

Jenui 2013

Castellón de la Plana

Actas del
**Simposio-Taller sobre estrategias y herramientas para el
aprendizaje y la evaluación**

Castellón, 9 de julio de 2013

Organizado por:
Escuela Superior de Tecnología y Ciencias Experimentales
Universitat Jaume I de Castelló
y
Asociación de Enseñantes Universitarios de la Informática

David López
Belén Vaquerizo
(responsables académicos)

Mercedes Marqués Andrés
José Manuel Badía Contelles
Sergio Barrachina Mir
(editores)



JENUI (19es. 2013. Castelló de la Plana)

Actas del Simposio-Taller sobre estrategias y herramientas para el aprendizaje y la evaluación [Recurs electrònic] : Jenui 2013 : Castellón, 9 de julio de 2013 / Organizado por: Escuela Superior de Tecnología y Ciencias Experimentales Universitat Jaume I de Castelló y Asociación de Enseñantes Universitarios de la Informática ; David López, Belén Vaquerizo (responsables académicos) ; Mercedes Marqués Andrés, José Manuel Badía Contelles, Sergio Barrachina Mir (editores) — Castelló de la Plana : Publicacions de la Universitat Jaume I, 2013

1 recurs electrònic — (e-Treballs d'Informàtica i tecnologia ; 14)

Bibliografia

e-ISBN: 978-84-695-8091-2 DOI: 10.6035/e-TiIT.2013.14

I. Informàtica – Ensenyament – Congressos. I. López Álvarez, David. II. Vaquerizo, Belén. III. Marqués Andrés, Mercedes. IV. Badía Contelles, José Manuel. V. Barrachina Mir, Sergio. VI. Asociación de Enseñantes Universitarios de la Informática. VII. Universitat Jaume I. Escola Superior de Tecnologia i Ciències Experimentals. VIII. Universitat Jaume I. Publicacions. IX. Títol. X. Sèrie 004:37.02(063)

Edita: Publicacions de la Universitat Jaume I. Servei de Comunicació i Publicacions
Campus del Riu Sec. Edifici Rectorat i Serveis Centrals. 12071 Castelló de la Plana
<http://www.tenda.uji.es>
publicacions@uji.es

Colección: e-Treballs d'Informàtica i tecnologia, 14
Primera edició, 2013



Este texto está sujeto a una licencia **Reconocimiento – NoComercial – CompartirIgual (by-nc-sa)**: No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Comités

Responsables académicos

- David López, *Universitat Politècnica de Catalunya, BarcelonaTECH*
- Belén Vaquerizo, *Universidad de Burgos*

Comité directivo

- **Fermín Sánchez Carracedo (Presidente)**, *Universitat Politècnica de Catalunya*
- Patricia Miriam Borensztein, *Universidad de Buenos Aires (Argentina)*
- Agustín Cernuda del Río, *Universidad de Oviedo*
- J. José Escribano Otero, *Universidad Europea de Madrid*
- Alberto Gómez Mancha, *Universidad de Extremadura*
- Inés Jacob Taquet, *Universidad de Deusto*
- Joe Miró Julià, *Universitat de les Illes Balears*
- Rosana Satorre Cuerda, *Universidad de Alicante*
- Edmundo Tovar, *Representante del Capítulo español de la Sociedad de Educación de IEEE*
- Belén Vaquerizo García, *Universidad de Burgos*

Comité organizador

- Mercedes Marqués Andrés, *Universitat Jaume I de Castelló*
- José Manuel Badía Contelles, *Universitat Jaume I de Castelló*
- Sergio Barrachina Mir, *Universitat Jaume I de Castelló*
- Pablo Aibar Ausina, *Universitat Jaume I de Castelló*
- José I. Aliaga Estellés, *Universitat Jaume I de Castelló*
- Juan Carlos Amengual Argudo, *Universitat Jaume I de Castelló*
- Asunción Castaño Álvarez, *Universitat Jaume I de Castelló*
- M^a Isabel Castillo Catalán, *Universitat Jaume I de Castelló*
- Juan Carlos Fernández Fernández, *Universitat Jaume I de Castelló*
- Reyes Grangel Seguer, *Universitat Jaume I de Castelló*
- Dolores M^a Llidó Escriba, *Universitat Jaume I de Castelló*
- M^a Ángeles López Malo, *Universitat Jaume I de Castelló*
- Ester Martínez Martín, *Universitat Jaume I de Castelló*
- Rafael Mayo Gual, *Universitat Jaume I de Castelló*
- Lledó Museros Cabedo, *Universitat Jaume I de Castelló*
- Jorge Sales Gil, *Universitat Jaume I de Castelló*
- Ismael Sanz Blasco, *Universitat Jaume I de Castelló*
- Vicente Ramón Tomás López, *Universitat Jaume I de Castelló*
- Vicente Javier Traver Roig, *Universitat Jaume I de Castelló*

Presentación

El *simposio-taller previo a Jenui*, en su formato actual de presentación de trabajos y discusión, nació en 2011 con las Jenui de Sevilla. El objetivo de esta actividad, como el de las Jornadas en las que se engloba, es favorecer el contacto e intercambio de experiencias docentes entre profesores universitarios de Informática. El taller ofrece un foro adicional al propio de las Jenui: un lugar donde presentar trabajos en desarrollo o que por su naturaleza no encajan en Jenui, pero que resultan muy interesantes para discutir con los colegas, recibiendo realimentación, ideas y ánimos (que a veces hacen falta en este entorno en el que nos movemos, especialmente en los tiempos que corren).

Para favorecer el intercambio de ideas, la actividad se articula en dos partes: en la primera los autores presentan sus ponencias; en la segunda, todos los asistentes discuten las ponencias en profundidad y sacan conclusiones sobre lo expuesto y discutido. A diferencia de un simposio tradicional, se prima la discusión sobre la presentación.

El simposio-taller se realiza tradicionalmente la tarde anterior al inicio de las Jenui y, al igual que éstas, está promovido por la Asociación de Enseñantes Universitarios de Informática (AENUI). La primera edición, celebrada conjuntamente con las Jenui 2011 de Sevilla, contó con Joe Miró (Universitat de les Illes Balears) y Juan José Escribano (Universidad Europea de Madrid) como responsables académicos, presentándose 14 trabajos. La segunda edición, celebrada junto a las Jenui de Ciudad Real en 2012, tuvo a Joe Miró y David López (Universitat Politècnica de Catalunya) como responsables académicos y contó con 11 trabajos presentados.

Este año, el taller y las Jenui están organizadas por la Escuela Superior de Tecnología y Ciencias Experimentales de la Universitat Jaume I de Castelló. Tenemos 13 interesantes trabajos para discutir. Esperamos que los disfrutéis.

David López y Belén Vaquerizo
Responsables académicos

Índice

El proceso de aprendizaje en proyectos y prácticas	1
Uso del portal web EvalTICs para analizar la idoneidad de las prácticas de las asignaturas de grado	3
<i>Lenin Guillermo Lemus Zúñiga, José Miguel Montañana Aliaga, Félix Buendía García y José V. Benlloch Dualde</i>	
Integración de metodologías en Ingeniería	11
<i>Antonio Polo Márquez</i>	
Una titulación de informática utilizando proyectos integradores con PBL	19
<i>María Cruz Gaya López, Juan José Escribano Otero y María José García García</i>	
Experiencia de aprendizaje de programación basada en proyectos	27
<i>Laura Igual y Xavier Baró</i>	
Analizando los resultados académicos en una materia de programación	35
<i>Mariano J. Cabrero Canosa, David Alonso Ríos, Noelia Barreira Rodríguez, Carlos Gómez Rodríguez, Beatriz Pérez Sánchez, Noelia Sanchez Maroño, Elena Hernández Pereira, Miguel Ángel Alonso Pardo, Bertha Guijarro Berdiñas y Jesús Vilares Ferro</i>	
Ingeniería del Software I: aprendizaje basado en proyectos mediante metodologías Ágiles	43
<i>Alfredo Goñi, Jesús Ibáñez, Jon Iturrioz y José Ángel Vadillo</i>	
Métodos y herramientas para la evaluación, coordinación y motivación del alumnado	51
Mejora del rendimiento en las sesiones de laboratorio de la asignatura Informática para ingenierías	53
<i>Jorge Sales y Gabriel Recatalá</i>	
Estrategias y tecnologías de la información para la coordinación de asignaturas multidisciplinares y multi- tiáreas en el grado de Medicina en la Universitat Jaume I: experiencia con la asignatura “Recursos informáticos y documentación”	61
<i>Oscar Coltell, Ricardo Tosca, Ximo Granell, José Salvador Sánchez, Pedro Latorre y Luís Lizán</i>	
Diseño en FPGAs como Método Docente para la Asignatura de Redes en Chip	69
<i>Antoni Roca, José Maria Martínez y José Flich</i>	
Motivación del Estudiante hacia la Arquitectura del Procesador	75
<i>Crispín Gómez, María E. Gómez y Julio Sahuquillo</i>	
Barcelona ZeroG Challenge: Experimentación en gravedad cero en la clase de Informática	83
<i>Antoni Perez Poch</i>	
Dirección estratégica de (las tecnologías de la información) la asignatura	87
<i>Faraón Llorens</i>	
Evaluación de la competencia transversal de comunicación oral y escrita mediante la realización de debates	93
<i>Juan Carlos Amengual y Mercedes Marqués Andrés</i>	
Índice de autores	101

El proceso de aprendizaje en proyectos y prácticas

Uso del portal web EvalTICs para analizar la idoneidad de las prácticas de las asignaturas de grado

Lenin Lemus-Zuñiga¹, José-Miguel Montañana-Aliaga², Félix Buendia-García¹,
y José-Vicente Benlloch-Dualde¹

¹ETSINF Dept. de Ing. de sistemas
y computadores (DISCA)
Universitat Politècnica de València
Valencia 46020 Spain

{lemus, fbuendia, jbenlloch}@disca.upv.es

²Depart. de Arquitectura de Computadores
y Automática (DACYA)
Universidad Complutense de Madrid
Madrid 28040 Spain

jmontanana@fdi.ucm.es

Resumen

En este artículo se presenta una metodología que permite evaluar la idoneidad de las prácticas, realizadas en dos asignaturas de grado, con el fin de determinar si los alumnos están adquiriendo las habilidades y competencias del grado. Esta metodología es una adaptación de la metodología propuesta en el Proyecto de Innovación y Mejora Educativa EvalTICs.

Para que la aplicación de esta metodología sobrecargue el trabajo del profesorado en la menor medida posible, se puede utilizar el portal web EvalTICs o cualquier aplicación Web que permita crear bancos de preguntas, crear tests utilizando el banco de preguntas, presentar al alumno el test y corregirlo de forma automática.

Una vez realizados los test, basta con obtener medidas estadísticas descriptivas (media, desviación estándar y varianza) y aplicar un análisis de medias.

Abstract

In this paper we present a methodology for evaluating the adequacy of practical training performed in two subjects of the computer engineering degree. This evaluation is based on the acquisition of abilities and competencies by the undergraduate students.

We suggest using our *web portal EvalTICs* or any other Web application for creating question banks, tests using those banks, which allows students to answer the tests, and evaluate their answers automatically. The main reason on using any of these applications, is that they will support the teacher's work not be overloaded when applying this methodology.

The methodology only requires some simple descriptive statistics (mean, standard deviation and variance) on the test results to provide conclusions about the practical works on the course.

Palabras clave

Evaluación actividad del alumno, uso de las TICs en evaluación del alumnado, tinta digital, experiencias docentes.

1. Introducción

El trabajo realizado durante los cursos 2010-11 al 2012-13, que describimos en este artículo, viene motivado y reforzado por la declaración [1] firmada por los responsables europeos de la Educación Superior en 1999, por la que se establece la creación de un Espacio Europeo de Educación Superior (EEES). Cuyos objetivos son:

- Fomentar la movilidad de estudiantes y profesionales mediante la homologación de las titulaciones.
- Aumentar la competitividad de la universidad Europea.

Junto a esa declaración, el Plan Bolonia potencia el proyecto universitario UPV. Éste consiste en introducir al alumno en una experiencia educativa exigente que le permite alcanzar la excelencia en todas las dimensiones de la persona:

- Saber (conocimientos)
- Saber hacer (competencias)
- Saber ser (Actitudes)

Acompañado por un equipo de grandes profesionales y maestros, el alumno debe recorrer un camino de madurez personal, profesional y social combinando el estudio exigente con el aprendizaje práctico.

Las universidades europeas emprendieron un plan de reformas para la homologación de sus títulos siguiendo el mencionado Plan Bolonia. Dichas reformas están llevando a la migración a títulos de grado en las

universidades españolas, entre las cuales podemos destacar las Universidad Politécnica de Valencia (UPV) y la Universidad Complutense de Madrid (UCM) por estar ofreciendo titulaciones de grado en las que los autores de este artículo, aplicaron la metodología que se propone en la siguiente sección.

A partir del curso 2010-2011, la Universitat Politècnica de València, de aquí en adelante (UPV) se puso como objetivo primario, promover el desarrollo de innovaciones educativas que incidiesen de forma directa, inmediata, eficaz y sostenible en la mejora del aprendizaje de los estudiantes, profundizando en la renovación pedagógica que demandan los nuevos modos de aprender y enseñar por los que nos conduce el proceso de armonización con el espacio europeo de educación superior.

Para lograr este objetivo, la UPV creó el programa de Proyectos de Innovación y Mejora Educativa (PIME), el cual pretende dotar de apoyo institucional, económico y dar visibilidad a las iniciativas del PDI de la UPV en estos ámbitos, facilitando así mismo el desarrollo de los proyectos y el conocimiento de los resultados en toda la comunidad universitaria.

Durante el curso 2011-2012, miembros del grupo InnoVaTink de la UPV, propusieron el PIME “Uso de las TICs para evaluar la actividad docente del alumnado” (PIME EvalTICs)[2]

El PIME EvalTICs, tiene como objetivo motivar a profesores de diferentes centros a participar en la experiencia del uso de TICs en las asignaturas impartidas en el grado de Ingeniería Informática.

Entre los objetivos primarios del PIME se tienen:

1. Motivar al profesorado participante para utilizar tinta digital en sus clases magistrales.
2. Durante el curso 2011-2012, se analizó la idoneidad de las actividades docentes utilizadas para que los alumnos adquirieran las habilidades indicadas en las asignaturas participantes en este proyecto. La metodología propuesta fue muy bien acogida por el profesorado y alumnado.

Para el curso 2012-2013 se ha analizado la idoneidad de las prácticas realizadas en una asignatura de grado, para determinar la forma en la que contribuyen a que los alumnos adquirieran habilidades. Para ello, se ha modificado la metodología propuesta en el PIME EvalTICs, la mejora es mínima, simplemente el profesor debe aplicar un mismo test antes y después de realizar una práctica.

En el estudio han participado profesores de la UPV y la UCM, analizando la idoneidad de las prácticas realizadas en varias asignaturas. Este artículo muestra resultados obtenidos al aplicar la metodología propuesta en las asignaturas “Tecnología y organización de computadores” (GII-TOC) y “Fundamentos de los

computadores” (GII-FCO).

La asignatura GII-TOC pertenece al grado de ing. Informática ofertado por la Universidad Complutense (UCM) mientras que la asignatura GII-FCO pertenece al grado en ing. Informática ofertado en la UPV.

A continuación se describen, en la sección dos, los objetivos y la metodología empleada para su consecución. En la sección tres, se muestran resultados obtenidos y su análisis. En la sección cuatro, se comenta una comparativa de los resultados entre los resultados obtenidos en ambas universidades. Finalmente, en la última sección, se aparecen conclusiones.

2. Objetivos y metodología

A continuación se describen los objetivos que se persiguen y la metodología utilizada para alcanzar dichos objetivos.

2.1. Objetivos

El objetivo principal de la propuesta en este artículo consiste (1) en primer lugar proponer un método para medir las habilidades adquiridas por los alumnos en las prácticas de las asignaturas de grado, (2) en segundo lugar deducir si las actividades de enseñanza-aprendizaje son apropiadas para esos alumnos en particular para alcanzar las habilidades u objetivos de aprendizaje en el laboratorio de prácticas. Y (3) en último lugar, con los resultados de seguimiento dar una realimentación a los alumnos sobre las habilidades adquiridas durante su proceso de aprendizaje en el laboratorio.

En paralelo, también nos proponemos que (4) la propuesta requiera un tiempo mínimo para su aplicación, tanto para los profesores como para los alumnos.

2.2. Metodología

La metodología es una adaptación de la metodología desarrollada en el PIME EvalTICs [3]. Y se compone de 7 etapas, las cuales vamos a describir a continuación y que están mostradas también en la figura 1.

Con el fin de facilitar la aplicación de la metodología, algunas de las etapas han sido automatizadas en la herramienta web denominada portal-EvalTiCs [4], con el objeto de minimizar el tiempo empleado a tan sólo el necesario para contestar un test breve.

En la figura 1, están indicados los números de cada etapa de la metodología. En las etapas 1 y 5 es necesaria la participación de los alumnos. En las etapas 2, 3, 6 y 7 las actividades están automatizadas en la aplicación-herramienta, por lo que no representarán un consumo de tiempo, y finalmente en la etapa 4 encontramos las actividades propias del curso.

Debe tenerse en cuenta que para aplicar esta metodología, se deben tener bien identificadas las habilidades y competencias que los alumnos deben adquirir al terminar de ser impartida la asignatura bajo análisis.

La sobrecarga de trabajo para el profesor, al inicio del curso, es la siguiente:

1. Preparar una batería de preguntas que sirvan para confeccionar los tests, así como las respuestas correctas para poder automatizar el proceso de evaluación. Se recomienda preparar el banco de preguntas en el portal institucional.
2. Crear los test que se aplicarán tanto al inicio como al final de cada una de las prácticas bajo análisis. Se recomienda crear los test en el portal institucional.
3. Crear una lista de actividades a proponer, en base a los resultados obtenidos en el test realizado al inicio de las prácticas bajo análisis.

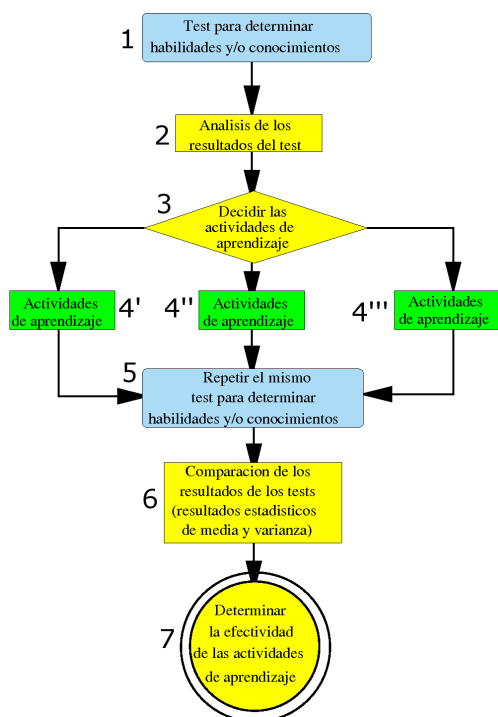


Figura 1: Etapas de la metodología propuesta.

Dicho esto, la primera etapa de la metodología que proponemos, consiste en aplicar un test al inicio de cada una de las prácticas bajo análisis, con el fin de evaluar las competencias y habilidades del alumno.

En la segunda etapa, se procesan los resultados de los tests. El proceso puede hacerse de forma manual o utilizando el portal EvalTICS. En pocas palabras, el proceso consiste en calificar los test y obtener la media y la desviación estándar de las notas obtenidas.

En la tercera etapa, (en base a la nota media obtenida en el test) la herramienta propone al profesor cuál de

las actividades a realizar (de entre las introducidas en la lista de actividades) resulta más conveniente para el estado de los alumnos.

Una vez hemos realizado las tres primeras etapas, que no deberían haber representado un coste importante en tiempo, los alumnos realizarán las tareas prácticas que el profesor les encomiende.

En la quinta etapa, antes de terminar la sesión de prácticas, se procede a aplicar el mismo test utilizado en la primera etapa al inicio de la sesión de prácticas.

Una vez realizado el test por segunda vez, se procede, en la sexta etapa, a un análisis estadístico (análisis de medias), de la progresión de los alumnos en dicha sesión de prácticas.

En la séptima y última etapa, se determina la efectividad de la actividad de aprendizaje visualizando el resultado del análisis de medias. No debe perderse de vista que el profesor en última instancia determina el grado de adquisición de competencias de los alumnos.

Es de vital importancia para la metodología para una mejora educativa, que el profesor se pregunte al finalizar las sesiones de prácticas las siguientes cuestiones:

- ¿La práctica se ha realizado en el tiempo apropiado?
- ¿Los alumnos precisan de una mejor preparación antes de la sesión práctica?
- ¿Se debería buscar la forma de motivar a los alumnos para que preparen la práctica antes de su realización?
- ¿Sería conveniente que antes de la práctica se realice un resumen de cómo realizarla?

También tiene una importancia relevante, que el profesor vuelva a considerar el conjunto de resultados de las diferentes ocasiones que se haya aplicado esta metodología al final del curso. Por lo que deberá:

- Analizar los resultados obtenidos en las prácticas, para determinar si estas refuerzan las actividades de aprendizaje.
- Preparar una encuesta para el alumnado, respecto a la idoneidad de las prácticas.

Finalmente, nos gustaría enfatizar, que la sobrecarga para el profesor es mínima, su trabajo consiste en:

1. Crear la lista de actividades a realizar pensando en diferentes escenarios.
 - Nota media muy baja,
 - Nota media satisfactoria
 - Nota media muy alta.
2. Crear un banco de preguntas a partir del cual se crean los test. A ser posible preguntas de tipo verdadero/falso o de selección múltiple respuesta única.

3. Crear los test. A ser posible en una aplicación Web (Portal institucional, portal-EvalTICs)
4. Aplicar los test. A ser posible en una aplicación Web (Portal institucional, portal-EvalTICs) para que dicha aplicación se encargue de la evaluación de los test.
5. Evaluar los test, y obtener estadísticas descriptivas (media, desviación estándar, varianza)
6. Analizar los resultados para proponer mejoras para el siguiente curso escolar.
7. Crear un cuestionario para medir la satisfacción de los alumnos.
8. Aplicar el cuestionario y tomar en cuenta las respuestas de los alumnos.

3. Resultados obtenidos

A continuación se presentan la asignatura GII-FCO y GII-FC, respectivamente. Y los resultados obtenidos en cada una de ellas.

3.1. Asignatura GII-FCO (UPV)

La asignatura de grado “Fundamento de los computadores” (GII-FCO) es un curso obligatorio del Grado en Ingeniería de Computadores, se imparte durante el primer semestre del Grado. Tiene asignados 4.5 ECTS[1] para teoría y 3.0 ECTS para prácticas de laboratorio. A continuación se describen su temario, los objetivos de aprendizaje y la distribución de alumnos.

Objetivos

El objetivo final de la asignatura es contribuir al entendimiento del funcionamiento de un procesador partiendo del diseño de los componentes que lo forman y de la comprensión del lenguaje máquina necesario para su funcionamiento. Para ello, en primer lugar se introduce una versión holística del funcionamiento de los computadores: arquitectura y unidades funcionales, representación de datos, software y sistemas operativos.

En segundo lugar se aprenden los métodos de diseño de los circuitos digitales básicos, tanto combinacionales como secuenciales, que se utilizarán para implementar los distintos componentes del computador.

En tercer lugar se aprende cómo se representan en los circuitos del computador tanto los datos numéricos (números enteros y reales) como los caracteres alfanuméricos.

Finalmente, se contribuye a la comprensión de los principios de la programación en ensamblador, introduciendo las instrucciones básicas que lo componen y programas básicos elementales.

Temario

- Tema 1. Introducción a los computadores.
- Tema 2. Principios del diseño digital.

- Tema 3. Bloques combinacionales básicos.
- Tema 4. Circuitos secuenciales.
- Tema 5. Bloques secuenciales básicos.
- Tema 6. Representación de la información.
- Tema 7. Introducción al lenguaje ensamblador.

Prácticas

- Práctica 1. Manejo del entrenador lógico
- Práctica 2. Implementación de circuitos
- Práctica 3. Decodificadores y multiplexores
- Práctica 4. Biestables
- Práctica 5. Registros de desplazamiento y contadores
- Práctica 6. Diseño de sistemas secuenciales
- Práctica 7. Ensamblador: Introducción al SPIM
- Práctica 8. Ensamblador: Acceso a datos en memoria
- Práctica 9. Ensamblador: Análisis de programas

Objetivos de aprendizaje

1. Describir el funcionamiento básico de un ordenador. Comprender el papel de las unidades funcionales en el funcionamiento de un ordenador.
2. Resolver cálculos simples relacionados con los parámetros característicos de un computador, tales como: capacidad de memoria, ancho de banda, etcétera.
3. Desarrollar habilidades para representar números naturales utilizando diferentes sistemas de representación de datos (Por ejemplo: decimal, binario natural, signo-magnitud, complemento a uno, complemento a dos, exceso a Z, etcétera)
4. Desarrollar habilidades para calcular sumas y restas con números representados en binario natural y en complemento a dos.
5. Desarrollar habilidades para representar números en coma flotante utilizando el formato IEEE-754. Y desarrollar habilidades para realizar sumas y restas con números representados en el formato IEEE-754.
6. Desarrollar habilidades para escribir en notación matemática la representación de un circuito, partiendo de la definición realizada en lenguaje natural.
7. Desarrollar habilidades para analizar y construir circuitos combinatorios y secuenciales utilizando puertas lógicas.
8. Desarrollar habilidades para escribir e interpretar instrucciones en lenguaje ensamblador y en lenguaje de máquina.
9. Desarrollar habilidades para analizar y construir programas sencillos escritos en lenguaje ensamblador.

Competencias a adquirir:

Genéricas:

- Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas in-

Grupo	Num. de alumnos	Idioma
A	31	Valenciano
B	28	Valenciano
C	51	Español
D	36	Español
E	36	Inglés
F	40	Español
G	37	Español
H	32	Español
I	25	Español
J	29	Español
K	27	Español

Cuadro 1: Alumnos matriculados en GII-FCO en la UPV.

formáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

Transversales:

- Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.
- Razonar de manera abstracta, analítica y crítica, sabiendo elaborar y defender argumentos en su área de estudio y campo profesional.

Método de evaluación de las prácticas

En total se realizan 9 prácticas, cada práctica se realiza en una sesión de laboratorio de 1,5 horas.

Para realizar las prácticas los alumnos pueden trabajar en grupos de 2 alumnos, pero cada alumno debe entregar su boletín de prácticas para ser evaluado.

Las prácticas 1, 2, 3 y 4 se evalúan mediante un examen escrito, la nota máxima del examen son 10 puntos.

Las prácticas 5, 6, 7, 8 y 9 se evalúan de forma independiente, la nota máxima que se obtiene en cada una de las prácticas 5, 6, 7, 8 y 9 son 10 puntos.

La nota final de prácticas tiene un peso del 25 % sobre la nota final de la asignatura.

Para obtener la nota final de prácticas (*NotaPrácticas*) se emplea la siguiente fórmula:

$$NotaPrácticas = Examen * 0,15 + \frac{Prac5 + \dots + Prac9}{50} * 0,10 \quad (1)$$

Distribución de Alumnos

Durante el curso 2012-2013 se matricularon 372 alumnos, los cuales se repartieron en 11 grupos.

Estadísticas por grupo de la nota final de prácticas (*NotaPrácticas*) La siguiente figura muestra la nota media obtenida por los alumnos de la asignatura GII-FCO por grupo.

Grupo	Num. de alumnos	Media	Desv tip.
A	31	1,750	0,684
B	28	1,848	0,521
C	51	2,116	0,427
D	36	1,846	0,586
E	36	2,282	0,365
F	40	1,989	0,527
G	37	1,982	0,402
H	32	1,962	0,545
I	25	2,277	0,432
J	29	2,098	0,271
K	27	1,784	0,559
Total	372	2,001	0,514

Cuadro 2: Estadísticas descriptivas de las notas de prácticas por grupo.

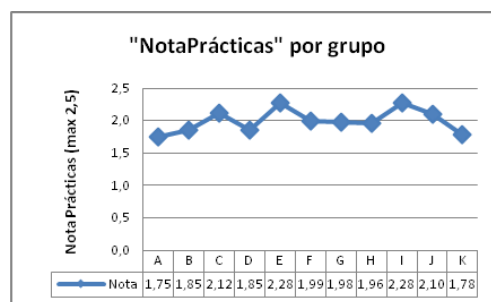


Figura 2: “Notas Prácticas” de los alumnos de la asignatura GII-FCO por grupo.

Estadísticas del grupo 1E

La mejora que se está analizando se aplicó a los alumnos del grupo 1E. A los alumnos de este grupo se les aplicó un test al inicio de las prácticas 6, 7, 8 y 9.

El mismo test se aplicó al final de la práctica. La siguiente tabla muestra las estadísticas descriptivas de los resultados obtenidos en cada uno de los test:

4. Asignatura GII-FC (UCM)

La asignatura de grado “Fundamento de los computadores” (GII-FC) [5, 6, 7] impartida en la UCM, también es un curso obligatorio del Grado en Ingeniería de

	Num. de alumnos	Media	Desviación típica	Varianza
TestPre_Prac6	36	7,31	1,193	1,422
TestPost_Prac6	36	9,92	0,500	0,250
TestPre_Prac7	36	7,42	1,198	1,434
TestPost_Prac7	36	9,94	0,232	0,054
TestPre_Prac8	36	6,00	1,486	2,208
TestPost_Prac8	36	9,92	0,500	0,250
TestPre_Prac9	36	4,99	1,344	1,807
TestPost_Prac9	36	9,97	0,167	0,028
Examen Prácticas	36	9,11	1,214	1,473

Cuadro 3: Estadísticas de los test realizados al inicio y final de las prácticas, y del examen de prácticas.

Computadores, se imparte durante el primer curso del Grado. Tiene asignados 12 ECTS incluyendo teoría y prácticas de laboratorio. A continuación se describen su temario, los objetivos de aprendizaje y la distribución de alumnos.

Competencias a adquirir

Genéricas:

CG4-Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

Transversales:

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

Temario y prácticas asociadas

1^{er} Cuatrimestre:

Tema 1. Representación de la Información

Tema 2. Especificación de sistemas combinacionales

Tema 3. Implementación de sistemas combinacionales

Práctica1: Conversor código Gray a binario

Práctica2: Implementación sumador de 4 bits

Tema 4. Módulos combinacionales básicos

Práctica3: Diseño con multiplexores

Tema 5. Especificación de sistemas secuenciales síncronos

Tema 6. Implementación de sistemas secuenciales

Práctica4: Diseño de un reconocedor con biestables y puertas

Tema 7. Módulos secuenciales básicos

Práctica5: Diseño de un registro

2^{do} Cuatrimestre:

Tema 8. Introducción

Tema 9. Lenguaje Máquina y Ensamblador

Práctica6: Introducción a un entorno de desarrollo y depuración

Práctica7: Programación en lenguaje ensamblador

Práctica8: Programación de subrutinas

Tema 10. Diseño del Procesador

Tema 11. Sistema de Memoria de un Computador

Práctica9: Cómo afecta la programación a la gestión de la memoria

Tema 12. El subsistema de entrada/salida

Práctica10: Cómo se comunica la CPU con su entorno

Grupo	Num. de alumnos	Idioma
A	85	Español
B	72	Español
C	79	Español
D	78	Español
E	66	Español
F	31	Español
G	76	Español

Cuadro 4: Alumnos matriculados en FC en la UCM.

Distribución de Alumnos

Durante el curso 2012-2013 se matricularon 487 alumnos, los cuales se repartieron en 7 grupos.

Método de cálculo de la calificación en FC

$$Nota = 8 * \frac{Examen}{10} + 2 * \frac{Prac1 + \dots + Prac10}{10} \quad (2)$$

5. Análisis de resultados

En esta apartado se analizan los resultados obtenidos en las asignaturas GII-FCO y GII-FC.

5.1. Asignatura GII-FCO (UPV)

A los alumnos del grupo E, se les aplicaron 4 test, cada uno de los test se pasó dos veces, al inicio (Test-Pre) y al final de la sesión de prácticas (TestPost).

A partir de los resultados de la figura 3, se puede deducir que a medida que se acerca el final del cuatrimestre los alumnos dedican menos tiempo a preparar las prácticas, por lo que su nota tiende a ser más baja. Sin embargo, una vez que han realizado la práctica, el examen post indica que los alumnos obtienen muy buenas notas, de hecho, en figura 2 se puede ver que la nota media de los alumnos del grupo 1E es la más alta de todos los grupos.

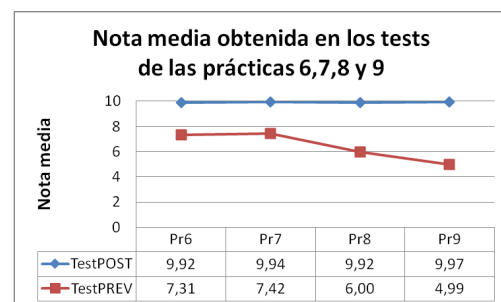


Figura 3: Nota media obtenida en los test pre y post de las prácticas 6, 7, 8 y 9.

5.2. Asignatura GII-FC (UCM)

A los alumnos de los grupos D y G, se les aplicó un test, dos veces como indica la metodología que proponemos, una al inicio (TestPre) y otra al final de la sesión de prácticas (TestPost). La nota media de los tests puede verse en la figura 4.

El test se aplicó en la práctica en la que el profesorado considera que en años anteriores no se han adquirido las habilidades para las que fue diseñada la actividad.

Hemos de decir que los alumnos fueron informados de la metodología que proponemos, por lo que se vieron motivados a prestar atención al saber que tendrían que hacer otro test al final de la clase acerca de los contenidos de la práctica. La primera pregunta era acerca de cómo depurar el diseño o montaje de un circuito y la segunda sobre el funcionamiento de las diferentes puertas lógicas. El conocimiento a evaluar en ambas preguntas es indispensable para poder realizar de las prácticas durante el curso.

Sin embargo, debido a la complejidad de ambas preguntas y para no representar un consumo de tiempo de la sesión de prácticas se optó por no añadir más preguntas.

Cabe indicar que todos los alumnos deben ser capaces de contestar dichas preguntas correctamente al finalizar el primer cuatrimestre. También debemos indicar que gracias a que los alumnos accedieron a indicar su nombre en los tests, se pudieron identificar aquellos alumnos que no habían comprendido perfectamente el método de depuración de circuitos (primera pregunta del test), y se les volvió a explicar individualmente durante la segunda sesión de prácticas.

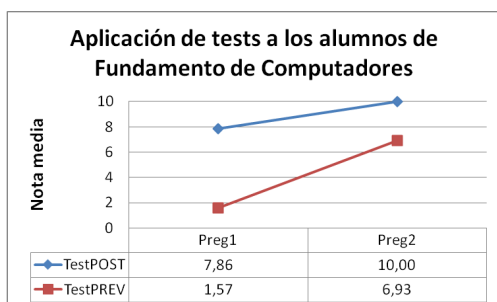


Figura 4: Nota media obtenida en los test pre y post de la primera práctica, en GII-FC en la UCM.

6. Conclusiones

En el caso concreto de la asignatura de grado GII-FCO, el realizar los tests al inicio y al final de la sesión de prácticas les ha servido para obtener una muy buena nota en las prácticas de laboratorio.

En el caso de la asignatura GII-FC, dado que se motivó a los alumnos a realizar los test, su rendimiento aumentó, creemos que en gran medida por la motivación.

En todos los casos analizados, los tests se han diseñado teniendo en cuenta las habilidades que los alumnos deben adquirir.

Por ello consideramos que la metodología propuesta es apropiada para mejorar el rendimiento de los alumnos de cualquiera de las asignaturas del grado en Ingeniería Informática.

7. Agradecimientos

Agradecemos a la Universitat Politècnica de València (UPV), al Departamento de Informática de Sistemas y Computadores (DISCA) de la UPV y a la Escuela Técnica Superior de Ingeniería Informática (ET-SINF) de la UPV, su apoyo para realizar y presentar este artículo.

También a los alumnos de ambas universidades por su colaboración en la realización de los tests.

Referencias

- [1] Espacio Europeo de Educación Superior (EEES) plan Bolonia: <http://ww.ufv.es/falumnos.aspx?sec=1935&gclid=CJS1wLHCsLUCFfLLtAodHT8APg#>
- [2] L.G. Lemus, J. V. Benlloch, F. Buendía, S. Terrasa, M. A. Mateo, J. Pons, A. Perles, M^a A Rodríguez y S. M. Alarcón. “PIME EvalTICs: Uso de las TICs para ayudar al profesorado en la evaluación de la actividad del alumno”
- [3] L.G. Lemus-Zúñiga, M.A.M. Pla, F.B. García, y J.V.B. Dualde. “Project for innovation and Educational improvement EVALTICs” En Technologies Applied to Electronics Teaching (TAEE), 13-15 June 2012 pp 267 – 272.
- [4] L. G. Lemus- Zúñiga, S.M. Alarcón, M.A.M. Pla, y J.V.B. Dualde. “Evaltics Web Portal” En Technologies Applied to Electronics Teaching (TAEE), Junio 2012, pp 308 – 313.
En IV jornada de Innovación Docente ETS Ingeniería informática Nov 2011. Póster.
- [5] Ficha docente de FC en la UCM: http://www.fdi.ucm.es/fdi_info/Pub/ImpresoFichaDocente.aspx?Id=28
- [6] Guía docente de la UCM para el curso 2012-13 http://www.fdi.ucm.es/Guias_Docentes/Guia\%20docente\%202012-13\%20GII.pdf
- [7] Guía docente de la UCM para el curso 2011-12 http://www.fdi.ucm.es/Guias_

Docentes/Guia_docente_2011-12_\%28grado_
Ing_Inform\C3\%A1tica\%29.pdf

Integración de metodologías en Ingeniería

Antonio Polo Márquez
Departamento de Ingeniería de Sistemas
Informáticos y Telemáticos
Universidad de Extremadura
Cáceres
polo@unex.es

Resumen

Una de las competencias más importantes en Ingeniería es conocer y saber aplicar metodologías de trabajo. Sin embargo, a menudo se presentan situaciones en que es necesario trabajar con metodologías diferentes y surge el problema de su integración. Una situación habitual en Ingeniería es partir de una metodología general de mejora continua, tener que desarrollarla con una metodología de trabajo en equipo y aplicar, además, una metodología específica para el dominio del campo en que se esté trabajando. En este trabajo se presenta la forma en que hemos integrado en nuestras asignaturas una metodología de mejora continua genérica como es la clásica del ciclo PDCA o de Deming, una metodología de trabajo en equipo ágil como es Scrum, y una metodología específica de desarrollo de servicios como ITIL. Este modelo se ha aplicado durante el curso 2012-2013 en una asignatura obligatoria anual de quinto curso de Ingeniería Informática y en otra asignatura de primer semestre común en un Máster de Informática, Telecomunicaciones y Dirección TIC. Las conclusiones ponen de manifiesto la utilidad de tales integraciones para el aprendizaje basado en proyectos en cualquier rama de la Ingeniería Informática.

Abstract

One of the most important skills in engineering is to know and be able to apply methodologies. However, there are often situations where it is necessary to work with different methodologies and we must solve the problem of their integration. A common situation in Engineering is starting with a general methodology for continuous improvement, having to develop a methodology of teamwork and furthermore applying a specific methodology to the domain of the field in which you are working. This paper presents how we have integrated in our subjects a classic methodology of continuous improvement as the Deming PDCA cycle with a methodology of agile teamwork as Scrum, and a specific methodology to services

development such as ITIL (Information Technology Infrastructure Library). This model has been applied during the year 2012-2013 in an annual mandatory subject of 5th course of Computer Engineering and in another subject of first semester common in a Master for Computing, Telecommunications and ICT Management. The conclusions highlight the usefulness of such integrations for project-based learning in any branch of Computing Engineering.

Palabras clave

Integración de metodologías, trabajo en equipo, Scrum, ITIL (Information Technology Infrastructure Library), Desarrollo de servicios, PDCA o ciclo de Deming, Mejora continua, Aprendizaje basado en proyectos.

1. Motivación

Una de las competencias básicas de cualquier Ingeniería consiste en conocer y saber aplicar la metodología de trabajo más adecuada. Sin embargo, según el aspecto que se aborde en la resolución de un proyecto, surgen diferentes metodologías con un enfoque concreto y que, no obstante, deben aplicarse de forma simultánea durante el desarrollo del trabajo. En nuestro caso, planteamos un entorno de desarrollo de proyectos de Ingeniería Informática en equipo, en los que vamos a utilizar una metodología genérica de trabajo basada en el ciclo de mejora continua. Por tanto, sobre esa metodología debe superponerse una metodología de trabajo en equipo y también la del campo específico en que se desarrolle nuestro proyecto. En muchos casos son proyectos multidisciplinares, por lo que deben superponerse diferentes metodologías de trabajo.

Para la enseñanza de estas metodologías, se suele incidir en cada una de ellas específicamente, y no suelen considerarse de forma conjunta.

En este trabajo presentamos la aproximación que hemos aplicado en diferentes asignaturas, mostrando las metodologías básicas que hemos seleccionado, las

estrategias para su integración y los resultados de su aplicación en dos asignaturas diferentes.

En nuestro caso proponemos como modelo general de trabajo una metodología de mejora continua, basada en el ciclo PDCA o de Deming [1], aplicada a una metodología de trabajo en equipo ágil basada en la metodología Scrum [2], y aplicando todo ello a una metodología general de desarrollo de servicios como ITIL [3]. Todas ellas se encuentran en la base de la *eficiencia productiva* o *eficiencia en las Tecnologías de la Información (lean manufacturing or lean IT)* [4], que constituye actualmente un extenso campo de estudio en sistemas de producción.

Además de indicar cómo se han integrado estas metodologías, nuestro objetivo es proporcionar un marco general para integrar cualquier otro conjunto de metodologías y las pautas para conseguirlo satisfactoriamente.

A continuación se describen los diferentes aspectos en el desarrollo de un proyecto y las metodologías que hemos seleccionado para cada uno de ellos. Después se trata el problema de la integración de estas metodologías y cómo los hemos resuelto en nuestras asignaturas. Finalmente se comentan los resultados de nuestra experiencia y presentamos nuestras conclusiones. Pensamos que esta propuesta puede aportar ideas para aprendizaje basado en proyectos [5], y ayudar a la adquisición de competencias de trabajo en equipo para el desarrollo, iterativo e incremental, de productos o servicios en cualquier dominio de la Ingeniería Informática.

2. Metodologías para el desarrollo de proyectos

Cada campo de la Ingeniería suele proporcionarnos una metodología concreta que permite la construcción de productos o herramientas que resuelvan los problemas en ese dominio. Sin embargo, hay dos aspectos que complican esta situación: la necesidad de disponer cuanto antes de la solución y la complejidad del problema. Como veremos, la primera de ellas nos conduce a introducir procesos de desarrollo iterativos de mejora continua, y la segunda a abordar el proyecto con equipos humanos cada vez más complejos.

Además, la especificación y uso de los resultados del proyecto necesitan de un modelo que facilite su gestión. En especial la calidad del producto se asocia a la utilidad del mismo y a la satisfacción del usuario que lo necesita. Los modelos de servicios nos aportan el nivel de abstracción necesario para especificar la calidad del resultado del proyecto y de su desarrollo.

La Figura 1.(A) muestra estos tres aspectos básicos a la hora de desarrollar un proyecto: a) considerar el desarrollo del proyecto como un proceso de mejora continua, b) aplicar técnicas de trabajo en equipo y

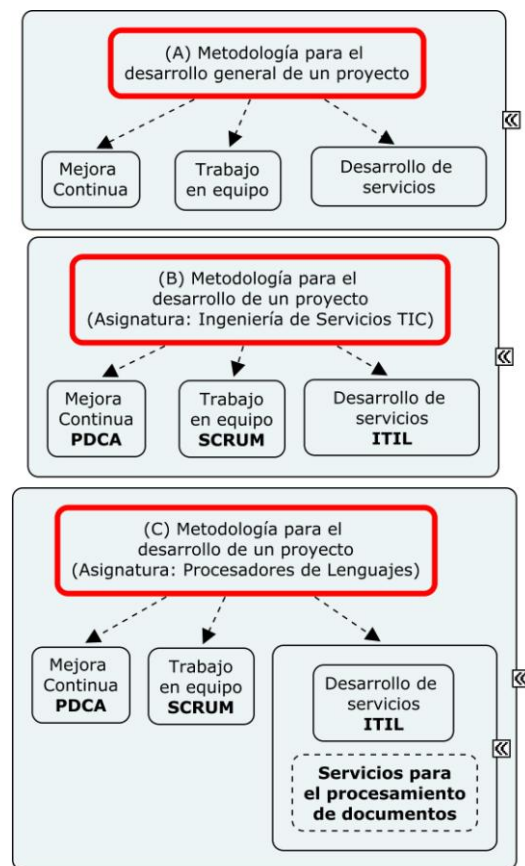


Figura 1: Tipos de metodologías para proyectos en Ingeniería Informática.

c) desarrollar los resultados del proyecto en forma de servicios.

Existen metodologías específicas para el desarrollo de proyectos en cada una de las anteriores dimensiones especificando, en cada caso, los procesos y tareas a seguir. Sin embargo, estas metodologías son independientes entre sí, por ejemplo, podemos considerar una metodología para desarrollo de un proyecto en equipo y aplicar en ese mismo proyecto una metodología de desarrollo en cascada [6], o utilizar una metodología de mejora continua para nuestro trabajo individual.

En actividades de aprendizaje basado en proyectos se potencia la adquisición de las competencias necesarias para el trabajo en este contexto, pero en muchas ocasiones sólo se centra en alguno de esos aspectos, cuando en la mayoría de los casos de la vida real se deben tener en cuenta simultáneamente.

Las metodologías seleccionadas para cada una de las dimensiones anteriores se han aplicado en las asignaturas tal y como se indica en la Figura 1.(B), donde la asignatura *Ingeniería de Servicios TIC* se centra en el desarrollo de servicios y la Figura 1.(C), donde la asignatura de *Procesamiento de Lenguajes* se enfocada hacia el estudio de los servicios para el procesamiento de documentos.



Figura 2: Ciclo PDCA o de Deming [http://www.calidad-gestion.com.ar/boletin/58_ciclo_pdca_estrategia_para_mejora_continua.html].

2.1. Metodologías de mejora continua en Ingeniería y educación

La necesidad de disponer cuanto antes de la solución hace que numerosos proyectos de Ingeniería fracasen o produzcan herramientas inservibles hasta que no han sido utilizadas durante un tiempo y han sido sometidas a un proceso de mejora.

Disponibilidad inmediata y calidad de la solución parecen estar enfrentados en esta situación, en especial si lo que queremos es obtener el producto o servicio final completo.

Por ello, una propuesta para gestionar la inmediatez de la respuesta con una calidad adecuada, es renunciar al producto completo y disponer de una solución parcial que no satisfaga todo lo que esperamos de ella, pero que trabaje correctamente al menos en aquellas funcionalidades que se han desarrollado.

Esto supone dar tiempo al ingeniero para que pueda completar la herramienta, mientras el usuario ve mejorada su situación al menos en las funciones desarrolladas.

Surge así el concepto de versión de la solución y el de ciclo como el tiempo de que se dispone para mejorar la solución entre una versión y la siguiente.

La idea de ciclo iterativo de mejora continua nos puede parecer muy simple, pero su introducción en el sistema de producción industrial japonés tras la segunda guerra mundial supuso una revolución.

Fue Peter Deming, basándose en unos estudios previos de Walter A. Shewhart quién introdujo el ciclo de mejora continua o ciclo de Deming [1] en las empresas japonesas a mediados del siglo pasado. Ello supuso una alternativa al sistema de producción americano basado en cadenas secuenciales de fabricación. Aunque aparentemente ambos sistemas podían competir entre sí, el proceso de mejora

continua pronto tomó un nicho de mercado en sistemas dinámicos y permitía adaptarse más fácilmente ante nuevos cambios. Gran parte de las metodologías actuales son simples adaptaciones de esas ideas, como los sistemas de producción JIT (just in time) del Lean Manufacturing [4] o el Sistema de Producción de Toyota [7].

