguessin una enquesta amb tres qüestions: opinió respecte el mètode docent, què els havia agradat i què no i quins consells donarien a un estudiant de l'any vinent. La majoria d'estudiants han coincidit en els següents comentaris :

- El nou mètode docent els ha permès aprendre més, si bé han hagut de treballar MOLT més, deixant de banda de vegades les altres assignatures



III Jornada d'Innovació Docent 2006

- La necessitat de treballar en equip "de veritat" ha estat valorada molt positivament, fins i tot en grups en els que hi varen haver conflictes al principi
- L'absència d'exàmens també ha estat ben acollida : la assignatura s'acaba el dia de la última fita
- El temps donat per a la primera fita, en comparació amb les altres, l'han trobat massa curt, tant pel volum de feina demanada com pel canvi de metodologia (espavilar-se sols) i familiaritzar-se amb algunes eines de desenvolupament
- Alguns estudiants demanen més sessions de teoria (per explicar les eines i que s'ha de fer) o més reunions tutoritzades

A continuació, una mostra dels consells donats:

- *Busca't uns bons companys ja que si no treballen et poden dur a la desesperació fins i tot a suspendre. Tingues present que les reunions tutoritzades i les fites són sagrades i que no podràs treballar [a més d'estudiar]*
- *Que no s'adormi, si treballa des del primer dia, la assignatura no és difícil*
- *Que no agafin aquesta assignatura fins que els en quedin molt poques, perquè genera un volum de treball increïble, i a més que s'oblidin de tenir un treball a mitja jornada*
- *Escollir [com a companys de grup] persones de confiança i intentar treballar una mica cada dia, sinó haurà de sacrificar més d'un cap de setmana*
- *El primer és escollir bé els integrants de grup, després no deixar la feina per l'última setmana i que miri d'estudiar-se la teoria abans de començar cada fita*
- *Que val la pena, però que ni es matriculi de molts crèdits ni treballi, sinó pot passar-ho malament*

No volem pas amagar aspectes negatius senyalats pels estudiants :

- *problemes de l'entorn, configuració de la xarxa i tots els problemes aliens a la pràctica. Més que aprendre a dissenyar software sembla que aquest curs ha servit per aprendre a espavilar-nos quan les coses van malament*
- *no em tingut en compte a la nota la càrrega de treball pels grups de dos en front els grups de quatre (somos dos y tenemos cuatro As, hemos faltado al trabajo y hemos estado noches sin dormir)*
- *pots tenir la desgràcia que et toquin companys "mantes"*

Els resultats de la avaluació es poden resumir en que la majoria de grups han obtingut la qualificació de notable o excel·lent. Són els grups que, tot i la càrrega de treball de la que es queixen, han tingut interès en realitzar les parts optatives, i no només assolir el mínim en cada fita.



6. CONCLUSIONS

La valoració que en fem els professors és molt positiva, encara que amb aspectes millorables i alguns interrogants. Positiva principalment perquè els estudiants que han seguit el curs han après molt més. I la demostració és el mateix treball que han anat fent al llarg del semestre i que els professors hem anat seguint. Han hagut d'aprendre molts conceptes pel seu compte i superar molts obstacles per si sols. També hem de dir que ens hem sentit molt més còmodes, més útils fins i tot, de cara a l'estudiant. El fet de mantenir vuit hores de reunions tutoritzades en un mateix dia no era tan pesat ni monòton com fer les dues hores de teoria matí i les mateixes dues hores de tarda, el curs passat.

L'interrogant és què succeirà quan la nostra ja no sigui la única assignatura del seu curs que apliqui aquest mètode, que pel que ens diuen els estudiants, els carrega excessivament de feina. Creiem que la solució passa per una mena de fusió amb Enginyeria del Software I i potser altres assignatures de la titulació, de manera que cada una faci servir el mateix problema per ensenyar coneixements diferents i, segurament, les mateixes habilitats de treball en grup.

Creiem que hem encertat l'enfocament de la assignatura. Segurament no és l'òptim ni extrapolable de manera general a altres assignatures, però el cas és que ara ens hi sentim molt més còmodes.