En lugar de considerar alguna de estas variantes, en este trabajo hemos tomado la propuesta primitiva de ciclo PDCA por su sencillez y fácil asimilación a otras metodologías.

Además de los trabajos de Shewhart y Deming, son muy interesantes las propuestas desarrolladas por Myron Tribus [8], en los que aplica los principios de calidad industrial derivados del ciclo de Deming, a la calidad en educación. Nuestro trabajo está fuertemente influido por estos estudios.

El ciclo PDCA aparece representado en la Figura 2, y propone un ciclo en cuatro fases:

- Check (C).- Comprobar y verificar los resultados del sistema actual.
- Action (A).- Tras analizar los resultados determinar las posibles acciones de mejora a llevar a cabo y decidir cuáles son las que deben tomarse antes de iniciar la fase de planificación de las acciones.
- Planning (P).- Planificar las acciones necesarias para llevar a cabo las mejoras en el sistema.
- Do (D).- Hacer o llevar a cabo las acciones planificadas.

Posteriormente, los resultados de la fase del Do pasan al proceso de comprobación (Check) iniciándose un nuevo ciclo de mejora.

Las características esenciales de esta metodología son:

1. Al final de cada ciclo se obtiene una versión del producto que debe estar completamente operativa en las funcionalidades que se hayan decidido desarrollar en ese ciclo.
2. El proceso se itera sobre un conjunto de ciclos, normalmente de duración constante, en el que cada versión del producto mejora a la anterior, hasta agotar los recursos del proyecto o alcanzarse el producto final.

2.2. Metodologías de trabajo en equipo

Además de la necesidad de la inmediatez de la solución, la Ingeniería se ha distinguido por abordar problemas cada vez más y más complejos. La mayor parte de ellos sólo pueden resolverse mediante equipos de ingenieros, en muchos casos procedentes de campos diferentes.

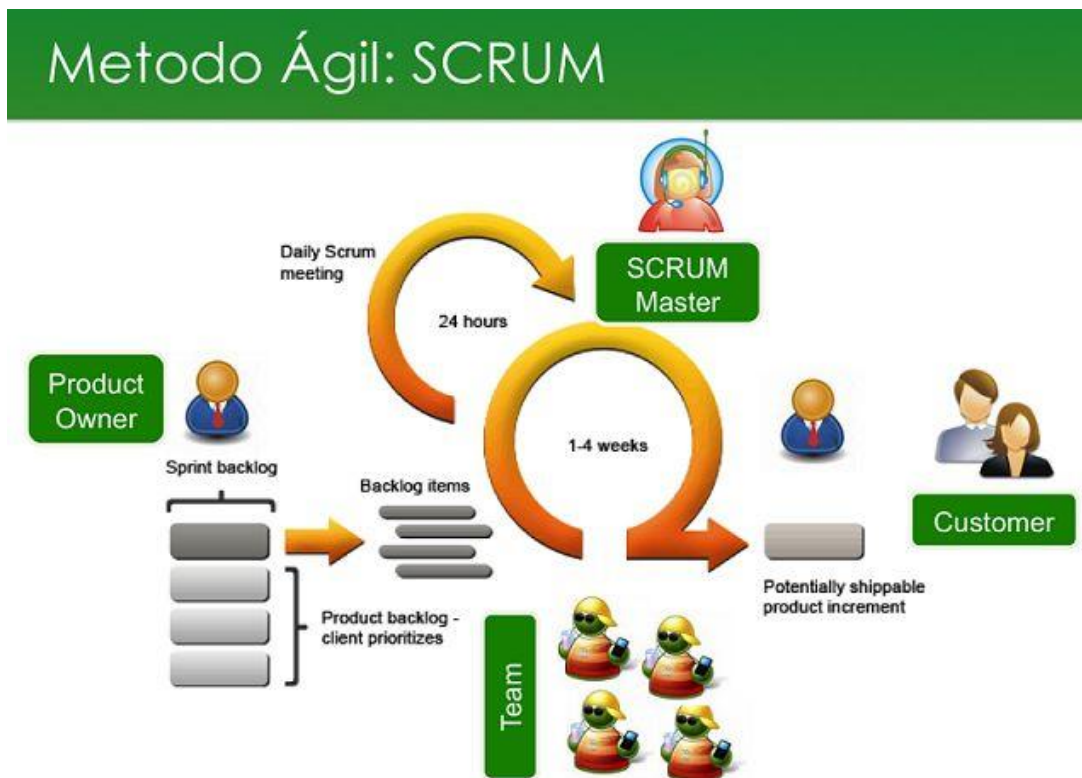


Figura 3: Metodología Scrum

[<http://santimacnet.wordpress.com/2010/11/04/curso-gratis-scrum-dia-a-dia/>]

Las metodologías que se han propuesto para conseguir mejoras de trabajo en equipo frente al individual aplican dos formas básicas de trabajo: 1) dividir el problema entre los integrantes del equipo y luego unir las soluciones parciales que hayan elaborado, o 2) resolver todos el mismo problema y luego mezclar las diferentes soluciones en una sola que aproveche todas las fortalezas que aporten las soluciones parciales.

Existen numerosas metodologías de trabajo en equipo, pero necesitamos alguna que permita adaptarse a desarrollos dinámicos de forma compatible con el ciclo de mejora continua. Una clase de ellas incluye las denominadas metodologías ágiles [9], de las cuales Scrum [2] es uno de los mejores exponentes, y cuyo origen está orientado hacia la producción de software. Sin embargo, nosotros tomaremos su filosofía general para adaptarla a cualquier desarrollo de trabajo en equipos en Ingeniería, de modo que faciliten la planificación, coordinación, seguimiento, integración y evaluación del trabajo realizado.

En Scrum, la adaptación a los cambios del mercado, o de los requisitos del cliente, y la rapidez y calidad de los resultados se consiguen con la noción de Sprint.

Una metodología ágil del tipo Scrum proporciona resultados en muy poco tiempo, mediante sprints muy cortos, con participación del propietario del producto

en el proceso de evaluación de cada resultado parcial y en la determinación de las acciones a desarrollar en el próximo Sprint. Como aparece representada en la Figura 3.

Algunas de las características de esta metodología son las siguientes:

- Cada ciclo se denomina Sprint.
- En el desarrollo del producto o servicio deben considerarse a todos los agentes interesados en el mismo. En particular se distinguen el Propietario del Producto (EP), el Propietario de la Empresa de Desarrollo, el Scrum-Máster y el resto de miembros del equipo.
- Al final e inicio de cada ciclo se mantiene una reunión con el Propietario del Producto para evaluar el producto y determinar las características a desarrollar en su próxima versión.

2.3. Una metodología del dominio de la Ingeniería basada en servicios

Además de la metodología de mejora continua y la de trabajo en equipo, necesitamos al menos otra metodología que permita obtener las soluciones concretas en el campo específico de ingeniería que estemos trabajando.

En nuestro caso, en lugar de abordar directamente la metodología específica, hemos superpuesto una

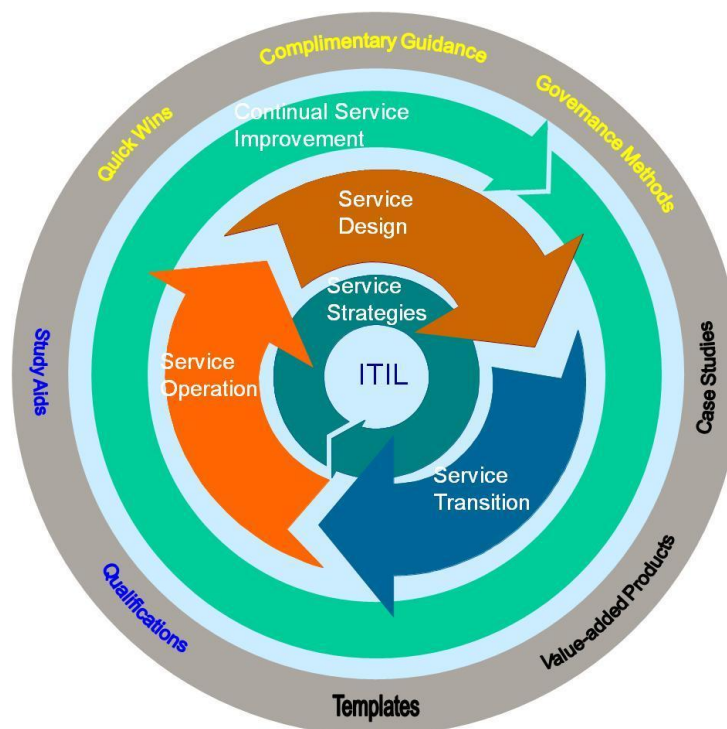


Figura 4: Metodología ITIL v3.

capa genérica que facilite la aplicación de los métodos concretos del dominio con las soluciones finales que se vayan a desarrollar.

Esta capa abstracta de la solución se trata de la “capa de servicios”, y se ha introducido una metodología para el desarrollo general de servicios como es ITIL v3.

Ello no sólo facilitará la integración posterior, sino que dota de una capa de abstracción a nuestro modelo que permite guiar la aplicación de cualquier metodología específica de nuestro campo hacia el desarrollo de las soluciones concretas. Y de otra parte, es una metodología genérica que suele utilizarse en el entorno empresarial actual dentro de lo que ha dado en llamarse “Ciencia de los Servicios”. [10].

Por ello tomamos como punto de partida la metodología ITIL v3 que aparece representada en la siguiente Figura 4.

Sin embargo, esta metodología difiere de las anteriores en varios aspectos importantes:

- No es una metodología en sí. En realidad, ITIL (Information Technology Infrastructure Library) es una Biblioteca de Infraestructuras para Tecnologías de la Información. Su origen fue un conjunto de “Buenas Prácticas” para el desarrollo de servicios.
- Sin embargo, se considera habitualmente que ITIL proporciona una metodología, tal y como aparece en la Figura 4.

Nótese que no es exactamente una secuencia clara de pasos a seguir. Por ejemplo, el núcleo más interno

del dibujo presenta la *Estrategia del Servicio* como un concepto que domina todo el desarrollo de cualquier servicio.

Estas observaciones implican que no es fácil interpretar este modelo en una metodología concreta. Nosotros hemos adoptado una interpretación en la que aparezcan secuencias de pasos, con ciclos naturales que permitan obtener productos parcialmente operativos al final de cada ciclo en un proceso de mejora continua.

Además, en esa secuencia se deben detectar las fases de cada ciclo PDCA para poder asociar las *indicaciones* de ITIL con una metodología compatible con el ciclo PDCA.

3. El problema de la integración de metodologías

Si nos fijamos en los tres aspectos anteriores: mejora continua, trabajo en equipo y campo de aplicación, al menos tendremos que integrar tres metodologías que estarán orientadas hacia el aspecto concreto para el que han sido desarrolladas.

En la mayoría de los casos, estas metodologías se estudian separadas, pero deben aplicarse de forma simultánea en el desarrollo de un proyecto. Para su aplicación real deberíamos proporcionar al alumno una estrategia que le permita integrarlas en una sola metodología de trabajo. El problema es que estas metodologías son independientes entre sí, aunque

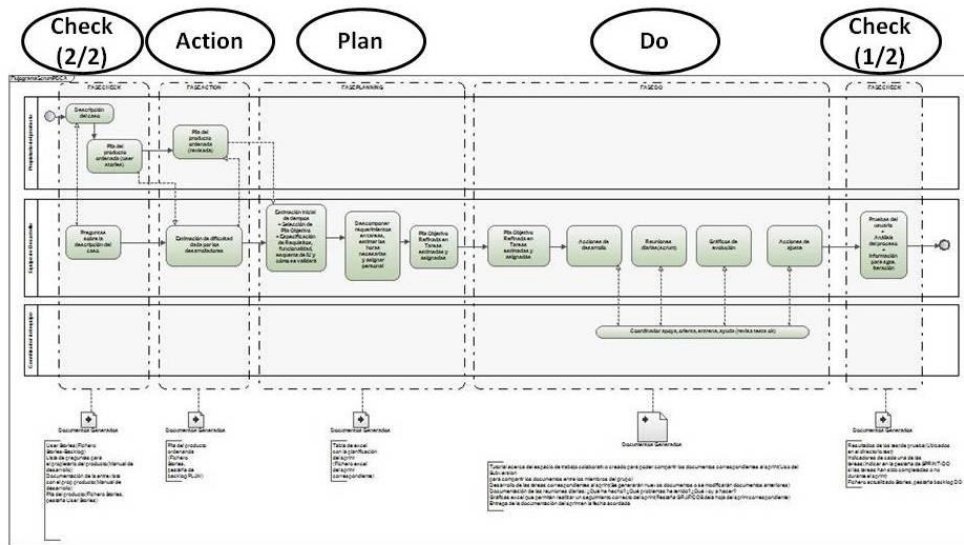


Figura 5: Integración PDCA y Scrum.

aparentemente parezcan más fáciles de integrar determinadas familias de metodologías según el espíritu que propongan. Por ejemplo, parece que será más fácil de integrar una metodología de desarrollo ágil como Scrum con la metodología de mejora continua, que con una metodología de desarrollo en cascada clásica.

A continuación mostramos la estrategia que proponemos para integrar las metodologías y cómo lo hemos aplicado para integrar PDCA, Scrum e ITIL.

3.1. La metodología PDCA o ciclo de Deming como núcleo central

Nuestra propuesta consiste en tomar como base la metodología de mejora continua PDCA, de forma que la integremos con cualquiera del resto de metodologías que vayamos a utilizar. Posteriormente se aplica en cada ciclo PDCA el conjunto de procesos que se han detectado en cada metodología para aplicarlas conjuntamente.

De esa forma, el ciclo PDCA actúa como un reloj que agrupa los procesos de las diferentes metodologías en cada una de sus subfases, por lo que resulta más sencillo realizar la integración a partir de todas las integraciones con PDCA.

En nuestro caso, veremos que la integración PDCA-Scrum es bastante simple, pero no tanto la integración PDCA-ITIL que requiere de una interpretación de ITIL para adaptarla a PDCA.

3.2. Integración de PDCA con SCRUM

Tanto la metodología de ciclo de mejora continua PDCA como la metodología Scrum se integran fácilmente. Para ello se determinan los procesos de Scrum

que coinciden conceptualmente con la fase correspondiente del ciclo PDCA.

Una posible solución aparece en la Figura 5. En primer lugar se han representado los procesos propuestos por la metodología Scrum, en un diagrama BPMN (Business Process Model and Notation) [11]. Posteriormente, se han agrupado los procesos definidos en dicho diagrama en las fases del ciclo PDCA, de forma que un ciclo PDCA se asocia con cada Sprint de Scrum en base a dichas agrupaciones.

Esta integración es sencilla dada la naturaleza de tipo iterativo e incremental que subyace en ambas. El problema surge al integrar con otra metodología, por ejemplo la específica de nuestro campo de aplicación, y más en concreto la *metodología abstracta* ITIL v3, que no tiene definido el concepto de ciclo aunque lo parezca inicialmente.

3.3. Integración PDCA-Metodología de desarrollo de servicios - ITIL_v3

Existen diferentes aproximaciones a la integración de las metodologías PDCA e ITIL. Una de ellas es intentar integrar directamente ITIL sobre PDCA-Scrum de la Figura 5. Sin embargo, no parece sencilla esta superposición, pues no está claramente definido el concepto de ciclo en ITIL.

En efecto, la Figura 4 de ITIL no define con exactitud los ciclos de desarrollo del servicio y lo que define son fases generales de desarrollo y para todos los servicios globalmente. Nótese que *Service Transition* representa todo el proceso de construcción del servicio, mientras que *Service Operation* representa todo el proceso de funcionamiento operativo del servicio. En ninguno de ellos aparece la noción de

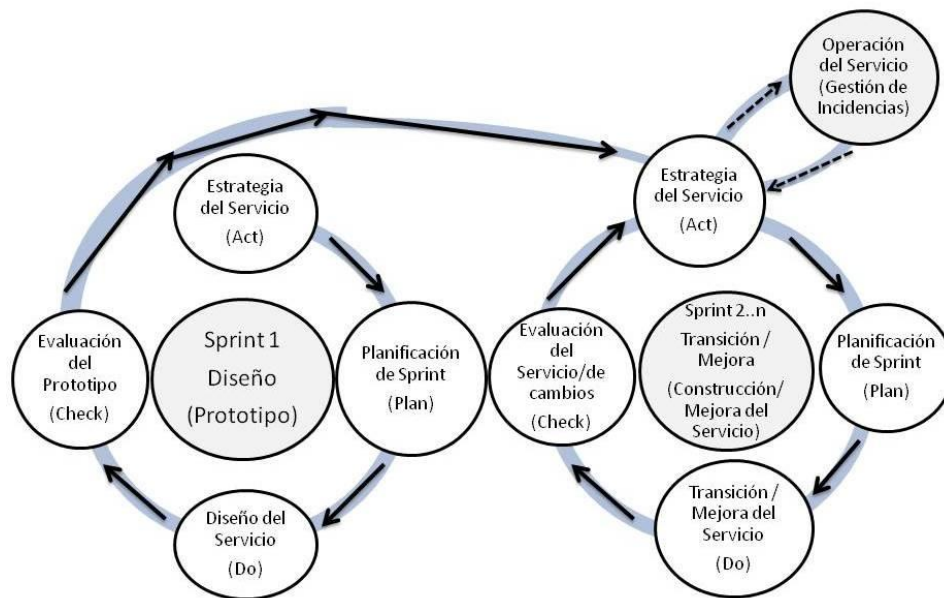


Figura 6: Integración PDCA e ITIL.

desarrollo parcial del servicio y por tanto no pueden ser ciclos por sí mismos.

Por tanto se ha optado por una *revisión* de los conceptos fundamentales de ITIL y su filosofía con la de mejora continua PDCA. El resultado se presenta en la Figura 6. En esta solución aparecen varias cuestiones interesantes:

- Dado que deseamos tener al final del primer sprint un primer producto, se ha optado por obtener un prototipo inicial al final del sprint 1, que sirve de entrada al sprint siguiente, que entra en un ciclo de mejora constante del servicio.
- Se ha separado la fase de operación, de las fases de construcción y mejora del servicio.
- La fase de operación no forma parte de los sprints, pero las salidas que se vayan generando (errores o incidencias del helpdesk), servirán de entrada a las fases de mejora de cada sprint.
- Se pueden estar desarrollando simultáneamente diferentes funcionalidades del servicio o varios servicios diferentes. Esto puede suponer una mayor complejidad a la hora de gestionar el desarrollo de cada sprint, pero aparece como una solución natural.
- Con esta interpretación de ITIL se ha incluido la noción del ciclo PDCA, lo que representa la integración de ITIL con PDCA.

Por último, ahora es fácil realizar la integración de las tres metodologías, tomando como claves las fases P, D, C y A y la noción de ciclo para integrar la Figura 5 y Figura 6 en un modelo que aúna las tres metodologías.

4. Aplicación de esta integración de metodologías

Este modelo se ha aplicado durante el curso 2012-2013 en dos asignaturas de cursos superiores impartidas en la Escuela Politécnica de la Universidad de Extremadura.

Ambas asignaturas están orientadas al aprendizaje basado en proyectos, que se ha desarrollado con las metodologías propuestas anteriormente.

Una asignatura era de primer semestre en un Máster común de Informática, Telecomunicación y Dirección TIC para el desarrollo de servicios TIC. El hecho de que el producto final de esta asignatura sean servicios ha facilitado directamente el uso de nuestro modelo.

La segunda, era una asignatura anual obligatoria de 5º de Ingeniería Informática para el Procesamiento de Lenguajes. Esta asignatura se ha orientado al desarrollo de servicios para el procesamiento de documentos. De esa forma, se ha integrado la metodología específica de diseño y uso de documentos en la metodología ITIL.

En ambos casos se ha necesitado un esfuerzo para realizar la integración de las metodologías, en especial hasta alcanzar la interpretación de ITIL que se presenta en la Figura 6.

5. Conclusiones

Se ha proporcionado una integración de las metodologías PDCA, Scrum e ITIL v3. El método de integración permite apreciar la filosofía de cada una

de ellas. Con ello se proporciona un punto de partida para aplicar en cualquier asignatura una metodología de desarrollo basada en el desarrollo de trabajos en equipo de forma incremental y con mejora continua.

Esta forma de integración permite mostrar al alumno una única metodología que permite abordar de forma integral un proyecto que utiliza metodologías para diferentes perspectivas.

Además se ha mostrado que la propuesta de integrar siempre cada metodología con PDCA y luego integrar todas unificando las fases PDCA es una forma simple, aunque debe probarse con otros modelos específicos y en campos diferentes.

La metodología ITIL parece ser adecuada para conectar nuestro modelo con cualquier otra metodología dado el grado de abstracción y generalidad que proporciona ITIL.

Proponemos ITIL para el desarrollo de proyectos en equipo con filosofía de mejora continua en cualquier campo de Informática. El único requerimiento es abstraer la metodología concreta a una capa de servicios. Este nivel de abstracción facilita todas las ventajas adicionales que proporcionan los servicios como publicación, reutilización y combinación.

Además, se ha propuesto una adaptación de las recomendaciones de ITIL v3 a una metodología de desarrollo incremental de servicios que se muestra en la Figura 6. Este ha sido un resultado de la integración real de estas metodologías en asignaturas donde han sido probadas.

En próximos cursos ofreceremos este modelo como punto de partida inicial para realizar el desarrollo de proyectos abarcando aspectos de mejora continua, desarrollo ágil en equipo y desarrollo de servicios o productos específicos del campo de la Ingeniería Informática en el que se desarrolle la asignatura.

Referencias

- [1] W. Edwards Deming. Out of the crisis. Center for Advanced Engineering Study, Massachusetts Institute of Technology, Cambridge, Massachusetts. 1982.

- [2] Hirotaka Takeuchi and Ikujiro Nonaka. New Product Development Game. Harvard Business Review 86116:137–146, 1986. January 1, 1986.
- [3] ITIL 2011 (R) Official Publications. In <http://www.itsil-officialsite.com>.
- [4] Daniel Roos, James P. Womack and Daniel T. Jones, The Machine That Changed the World : The Story of Lean Production. Harper Perennial (November 1991).
- [5] Barbara J. Duch, Susan E. Groh and Deborah E. Allen. The Power of Problem-Based Learning: A Practical "How To" for Teaching Undergraduate Courses in Any Discipline. Stylus Publishing, 2001.
- [6] El expendedor - Juego de simulación de Scrum. Disponible en: <http://www.proyectosagiles.org/expendedor-juego-simulacion-scrum>.
- [7] Jeffrey K. Liker. The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer. McGraw-Hill, 2004.
- [8] Myron T. Tribus. Quality Management in Education. In Myron Tribus' Essays Collection. Disponible en <http://www.qia.com.au/pages/Tribus.html>.
- [9] Kent Beck et al. Manifesto for Agile Software Development. Agile Alliance, 2001.
- [10] Bill Hefley and Wendy Murphy (eds.). Service Science, Management and Engineering Education for the 21st Century. Springer, 2008.
- [11] Thomas Allweyer. BPMN 2.0: Introduction to the Standard for Business Process Modeling. Books on Demand GmbH, 2010.

Una titulación de informática utilizando proyectos integradores con PBL

María Cruz Gaya López¹, Juan José Escribano Otero¹, María José García García²

¹Departamento de Informática Automática y Comunicaciones

²Departamento de Dirección Académica

Escuela Politécnica

Universidad Europea de Madrid

Villaviciosa de Odón

{mcruz, juanjose.escribano, mariajose.garcia}@uem.es

Resumen

El alumno de primero de ingeniería informática empieza a estudiar la titulación con una visión idealizada de los contenidos, más relacionada con las redes sociales y los videojuegos que con la programación o los sistemas digitales. Con el doble objetivo de motivar a los alumnos de primeros cursos, incrementando el porcentaje de éxito en las asignaturas y disminuyendo el número de abandonos en la titulación, y de acercar a los alumnos de los últimos cursos al mundo profesional, hemos incorporado el método del Aprendizaje Basado en Proyectos (ABP) a todos los cursos del Grado en Ingeniería Informática (GII). En este artículo se describe la motivación de la incorporación del método ABP a nuestra titulación, la descripción de la experiencia (el diseño, las dificultades encontradas y las soluciones aportadas) y la evaluación de la misma (diseño de los mecanismos para la obtención de datos y el análisis de los datos obtenidos).

Abstract

Freshmen begin to study a computer engineering degree with an idealized vision of content, more related to social networks and video games than to programming or digital systems. With the dual purpose of motivating students on their first years at university (increasing the success rate in our courses and decreasing the number of dropouts), and get students in their final years close to the professional world, we have incorporated Project Based Learning method (PBL) to all years in Computer Engineering Degree. This article describes the motivation of including the method PBL to our degree, the description of the experience (design, difficulties founded and solutions) and the evaluation of the same (design of the mechanisms for data collection and analysis of data obtained).

Palabras clave

PBL, ABP, motivación, competencias, proyecto integrador, aprendizaje profundo.

1. Motivación

“Aprender haciendo” es, sin duda, una buena forma de aprender. Cuando un estudiante aplica lo aprendido sobre un proyecto en concreto, ve un resultado de su aprendizaje que le ayuda a fijar conocimientos y adquirir habilidades que en el futuro le serán útiles.

Pero si, en lugar de desarrollar un proyecto a partir de los conocimientos ya adquiridos, aplicamos el método de aprendizaje basado en proyectos [1], [5] aparecerán, además de la necesidad de adquirir nuevos conocimientos técnicos, otras necesidades más generales relacionadas con el proceso, con su documentación o con la participación de varias personas en el mismo proyecto.

Así, el aprendizaje basado en proyectos resulta un marco propicio para que se produzca aprendizaje profundo en el futuro ingeniero, facilitando el desarrollo y evaluación de sus competencias [3]. Este método utiliza un proyecto como columna vertebral del aprendizaje, desarrollándose en equipo con la guía de los docentes y terminando con una exposición final, pública, donde se muestra lo conseguido y el proceso que lo hizo posible.

Otra forma de enriquecer la experiencia de aprendizaje de los estudiantes utilizando ABP consiste en diseñar los proyectos tomando como referencia futuros proyectos-tipo profesionales con el objetivo de acercar al estudiante al mundo profesional, pero enmarcándolos en los contenidos del plan de estudios actual. De esa forma se prepara al estudiante posicionándolo como un agente activo, en contacto con su entorno, dispuesto a intervenir profesionalmente en él

a través trabajos y proyectos como los que ya ha desarrollado a lo largo de su titulación [12].

Por otra parte, si en cada proyecto a desarrollar se implican varias asignaturas, con contenidos teórico-prácticos distintos y complementarios, se producirá una integración que no sólo transmitirá una idea global del conocimiento al estudiante, intentando contrarrestar la posible parcelación de los conocimientos que se produce por el diseño tradicional de los planes de estudio, sino que además favorecerá una mayor coordinación entre docentes evitando el aislamiento y la pérdida de visión global de la titulación que a veces se produce en estos, y generando también en los profesores una mayor motivación.

Partiendo del objetivo inicial de combatir la desmotivación [9][10][11], en la Escuela Politécnica de la UEM se ha puesto en marcha un proyecto para incluir la metodología del Aprendizaje Basado en Proyectos (ABP) en todos los cursos de todas las titulaciones de grado, ya que de este modo [13] los estudiantes logran:

- Aprender más y mejor, consecuencia de un aprendizaje multidisciplinar.
- Integrar los conocimientos técnicos con otros elementos fundamentales en la formación de un ingeniero.
- Fomentar el desarrollo de competencias transversales.

Son muchas las experiencias previas en la aplicación de ABP en las ingenierías (por ejemplo, podemos citar [1][2][4][7][8]). Con todas estas premisas se ha llevado a cabo la experiencia descrita en este artículo y se han conseguido algunos resultados interesantes.

2. Descripción de la experiencia

2.1. Diseño

Cuando se estaba preparando el curso 2012-2013 en la Escuela Politécnica de la Universidad Europea de Madrid (UEM) se decidió implantar un enfoque común en sus titulaciones de grado, construyéndose así la que se ha venido a llamar *Project Based Engineering School* (PBES).

La forma de articular este enfoque consiste en que, en todas las titulaciones de grado, se han diseñado proyectos integradores (PI) de modo que, en cada curso, los estudiantes realizarán al menos un proyecto de ingeniería que integre varias materias y que se adapte a la etapa formativa en que se encuentran. Los proyectos se realizarían en equipos de trabajo, supervisados por los profesores de las materias que lo componen.

Estos proyectos, además de favorecer un aprendizaje profundo de las competencias y conocimientos

generales y específicos, incorporan elementos transversales a los que en la escuela se quiere dar relevancia como son el acercamiento a las profesiones (permitiendo la participación de empresas en los proyectos), la internacionalidad, la sostenibilidad (vista desde los tres ámbitos social, económico, ambiental), el emprendimiento y la innovación tecnológica

El cuadro de la figura 1 describe los proyectos diseñados y puestos en práctica en el Grado de Ingeniería Informática (GII).

A continuación, se expone en más detalle los proyectos realizados en los tres primeros cursos de la titulación:

El proyecto integrador llevado a cabo en primer curso consiste en la realización de una aplicación de escritorio que maneje listas. En este proyecto se aplican los conocimientos que el alumno adquiere en 4 de las asignaturas de primer curso: Fundamentos de programación, Programación con Estructuras Lineales, Laboratorio de Programación y Habilidades Comunicativas. Este proyecto ha sido propuesto y diseñado por los profesores de las asignaturas y es muy cercano al programa de las mismas, puesto que se considera que el alumno todavía no tiene autonomía suficiente como para afrontar proyectos más complejos de forma independiente. Sin embargo la temática concreta sobre la que se desarrollaba el proyecto era seleccionada por cada equipo de trabajo. Ejemplos de proyectos han sido: un gestor de citas médicas; un sistema para el control de inventario de mercancías de un almacén; una agenda electrónica con eventos y contactos; una agenda de tareas. Como resultado de aprendizaje se obtiene:

- Una memoria técnica del proyecto.
- Una URL donde el alumno ha sindicado su aplicación.
- Una presentación oral en un auditorio frente a un tribunal compuesto, al menos, por todos los profesores que han intervenido en el proyecto

En el proyecto integrador de segundo cada grupo debía implementar un sistema gestor de base de datos con dos interfaces de acceso: una aplicación de escritorio para un tipo de usuario "privilegiado" y otra interfaz web para otro tipo de usuario. Interviene las asignaturas de Introducción a la Ingeniería del Software (donde el alumno realiza el análisis de la aplicación), Programación Orientada a Objetos (donde el alumno realiza el diseño de clases y un prototipo de la aplicación de escritorio), Bases de Datos (donde el alumno diseña e implementa la BD e introduce en la aplicación de escritorio las sentencias para la interacción con la BD) y Programación web (donde se diseña e implementa la aplicación web). En este proyecto se contó con el apoyo de una empresa externa que intervino definiendo el proyecto, actuando como cliente en la toma de requisitos y verificación

del diseño, y como parte del tribunal evaluador. El proyecto, consistente en un sistema gestor de niños con TDAH, fue a propuesta suya. Esta propuesta inicial fue modificada por los profesores de las asignaturas quienes decidieron las tecnologías a utilizar. Los resultados del aprendizaje fueron:

- Memoria técnica del proyecto.
- Software de la aplicación de escritorio
- Software de la aplicación web.

El proyecto de realizado en tercer curso consiste en la realización de un sistema inteligente con diseño de interfaces avanzadas y programación distribuida. Incluye las asignaturas de Inteligencia artificial, programación concurrente y distribuida e interfaces de usuario. Este proyecto se ha diseñado para estar ligado a un proyecto de investigación. En concreto este año se ha desarrollado una aplicación, CER-VANTA, que ayuda a enriquecer textos con contenido semántico de forma automática. El sistema es capaz de analizar un texto (una noticia, una entrada de un blog o un foro, un enunciado de una práctica,...), reconociendo de forma automática las entidades importantes que aparecen en el texto (personas, lugares, organizaciones,...), y enriquece el texto de esas entidades con información extra obtenida de la web, como por ejemplo imágenes, mapas de Google y videos, cuentas de Twitter, información de Wikipedia, etc. Los resultados de aprendizaje obtenidos son:

- Documentación con el proceso de desarrollo y el manual de usuario.
- URL donde está alojada la aplicación

2.2. Desarrollo, dificultades y soluciones

Dentro de la estructura organizativa de nuestra escuela existen varios roles que han jugado un papel importante tanto en el diseño como en la implementación del proyecto. Son la directora de la escuela y la directora académica, encargadas de dar coherencia y unificar criterios a nivel de escuela, y los responsables de los programas de cada una de las áreas, encargados de la coordinación de programas y profesores y de la garantía de calidad de los títulos. Existe además la figura del coordinador de asignatura para garantizar la coherencia y unicidad de criterios en aquellas asignaturas con varios grupos.

Para desarrollar este proyecto se estableció además la figura de coordinador de proyecto integrador, cuyas funciones principales son:

- unificar criterios entre los profesores que imparten las asignaturas implicadas en un proyecto, facilitando la resolución de conflictos
- garantizar la comunicación entre estudiantes y profesorado implicados mediante la creación y

organización de la sala virtual del proyecto integrador,

- recoger y analizar los datos que permitan medir la efectividad del proyecto integrador

Durante el desarrollo de los proyectos integradores en GII los coordinadores de cada proyecto integrador han estado en contacto permanente con los profesores implicados, y además se han realizado reuniones con la responsable de programa de la titulación y con la directora académica. A continuación se exponen las dificultades más relevantes detectadas, según la percepción de los coordinadores de proyecto, así como las decisiones adoptadas para solucionarlas:

- Las asignaturas que componen un mismo proyecto no siempre coinciden en el mismo periodo lectivo. Esto ha obligado en algunos casos a realizar entregas parciales evaluables en las asignaturas que finalizaban antes. Para solucionar este problema se diseñaron proyectos incrementales, y se pidió a los estudiantes como entregable en las asignaturas que finalizaban antes un documento que recogiese los resultados iniciales y que permitiese continuar el proyecto en el siguiente periodo. Además en la evaluación de esta entrega parcial participaban todos los profesores implicados en el proyecto. También participan todos los profesores en la evaluación de la entrega final.
- Los alumnos no están matriculados en todas las asignaturas que componen el proyecto integrador. Cada proyecto involucra a varias asignaturas del mismo curso, pero en cada una de ellas el grupo de alumnos matriculados puede ser distinto. La consecuencia es que los alumnos entran y salen de los proyectos en función de las asignaturas matriculadas. Para que cada grupo de trabajo realice el ciclo de vida completo del proyecto pusimos en práctica dos acciones: por una lado seleccionar los grupos de alumnos de forma que hubiera, al menos 1 estudiante que estuviese matriculado en todas las asignaturas; y, por otro lado, ser muy estrictos en cuanto a la documentación generada como resultado de las tareas correspondientes a cada asignatura. Esta última acción permitía que los alumnos que se incorporasen a un proyecto ya comenzado pudiesen ponerse al día.
- Problemas de trabajo en equipo. A los alumnos les han ocasionado muchos inconvenientes el trabajo en equipo: los grupos no han sido elegidos por ellos por lo que necesitaban trabajar con otros compañeros que no conocían previamente, en los grupos de trabajo había estudiantes que no tenían los conocimientos previos que se les suponía y no trabajaban al rendimiento esperado, los estudiantes más aventajados hacían modifi-

caciones sin avisar a sus compañeros, afloraban problemas de falta de responsabilidad etc. Este tipo de problemas de trabajo en equipo no son únicos de los proyectos integradores pero el hecho de ser más ambiciosos (más grandes, más dedicación del alumnos, más nota en la evaluación final, etc.) hacen que se intensifiquen. Como solución hemos realizado sesiones de mentoría grupales con los componentes del equipo para que ellos expusieran los problemas que surgían y plantearan posibles soluciones.

- Evaluación individual entre los componentes del equipo. Hay que evitar situaciones en las que un componente del equipo se aproveche del trabajo realizado por sus compañeros. El grado de implicación es desigual y esto debe reflejarse en la nota. En este aspecto en cada proyecto se han puesto en práctica estrategias distintas que van desde la defensa individual del proyecto hasta la modulación de las notas según el porcentaje de intervención que ellos mismos otorgan a cada componente del equipo.

3. Descripción del proceso de evaluación de la percepción de los estudiantes

Al no haber aún finalizado el primer curso académico de su implantación, es aún pronto para evaluar el impacto en las calificaciones finales de la utilización conjunta de este método en el grado en ingeniería informática. Se realizará este análisis una vez finalizado el periodo de convocatoria ordinaria.

Para evaluar la percepción de los estudiantes respecto a esta innovación, se han utilizado dos herramientas: en primer lugar se han utilizado cuestionarios en los que se utilizaban tanto preguntas cerradas (con escalas de tipo Likert) como preguntas abiertas para permitir los comentarios de los estudiantes con campos de observaciones. Además se ha realizado una entrevista grupal a estudiantes seleccionados

3.1. Cuestionarios

En el campus virtual de la escuela politécnica (construido con moodle) se creó un espacio específico para la recogida de datos, restringiendo el acceso por contraseña. Se proporcionaron las instrucciones para realizar el cuestionario a los estudiantes participantes en los proyectos integradores, incluyendo la URL y contraseña para acceder a la misma. Se trató la información recogida de manera anónima. El objetivo era recoger la opinión de los estudiantes en relación a los proyectos integradores desarrollados entre varias asignaturas, con el fin de mejorar su implantación el curso 2013-2014.

El cuestionario incluye una serie de cuestiones previas para determinar el perfil de los estudiantes y poder posteriormente correlacionar las respuestas en función de dicho perfil (número de proyectos integradores en los que participa, matriculado o no en todas las asignaturas que formaban parte de los proyectos en los que participa, participa o no en proyectos que duraban más de un trimestre, curso superior en el que está matriculado, es o no el primer año en la universidad). Estas preguntas iniciales son cerradas de alternativa simple o múltiple según el caso.

A continuación se incluye una serie de preguntas para medir (desde la perspectiva de los estudiantes) si se han conseguido los objetivos perseguidos: aumento de la motivación de los estudiantes (tanto en las asignaturas relacionadas con el proyecto como en la titulación en general), vinculación con el mundo profesional, refuerzo en el aprendizaje de los contenidos técnicos, inclusión de elementos relacionados con la sostenibilidad, refuerzo en la adquisición de competencias generales. Estas son preguntas cerradas de acuerdo a una escala Likert, y para cada una de ellas se incluye una pregunta abierta en que los estudiantes pueden incluir sus comentarios.

Por último, se lanzaban 3 preguntas abiertas (¿qué fue lo peor? ¿qué fue lo mejor? ¿cómo lo mejorarías?

El cuestionario se abrió durante un periodo limitado. Se fomentó la participación de los estudiantes mediante mensajes enviados a través de los espacios virtuales de los proyectos integradores y presencialmente en alguna sesión de las asignaturas implicadas.

3.2. Entrevistas

El objetivo de esta entrevista era dar respuesta a las siguientes preguntas desde el punto de vista del estudiante: ¿El uso de la metodología PBES en el GII prepara a los alumnos con éxito? ¿Están los alumnos más motivados cuando utilizan la metodología PBES? ¿Se obtienen mejores resultados académicos? ¿El alumno entiende el trabajo realizado como el que realizará en su futuro profesional? ¿Se mejoran las competencias transversales? ¿Qué cambios hay que hacer en esta metodología tal y como se ha puesto en práctica para conseguir mejorarla?

La entrevista se realizó a grupo de estudiantes seleccionados que participaron en el proyecto de 2º. Estos estudiantes fueron elegidos teniendo en cuenta los siguientes criterios: si el estudiante estaba matriculado de todas las asignaturas, si los resultados habían mejorado o empeorado, si se mostró durante el curso motivado o no. De esta forma los perfiles fueron:

- Alumno matriculado sólo de algunas asignaturas, que suspendió la asignatura del segundo trimestre (BD) a causa del proyecto (sus compañeros lo superaron pero su evaluación individual fue insuficiente).

- Alumno matriculado de todas las asignaturas del proyecto, cuyos resultados se han visto afectados de forma negativa por el proyecto, aunque lo superó.
- Alumno matriculado de todas las asignaturas menos una. Sus resultados no empeoraron a causa del proyecto pero realizó la mayor carga del trabajo (planteó problemas de grupo durante la realización del proyecto) para superar el mismo.
- Alumno matriculado sólo de algunas asignaturas del proyecto, cuyo resultado mejoró gracias al mismo y que se expresó como muy motivado durante la realización del mismo.

Se preparó una entrevista estructurada con el siguiente contenido:

Preámbulo: Introducción al PBES. (2 min) Se les explicó a los alumnos en qué consistió la metodología empleada y cuáles eran los objetivos de la entrevista.

Competencias específicas. (5 min) Se lanzó a los alumnos la pregunta: ¿Crees que has aprendido de forma más profunda los conocimientos técnicos de la asignatura?

Competencias genéricas. (5 min). Se les formuló la pregunta. “Otro de los objetivos que se persiguen con esta metodología es crear un entorno de trabajo que permita al estudiante desarrollar lo que se llaman las competencias genéricas o transversales. Ejemplos son: aprendizaje autónomo, búsqueda de información, Trabajo en equipo, Habilidades de comunicación, Planificación, etc. ¿Qué opináis al respecto?”

Acercamiento al mundo profesional. (5 min) A través de las preguntas: “¿Crees que el trabajo realizado en el proyecto es similar al que realizarás en tu futuro profesional? ¿Ves interesante este enfoque? Hemos trabajado con una empresa externa. ¿Te parece interesante?” Además uno de los profesores involucrados en los proyectos integradores planteó la inquietud sobre los derechos del software desarrollado. Para trasladarles la inquietud y que nos dieran su punto de vista se les plantearon dos preguntas más: ¿Ves bien desarrollar un proyecto para una empresa que podría ser comercializado? ¿Crees que la dinámica de la empresa, y sus intereses, pueden perjudicar el desarrollo del proyecto y tu formación?”

Motivación. (5 min) Se pretende medir la motivación del alumno en dos ámbitos: motivación del estudio de la asignatura (“¿Te cuesta más faltar a clase que en otras asignaturas que no incluyen PBES?”, “¿Has estudiado los contenidos de las asignaturas con más motivación?”) y motivación por la titulación (“¿Estás más motivado por continuar con el estudio de tu titulación?”).

Otros. (20min) ¿Crees que tus resultados han sido mejores o peores por causa del proyecto? (5 min) ¿Qué fue lo mejor? (5min) ¿Qué mejoras hay que hacer? (5min) ¿Se te ha quedado algo sin decir? (5min)

Se citó a los alumnos a una hora acordada, se grabó la entrevista, se transcribió y se realizó un análisis cualitativo de la misma, codificando categorías y subcategorías en función del contenido de las respuestas. Los resultados de dicho análisis se detallan en el siguiente apartado.

4. Análisis de resultados

4.1. Resultados cuantitativos

En este apartado se van a detallar los resultados preliminares obtenidos a través del cuestionario. En el momento de recoger los datos sólo se había solicitado respuesta a los alumnos participantes en los proyectos integradores de primer y segundo curso. El porcentaje de respuesta ha sido del 62.7% (de la población total de alumnos a los que se solicitó que realizaran el cuestionario, 51, han contestado un total de 32). Para el 50% de los alumnos este es el primer año en la UEM. Más de la mitad de los consultados (59,4%) participa en un único proyecto integrador, y el resto participan en los proyectos de dos cursos. El 75% de las respuestas indican que alguno de los proyectos en los que participan tiene una duración de más de un trimestre, y sólo la mitad (50%) tienen matriculadas todas las asignaturas implicadas en los proyectos integradores en los que han participado. De entre los alumnos que no son nuevos en la universidad, el 75% indica que sus resultados este curso son mejores que en cursos anteriores.

La escala utilizada para recoger la información en el resto de los ítems fue una escala Likert compuesta por los valores: 1=Totalmente en desacuerdo, 2=En desacuerdo, 3=Neutro, 4=De acuerdo, 5=Totalmente de acuerdo.

Se detallan a continuación las preguntas y las medias obtenidas:

- He estado más motivado en las asignaturas del proyecto que en otras que no utilizan este método: 3.5
- Realizar el proyecto me ha motivado para continuar con la titulación: 3.6
- Creo que el trabajo realizado en el proyecto se parece mucho a lo que haré en mi futuro profesional: 3.5
- El desarrollo de un proyecto para una empresa externa que podría, incluso, llegar a comercializarse, me parece una oportunidad: 4.0
- Desarrollar un proyecto ha facilitado el conocimiento y comprensión de los contenidos técnicos de la asignatura: 4.1
- La realización del proyecto me ha permitido reflexionar sobre las implicaciones económicas, sociales y/o medioambientales relacionadas con el desempeño de mi futura profesión: 3.2

A los estudiantes se les proporcionaba además el listado de las 18 competencias generales que forman parte del catálogo de competencias que se utiliza en la UEM a la hora de definir las que deben desarrollarse en los estudiantes, y se les preguntó sobre cada una de ellas. En este caso se incluía entre las opciones de respuesta NS (No sé) para distinguir aquellos casos en los que los estudiantes no identificaban o entendían la competencia sobre la que se le estaba preguntando.

A la pregunta: “Considero que realizar el proyecto me ha ayudado a mejorar en las siguientes competencias”. Las respuestas se presentan a continuación ordenadas en función de los resultados, indicando en primer lugar la de mayor puntuación:

- Toma de decisiones 4,0938
- Capacidad de adaptarse a nuevas situaciones 4,0645
- Utilización de las tecnologías de la información y las comunicaciones (TIC) 4,0625
- Responsabilidad 4,0625
- Capacidad de aplicar los conocimientos a la práctica 4,0625
- Iniciativa y espíritu emprendedor 3,9688
- Capacidad de análisis y síntesis 3,9355
- Resolución de problemas 3,9063
- Autoconfianza 3,9032
- Habilidades en las relaciones interpersonales 3,8667
- Gestión de la Información (búsqueda, selección e integración) 3,8438
- Aprendizaje autónomo 3,8000
- Innovación y Creatividad 3,7742
- Trabajo en equipo 3,7188
- Razonamiento crítico 3,7097
- Comunicación oral. Comunicación escrita 3,4839
- Planificación y gestión del tiempo 3,3871
- Conciencia de los valores éticos 3,1538

Como puede verse, la competencia que en mayor grado se ha mejorado (desde la perspectiva de los estudiantes) al desarrollar los proyectos integradores ha sido la toma de decisiones, mientras que la menos desarrollada ha sido conciencia de los valores éticos, seguida por planificación y gestión del tiempo. De hecho, al analizar las respuestas, puede verse que la moda es 4 para todas las competencias listadas excepto para conciencia de los valores éticos, en la que la respuesta mayoritaria fue el valor neutro (3). Esta es claramente un área a mejorar, que puede ser reforzada al hacer en futuros años un mayor hincapié en el estudio de las implicaciones sociales, medioambientales y económicas de los proyectos.

4.2. Resultados cualitativos

Del análisis cualitativo de la entrevista grupal se pueden extraer los siguientes resultados:

1. Respecto al desarrollo de las competencias específicas destacan:
 - Autoconsciencia de lo aprendido: son más conscientes de lo que aprenden en la asignatura y el nivel de profundidad en que lo hacen.
 - Reforzar la necesidad de aprender. Con el desarrollo de los proyectos integradores el alumno tiene la oportunidad de adelantar preguntas relacionadas con los objetivos específicos de la asignatura, lo que les permite estar alertas a las respuestas (creamos expectativas) y aprender a un nivel de profundidad mayor. El alumno es consciente desde el inicio qué tiene que saber para finalizar el proyecto, cuáles de estos conocimientos y habilidades tiene y cuáles no, lo que le hace recibir de forma activa los contenidos de las asignaturas.
 - Perdurabilidad de lo aprendido. Los estudiantes perciben que los proyectos integradores son una herramienta mucho más potente que las tradicionales, como exámenes, para obtener un aprendizaje más profundo y perdurable en el tiempo.
 - Generación de islas de conocimiento. Uno de los inconvenientes detectados por los estudiantes es la generación de islas de conocimiento debido a la especialización de cada componente del equipo en una tarea concreta del desarrollo del mismo.
2. En el tema del desarrollo de las competencias genéricas o transversales, su percepción es:
 - Trabajo en equipo. Ya tienen experiencia de trabajo en grupo puesto que están acostumbrados a realizar prácticas de forma colaborativa. Sin embargo, notan diferencia con los proyectos integradores pues supone mayor dificultad, la adquisición de roles más variados dentro del grupo y mayor organización y planificación entre los componentes del equipo.
 - Planificación. Destacan la necesidad de una buena planificación para sacar adelante el proyecto.
 - Responsabilidad. Le dan mucha importancia a la responsabilidad. La plantean desde dos puntos de vista: la necesidad de que todos los componentes del equipo lleven los aprendizajes necesarios actualizados para no retrasar el trabajo del grupo y la responsabilidad de defender el proyecto ante un tribunal. Describen varias situaciones en las que la falta de responsabilidad de componentes del grupo dio lugar a cargas de trabajo excesivas para el resto de compañeros del equipo y peores resultados.

- Comunicación. Destacan la puesta en práctica de esta habilidad para explicar el proyecto, negociar para la toma de decisiones, y llevar a cabo la coordinación del proyecto.
 - Autoaprendizaje. Consideran que sí han tenido que fomentar esta habilidad destacando como fuentes: otros compañeros, búsquedas de información, etc.
 - Sostenibilidad. Sí que han desarrollado esta competencia pero de forma muy tangencial, en algunos casos no han sido conscientes hasta que se planteó en la entrevista.
3. En lo que se refiere al acercamiento o vinculación con el mundo profesional, indican:
- Similitud proyecto-futuro profesional. Consideran que, aunque sea difícil de predecir, el proyecto se parece a lo que será su futuro profesional, al menos, en cuanto a desarrollo de competencias transversales se refiere. Además la mayoría de ellos destacaron este aspecto como lo que más les gustó del mismo.
 - Alejamiento proyecto-CE asignaturas. Se les planteó la inquietud de que el contenido del proyecto, al cubrir una necesidad real de una empresa externa, se alejara mucho de las competencias específicas de las asignaturas que lo cubrían. A este respecto comentan que, en algunos casos, necesitaban profundizar más en el conocimiento de las competencias específicas del nivel en que se explicaba en clase.
 - Derechos de explotación del proyecto resultado. Consideran que el hecho de trabajar para un proyecto de una empresa externa que finalmente pueda ser comercializado es más bien una oportunidad, como aprendizaje y como currículum. Además no consideran que el resultado pueda tener la calidad suficiente como para ser comercializado.
4. En relación con la motivación cabe resaltar:
- Motivación por el aprendizaje. Trabajar en el proyecto les hace ver la utilidad de los contenidos teóricos de las asignaturas y facilita el estudio.
 - Orgullo del resultado. Se muestran orgullosos del proyecto resultado, mostrándolo entre sus familiares.
 - Remedio abandono de primeros cursos. Consideran los proyectos como una herramienta útil para mostrar a los alumnos de primeros cursos el contenido real de su titulación evitando así abandonos en primeros cursos y, en caso de producirse, creen que sería realmente por una visión incorrecta de lo que se estudia en la titulación.
 - Visión globalizadora de los estudios. Consideran que el proyecto le da continuidad a los estudios.

Ven el final de un trimestre como “un punto y seguido” en lugar de como un “punto y aparte”.

5. Relacionado con propuestas de mejora, los alumnos han detectado:
- Distribución de la carga de trabajo: consideran que esta forma de trabajo conlleva un desplazamiento de la carga de trabajo al final de la asignatura por lo que pasan las dos últimas semanas de cada trimestre muy agobiados.
 - Información para la matrícula. Consideran que antes de realizar la matrícula es muy conveniente que el alumno sepa cuáles son las asignaturas que intervienen ese curso en proyectos (pueden variar de un año académico a otro).
 - Evaluación. Una propuesta de mejora es que las asignaturas que intervienen en el proyecto se evaluaran en base a éste.

5. Conclusiones

Como ya hemos indicado, este es un primer estudio previo a la finalización de los proyectos. Aún no se ha completado un curso académico desde la implantación de estos proyectos integradores, y no hemos recibido el feedback de los todos los estudiantes y profesores implicados. Sin embargo, pueden ya extraerse unas primeras conclusiones:

- Los estudiantes perciben los proyectos integradores como una herramienta para potenciar un conocimiento más profundo y perdurable de las competencias específicas, aunque, como contrapartida, detectan la generación de islas de conocimiento por la especialización de los componentes del equipo en tareas concretas.
- Consideran que han profundizado en el desarrollo de competencias transversales, si bien las que se destacaron en la entrevista grupal (trabajo en equipo, la comunicación y el autoaprendizaje) no coinciden plenamente con las destacadas por los estudiantes mediante el cuestionario (Toma de decisiones, Capacidad de adaptarse a nuevas situaciones, Utilización de las tecnologías de la información y las comunicaciones (TIC) , Responsabilidad, Capacidad de aplicar los conocimientos a la práctica). Es de resaltar la especial importancia que otorgan a la responsabilidad, han destacado que, en muchos casos, la falta de la misma dio lugar a malos resultados.
- La sostenibilidad se ha desarrollado de forma tangencial y debemos hacer un mayor hincapié en su desarrollo, lo que a la vez redundará en el desarrollo de los valores éticos.
- Valoran muy positivamente el desarrollo de proyectos reales respaldados por una empresa ex-

terna, estando de acuerdo en que les parece una oportunidad.

Esperamos a final del curso académico poder hacer un análisis más profundo de la experiencia relacionando resultados finales de los proyectos, de las asignaturas y perfiles de estudiantes.

Referencias

- [1] Amante, B., Lacayo, A., Piqué, M., Oliver, S., Ponsa, P., Vilanova, R., (2010). Evaluation of Methodology PBL Done by Students. Transforming Engineering Education: Creating Interdisciplinary Skills for Complex Global Environments, 2010 IEEE. Pages 1-21
- [2] Calvo, I., López-Guede, J.M., Zulueta, E., Aplicando la metodología Project Based Learning en la docencia de Ingeniería Técnica en Informática de Gestión. Revista de Formación e Innovación Educativa Universitaria. Vol. 3, Nº 4, 166-181 (2010).
- [3] Castellanos, L., Hernández, A., Goytisolo, R. (2011). Cómo Formar y Evaluar las Competencias a través de los Proyectos Formativos en las Disciplinas de las Carreras de Ingeniería. Latin american and caribbean journal of engineering education, Vol. 5(2), 2011, Pages 6-14.
- [4] Escoda Acero, M.L., Planella Morató, J., i Suñol Martínez, J.J. (2008). La Implantació de l'aprenentatge basat en problemes (APB-APP) en l'educació universitària: anàlisi d'una experiència'. A 'Experiències i materials d'innovació docent'. Girona: Universitat. Disponible en: <http://hdl.handle.net/10256/574>
- [5] Grant, M., (2002). Getting a grip on project-based learning: Theory, cases and recommendations. Meridian. Volume 5, Issue 1, Winter 2002 ISSN 1097 9778
- [6] Lee, N. (2009). Project methods as the vehicle for learning in undergraduate design education: a typology Design Studies, Volume 30, Issue 5, September 2009, Pages 541-560
- [7] Marquez, J.J., Martinez, M.L., Romero, G., Perez, J.M. (2011). New Methodology for Integrating Teams into Multidisciplinary Project Based Learning. International Journal of Engineering Education 01/2011; 27:746-756.
- [8] Machado, F., Borromeo, S., Malpica, N. (2009). Project Based Learning Experience in VHDL Digital Electronic Circuit Design. IEEE International Conference on Microelectronic Systems Education, 2009. MSE '09. Pages 49-52
- [9] Moesby, E. (2005a). Curriculum Development for Project-Oriented and Problem-Based Learning (POPBL) with Emphasis on Personal Skills and Abilities. Global Journal of Engineering Education, 9(2), 121-128.
- [10] Moesby, E. (2005b). Personal skills and abilities in curriculum development planning for Project Oriented and Problem-Based Learning (POPBL). Proceedings for 8th UICEE Annual Conference on Engineering Education (págs. 155 - 158). Kingston: UNESCO International Centre for Engineering Education (UICEE).
- [11] Nunes de Oliveira, J. M. (2011). Nine Years of Project Based Learning in Engineering. Revista de Docencia Universitaria, 9(1), 45 - 55.
- [12] Steiner, M., Ramírez, M.C., Hernández J.T., Plazas, J. (2008). Aprendizaje en Ingeniería basado en Proyectos, algunos casos. Ciencia e Ingeniería en la Formación de Ingenieros para el siglo XXI, Sección 2, Pages 129-147, ACOFI, Bogotá, Colombia.
- [13] Valero García, M., y García Zubia, J. (2011). Cómo empezar fácil con PBL. Actas de las XVII Jornadas de Enseñanza Universitaria de Informática. Sevilla: JENUI 2011.

Experiencia de aprendizaje de programación basada en proyectos

Laura Igual

Departament de Matemàtica Aplicada i Anàlisis
Universitat de Barcelona
08007 Barcelona
ligual@ub.edu

Xavier Baró

Estudis d'Informàtica, Multimèdia i Telecomunicació
Universitat Oberta de Catalunya
08018 Barcelona
xbaro@uoc.edu

Resumen

En este artículo se presenta una experiencia desarrollada los últimos cuatro años dentro del contexto de una asignatura de programación del primer curso del Grado en Ingeniería Informática. El objetivo principal de esta experiencia es que los estudiantes trabajen en un proyecto realista de programación que mejora su motivación e implicación en el proceso de aprendizaje. Los profesores facilitan una librería de código que permite a los estudiantes explotar sus conocimientos en los aspectos importantes de los paradigmas de programación para obtener resultados más realistas desde el primer día. En particular, se consigue visualizar una imagen o escuchar una canción en lugar de mostrar un simple mensaje por pantalla. Los enunciados de los proyectos están planteados para que aparezcan de forma natural las necesidades de aplicar ciertos conceptos y estrategias de programación. Esto permite introducir de forma práctica los conceptos básicos trabajados en las clases de teoría, como son los mecanismos de programación orientada a objetos, las nociones básicas de control de errores basado en excepciones y la gestión de eventos. De esta forma los estudiantes ven la utilidad de cada concepto asimilándolo mejor. Para facilitar el desarrollo de las prácticas, el proyecto se divide en diferentes fases incrementales con un enunciado asociado donde se facilita toda la información sobre los métodos a implementar así como la ayuda adicional para alcanzar los objetivos. A partir de esta experiencia se observa una gran implicación de los estudiantes en las prácticas y una alta correlación entre las calificaciones de los proyectos y de los exámenes teóricos. En el artículo se detallan los proyectos de programación implementados, su organización en etapas y división en entregas, así como los métodos de seguimiento y evaluación. Por último, se presentan los resultados académicos y de satisfacción del alumnado y se discuten los problemas encontrados y las posibles mejoras.

Abstract

This article presents an experience developed over the last four years within the context of a programming course of the first year of the Degree in Computer Engineering. The main objective of this experience is that students work in a realistic programming that enhances their motivation and involvement in the learning process. Teachers provide a library of code that allows students to exploit their knowledge of the important aspects of programming paradigms to obtain more realistic results from day one. In particular, they can visualize an image or hear a song instead of displaying a simple message on the screen. The statements of the projects are posed to make appear naturally the needs of applying certain concepts and programming strategies. This allows a practical way to introduce basic concepts presented in the lectures, as are the mechanisms of object-oriented programming, the basics of exception-based error handling and event management. In this way, students see the usefulness of each concept assimilating it better. To facilitate the development of the practices, the project is divided into incremental phases associated with a statement which give all the information on the methods to implement and further assistance to achieve the objectives. From this experience it is noted a great student involvement in practices and a high correlation between the qualifications of the project and the theoretical examinations. The article details the programming projects implemented, the organization in staged and division in deliveries, as well as the monitoring and evaluation methods. Finally, we present the academic results and student satisfaction results and discuss the problems encountered and the possible improvements.

Palabras clave

Experiencia de aprendizaje de programación, aprendizaje basada en proyectos, programación orientada a

objetos y a eventos.

1. Introducción

En el curso 2009-2010 se inició en la Universitat de Barcelona la implantación de los nuevos estudios del Grado de Ingeniería Informática dentro del marco del Espacio Europeo de Educación Superior (EEES). Se crearon nuevas asignaturas que antes no existían como tales, y que se tenían que planificar de nuevo siguiendo nuevos planes de estudios que abarcaran los retos planteados en el EEES. Este contexto de cambio se planteó como un momento idóneo para introducir una nueva metodología pedagógica reconocida como innovadora y exitosa, pero con poca presencia en los antiguos estudios de informática de la UB: “El Aprendizaje Basado en Proyectos” (ABP) [1], [2].

Hasta entonces, en los estudios de Ingeniería Técnica en Informática de Sistemas, la mayoría de las asignaturas de programación se estructuraban en clases magistrales de teoría y clases de prácticas donde se resolvían una serie de problemas pequeños o medios para trabajar los conceptos presentados. Dentro de los nuevos planes de estudios, los conceptos de la Programación Orientada a Objetos y Eventos se reestructuraron dentro de la asignatura de “Programación II” del segundo cuatrimestre del primer curso del Grado de Ingeniería Informática.

El valor de la estrategia ABP aplicada a la enseñanza de Programación II se centra en reproducir los pasos que se siguen en procesos de desarrollo de código, aquellos que se realizan cuando se plantean problemas reales. En consecuencia, el alumnado se enfrenta, aprende y desarrolla, un conjunto de habilidades que pueden ser de especial importancia en la formación de los futuros programadores.

En particular, nuestra propuesta de aplicar la metodología ABP en esta asignatura tenía los siguientes objetivos:

1. Mejorar la estrategia para una evaluación continuada del alumnado.
2. Aprender a utilizar los conocimientos adquiridos para resolver un problema de la vida real.
3. Aumentar la implicación y motivación del alumnado.
4. Aprender a trabajar con módulos de programación externos proporcionados por el profesorado.
5. Familiarizarse con el uso de herramientas informáticas de apoyo a la programación.
6. Obtener un producto final de programación perdurable y ampliable a lo largo de la carrera.

La metodología general que propusimos, se basa en plantear el desarrollo de un proyecto de programación pasando por diferentes etapas de la implementación,

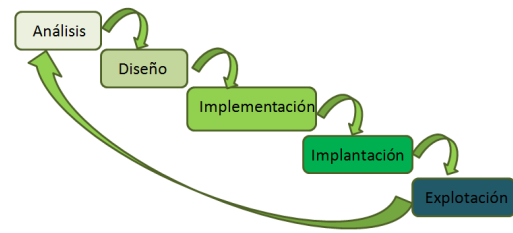


Figura 1: Ciclo de Vida del Software.

las cuales se tienen que alcanzar paulatinamente para llegar al objetivo final.

El resto del artículo se estructura de la siguiente manera: En el apartado 2 se contextualiza la experiencia realizada. En el apartado 3, se describe la metodología utilizada y se detalla el portafolio utilizado. En el apartado 4, se presentan los resultados obtenidos y las dificultades encontradas. Por último, en el apartado 5, se concluye el trabajo y se plantean posibles mejoras.

2. Contexto y experiencia

La asignatura de Programación II forma parte de la materia de programación integrada por un total de 5 asignaturas impartidas desde primero a cuarto curso del Grado de Ingeniería Informática de la UB. En esta materia se cubre todo el ciclo de vida del software o proceso de desarrollo de software en el que las necesidades del usuario son traducidas en requisitos de software, éstos transformados en diseño y el diseño implementado en código, seguido de la implantación y explotación de este código. Este último paso da pie a un nuevo análisis que cierra de nuevo el ciclo. En la Figura 1 se ilustra este ciclo.

La asignatura de Programación II se centra en el paso de implementación. En particular, se concentra en introducir nuevos conceptos de módulo y de abstracción de datos, técnicas de descomposición modular de programas, paradigmas de programación orientada a objetos y orientada a eventos [3]. Se explican los principios de los paradigmas de programación y su justificación, así como las ventajas y los inconvenientes de adoptar estos paradigmas en un proyecto de programación.

La asignatura de 6 créditos ECTS, contiene 60 horas de actividades presenciales organizados en clases teóricas y clases prácticas. Las primeras son clases magistrales para la introducción de conceptos teóricos y las segundas son clases realizadas en laboratorios de ordenadores, en los cuales se ponen en práctica los conocimientos adquiridos en las clases de teoría mediante una metodología docente basada en proyectos.

En esta asignatura, además de los objetivos específicos de aprendizaje, se desarrollan las siguientes competencias:

- Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad.
- Tener capacidad de abstracción: crear y utilizar modelos que reflejen situaciones reales.
- Tener capacidad para el razonamiento crítico y lógico.
- Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

3. Metodología y Portafolio

El ABP es una estrategia de enseñanza que invierte la organización tradicional de los procesos de aprendizaje. En los procesos tradicionales, en primer lugar y de forma muy diversa, se exponen los objetivos; a continuación, a través de un conjunto heterogéneo de actividades, se presenta la información pertinente y se trabaja el marco conceptual. Posteriormente, pero no siempre, se promueve la aplicación del conocimiento adquirido en el marco de la resolución de un problema. En cambio, en el Aprendizaje Basado en Problemas, primero se presenta el problema, y a partir de la delimitación de aquello que ya se conoce y de aquello que no se sabe, se establecen los objetivos, se identifican las necesidades de aprendizaje que son básicas, se diseña y se sigue un plan de actuación, para que a lo largo del proceso en que se produce el aprendizaje, se resuelva simultáneamente el problema.

Durante cuatro cursos seguidos (2009-2010, 2010-2011, 2011-2012, 2012-2013) se ha utilizado la estrategia ABP en las prácticas de la asignatura de Programación II. Se ha planteado un proyecto (problema real) a los estudiantes que se ha dividido en diferentes módulos, los cuales se tienen que ir desarrollando para llegar al objetivo final. Este objetivo final es la realización de un producto útil para el alumno que, además representa una aplicación perdurable y ampliable por él. De este modo se motiva al alumno con un problema atractivo desde el punto de vista de programación y de aplicación. Hasta ahora se han planteado dos proyectos de prácticas diferentes: un reproductor de música (2009-2010 y 2010-2011) y un visor de imágenes (2011-2012 y 2012-2013). Para la realización del proyecto de prácticas se ha utilizado el lenguaje de programación Java y el entorno de desarrollo integrado, NetBeans. Además, para asegurar un buen funcionamiento del ABP, se ha llevado a cabo un control exhaustivo y sistemático de las entregas periódicas, con un acotamiento y

seguimiento de las mismas. Esta gestión se ha realizado mediante la plataforma Moodle.

A continuación se presentan los detalles del proyecto de desarrollo de un visor de imágenes, que se ha basado en una biblioteca de imágenes formada por distintos ficheros y un conjunto de álbumes para organizar estas imágenes.

3.1. Definición de las entregas

El proyecto se presenta en un primer enunciado general. Cada uno de las entregas del proyecto viene guiado por un enunciado con las especificaciones necesarias.

1. Objetivos
2. Descripción de la entrega
3. Material para la entrega
4. Ayuda para la entrega
5. Forma de la entrega y contenido
6. Código con una estructura de paquetes determinada
7. Memoria con una serie de puntos marcados.

Estos enunciados se van publicando en el Campus Virtual y las entregas se hacen también a través del Campus Virtual los días antes de las sesiones de corrección. Cada enunciado viene acompañado por el material proporcionado para la entrega. Como parte del material, se proporciona a los estudiantes una librería con las funcionalidades necesarias para visualizar las imágenes y generar los eventos temporales. Tal como se indica en los enunciados, cada una de las entregas consiste en un código y una memoria que hay que preparar. El código corresponde a una versión cada vez más completa de un mismo visor de imágenes. Por esto, se nombra el proyecto igual que la entrega: Visor1, Visor2, Visor3 y Visor4.

Para ayudar a distribuir el trabajo, se divide el proyecto en cuatro entregas graduales, de forma que se debe finalizar una entrega para poder realizar la siguiente. El proyecto se diseña siguiendo el modelo MVC (Modelo-Vista-Controlador), y en cada entrega se avanza en las tres vertientes:

Entrega 1: Diseñar e implementar una aplicación con un menú de texto que nos permita gestionar ficheros dentro de una lista.

Entrega 2: Diseñar e implementar una aplicación con un menú de texto que nos permita organizar las imágenes dentro de la biblioteca de imágenes y los álbumes, así como visualizarlas.

Entrega 3: Implementar la visualización de una de las listas de imágenes, con las opciones de iniciar, parar y pausar la visualización, mediante el control de eventos.

Entrega 4: Implementar una interfaz gráfica que ten-

ga las mismas opciones que en el caso de la Entrega 3. Se trata de evidenciar las ventajas de un buen diseño, cambiando la vista de la aplicación sin necesidad de alterar el controlador y el modelo.

En la entrega 1 se repasan conceptos vistos en la asignatura previa de programación. Esta fase permite al estudiante situarse en el proyecto y comenzar una primera versión básica del visor. Aquí se comienza a incluir la utilización de estructuras de datos implementadas en la librería de Java. En la entrega 2 se implementan nuevas clases y aparece de forma natural la primera herencia con la que los alumnos trabajan (fichero y fichero de imagen). Además, se amplía la utilización de la librería de útiles para la funcionalidad de visualización de las imágenes. En la entrega 3 se amplía el número de funcionalidades posibles del visor. Esto complica la implementación que debe estar bien organizada de forma modular, tal como se indica durante el curso, para que sea fácilmente ampliable. Por último, en la entrega 4, no se amplía ninguna funcionalidad del visor, pero se cambia totalmente el aspecto del mismo, pasando de un menú de texto a una interfaz gráfica.

Las entregas son individuales debido a que así se ve necesario y se establece en todas las asignaturas de primero del grado. Con la finalidad de evaluar la progresión de los estudiantes, además de las entregas se realizan pruebas de validación. Estas pruebas permiten simplemente asegurar que los estudiantes han realizado realmente el trabajo y que éste no ha sido copiado. Esto constituye un peligro común, ya que todos los alumnos deben realizar el mismo proyecto.

Con la finalidad de evitar subjetividades en las correcciones, se definen unos criterios de evaluación específicos para cada entrega de prácticas y se corrigen todas las entregas siguiendo éstos criterios. Además es muy importante proporcionar un continuo retorno a los estudiantes. Para ello, se establece un plazo máximo de una semana para la corrección de cada entrega y se realizan comentarios sobre cada práctica mediante el Campus Virtual y personalmente, durante las sesiones de prácticas, si es necesario.

En el documento de la normativa de prácticas, que se explica el primer día de clase, se presentan los métodos y criterios de evaluación y se definen también unos requisitos mínimos para que la práctica sea considerada "Apta" y por tanto pase a ser evaluada y comentada. Estos requisitos son los siguientes:

- La entrega se realiza dentro del plazo indicado y en el formato especificado.
- El código entregado compila correctamente.
- El programa entregado obtiene todos los resultados que se piden en el enunciado.
- La documentación contiene todos los puntos pedidos en el enunciado.

De esta manera se reduce el trabajo del profesor que no tiene que corregir prácticas no acabadas.

Uno de los peligros de la metodología utilizada es el abandono de algún estudiante que no cumple con las fechas y acaba quedándose atrás. Por este motivo, desde el primer día, se dan unas pautas bien claras a los alumnos de cómo se tiene que trabajar dentro de la metodología utilizada. Se explica que es necesario ir al día, presentar cada entrega y seguir los comentarios y correcciones de los profesores para poder finalizar bien el proyecto. Se ha observado que la gran mayoría de los alumnos que realizan todas las entregas aprueban las prácticas. Para evitar que un estudiante que no pueda llegar a una entrega quede fuera del proceso de aprendizaje, los estudiantes disponen de una segunda oportunidad de entregar sus proyectos con una penalización.

3.2. Método de evaluación

La evaluación de la asignatura se hace a partir de la realización de dos exámenes parciales de contenido teórico, para calcular la nota de teoría, y las entregas del proyecto de prácticas y la realización de pruebas de validación individuales durante el curso, para calcular la nota de prácticas. Así, la nota final se calcula ponderando 50 % de la nota de prácticas y 50 % de la nota de teoría. De esta manera se le da la importancia merecida al esfuerzo que supone el proyecto realizado en las prácticas. Además, se ha observado que la gran mayoría de los estudiantes que aprueban las prácticas aprueban la teoría.

Para poder hacer el cálculo de la nota final se impone como condición indispensable que el estudiante obtenga una nota de prácticas y de teoría mayor o igual que 4.

Para aprobar las prácticas se impone como requisito indispensable tener un mínimo de 3 de las 4 entregas como "Aptas", tener notas de las dos pruebas de validación mayor o igual que 4 y tener una nota final mayor o igual que 4. Para recuperar una entrega que haya sido considerado "No Apta" o con nota inferior a 5, hay que volver a entregar todo el material pedido y una entrega recuperada puede obtener una calificación máxima de 7.

4. Resultados

Los resultados de esta experiencia son satisfactorios desde el punto de vista académico y de satisfacción de los estudiantes.

En el Cuadro 1, se presentan los resultados académicos de los cursos 2009-2010, 2010-2011 y 2011-2012. Se incluye el porcentaje de estudiantes que obtuvieron

	MH	S	N	A	SP	NP	Total
2011-2012	2.2 %	1.1 %	23.1 %	22.0 %	18.7 %	33.0 %	93
2010-2011	2.1 %	3.1 %	16.7 %	25.0 %	30.2 %	22.9 %	96
2009-2010	0.0 %	1.1 %	16.8 %	29.5 %	14.7 %	37.9 %	95

Cuadro 1: Resultados académicos: Porcentaje de Matriculas de Honor (MH), Sobresalientes (S), Notables (N), Aprobados (A), Suspensos (SP), No presentados (NP) y el total de estudiantes matriculados.

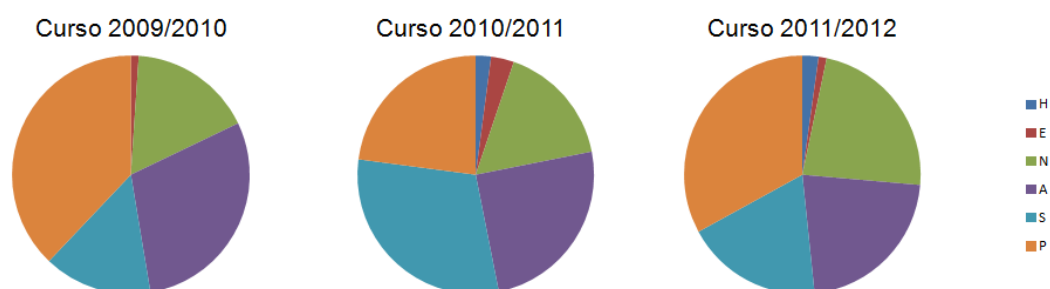
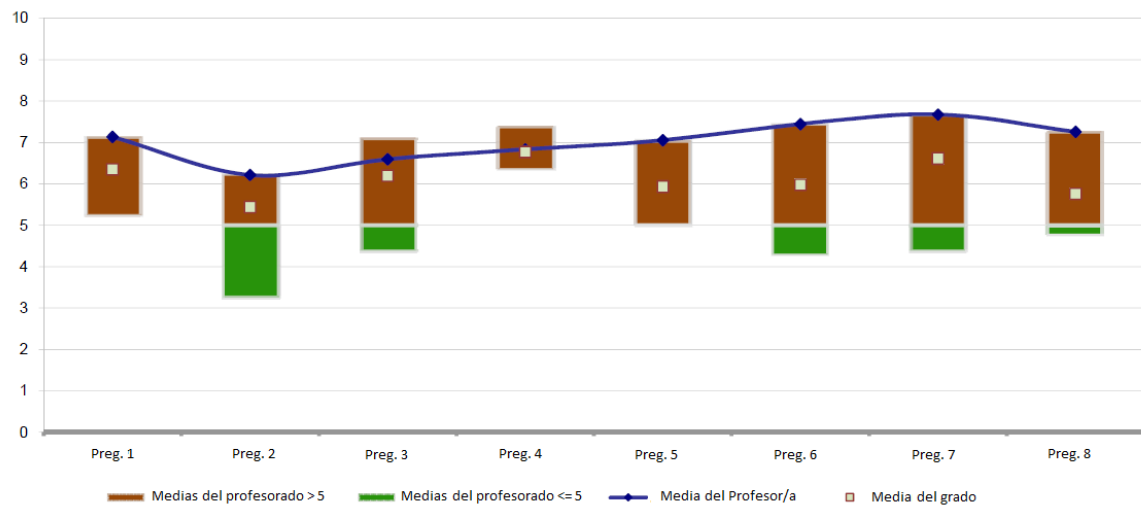


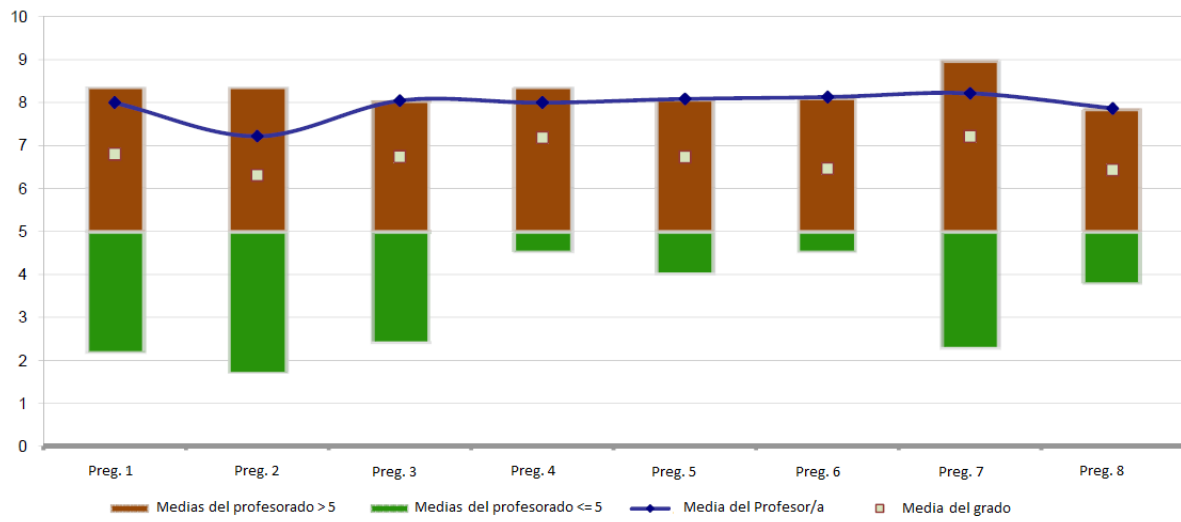
Figura 2: Gráfico de los resultados académicos de la asignatura.

Puntuación pregunta por curso.	Media (Des.) 2009-2010	Media (Des.) 2010-2011	Media (Des.) 2011-2012
1. En general, estoy satisfecho/a con la actividad docente llevada a cabo por el profesor/a de la asignatura	7,14 (2,06)	8,00 (1,95)	8,00 (2,74)
2. La manera de desarrollar la actividad docente consigue motivar el alumnado	6,22 (2,92)	7,22 (2,52)	8,40 (2,30)
3. Transmite con claridad los contenidos de la asignatura	6,59 (2,40)	8,05 (1,89)	8,60 (2,19)
4. La actividad docente ha permitido cumplir el programa de la asignatura	6,83 (2,16)	8,00 (2,13)	8,60 (2,19)
5. Las actividades de evaluación propuestas por el profesor/a han sido adecuadas a la asignatura	7,05 (2,48)	8,09 (1,76)	7,40 (2,30)
6. El material de estudio y de consulta propuesto (bibliografía, documentos, recursos didácticos, etc.) ha sido útil para el aprendizaje de la asignatura	7,44 (2,29)	8,13 (2,16)	8,80 (2,17)
7. Mantiene un buen clima de comunicación y relación con el alumnado	7,68 (2,33)	8,22 (2,19)	8,80 (2,17)
8. (En caso de que hayáis hecho) Las actividades propuestas por el profesor/a en el Campus Virtual han sido útiles en el proceso de aprendizaje de la asignatura	7,26 (2,37)	7,86 (2,29)	8,00 (2,65)
Nota media de todas las preguntas (porcentaje de respuesta)	7,03 (39,58 %)	7,95 (23,71 %)	8,33 (5,43 %)

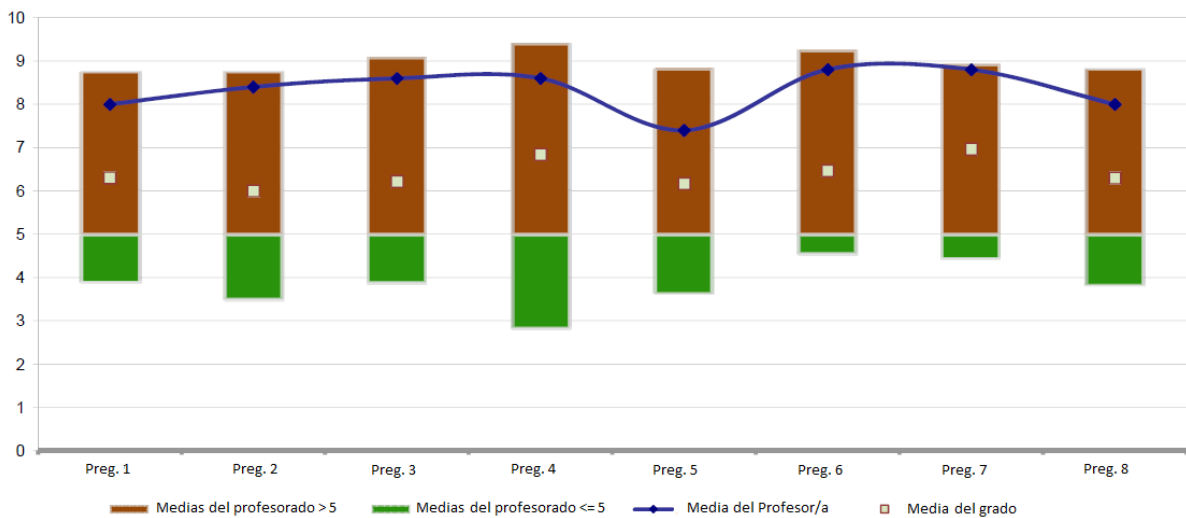
Cuadro 2: Detalles de las respuestas: Puntuación media (Desviación) de cada pregunta, puntuación media de todas las preguntas (porcentaje de respuesta).



(a) Curso 2009-2010



(b) Curso 2010-2011



(c) Curso 2011-2012

Figura 3: Gráfico de medias: situación del profesorado en el conjunto del Grado para las 8 preguntas de la encuesta.

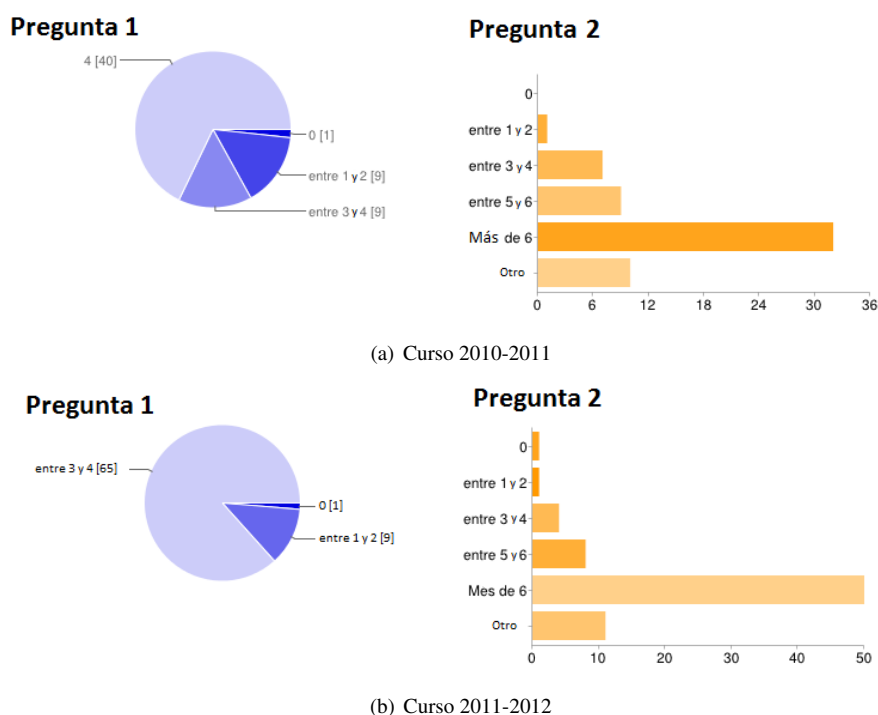


Figura 4: Resultados de las encuestas de dedicación. Preguntas: 1) ¿Cuántas horas semanales has asistido en promedio a las actividades presenciales de la asignatura (las últimas dos semanas)? y 2) ¿Cuántas horas semanales en promedio has dedicado al trabajo autónomo (fuera de las aulas) de la asignatura (las últimas dos semanas)?

Matriculas de Honor (MH), Sobresalientes (S), Notables (N), Aprobados (A), Suspensos (SP), No presentados (NP) y el total de alumnos matriculados. En la Figura 2 se puede ver un gráfico de estos mismos resultados.

Como se puede observar el número de estudiantes no presentados es elevado, pero no dista de los porcentajes de otras asignaturas de primer curso del grado. Este fenómeno puede ser debido a múltiples factores: 1) alumnos sin experiencia se matriculan de demasiadas asignaturas para ellos y tienen demasiada carga docente, 2) la asignatura de Programación II es la continuación de Programación I que tiene un número similar de no presentados, 3) cuando ciertos alumnos no adquieren los conocimientos de Programación I puede ser difícil seguir la asignatura. Además, somos conscientes que la metodología docente utilizada puede añadir al menos una dificultad. Esta consiste en que las entregas periódicas no son independientes entre si y, por ello, es necesario finalizar correctamente cada una de las entregas para realizar las siguientes.

Por último, cabe destacar que se ha observado que el 82,14 % de los estudiantes que realizan las prácticas aprueban la asignatura.

Además de los resultados académicos, es interesante analizar los resultados de satisfacción del alumnado.

Para ello, se utilizan las encuestas de opinión anónimas realizadas por parte de la Universidad cada curso. En particular, cada curso se ha realizado una encuesta de opinión que contiene ocho preguntas sobre la asignatura. Las preguntas se puntúan en una escala de valoración de 0 a 10 para indicar de "totalmente en desacuerdo" a "totalmente de acuerdo". Los resultados de los cursos 2009-2010, 2010-2011 y 2011-2012 se incluyen en el Cuadro 2. La primera columna especifica la pregunta realizada y las tres columnas restantes contienen la puntuación media y desviación de cada pregunta para los tres cursos. La última fila indica la puntuación media de cada curso, así como el porcentaje de respuesta realizadas ese curso.

En la Figura 3 se pueden ver los gráficos de las medias, para comparar la situación de la asignatura/profesor en el conjunto de asignaturas/profesorado del grado durante los tres cursos 2009-2010, 2010-2011 y 2011-2012. Se muestran las medias del profesorado con puntuaciones >5 y ≤ 5 (frangas marrones y verdes, respectivamente), la media del profesor de la asignatura (línea azul) y la media del grado (cuadrado beis).

A partir de estas encuestas y de conversaciones con los profesores y miembros del equipo de estudios, se puede afirmar que el proyecto de prácticas es atractivo

para los estudiantes y que los contenidos y el modo de trabajarlos resulta interesante para ellos.

Se realizaron también encuestas de dedicación, para saber la carga docente que suponía esta asignatura y si estaba dentro de las estimaciones que se realizan en el grado. Se formularon dos preguntas: 1) ¿Cuántas horas semanales has asistido, en promedio, a las actividades presenciales de la asignatura (las últimas dos semanas)? y 2) ¿Cuántas horas semanales, en promedio, has dedicado al trabajo autónomo (fuera de las aulas) de la asignatura (las últimas dos semanas)? Los resultados se pueden ver en la Figura 4. Estos resultados de las encuestas muestran que la carga de trabajo autónomo para la realización del proyecto de prácticas puede ser excesiva. Sin embargo, creemos que la valoración positiva de las encuestas de opinión compensa este sobreesfuerzo realizado.

5. Conclusiones y trabajo futuro

En este artículo se ha presentado la experiencia de aprendizaje de programación basada en proyectos para la asignatura de Programación II del grado de Ingeniería Informática. En esta experiencia, se profundiza en el desarrollo de las competencias de capacidad para resolver un proyecto con iniciativa, toma de decisiones, autonomía y creatividad. Las tareas a desarrollar dentro del proyecto de programación se delimitan y se proporciona material suficiente para realizarlo en los periodos establecidos. Mediante esta metodología se han conseguido unos resultados académicos satisfactorios y una valoración positiva por parte de los estudiantes. A pesar de esto, se ha observado que la carga docente es mayor que para otras asignaturas. Este es un tema que se debe estudiar y tratar de adaptar en el futuro.

En los cursos siguientes se adoptarán medidas para captar y evitar el abandono de la asignatura, especialmente para captar a aquellos estudiantes que debido a haber tenido problemas en la asignatura predecesora, no llegan a asistir a las sesiones prácticas. Además, se hará énfasis en el seguimiento y soporte de los estudiantes que presentan las entregas.

6. Agradecimientos

Este trabajo se inició con el proyecto de innovación docente *Análisis y implantación de la metodología de aprendizaje basado en problemas en la asignatura de programación II del grado de ingeniería informática* concedido y financiado por la Universitat de Barcelona en 2009.

Referencias

- [1] Pablo Sánchez and Carlos Blanco. Implantación de una metodología de aprendizaje basada en proyectos para una asignatura de Ingeniería del Software. In *Actas de las XVIII Jornadas de Enseñanza Universitaria de Informàtica (JENUI)*, páginas 41 – 48, Ciudad Real (España), Julio 2012.
- [2] Miguel Valero-García and Javier García Zubia. Cómo empezar fácil con PBL. Julio 2011. In *Actas de las XVII Jornadas de Enseñanza Universitaria de Informàtica (JENUI)*, páginas 109 – 116, Sevilla (España), Julio 2011.
- [3] Ian Sommerville. *Software Engineering*. Addison Wesley, March 2010.

Analizando los resultados académicos en una materia de programación

M. Cabrero Canosa, D. Alonso Ríos, N. Barreira Rodríguez, C. Gómez Rodríguez, B. Pérez Sánchez, N. Sánchez Maroño, E. Hernández Pereira, M. Alonso Pardo, B. Guijarro Berdiñas y J. Vilares Ferro

Departamento de Computación
Facultad de Informática
Universidad de A Coruña

{mariano.cabrero,dalonso,nbarreira,cgomezr,bperezs,nsanchez,
elena.hernandez,miguel.alonso,cibertha,jvilares}@udc.es

Resumen

En este trabajo se presentan las conclusiones extraídas tras evaluar los resultados académicos obtenidos por los estudiantes en la asignatura de Programación II, del primer curso del Grado en Ingeniería Informática en la Universidad de A Coruña. Los datos, pertenecientes al segundo año de implantación de la asignatura bajo las directrices del EEES, se han contrastado con los resultados obtenidos en el curso previo. Se han evaluado aspectos como el rendimiento y el éxito, se han investigado las razones del fracaso, y se ha cuestionado la utilidad de la segunda oportunidad (junio-julio) de evaluación. Por otro lado, se han demostrado útiles las propuestas de mejora del curso previo y se han planteado otras, para potenciar y valorar más el trabajo continuo del estudiante.

Abstract

We present in this work the conclusions obtained after evaluating the academic results of the subject Programming II, in the first course of Computer Engineering in the University of A Coruña. The results, obtained from the second year of the course implementation under the guidelines of the EHEA, were contrasted with those of the previous year. Aspects like performance and success were evaluated, the reasons for the academic failure were investigated, and the second opportunity test (june-july) is questioned. In addition, the improvement proposals for the previous course have been proven useful and some other improvements have been proposed for the next year, in order to promote and further assess the student's continued work.

Palabras clave

Evaluación, rendimiento, competencias.

1. Introducción

La materia de Programación II (ProII) es una asignatura de primer curso perteneciente al bloque de materias de Lenguajes y Programación del Módulo de Formación Básica del grado en Ingeniería Informática. Se imparte durante el segundo cuatrimestre a razón de 30h de docencia expositiva, 20h de docencia interactiva y 10h de tutorías en grupos reducidos. Mantiene una fuerte dependencia de las materias de Programación I (Pro I), Matemática Discreta y Informática Básica. A su vez, de ella dependen Algoritmos, Paradigmas de Programación, Diseño Software, Bases de Datos, Concurrencia y Paralelismo, Redes y Sistemas Inteligentes. La relación temporal entre todas ellas se muestra en la Figura 1.

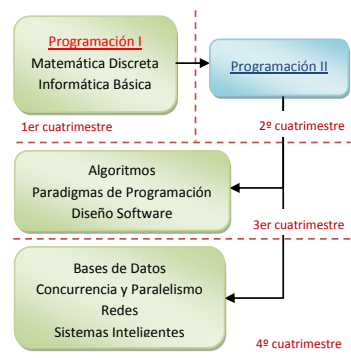


Figura 1: Relación temporal entre asignaturas fuertemente dependientes con ProII

El curso 2011/2012 fue el segundo año de impartición de esta titulación en la Universidad de A Coruña [1]. En este momento es cuando, tras la experiencia acumulada, se pueden empezar a extraer conclusiones significativas a partir de los resultados

académicos de los estudiantes, y así evaluar su rendimiento y nuestra capacidad docente, en aras de plantear las oportunas acciones correctoras.

Este artículo se estructura de la siguiente forma. El Apartado 2 detalla los objetivos y la metodología de enseñanza; el Apartado 3 analiza el rendimiento y el éxito de los alumnos durante los cursos 2010/2011 y 2011/2012; el Apartado 4 expone las mejoras propuestas para el curso 2011/2012; el Apartado 5 explica las nuevas propuestas para evaluar el trabajo continuo del alumno de forma justa y equitativa durante el curso 2012/2013; y el Apartado 6 expone las conclusiones extraídas del estudio.

2. Descripción de la asignatura

2.1. Objetivos y temario

Los contenidos orientativos de la asignatura de ProII son el manejo de punteros y memoria dinámica, introducción a los tipos abstractos de datos, y programación con estructuras de datos dinámicas y complejas [2]. Como resultado de su aprendizaje, se espera que los estudiantes, adquieran las siguientes competencias:

- Dominar mecanismos de gestión dinámica de la memoria.
- Conocer mecanismos de abstracción para el diseño de *Tipos Abstractos de Datos (TADs)*.
- Construir una especificación de un TAD y programar los algoritmos de manipulación de las estructuras de datos diseñadas para solucionar problemas reales.
- Interiorizar buenas prácticas de programación.

Los contenidos de la asignatura se distribuyen en cuatro partes. La primera parte trata sobre el uso de la memoria dinámica. En la segunda se presenta al estudiante el concepto de abstracción de datos y especificación de tipos, la base de los restantes temas de la asignatura. La tercera y cuarta parte analizan estructuras de datos lineales y no lineales.

Esta materia constituye una continuación de la asignatura ProI, impartida en el primer cuatrimestre del mismo curso, una primera toma de contacto del estudiante con las técnicas necesarias para acometer con éxito un problema de programación de tamaño reducido. Al igual que en ella se utiliza un enfoque de Programación Imperativa con el Pascal como lenguaje. Aunque la utilización de este enfoque es un tema ampliamente debatido [5,8], la experiencia docente en asignaturas similares de planes de estudios ya extinguidos siguen apoyando la idea de posponer al segundo año la enseñanza del paradigma de orientación a objetos.

2.2. Metodología docente

ProII se imparte durante 15 semanas, a razón de 4 horas semanales, todas ellas presenciales. Cada semana consiste en dos sesiones de 1 hora de teoría/problemas y en otra sesión de 2 horas de laboratorio o tutoría en grupo reducido. Los grupos de teoría/problemas son de unos 60 alumnos, los de laboratorio de unos 30, con un ordenador por alumno y, los de tutoría en grupo reducido de unos 15 alumnos. El total de alumnos matriculados en el curso 2011/2012 fue de 290, la mitad de segunda matrícula.

En las clases presenciales de teoría se describen los contenidos temáticos y los objetivos perseguidos, con una orientación totalmente práctica, requiriendo desde el primer momento la colaboración del alumno en el diseño e implementación de los TADs.

En las clases de tutorías en grupo reducido, para afianzar los conceptos teóricos, se presentan supuestos prácticos, resueltos inicialmente por el profesor para orientar a los alumnos. A medida que se avanza en el desarrollo teórico serán los alumnos, constituidos en grupos de trabajo, los encargados de su resolución y puesta en común. Este tipo de actividad en grupo promueve el desarrollo de habilidades útiles para la práctica profesional como la comunicación y el trabajo en equipo [4].

Las clases de laboratorio obligan a la construcción y utilización de estructuras de datos en un lenguaje de alto nivel para la resolución de un pequeño supuesto práctico. Este trabajo, de carácter obligatorio, se desarrolla en parejas para potenciar competencias transversales de la titulación como el trabajo en equipo, la toma de decisiones y la organización y planificación [2]. Este tipo de colaboración está avalada por diversos estudios que le atribuyen numerosos beneficios a nivel pedagógico para el estudiante: mejores calificaciones, mayor satisfacción/menor frustración y mayor confianza en los resultados del trabajo [9]. Durante estas sesiones prácticas, la labor del profesor será la de supervisión, resolución de dudas y corrección de errores, a fin de que el alumno pueda ir solventando pequeñas desviaciones de los objetivos inicialmente marcados antes de someter su trabajo a la evaluación final.

2.3. Evaluación

La evaluación continua del alumno, a través de una relación constante con su profesor durante el curso, resulta, en la Universidad en general, y en esta asignatura en particular, difícil de llevar a cabo. Al elevado número de alumnos matriculados se une el tratarse de una asignatura cuatrimestral lo cual reduce considerablemente el número de posibles pruebas parciales evaluativas. Nuestro proceso de seguimiento continuo de los alumnos se realiza a través de: a) las clases de solución de problemas, con grupos más

reducidos y una relación profesor-alumno más estrecha, y b) la estructuración del sistema de prácticas de laboratorio. En este sistema el profesorado establece una serie de hitos distribuidos a lo largo del cuatrimestre que el alumno debe alcanzar. En las fechas establecidas, el profesor comprueba si se han satisfecho los requisitos y puntúa el trabajo realizado por el alumno. La calificación global del trabajo práctico se pondera con las notas obtenidas en estos hitos. Quien no consigue el nivel mínimo en la oportunidad de junio puede mejorar su trabajo práctico y presentarlo en la de julio. En caso de no aprobar la asignatura, estaría obligado a realizar los trabajos prácticos planteados en el curso siguiente.

Teniendo en cuenta que esta asignatura tiene diferente número de créditos de carácter teórico, práctico y de resolución de problemas, su evaluación debe contemplar estos tres aspectos. Para el curso 2011/2012, en el cómputo total de la nota, la parte teórica tiene un peso del 75%, la práctica un 20% y la solución de problemas, el 5% restante. Se requiere una nota mínima en la parte teórica de 3,4 puntos para poder sumar las restantes categorías. La evaluación práctica califica el trabajo como APTO--, APTO-, APTO+ y APTO++, a razón de 0, 0,66, 1,33 y 2 puntos respectivamente. La nota de NO APTO implica el suspenso general en la asignatura. La resolución de problemas no tiene, en ningún caso, carácter eliminatorio.

3. Análisis de los resultados

Con el objetivo de sacar conclusiones válidas para proponer los cambios oportunos en la planificación docente de la materia, analizamos los resultados cosechados por los alumnos en el curso 2011/2012.

3.1. El problema de los no presentados

Como primer paso del análisis de resultados, comenzamos explorando el número de alumnos presentados en las dos oportunidades existentes. Cabe destacar que, en el curso 2011/2012, se ha conseguido un mayor número de presentados en la primera oportunidad que en el curso anterior. Así, frente al 46,6% del curso previo, un 57,9% de los alumnos matriculados en el siguiente curso ejercieron su derecho a examen en la primera oportunidad (Cuadro 1). Este dato se eleva hasta el 60,7% si consideramos las dos oportunidades de examen, a razón de 42,1% de alumnos que superaron la asignatura frente al 18,6% que no lo consiguieron. Se podría pensar que el origen de estos valores es la bolsa de alumnos de segunda matrícula. Sin embargo, esta suposición no es correcta, ya que los alumnos de nuevo ingreso se presentan mayoritariamente a la oportunidad de junio, 79,3%, frente a los de segunda matrícula, 36,6%.

Con estos datos, estableciendo una ordenación decreciente en cuanto al porcentaje de alumnos presentados, ProII ocupa la posición 7 con respecto al resto de asignaturas del primer curso de grado (10 en total). Estos resultados, que suponen una mejoría con respecto al curso anterior, indican que es posible progresar y que se debe seguir trabajando para alcanzar un mayor número de alumnos presentados.

Curso 2011/2012	Junio			Julio			Global		
	Nº	Nº/Pres.	Nº/Total	Nº	Nº/Pres.	Nº/Total	Nº	Nº/Pres.	Nº/Total
Suspenso	60	35,7%	20,7%	35	71,4%	19,2%	54	30,7%	18,6%
Aprobado	61	36,3%	21,0%	11	22,4%	6,0%	72	40,9%	24,8%
Notable	39	23,2%	13,4%	2	4,1%	1,1%	41	23,3%	14,1%
Sobresaliente	4	2,4%	1,4%	1	2,0%	0,5%	5	2,8%	1,7%
Matrícula de Honor	4	2,4%	1,4%	0	0,0%	0,0%	4	2,3%	1,4%
Superada	108	64,3%	37,2%	14	28,6%	7,7%	122	69,3%	42,1%
No superada	60	35,7%	20,7%	35	71,4%	19,2%	54	30,7%	18,6%
No presentado	122		42,1%	133		73,1%	114		39,3%
Total	290			182			290		

Cuadro 1: Resultados académicos del curso 2011/2012 sobre el total de presentados y el total de matriculados. Primera y segunda oportunidad

Por ello hemos tratado de ahondar en las razones del abandono, entendido éste como la renuncia *de facto* a examinarse de la asignatura. Aunque ya hemos dicho que hemos conseguido reducir el número de no presentados, todavía sigue suponiendo un número demasiado alto. Muchas de estas renunciaciones se producen incluso antes de iniciar el cuatrimestre, puesto que hay alumnos que ya no se incorporan como tales a la asignatura, y en principio, están muchas veces fuera de cualquier proceso evaluador [3]. Otro porcentaje de alumnos dejan de asistir a clase, ya sea porque prefieren concentrar su esfuerzo en otras asignaturas, más asequibles con su nivel de competencia o de conocimiento actuales, o porque sienten que su capacitación no es suficiente para afrontar con éxito la asignatura. Pocos estudios arrojan luz sobre este tipo de estudiantes [3,6], un efecto “inevitable” (o no tanto) de los antiguos y los nuevos planes de estudio de la universidad. Los informes que aclaran esta problemática la atribuyen, sorprendentemente y a tenor de la opinión de los alumnos, principalmente a causas externas como la organización de horarios de las asignaturas y, en menor medida, a la metodología docente [7].

Pese a estos antecedentes hemos tratado de analizar el problema. Teniendo sólo en cuenta los alumnos no presentados al examen final, el Cuadro 2 muestra la calificación obtenida en las prácticas (no entregada, entregada, suspensa, o aprobada). Se observa que un 70,5% de estos alumnos ya han abandonado al inicio de las clases puesto que no entregan ninguna práctica obligatoria. Excluyendo a esta gran mayoría, sólo un 9,0% de los alumnos no pueden presentarse en la primera oportunidad al no haber superado la evaluación de su primera práctica. Respecto al resto,

el 20,5%, que tienen una evaluación positiva en su primera práctica, un 8,2% aprueban su segunda práctica, un 3,3% suspenden y el 9,0% deciden no presentarla. Estas cifras contrastan fuertemente con el número de alumnos que han realizado y superado sus dos prácticas, 168, ya en la primera oportunidad.

Alumnos no presentados al examen (Junio 2012)					
Práctica 1			Práctica 2		
	No entrega		No entrega		
Aprueba	25	20,5%	No entrega	11	9,0%
			Suspende	4	3,3%
			Aprueba	10	8,2%
Suspende	11	9,0%	No entrega	10	8,2%
			Suspende	1	0,8%
			Aprueba	0	0,0%
Total no entregadas			86	70,5%	
Total entregadas			36	29,5%	
Total no presentados al examen			122		

Cuadro 2: Situación de prácticas (no entregada, entregada, suspensa, aprobada) de alumnos no presentados a ProII en el curso 2011/2012. Primera oportunidad

Alumnos suspensos en PROI Primera Oportunidad					
Calificación final PROII		JUNIO		JULIO	
No presentado	Sin practicas	70	51,9%	99	82,5%
	Con 1ª práctica sólo	12	8,9%	0	0,0%
	Con 2 prácticas	7	5,2%	0	0,0%
Suspense		31	23,0%	16	13,3%
Aprobado		12	8,9%	3	2,5%
Notable		3	2,2%	1	0,8%
Sobresaliente/M. de Honor		0	0,0%	1	0,8%
Total No presentados		89	65,9%	99	82,5%
Total Suspensos		31	23,0%	16	13,3%
Total Superados		15	11,1%	5	4,2%
Total alumnos		135		120	

Cuadro 3. Resultados obtenidos en ProII por los alumnos *suspensos* en la primera oportunidad de ProI del curso 2011/2012

Visto que uno de los posibles motivos de abandono es el no poder alcanzar las capacidades, competencias y aptitudes que el alumno debe adquirir en nuestra asignatura para resolver los supuestos prácticos propuestos, nos planteamos la posibilidad de que el problema también radique en no haber alcanzado las capacidades, competencias y aptitudes requeridas y que se suponen adquiridas previamente. Dado que la asignatura de ProI, es una materia clave para afrontar con garantías de éxito ProII, nos interesamos por los resultados cosechados por nuestros alumnos.

Como se puede observar en el Cuadro 3, de los 135 alumnos suspensos en la primera oportunidad de ProI, y que se encuentran matriculados en ProII, 70 no entregarían ninguna práctica. Podemos suponer, por tanto, que “abandonarían” la asignatura al considerar que no tienen los conocimientos necesarios para

afrontarla. En la oportunidad de julio el número de no presentados ya es de 99. Además, los resultados académicos obtenidos en ProII por parte de los alumnos suspensos en la primera oportunidad de ProI confirman la hipótesis: 65,9% de no presentados y 23,0% de suspensos, supone que sólo un 11% de estos alumnos aprueban la asignatura de ProII (Figura 2). Los resultados obtenidos por los alumnos que no se presentan a la primera oportunidad de ProI son todavía peores y en la misma línea que hemos comentado (Cuadro 4).



Figura 2. Resultados obtenidos en ProII (primera oportunidad) por los alumnos *suspensos* en la primera oportunidad de ProI del curso 2011/2012

Alumnos no presentados en PROI Primera Oportunidad				
Calificación final PROII		JUN		JUL
No presentado	Sin practicas	16		16
	Con 1ª práctica sólo	0	72,7%	0
	Con 2 prácticas	0		0
Suspense		2	9,1%	2
Aprobado		3	13,6%	0
Notable		1	4,5%	0
Sobresaliente/Matricula de Honor		0	0,0%	0
Total No presentados		16	72,7%	16
Total Suspensos		2	9,1%	2
Total Superados		4	18,2%	0
Total alumnos		22		18

Cuadro 4. Resultados obtenidos en ProII por los alumnos *no presentados* en la primera oportunidad de ProI del curso 2011/2012

3.2. Tasa de éxito

El segundo dato a destacar es el que concierne a las calificaciones obtenidas por los alumnos (Cuadro 1). Un poco más de un tercio de los presentados no consigue superar la asignatura en esta primera oportunidad, otro tercio sí que lo consigue con la nota mínima, y el resto obtiene mayoritariamente la calificación de notable, 23,2%. Si comparamos estos resultados con los obtenidos en el curso previo, podríamos esperar un incremento en todas las categorías, puesto que el número de estudiantes que se presentan en la primera oportunidad ha sido mayor. Sin embargo, esto no es así. Se produce una mejoría

en casi todas las calificaciones (Figura 3), especialmente importante en la categoría de suspensos, 9 puntos menor. Tan sólo el número de sobresalientes ha disminuido en 3 puntos.

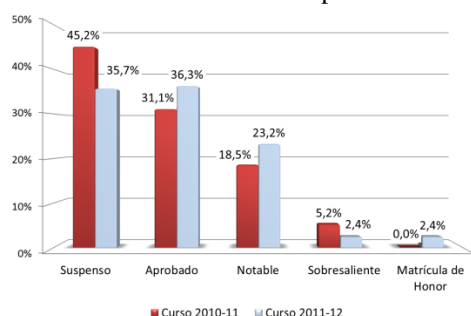


Figura 3. Resultados comparativos en ProII entre los cursos 2010/2011 y 2011/2012. Primera oportunidad

Separando el total de alumnos entre estudiantes de nuevo ingreso y estudiantes de segunda matrícula, resulta curioso comprobar cómo los primeros se presentan mayoritariamente a la oportunidad de junio, 79,3%, frente a los segundos, 36,6%. La misma proporción se mantiene en la segunda oportunidad, 40,0% frente a 19,7%. La razón podría ser que los alumnos repetidores ocupan su tiempo con asignaturas de otros cursos, que creen que podrían superar, frente a otras, como ProII, que exige una dedicación más continuada. Otro motivo podría ser el calendario, que es mucho más exigente con estudiantes con más asignaturas. Nuestra intuición, que a tenor de los datos no es la correcta, es que un estudiante de segunda matrícula debería tener un cierto dominio de la asignatura, lo cual supondría una ligera ventaja sobre el alumno de nuevo ingreso a la hora de superar la materia. Los datos de sus calificaciones vuelven a contradecirnos: un 68,7% de los alumnos de primera matrícula que se presentan superan la asignatura frente al 54,7% de los “repetidores” (Figura 4). Las diferencias aparecen entre las categorías de suspenso y aprobado, con mejores índices para los alumnos de nuevo ingreso.

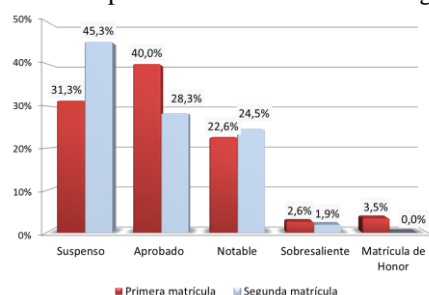


Figura 4. Resultados obtenidos en ProII por alumnos de primera y segunda matrícula en el curso 2011/2012 (sobre presentados de cada categoría). Primera oportunidad

Calificaciones		
Junio	Julio	Alumnos
Suspenso	Suspenso	30
	Aprobado	9
	Notable	2
	Sobresaliente	0
	No presentado	19
No presentado	Suspenso	5
	Aprobado	2
	Notable	0
	Sobresaliente	1

Cuadro 5. Calificaciones obtenidas en la segunda oportunidad por alumnos que no superan o no se presentan a la primera oportunidad de ProII del curso 2011/2012

Los resultados de los alumnos en las dos oportunidades de calificación también han sido objeto de análisis (Cuadro 5). Podemos comprobar cómo la mayor parte de los alumnos que no han conseguido aprobar en la primera oportunidad, tampoco lo hacen en la segunda; sólo lo consiguen 11 de los 41 suspensos en la primera oportunidad. De estos 11, 8 lo hubieran conseguido en la primera oportunidad de haber alcanzado en el examen la calificación mínima de 3,4 puntos, de la cual tan sólo les separaban 6 décimas de media. Todo ello nos vendría a confirmar que la gran mayoría de alumnos que superan la asignatura lo hacen porque están suficientemente preparados y lo consiguen en la primera oportunidad. Consideramos que quien acude a una segunda oportunidad debería hacerlo tras haber adquirido un cierto nivel durante el cuatrimestre y no haber conseguido superar la asignatura en la primera oportunidad por un escaso margen, merced probablemente a despistes o errores inoportunos y no por carencias de formación. Sin embargo esta suposición no es correcta ya que un buen número de los alumnos que acuden a la segunda oportunidad han obtenido una nota global comprendida entre 1 y 3 en la primera, un claro síntoma de falta de capacidades, aptitudes y competencias, que lógicamente no consiguen obtener en el período de tiempo que transcurre entre ambas oportunidades. Del resto de los datos mostrados en el Cuadro 5, tan sólo destacar el reducido número de alumnos (8) que sólo utilizan la segunda oportunidad, quizás por haber aprobado en julio las prácticas o por problemas con el calendario de exámenes de la oportunidad de junio.

A tenor de las calificaciones obtenidas por los alumnos en el apartado práctico, podemos decir que el método de evaluación continua anteriormente descrito sigue proporcionando buenos resultados. Además, aunque la contribución a la nota final no es grande, el esfuerzo realizado resulta beneficioso para el alumno porque contribuye a asentar los contenidos expuestos en las clases presenciales. Y aunque es preciso superarlas para poder aprobar la asignatura,

ello no constituye un obstáculo, pues sólo 5 alumnos no pudieron presentarse al examen en la primera oportunidad (Cuadro 2). Como puede verse en el Cuadro 1, una gran mayoría, 168, han conseguido la calificación mínima en el apartado práctico para poder optar a un aprobado en su nota global. Si comparamos las notas globales de prácticas obtenidas por los alumnos del curso previo con las obtenidas por los estudiantes del curso 2011-2012, los resultados son mejores puesto que un 60% de las notas se encuentran en el intervalo [1,5-2], frente al 47% del curso previo (Figura 5).

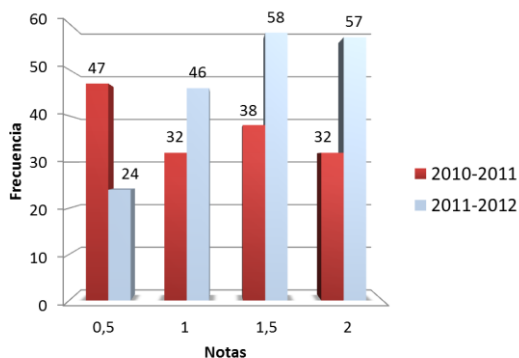


Figura 5. Distribución de notas globales de prácticas de alumnos de ProII. Cursos 2011-2012 y 2010-2011

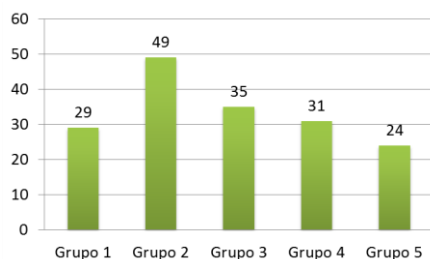


Figura 6: Alumnos por grupo de teoría presentados a la primera oportunidad de ProII del curso 2011/2012

Una hipótesis que quisimos contrastar con los datos es la aparente relación entre la tasa de éxito y el grupo de teoría. Observando el trabajo de los alumnos en el aula, el equipo docente percibía, aunque de forma subjetiva, que existían grupos aparentemente “mejores” que otros. Como puede observarse en los datos que aparecen en la Figura 6 y Figura 7, el grupo 2 de teoría es el que mejores resultados cosecha, tanto en número de alumnos presentados como en alumnos aprobados, seguido de cerca por el grupo 1. No encontramos ninguna razón objetiva para ello más que el horario asignado, más atractivo que otros para el alumno. Dado que la asignación de un alumno a un grupo se basa en sus preferencias y se prioriza según su nota de acceso a la titulación, parece lógico que los

mejores expedientes sean los que se encuentran en dichos grupos.

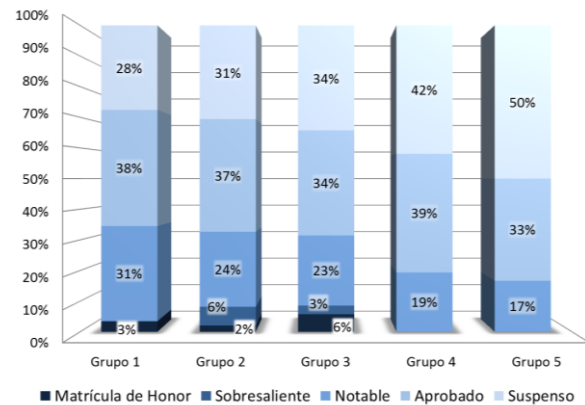


Figura 7. Resultados académicos por grupo de teoría en ProII del curso 2011/2012. Primera oportunidad

4. Propuestas para el curso 2011/2012

El curso 2010/2011 fue el inicio del grado en Ingeniería Informática en la Universidad de A Coruña. Tras analizar los resultados cosechados por los alumnos, constatamos la excesiva rigidez del método de calificación de trabajos prácticos, por cuanto sólo permitía asignar 3 tipos de calificaciones, APTO-, APTO y APTO+, a razón de 0, 0,5 y 1 punto respectivamente. En el curso 2011/2012 pusimos en práctica un nuevo esquema de evaluación tal y como se detalla en la sección 2.3. Creemos que el cambio ha sido beneficioso para el alumno, que ahora tiene una calificación mucho más ajustada. Para confirmar si estábamos en lo cierto comprobamos qué habría pasado si, a los alumnos del curso actual les aplicásemos el criterio del curso previo, es decir, el esquema de 3 calificaciones en sus prácticas. Cada nota práctica del alumno se redondeó al entero más cercano: así una nota global de prácticas de 1,33 se convertía en 1 y una nota de 1,66 en 2. Con este ajuste, se calcularon las diferencias en la calificación final entre la nota real y la nota corregida con esta modificación. Como puede intuirse, hubo alumnos que subieron su nota y otros bajaron merced al redondeo. Si bien el número de alumnos que consiguen mayor nota supera al número que obtiene peor calificación, en términos de las cuatro categorías clásicas de suspenso, aprobado, notable y sobresaliente, los resultados habrían sido peores. No hemos podido encontrar ninguna otra medida estadística que corrobore la percepción subjetiva del profesorado de que este esquema califica de forma más justa el trabajo práctico.

5. Propuestas para el curso 2012/2013

Tras analizar los resultados académicos en términos de tasa de rendimiento y éxito, en los dos cursos mencionados, el grupo de docentes de esta asignatura ha decidido como propuesta de mejora la modificación de los pesos de las distintas actividades en el esquema de evaluación de la asignatura. La idea es reducir el peso de la prueba teórica en favor de otras actividades que evalúen el trabajo continuo del alumno. Para comprobar cómo afectaría este cambio en las calificaciones, utilizamos los resultados reales de la primera oportunidad del curso 2011-2012. Sorprendentemente, si disminuimos la importancia del examen teórico en la nota final en favor de las tutorías o de las prácticas, obtendríamos resultados mucho peores. Por ejemplo, con un esquema 65% (teoría) + 30% (prácticas) + 5% (tutorías), un 67% de las notas empeorarían; con 65% (teoría) + 20% (prácticas) + 15% (tutorías) las notas empeoran mucho más, un 84%. Un análisis superfluo nos llevaría a la conclusión de que los alumnos no se adaptan a este nuevo esquema de Bolonia ya que arriesgan sus posibilidades de éxito en la asignatura a la realización del examen teórico y no dedican el tiempo necesario durante el curso para ir preparando la materia y obtener buenos resultados en los otros aspectos evaluables. Sin embargo este comportamiento es hasta cierto punto entendible puesto que la contribución a la nota final del examen es muy superior a la del resto de actividades puntuables. Empleando un sistema de ponderación diferente, el alumno plantearía su aprendizaje de forma distinta, y el resto de actividades serían más decisivas en la calificación final.

A tenor de lo expuesto, y decididos a aumentar el protagonismo de las actividades, nos centramos en las tutorías en grupo reducido, cuya contribución a la nota final es mínima (0,5). Para ello comprobamos en primer lugar en qué medida resultan decisivas para obtener una calificación u otra y, en segundo lugar, si no lo fueran, cómo podríamos cambiar esto. La Figura 8 muestra el número de alumnos cuya calificación final se incrementa considerando la nota obtenida en las tutorías en grupo reducido, y el número de alumnos cuya calificación final se incrementa de haber obtenido la puntuación máxima de 0,5 puntos. Como vemos, la aportación a la calificación final de la nota obtenida en las tutorías en grupo reducido es mínima: sólo 22 alumnos incrementan su calificación y de ellos, 5 consiguen aprobar. Considerando que la nota obtenida en las tutorías fuese la máxima tampoco se observa una mejora significativa en los resultados. Todo ello confirma que el alumno se centra en aquello que premia en mayor medida su esfuerzo, el examen, a

pesar de que las tutorías refuerzan los conocimientos ya adquiridos, un beneficio a priori no tan tangible.

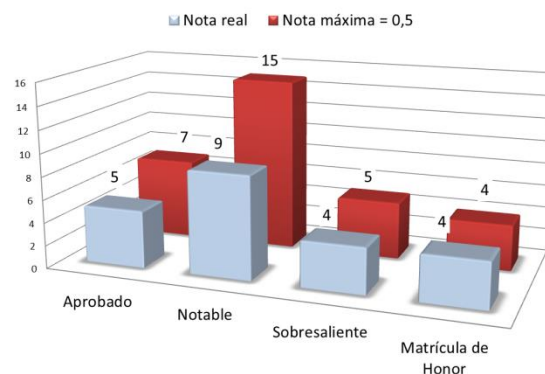


Figura 8. Aportación de la nota máxima y la obtenida en tutoría en grupo reducido en la nota final de ProII del curso 2011/2012. Primera oportunidad

Quedando patente la poca influencia de la nota de tutorías de grupo en la calificación final, estudiamos para el curso 2012/2013 una modificación en el esquema de evaluación. Dar mayor importancia a la nota obtenida en las tutorías en grupo reducido, supondría reducir alguna de las otras dos pruebas evaluables, el examen o las prácticas. Así se valoraría hasta un 10% la nota de las tutorías en grupo reducido, y dicho porcentaje se restaría al 80% de contribución del examen a la calificación final. De esta forma, también se reduciría la nota mínima necesaria en el examen para sumarla con las notas obtenidas en las prácticas obligatorias y las tutorías en grupo reducido. En resumen, la valoración sería 20% prácticas, hasta un 10% las tutorías y hasta un 80% el examen de teoría.

En este nuevo esquema, un alumno cuya evaluación en las tutorías no fuera positiva, podría conseguir la calificación máxima en la asignatura, puesto que la valoración del examen sería máxima (80%). Por el contrario, un alumno que obtuviera buenos resultados necesitaría menor nota en el examen para conseguir la calificación deseada, puesto que su importancia es menor (hasta un 10% menos).

Para apoyar esta propuesta decidimos volver a recurrir a los resultados de la primera oportunidad del curso 2011/2012. Aumentamos en un 5% las calificaciones de las tutorías en grupo reducido y redujimos en la misma medida la contribución de la nota obtenida en el examen, así como la nota mínima requerida para efectuar la media. Tras la aplicación de este criterio, las notas son prácticamente idénticas, incrementándose sólo en 7 casos. En promedio subirían en 0,18 puntos de media y la aportación del examen en la calificación final se reduciría hasta el 76%. En todo caso, no se empeoraría. Realmente el objetivo no era subir las notas de los alumnos, sino

que el método pueda servir como acicate para que se impliquen más en el trabajo realizado durante las tutorías en grupo reducido, porque de esa forma quizás sea más fácil superar el examen. Y todo lo que no pueda obtenerse en ellas, podría recuperarse en el examen. En la misma medida, obligaría al profesorado a ser más meticuloso y selectivo con las pruebas desarrolladas en las tutorías. Una reflexión detallada hace ver que esta aproximación no es tan ventajosa. Por ejemplo, un alumno con nota máxima en las prácticas y 0 en las tutorías, tendría que obtener la máxima calificación en su examen (valorado en un 80%) para obtener un 10. Sin embargo, un alumno con una valoración positiva en las tutorías y las prácticas tendría que obtener también la máxima calificación en su examen (valorado en un 70%) para alcanzar la misma nota final de 10 puntos. De esta forma, si bien se facilita que un alumno compense con el examen el no haber obtenido una nota positiva en las tutorías, no se recompensa en el examen el esfuerzo de un estudiante que sí la ha obtenido.

Así pues, y para corregir esta deficiencia, el nuevo esquema de evaluación durante el curso 2012/2013 mantendrá en un 20% el peso de las notas de prácticas en la calificación final y subirá al 80% la contribución del examen y al 10% la de las tutorías. De esta forma, se premiará el esfuerzo realizado durante las tutorías ya que la nota obtenida se suma a las otras dos y, a la vez, se permite que alguien que no haya realizado ese esfuerzo pueda, a través del examen, compensarlo. La nota mínima requerida en la parte teórica para promediar con las restantes notas será de 3,6. Con este esquema, se realizó el estudio anteriormente mencionado sobre los resultados de la primera oportunidad del curso 2011/2012. Las notas se incrementarían de media en 0,43 puntos, con 33 estudiantes que mejorarían su calificación final, destacando aquellos que alcanzarían la calificación de notable, 20. A la vista de esta proyección adoptaremos esta modificación en el curso actual.

6. Conclusiones

Reflexionar sobre los resultados cosechados por nuestros estudiantes debería ser una práctica habitual, no sólo para detectar fortalezas y debilidades en nuestro proyecto docente sino para buscar las soluciones que permitan corregir las deficiencias. Con el objetivo no solo de mejorar las tasas de éxito, sino con el propósito de formar mejor a nuestros alumnos, hemos realizado un pequeño trabajo de introspección en la materia de ProII, una asignatura fundamental en primer curso de cualquier grado en informática. Hemos constatado que la fuerte dependencia con ProI explica los altos índices de abandono de la asignatura. En cuanto a la tasa de éxito, a pesar de que los estudiantes de segunda

matrícula demuestran peor nivel de competencia que el resto, es mejor que la del curso previo. Por último, hemos constatado que la segunda oportunidad de examen es precisamente eso, una segunda opción para quien, habiendo desarrollado un trabajo continuado, no supera la asignatura, por escaso margen, en el primer intento. A la vista de todo ello, hemos propuesto las modificaciones oportunas en nuestro proyecto para el curso 2012/2013.

Referencias

- [1] B.O.E. núm 278 de 18 de Noviembre de 2011, pp. 121018-121029. Resolución de 5 de Octubre de 2011, de la Universidad de A Coruña, por la que se publica el plan de estudios conducente a la obtención del título de Graduado en Ingeniería Informática.
- [2] Memoria del grado en Ingeniería Informática de la Universidad de A Coruña. Disponible en <http://www.fic.udc.es/files/23661/23661ingenieriainformaticaUDCv3.pdf>.
- [3] Cernuda A., Hevia S., Suárez M. C., y Gayo D. Un estudio sobre el absentismo y el abandono en asignaturas de programación. En *Actas de las XIII Jornadas de Enseñanza Universitaria de Informática*, pp. 487–494, Alicante, Julio 2007.
- [4] Crosby, J. Learning in small groups. *Medical Teacher*, 18(3): 189–202, Enero 1996.
- [5] García Molina, J. ¿Es conveniente la OO en primer curso de programación? En *Actas de las VII Jornadas de Enseñanza Universitaria de la Informática*, pp. 293–298, Palma de Mallorca, Julio, 2001.
- [6] Rodríguez, R., Hernández, J., Diez-Itza, E., y Alonso García, A. El absentismo en la Universidad: resultados de una encuesta sobre motivos que señalan los estudiantes para no asistir a clase. *Aula Abierta*, 82:117-146, 2003.
- [7] Sanz, I., Aramburu, M., Museros, LL., Pérez, M. y Barrachina, C. En busca del estudiante perdido: caracterización de los “no presentados”. En *Actas de las XVII Jornadas de Enseñanza Universitaria de Informática*, pp. 403–410, Sevilla, Julio 2011.
- [8] Vilner, T., Zur, E., y Gal-Ezer, J. Fundamental concepts of CS1: procedural vs. object oriented paradigm—a case study. *ACM SIGCSE Bulletin* 39:171-175, Sept. 2007.
- [9] Williams, L., Wiebe, E., Yang, K., Ferzli, M. y Miller, C. In Support of Pair Programming in the Introductory Computer Science Course. *Computer Science Education*, 12:197-212, 2002.

Ingeniería del Software I: Aprendizaje Basado en Proyectos mediante metodologías Ágiles

Alfredo Goñi, Jesús Ibáñez, Jon Iturrioz, José Ángel Vadillo

Dpto. de Lenguajes y Sistemas Informáticos
Universidad del País Vasco
San Sebastián

(alfredo, jesus.ibanez, jon.iturrioz, vadillo)@ehu.es

Resumen

En este trabajo se presenta el proyecto docente para la asignatura Ingeniería del Software I (segundo curso, segundo cuatrimestre) siguiendo la metodología del Aprendizaje Basado en Proyectos (ABP). Para su implementación, proponemos la utilización de técnicas ágiles ampliamente utilizadas en el entorno profesional para el desarrollo de proyectos software. Nuestra hipótesis se basa en que estas metodologías recogen en su propia esencia los fundamentos del ABP, y por tanto, su correcta utilización garantiza los objetivos de aprendizaje propuestos. Durante la asignatura, los alumnos desarrollarán un proyecto aplicando el proceso unificado de desarrollo de software (UP) encuadrado en Scrum como metodología ágil. La elaboración del proyecto se realizará en grupos que, a lo largo de tres iteraciones o sprints, deberán ir explorando las herramientas de modelado y las tecnologías de implementación apropiadas. La realización del proyecto comenzará prácticamente el primer día de clase, y su evaluación supondrá el 75% de la calificación final. La estructura iterativa del desarrollo del proyecto proporcionará, además, un esquema de aprendizaje incremental que resulta muy adecuado para asimilar los conceptos de la Ingeniería del Software.

Abstract

This paper presents the teaching project for the subject Software Engineering I (second year, second semester) based on Project-Based Learning (PBL) methodology. For its implementation, we propose the use of agile techniques widely applied in the software development industry. We believe that agile methodologies do capture the very principles of PBL, and its use can therefore guarantee the acquisition of the proposed learning goals. During the course,

students will develop a project using the unified software development process, or simply Unified Process (UP), encompassed in the agile methodology Scrum. The elaboration of the project will be undertaken in teams that will need to explore the appropriate modeling tools and implementation technologies along three successive iterations or sprints. The project will be in place almost from the first day of class, and its assessment will compute for a 75% of the final grade. Additionally, the iterative structure of the project development will entail an incremental learning framework that results especially useful to assimilate the concepts involved in Software Engineering.

Palabras clave

Ingeniería del Software, Aprendizaje Basado en Proyectos, metodologías ágiles, Scrum.

1. Introducción

La introducción al proceso de desarrollo del software suele incluirse entre el cuarto y el sexto cuatrimestre en las diversas enseñanzas de Grado de Ingeniería Informática, y la duración del curso introductorio correspondiente es variable. La asignatura objeto de este artículo, Ingeniería del Software I, es de las que podemos llamar tempranas (se imparte en el cuarto cuatrimestre del del Grado en Ingeniería Informática en la UPV/EHU) y cortas (cuenta con seis créditos). El estudiante ha cursado previamente tres asignaturas del área de programación y estructuras de datos: un curso de Programación Básica de iniciación a la programación imperativa (1^{er} cuatrimestre), otro de Programación Modular y Orientación a Objetos (2^o cuatrimestre) y otro de Estructuras de Datos y Algoritmos (3^{er} cuatrimestre). En el momento de llegar a Ingeniería del Software I el estudiante ya cuenta unas

habilidades de programación en Java razonablemente desarrolladas, dominio de Eclipse y nociones básicas de diseño de algoritmos eficientes.

Se pretende que al completar la asignatura de Ingeniería del Software I se haya producido un salto cualitativo, en el sentido de que el estudiante sea capaz de abordar y construir una aplicación siguiendo los principios de Análisis y Diseño Orientado a Objetos (OOA&D), utilizando de manera provechosa diversas herramientas de modelado e implementando su solución mediante una arquitectura cliente-servidor en tres niveles y un código que respete los patrones de software más habituales.

En el momento de incorporarse a la asignatura, el estudiante no tiene aún nociones de programación de interfaces de usuario ni de utilización de mecanismos de persistencia. La asignatura Bases de Datos, en la que se estudia el modelado de datos y su paso al modelo relacional, es impartida en el mismo cuatrimestre. Naturalmente, la programación distribuida también le es ajena. Esto plantea formidables obstáculos de tipo tecnológico, que hemos intentado superar maximizando el número de problemas resolubles con el mínimo esfuerzo de adaptación a las exigencias de las técnicas utilizadas. Como resultado de algunos años de experiencia con asignaturas del área hemos seleccionado un conjunto de soluciones tecnológicas que respondan lo mejor posible a un mismo modelo:

- Lenguaje de programación Java.
- Entorno de programación Eclipse.
- AWT/Swing para el diseño de interfaces gráficas de usuario (GUI)
- Db4o y Object Manager para implementar la persistencia en bases de datos orientadas a objetos.
- RMI para desarrollar aplicaciones distribuidas.

Para conseguir una buena curva de aprendizaje hemos desarrollado una serie de laboratorios, concentrados en la primera mitad de la asignatura, que permiten a los estudiantes plantear y resolver la mayor parte de las dificultades técnicas aparejadas.

- **L1:** Arquitecturas por niveles
- **L2:** Iniciación a AWT/Swing
- **L3:** Herramientas para el modelado del análisis de requisitos
- **L4:** Bases de datos orientadas a objetos (*Db4o*)
- **L5:** Herramientas para el diseño de los casos de uso (diagramas de secuencia)
- **L6:** Lógica de negocio distribuida (*RMI*)

Una vez resuelto el problema tecnológico queda la parte fundamental, es decir la metodológica. También en este caso tenemos que ser avaros con la metodología de desarrollo, centrándonos en los siguientes aspectos:

- Análisis y diseño orientado a objetos.
- El Proceso Unificado de Desarrollo del Software (UP) como ciclo de vida de referencia, centrándonos en las disciplinas de Requisitos, Diseño e Implementación.
- Artefactos UML: diagrama de casos de uso, diagrama de clases, flujo de eventos y diagrama de secuencia.
- StarUML como herramienta de modelado.

En clase se utilizan numerosos ejemplos, tanto en el aula como en el laboratorio, para dominar y poner en práctica las distintas técnicas y conceptos. Pero el trabajo nuclear del estudiante es un proyecto software, que debe realizar en paralelo a las clases regladas. Se le proporciona un enunciado y una solución parcial en el que se han aplicado las tecnologías indicadas. Su trabajo consistirá en completar el proyecto con nuevos casos de uso, un modelo de dominio más rico, etc.

Las listas indicadas más arriba no son cerradas. Los estudiantes más avanzados pueden experimentar en sus proyectos con otras tecnologías (como JWS), y en las clases se les mencionan otros artefactos útiles (como diagramas de componentes), pero sabemos que con las herramientas mencionadas pueden abordar con garantías todos los objetivos de aprendizaje.

2. El cambio de paradigma

Hasta aquí hemos descrito el planteamiento de la asignatura tal como ha sido diseñada desde el punto de vista de los contenidos. Todo lo dicho hasta ahora se ha utilizado usando metodologías docentes más o menos tradicionales, en las que el proyecto es complementado por diversas pruebas individuales sumativas y eliminatorias. Ahora bien, el objeto de la presente contribución es describir cómo se ha adaptado una asignatura con un contenido bastante ajustado y milimetrado al Aprendizaje Basado en Proyectos (ABP).

2.1. ABP en Ingeniería del Software I

La Facultad de Informática de la UPV/EHU ha desarrollado una serie de líneas estratégicas para incorporar a la oferta docente de los nuevos Grados. Así se ha puesto en marcha un programa específico de implantación de asignaturas cuya docencia se encuadre en el paradigma del Aprendizaje Basado en Proyectos (ABP). El objetivo es que el alumno pueda desarrollar una asignatura por cuatrimestre siguiendo dicha metodología.

Los profesores de Ingeniería del Software I nos implicamos en dicho programa y adaptamos el proyecto docente de la asignatura para incorporar técnicas de aprendizaje colaborativo y conseguir que el proyecto que constituye el núcleo de la asignatura

se desarrollase en el marco del ABP. Así, si anteriormente se realizaba de manera individual, pasó a ser concebido como una actividad para equipos de tres personas. Afortunadamente es bastante sencillo redimensionar la especificación de un proyecto software variando la cantidad y complejidad de los casos de uso propuestos.

Ahora bien, el primer principio del ABP es dotar al estudiante de autonomía para buscar sus propias soluciones. Las herramientas y artefactos utilizadas en el proceso de desarrollo del software se caracterizan por su multiplicidad y por la complejidad de su uso combinado (que al fin y al cabo es lo que asegura que la producción de software pueda acometerse con estándares de tipo industrial). El alumnado debe dominar un subconjunto de ellas y luego desarrollar criterios para decidir cuándo son útiles y cuándo suponen una sobrecarga de trabajo, por no mencionar un estorbo.

Dicho todo esto, ¿cómo conseguir que las alumnas y alumnos “descubran” por sí mismos las tecnologías y los métodos sin perderse en el propio proceso de búsqueda? El riesgo cierto es que los estudiantes tengan una visión cortoplacista y se lancen a codificar con un conocimiento muy superficial de las herramientas de modelado.

La aproximación que vamos a elegir trata de abordar el problema desde varias direcciones:

1. La utilización de los laboratorios y otras actividades que se citarán más adelante tendrán un diseño ejemplificador. Cada laboratorio, por ejemplo, contiene una parte guiada paso a paso y otra abierta para aplicar lo aprendido a otro problema.
2. Como se ha indicado, el enunciado que se le proporcionará a los equipos como parte de la documentación del proyecto vendrá con una solución parcial, es decir, bajo la hipótesis de que algunas características ya están resueltas y otras no. El trabajo con versiones ya iniciadas permitirá que experimenten la incertidumbre y la frustración de tener que revisar, corregir y ampliar un trabajo ajeno en condiciones de insuficiente documentación. Posteriormente se les proporcionará la documentación que faltaba para que aprecien la diferencia, y por tanto la utilidad de los distintos artefactos. Este aspecto, así como el perfil de los estudiantes, es el que determina que no planteemos un proyecto desde cero, como otros autores han hecho con éxito [6].
3. Dentro del escenario que se les planteará introduciremos la figura de los “marrones”. Se trata de encargos inesperados, no relacionados directamente con el proyecto, en los que tendrán que corregir errores y disfunciones en el código supuestamente producidos por otros equipos de

desarrollo. La solución de los marrones permitirá aprender a utilizar con más precisión tanto tecnologías concretas como herramientas de modelado.

Como se ha indicado, puede haber situaciones en las que los estudiantes encuentren (o conozcan de antemano) alternativas a nuestras propuestas. Salvo en el caso de que consideremos que su elección pone en riesgo la realización del proyecto, dejaremos siempre abierta esa posibilidad.

2.2. La gestión del trabajo en equipo: Scrum

Uno de los principales obstáculos para el éxito del trabajo cooperativo es que el alumnado raras veces cuenta con herramientas o habilidades básicas para gestionar el mismo, empezando por el hábito negociar y tomar decisiones y terminando por la forma de mantener una documentación efectiva que promueva el reparto de tareas.

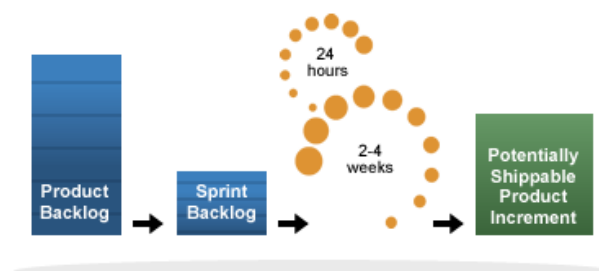


Figura 1: Un sprint en el proceso Scrum [3]

Para salvar al menos parcialmente este obstáculo hemos apreciado que el dotar a las y los estudiantes de una metodología sencilla de entender y aplicar puede ser un impulsor crucial. Es por ello que les propondremos el uso de Scrum, o al menos de una versión aligerada del mismo. Esta metodología es la más popular entre todas las que compiten por el mercado del desarrollo ágil [1], por su naturaleza de proceso, en el que se aplican de manera regular [un conjunto de buenas prácticas](#) para trabajar en grupo de manera colaborativa y obtener [el mejor resultado posible](#) de un proyecto.

La metodología está basada en roles, lo que supone una ventaja a la hora de dotar a los equipos una manera natural de distribuir las responsabilidades y tareas:

1. *El Dueño del Proyecto*. Responsable de la priorización de requisitos, de la validación de resultados, etc. En un proyecto real sería el cliente, pero en nuestro caso tendrá que ser desempeñado por el profesor. Idealmente sólo tendría que hacer intervenciones puntuales.
2. *El ScrumMaster*: Su función es asegurarse de la aplicación de la metodología Scrum. Por tanto no tiene un rol jerárquicamente superior en el

plano técnico, sino en el organizativo. Esta función será rotatoria entre los estudiantes del equipo.

3. El *Scrum Team*: Lo forman los tres miembros del grupo.

En Scrum un proyecto se ejecuta en bloques temporales cortos y fijos llamados *sprints*. La noción de *sprint* es absolutamente análoga a la de iteración, y el nombre únicamente recalca la importancia del tiempo y la distancia fija a una meta. Cada *sprint* tiene que proporcionar un resultado previsto y completo *en forma de incremento de producto final* que sea susceptible de ser entregado con el mínimo esfuerzo al cliente. La duración típica de un *sprint* es de 2 a 4 semanas para mantener manejable la planificación.

Los *sprints* se gestionarán, de acuerdo con la metodología, mediante cuatro acciones principales, llamadas ceremonias:

1. *Selección de requisitos* (una sesión de clase con todos los grupos). El profesor, haciendo el papel de cliente, discute con el [equipo](#) la lista de requisitos más o menos priorizada del producto deseado.
2. *Planificación del sprint*. Una reunión del equipo, en tiempo no presencial, para elaborar la [lista de tareas del sprint](#) necesarias para desarrollar los requisitos a que se ha comprometido. La estimación de esfuerzo necesario se hace de manera conjunta y las personas del equipo se auto-asignan las tareas. Cada estudiante se centra en su parte, pero no puede desentenderse de lo que hacen los compañeros, porque al final debe estar en condiciones de explicar el estado del desarrollo del proyecto.
3. *Scrum diario*. Una reunión diaria de sincronización (15 minutos máximo). Cada persona del equipo expone a los demás el estado de sus tareas. En la reunión cada persona del equipo responde a tres preguntas: ¿Qué he hecho desde la última reunión de sincronización? ¿Qué voy a hacer a partir de este momento? ¿Qué impedimentos me he encontrado? El Scrum diario elimina la necesidad de reuniones farragosas, promueve la corresponsabilidad y permite monitorizar las dependencias entre tareas y el progreso hacia el objetivo.
4. *Revisión del sprint*: El equipo muestra sus logros en presencia del cliente. Es el momento clave en que su trabajo de todo el *sprint* va a ser evaluado. Se realiza en una sesión lectiva especial.
5. *Retrospectiva*: Fuera de clase el equipo ha de contestar a tres preguntas sobre su funcionamiento como grupo de trabajo. ¿Qué debemos seguir haciendo? ¿Qué debemos dejar de hacer?

¿Qué deberíamos empezar a hacer? Hace un breve informe y lo deja a disposición del profesor. Esta ceremonia no se realiza después del último *sprint*.

2.3. Repositorios Scrum

La utilización de Scrum evita la generación de mucha documentación superflua, pero la que se produce es extraordinariamente dinámica y es muy importante que esté accesible en todo momento para su consulta y actualización. Las estructuras que gestionan esta documentación que representa la auténtica instantánea del proyecto se denominan *backlogs*.

El *backlog del producto* es la lista de requisitos o casos de uso del sistema, normalmente llamados características (*features*) o historias de usuario (*stories*). Este se incrementa en las reuniones de selección de requisitos. A su vez, en cada iteración se construye un *backlog de sprint*, que contiene:

- inicialmente, un subconjunto del *backlog del producto*: aquellas historias que se van a incorporar en el presente sprint.
- cada historia se habrá descompuesto en tareas, con un esfuerzo estimado, un responsable y un entregable.
- a medida que los miembros del equipo van completando sus tareas lo indican en el *backlog de sprint* e incluyen enlaces o adjuntos a los entregables finalizados. Hay una herramienta de representación (*task manager*) que permite observar el estado de cada tarea e historia del sprint (futura, en curso o terminada), y otra que exhibe el ritmo al que avanza el trabajo frente a una evolución ideal a ritmo constante (*burndown chart*)

Por tanto los backlogs de un proyecto son una herramienta excelente que permiten al equipo (y al profesor) apreciar el avance de cada proyecto, la contribución individual de los miembros del equipo y los riesgos y problemas identificados sin exigir ni mucho tiempo ni procesos complejos de gestión. Es también una herramienta de soporte para la comunicación directa del equipo y promueve la reflexión crítica del grupo.

Actualmente existen numerosas herramientas que permiten gestionar los *backlogs* de un proyecto. De entre las disponibles libremente hemos seleccionado *VersionOne*¹ por su buen balance entre usabilidad, accesibilidad, funcionalidades y complejidad.

2.4. Ingredientes del aprendizaje colaborativo en el proceso Scrum

Una de los requisitos a tener en cuenta a la hora de planificar las actividades dentro del ABP, consiste en

¹ http://www.versionone.com/pricing_and_editions/ (TEAM)

garantizar que se cumplen al menos los cinco ingredientes del aprendizaje colaborativo [2]. Consideramos que Scrum recoge de manera natural tales ingredientes.

Uno los aspectos metodológicos de Scrum, consiste en realizar reuniones de Scrum diario para realizar el seguimiento del proyecto. Dado que los estudiantes no se dedican a tiempo completo al proyecto, en nuestro contexto “diario” significa “dos veces a la semana”, concretamente al comienzo de las clases de lunes y jueves. De esta forma fomentamos y garantizamos la *interacción cara a cara*, uno de los ingredientes básicos mencionados. Adicionalmente solventamos el problema de encontrar tiempo compatible, podemos asistir parcialmente a esas reuniones y podemos leer el informe o resultado de las mismas. Los *backlogs* son excelentes mecanismos de *exigibilidad individual*: es poco probable que un equipo tolere que un miembro se atribuya la realización de tareas que no están completadas y por las que todo el equipo va a ser responsabilizado.

El mantenimiento del *backlog de sprint* es útil porque obliga a los personas del equipo a una *comunicación directa* para determinar qué actividades hay que desarrollar y cómo abordarlas de manera colaborativa (repartiéndose el trabajo). También somos conscientes de que durante el trabajo colaborativo inevitablemente surgirán desavenencias e incidentes, que les deberá llevar a una *reflexión crítica del grupo*, donde deberán tomar decisiones que permitirán mejorar la dinámica de equipo. Esta se verá reforzada por la reunión de retrospectiva, donde deben identificar qué es lo que funciona bien, qué es lo que no, y qué medidas podrían tomar para mejorar.

En definitiva, Scrum por su propia naturaleza, se basa en el *compromiso conjunto* y la colaboración entre las personas del equipo. La *transparencia* entre todos es fundamental para poder entender la situación real del proyecto y así poder hacer las mejores adaptaciones que permitan conseguir el *objetivo común*. Por ello, su implantación como modelo de trabajo en el desarrollo del proyecto creemos que fomentará el ejercicio de *habilidades interpersonales y de trabajo en grupo*.

2.5. La influencia del desarrollo iterativo en el aprendizaje

Para desarrollar el proyecto hemos intentado integrar cuatro de las tendencias más importantes que han caracterizado la evolución de la Ingeniería del Software:

1. La necesidad de utilizar una metodología sólida integrada en la cultura de la organización (en nuestro caso el Proceso Unificado de Desarrollo del Software UP).
2. El uso de notaciones estandarizadas orientadas a producir software manejable, mantenible y reutilizable (en nuestro caso lenguaje visual de modelado UML).
3. El desarrollo del software en iteraciones, por oposición a los ciclos de vida que pretenden agotar al máximo cada fase antes de proceder a la siguiente.
4. La adopción de metodologías ágiles que eviten la sobrecarga de artefactos, fomenten la discusión entre grupos de interés y produzcan versiones ejecutables y utilizables de manera continua.

La utilización de Scrum, ya discutida en secciones anteriores, tiene una cualidad muy interesante para el proceso de aprendizaje: dado que el final de un *sprint* se ha de producir un producto software (ejecutable y funcional) que mejore el desarrollado en la iteración anterior, ello implica que en todos los *sprints* se ha de trabajar todas las disciplinas (Requisitos, Diseño e Implementación). Esto permite una gran flexibilidad a la hora de distribuir los objetivos de aprendizaje, ya que cada *sprint* puede ser visto como una *pasada* sobre la metodología, de forma que en cada iteración el producto se irá enriqueciendo con funcionalidades, aspectos, casos de uso, interfaces y modelos de dominio crecientemente complejos, que estimularán al o la estudiante a dominar de manera progresiva las diferentes técnicas y herramientas de la Ingeniería del Software.

Los ciclos de vida iterativos favorecen la trazabilidad y el control de cambios de los proyectos software. Es el tiempo la variable que gobierna el desarrollo, y no el producto “ideal”. Por otro lado cobra importancia el concepto de “el mejor producto que podemos producir en tiempo X”, idea muy interesante desde el punto de vista didáctico, ya que permite que:

- manejar de la misma manera los objetivos de aprendizaje, en el sentido de poder definir “el mejor dominio que podemos adquirir sobre el proceso de software en tiempo X”
- flexibilizar dichos objetivos en función de las especificidades de los equipos de estudiantes; grupos con diferentes preparaciones, diferentes capacidades y, sobre todo, diferentes expectativas académicas, puedan sacar el mejor provecho de sí mismos sin verse negativamente afectados por las características de los restantes grupos.

Por ello hemos configurado nuestro proyecto docente inspirados en la propia materia que impartimos. Consideramos que si vemos los objetivos de aprendizaje de esta asignatura como un producto a desarrollar, podemos aplicar estas metodologías al diseño y planificación de la asignatura. Nos gustaría

resaltar el aspecto novedoso de este enfoque, en el que los temas no se desarrollan secuencialmente ni dotan de estructura al curso, sino que se desarrollan de manera iterativa, revisando y profundizando cada concepto y técnica en varias ocasiones.

3. El Proyecto de la asignatura

El Proyecto para la asignatura se planteará en forma de enunciado incremental que se enriquece a medida que los alumnos y alumnas van superando las iteraciones. Asimismo es muy importante la idea de suministrarles junto al enunciado una solución parcial, es decir un punto de partida que, por un lado, les limita y obliga a hacer un trabajo de análisis y comprensión, y por otro les da pistas acerca de cómo pueden abordar los problemas.

3.1. El escenario de partida

Antes incluso de la formación de grupos el planteamiento será el siguiente. Cada estudiante acaba de llegar a la importante empresa *Sinking Soft*. Inicialmente el desconcierto es total: el recién llegado no sabe qué se espera de él, y el equipo técnico que le recibe tampoco sabe hasta qué punto puede contar con él ni tiene tiempo o ganas de averiguarlo. El profesor tiene aquí que desempeñar un tercer rol: además de su función ordinaria y de su figuración como cliente, debe en ocasiones hacer las veces de responsable de la empresa de acogida..

Una de las primeras cosas que le sucederán es que le caerán los “marrones” que hemos descrito en el apartado 2.1: trabajos defectuosos o inconclusos que hay que pulir para que cumplan todas las especificaciones. Estos marrones serán de tipo individual.

Tras la formación de grupos (en principio libre) tendrá un encargo de mayor responsabilidad: el Proyecto, que también será un trabajo dejado a medias por otros compañeros (la solución parcial). Su actuación está siempre a prueba, y que hasta que no complete dicho proyecto no se le ofrecerán responsabilidades reales ni condiciones laborales mejores. El proyecto será el mismo para todos los grupos y consistirá en una aplicación de software que permita el alquiler remoto de casas rurales y la gestión de las mismas por sus propietarios. El alcalde del pueblo en el que están situadas las casas es el cliente.

En este contexto, la pregunta motriz asociada al proyecto es la siguiente: *¿Tú crees que vas a durar mucho en Sinking Soft?*

En la primera iteración se introduce el Proyecto y se forman los grupos de tres alumnos, o en caso de necesidad, de cuatro. El planteamiento inicial está orientado a la identificación de los requisitos iniciales del Proyecto. De entre los requisitos capturados cada equipo de proyecto debe seleccionar los más importantes, teniendo en cuenta su centralidad para la

arquitectura de la aplicación, los riesgos estimados para su implementación y, por supuesto, las prioridades del cliente. Se estudiará cómo obtener los diagramas de secuencia a partir de los casos de uso y flujos de eventos seleccionados, cómo conseguir una evolución consistente del Modelo del Diseño a partir del Modelo del Dominio, y cómo mantener la consistencia entre los diferentes artefactos. La iteración termina en fecha prevista de antemano y la solución de los requisitos debe estar incorporada a la aplicación. En esta iteración el profesor o profesora tiene que prestar atención para evitar, por un lado, que se prescinda del soporte metodológico, y en el otro extremo que se adopten metodologías inadecuadas (por ser demasiado restrictivas o por exigir una carga de gestión desproporcionada). En esta iteración se prevé que los incumplimientos pueden ser más frecuentes y se tolerarán siempre que no indiquen una clara dejación del equipo. La mayor parte de los laboratorios introductorios se realizará coincidiendo con esta iteración.

En la segunda iteración la exigencia formal será mayor y se deberán completar al menos un conjunto de casos de uso previstos desde el principio o su equivalente. Además se trabajarán específicamente los ejercicios complicados relacionados con la disciplina de Requisitos. Se realizará el laboratorio de RMI.

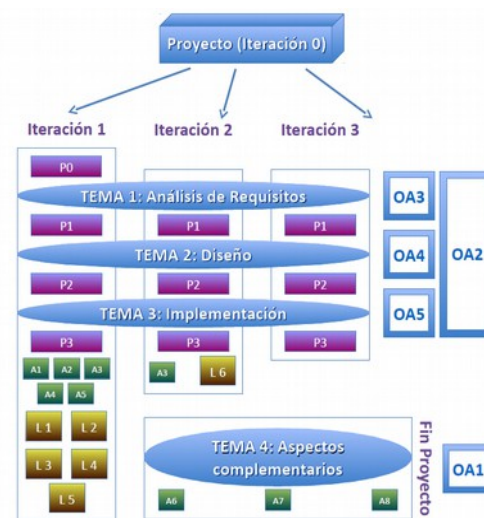


Figura 2: Temario, Iteraciones y Actividades

En la tercera iteración los equipos tendrán mayor libertad para proponer modificaciones en las funcionalidades según sus propios criterios, capacidad creativa y eficacia trabajando. El profesorado seguirá proporcionando soporte de clientes y usuarios finales para evitar que los equipos de proyecto intenten abarcar funcionalidades de alto coste y reducido valor añadido (no se puede suponer que el alumnado tenga mucha experiencia ni como propietarios ni como usuarios de casas rurales). El resultado final de esta iteración será el proyecto resultante del grupo y será

evaluado como tal. en sus diversos apartados (véase la figura 2):

- **P0:** Análisis de requisitos globales del proyecto.
- **P1:** Análisis de requisitos de la iteración en curso.
- **P2:** Diseño de la iteración en curso
- **P3:** Implementación y prueba de la iteración en curso.

Las actividades de desarrollo del Proyecto (P1 a P3) serán recurrentes y deberán revisarse y completarse en cada una de las tres iteraciones. Para ello los estudiantes tendrán que buscar documentación e inspirarse en el material que se les ha entregado para conocer y aplicar las herramientas y artefactos de cada disciplina.

4. Actividades complementarias y evaluación

Todas las actividades a realizar serán obligatorias, y por tanto conllevarán algún tipo de evaluación. Sin embargo, para evitar una excesiva rigidez y un número demasiado alto de notas a integrar hemos considerado dos modalidades principales y una variante adicional:

- **Filtro.** La valoración de la actividad se calificará como *Apto* o *No apto*, pero no tendrá capacidad aditiva sobre la calificación final. Se tolerará un máximo de tres actividades con *No Apto*.
- **Punto Fijo:** La actividad tendrá un peso concreto en la calificación final.

- **Extra Bonus:** Independientemente de que la actividad se evalúe según alguno de los esquemas anteriores, se premiarán las actividades que de alguna forma sean sobresalientes con puntos adicionales sobre la calificación final.

Las actividades de tipo filtro serán las siguientes. La primera y las tres últimas serán individuales, y el resto serán realizadas por los equipos.

- **A1:** Panel del Modelo de Referencia, en nuestro caso el Proceso Unificado de desarrollo de Software.
- **A2:** El cliente siempre tiene razón. El objetivo es simular un entorno de trabajo real con una reunión entre el cliente y un equipo de desarrollo
- **A3:** Discusión dirigida sobre los requisitos del proyecto. Inicialmente se hace un contraste por pares de las propuestas de los y las estudiantes.
- **A4:** Juego de los objetos animados. Simular de forma animada la construcción y uso de un diagrama de secuencia.
- **A5:** Puzzle sobre patrones GRASP.
- **A6:** Concurso de desastres informáticos.
- **A7:** Presentación de temas que habitualmente se hacen como introducción de un curso de Ingeniería del Software
- **A8:** Vendedores del *Rational Unified Process* (RUP)

Puede consultarse una descripción completa de las mismas en [4]. Para su organización temporal véase la figura 2.

	ITERACION-1	ITERACION-2	ITERACION-3	TOTAL
P0	15%			75%
P1				
P2		25%	35%	
P3				
Control Anti-Doping + Laboratorios + Actividades Filtro	10%	10%	5%	25%

Tabla 1: Porcentaje asignado a las actividades en cada Iteración del Proyecto.

Dado que al final de cada iteración cada grupo debe tener una aplicación operativa, aunque quizás incompleta, debería ser posible hacer una presentación de control ante el cliente. Sin embargo esto puede suponer un indeseado cuello de botella si las o los profesores tienen que revisar en tiempo real todos los trabajos. La solución será introducir una sesión de contraste entre pares, a realizar en un laboratorio al final de las iteraciones 1 y 2. Cada grupo tendrá como tarea asistir a la defensa del

Proyecto de otro grupo y detectar posibles errores, inconsistencias o fallos de funcionamiento. Como conclusión debe escribir un *informe de inspección* que se entregará al profesor o profesora que se incorporará al *backlog* del grupo auditado, junto con una versión “congelada” del proyecto.

Este contraste será una tarea de tipo filtro, pero podrá tener consecuencias posteriores. Si un fallo grave es detectado e incluido en el Informe pero no es corregido por el grupo inspeccionado, este

sufrirá una penalización adicional que en casos extremos puede implicar la anulación del proyecto. Pero si un fallo grave no es detectado y se propaga a iteraciones posteriores la responsabilidad será compartida entre el grupo inspeccionado y la persona inspectora.

La inspección final será la defensa del proyecto y se efectuará por el profesorado al terminar la tercera iteración. Esta tendrá forma de exposición pública ante la clase. Finalmente, en las últimas semanas del cuatrimestre se realizará un control para confirmar el nivel de exigibilidad individual adecuado en el desarrollo del proyecto (lo denominamos *control anti-doping*).

La tabla 1 recoge el porcentaje de la nota asignado a las diferentes actividades en cada iteración. La primera fila, un 75% es la parte dedicada a la ejecución del proyecto según el modelo ABP.

La segunda fila, un 25% recoge las actividades complementarias, no centradas en el modelo ABP, aunque si estrechamente relacionadas con el. Se han agrupado las evaluaciones de las pruebas anti-doping (15% y nota mínima eliminatoria) con los laboratorios dirigidos (10%) y las actividades de tipo filtro, sin una puntuación fija.

5. Conclusiones

Este plan actualmente se está implantando dentro del marco lectivo del segundo cuatrimestre (curso 2012-13). En el momento de la presentación del no podemos incluir como parte del artículo los resultados obtenidos. Sin embargo, a principios de junio tendremos información obtenida de las siguientes evidencias planificadas:

1. Encuesta de opinión al alumnado sobre la implementación (encuesta que se facilitará por parte de las personas responsables de implantar el programa en la Universidad)
2. Análisis comparativo de resultados respecto al curso anterior en el cual no se siguieron metodologías activas.
3. Seguimientos de los *backlogs* de los equipos.

Se ha presentado, en el marco del ABP, una implementación concreta para evaluar una asignatura con un carácter práctico muy marcado y un temario bastante denso (véase la bibliografía de referencia [5]). La inclusión de Scrum como metodología de desarrollo y gestión del trabajo en equipo presenta unas características idóneas para la inserción de los principios del ABP: fomenta el compromiso conjunto y la colaboración entre las personas del equipo. Creemos que su implantación como marco de trabajo contribuirá decisivamente a superar uno de los problemas endémicos de los proyectos de aprendizaje colaborativo: la dificultad de dotar a los estudiantes de un marco de expresión

y desarrollo de sus habilidades interpersonales y de su capacidad de trabajar en grupo.

Finalmente resaltaremos que el uso de una metodología estrictamente iterativa y orientada a plazos (*timeboxed*) ha contribuido a repensar la asignatura de forma que los grandes temas no se vean como obstáculos, necesarios pero formidables, que se interponen entre el alumno y su aprendizaje práctico, sino como vetas de las que pueden ir extrayendo el material que necesitan a medida de que su ambición, su madurez y su comprensión de los problemas se van desarrollando.

Referencias

- [1] Albadalejo X. y otros autores: *Como gestionar proyectos con Scrum*: <http://www.proyectosagiles.org/>
- [2] Bará J., Dominguez J. y Valero M. *Técnicas de Aprendizaje Cooperativo y Aprendizaje Basado en Proyectos*. Taller de Formación. Univ. Polit. de Cataluña. 2011
- [3] Cohn M.: *What is Scrum methodology*. <http://www.mountaingoatsoftware.com/topics/scrum>
<http://pkab.wordpress.com/2008/07/11/scrum>
- [4] Goñi A., Ibáñez J., Iturrioz, J. y Vadillo J.A.: *ABP aplicado a la asignatura Ingeniería del Software: Guías del docente y del estudiante*. Informe Interno UPV/EHU/LSI/TR 03-2012. (<https://addi.ehu.es/bitstream/10810/8428/1/TR%2003-2012.pdf>)
- [5] Larman C. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development* (3rd edition) Pearson Education, 2004.
- [6] Sánchez P. y Blanco C. *Implantación de una metodología de aprendizaje basada en proyectos para una asignatura de Ingeniería del Software*. JENUI 2012 pp.41-48.

**Métodos y herramientas para la
evaluación, coordinación y
motivación del alumnado**

Mejora del rendimiento en las sesiones de laboratorio de la asignatura Informática para ingenierías

Jorge Sales, Gabriel Recatalá
Departamento de Ingeniería y Ciencia de los Computadores
Universitat Jaume I
Castellón
{salesj, grecata}@uji.es

Resumen

Los nuevos grados de ingenierías surgidos del EEES cuentan en primer curso con una asignatura de informática en la que los alumnos han de ser capaces de diseñar programas estructurados con aplicación en ingeniería. De la experiencia del profesorado en los últimos dos años desde su implantación, se observa que una de las partes en la que los alumnos encuentran más dificultades es en el aprendizaje y dominio de la programación. En este trabajo presentamos una mejora en la metodología docente que hemos llevado a cabo en algunos grupos de laboratorio con el objetivo de mejorar el aprovechamiento de las sesiones por parte de los alumnos así como el rendimiento en las pruebas de evaluación presencial. Las encuestas de satisfacción pasadas a los alumnos revelan que el uso de esta metodología mejora significativamente el proceso de enseñanza-aprendizaje, ya que aumenta su participación en clase e incrementa su motivación y expectativas globales en los resultados finales de la asignatura.

Abstract

The new degrees in engineering which have emerged from the EHEA, include in the first year a computing course in which students must be able to design structured programs applied to engineering. From the experience of teachers over the past two years since its establishment, it can be observed that one aspect which students find most difficult is in learning and mastering programming. In this work, we present an improvement in teaching methodology that we have been developing in some laboratory groups in order to improve the academic progress of the sessions by the students and the performance in the evaluation tests. The satisfaction surveys filled in by the students reveal that the use of this methodology significantly improves the teaching and learning process, as it increases their participation in class and increases their motivation and overall expectations in the final results of the course.

Palabras clave

Lenguajes de programación, laboratorio, ingeniería, rendimiento, entrenamiento, motivación.

1. Introducción

La asignatura Informática, objeto de la mejora pedagógica que describimos en este trabajo, forma parte del bloque de asignaturas de formación básica de primer curso de los grados en Ingeniería Agroalimentaria y del Medio Rural, en Ingeniería Eléctrica, en Ingeniería en Tecnologías Industriales, en Ingeniería Mecánica y en Ingeniería Química. Esta asignatura arrancó en la Universitat Jaume I (UJI) el curso académico 2010-11, año en el que se empezaron a implantar los nuevos grados de ingenierías adaptados al EEES.

De acuerdo con el plan de estudios de dichas titulaciones, la asignatura debe abordar: bases de datos, estructura de ordenadores, herramientas informáticas de utilidad en las ingenierías y programación. De estos temas, se ha observado que es éste último el que suele causar mayores dificultades a los estudiantes. Además, esta dificultad se agrava por la actitud, también señalada por otros autores [1, 2, 3], de rechazo a la asignatura por parte del estudiantado, que suele considerarla lateral o ajena al ámbito de su titulación. También cabe señalar que dichos estudiantes han tenido históricamente una base previa muy heterogénea de conocimientos de informática [1, 4]. En otros trabajos, se han intentado abordar estas dificultades mediante diferentes estrategias orientadas principalmente a fomentar la participación del estudiante y mejorar la realimentación sobre su rendimiento a lo largo del curso.

Los autores de este trabajo han estado involucrados tanto en la docencia como en la coordinación de la asignatura desde el momento de su implantación, impartiendo clases de teoría, problemas y laboratorio. Durante el primer año en el que se puso en marcha la asignatura, ya se llevó a cabo una acción de mejora en

la parte de proyectos finales (proyectos de programación en la que los alumnos han de trabajar en grupos de 2 a 6 personas), cuyo objetivo era fomentar una buena planificación inicial y preparación para el trabajo en equipo. La mejora consistía en la entrega a los alumnos, durante la sesión inicial y arranque del proyecto, de unas fichas que les ayudaban a identificar desde un primer momento las principales fases de que constaba su proyecto, así como de una supervisión directa por parte de los profesores de qué roles se asignaban dentro de cada equipo.

Tras observar resultados positivos en la aplicación de esta mejora, nos propusimos en este curso abordar otros aspectos como la detección de problemáticas y dificultades de aprendizaje, el análisis de la evaluación y el aumento de la motivación, especialmente en las sesiones prácticas de laboratorio, que es donde los alumnos ponen en práctica los conocimientos y conceptos aprendidos en las sesiones expositivas. El sistema EEES requiere un cambio en la forma de entender la docencia, y también un cambio en la forma en que los alumnos aprenden y desarrollan las nuevas competencias. Esto hace que durante los primeros años de la implantación del nuevo espacio europeo, sea necesario llevar a cabo acciones de innovación y mejora en todos los contextos de la enseñanza superior, y especialmente en asignaturas de primer curso como la que se describe.

2. Descripción de la asignatura

La asignatura Informática para los nuevos grados de ingeniería, tal como se imparte en la UJI, tiene unas características especiales que la hacen adecuada para poder incorporar estas propuestas de innovación y mejora, pues siendo una asignatura de primer curso se abordan aspectos tan trascendentales como la programación de ordenadores, el trabajo autónomo y el trabajo en equipo. Se enumeran a continuación algunas de las competencias que recoge la guía docente de la asignatura:

- CB03 - Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
- CG02 - Aprendizaje autónomo.
- CG06 - Resolución de problemas.

Esta asignatura, al igual que la mayor parte de las asignaturas de los dos primeros cursos, es compartida entre las titulaciones mencionadas anteriormente. En el curso 2012/13, el alumnado se encontraba distribuido en 6 grupos de teoría, 13 de problemas y 24 de laboratorio, a cargo de los cuales se encontraban 9 profesores. Además, cada grupo incluía alumnos de las distintas ti-

tulaciones. Todas estas circunstancias aumentan la dificultad de abordar los problemas de motivación y seguimiento mencionados anteriormente.

Como se ha indicado, una de las partes en que los alumnos encuentran más dificultades es en el aprendizaje del lenguaje de programación visto en el contexto de la asignatura. Aprender programación requiere mucho tiempo y esfuerzo. Los alumnos son de primer curso (y primer semestre), de titulaciones no relacionadas con la informática y no podemos presuponer que tengan conocimientos previos sobre programación de ordenadores. Además, la asignatura dura sólo un semestre, lo que obliga a que la curva de aprendizaje tenga que ser muy inclinada.

2.1. Las sesiones de laboratorio

En la asignatura se realizan un total de 5 sesiones prácticas de laboratorio, de 2 horas cada una. Durante la primera hora, los alumnos trabajan de manera autónoma y pueden plantear dudas al profesor, pero no disponen de ningún material de apoyo específico para ello. Normalmente se recomienda la resolución de problemas del boletín de problemas del tema a tratar en la sesión (el profesor puede proponer algún problema en particular). Este boletín es el mismo que se utiliza en las sesiones de problemas, donde se trabaja en un aula tradicional, sin disponibilidad de ordenadores. A continuación, pasada la primera hora, disponen de una segunda hora para realizar una prueba evaluable y puntuable (ver un ejemplo de prueba en la Figura 1), que consiste en la resolución de ejercicios de programación, y que deben entregar al finalizar la sesión. En esta segunda parte, los alumnos trabajan en equipos de 2 personas, pudiendo ser de 3 si queda algún alumno o alumna sin pareja.

A modo de ejemplo, la prueba de evaluación que aparece en la Figura 1 corresponde a la sesión 4 de laboratorio, que trata sobre estructuras condicionales (instrucción o grupo de instrucciones que se pueden ejecutar o no en función del valor de una condición). El primer ejercicio plantea implementar una función que a partir del consumo eléctrico, que se puede producir por ejemplo en una vivienda, calcule el importe a pagar por el cliente. La resolución de este ejercicio requiere el uso de condicionales, ya que la tarificación se realiza por tramos, es decir, que cada tramo de consumo se pagará a un precio diferente. El segundo ejercicio plantea la implementación de dos funciones: la primera calculará la media de las notas de los alumnos almacenada en una matriz (pero sólo las que sean iguales o superiores a 5, de ahí el uso de condicionales), la segunda, hará uso de la primera, pero en este caso mostrará un mensaje al usuario en caso de que no haya ningún aprobado (nuevamente se requiere el uso de condicionales para detectar esta situación).


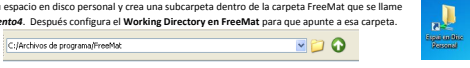
NOMBRE ALUMNO 1:	NOMBRE ALUMNO 2:
	<p>Informática Grado en Ingeniería Eléctrica, Mecánica, Química, Tecnologías Industriales y Agroalimentaria y del Medio Rural</p>
<h2 style="margin: 0;">Laboratorio 4</h2>	
<p>1. De acuerdo con la nueva normativa de tarificación de consumo eléctrico, que se aplicará a partir de 2013, el consumo se cobrará en función no sólo de la potencia contratada, sino también por tramos. En particular, para una potencia contratada de entre 3 y 4 kW, algunos de estos tramos y sus precios correspondientes son los siguientes:</p> <ul style="list-style-type: none"> ■ El tramo de consumo desde 190 (incluido) hasta 225 (no incluido) kWh se pagará 0.00138 euros por kWh. ■ El tramo de consumo desde 225 (incluido) hasta 260 (no incluido) kWh se cobrará a 0.00276 euros el kWh. <p>Para completar la información anterior, consideraremos que el consumo inferior a 190 kWh se cobra a 0.001 euros por kWh, y el igual o superior a 260 kWh a 0.005 euros por kWh.</p> <p>Escribe una función de nombre <i>coste_consumo_electrico</i> que reciba como parámetro el consumo y devuelva el importe a pagar por él.</p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <pre>%Ejemplo de uso --> coste_consumo_electrico (100) ans = 0.1000 --> coste_consumo_electrico (200) ans = 0.2038 --> coste_consumo_electrico (400) ans = 1.0349</pre> </div> <p>2. Una determinada matriz contiene las notas (de 0 a 10 puntos) de un grupo de alumnos, de manera que cada fila de la matriz corresponde a un alumno y cada columna a una asignatura. Se desea calcular la media de las notas superiores al aprobado (5 puntos) en este grupo.</p> <p>a) Escribe una función de nombre <i>media_aprobadas_clase</i> que reciba como parámetro una matriz como la anterior, y devuelva como resultado la media de los valores de la matriz iguales o superiores a 5. Si no hay ninguna nota que cumpla este criterio, la función devolverá el valor 0.</p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <pre>%Ejemplo de uso --> notas_clase = [3 7 5 8 7; 8 9 6 7 5; 5 3 2 7 6; 5 7 6 4 5]; --> media_aprobadas_clase (notas_clase) ans = 6.4375 --> notas_clase = [4 6 5 4 5; 3 5 9 8 4; 1 7 5 6 5; 7 6 5 6 4]; --> media_aprobadas_clase (notas_clase) ans = 6.0714</pre> </div> <p>b) Escribe un programa llamado <i>consulta_notas_aprobadas</i> que pida al usuario una matriz de notas, calcule la media de las notas aprobadas utilizando la función anterior, y muestre el resultado. En caso de que no haya ninguna nota superior al aprobado, se indicará dicha circunstancia.</p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <pre>%Ejemplo de uso --> consulta_notas_aprobadas Indica las notas de la clase: [4 6 5 4 5; 3 5 9 8 4; 1 7 5 6 5; 7 6 5 6 4] La nota media de las asignaturas aprobadas es: 6.0714 --> consulta_notas_aprobadas Indica las notas de la clase: [4 4 4 4 4; 4 4 4 4 4; 4 4 4 4 4; 4 4 4 4 4] No hay ningun alumno que haya superado alguna asignatura.</pre> </div>	

Figura 1: Ejemplo de prueba de evaluación entregada a los alumnos para su realización presencial durante la segunda parte de la sesión 4 de laboratorio. La prueba se realiza en grupos de dos estudiantes y debe ser entregada al finalizar la sesión para ser corregida y puntuada por el profesor.

Ejercicios de entrenamiento Laboratorio 4 con FreeMat
Valores Lógicos y Sentencias Condicionales.

Configuración del Working Directory (Carpeta de Trabajo):
Accede a tu espacio en disco personal y crea una subcarpeta dentro de la carpeta FreeMat que se llame **entrenamiento4**. Después configura el **Working Directory en FreeMat** para que apunte a esa carpeta.



Ejercicio 1: Crea un nuevo Fichero-M. (File > New File, o utiliza el icono New File). Observa que el nombre del fichero que FreeMat utiliza por defecto es **untitled.m**. Pula el botón Save para guardar el fichero. Llámale **esmayorque100.m**. Comprueba que se haya guardado en la carpeta de trabajo. Introduce las siguientes instrucciones:

```

% NOMBRE APELLIDO (sustituye por tu nombre y apellido)
% ESMAYORQUE100 función que determina si un número es mayor que 100
%
% Uso
% resultado = esmayorque100(x)
function resultado = esmayorque100(x)
resultado = x > 100;

```

Ahora, desde la ventana principal de FreeMat, comprueba que la función "funciona" bien:

```

--> esmayorque100(10)
ans =
    0
--> esmayorque100(100)
ans =
    0
--> esmayorque100(101)
ans =
    1

```

Comprueba también la ayuda de tu función con el comando **help**:

```

--> help esmayorque100

```

Crea un nuevo Fichero-M. (File > New File, o utiliza el icono New File). Observa que el nombre del fichero que FreeMat utiliza por defecto es **untitled.m**. Pula el botón Save para guardar el fichero. Llámale **ejercicio1.m**. Comprueba que se ha guardado en la carpeta de trabajo. Introduce las siguientes instrucciones:

```

a = 10;
b = 100;
c = 200;
esmayorque100(a);
esmayorque100(b);
esmayorque100(c);

```

Ahora, desde la ventana principal de FreeMat, ejecuta el ejercicio:

```

--> ejercicio1
ans =
    0
ans =
    0
ans =
    1

```

¿Entiendes la diferencia entre los dos ficheros que hemos creado?
El primero contiene una función (la primera línea comienza por la palabra reservada **function**), y el segundo es un **script** o **programa** que utiliza la función.

Ejercicio 2: Desde la ventana principal de FreeMat, define las siguientes variables:

```

--> a=0; b=10; c=30; d=100; e=false; f=true;

```

Intenta adivinar ahora el resultado (verdadero o falso) de las siguientes operaciones lógicas:

a<b	a==false	a & b	a < b a >=b	false true & 1	-1
a==b	b==true	a b	a < b false	1 0 0 0 & 1	-false

Ejercicio 3: Crea un nuevo Fichero-M. (File > New File, o utiliza el icono New File). Observa que el nombre del fichero que FreeMat utiliza por defecto es **untitled.m**. Pula el botón Save para guardar el fichero. Llámale **esmayorque200.m**. Comprueba que se ha guardado en la carpeta de trabajo. Introduce las siguientes instrucciones:

```

% NOMBRE APELLIDO (sustituye por tu nombre y apellido)
% ESMAYORQUE200 función que determina si un número es mayor que 200
%
% Uso
% resultado = esmayorque200(x)
function resultado = esmayorque200(x)
if x > 200
resultado = true;
else
resultado = false;
end

```

Ahora, desde la ventana principal de FreeMat, comprueba que la función "funciona" bien:

```

--> esmayorque200(200)
ans =
    0
--> esmayorque200(201)
ans =
    1

```

Ejercicio 4: Crea un nuevo Fichero-M. Llámale **preciodelcafe.m** i guárdalo en tu carpeta de trabajo. Introduce las siguientes instrucciones y completa la función:

```

function res = preciodelcafe(precio)
if precio <= 0
res = '¡¡Que barato!! ¡Café para todos!';
elseif precio > 0 & precio <=50
res = 'Café económico!';
elseif precio > 50 & precio <=100
res = 'Café expresso';
else
res = '¡¡Mejor si me invitas!!';
end

```

Crea un nuevo Fichero-M. (File > New File, o utiliza el icono New File). Observa que el nombre del fichero que FreeMat utiliza por defecto es **untitled.m**. Pula el botón Save para guardar el fichero. Llámale **ejercicio4.m**. Comprueba que se ha guardado en la carpeta de trabajo. Introduce las siguientes instrucciones:

```

x = input('¿Cuánto vale el café? ');
resultado = preciodelcafe(x);
disp(['Resultado de la función: ' resultado]);

```

Ahora, desde la ventana principal de FreeMat, ejecuta el ejercicio:

```

--> ejercicio4
¿Cuánto vale el café? 75
Resultado de la función: Café expresso

```

Figura 2: Ejemplo de boletín de entrenamiento utilizado en la sesión 4 de laboratorio. Este boletín se entrega a los alumnos durante la primera parte de la sesión para ayudarles en la preparación de la segunda parte, que consiste en una prueba entregable y puntuable.

2.2. Las dificultades encontradas

Tal como se apunta en [5], las "diferencias de nivel con que llega el alumnado a la universidad, las deficiencias de conocimientos que presenta, su falta de autoorganización, y la poca disposición a trabajar las materias con regularidad", son algunos de los aspectos que nosotros también hemos detectado y pretendemos abordar con nuestra propuesta metodológica. Además, sabemos que el perfil de entrada de los estudiantes es variado en función de la titulación a la que acceden y el origen académico del que provienen (bachillerato, ciclos formativos, etc.), y que estos factores influyen en sus resultados académicos, según se desprende de las conclusiones del proyecto *Coordinación de asignaturas compartidas en la implantación del primer curso de los grados en: Ing. Agroalimentaria y del Medio Rural, Ing. Eléctrica, Ing. Mecánica, Ing. Química e Ing. en Tecnologías Industriales*, desarrollado en el curso 2010-11 en el marco de la implantación de los nuevos grados del EEES en la UJI.

Así pues, de nuestra experiencia en particular, los problemas concretos más frecuentes que hemos observado en los grupos de laboratorio son los siguientes:

- No haber realizado el trabajo previo a la sesión.
- Conocimientos necesarios para el desarrollo de la sesión poco asimilados (esto es una consecuencia del punto anterior).
- Desorientación sobre cómo realizar las tareas propuestas.
- Dificultad para completar las tareas satisfactoriamente.
- Consideración de inaccesible de la tarea propuesta y falta de motivación para terminarla.

3. Objetivos y propuesta metodológica de mejora docente

El objetivo general de mejora que nos hemos planteado en el diseño de la propuesta metodológica consiste en mejorar el rendimiento de los alumnos en las

sesiones de laboratorio. No obstante, ya desde un primer momento, nos planteamos el que esta propuesta pudiera tener un efecto positivo en el global de la asignatura, como por ejemplo la mejora en los resultados de las pruebas finales escritas. En particular, nos planteamos los siguientes subobjetivos:

1. Aumentar el aprovechamiento de las sesiones por parte del alumnado.
2. Detectar dificultades de aprendizaje.
3. Aumentar la motivación del alumnado.
4. Acelerar el proceso de aprendizaje y dominio del lenguaje de programación.
5. Mejorar las tasas de éxito y rendimiento global en la asignatura.

Para conseguir estos objetivos, hemos seguido una metodología que consiste en la utilización de unos *boletines* u *hojas de entrenamiento* que se entregan a los alumnos durante la primera mitad de la sesión de laboratorio, y que previamente hemos diseñado y elaborado específicamente para cumplir los objetivos de mejora planteados y basándonos en nuestra experiencia de cursos anteriores. Los boletines cubren, de manera gradual, todos los contenidos necesarios para que los alumnos puedan resolver de manera satisfactoria todos los problemas que se les plantean en la segunda mitad de la sesión, donde se realiza la prueba de evaluación puntuable. Los alumnos saben que la correcta comprensión de los ejercicios planteados en los boletines sirve de *entrenamiento* para la prueba posterior, ya que los problemas que después van a tener que resolver son similares a los que han realizado con las *hojas de entrenamiento*.

4. Descripción de la propuesta de mejora

A modo de ejemplo, en la Figura 2 se puede ver un *boletín de entrenamiento* correspondiente a la sesión 4 de laboratorio. Como se ha dicho anteriormente, trata sobre condicionales, esto es, instrucciones que se pueden ejecutar o no en función del valor de una condición. Se trata de dos hojas con indicaciones guiadas (y en este caso en particular, con ejercicios resueltos) y está pensado para ser realizado por los alumnos durante la primera parte de la sesión, que dura una hora. Los profesores les indicamos a los alumnos, como motivación para que controlen el tiempo y que empiecen inmediatamente con el ejercicio, que deben ser capaces de llegar al final del boletín en el tiempo previsto de una hora, ya que si no lo hacen, probablemente tampoco puedan terminar el ejercicio puntuable de la segunda parte, puesto que se trata de una prueba que requiere un esfuerzo del mismo estilo. Asimismo, les indicamos que lo que harán en estos ejercicios les sir-

ve de *entrenamiento* para la prueba puntuable, ya que se trata de problemas *similares*, con la ventaja de que en esta primera parte, los ejercicios son una guía para llegar a entender cómo se resuelven este tipo de problemas.

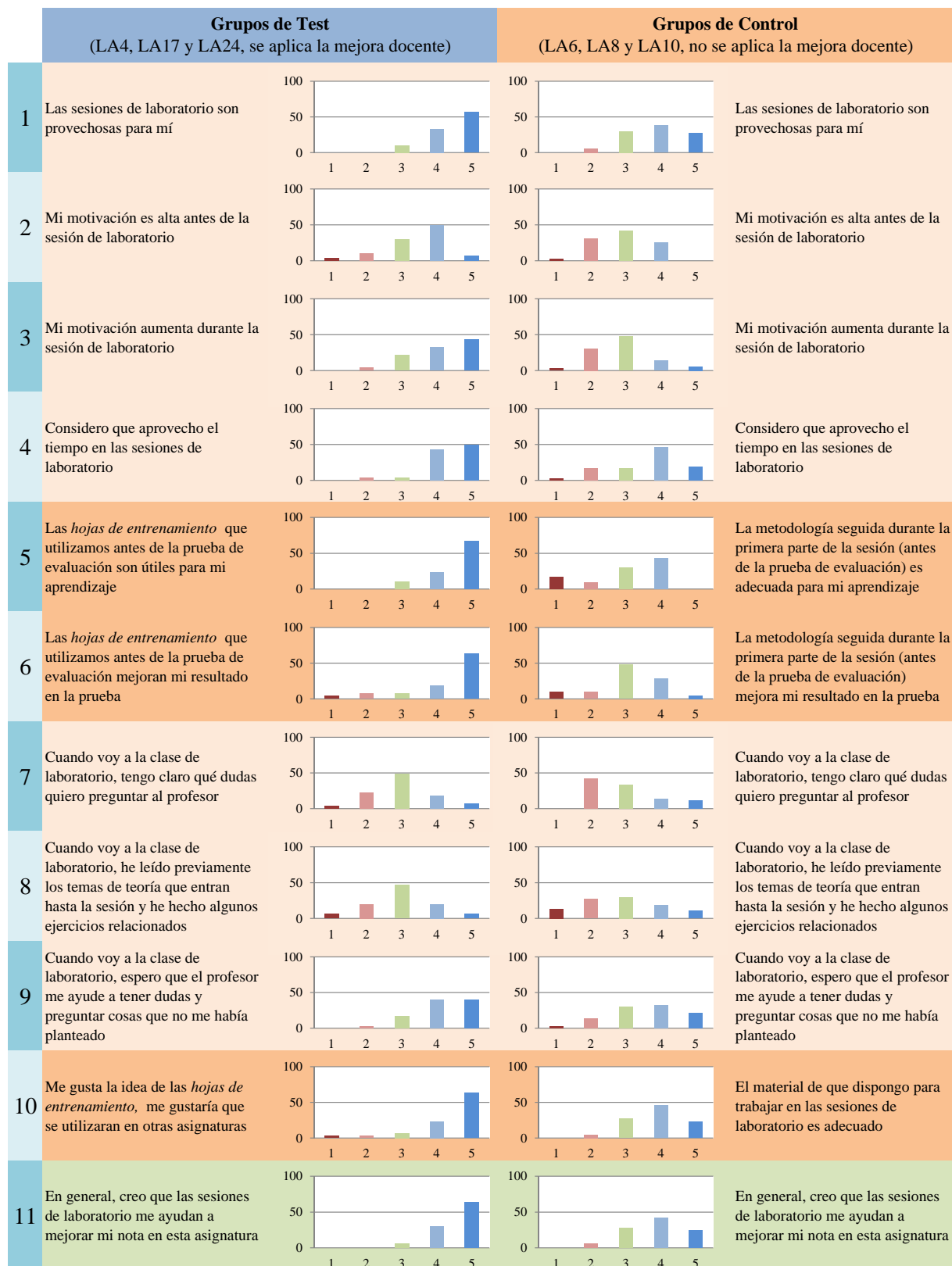
En particular, el *boletín* que se muestra en la Figura 2, empieza explicando cómo se configura el entorno de trabajo, algo que aunque pueda resultar repetitivo, observamos que los estudiantes olvidan fácilmente de una sesión a la siguiente. A continuación, se recuerda cómo se debe crear correctamente un fichero-M (tipo de fichero donde los alumnos deberán escribir cada uno de los programas), y se muestra la conveniencia de poner sus nombres y apellidos en la cabecera del fichero, así como una descripción de la función que van a implementar. Estos pasos se recuerdan siempre en cada ejercicio de los boletines, para crear el hábito entre los estudiantes de documentar el código, y poner los nombres de los autores, un requisito de la posterior prueba de evaluación. Además, pensamos que les ayuda a entender cómo se declaran las funciones y cuáles son los parámetros de entrada y de salida, aspectos todos ellos en los que hemos observado dificultades entre los estudiantes.

En el primer ejercicio, se muestra cómo resolver con una función si un número es mayor que una cierta cantidad. A continuación, se explica cómo utilizar esta función desde la línea de comandos, y posteriormente, cómo se crea un fichero con instrucciones que la utilizan. Al mismo tiempo, y a modo de repaso, se les hace reflexionar a los alumnos sobre la diferencia entre un programa o *script* y una función, un concepto visto en temas anteriores.

Ya en el ejercicio 2, nos aseguramos de que los alumnos entienden y conocen los operadores lógicos, un requisito previo para comprender bien las sentencias condicionales, objetivo de la práctica. Para conseguirlo, hay una tabla con operaciones cuyo resultado tienen que intentar adivinar y posteriormente comprobar con la ayuda del ordenador. Una vez completado este paso, en el ejercicio 3 se plantea un ejemplo de condicional que resuelve un problema similar al del ejercicio 1, pero esta vez utilizando una sentencia condicional. Llegados a este punto se hace reflexionar al alumno sobre las dos formas de llegar a una misma solución.

Superados todos estos pasos, se llega al ejercicio 4, un ejercicio de dificultad *similar* a los que el alumno encontrará en la prueba evaluable. Se trata de un enunciado fácil de entender para el alumno (en el ejemplo, sacar un mensaje por pantalla en función del precio de un café), a diferencia de las pruebas puntuables, que aunque de dificultad similar en cuanto a su resolución, suelen plantear problemas menos intuitivos.

Como se verá en el apartado de análisis de resulta-



Cuadro 1: Resultados de las respuestas obtenidas por los alumnos en las encuestas de satisfacción en los Grupos de Test (LA4, LA17 y LA24, se aplica la mejora docente) y Grupos de Control (LA6, LA8 y LA10, no se aplica la mejora docente). Las gráficas representan el porcentaje de respuestas en una escala del 1 al 5, donde 1 significa “Totalmente en desacuerdo” y 5 “Totalmente de acuerdo”. Las preguntas son las mismas para los Grupos de Test y de Control, excepto la 5, 6 y 10 que hacen referencia a las *hojas de entrenamiento*.

Sugerencias y Comentarios de los alumnos de los Grupos de Test (LA4, LA17 y LA24, donde se aplica la mejora docente)	
LA4	Evaluar el examen la parte de programación por ordenador y no a mano.
LA24	Me encanta la idea de las hojas de entrenamiento, nos podemos dar cuenta qué es lo que más preparado tenemos y lo que menos, salen dudas que al explicarlas el profesor quedan en la gran mayoría aclaradas y vemos el fruto de estas fichas de entrenamiento a la hora de hacer la prueba la última hora.
Sugerencias y Comentarios de los alumnos de los Grupos de Control (LA6, LA8 y LA10, donde no se aplica la mejora docente)	
LA6	Más clases prácticas y menos clases teóricas, o clases teóricas con ordenador delante.
LA8	Menos tiempo de dudas y mas tiempo para realizar los problemas.
LA8	errores que para una asignatura de primer curso no deberían restar tanto. Esos errores los puede cometer cualquiera, aunque un alumno sea el mejor de la clase puede cometer dos errores de estos y ver su nota disminuida a la mitad... HASTA EL PROFESOR en clase comete a veces esos errores.
LA10	Dar la opción de comenzar la sesión de evaluación del laboratorio sin tener que hacer una hora de repaso
LA10	Ofrecer actividades antes de realizar la actividad de laboratorio.

Cuadro 2: Sugerencias y Comentarios de los alumnos de los Grupos de Test (LA4, LA17 y LA24, donde se aplica la mejora docente) y Grupos de Control (LA6, LA8 y LA10, donde no se aplica la mejora docente).

dos, los estudiantes encuentran estos boletines de utilidad, y aunque puedan parecer muy sencillos de leer y resolver, no nos planteamos aumentar su dificultad o extensión, ya que observamos que la mayoría de los alumnos lo terminan en el tiempo justo de que disponen.

La preparación de estos boletines de entrenamiento supone un esfuerzo relativamente pequeño para el profesor. El tiempo de elaboración incluye una fase importante en la que debemos de elegir cuidadosamente los enunciados de los ejercicios: que sean fáciles de entender por los estudiantes, que les resulten cercanos (esto es, que tengan relación directa con su titulación, su día a día, etc.), y que su resolución resulte motivadora. Por ejemplo, puede que no les resulte igual de atractivo ni fácil de entender, un enunciado que pida un programa que calcule la concentración de un determinado elemento en una mezcla química, en función de algunas condiciones y parámetros, que el cálculo del precio de una pizza en función de la oferta elegida, el tamaño e ingredientes. También, a la hora de elaborar los boletines, hemos de pensar en qué conocimientos previos necesitan para llegar a resolver los enunciados planteados, y repasarlos mediante ejercicios si es necesario, aunque sea de forma breve y rápida.

5. Análisis de los resultados

Para medir el impacto de la mejora propuesta, pasamos, durante la última sesión de laboratorio, unas encuestas anónimas de satisfacción a un total de 6 grupos de laboratorio, 3 de ellos donde hemos aplicado la mejora propuesta (grupos de test) y 3 donde se ha seguido la metodología de cursos anteriores (grupos de control). La encuesta constaba de 11 preguntas de respues-

ta no obligatoria (ver Cuadro 1), en una escala del 1 al 5, donde 1 significaba “Totalmente en desacuerdo” y 5 “Totalmente de acuerdo”. Además había un cuadro de texto libre titulado “Sugerencias y comentarios”, para que los alumnos pudieran expresar sus inquietudes y sugerencias de manera abierta (ver Cuadro 2).

Las preguntas 1-4 hacen referencia al aprovechamiento y motivación con que los alumnos afrontan las sesiones de laboratorio, las preguntas 7-9 hacen referencia a cuestiones de aprendizaje y conocimiento previo en las sesiones de laboratorio. La pregunta 11 valora la influencia de las sesiones de laboratorio en el global de la asignatura.

Las respuestas de los alumnos revelan que el uso de esta metodología mejora significativamente el proceso de enseñanza-aprendizaje, ya que aumenta su participación en clase e incrementa su motivación y expectativas globales en los resultados finales de la asignatura.

Asimismo, de las respuestas del Cuadro 2 (sugerencias y comentarios de los estudiantes) se deduce que los alumnos están satisfechos con las *hojas de entrenamiento*, y que les gustaría que se utilizaran incluso en otras asignaturas. Los alumnos que no han seguido esta metodología, sugieren empezar antes la prueba de evaluación, o realizar actividades antes de la prueba, lo que indica que no encuentran útil la primera hora de resolución de dudas.

Por último, una vez finalizado el curso y el alumnado ha realizado todas las pruebas de evaluación, hemos realizado un análisis del impacto de la mejora propuesta sobre el rendimiento de los estudiantes en otros aspectos más generales de la asignatura. En particular, se ha analizado la influencia de la mejora en la nota de los exámenes escritos finales y el porcentaje de no presentados a las pruebas. Para ello, hemos comparado los

Grupo LAB	LA4	LA17	LA24
Grupo TEORÍA	B	E	E
N Grupo LAB	16	9	6
N Grupo TEORÍA	66	50	50
Media Examen Grupo LAB	4,81	5,78	5,14
Media Examen Grupo TEORÍA	4,7	4,56	4,56
NP Grupo LAB	1	2	0
NP Grupo TEORÍA	13	10	10
% NP Grupo LAB	6%	22%	0%
% NP Grupo TEORÍA	20%	20%	20%

Cuadro 3: Análisis de resultados al término de la evaluación. Se comparan los resultados de los estudiantes pertenecientes a cada grupo de laboratorio donde se ha aplicado la mejora propuesta, frente al conjunto de estudiantes del grupo de teoría al que pertenecen. Se muestra el número de estudiantes de cada grupo (N), la media de la nota del examen escrito, el número de estudiantes no presentados (NP), y el porcentaje que representan respecto al total (% NP).

resultados de los estudiantes pertenecientes a cada grupo de laboratorio donde se ha aplicado la mejora propuesta, frente al conjunto de estudiantes del grupo de teoría al que pertenecen, donde se incluyen también el resto de grupos de laboratorio donde no se ha aplicado la mejora. Tal como ya se ha mencionado antes, se observan diferencias en el rendimiento de los diferentes grupos, dependiendo del perfil de entrada de los estudiantes y de la titulación a la que pertenecen, razón por la cual no podemos extraer conclusiones simplemente realizando comparaciones entre grupos de laboratorio donde se ha aplicado la mejora frente a los grupos donde no se ha aplicado.

Como se puede ver en el Cuadro 3, para cada grupo de laboratorio y su correspondiente grupo de teoría al que pertenece, hemos recogido el número de estudiantes de cada grupo (N), la media de la nota del examen escrito, el número de estudiantes no presentados (NP), y el porcentaje que representan respecto al total (% NP). Del análisis de los resultados, se observa que la nota media de los grupos de laboratorio es superior en los tres casos a la del grupo al que pertenecen. En cuanto al porcentaje de no presentados, la cifra es inferior en dos de los grupos (LA4 y LA24). En el caso del grupo LA17, es ligeramente superior, aunque esta diferencia tampoco es muy significativa teniendo en cuenta que se trata de un grupo de sólo 10 estudiantes.

6. Conclusiones

En asignaturas de primer curso como la descrita en este trabajo, es importante que los alumnos maximicen el aprovechamiento de las sesiones presenciales. Para conseguirlo, es importante no presuponer, por ejemplo,

que los alumnos vayan a realizar un trabajo autónomo de estudio y que plantearán sus dudas en las sesiones presenciales, ya que los resultados demuestran que un gran porcentaje de ellos no lo hacen así. Aprovechar nuestra experiencia como docentes para guiarles en el proceso de aprendizaje, mejorará su desempeño y motivación, y para conseguirlo proponemos el uso de las *hojas de entrenamiento* como ejercicio previo a las pruebas de evaluación presencial.

De nuestra experiencia en la aplicación de esta propuesta hemos observado que los alumnos están muy activos en clase y les surgen dudas que nos plantean a los profesores, lo cual fomenta, a la vez, su participación en clase y mejora su rendimiento en general en la asignatura.

7. Agradecimientos

Este trabajo ha sido financiado por la Unitat de Suport Educatiu de la Universitat Jaume I en el marco de los Proyectos de Innovación Educativa 10G136-331 y 10G136-329 del curso 2012/13.

Referencias

- [1] M.C. Aranda, A.J. Fernández, J. Galindo y M. Trella. Valoración del Marco Docente de la Informática en la Ingeniería Técnica Industrial: Propuesta de una Nueva Metodología. En *Actas de las VII Jornadas de Enseñanza Universitaria de la Informática, Jenui 2001*, Palma de Mallorca, Julio 2001.
- [2] Ana Belén Moreno Díaz, Juan José Pantrigo y Rosalía Peña. Propuesta para la enseñanza de Informática en titulaciones de Ingeniería Química. En *Actas de las IX Jornadas de Enseñanza Universitaria de la Informática, Jenui 2003*, páginas 199 – 206, Cádiz, Julio 2003.
- [3] María Vaquero y Roberto Therón. Informática para profesionales de la Geología: docencia, aprendizaje y práctica. En *Actas de las IX Jornadas de Enseñanza Universitaria de la Informática, Jenui 2003*, páginas 215 – 222, Cádiz, Julio 2003.
- [4] Mavis Lis Stuart Cárdenas, Diana Aguilera Reina, Miguel Angel Díaz Martínez y Yadary Ortega González. Experiencia del ISPJAE en la formación Informática de los Ingenieros Industriales. En *Actas de las IX Jornadas de Enseñanza Universitaria de la Informática, Jenui 2003*, páginas 207 – 214, Cádiz, Julio 2003.
- [5] Jon Ander Gómez. Reflexiones sobre el desarrollo de competencias y habilidades en alumnos de primer curso. En *Actas de las XVIII Jornadas de Enseñanza Universitaria de la Informática, Jenui 2012*, páginas 177 – 184, Ciudad Real, Julio 2012.

Estrategias y tecnologías de la información para la coordinación de asignaturas multidisciplinares y multiáreas en el grado de Medicina en la Universitat Jaume I: experiencia con la asignatura “Recursos informáticos y documentación”

Oscar Coltell Simón
Departamento de Lenguajes y Sistemas
Informáticos
Universitat Jaume I
Castellón
oscar.coltell@uji.es

Ximo Granell Zafra
Departamento de Traducción y
Comunicación
Universitat Jaume I
Castellón
granell@trad.uji.es

Pedro Latorre Carmona
Departamento de Lenguajes y Sistemas
Informáticos
Universitat Jaume I
Castellón
latorre@lsi.uji.es

Ricardo Tosca Segura
Unidad Predepartamental de Medicina /
Servicio de Pediatría
Universitat Jaume I / Hospital General
Castellón
tosca@upim.uji.es

José Salvador Sánchez Garreta
Departamento de Lenguajes y Sistemas
Informáticos
Universitat Jaume I
Castellón
Salvador.Sanchez@lsi.uji.es

Luis Vicente Lizan Tudela
Unidad Predepartamental de Medicina
Universitat Jaume I / Servei Valencia de
Salut
Castellón
lizan@eco.uji.es

Resumen

Antecedentes: El grado de Medicina en la Universitat Jaume I, empezó a impartirse en el curso 2011/2012 con un diseño curricular sostenible donde las asignaturas están compartidas por varias áreas y departamentos. Una de estas asignaturas, “Recursos informáticos y documentación” (MD1112), con 6 créditos ECTS en el primer semestre del segundo curso, introduce las Tecnologías de Información y Comunicación (TIC), el método científico y la medicina basada en la evidencia. En la memoria VERIFICA de ANECA se comparte entre cinco áreas de conocimiento distintas: Lenguajes y Sistemas Informáticos (LSI, D. de Lenguajes y Sistemas Informáticos); Ciencias de la Computación e Inteligencia Artificial (CCIA), Arquitectura de Computadores (AC) (ambas del D. Ingeniería y Ciencias de la Computación); Biblioteconomía y Documentación (BID, D. de Traducción y Comunicación); y Medicina (MED, Unidad Pre-departamental de Medicina). En el curso

2012-2013, el total de carga docente presencial se distribuyó entre las áreas: LSI con 2 profesores funcionarios –Se acordó entre los dos departamentos de informática que la docencia se asignara sólo a LSI-; BID con 1 profesor contratado a tiempo completo; y MED con 2 profesores asociados a tiempo parcial con dedicación laboral en hospital y atención primaria y reducida disponibilidad académica. Esta situación planteaba muchos problemas de gestión y coordinación con estrategias y técnicas tradicionales y un reto en la normalización de estilos docentes y enfoques didácticos. Por tanto, era necesario abordar esto con nuevos puntos de vista, tanto en estrategias de gestión como en el uso intensivo de TIC de soporte.

Objetivo: El objetivo de este trabajo es presentar y discutir la experiencia, durante el curso 2012-2013, en coordinación y gestión virtual intensiva de la asignatura MD1112 basada en un enfoque vertical de procesos y otro horizontal de recursos on-line.

Metodología: Este enfoque ya se aplicó con éxito en varios procesos verticales: elaboración de la guía

docente; reparto de actividades y horas por área y profesor; seguimiento y evaluación. El enfoque horizontal se basa en el uso intensivo de la plataforma Moodle dentro del aula virtual oficial de la UJI, donde se gestionan en paralelo dos cursos: el curso docente para los alumnos (84 matriculados), y otro específico para la coordinación sólo para profesores.

Resultados: En el curso de coordinación, las reuniones presenciales se han reducido a lo imprescindible, pero son mucho más efectivas porque son precedidas por intensos debates mediante un foro de discusión, o son suplidas por los mismos. También se han aplicado otros recursos de Moodle para complementar los mecanismos de coordinación: tablón de anuncios, encuestas, repositorio de documentación y simulación de escenarios. En el curso de docencia, se ha amplificado la atención a los alumnos mediante tutorías virtuales y foro de discusión; la evaluación continua está apoyada por tareas de entregas y los alumnos pueden seguir su progreso mediante la estructura de calificación que recoge todas las actividades evaluables.

Conclusiones: La experiencia ha resultado positiva y se ha ahorrado mucho tiempo y esfuerzo frente a una coordinación clásica. Los alumnos también han valorado positivamente la facilidad en el seguimiento de la asignatura.

Abstract

Background: The new Medicine Degree at the Universitat Jaume I started in the year 2011/2012 with a sustainable curriculum where subjects are shared by several areas and departments. One of these subjects, "Computer resources and documentation" (MD1112) with 6 ECTS credits in the first semester of the second year, introduces the Information and Communication Technologies (ICT), the scientific method and the evidence-based medicine. In ANECA VERIFICA memory this subject is shared by five different knowledge areas: Computing Languages and Systems (LSI, D. of Computing Languages and Systems); Computer Science and Artificial Intelligence (CCIA), Computer Architecture (CA) (both of D. of Engineering and Computer Science); Biblioteconomy and Documentation (B&D, D. of Translation and Communication); and Medicine (MED, Pre-departmental Unit of Medicine). During the academic year 2012-2013, the total classroom teaching load was distributed on areas: LSI with 2 faculty members -under agreement between both IT departments, teaching was assigned only to LSI-; B&D with 1 full time hired teacher; and MED with 2 part-time lecturers from hospital and primary care and reduced academic availability. This situation posed many management and coordination problems using traditional strategies and techniques, and moreover, a challenge in stand-

ardizing teaching styles and approaches. It was therefore necessary to address this with new points of view, both with management strategies and ICT intensive support.

Objective: The objective of this paper is to present and discuss the experience during the academic year 2012-2013, in virtual course coordination and intensive management of MD1112 based on a vertical processes approach and horizontal on-line resources.

Methodology: This approach has already been applied successfully in several vertical processes: development of the teaching guide; distribution of activities and schedule by area and teacher; monitoring and evaluation. The horizontal approach is based on the intensive use of the UJI Moodle virtual classroom, where two parallel courses are managed: the course for students (84 enrolled), and another specifically for coordination only for teachers.

Results: In the coordination course, face-to-face meetings have been dramatically reduced, but they are much more effective because they are preceded by intense discussions through a discussion forum, or are supplied by discussion forum. Other Moodle resources have also been applied to complement coordination mechanisms: message board, surveys, documentation repository and scenario simulation. In the students course, the attention to students has been amplified through virtual personal and discussion forums; continuous assessment is supported by delivery tasks and students can track their progress through the qualification structure that includes all the activities evaluated.

Conclusions: The experience has been positive and we saved so much time and effort versus classical coordination. Students also have welcomed the ease in tracking the subject.

Palabras clave

Grado de Medicina, coordinación virtual, estrategias y técnicas on-line, uso de aula virtual con Moodle.

1. Introducción

El nuevo grado de Medicina en la Universitat Jaume I (UJI) [1] arrancó en el curso 2011/2012, por lo que, en 2013 están funcionando el primer y segundo cursos. La asignatura "Recursos informáticos y documentación", cuyo código interno es "MD1112", tiene asignados 6 créditos ECTS y se ha programado en el primer semestre del segundo curso. MD1112 es la única asignatura que introduce las Tecnologías de Información y Comunicación (TIC), el método científico, la documentación médica y la medicina basada en la evidencia, como contenido fundamental de la misma. Esto determinó que, en el diseño de la asignatura en la memoria VERIFICA [2], se asignara su docencia a cinco áreas de conocimiento distintas:

Lenguajes y Sistemas Informáticos, Ciencia de la Computación e Inteligencia Artificial, Arquitectura de Ordenadores, Biblioteconomía y Documentación y Medicina.

1.1. Desarrollo de la asignatura en el curso 2012/2013

En su debut en el actual curso 2012-2013, el vicerrectorado de Ordenación Académica le asignó un total de 122 horas presenciales de carga docente en la planificación del curso, distribuyendo dichas horas entre las áreas de Lenguajes y Sistemas Informáticos (LSI: 71,2 h – previamente se acordó que solamente esta área de informática asumiera la docencia), Biblioteconomía y Documentación (BID: 30,5 h) y Medicina (MED: 20,3 h). Posteriormente, los correspondientes departamentos (D. de Lenguajes y Sistemas Informáticos, D. de Traducción y Comunicación y Unidad Pre-departamental de Medicina) asignaron al profesorado para impartir la docencia: LSI 2 profesores; BID 1 profesor; y MED 2 profesores. Dichos profesores son los autores de este trabajo y el profesor responsable es el primer autor. Los profesores de LSI y BID son funcionarios y contratados a tiempo completo. En cambio, los profesores de MED son asociados a tiempo parcial con dedicación laboral en hospital y atención primaria.

1.2. El problema de la gestión y coordinación entre profesores y áreas

Dado que el profesorado asignado está adscrito a distintas áreas y departamentos y, además por su régimen de dedicación docente, algunos tienen disponibilidad reducida, existía el riesgo de que surgieran problemas de gestión y coordinación, tanto entre profesores como entre profesores y alumnos. Si esto se llevara a cabo con estrategias y técnicas tradicionales (estructura piramidal, reuniones presenciales, comunicación basada en papel, teléfono y correo electrónico, etc.), el nivel de riesgo sería alto. Por otra parte, la combinación de profesores adscritos a determinadas áreas que nunca habían trabajado juntas, suponía un reto en la normalización de estilos docentes y enfoques didácticos. Por tanto, era necesario abordar la coordinación de la asignatura con nuevos puntos de vista, tanto en estrategias de gestión como en el uso intensivo de TIC de soporte.

1.3. Desarrollo de la asignatura en el curso 2012/2013

Entonces, el objetivo de este trabajo es presentar y discutir la experiencia, durante el año 2012, en coordinación y gestión de la asignatura MD1112 basada en un enfoque vertical de procesos y otro horizontal de recursos en línea aplicados, tanto a la docencia como a la planificación y seguimiento. Este enfoque

ya se aplicó con éxito en el proceso de la elaboración de la guía docente, donde intervino un profesor por cada área (R. Tosca, X. Granell y O. Coltell); en el proceso de reparto de actividades y horas por área y profesor (los seis profesores); en el proceso de seguimiento y en el proceso de evaluación. El enfoque horizontal se basa en el uso intensivo de la plataforma Moodle dentro de la infraestructura de aula virtual oficial de la UJI, donde, además del curso oficial para MD1112, se solicitó otro específico para la coordinación.

En este trabajo, en la Sección 2 se describe la infraestructura del Aula Virtual y su aplicación a la asignatura. En la Sección 3 se muestran las conclusiones obtenidas.

2. El Aula Virtual de la UJI

La UJI dispone de una infraestructura informática específica para el apoyo en línea a la docencia y docencia virtual denominada “Aula Virtual” (AV) (<http://aulavirtual.uji.es>). El elemento esencial de esta infraestructura es la plataforma Moodle, una aplicación web gratuita que los educadores pueden utilizar para crear sitios de aprendizaje efectivo en línea (<http://www.moodle.org>). Moodle es un Sistema de Gestión de Cursos de Código Abierto (*Open Source Course Management System, CMS*), conocido también como Sistema de Gestión del Aprendizaje (*Learning Management System, LMS*) o como Entorno de Aprendizaje Virtual (*Virtual Learning Environment, VLE*).

Habitualmente, el AV contiene cursos correspondientes a las asignaturas de títulos y másteres oficiales y propios, formación interna del PAS y PDI, y otros tipos de cursos. En los títulos y másteres, la creación de cursos la realiza el administrador del AV, bajo demanda del profesor responsable de cada asignatura antes de iniciarse el curso académico, y activando un mecanismo que da de alta automáticamente a todos los profesores adscritos y alumnos matriculados en cada asignatura.

2.1. La asignatura “Recursos informáticos y documentación” en el AV




Con respecto a la asignatura Recursos informáticos y documentación, los autores, como profesores adscritos a la misma, han aprovechado el AV en las dos funciones principales de todo desarrollo académico de materias a cargo de un grupo de profesores en un entorno virtual: (A) la gestión académica, apoyo a la docencia y entorno virtual de aprendizaje; y (B) la organización, coordinación y gestión de recursos.

2.2. El AV en la gestión académica, apoyo a la docencia y entorno virtual de aprendizaje

Para poder llevar a cabo adecuadamente la función A de gestión académica, apoyo a la docencia y entorno virtual de aprendizaje, se solicitó un curso específico en el AV, dentro de la sección de docencia del curso 2012/2013 (Figura 1).

ID1112. Recursos Informáticos y Documentación

Curso académico 2012/2013

-  MD1112. Foro de Noticias y Novedades
-  MD1112. Foro de Consultas y Discusión General
-  MD1112. Consultas con los profesores (tutorías virtuales)

Presentación y objetivos

-  MD1112. Guía Docente (castellano)
-  MD1112. Guia Docent (valencià)

Departamentos y áreas implicadas:

Área	Departamento	Profesor	Coordinador
Llenguatges i Sistemes Informàtics	Llenguatges i Sistemes Informàtics	Oscar Coltell	SI
		J. Salvador Sánchez	
		Pedro Latorre	
Biblioteconomia i Documentació	Traducció i Comunicació	Ximo Granell	
Medicina	Medicina	Luis Lizán	
		Ricardo Tosca	

Figura 1: Curso en el AV correspondiente a la asignatura MD1112.

Si bien las aulas de la facultad en las que se desarrollan las distintas actividades de la asignatura son el espacio habitual en torno al que se programa la docencia presencial y en el que tiene lugar la interacción principal con el alumnado de Medicina, el AV supone una valiosa extensión de estas aulas. Más allá de entender el curso del AV como un mero repositorio de apuntes y documentos de la asignatura, se planteó como un entorno (virtual) de apoyo continuado a la docencia impartida en las sesiones presenciales y un espacio para desarrollar el trabajo autónomo y en grupo que tiene lugar fuera de las horas lectivas.

Organización

-  MD1112. FICHA LLEU
-  MD1112. Calendario detallado. Ver. 1, 20121113
-  MD1112. Temario de la asignatura. Ver 1
-  MD1112. Bibliografía. Ver. 1
-  Normativa de permanència i progrés per als ensenyaments universitaris oficials de Grau i Màster de la Universitat Jaume I (versión en español)
-  MD1112. Estructura de la Evaluación. Ver. 1, 20121106
-  MD1112. Evaluación BID
-  MD1112. Temario de la asignatura
-  MD1112. Bibliografía
-  MD1112. Evaluación. Estructura general

Figura 2: Sección de organización en AV de la asignatura MD1112.

Como muestra la Figura 2, haciendo uso de la gran funcionalidad que ofrece un entorno de aprendizaje

virtual basado en Moodle, se han organizado en un mismo espacio todos los elementos que forman parte de la docencia: los materiales docentes, las actividades prácticas y los canales de comunicación. Además, se ha dedicado un apartado específico para publicar la información correspondiente a la gestión de la asignatura que se consideraba de interés para el alumnado.

2.3. El AV en la organización, coordinación y gestión de recursos

Dada la complejidad organizativa de la asignatura y los condicionantes que implica compartir la docencia entre tres áreas y seis profesores (Sección 1.2.), se solicitó la creación de un nuevo curso en el AV para dedicarlo específicamente a la organización docente, coordinación y gestión de recursos desde el punto de vista del profesor responsable de la asignatura (Figura 3).

Grado en Medicina

COORDINACIÓN: MD1112. Recursos informáticos y documentación

(Informática)

Curso académico 2012/2013

Departamentos y áreas implicadas:

Área	Departamento	Persona de contacto	Coordinador
Arquitectura i Tecnologia de Computadors	Ingeniería y Ciencia de los Computadores	Oscar Coltell	X
Biblioteconomia i Documentació	Traducció i Comunicació	Dora Sales, Ximo Granell	
Ciència de la Computació i Intel·ligència Artificial	Ingeniería y Ciencia de los Computadores	Oscar Coltell	X
Llenguatges i Sistemes Informàtics	Llenguatges i Sistemes Informàtics	Oscar Coltell	X
Medicina	Medicina	Luis Lizán, Ricardo Tosca	

Actualización del curso: 22 de Enero de 2013

-  Novedades
-  Foro de Noticias y Novedades
-  Foro de discusión del diseño y gestión de la asignatura
-  Foro de discusión en el proceso de elaboración
-  Consultas particulares

Figura 3: Curso en el AV correspondiente a la coordinación de

En este curso solamente están matriculados los profesores de la asignatura para evitar problemas de filtraciones y simplificar el mecanismo de control de usuarios y destinatarios de recursos en Moodle.

En el diseño del curso se ha tenido en cuenta la estructuración en función de las estrategias asociadas a la organización, coordinación y gestión de recursos, y a los mecanismos que las desarrollan. Esto se describe en la siguiente subsección.

2.4. Estrategias y mecanismos de coordinación y gestión

Para llevar a cabo con eficiencia la organización y coordinación docentes y la gestión de recursos entre los profesores, se plantearon las siguientes estrategias: (1) Facilitar la gestión, almacenamiento y recuperación de la documentación de coordinación; (2)

Establecer los canales de comunicación entre los profesores de tipo general y privado; (3) Promover el debate virtual como alternativa a las reuniones presenciales; y (4) Ofrecer un entorno inerte para pruebas de actividades que luego se incluirán en el curso del AV para los estudiantes.

Para ello se ha estructurado el curso por temas, donde cada tema contiene los mecanismos necesarios para el desarrollo de las estrategias, además de temas con información general. Los principales mecanismos son los siguientes:

- *Estrategia 1: Facilitar la gestión, almacenamiento y recuperación de la documentación de coordinación.* Se han usado los distintos elementos de Moodle para la inclusión de información o metainformación, como páginas, URL, carpetas, archivos y etiquetas, organizados en temas correspondientes a las distintas actividades realizadas (se detallarán más adelante).
- *Estrategia 2: Establecer los canales de comunicación entre los profesores de tipo general y privado.* Se han creado distintos foros con misiones específicas y se han concentrado en una única sección (Figura 3). Estos foros son: el Foro de Noticias y Novedades, para difusión unilateral; el Foro de discusión del diseño y gestión de la asignatura, para la gestión del desarrollo de las actividades de la asignatura; el Foro de discusión en el proceso de elaboración, para el diseño de la asignatura; y Consultas particulares, para la intercomunicación punto a punto entre profesores.
- *Estrategia 3: Promover el debate virtual como alternativa a las reuniones presenciales.* Se han ido creando los hilos de discusión pertinentes en los dos foros de discusión para acometer el debate de temas específicos. El apoyo documental de la discusión se ha conseguido facilitando la documentación según los mecanismos de la Estrategia 1.
- *Estrategia 4: Ofrecer un entorno inerte para pruebas de actividades que luego se incluirán en el curso del AV para los estudiantes.* Dado que no hay estudiantes matriculados, en este curso se pueden crear y probar actividades orientadas a los estudiantes para ponerlas a punto y pasarlas al curso docente mediante el mecanismo de importación de Moodle. Entre las actividades ofrecidas por la plataforma, las más comunes son: Base de datos; Chat; Consulta; Cuestionario; Glosario; Módulo de encuesta; Taller; y Tarea.

En la gestión y coordinación de la asignatura, las estrategias forman el enfoque vertical y los mecanismos el enfoque horizontal en la plataforma Moodle como entorno de coordinación del equipo docente.

3. Experiencia de coordinación y organización docente

La utilización de estas plataformas como herramientas de coordinación ha contribuido positivamente a la labor docente, ha permitido una mejor integración de los esfuerzos del profesorado, ha servido de canal de comunicación continuado y ha facilitado la toma de decisiones. Para acometer las distintas tareas de gestión, se generaron los siguientes subespacios de trabajo en el AV de coordinación:

- Documentos relativos a la organización docente de la asignatura (horarios, distribución de actividades y horas entre las distintas áreas, etc.).
- Registro de peticiones de software y de necesidades de los espacios docentes de la asignatura.
- Archivo de documentación relativa a las convocatorias de la asignatura (elaboración del examen, revisión y notas, etc.)
- Versiones de la guía docente de la asignatura.
- Documentos de trabajo con propuestas de modificaciones de la guía docente y plan de estudios.
- Publicaciones científicas: borradores de artículos de difusión de la experiencia docente, documentación de referencia, documentos de apoyo con datos recopilados sobre los contenidos de los artículos y versiones finales presentadas para su publicación.

Cabe mencionar la utilidad de los canales de comunicación abiertos, tanto en la gestión (Sección 2.3) como en la interacción con el alumnado (foros de discusión generales o específicos, diálogos de consulta con el profesorado, cuestionarios de opinión sobre las actividades, etc.)

En este último sentido, también se ha perseguido mejorar la comunicación entre alumnado y profesorado gestionando adecuadamente los recursos de espacio en el AV. En primer lugar, por medio de espacios para recopilar valoraciones del alumnado a lo largo de todo el transcurso de la asignatura, a modo de diario reflexivo de cara al alumnado y de realimentación para el profesorado sobre su acción docente (Figura 4). En segundo lugar, con la incorporación de un espacio de trabajo en grupo para favorecer, tanto el seguimiento y tutorización por parte del profesorado, como el trabajo en grupo y autónomo. Además, este espacio virtual de trabajo colaborativo ha facilitado la evaluación continua del proceso de aprendizaje del alumnado y ha permitido desarrollar una de las actividades principales de la asignatura, un taller de investigación en Medicina (Figura 5), en un entorno completamente electrónico y apoyándose en el uso de las TIC.

Editar categorías e ítems: Vista simple

Nombre	Cálculo total	Puntos extra	Calif. máx.	Acciones	Seleccionar
MD1112 - Recursos informáticos i documentación (2012/2013)	Suma de calificaciones	-	-	🔍 🗑️ 📄	Todos Ninguno
1-Examen escrito	Suma de calificaciones	-	-	🔍 🗑️ 📄	Todos Ninguno
Examen 1º convocatoria: Cuestión 01. NOTAS	-	☑️	3,00	🔍 🗑️ 📄	☑️
Examen 1º convocatoria: Cuestión 02. NOTAS	-	☑️	3,00	🔍 🗑️ 📄	☑️
Examen 1º convocatoria: Cuestión 03. NOTAS	-	☑️	3,00	🔍 🗑️ 📄	☑️
Examen 1º convocatoria: Cuestión 04. NOTAS	-	☑️	3,00	🔍 🗑️ 📄	☑️
Examen 1º convocatoria: Cuestión 05. NOTAS	-	☑️	3,00	🔍 🗑️ 📄	☑️
Examen 1º convocatoria: Cuestión 06. NOTAS	-	☑️	3,00	🔍 🗑️ 📄	☑️
Examen 1º convocatoria: Cuestión 07. NOTAS	-	☑️	3,00	🔍 🗑️ 📄	☑️
Examen 1º convocatoria: Cuestión 08. NOTAS	-	☑️	3,00	🔍 🗑️ 📄	☑️
Examen 1º convocatoria: Cuestión 09. NOTAS	-	☑️	3,00	🔍 🗑️ 📄	☑️
Examen 1º convocatoria: Cuestión 10. NOTAS	-	☑️	3,00	🔍 🗑️ 📄	☑️
Examen Nota Total	-	☑️	3,00	🔍 🗑️ 📄	☑️
2-Laboratorio	Suma de calificaciones	-	-	🔍 🗑️ 📄	Todos Ninguno
2-LABORATORIO: Evaluación de Práctica de Medicina Basada en Evidencia	-	☑️	1,00	🔍 🗑️ 📄	☑️
Entrega del ejercicio de Base de Datos con LibreOffice Base	-	☑️	100,00	🔍 🗑️ 📄	☑️
Laboratorio Nota Total	-	☑️	3,00	🔍 🗑️ 📄	☑️
3-Ejecución de tareas y prácticas	Suma de calificaciones	-	-	🔍 🗑️ 📄	Todos Ninguno
Evaluación de la P1	-	☑️	100,00	🔍 🗑️ 📄	☑️
Entrega de la Práctica 2	-	☑️	100,00	🔍 🗑️ 📄	☑️
Entrega de la Práctica 3	-	☑️	100,00	🔍 🗑️ 📄	☑️
Entrega de la Práctica 4	-	☑️	100,00	🔍 🗑️ 📄	☑️
Tareas Nota Total	-	☑️	2,00	🔍 🗑️ 📄	☑️
4-Presentaciones orales y pósters	Suma de calificaciones	-	-	🔍 🗑️ 📄	Todos Ninguno
Entrega del póster	-	☑️	100,00	🔍 🗑️ 📄	☑️
Presentaciones Nota Total	-	☑️	2,00	🔍 🗑️ 📄	☑️
Asignatura Nota Total	-	☑️	10,00	🔍 🗑️ 📄	☑️

Figura 4: Estructura de valoración del alumnado.

Taller de Investigación Médica (TIM)

→ 🔍 🗑️ 📄 ✖️ 📄 📄	Instrucciones para la realización del Taller de Investigación Médica → 🔍 🗑️ 📄 ✖️ 📄 📄
👤 Espacio de trabajo en grupo → 🔍 🗑️ 📄 ✖️ 📄 📄	Evaluación del TIM → 🔍 🗑️ 📄 ✖️ 📄 📄
📄 Entrega del póster → 🔍 🗑️ 📄 ✖️ 📄 📄	Hojas de evaluación intergrupales → 🔍 🗑️ 📄 ✖️ 📄 📄
🔍 Valoración del TIM → 🔍 🗑️ 📄 ✖️ 📄 📄	

Figura 5: Estructura del taller de investigación en Medicina.

Para medir la respuesta de los estudiantes con respecto a la introducción del método científico en los contenidos del curso y la extensión del plan de estudios de licenciatura, se diseñó una breve encuesta con preguntas según el modelo de escala Likert que fue contestada por los estudiantes hacia el fin del período de clase.

4. Resultados

Por lo que respecta a los canales de comunicación en el ámbito de la coordinación, la utilización de los foros destinados a cada cuestión garantizó la comunicación entre todo el profesorado y originó un volumen considerable de tráfico de información. Por ejemplo, en el foro de diseño y gestión se iniciaron un total de 15 temas, en los cuales se intercambiaron 106 mensajes y se trataron cuestiones como los siguientes: organización y reparto de actividades y horas, pautas de comunicación entre alumnado y profesora-

do, incidencias y cambios, solicitudes de convalidaciones, evaluación de la asignatura, propuestas de modificaciones al plan de estudios y la guía docente y difusión de experiencias docentes por medio de publicaciones.

Pese a estar los seis profesores de la asignatura en el mismo campus, las reuniones presenciales se han reducido a lo imprescindible por dificultades de agenda, cuatro en un periodo comprendido entre febrero de 2012 a febrero de 2013. Así, gracias a los mecanismos generados para la comunicación y el debate, promoviendo el trabajo colaborativo y la discusión en línea, dichas reuniones han sido suplidas por intensos debates mediante los foros de discusión correspondientes. Por tanto, las reuniones presenciales han resultado cortas y efectivas porque se han preparado previamente en línea.

Es interesante destacar que los estudiantes reclamaron sobre la excesiva diversidad de actividades ECTS de la asignatura, algo común en el resto de asignaturas del grado en los primeros cursos, porque cada actividad requiere su propio esfuerzo específico y, a condición de que las actividades sean compartidas por más de un profesor, no era fácil tener una visión homogeneizada de los objetivos correspondientes. Esta diversidad contribuía además a elevar la complejidad del calendario académico 2012-2013 en tercer y cuarto semestres. Dado que esto último venía fijado oficialmente desde antes del inicio del curso, los profesores de MD1112 no teníamos competencias para cambiarlo.

Con respecto a la encuesta, de una población total de 81 alumnos matriculados, se obtuvieron 17 respuestas (21%) y 11 hombres (64,71%) y 6 mujeres (35,29%), con una media de 21,41 años de edad. Hay un cuasi-equilibrio entre los encuestados que estaban satisfechos en el desarrollo teórico y práctico del método científico en la MD1112, el 47,06%, frente a los participantes no satisfechos, 52,94%. La mayoría de ellos conocían el método científico aplicado a las disciplinas científicas, 82,35%, y que era necesario introducir el método científico en los estudios de medicina, 94,12%. También había una opinión generalizada en relación con que el método científico podía ayudar en lo académico antes de graduarse, 70,58%, y en el desarrollo profesional una vez graduados, 88,24%.

En cuanto a cuándo introducir el método científico, la mayor parte de los participantes opinaba que debía realizarse en el primer año, 88,24%, contra los que preferían el segundo año, 64,71%. En el caso de MD1112, la satisfacción sobre la aplicación y el desarrollo del método científico fue que el 94,12% estaba de acuerdo con la introducción del método científico, pero un promedio de 67,65% estaban en desacuerdo con la forma en que este se había realizado. Afortunadamente, la mayoría de ellos pensaban que podríamos mejorar el estudio y la aplicación del método científico en la MD1112, 76,47%.

5. Conclusiones

Este documento recoge la experiencia, con respecto a la organización y coordinación docentes, en el debut de la asignatura "MD1112. Recursos informáticos y documentación" del nuevo grado de Medicina en la Universitat Jaume I, durante el curso 2012/2013.

En la asignatura Recursos informáticos y documentación, los autores, han aprovechado el AV creando y usando dos cursos virtuales distintos: uno para la gestión académica, apoyo a la docencia y entorno virtual de aprendizaje; y otro para la organización, coordinación y gestión de recursos.

Con respecto al uso del AV en la gestión académica, apoyo a la docencia y entorno virtual de aprendizaje, se han organizado en un mismo espacio todos los elementos que forman parte de la docencia. Con

respecto al uso del AV en la organización, coordinación y gestión de recursos, se ha diseñado un curso basado en un enfoque vertical de estrategias asociadas a la organización, coordinación y gestión de recursos, y otro horizontal con los mecanismos que las desarrollan.

Se ha podido confirmar la utilidad de los canales de comunicación abiertos, tanto en la gestión como en la interacción con el alumnado. También se ha mejorado la comunicación entre alumnado y profesorado gestionando adecuadamente los recursos de espacio en el AV. Se ha incorporado un espacio de trabajo en grupo que ha favorecido el seguimiento y tutorización por parte del profesorado y el trabajo en grupo y autónomo, facilitando además la evaluación continua del proceso de aprendizaje del alumnado.

Los resultados de la encuesta y los comentarios de los alumnos han sido muy valiosos en cuanto a animarnos a solicitar la modificación de la ficha VERIFICA, para racionalizar y reducir actividades, así como carga presencial de la asignatura, y también para replantear la estructura de contenidos a fin de mejorar la introducción del método científico.

Y gracias a los mecanismos de comunicación y debate, las reuniones presenciales se han reducido a las imprescindibles habiendo sido muy efectivas por su preparación previa en línea. La experiencia ha resultado positiva y todos los profesores confirman que se ha ahorrado mucho tiempo y esfuerzo frente a una coordinación clásica, y que se ha aprendido mucho en la combinación práctica de recursos en línea y estrategias de coordinación.

Referencias

- [1] Grado en Medicina de la Universitat Jaume I. <http://www.uji.es/ES/infoest/estudis/titols/e@/2891/?pTitulacionId=229>. Acceso 15/02/2013.
- [2] Universitat Jaume I. *Memoria del Título Graduado o Graduada en Medicina por la Universitat Jaume I*, julio 2010. <http://www.uji.es/bin/organs/vices/veees/titula/med/maneca.pdf>. Acceso 15/02/2013.

Diseño en FPGAs como Método Docente para la Asignatura de Redes en Chip

Antoni Roca, Jose María Martínez, José Flich
Grup de Arquitectures Paraleles
Universitat Politècnica de València
46022 València
anrope2@gap.upv.es

Resumen

La formación de los alumnos en el Máster de Ingeniería de Computadores de la Universitat Politècnica de València requiere de un sistema integral de aprendizaje que les acerque a los sistemas utilizados en la Industria. En el caso de la asignatura Redes en Chip, se ha introducido un novedoso sistema que integra tanto los entornos de simulación típicos en el ámbito de la investigación, como la implementación en sistemas reales basados en FPGAs permitiendo un prototipado directo de las soluciones obtenidas en las clases teóricas. Dado que el entorno de programación hardware con FPGA es relativamente desconocido en Informática, se han desarrollado unas prácticas guiadas en las que el alumno debe implementar una red en el chip paso a paso, al mismo tiempo que aprende a programar la FPGA. El aprendizaje paso a paso permite a los alumnos reforzar los conceptos vistos en teoría. Además, el entorno hardware de la FPGA le permite al alumno de una manera atractiva e intuitiva entender conceptos abstractos. A las ventajas que introduce el diseño de redes en el chip con FPGA se le suma la motivación del alumno propia de trabajar con herramientas de diseño típicas en la industria dedicada al prototipado y la fabricación de chips. En esta ponencia describimos la herramienta desarrollada y el método de aprendizaje de los alumnos, recorriendo los diferentes módulos y detallando las consiguientes habilidades que el alumno incorpora a su currículo. Durante la puesta en marcha de la experiencia se ha obtenido un elevado grado de interés y motivación por parte del alumno.

Palabras clave

Redes en el Chip, FPGA, Hardware.

1. Introducción

La asignatura Redes en Chip se imparte en el Máster de Ingeniería de Computadores en el departamento DISCA de la Universitat Politècnica de València. En dicha asignatura, se imparten los conceptos de diseño y análisis de redes dentro de un chip [1, 2], un concepto que surgió a inicios del año 2000, para dar servicio a la creciente complejidad de los sistemas en chip, necesitando la conexión de múltiples procesadores en el denominado ámbito multicore o many-core.

Las redes en chip se suelen analizar por medio de potentes herramientas de simulación, que modelan el comportamiento de cada componente de la red. Ahora bien, en el contexto de las redes en el chip, la arquitectura de computadores se mezcla con el diseño electrónico, ya que en las redes en chip, su diseño a nivel de puertas lógicas, cobra especial relevancia, afectando a las prestaciones y funcionalidad final del sistema bajo análisis.

Por ese motivo, durante el curso 2012/2013 se inició una experiencia práctica en la asignatura, con el fin de introducir al alumno, tanto en el uso de simuladores como en el diseño de la red a nivel de lenguaje de definición hardware. En concreto, el alumno evalúa una determinada configuración con el simulador, para después implementar la solución en una FPGA. Con dicha experiencia, la motivación del alumno, y el nivel de asimilación en la asignatura se han visto incrementados significativamente.

En la siguiente sección describimos la plataforma hardware utilizada.

2. Plataforma Hardware

Una *Field Programmable Gate Array* (FPGA) es un dispositivo semiconductor hecho a base de bloques de lógica, llamados celdas, cuya funcionalidad puede programarse. En su definición más básica, una celda puede implementar un registro y una función lógica. La pro-

gramación de cada una de las celdas, así como, la conexión entre ellas, permite que una FPGA pueda emular cualquier diseño, como por ejemplo una red en el chip, o un procesador.

El diseño de hardware sobre FPGAs no es tan eficiente – en prestaciones ni consumo de potencia – como implementar el mismo diseño en un circuito integrado específico. No obstante, el interés de la industria por las FPGAs ha crecido de forma exponencial, conforme la capacidad, robustez, fiabilidad y versatilidad de dichos dispositivos crece. En primer lugar, son sencillas de programar, y por tanto, requieren de menor coste – en tiempo y dinero – de implementación. Además, las FPGAs son re-programables, es decir, su funcionalidad puede modificarse tantas veces como sea necesario. Sin embargo, la motivación principal para el uso de las FPGAs, es su posibilidad de ser usadas para prototipar sistemas finales, lo que le da una credibilidad mayor a la solución cuando es comparada con un modelo de simulación.

Para poder trabajar con FPGAs básicamente se requieren conocimientos básicos en:

- Lenguaje de diseño hardware
- Entorno FPGA: estructura y funcionamiento
- Herramientas de programación de FPGAs

Respecto al diseño hardware se ha utilizado Verilog como lenguaje de implementación hardware (HDL). Se ha descartado otras opciones más cercanas a lenguajes de programación más conocidos, como SystemC, para fomentar el interés de los alumnos, ya que Verilog es un lenguaje muy utilizado en la industria de diseño de FPGAs, poco conocido por los estudiantes.

Para la asignatura de Redes en Chip, hemos utilizado una FPGA Spartan-6 de Xilinx [6], empotrada en una placa NEXYS 3 de Digilent 1 [4]. Dicha placa contiene, además de la FPGA, diversos elementos como leds, interruptores, y monitores, que la hacen muy atractiva para la docencia.

Uno de los principales problemas que hemos observado a la hora de introducir las FPGAs en el máster de informática ha sido la enorme diversidad de conocimientos previos de los alumnos. Un 25 % de los alumnos ya conocía e incluso había programado una FPGA, mientras que el resto no sólo desconocía los HDL sino el funcionamiento de las FPGAs. Para poder armonizar los distintos niveles de conocimientos de los alumnos, se ha estructurado la experiencia en 2 bloques.

La primera parte de la experiencia consiste en introducir a los alumnos en el lenguaje de descripción hardware Verilog y la programación de FPGAs. Este aprendizaje se realiza diseñando módulos básicos de una red en el chip, para que los alumnos refuercen conceptos básicos de redes en el chip vistos en teoría, especial-

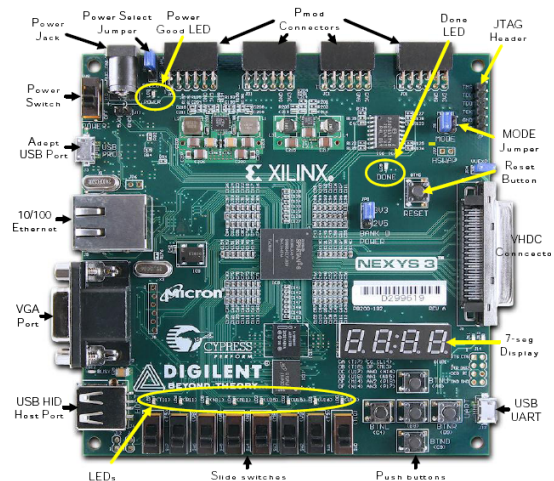


Figura 1: Placa NEXYS 3 con una FPGA Spartan-6.

mente la técnica de enrutamiento LBDR [3]¹, a la que se le dedica gran parte de la teoría. Para esta fase, se utilizan los diversos elementos de la placa NEXYS 3 (leds e interruptores), con los que los alumnos tienen de manera visual e interactiva, una forma de desgranar el funcionamiento del LBDR.

Una vez adquiridos los conocimientos básicos, se utiliza la FPGA para una nueva sesión donde los alumnos pueden asentar y reforzar conocimientos complejos de las redes en el chip. Los alumnos han de implementar en la FPGA una red en el chip completa ya diseñada. La red facilitada a los alumnos presenta algunos parámetros configurables a través de un interfaz gráfico con lo que los alumnos pueden monitorizar la red bajo diferentes condiciones y así extraer conclusiones del comportamiento de la red que afianza los conceptos vistos en teoría. Las limitaciones físicas debido a la poca capacidad de la FPGA impiden que la red sea totalmente configurable. No obstante, el alumno puede configurar el tipo de tráfico inyectado, la tasa de inyección, el tipo de algoritmo de encaminamiento, y la topología de la red. No obstante, para futuros cursos, nuestra intención es utilizar sistemas FPGAs más avanzados que ofrezcan una mayor capacidad y versatilidad.

Para las experiencias prácticas con FPGAs, se han dedicado 3 sesiones. En la primera sesión, se introduce el uso de FPGAs y el lenguaje Verilog. Se realiza una introducción teórica breve sobre la estructura de la FPGA así como a la estructura básica de Verilog. También se realiza un primer diseño, introduciendo los pasos en cada herramienta.

¹LBDR (Logic Based Distributed Routing) es una solución de implementación del algoritmo de encaminamiento sobre una red en chip

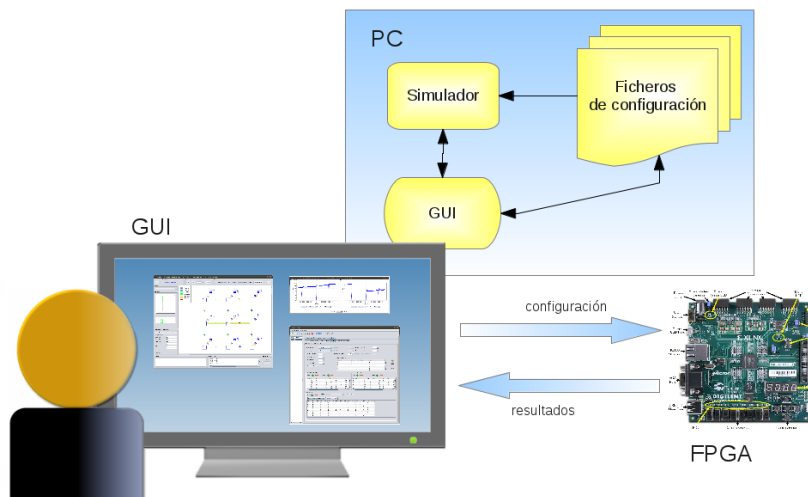


Figura 2: Diagrama de componentes del interfaz gráfico.

En la segunda sesión, se realiza el diseño en Verilog de uno de los bloques de la red en el chip, concretamente del módulo encargado del procesamiento del algoritmo de encaminamiento LBDR en los routers. En esta sesión se persigue un doble objetivo, afianzar los conocimientos sobre el algoritmo LBDR y asentar los conocimientos de Verilog adquiridos en la sesión anterior.

En la tercera, y última sesión, se lleva a cabo la programación de la FPGA con una red facilitada a los alumnos, quienes evaluarán el comportamiento de las diferentes configuraciones mediante la observación y el análisis de los datos obtenidos en la realización de diferentes emulaciones mediante el GUI, variando el tipo de tráfico, encaminamiento y topología de la red.

Dado que el punto de partida de los alumnos difería enormemente, los boletines prácticos son muy detallados para que un alumno experimentado pudiera realizar una práctica totalmente autoguiada.

3. Interfaz Gráfico con la FPGA

El interfaz de usuario es una aplicación de PC desarrollada para simplificar y automatizar la ejecución de las emulaciones con la FPGA, las simulaciones y el procesamiento de los datos obtenidos tanto de la FPGA como del simulador (figura 2), abstrayendo al alumno de la complejidad de los ficheros de configuración y de la comunicación con la FPGA. De este modo se reduce el tiempo que los alumnos dedican a la realización de la práctica por motivos ajenos a los objetivos de aprendizaje del curso.

El interfaz gráfico es un entorno desarrollado en Java por su fácil portabilidad a diferentes sistemas ope-

rativos, aunque actualmente la ejecución se encuentra limitada a entornos Linux/MacOS.

Esta interfaz permite:

- Crear y modificar ficheros de configuración (figura 3)
- Ejecutar simulaciones
- Configurar e iniciar emulaciones en la FPGA (figura 4)
- Visualizar la topología contenida en el fichero de configuración. Esta ventana permite representar la información de los estadísticos recibidos de la FPGA en tiempo real, así como visualizar la información de emulaciones anteriores (figuras 5 y 6)
- Visualizar la evolución de los estadísticos más importantes de la red en chip a lo largo de diferentes emulaciones (latencias, tasa de tráfico admitido, caudal), mediante la representación de los valores promedios de cada emulación como se muestra en la figura 7.

Dado que los conocimientos sobre el funcionamiento del interfaz gráfico² tan sólo son necesarios para el desarrollo de la práctica en que los alumnos configuran la FPGA, en la última sesión, se les facilita una breve introducción al funcionamiento del GUI dentro del boletín y se realiza una demostración con el fin de agilizar el proceso y asentar el conocimiento de los pasos a seguir para lanzar la emulación en la FPGA.

²El interfaz gráfico forma parte de un entorno de simulación más complejo y completo que ofrece un abanico de funcionalidades mayor a los vistos en estas prácticas, en las que se utiliza como medio para agilizar el uso de la FPGA.

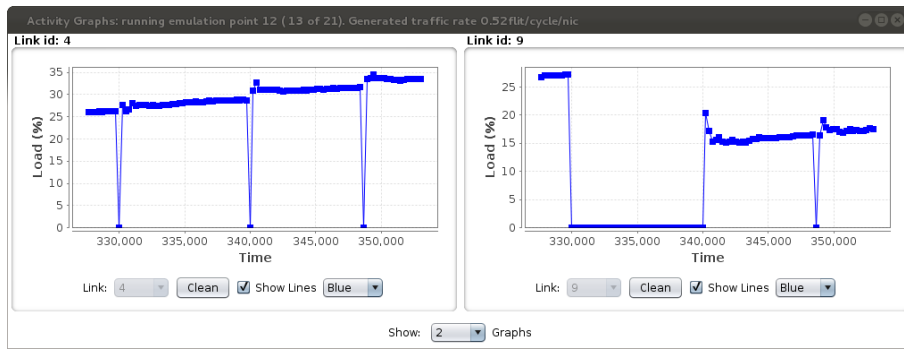


Figura 6: Ventana de detalle de utilización de los canales en tiempo real.

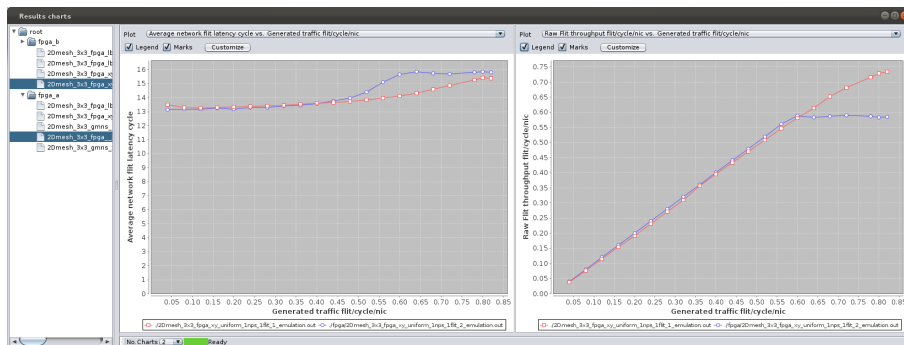


Figura 7: Ventana visualización de estadísticos promedios de diferentes emulaciones.

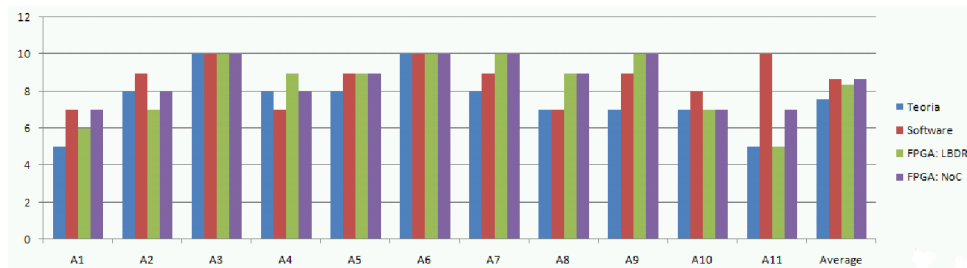


Figura 8: Evaluación de la experiencia con FPGAs.

4. Simulador

gMemNoCsim es un simulador de Redes en Chip que está desarrollado en C/C++ y puede ser utilizado de forma aislada, haciendo uso de los generadores de tráfico internos, o en combinación con otros simuladores de sistema como GEMS/GEM5/SIMICS [5], en cuyo caso el simulador se emplea para modelar la red en el chip y tanto el tráfico como las dependencias vienen determinados por la aplicación externa.

El flujo del programa está conducido por medio de eventos que se suceden en el tiempo. La ejecución de los eventos conlleva la generación de nuevos eventos, lo que finalmente modela la red. Es un simulador con granularidad de ciclo, es decir la unidad en que se mide el coste de las operaciones es un ciclo.

gMemNocsim incluye los diferentes elementos clave de las redes en chip y que están modelados con precisión: topología, tipo de conmutadores, gestión de la memoria, algoritmo de encaminamiento, control de flujo y tráfico. De este modo se permite al alumno poner en práctica y reforzar los conocimientos vistos en la parte teórica mediante la realización de diferentes prácticas, en las cuales se pide al alumno que simule diferentes configuraciones y extraiga conclusiones de los resultados obtenidos. Del mismo modo también se pide al alumno que realice modificaciones al código como puede ser modificar funciones de routers, algoritmos de encaminamiento, . . .

Tras realizar simulaciones y emulaciones, el alumno puede contrastar y validar los resultados con los obtenidos mediante la FPGA.

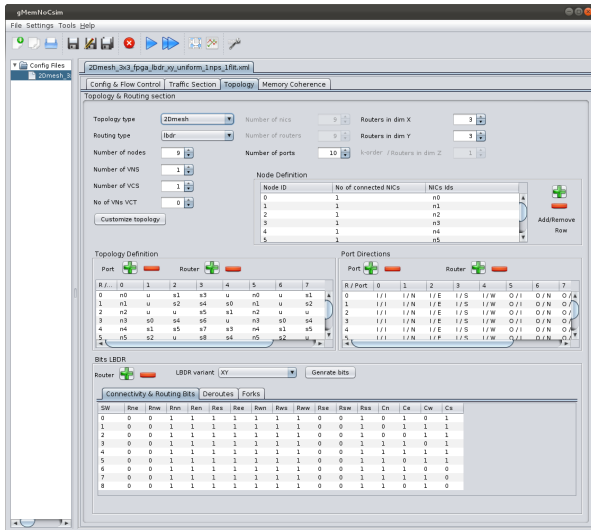


Figura 3: Ventana principal del GUI. Fichero de configuración.

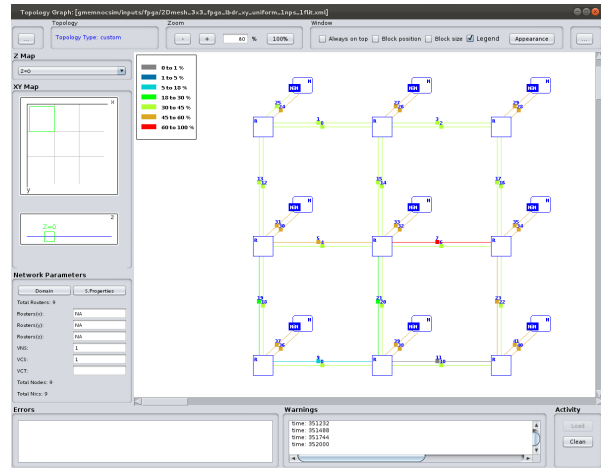


Figura 5: Ventana de visualización de la topología y utilización de los canales en tiempo real.

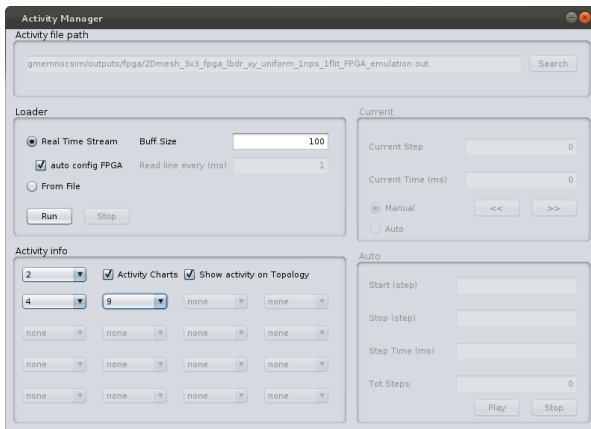


Figura 4: Ventana de control de la FPGA.

5. Evaluación

Para evaluar la experiencia se ha pasado una encuesta anónima para ver el grado de aceptación de los alumnos. Un total de 15 alumnos han realizado la encuesta. La primera parte de la encuesta se pide a los alumnos puntuar – desde 1 a 10 – cuatro aspectos:

- Teoría impartida sobre el uso de las herramientas de diseño (Teoría).
- Experiencia con el uso del software en los equipos del laboratorio (Software).
- Uso de la FPGA para entender y/o afianzar el mecanismo LBDR (FPGA: LBDR).
- Uso de la FPGA como herramienta en el aprendizaje dentro de las redes en el chip (FPGA: NoC).

Como puede verse en la gráfica de la figura 8, el grado de satisfacción ha sido notable, siendo la teo-

ría impartida el aspecto más débil, con una puntuación media de 7.53. En cambio, el uso de la FPGA como medio de aprendizaje de una red en el chip tiene la máxima puntuación con una media de 8.93.

Para mejorar la parte cualitativa de la puntuación, la encuesta tiene una segunda parte donde se pide a los alumnos que describan qué aspectos mejorarían de la experiencia con FPGAs, así como lo más positivo de la experiencia. Respecto a qué aspectos se deberían mejorar encontramos:

- Un 60 % de los alumnos dedicaría más sesiones en definir la red en el chip a través de FPGA.
- Un 50 % propone juntar teoría y diseño con FPGA desde el inicio del curso.

Mientras que otras propuestas muestran la enorme diferencia de conocimientos previos con los que acuden los alumnos:

- Un 13 % propone prácticas menos guiadas, o incluso un poco de más dificultad.
- Un 20 % propone dar más teoría sobre FPGA y lenguajes *hardware* para poder aprovechar mejor la experiencia.
- Un 13 % piden que el material entregado sea más detallado para poder seguir mejor las clases.

Finalmente, se pide a los alumnos que describan qué beneficios tiene introducir las FPGA en el ámbito de las redes en el chip. Los comentarios se pueden resumir en los siguientes:

- Un 60 % reconoce haber mejorado el conocimiento y la apreciación de los conceptos vistos en teoría.
- Un 13 % refleja que no entendía el mecanismo LBDR y que gracias a las prácticas lo ha entendido.

- Un 13 % reconoce que trabajar con FPGAs ha sido lo más útil por encima de otros conocimientos más específicos de la asignatura.

6. Conclusiones

En es artículo detallamos la experiencia práctica en al introducción de la emulación con FPGAs de redes de interconexión en el ámbito de la asignatura Redes en Chip en el máster de Ingeniería de Computadores del departamento DISCA de la Universitat Politècnica de València.

Los resultados obtenidos son altamente satisfactorios ya que se consigue una motivación adicional del alumnado, y un mayor aprovechamiento de las sesiones de teoría.

Referencias

- [1] Giovanni de Micheli and Luca Benini. *Networks on Chips: Technology And Tools*. In Systems on Silicon. Morgan Kaufmann Pub, July 2006.
- [2] Jose Flich and Davide Bertozzi. *Designing Network On-Chip Architectures in the Nanoscale Era*. CRC Press, 2010.
- [3] Jose Flich and Jose Duato. Logic-based distributed routing for nocs. *IEEE Comput. Archit. Lett.*, 7(1):13–16, January 2008.
- [4] Digilent Inc. Nexys 3 reference manual.
- [5] Multifacet GEMS Project University of Wisconsin-Madison. Gems.
- [6] Xilinx. Spartan 6 family product.

Motivación del Estudiante hacia la Arquitectura del Procesador

Crispín Gómez,
Departamento de Sistemas Informáticos
Universidad de Castilla-La Mancha, Albacete
Crispin.Gomez@uclm.es

María E. Gómez y Julio Sahuquillo.
Departamento de Sistemas Informáticos y Computación
Universitat Politècnica de València, Valencia
{megomez, jsahuqui}@disca.upv.es

Resumen

Debido a problemas principalmente de consumo energético y temperatura, prácticamente la totalidad de fabricantes de procesadores actualmente ofrecen productos con al menos cuatro núcleos de procesamiento. Aunque inicialmente algunos de ellos ofrecían ejecución en orden, la mayoría de los núcleos actuales soporta ejecución fuera de orden y muchos de ellos ofrecen soporte para ejecución multithilo para ocultar las altas latencias de memoria. Esta complejidad se acrecienta con las actuales jerarquías de cache, las redes de interconexión dentro del chip y controladores de memoria.

Por este motivo resulta muchas veces muy difícil acercar la docencia al hardware de los procesadores actuales. Sin embargo, estamos obligados a reducir dicha distancia si se desea motivar al alumno en el estudio de esta interesante temática. Este trabajo presenta una metodología diseñada para tal fin que se centra en el diseño de prácticas con orden creciente de dificultad hasta llegar a los detalles de funcionamiento de los procesadores actuales. Esta metodología se ha llevado con éxito con resultados muy notables, consiguiendo publicaciones en congresos de primer nivel y puestos de trabajo en empresas como Intel o Universidades de Estados Unidos.

Palabras clave

Docencia, arquitectura de computadores, motivación del alumno, procesadores multicore, simuladores.

1. Introducción

En la última década la arquitectura de los computadores ha evolucionado de los sistemas monolíticos a los procesadores multicore, con complejas jerarquías de cache y controladores de memoria. Estos dispositivos se encuentran presentes en la mayoría de los dispositivos electrónicos de

uso cotidiano en los diversos aspectos de nuestras vidas. La mayoría de los ciudadanos disponen de teléfonos móviles inteligentes, ordenadores portátiles, televisores programables, etc.

A pesar de esta innegable onnipresencia de procesadores altamente potentes, curiosamente los alumnos de Ingeniería Informática se muestran con frecuencia poco motivados en el estudio de las asignaturas relacionadas con esta temática. En este trabajo se describen posibles razones de esta desmotivación y se presenta una posible solución.

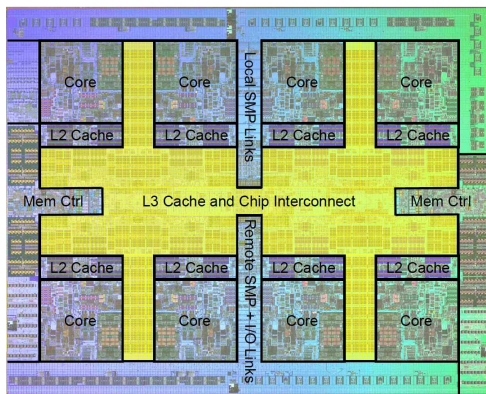
Son muchos los aspectos que influyen en la baja motivación, entre otros, cabe destacar:

- Las opciones de trabajo en temas relacionados con esta materia.
- Distanciamiento entre la teoría y la práctica.
- El peso de la asignatura en la carrera.
- La dificultad de la materia.

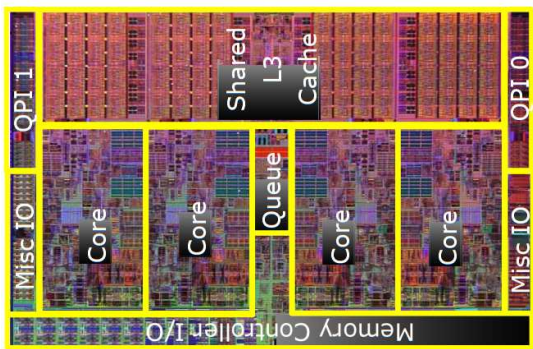
Considerando el entorno de nuestra comunidad en particular, las posibilidades de trabajo son escasas y se restringen casi únicamente, a la integración en un grupo de investigación. Este hecho se relaja un poco más si se considera Barcelona, donde actualmente se encuentran el BSC (Barcelona Supercomputing Center) e Intel Barcelona, que en los últimos años han incorporado a una buena cantidad de doctores. Aún así, esta cantidad es mínima si se compara con el número total de titulados.

El distanciamiento entre la teoría y la práctica se debe principalmente a la gran complejidad y la rápida evolución de los procesadores. Esto causa la inexistencia de simuladores sencillos que modelen las características más complejas y recientes de los procesadores actuales, lo cual lleva a que en la mayoría de las ocasiones no se experimente con los conceptos más complejos en el laboratorio por la falta de simuladores. Esta metodología es un error porque justamente estos conceptos más difíciles de entender son los que deberían ejercitarse en el laboratorio.

A modo de ejemplo, la figura 1 presenta dos procesadores comerciales multinúcleo con capacidad



(a) IBM Power7.



(b) Intel Nehalem.

Figura 1: Procesadores comerciales multinúcleo.

para soportar la ejecución de varios hilos. Estos procesadores incorporan una jerarquía de memoria con tres niveles, ocupando la caché L3 o de último nivel (LLC) aproximadamente la mitad del área del chip. La arquitectura y la complejidad del sistema va aumentando, ofreciendo muchas alternativas de diseño y aumentando las prestaciones alcanzadas. Sin embargo frente a esta realidad, el alumno titulado tiene una visión no solo alejada sino con frecuencia distorsionada de la realidad.

La motivación de muchas asignaturas es intrínseca a la carrera, por ejemplo, casi todos los estudiantes de Ingeniería Informática se encuentran muy motivados con las asignaturas de programación desde el primer día en la universidad porque es más fácil que hayan experimentado con ella, sin embargo, los alumnos desconocen qué es la arquitectura de los computadores hasta haber cursado varios cursos de la carrera. Por dicho motivo, los profesores de esta materia deben hacer un énfasis especial para motivar al alumno sobre la importancia de la materia dentro de la carrera. Esta labor debe ser insistente y enlazada con la realidad social, y el peso de las empresas relacionadas con los procesadores (procesadores, dispositivos empotrados, servidores, memorias, etc) existentes en Europa y no limitarse solo a la localidad donde se encuentra la

Universidad. La motivación no es fácil y se dificulta aún más con las materias consideradas generalmente como las más duras de la carrera, como lo es la que tratamos en este trabajo.

Para tratar de acercar el mundo comercial al pedagógico, en este trabajo se presenta una nueva metodología con distintos grados de dificultad que facilita al alumno de Ingeniería Informática tener un amplio y profundo conocimiento de los procesadores comerciales. Estos conocimientos, sin duda estimulan, el interés del alumno en el estudio de esta materia.

La propuesta se basa en el uso de un simulador detallado ciclo-a-ciclo de procesadores multinúcleo y multihilo de uso difundido entre estudiantes de doctorado e investigadores de todo el mundo así como por empresas de diseño de procesadores. La metodología propone que este entorno de simulación se utilice en distintas asignaturas y dada su complejidad se sugiere que su uso se realice de forma progresiva empezando con un uso muy guiado en las primeras etapas y aumentando el grado de implicación por parte del alumno en trabajos de asignatura o proyectos fin de carrera.

Los autores del trabajo piensan que es una obligación de los profesores mostrar al alumno las posibilidades de trabajo ya no solo en la comunidad, sino también en el resto de España, Europa, Estados Unidos u otros países. De hecho, son muchas las ofertas que llegan a través de la red de excelencia HiPEAC-2. Tras la implantación de la metodología, el interés de los alumnos ha crecido. Muestra de ello es que son muchos los alumnos que han elegido cursar las asignaturas relacionadas. Algunos de ellos han publicado su proyecto fin de carrera en congresos de primer nivel y otros alumnos que se han formado durante años anteriores y que son la base para la propuesta de esta metodología se encuentran actualmente trabajando en empresas internacionales relacionadas con la arquitectura del procesador.

El resto de este trabajo se organiza como sigue. La sección 2 describe como se aborda el estudio de la arquitectura del computador en nuestra Universidad y presenta el simulador utilizado. La sección 3 describe la metodología propuesta. En la sección 4 se presenta un caso de estudio. Finalmente, presentamos algunas conclusiones en la sección 5.

2. Docencia de arquitectura de computadores

2.1. Marco universitario

La enseñanza de la arquitectura del procesador en el plan de estudios de nuestra Universidad se cubre de manera incremental en distintas asignaturas.

El primer contacto, es en segundo curso en la asignatura Estructura de Computadores, donde se estudia un procesador con un pipeline sencillo. Posteriormente en cuarto curso en la asignatura Arquitectura e Ingeniería de Computadores se amplía para dar soporte a la ejecución especulativa. En ambas asignaturas el enfoque es totalmente didáctico. Lo que se pretende con ambas asignaturas es dejar claros los conceptos más básicos aunque sea a costa de simplificaciones que hagan que el procesador explicado difiera bastante de los procesadores actuales. El objetivo de tales simplificaciones es no abrumar al alumno con la complejidad inherente a estos procesadores y no desmotivarles en el estudio de ellos. Los conceptos estudiados de forma simplista en ambas asignaturas sin embargo constituyen la base para que en asignaturas posteriores se profundice mucho más en el funcionamiento de los procesadores realistas proporcionando una formación mucho más amplia.

Una vez que se asientan los conceptos fundamentales de la arquitectura de computadores, los alumnos están en disposición de estudiar temas más avanzados y en la mayoría de las ocasiones también están deseosos de acercarse a los conceptos estudiados al funcionamiento real de los procesadores comerciales. Las asignaturas que permiten saciar esta curiosidad son: Arquitecturas Avanzadas de Procesadores de quinto curso de la Intensificación de Ingeniería de Computadores y ofertada también en el Máster de Ingeniería de Computadores y la asignatura Redes en Chip del Máster en Ingeniería de Computadores. Estas asignaturas proporcionan al alumno una aproximación más realista a los procesadores comerciales recientes, lo cual repercute muy positivamente en la motivación del alumno hacia el estudio de los procesadores reales. Se trata de dos asignaturas complementarias ya que es imposible entender el impacto en las prestaciones de los procesadores comerciales actuales considerando solo una de ellas sin tener en cuenta la otra. Por este motivo, durante el último curso académico y el presente los profesores de ambas asignaturas hemos puesto en marcha un enfoque metodológico conjunto con el objetivo de fomentar el interés del alumno hacia la arquitectura de computadores. La metodología se basa en que los alumnos puedan experimentar en el laboratorio con los conceptos complejos estudiados en las clases teóricas para afianzarlos. Para ello, ambas asignaturas han utilizado un mismo simulador que permite modelar procesadores realistas de forma progresiva. El enfoque tiene dos importantes beneficios: por una parte, el alumno tiene una visión total del sistema entendiendo el impacto de cada componente en las prestaciones; por otra, el alumno solo necesita aprender una única herramienta que va utilizando de forma incremental y muy guiada,

dada la complejidad del simulador, pero que le permite satisfacer sus curiosidades. Otra pieza clave de este nuevo enfoque es la estrecha coordinación de los profesores de ambas asignaturas que de nuevo repercute en la motivación de los estudiantes ya que los alumnos ven la relación entre ambas asignaturas.

2.2. Dificultades para la motivación

Los procesadores comerciales evolucionan constantemente incluyendo cada vez características más complejas para aumentar sus prestaciones. Esto hace que la enseñanza de la arquitectura de computadores tenga que estar en continua actualización para acercarse, en la medida de lo posible, a los productos reales existentes en el mercado, a fin de que los alumnos consigan una formación relativamente actualizada y realista. Esta actualización requiere la adaptación tanto de los contenidos teóricos impartidos en el aula como de los contenidos prácticos ejercitados en el laboratorio.

Centrándonos en los contenidos del laboratorio, por ser la parte más complicada de abordar, éstos se suelen trabajar por medio del uso de distintos simuladores según el curso y los objetivos perseguidos. Entre los simuladores típicos podemos encontrar el DLXV para el estudio del pipeline de un procesador: ruta de datos, ver como avanzan las instrucciones por el pipeline, resolución de riesgos, etc. Se trata de un simulador relativamente sencillo de aprender y utilizar lo que facilita la labor de aprendizaje, además de útil para facilitar la comprensión de procesadores escalares simples que son la base de los procesadores superescalares actuales. Sin embargo el DLXV no puede proporcionar apoyo para la enseñanza de conceptos actuales más avanzados como procesadores monolíticos superescalares o multihilo debido a que no es capaz de modelarlos. Este hecho provoca un distanciamiento importante entre los contenidos teóricos y los prácticos, dejando contenidos teóricos sin ser desarrollados de forma práctica en el laboratorio, lo cual provoca una decepción en los alumnos interesados en experimentar con estos conceptos. La mayoría de las universidades no resuelven este vacío y se deja el uso de simuladores más complejos que permiten modelar estas características, como el SimpleScalar [1] o el Multi2Sim [3], exclusivamente a los estudiantes de doctorado. Este hecho es claramente un error por dos motivos principalmente. Primero porque se pierde el apoyo de dichas herramientas para afianzar y conseguir un conocimiento sólido de conceptos complejos que desarrollados en el laboratorio pueden facilitar su comprensión. Y segundo, pero no menos importante porque se pierde la oportunidad de tener alumnos altamente motivados en profundizar más en el estudio

de la arquitectura de computadores.

La metodología propuesta permite superar la complejidad del simulador por medio de varias etapas que permiten un aprendizaje muy gradual del simulador lo que permite incrementar el interés del alumno hacia el estudio de la arquitectura de computadores. Este interés es fundamental para incentivar al alumno a realizar el proyecto, tesina de master e incluso tesis doctoral en esta materia o trabajar en este campo. En la siguiente sección se describen las características del simulador utilizado para implementar la metodología.

Con respecto a las oportunidades de trabajar en el campo de la arquitectura de computadores, esta es otra de las dificultades que tenemos en nuestro país para tener alumnos altamente motivados hacia la arquitectura de computadores. Es verdad que las posibilidades de encontrar trabajo en esta materia en nuestra país son bajas pero no nulas. Como ejemplo de ello, podemos decir que varios de los alumnos formados en nuestro grupo de investigación están trabajando en los laboratorios de investigación de Intel en Barcelona. Además otros de los alumnos formados en el grupo, que no les ha importado irse a trabajar al extranjero, no han tenido problema alguno en encontrar puestos de trabajo con condiciones muy ventajosas para ellos.

2.3. Simulador usado

El simulador seleccionado debe ser capaz de modelar los procesadores recientes además de permitir, al alumno interesado, trabajar durante todo el tiempo de formación en la universidad (Ingeniero o Grado, Máster y Tesis Doctoral), con una misma herramienta. Por ello, hemos elegido Multi2sim, entre cuyas características destacan la capacidad para modelar procesadores *pipeline* superescalares, arquitecturas multihilo y multinúcleo, unidades de procesamiento gráfico (GPUs), y soporte para los benchmarks más comunes en investigación. Se trata de un simulador de tipo *application-only* que permite ejecutar una o más aplicaciones sin tener que arrancar primero un sistema operativo. En añadidura se trata de un proyecto de código abierto, que se puede descargar de [3].

El paradigma de simulación usado en este simulador se puede dividir en dos módulos: la simulación funcional y la detallada. La funcional es solo una emulación del programa que recibe como entrada, y tiene el mismo comportamiento que tendría dicho programa ejecutado nativamente en una máquina x86. El simulador detallado ofrece un modelo de estructuras de CPU como cachés, unidades funcionales, redes de interconexión, etc., y un *pipeline* con seis etapas (*fetch*, *decode*, *dispatch*, *issue*, *writeback* y *commit*), ofreciendo soporte para ejecución especulativa.

El número de hilos y el número de núcleos es configurable. Respecto al soporte de ejecución multihilo, se pueden obtener diseños multihilo de grano grueso (CGMT), de grano fino (FGMT), y multihilo simultáneo (SMT).

La configuración de la jerarquía de memoria es muy flexible. Se puede usar cualquier número de niveles de caché, con cualquier número de cachés en cada nivel (se mantiene la coherencia mediante el protocolo MOESI). Las cachés pueden ser unificadas o separadas para datos e instrucciones, privadas o compartidas por núcleo, por hilo, o por unidad de cómputo de GPU, y pueden servir rangos específicos de direcciones físicas.

El modelo de red de interconexión entre diferentes niveles de la jerarquía de memoria también es ampliamente configurable. Incluye un conjunto de nodos finales, de switches y de enlaces, que permiten definir la topología de la red; y una tabla de encaminamiento bidimensional que permite definir cualquier algoritmo de encaminamiento.

Por otra parte, se puede obtener un informe detallado de las estadísticas relacionadas con el *pipeline* del procesador, clasificadas por hilo y por núcleo de ejecución. Hay resultados de simulación genéricos, y estadísticas tanto para las estructuras hardware (buffer de reordenación, cola de instrucciones, archivo de registros, buffer de destino de salto), como para cada etapa del *pipeline*.

El informe de estadísticas de jerarquía de memoria contiene una sección por cada caché, módulo de memoria principal y red de interconexión. Para cada módulo de memoria principal o caché, el informe muestra información de accesos, fallos, aciertos, lecturas, escrituras, etc. Así mismo, el informe de la simulación de la red de interconexión está formado por variables como las transferencias, los mensajes, el ancho de banda, la utilización, etc.

El entorno de simulación proporciona herramientas de depuración amigables y de fácil uso. Por ejemplo, el depurador del *pipeline* es una herramienta del paquete del simulador que se usa para ver gráficamente los diagramas de tiempo para simulaciones detalladas, mediante el cual se puede ver la traza ciclo a ciclo de una determinada etapa con una representación de las uops y macroinstrucciones ejecutadas.

Además, desde la versión 3.0, Multi2Sim ha aumentado sus capacidades de simulación con un modelo para las GPUs de AMD, que también cuenta con su propio depurador de *pipeline*. Dicho depurador es una herramienta para analizar la ejecución del código OpenCL en la unidad correspondiente del simulador. Mediante una interfaz gráfica, actúa como una ayuda visual en la prueba del código y de la arquitectura del propio simulador.

En sus últimas versiones, Multi2Sim usa el formato

INI para todos sus archivos de entrada y salida. Un fichero INI es un archivo de texto plano utilizado para almacenar la información de configuración e informes estadísticos de las aplicaciones.

Finalmente, Multi2Sim se ha adaptado para proporcionar las estadísticas que requiere McPAT como archivos de entrada. McPAT es un entorno de modelado que integra consumo, área y tiempo, y que soporta la exploración exhaustiva del espacio de diseño para procesadores de muchos núcleos.

3. Metodología propuesta

El enfoque metodológico aplicado durante el curso anterior y que también estamos aplicando el presente curso persigue aumentar el grado de motivación del alumnado hacia la arquitectura de computadores por medio de la experimentación con mecanismos y políticas de los procesadores actuales de una manera gradual de forma práctica. Para realizar esta experimentación, superando la dificultad que esto supone, se han definido varias etapas de uso del simulador que pretenden acercar el simulador utilizado de forma gradual al alumno. Así la metodología consta de cuatro etapas con un grado de dificultad creciente. Las etapas son: modificación de parámetros de simulación, modificación de partes simples del código del simulador, implementación de funcionalidades completas y autonomía completa.

En la primera etapa al alumno únicamente se le pide que *modifique parámetros de simulación* para analizar cómo afectan éstos a las prestaciones obtenidas. El alumno debe modificar el valor de algunos parámetros clave del sistema estudiados previamente de forma teórica (por ejemplo, el predictor de saltos, el ancho de lanzamiento, o el ancho del enlace de la red), sacar resultados y realizar un análisis de la influencia de los mismos sobre las prestaciones del sistema. Esta primera etapa debe hacerse de manera muy guiada, por ello se propone la realización de la misma en prácticas de laboratorio con la presencia y el apoyo del profesor.

En la segunda etapa se va un poco más allá y se pide al alumno que *modifique partes simples del código*. Una vez el alumno sabe lanzar simulaciones modificando valores de parámetros y es capaz de interpretar los resultados obtenidos, debe estar en disposición de modificar partes del código muy específicas y acotadas como puede ser el predictor de saltos o el mecanismo de prebúsqueda. De nuevo esto debe hacerse de manera muy guiada, por ejemplo, se le puede proporcionar el código correspondiente a un mecanismo de prebúsqueda como base, para que lo analice y lo modifique para implementar otro distinto. Dada la complejidad del simulador, debe prepararse una memoria muy detallada con el fin de facilitar la

labor de comprensión del alumno. Por otra parte, al finalizar esta etapa, el alumno está preparado para enfrentarse a mayores retos con el simulador. Esta etapa se puede llevar a cabo de manera fructífera con trabajos de asignatura. Nuestra experiencia es que esta etapa consigue despertar el interés de los alumnos interesados en la arquitectura de computadores.

Aquellos alumnos que finalicen con éxito la segunda etapa, tendrán probablemente mayores inquietudes y desearán enfrentarse a mayores retos *implementando funcionalidades completas de los procesadores*, ya que esto les permitirá conocer el funcionamiento interno de los procesadores con mucho más detalle. Esta etapa se puede llevar a cabo en forma de Proyectos Fin de Carrera (PFC) o Trabajo Fin de Máster (TFM). Estos trabajos son realmente motivadores para los alumnos porque pueden ver cómo funcionan de manera realista los procesadores. Los resultados alcanzados en los PFCs o en las TFMs por algunos de nuestros alumnos han sido espectaculares y han llegado a publicarse en congresos internacionales de primer nivel como el IPDPS o el Euro-Par. Esta etapa permite despertar el interés de los alumnos por realizar una tesis doctoral en el ámbito de la arquitectura de computadores.

Aquellos alumnos que finalizan las tres etapas anteriores con éxito se encuentran en una posición de privilegio para empezar la tesis doctoral, ya que llevarán al menos un año de ventaja o incluso varios, sobre aquellos que no hayan empezado con esta metodología desde el principio.

4. Caso de estudio

Esta sección se centra en la aplicación de la metodología en las dos primeras etapas. La metodología se ha llevado a cabo durante el anterior curso y el actual con éxito en las asignaturas de Arquitecturas Avanzadas (AAV), asignatura de segundo ciclo y máster, y Redes en Chip (REC), asignatura de máster, tanto en el ciclo de Ingeniería en Informática como en el máster en Ingeniería de Computadores de nuestro departamento, llevando a cabo un acercamiento gradual al simulador Multi2Sim por parte del alumnado.

4.1. Elección del trabajo y etapas de aprendizaje

Es importante que el trabajo a realizar sea sugestivo y motive al alumno a su realización. Con este propósito hemos elegido como sistema base un chip multiprocesador con ocho núcleos conectados mediante una malla. El alumno deberá ir alcanzando etapas de aprendizaje de manera progresiva, y cada

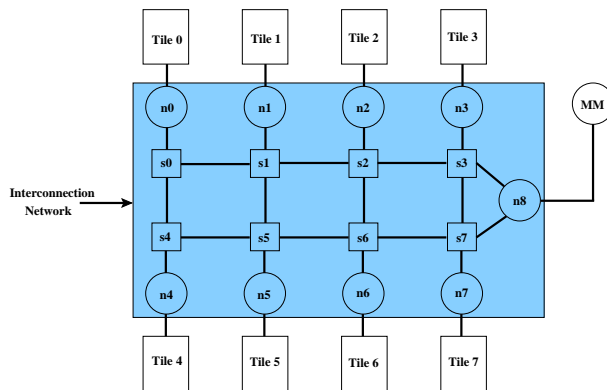


Figura 2: Sistema inicial propuesto.

una de ellas debe motivarle para alcanzar la siguiente. Hemos propuesto las siguientes etapas.

- Modelado del sistema.
- Modelado de las jerarquías de memoria.
- Ejecución de cargas.

4.2. Modelado del sistema base

En cada etapa de aprendizaje, nuestra metodología persigue que el estudiante adquiriera únicamente el conocimiento de las partes del simulador necesario, no es aconsejable hacer un tutorial detallado antes de empezar el trabajo, ya que esto desalentaría casi con toda seguridad al estudiante. Por ello, hemos optado por la metodología “aprender haciendo”.

Para poder llevar a cabo dicha labor de acercamiento, el profesorado ha preparado como primer paso para el estudiante una guía detallada para el uso del simulador. En ella, se explican todas los elementos que es capaz de modelar el simulador: core, cachés (privadas o compartidas) y red de interconexión. Para cada uno de estos elementos, se explica su funcionamiento y papel en el sistema y se enumeran cuáles son los principales parámetros de configuración en el simulador y las consecuencias de cambiarlos. Por ejemplo, en el caso del core se le explica cómo cambiar el predictor de saltos o el ancho de lanzamiento del mismo, en el caso de una caché se explica cómo cambiar el tamaño de bloque o la latencia de acceso, y en el caso de una red de interconexión se le explica cómo cambiar el ancho de banda de los canales o el tamaño de los buffers de los conmutadores. Además, se le explica en el caso de las cachés cómo modelar un sistema de memoria completo, conectando las cachés de los diferentes niveles de la jerarquía de memoria hasta llegar a la memoria principal. De forma similar, se le explica al alumno como definir una red de interconexión dentro del chip que interconecte las diferentes cachés.

Ejemplo 1: Archivo de configuración de la red de interconexión.

```
[ Network.mesh ]
  DefaultInputBufferSize = 128
  DefaultOutputBufferSize = 128
  DefaultBandwidth = 64

;;;;;;;;;;;;;;;;; NODES ;;;;;;;;;;;;;;;;;;
[ Network.mesh.Node.n0 ]
Type = EndNode
...

;;;;;;;;;;;;;;;;; SWITCHES ;;;;;;;;;;;;;;;;;;
[ Network.mesh.Node.s0 ]
Type = Switch
...

;;;;;;;;;;;;;;;;; LINKS ;;;;;;;;;;;;;;;;;;
[ Network.mesh.Link.s0s1 ]
  Source = s0
  Dest = s1
  Type = Bidirectional

[ Network.mesh.Link.s0s2 ]
  Source = s0
  Dest = s2
  Type = Bidirectional

[ Network.mesh.Link.n0s0 ]
  Source = n0
  Dest = s0
  Type = Bidirectional
...

;;;;;;;;;;;;;;;;; ROUTING TABLES ;;;;;;;;;;;;;;;;;;
[ Network.mesh.Routes ]
...
s0.to.n1 = s1
s0.to.n2 = s2
s0.to.n3 = s1
s0.to.n4 = s1
...
```

Una vez explicado cómo llevar a cabo la configuración de todos los elementos del simulador, se le explica al alumno el sistema que debe modelar. Para comenzar hemos elegido un sistema CMP de 8 tiles conectados mediante una malla 2×4 (véase Fig. 2). Se ha elegido este sistema puesto que es una extensión a la arquitectura empleada en los procesadores de varios núcleos que se pueden encontrar en el mercado. Para facilitar que el alumnado vea el efecto de las memorias caché, se ha decidido empezar con un sistema con un único nivel de cachés de 32KB (tamaño de caché de nivel 1 empleado en muchos

procesadores comerciales). Por lo tanto en este caso, cada tile de la Fig. 2 estará compuesto por 1 núcleo de procesamiento y una caché de nivel 1. Posteriormente, se insertará un segundo nivel de cachés tanto privadas, cómo compartidas. A fin de guiar al alumno en este paso, se ha proporcionando al alumno los ficheros de configuración que modelan el sistema inicial.

En el Ejemplo 1 se muestra parte del archivo de configuración de la red de interconexión cuyo nombre es *mesh*. Concretamente, se muestra la configuración de los parámetros globales de la red (anchos de banda y buffers), y la configuración para especificar las conexiones del nodo 0 y el switch 0, así como la tabla de encaminamiento del switch 0.

4.3. Modelado de las jerarquías de memoria

Una vez el alumno se ha familiarizado con los ficheros de configuración del simulador, se le pide que modele el mismo sistema pero añadiendo un segundo nivel de cachés privadas de 512KB.

De esta forma, todas las cachés de nivel 1 se conectarán a su correspondiente caché de nivel 2, y éstas se conectarán mediante la red de interconexión con la memoria principal. En este caso, los tiles de la Fig. 2 estarán compuestos por un núcleo de procesamiento, una caché de nivel 1 y otra caché de nivel 2. Nótese que para llevar a cabo esta ampliación únicamente es necesario modificar el fichero de configuración de la jerarquía de memoria, ya que tanto la red de interconexión, cómo los procesadores son exactamente los mismos. Mediante esta extensión del anterior sistema se persigue que el alumnado consiga una mayor capacidad y confianza en sí mismo a la hora de modificar los parámetros del simulador.

El siguiente paso que se ha fijado en el proceso de aprendizaje del simulador consiste en diseñar caches L2 compartidas cada dos núcleos (0 y 1, 2 y 3, ...). En este caso, al haber la mitad de cachés, se ha duplicado el tamaño (1MB) y aumentado la latencia (de 6 ciclos a 8). Para esta extensión, el alumno no debe modificar únicamente el fichero de configuración del sistema de jerarquía de memoria, sino que además debe modificar el archivo de configuración de la red de interconexión, ya que la red es la encargada de conectar las cachés L2 y la memoria principal y en este caso el número de cachés L2 se ha reducido a la mitad.

4.4. Ejecución de cargas

Es importante que el alumno aprecie como su modelo es capaz de ejecutar correctamente benchmarks que se ejecutan en los procesadores reales y son de uso generalizado por la comunidad

científica. En este sentido hemos optado por el suite de benchmarks paralelos SPLASH2 [2].

Para facilitar el trabajo, la guía de la prácticas proporciona los scripts de configuración para poder ejecutar las cargas. Los alumnos deben obtener y analizar de prestaciones de los tres sistemas modelados midiendo tres índices: tiempo de ejecución, latencia media de la red y número de accesos a la red (de coherencia o acceso a MP). De esta forma el alumno relaciona el tiempo de ejecución con el número de accesos a la red (otras L2 o memoria principal), así como el tiempo que tardan en resolverse los mismos (latencia).

En este estudio, los alumnos observan que las prestaciones dependen tanto de la jerarquía de memoria, como de la carga ejecutada. Para ver las prestaciones de una forma simple se les indica que obtengan gráficas similares a las presentadas en la Figura 3, que muestra los índices de prestaciones normalizados respecto al sistema con L2 privadas. Gracias a esta gráfica, los alumnos serán capaces de ver que en la mayoría de los casos tener un segundo nivel de cachés beneficia mucho las prestaciones del sistema puesto que se reduce la penalización de fallo de L1.

Por otra parte, los alumnos también serán capaces de advertir que una caché compartida, cómo la del último sistema modelado, ofrece mejores beneficios que la privada. Este efecto es debido a que las cargas paralelas comparten código y datos a lo largo de todos los núcleos, por lo que si las cachés son compartidas se reducirá el tráfico de coherencia entre las diferentes cachés. Además, si un bloque compartido es accedido para lectura por dos núcleos que comparten caché, esta caché tendrá una única copia del bloque, mientras que en el caso de cachés privadas cada caché tendría su copia del bloque. El alumno puede comprobar que algunas cargas reducen el tiempo de ejecución al tener una caché compartida pese a aumentar la latencia media de acceso, puesto que se reduce de forma significativa el número de accesos a la red, por ejemplo en la aplicación FFT.

Finalmente, para aquellos alumnos que se hayan sentido atraídos por la arquitectura de computadores se ha dispuesto de una serie de Proyectos de Final de Carrera o Trabajos de Final de Grado que aprovechan enormemente el conocimiento adquirido con el simulador por parte del alumnado llegado a este punto. Por ejemplo, se ha propuesto y llevado a cabo una implementación y estudio sobre el simulador de la jerarquía de cache, de los mecanismos de prebúsqueda en memoria más comúnmente usados en los procesadores a lo largo de la historia o de diferentes mecanismos de gestión del controlado de memoria.

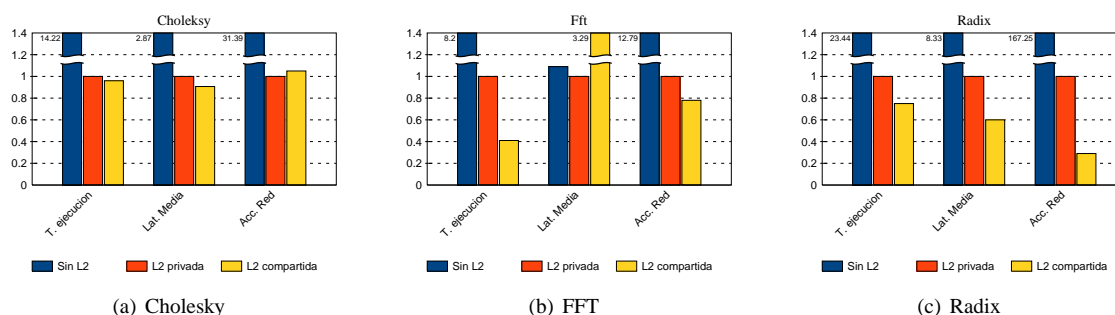


Figura 3: Prestaciones normalizadas respecto al sistema con L2 privadas.

4.5. Resultados

Para evaluar de forma objetiva la puesta en marcha de esta nueva metodología, se ha preparado un pequeño cuestionario de 25 preguntas sobre diferentes aspectos de la misma, como son: i) el uso de una única herramienta entre varias asignaturas, ii) el detalle de los aspectos estudiados, iii) la ocultación de detalles no relacionados y iv) motivación hacia la arquitectura de procesadores. El cuestionario fue completado de forma voluntaria por 22 estudiantes en la última sesión de prácticas, y las respuestas se valoraban de 0 a 100, siendo a 100 la máxima puntuación. La puntuación media obtenida ha sido de más de 70 puntos, lo que nos indica que la propuesta está en el buen camino pero que aún hay que refinar ciertos aspectos.

Por otra parte, este año hemos experimentado un importante aumento en el número de estudiantes en ambas asignaturas, aumentando de únicamente 4 hasta 22 alumnos actuales. Creemos que la presente metodología y los contenidos usados son la causa principal de este despertar en el interés de los alumnos hacia la arquitectura de los computadores.

5. Conclusiones

En este artículo se ha presentado la metodología puesta en marcha durante el curso anterior y el actual en nuestra universidad para la enseñanza práctica de la Arquitectura de Computadores con un enfoque realista y actualizado. La metodología se basa en el uso de un simulador muy completo que permite modelar los aspectos más interesantes de los procesadores actuales. Debido a la complejidad del simulador, éste se utiliza en distintas asignaturas y en ellas el alumno va adquiriendo de forma progresiva un conocimiento y manejo más profundos del simulador, lo que le permite llegar a un aprendizaje realista de los procesadores

actuales. Los resultados nos han demostrado que el alumno en un tiempo razonable es capaz de manejarse con el simulador y obtener resultados muy interesantes que han permitido ser publicados en congresos de relevancia internacional.

Por otra parte, desde el punto de vista del profesor, una vez diseñada la metodología y montado el entorno, hemos construido una plataforma de futuro, que cambiando el tipo de prebúsqueda, topología de red, etc. se pueden obtener una gran cantidad de trabajos para el estudio de procesadores realistas, muchos de los cuáles llegan realmente a motivar tanto al alumno como al profesor.

Agradecimientos

Este trabajo ha sido apoyado por el MINECO bajo las claves TIN2012-38341-C04-01 y TIN2012-38341-C04-04, y por el Programa de Apoyo a la Investigación y Desarrollo (PAID-05-12) de la Universitat Politècnica de València bajo la clave SP20120748.

Referencias

- [1] Todd Austin, Eric Larson and Dan Ernst *SimpleScalar: An Infrastructure for Computer System Modeling*. IEEE Computer vol. 35(2), 2002.
- [2] S. Woo, M. Ohara, E. Torrie, J. Singh and A. Gupta. *The Splash-2 programs: Characterization and methodological considerations*. In 22th International Symposium on Computer Architecture (ISCA), pages 24–36, 1995.
- [3] The Multi2Sim Simulation Framework Website, <http://www.multi2sim.org>

Barcelona ZeroG Challenge: Experimentación en gravedad cero en la clase de Informática

Antoni Pérez-Poch

Escola Universitària d'Enginyeria Tècnica Industrial de Barcelona
(EUETIB)

Consorci Escola Industrial de Barcelona
Universitat Politècnica de Catalunya (UPC)
Departament Llenguatges i Sistemes Informàtics
antoni.perez-poch at upc.edu

Resumen

El concurso internacional Barcelona ZeroG Challenge, organizado por la UPC Barcelona Tech ofrece a estudiantes de Grado, Máster o Doctorado la oportunidad de diseñar su propio experimento y volar con él en gravedad cero en vuelo acrobático. La plataforma de microgravedad de la UPC, pionera en Europa en realizar vuelos parabólicos de gravedad cero con aviones acrobáticos de una sola hélice, está abierta desde 2009 a investigadores y estudiantes para realizar este tipo de experimentos mediante un convenio con el Aeroclub Barcelona-Sabadell y el clúster "BAIE Barcelona Aeronáutica y Espacio". Hasta la fecha se han realizado hasta la fecha dos ediciones de este concurso educacional para estudiantes con notable éxito, y actualmente nos encontramos en la tercera edición. Con esta infraestructura, se consiguen hasta diez segundos de gravedad cero durante la maniobra parabólica del avión, pudiendo repetirse la maniobra hasta 15 veces en un solo vuelo. El coste total de una campaña resulta ser de unas 1000 veces inferior respecto del coste de una campaña convencional con avión parabólico reactor, y el tiempo de acceso a la plataforma mucho más rápida. La experiencia docente consiste en aplicar datos de estas investigaciones a una clase de primer curso de Informática, en un grupo piloto de la EUETIB de docencia en inglés. El objetivo de la experiencia es motivar e introducir a los alumnos en las tareas de investigación que se realizan en nuestra universidad. Para ello, se muestran vídeos y presentaciones de los experimentos realizados en una primera parte de la sesión. En la segunda parte, se propone a los alumnos la realización de ejercicios algorítmicos para procesar los datos obtenidos en el experimento. Entre otras, se obtiene el perfil de gravedad reducida que se obtiene en la maniobra del avión. Se muestran datos de una breve encuesta obtenida al final de la actividad. La actividad resulta ser altamente motivadora para el alumna-

do, y les permite entrar en contacto con una infraestructura de investigación única en España.

Abstract

The international contest 'Barcelona ZeroG Challenge' organized by UPC BarcelonaTech offers to undergraduate, master and PhD students the outstanding opportunity of designing its own experiment and fly with it in zero gravity in aerobatic flight. The UPC Microgravity platform, pioneer in Europe providing microgravity access in parabolic flights with single-engine planes is open since 2009 to researchers and students. Its capabilities cover experimentation and technological demonstrations and it is possible thanks to a joint venture between UPC BarcelonaTech, Aeroclub Barcelona-Sabadell and BAIE, the aerospace cluster in the Barcelona area. Two student editions of the Barcelona ZeroG campaign have already successfully taken place, now looking forward to the third edition. With this research infrastructure, it is achieved zero gravity parabola of up to 10 seconds duration, a manoeuvre that can be repeated up to 15 times at each flight. The overall cost of such a campaign is more than 1000 times less than that of a conventional campaign, and the time-to-access to the platform is significantly smaller. Our teaching experience consists of applying real data from a zeroG research campaign to the syllabus of a Computer Engineering fundamental course. The experience was performed in a pilot group at EUETIB (Barcelona College of Industrial Engineering) where classes are given entirely in English. The objective of this experience is to motivate the students, and immerse them to the actual research being done at our University. Videos and presentations of the experiments actually flying are presented in a first part. Afterwards, the students are given an exercise in which they should design an implement algorithms to process the data. Data of a final survey filled by the

students in the group are finally presented. In conclusion, this is an original way to effectively motivate students to study computer engineering in the initial phase of their studies.

Palabras clave

Motivación, informática básica, gravedad cero, investigación.

Keywords

Motivation, computer fundamentals, zero gravity, research.

1. Introducción

El concurso ‘Reto Barcelona Gravedad Cero para Vuelos Acrobáticos’, realizado en el Centre de Recerca en Aeronàutica i Espai (CRAE) de la Universitat Politècnica de Catalunya. BarcelonaTech (UPC), tiene por objetivo que estudiantado universitario de grado y de posgrado diseñe y construya un experimento relacionado con la gravedad cero en un periodo de tiempo limitado. El proyecto nació el 2009 partir de un convenio de colaboración entre la UPC, el Aeroclub Barcelona-Sabadell y el Cluster Aeroespacial de Catalunya, BAIE, mediante el cual se ha trabajado conjuntamente en el desarrollo de esta Plataforma, que coordina el profesor Pérez-Poch, y que permite a estudiantes y grupos de investigación de todo el mundo poder desarrollar experimentos en condiciones de microgravedad sin tener que salir al espacio, y a costes razonables [1, 2, 3, 4].

Desde el Aeroclub Barcelona-Sabadell se ponen a disposición de los investigadores los medios aéreos y los pilotos experimentados en las maniobras acrobáticas que permiten la obtención de microgravedad, así como la asistencia técnica necesaria para el acoplamiento del experimento en la aeronave y los procedimientos de seguridad. Desde BAIE, se ofrece el establecimiento de sinergias con la industria local y asesoramiento en el proceso dada la experiencia previa de la Asociación en la campaña de investigación en microgravedad de la Agencia Espacial Europea (ESA). No es necesario salir al espacio exterior para la obtención de condiciones de microgravedad. Éstas se pueden conseguir con cualquier sistema de vuelo capaz de generar condiciones de caída libre (es decir, precipitación al vacío haciendo que sólo actúe la fuerza de la gravedad sobre el cuerpo).

Así, por ejemplo, cualquier objeto en órbita se encuentra en estas condiciones, pero también podemos recrearlas sin entrar en órbita. Por ejemplo, con cohetes que realizan vuelos suborbitales (no entran en órbita) con una trayectoria en caída libre, se pueden obtener varios minutos de microgravedad; con aviones de tipo comercial especialmente modificados para

tal fin se pueden obtener hasta 25s (como el Airbus A300, que es el que utiliza la Agencia Espacial Europea, o el KC-135, utilizado por la NASA, o el Ilyushin-76 de la Agencia Espacial Rusa); asimismo, con las llamadas “torres de caída libre” desde donde el experimento simplemente se deja caer por el tubo interior de una torre de aproximadamente unos 150m de altura, se pueden obtener entre 5 y 10s de ingravidez.

Pero estas condiciones, y esta es la novedad, se pueden obtener también con avionetas acrobáticas más pequeñas que los aviones de tipo comercial, que realizan la misma maniobra (llamada “vuelo parabólico”, por la forma que hace la trayectoria de la aeronave durante el vuelo: una parábola), y con las cuales podemos obtener hasta 8s de gravedad cero, aproximadamente del mismo orden que en el caso de las torres de caída libre.

Este último método no se había utilizado hasta ahora en Europa con finalidades de investigación, y la Plataforma creada en Barcelona por la UPC, el Aeroclub de Sabadell y BAIE, lidera su establecimiento con una primera campaña de vuelos que se emprendió en el año 2010.

Cuanto más segundos de microgravedad sean los requeridos y más compleja sea la aeronave, más coste se añade al experimento. Los aviones acrobáticos son de uso habitual y son aeronaves pensadas para la operación de esta maniobra de forma que no hay que realizar ninguna modificación a la aeronave. Esto abarata sensiblemente los costes. De hecho, el coste de mantenimiento y operatividad de la Plataforma se ha estimado mil veces inferior al de los vuelos parabólicos convencionales. La nueva plataforma, por lo tanto, permite ampliar el abanico de sistemas disponibles para la obtención de ingravidez, con lo cual los investigadores disponen de mayor grado de libertad, escogiendo una u otra plataforma en función del tiempo de ingravidez mantenido requerido y del presupuesto disponible.

La plataforma se presentó a la comunidad científica internacional por primera vez en agosto del 2010 durante la celebración de un Seminario Aeroespacial en la Universidad Internacional del Espacio (ISU), en Francia. El éxito fue inminente puesto que varios grupos respondieron la convocatoria del concurso, lanzada bajo el nombre ‘Reto Barcelona Gravedad Cero para Vuelos Acrobáticos’.

El proyecto ganador de la primera edición fue ‘Figuras de Percepción Reversible en Vuelo Parabólico’, del equipo de la Universidad Internacional del Espacio, liderado por el francés Gilles Clément, experto en investigación en microgravedad, y formado por los canadienses Alexandra Kindrat, Heather Allaway y Alexander Melynshtyn, el británico Jagruti Pankhania y el francés Jonathan Muller. La campaña de vuelos se realizó el 31 de octubre. Esta plataforma sirvió

como campo de pruebas para que el mismo experimento se pudiera realizar con éxito durante el año 2012 en la Estación Espacial Internacional por astronautas profesionales. En el año 2011 se realizó una segunda convocatoria del premio internacional, y actualmente se ha convocado una tercera dado el éxito obtenido.

El Aeropuerto de Sabadell es ahora —gracias al acuerdo entre la UPC, el Aeroclub y BAIE— una plataforma de experimentación en microgravedad abierta a los científicos europeos de la Asociación Europea de Investigadores de Gravedad Reducida (ELGRA) y a estudiantes universitarios. La investigación también ha constatado que los efectos a largo plazo de no acondicionamiento del sistema cardiovascular a la microgravedad —arritmias, debilitación de la musculatura del corazón, disminución de la resistencia vascular, desmayos y dificultad para caminar, entre otras— son prácticamente los mismos en condiciones de gravedad marciana (un tercio inferior que en la Tierra).

La experimentación en este campo y las tecnologías que se desarrollan se han revelado esenciales para comprender procesos físicos, químicos o fisiológicos. Un ejemplo es el descubrimiento de un mecanismo fundamental en el control de las células productoras de glóbulos rojos, a raíz del estudio de la anemia que suelen sufrir los astronautas. [4, 5, 6].

2. Experiencia docente en Informática

2.1. Contexto Académico

La experiencia docente se basó en la integración de algunos resultados obtenidos en experimentos profesionales realizados en campañas de investigación previas de vuelos parabólicos en el Aeropuerto de Sabadell. Se obtuvieron series de datos que comprendían el valor de la gravedad en los tres ejes, obtenida en intervalos de 10ms, y los valores de tensión arterial, impedancia eléctrica, y pulso arterial de un sujeto sometido a estas variaciones de gravedad. Se suministraron los datos de hasta tres parábolas con una duración media de 6,7 segundos.

El contexto académico en que se realizó la experiencia docente fue con alumnos de un grupo de la fase inicial (primer curso) de Grado en Ingeniería en la EUETIB de la UPC. El grupo consta de 30 alumnos, siendo éste un grupo piloto en la UPC donde sus integrantes tienen una nota de entrada en la universidad mínima de 11,5 sobre 14 puntos; y la docencia se realiza íntegramente en inglés.

Dentro de la asignatura Informática, cuyos objetivos corresponden a los de un curso básico de Fundamentos de Informática de 6 créditos es habitual proponer ejercicios de diseño algorítmico. Los ejerci-

cios son en su mayoría académicos teniendo poca conexión con la experimentación, y menos aún con datos procedentes de experimentos reales realizados en la propia universidad o fuera.

2.2. Ejercicios de programación

El ejercicio que se describe en este artículo se propuso a los alumnos en clase de forma integrada con su currículum. Se dedicó media hora a realizar un seminario teórico con videos y demostración de la plataforma de investigación. En la siguiente media hora se realizó un taller práctico de diseño de experimentos para la convocatoria del concurso internacional *Barcelona ZeroG Challenge*. De este taller se obtuvieron ocho propuestas de experimento inicial con su hipótesis científica, justificación, detalle del experimento y diseño, así como el análisis de riesgos de seguridad asociados.

En la segunda hora se propuso un ejercicio de diseño algorítmico basado en los datos que se obtienen en un experimento previamente realizado por nuestro grupo de investigación del CRAE-UPC. El ejercicio está diseñado como una posible práctica de programación de la asignatura, a la vez que tiene el formato del tipo de ejercicios que se encontrarán en el examen final. Los alumnos realizaron en pareja la actividad como si de una actividad más de programación de la clase de problemas se tratara. Además, se ofreció la oportunidad que pueden aprovechar las clases de laboratorio para programarlo. Para ello, existe entonces la ventaja de poder utilizar un fichero con datos reales obtenidos de la investigación en vuelo parabólico, que los alumnos deben de extraer y procesar.

3. Resultados obtenidos

Al final de la experiencia docente se suministró a los alumnos una encuesta estandarizada SEEQ [7] para conocer el grado de satisfacción de los alumnos del grupo. En esta encuesta además, se ofrece un espacio en blanco para comentarios y sugerencias libres y anónimas. La encuesta comprende 37 ítems de valoración de la experiencia docente del alumno agrupados en 10 apartados: calidad del aprendizaje, entusiasmo, organización, interacción con el grupo, actitud personal, contenido, exámenes, carga de trabajo y dificultad, visión general, calificación global. Las encuestas fueron respondidas por 24 de los 28 alumnos matriculados (87%) en el grupo objeto de esta experiencia.

La experiencia fue calificada con una media de 3,8 sobre 5. En los comentarios se resaltó por parte de cuatro alumnos que era interesante realizar “por fin, algoritmos que tienen un significado real” o que “las clases son interesantes”. Otros comentarios de los alumnos se refirieron a la carga de trabajo: “nos ha dado mucho trabajo, y estamos a final de cuatrimes-

tre”, “tenemos más trabajo que en el cuatrimestre anterior”, “no se valora suficiente para la cualificación final” o también “habría que hacer más ejercicios parecidos a los del examen”.

La media de cada uno de los apartados es superior a 3 sobre 5, destacando el apartado de “interacción con el grupo” con un 4,2.

En el momento de entregar la comunicación no se conoce el rendimiento académico final de los alumnos, pero se dará a conocer y se discutirá éste en relación con esta experiencia, en la presentación del taller de Jenui.

Debemos destacar que el grupo de alumnos fue un grupo reducido (30) en fase inicial, ya que se trata de un grupo piloto de enseñanza en inglés en los Grados de Ingeniería ofertados en la EUETIB de la UPC. Estos alumnos cumplían dos requisitos de entrada: un nivel de inglés inicial medio (B2) y una nota de acceso a la universidad elevada (11,5/14). Es muy probable que el buen nivel de los alumnos haya influido en su interés por esta experiencia. Hacen falta más estudios para evaluar el impacto de experiencias como ésta en grupos más numerosos y con notas de entrada más bajas.

4. Conclusiones

Se ha presentado una experiencia docente original cuyo objetivo ha sido la de acercar la investigación que se realiza en la universidad además de incrementar la motivación de los alumnos. En las asignaturas de fase inicial, la motivación en general es un factor crucial para garantizar que, ante las dificultades, muchos alumnos desconecten. A este problema se suma que el carácter fundamental de una asignatura como Informática Básica donde es difícil a veces elaborar un discurso docente fluido y motivador.

La experiencia ha dado resultados preliminares positivos. Sería temprano analizar si esta primera inclusión en el currículum de Informática de esta experiencia es extrapolable a otras asignaturas y otros contextos docentes. Sin embargo, el alto interés de los alumnos por la iniciativa se ha puesto de manifiesto con el número de propuestas de experimentos recibidas y la buena satisfacción que han mostrado en las encuestas.

Finalmente, las actividades para inspirar a los alumnos a dedicarse en el futuro a la investigación científica creemos que son muy importantes, y deberían ser incluidas como currículum transversal en la docencia obligatoria de Grado. Corremos el peligro que, limitados por la obligación de dar los temarios, olvidemos fomentar la creatividad científica entre el alumnado universitario.

Agradecimientos

Debemos agradecer en primer lugar a los alumnos del grupo piloto en inglés de la fase inicial de la EUETIB, curso académico 2012-13, cuatrimestre de primavera por su buena disposición e interés para someterse a la experiencia y rellenar los cuestionarios de la encuesta.

El personal del Aeropuerto de Sabadell especialmente el grupo de vuelo acrobático, así como el piloto Daniel Ventura González, y la ex-directora de BAIE, Gloria García Cuadrado merecen también nuestro agradecimiento.

Referencias

- [1] Messerschmid E, Bertrand R, Space Stations. Systems and Utilization. pp 300-310. Springer Verlag. ISBN 3-540-65464-X Berlin 1999.
- [2] Pletser V, Short duration microgravity experiments in physical and life sciences during parabolic flights: the first 30 ESA campaigns. *Acta Astronautica*, 55(10) 829-854. 2004.
- [3] Pérez-Poch A., González, D.V. “Aerobatic flight: an innovative access to microgravity from a centennial sport”. 58th International Astronautical Conference, Glasgow, 2008. Conference Paper # IAC-08-A2.3-12
- [4] Pérez-Poch A. "On the role of numerical simulations in studies of reduced gravity-induced physiological effects in humans. Results from NELME.". Proceedings of the 38th COSPAR General Assembly, Bremen,
- [5] BAIE Announcement of Opportunity for European researchers/students: Recuperado el 8 de junio de 2013: <http://www.bcnaerospace.org/public/new.php?id=51> ,
- [6] CRAE-UPC Announcement of Opportunity for European researchers and students. Recuperado el 8 de junio de 2013: <http://recerca.upc.edu/crae/news/acrobatic-flight-an-innovative-access-to-microgravity-from-a-centennial-sport-announcement-of-opportunity-for-upc-students-researchers>
- [7] SEEQ: Students Evaluation of Educational Quality. Institut de Ciències de l'Educació UPC. Recuperado el 8 de junio de 2013 de: http://www.upc.edu/ice/innovacio-docent/eines_i_recursos/eines-upc/enquesta-de-satisfaccio-seeq/

Dirección estratégica de (~~las tecnologías de la información~~) la asignatura

Faraón Llorens Largo
Departamento de Ciencia de la Computación e Inteligencia Artificial
Universidad de Alicante
Faraon.Llorens@ua.es

Resumen

El curso 2012-2013 se ha puesto en marcha la asignatura *Dirección estratégica de las tecnologías de la información* perteneciente al Máster Universitario en Ingeniería Informática de la Escuela Politécnica Superior de la Universidad de Alicante. Basándonos en el principio *predicar con el ejemplo*, se utilizan las técnicas y los principios de la dirección estratégica y se aplican al diseño y la ejecución de la propuesta docente de la asignatura. Además, en su diseño se han utilizado una diversidad de principios y herramientas de innovación docente. Así, los objetivos de la asignatura se han simplificado y transformado en 7 *mantras* (pensamientos), es decir, en siete principios que los estudiantes deben interiorizar y que se repiten al principio de cada clase. Para la evaluación se utilizan distintos instrumentos (cuestionario, mapa conceptual, trabajo práctico, participación en los talleres, exposición de los trabajos, ...) a modo de cóctel, que permita medir las distintas dimensiones del aprendizaje. Algunas de las propuestas innovadoras, que tienen el doble objetivo de ser un instrumento de la asignatura pero al mismo tiempo que recogen aspectos de dirección y de mejora continua, son el *SLA de aprendizaje*, las *actas de aprendizaje*, la *espiral de conocimiento*, el *análisis de tendencias TI* y el *proyecto personal creativo*.

Abstract

During the 2012-2013 academic year, the subject *Strategic Governance of the Information Technologies* has been launched. It belongs to the Computer Engineering Master of the Polytechnic School of the University of Alicante. Based upon the principle *practise what you preach*, we propose the use of techniques and principles of strategic management to be applied to the design and implementation of the subject. In addition, a variety of principles and tools of teaching innovation have been used. Thus, the objectives of the course have been simplified and transformed in 7 *mantras* (thoughts), that is, in seven

principles that students must internalize and repeat before the beginning of each class. The evaluation is performed through different tools (questionnaire, conceptual map, practical work, participation in workshops, presentation of their works...) as a cocktail, to measure the different dimensions of learning. Some of the innovative proposals, which have the dual purpose of being a tool of the subject but also including aspects of leadership and continuous improvement, are: *learning SLA* ("service level agreement"), *learning minutes*, *knowledge spiral*, *analysis of IT trends* and *creative personal project*.

Palabras clave

Dirección estratégica, mejora continua, SLA de aprendizaje, acta de aprendizaje, espiral de conocimiento, tendencias TI, creatividad, innovación.

1. Predicar con el ejemplo

El presente curso 2012-2013 se ha puesto en marcha en la Universidad de Alicante la asignatura *Dirección estratégica de las tecnologías de la información* dentro del Máster Universitario en Ingeniería Informática de la Escuela Politécnica Superior. En su diseño e impartición y basándose en el principio *predicar con el ejemplo*, se han utilizado técnicas y los principios de la dirección estratégica. Además, también se han utilizado los principios y las herramientas de innovación docente por los que he abogado durante estos años en que he ocupado el cargo de Vicerrector de Tecnología e Innovación Educativa de la Universidad de Alicante.

Un primer objetivo perseguido con esta iniciativa ha sido facilitar las cosas al estudiante, utilizando un lenguaje sencillo, claro y visual. Las acciones encaminadas en este sentido son descritas en detalle en el apartado 2, que se titula *la maldición del conocimiento*. Para presentar la estructura de la asignatura y los distintos contenidos tratados en la misma se han utilizado distintos mapas visuales. Los objetivos de la asignatura se han simplificado y transformado en 7

mantras (pensamientos), es decir, en siete principios que los estudiantes debían interiorizar y que se repetían al principio de cada clase. Para la evaluación se han utilizado distintos instrumentos (cuestionario, mapa conceptual, trabajo práctico, participación en los talleres, exposición de los trabajos, ...) a modo de cóctel, que permita medir distintas dimensiones del aprendizaje.

El segundo objetivo ha sido utilizar herramientas útiles en la dirección, adaptadas a la realidad de las aulas. Algunas de las propuestas innovadoras, que tienen el doble objetivo de ser un instrumento de la asignatura pero al mismo tiempo que recogen aspectos de dirección y de mejora continua, son el SLA de aprendizaje, las actas de aprendizaje, la espiral de conocimiento, el análisis de tendencias TI y el proyecto personal creativo.

El artículo se cierra con un apartado dedicado a reflexionar y presentar las lecciones aprendidas con la experiencia y las posibilidades de transferencia de esta a otras asignaturas.

2. La maldición del conocimiento

Como se ha dicho uno de los objetivos perseguidos ha sido “hablar para que te entiendan”, no para que te admiren, aunque no comprendan lo que estás diciendo. Un buen profesor debe hacer frente y superar la *maldición del conocimiento*, por la cual cuanto más sabes acerca de algún tema, más difícil es para ti imaginarte lo que significa no tener ese conocimiento y por tanto peor te explicas. El término es muy elocuente e ilustrativo. Es muy difícil ponerse dentro de la mente de la audiencia y por tanto conectar con ella para transmitir correctamente el mensaje: unas instrucciones de trabajo, la presentación de un proyecto o la explicación de una clase.

Vamos a centrar la reflexión en el papel del profesor como transmisor de conocimiento. La maldición del conocimiento es muy habitual en nuestro trabajo, ya que los profesores universitarios somos expertos en nuestros campos (algunos incluso “los más expertos del mundo”). Y además, la mayoría de la veces estamos explicando conceptos complejos y especializados, por lo que utilizamos términos abstractos y jerga profesional. Hay que saber ponerse en la piel del receptor del mensaje (el estudiante), si es que queremos que el mensaje le llegue. Lo que a una persona le parece sencillo y obvio, no tiene porque serlo para todos. Hay que conocer el grado previo de conocimiento de nuestros estudiantes, para así bajar hasta el nivel en que se encuentran y acompañarlos en el proceso de aprendizaje, en el proceso de elevación del nivel. Los profesores debemos ser capaces de recordar como era nuestra “cabeza” cuando no teníamos esos conocimientos e intentar partir de esa posición para poder explicar lo que ahora conocemos a la

perfección. Veamos algunas propuestas para hacer más fácil a los estudiantes entender nuestro mensaje y por tanto transmitir de una manera más efectiva los conocimientos que tenemos sobre nuestra asignatura.

2.1. Mapas de la asignatura

Según el diccionario de la RAE, un mapa es la “representación geográfica de la Tierra o parte de ella en una superficie plana”. Los mapas se crearon con el propósito de conocer el mundo que se iba explorando, integrando la información que aportaban los distintos viajeros y la que los estudiosos deducían. Su representación gráfica debe ser clara, sin sacrificar por ello la precisión y permitiendo en un plano de lectura más profundo, obtener elementos informativos más detallados. Estos son los principios que nos han movido a elaborar distintos mapas de la asignatura. La visualización es importante debido a su fuerza expresiva y nos sirve para alcanzar cierta comprensión acerca del contenido. Recurriremos a la representación gráfica para organizar de forma espacial los conceptos básicos del dominio o ámbito del conocimiento científico de la asignatura.

En un primer momento visualizaremos la organización temporal y la estructura de la asignatura. La asignatura se compone de cuatro bloques: conceptos básicos, elaboración de un plan estratégico, talleres sobre habilidades directivas y prospectiva tecnológica. La organización temporal se ha visualizado en una hoja de cálculo que se actualizaba semanalmente.

Las disciplinas cognitivas del ámbito pedagógico, emplean representaciones gráficas para describir la organización del conocimiento, dándoles el nombre de *mapas conceptuales*. Los mapas conceptuales hacen su aparición en el ámbito de la didáctica de las disciplinas científicas de la mano de Novak, que considera estos mapas como auténticas *herramientas metacognitivas*, que permiten visualizar la articulación de los conceptos y expresar los elementos conocidos acerca de un tópico para, a partir de ahí, desarrollar estrategias de profundización. Cuando las palabras seleccionadas para representar los conceptos y las relaciones se escogen cuidadosamente, los mapas conceptuales pueden ser herramientas muy útiles para observar matices de significado, ayudado a los estudiantes a organizar sus pensamientos y resumir áreas de estudio. Con estos mapas conceptuales podemos cubrir un doble objetivo:

- Por una parte, el profesor al construirlo entiende mejor las interrelaciones entre los distintos conceptos y le permite visualizar e identificar los aspectos clave, de forma que puede organizar mejor la asignatura. Durante el proceso de construcción del mapa conceptual se estimula la reflexión y nos ayudará a conectar el contenido con la práctica pedagógica.

- Por otra parte, el alumno se verá beneficiado ya que el disponer de una representación gráfica de la materia le ayudará a comprender mejor la asignatura, identificando fácilmente los conceptos clave y las interrelaciones entre los mismos. Al mismo tiempo dispondrá de un mapa que le permitirá situarse en cualquier momento a lo largo del curso.

Este curso se ha optado por no hacer público el mapa conceptual de la asignatura hecho por el profesor y así pedir a cada estudiante que elabore su propio mapa conceptual de la asignatura, formando parte de la evaluación de la misma, como veremos en el apartado 2.3.

2.2. Mantras

Según el diccionario de la RAE, un *mantra*, palabra que significa literalmente pensamiento, es “en el hinduismo y en el budismo, sílabas, palabras o frases sagradas, generalmente en sánscrito, que se recitan durante el culto para invocar a la divinidad o como apoyo de la meditación”. En aras a la claridad, hemos convertido los objetivos en siete mantras, es decir, en siete principios que los estudiantes deben repetirse a lo largo del curso y que deben interiorizar, a modo de un “mantra” que se recita todos los días antes de empezar las clases de la asignatura.

Veamos algunos de los objetivos que aparecen en las fichas ANECA de la asignatura. Está es una competencia específica relacionada con la asignatura:

CE: Capacidad para la planificación estratégica, elaboración, dirección, coordinación, y gestión técnica y económica en los ámbitos de la ingeniería informática relacionados, entre otros, con: sistemas, aplicaciones, servicios, redes, infraestructuras, o instalaciones informáticas y centros o factorías de desarrollo de software, respetando el adecuado cumplimiento de los criterios de calidad y medioambientales y en entornos de trabajo multidisciplinares.

Y este sería un objetivo específico de la asignatura tal como aparece en la ficha de la asignatura:

Adquirir una visión global de cómo se deben dirigir las TI de una organización, comprendiendo los problemas, desafíos y la importancia cada vez mayor de las TI en el desarrollo de estrategias, la ejecución y la gestión, con el objetivo de mejorar el rendimiento global y la rentabilidad de una organización.

Y, finalmente, estos serían los siete (más uno) mantras que se proponen para la asignatura:

0. A partir de este momento ya no soy un estudiante de ingeniería informática sino el director de TI (CIO) de una organización para la que trabajo.
1. Ya tengo suficientes conocimientos técnicos, así que ahora mi objetivo es prepararme en habilidades directivas y conocer mi organización, para

hacer que las TI representen un valor que permita que mi organización tenga una ventaja competitiva.

2. Como CIO debo salir del territorio conocido y cómodo del departamento TI y debo aliarme con las áreas funcionales para crear valor en la organización.
3. Debo dirigir y planificar estratégicamente las TI de acuerdo a los objetivos perseguidos por la organización, y por tanto, la estrategia de TI debe estar alineada con la estrategia de la organización y al servicio de esta.
4. Las TI son una realidad y una necesidad inexcusable, como instrumento de cambio y modernización, como posibilitadoras de hacer las cosas de otra forma, por lo que tienen un carácter estratégico y horizontal y por tanto deberían formar parte de la planificación global de la organización.
5. Las principales responsabilidades relacionadas con la gobernanza de las TI deben recaer y ser apoyadas directamente por la más alta dirección.
6. Las TI son una herramienta fundamental para la dirección y planificación estratégica de las organizaciones y el marco de la gestión de la información ya no es tan sólo una estructura de apoyo accesoria sino que es la base esencial del rendimiento corporativo.
7. La información es la clave no solamente para gestionar e interpretar el presente, sino sobre todo para construir el futuro del negocio.

Este lenguaje y este estilo creemos que es más cercano al estudiante y que le permite interiorizar mejor los aspectos clave de la asignatura. También permite ser repetido al inicio de cada sesión, lo que asegura su aprendizaje.

2.3. Evaluación coherente

La clave de cualquier propuesta docente (y más si ésta es innovadora) es la evaluación ya que dada la trascendencia que tiene para el estudiante la calificación, como determinante de su acceso al título, éste termina por sustituir los objetivos fijados en el proceso educativo por los conocimientos y conductas que le permitan superar las asignaturas. Y el aspecto más destacado de la evaluación es la coherencia de los criterios de evaluación con los elementos restantes de la planificación docente (competencias, objetivos, contenidos y actividades). Así, la evaluación es parte integrante del planteamiento docente, desde el principio hasta el final.

Podemos definir la *evaluación* como aquel proceso complejo que comprende la obtención, por medio de los más variados procedimientos, de información útil acerca de cualquier tema, que permitirá emitir juicios, y en consecuencia, tomar decisiones al respecto. Así, la diversidad en la evaluación, no solo es necesaria

por definición, sino que enriquece el proceso de enseñanza+aprendizaje. Se trata de concebir y utilizar la evaluación como un instrumento y recurso de aprendizaje que permita suministrar retroalimentación adecuada a los alumnos, y al propio profesor, contribuyendo a la mejora continua. Se ha planteado la siguiente distribución y las siguientes herramientas de evaluación:

- Los conceptos fundamentales se evalúan de forma individual a través de un cuestionario y de la elaboración y presentación del mapa conceptual. Esto tendrá un peso en la nota global del 15%.
- El trabajo práctico se evalúa en grupo por medio del documento del Plan Estratégico elaborado a lo largo del curso. Tiene un peso en la nota global del 40%.
- Las habilidades directivas se evalúan a partir de la participación en los talleres. Tiene un peso de un 20% de la nota global.
- La parte de prospectiva tecnológica se evalúa a partir de la asistencia a las charlas organizadas y la presentación de la tendencia TI por parte de cada estudiante (ver el apartado 3.4). Tiene un peso del 10% de la nota global.
- Finalmente el proyecto personal creativo (ver el apartado 3.5) se evalúa con la presentación del mismo y tiene un peso del 15%.

Como aspecto innovador, el cuestionario sobre los conceptos fundamentales ha sido titulado *la dirección estratégica de las TI en diez tuits*. El último día de clase, a la hora del examen, se les pidió que en 30 minutos cada alumno tenía que escribir diez tuits que globalmente recojan los principales conceptos vistos en la asignatura y reflejen las ideas claves de la misma. Los principios perseguidos con esta evaluación han sido:

- **Transparencia:** los tuits se pueden ver buscando la etiqueta #DETI2013.
- **Claridad:** diez mensajes (titulares) que se os han quedado tras cursar la asignatura.
- **Concisión:** los tuits tienen un máximo de 140 caracteres, lo que ha obligado a hacer un ejercicio de síntesis.
- **Creatividad:** aunque se han propuesto una serie de etiquetas (#gobiernoTI, #tendenciasTI, ...) era un ejercicio de respuesta abierta.
- **Tecnología:** han utilizado, para un uso que no era para el que se diseñó, una herramienta tecnológica de última generación que está revolucionando el modo de hacer las cosas, de forma que les sirve también de ejercicio de prospectiva tecnológica.

Algunos de los tuits que han escrito los estudiantes son:

#DETI2013 @FaraonLlorens Como #DirectorIT solo externalizaras los servicios que no formen parte de tu core.

#DETI2013 @FaraonLlorens En el #PlanEstrategico escucha todas las voces posibles, los afectados por él deben de hacerlo suyo.

#DETI2013 @FaraonLlorens El Plan Estratégico como guía contra la improvisación.

#DETI2013 Existen numerosas certificaciones TI, muy bien valoradas por las empresa, que pueden diferenciar tu curriculum.

#DETI2013 #ISO38500 9.Debatimos sobre esta norma, que es un sistema que dirige y controla la utilización de las TI actuales y futuras

Esta sería la evaluación del estudiante. Además en la última sesión se ha evaluado la asignatura y el profesor por medio de una encuesta de satisfacción. Y a lo largo del curso, al finalizar una actividad (un bloque conjunto) se debatía brevemente sobre lo positivo y lo negativo de la misma. Un ejercicio de evaluación de la asignatura muy interesante ha sido cruzar los siete mantras con los diez tuits de cada alumno, de forma que ha permitido ver qué mensajes han calado más en los estudiantes.

3. La dirección estratégica

En este apartado vamos a detallar algunas de las propuestas innovadoras, que tienen el doble objetivo de ser un instrumento de la asignatura pero que al mismo tiempo recogen aspectos de dirección y de mejora continua de un proceso (planificar, ejecutar, evaluar y replanificar), contenidos clave de la asignatura.

3.1. SLA de aprendizaje

Los SLA (“Service Level Agreement”) son acuerdos habituales en la gestión de servicios TI. Fijar claramente los SLA es clave para la firma de un contrato de externalización (*outsourcing*), aspecto que un Director TI debe contemplar como una opción ya que no todo se puede hacer con recursos propios. Un acuerdo de nivel de servicio es un contrato escrito entre el proveedor de un servicio y el cliente con objeto de fijar el nivel de calidad acordado. Es por tanto una herramienta que ayuda a ambas partes a llegar a un consenso en términos del nivel de calidad del servicio. Suelen recoger aspectos como tiempo de respuesta, disponibilidad horaria, documentación disponible, personal asignado al servicio, etc. Al mismo tiempo constituye una referencia para la mejora continua de los servicios prestados, ya que tener definidos los indicadores y el nivel de servicio deseado facilita el seguimiento.

Adaptándolo a la labor docente, podemos definir el *acuerdo de nivel de servicio de aprendizaje* como un contrato escrito entre el profesor y cada estudiante

con el objetivo de fijar el nivel acordado de servicio para la calidad de la actividad docente. Es una herramienta que ayudará a consensuar, en términos de calidad del servicio, distintos aspectos de la tarea docente, al mismo tiempo que constituye un punto de referencia para la mejora continua del proceso. Los objetivos que se perseguían con la elaboración del SLA de aprendizaje eran:

- Firmar un contrato pedagógico entre el profesor y el estudiante.
- Posibilidad de personalizar este contrato pedagógico (aprendizaje personalizado).
- Aprender qué es un SLA: en qué consiste, qué partes tiene, ...

Las fichas de las asignaturas sirven para que el alumno conozca las reglas del juego antes de matricularse en la misma. Dando una vuelta de tuerca podemos establecer contratos pedagógicos con nuestros alumnos, de forma que conjuntamente el estudiante y el profesor elaboran y firman dicho contrato pedagógico personalizado que establece de forma más detallada estas reglas del juego. Dando una vuelta más, podemos convertir ese contrato pedagógico en un SLA que establezca además de las reglas del juego, el nivel del servicio que nos comprometemos a prestar.

3.2. Acta de aprendizaje

Las actas de las sesiones de trabajo son documentos escritos que recogen los temas tratados y los acuerdos esenciales, con la finalidad de certificar y dar validez a lo acordado, por lo que son un buen instrumento de dirección. Además, las actas son necesarias, aunque no suficientes, para lograr que las reuniones sean productivas. Es evidente que toda reunión tiene que quedar registrada de algún modo, y el acta es el resumen que nos permitirá, entre otras cosas, comprobar los asistentes reales y los que excusaron su asistencia, registrar las decisiones tomadas en la reunión, determinar el plan de actuación y plantear una nueva reunión. El acta nos permite una visión general de la reunión y como no deben ser demasiado extensas, deben centrarse en las cuestiones importantes.

Si lo adaptamos a la labor docente, podemos definir el *acta de aprendizaje* como el documento escrito, consensuado, que resume lo tratado en las sesiones de clase: asistentes, tareas realizadas, tareas pendientes, acuerdos tomados, documentos utilizados, etc. Es enviada por correo electrónico al finalizar cada sesión y aprobada al inicio de la siguiente sesión. Pretende ser una guía de lo que se va haciendo y de lo que tenemos que hacer para la siguiente sesión.

3.3. Espiral de conocimiento

Un aspecto que deben conocer como futuros directivos de TI son la gestión del conocimiento y de la

innovación. El conocimiento puede ser tácito, cuando únicamente la persona lo conoce, o explícito, cuando puede ser transmitido a otras personas y codificado. La espiral del conocimiento pretende pasar de lo tácito (o implícito) a lo explícito, de lo individual a lo colectivo. La manera de avanzar en el conocimiento es crear espirales de conocimiento, de forma que el conocimiento aumenta al crearlo, explicitarlo, hacerlo accesible para que sea utilizado y así crear nuevo conocimiento apoyándonos en este.

En el ámbito educativo, la difusión en abierto (por ejemplo en un blog) de todo el material de la asignatura y del trabajo realizado por los estudiantes nos permite entrar en estas espirales de conocimiento. En lugar de reproducir todos los años el mismo curso y pedir los mismos trabajos, cada nuevo curso los estudiantes utilizaran los trabajos de los años anteriores para añadir una pequeña mejora y ponerlo a disposición de los estudiantes de los siguientes años.

3.4. Análisis de tendencias TI

Otra cualidad deseable en un directivo de TI es la capacidad de análisis de las tendencias tecnológicas. Cuando empieza (o acaba) el año es habitual la aparición en las revistas especializadas de artículos tipo “las 10 tendencias tecnológicas para el 2013”. Saber detectar las verdaderas tendencias diferenciándolas de las meras modas pasajeras es una cualidad deseable para el directivo de TI. La prospectiva tecnológica se encarga de analizar el estado actual y las perspectivas de progreso científico y tecnológico para identificar aquellas áreas estratégicas de investigación y las tecnologías emergentes en las que concentrar los esfuerzos de inversión para obtener los mayores beneficios para su organización.

El objetivo de este módulo es analizar, desde el punto de vista estratégico, no técnico, qué aportan a la organización las tendencias TI de ese momento. Para ello, cada alumno buscará, seleccionará, presentará y defenderá una tendencia, en una sesión conjunta organizada según el modelo *pecha-kucha* (presentación automática de 20 diapositivas de 20 segundos cada una). En la vida profesional, posiblemente no dispongan de más tiempo (unos escasos 7 minutos) para convencer a la alta dirección de que es interesante explorar las posibilidades de incorporar una nueva tecnología en su organización. El resto de compañeros actuarán de críticos de la tendencia, poniéndose en el lugar de los directivos y usuarios de la organización.

3.5. Proyecto personal creativo

Google utiliza, como técnica de motivación e incentivar de la creatividad, una política que denomina *Innovation Time Off*, mediante la cual a sus ingenieros se les anima a dedicar un 20% de su tiempo a proyectos de su interés. La mitad de los nuevos

productos lanzados por esta compañía (Gmail, Google News, Orkut, o AdSense) surgieron de esta iniciativa. 3M lo llamaba *la hora del contrabando* ("permitted bootlegging"), permitiendo que parte de la jornada laboral de sus investigadores se dedicará a nuevas ideas especulativas, con el único requisito de que compartieran sus ideas con sus colegas. Fruto de esta iniciativa surgió el uso de los "Post-it".

Siguiendo los principios de la "hora del contrabando" de 3M o el "Innovation Time Off" de Google, los estudiantes dedicarán un 15% del tiempo (unas 22 horas aproximadamente) a un proyecto personal y creativo, de exploración de nuevas ideas, que serán compartidas con sus compañeros, debiendo ser presentado y justificado en público en la recta final del curso. En la exposición deberán convencernos de lo que han hecho, independientemente de que hayan tenido éxito o no, y no se evaluará los resultados sino el trabajo realizado en ese tiempo.

4. Conclusiones

La idea clave que pretendemos expresar en esta ponencia es que los profesores podemos, mejor dicho debemos, utilizar las técnicas y herramientas del ámbito de nuestra asignatura en el propio diseño y puesta en práctica de nuestra labor docente. *Predicando con el ejemplo* sería la frase que mejor expresaría lo que estamos argumentando. Visualmente, la imagen más elocuente de este principio la tenemos en el grabado "Drawing hands" de Maurits Escher, en el que se muestra un par de manos, cada una de ellas siendo dibujada y dibujando al mismo tiempo a la otra. De esta forma, al mismo tiempo que el profesor

les está diciendo a los estudiantes lo que deben hacer, lo está demostrando en su forma de actuar.

Cuando estaba diseñando la asignatura "Dirección estratégica de las TI", buscando las estrategias y herramientas para una dirección eficaz y eficiente y pensando en las competencias de un buen directivo, inmediatamente me di cuenta de que algunas de estas herramientas también nos podían servir en nuestro trabajo diario como profesor. Al fin y al cabo, un buen profesor debe dirigir y conducir a sus estudiantes por una ruta, previamente trazada (programación docente), que les conduzca a una destino en el que estén mejor preparados en la temática de la asignatura.

Este principio es aplicable a cualquier asignatura, más allá de la materia concreta a impartir. Hace ya muchos años que en el ámbito de la enseñanza de segundas lenguas se sabe que la mejor manera de aprender un idioma es mediante la integración de lengua y contenidos. Lo llaman "content-based instruction" (CBI) y lo califican como una aproximación pedagógica nueva y motivadora, tanto del profesorado como de los alumnos, en el aprendizaje de lenguas.

El objetivo de la ponencia es hacer reflexionar al profesor, de forma que piense de qué manera lo que está intentando comunicar a sus estudiante, fundamentalmente si son habilidades, puede ser utilizado por el mismo, dado que muchas veces nuestros estudiantes se quedan más con lo que hacemos que con lo que decimos.

Evaluación de la competencia transversal de comunicación oral y escrita mediante la realización de debates

Juan Carlos Amengual

Dep. Lenguajes y Sistemas Informáticos Dep. de Ingeniería y Ciencia de los Computadores

Universitat Jaume I, 12071 Castellón

{jcamen, merche.marques}@uji.es

Mercedes Marqués

Resumen

Uno de los principales problemas de la evaluación de la competencia transversal de comunicación oral y escrita en asignaturas técnicas reside en la dificultad de planear actividades que permitan la evaluación de todos los aspectos involucrados en la misma: capacidad de síntesis, resumen de ideas, elaboración de argumentos, correcta ortografía y sintaxis, correcta expresión escrita, adecuada capacidad de comunicación oral, lenguaje gestual, dominio de la materia y control del tiempo de exposición. Muy a menudo, el profesorado se limita a evaluar parcialmente estos aspectos como parte de la realización de una actividad que permite evaluar una competencia técnica. Esto es más evidente aún cuando el número de alumnos por grupo es elevado, lo que ocurre muchas veces en asignaturas de primer curso. En este artículo, planteamos la posibilidad de organizar debates entre grupos de alumnos para evaluar una determinada competencia técnica y, al mismo tiempo, poder evaluar esta competencia transversal.

Abstract

One of the main drawbacks when assessing oral and written communication skills in engineering subjects lies on the difficulty of setting activities that enable every aspect of the evaluation to be carried out satisfactorily: Synthesis ability, adequate summary of concepts, developing arguments, correct spelling and syntax, appropriate oral communication, body language, subject mastery and presentation time control. Frequently, teachers tend to assess only part of these aspects as students perform an assignment where a technical skill is exercised. This becomes even more evident when the number of students in the class is high, which typically happens among first course subjects. In this paper, we propose debates between students as a basis for assessing a specific technical skill while, at the same time, teachers are able to assess almost every skill involved in oral and writing communication.

Palabras clave

Evaluación sumativa, evaluación formativa, trabajo en equipo, estrategias de comunicación, autoevaluación, evaluación entre pares, rúbricas.

1. Motivación

La competencia transversal centrada en las capacidades de comunicación oral y escrita está presente, de una u otra manera, en todos los planes de estudio de los grados técnicos —ingenierías y grados de ciencias experimentales— que se ofrecen en el mapa de titulaciones del estado español (véase el RD 1393/2007, de 29 de octubre, por el que se establece la ordenación de las enseñanzas universitarias oficiales). Si bien es una competencia que se suele tener muy en consideración en los grados del ámbito de las ciencias humanas, sociales, jurídicas y económicas, no suele ser el caso de los grados técnicos y experimentales.

Los estudiantes de grados de ingeniería no suelen estar motivados a la hora de cuidar la redacción, sintaxis y ortografía de las memorias que suelen acompañar la entrega de trabajos técnicos. Como suele resultar difícil la resolución del trabajo técnico, tienden a descuidar el estilo de redacción, el empleo de construcciones gramaticales adecuadas o la propia ortografía del documento que lo acompaña o describe —y no nos estamos refiriendo al uso de los correctores ortográficos automáticos, que sí que suelen emplearse, sino a la necesaria y cuidadosa revisión que requiere la entrega de cualquier documento formal.

En la misma línea, las exposiciones que el alumnado realiza de sus trabajos suelen carecer del enfoque correcto —las diapositivas escogidas no son adecuadas, ya que o bien sintetizan en exceso o bien resultan excesivamente verbosas— y del ritmo y actitud necesarios —el alumnado se limita a repetir meramente lo que ya se dice en las diapositivas y adopta una actitud corporal y entonación pasivas.

Creemos que, inconscientemente, esta actitud se

contagia al profesorado, el cual se limita a corregir la ortografía y la sintaxis de una memoria, o bien a evaluar la claridad expositiva de una presentación o a analizar el contenido de un vídeo elaborado por los alumnos para transmitir una idea [1]. La tendencia general, en el entorno de evaluación sumativa en el que mayoritariamente nos movemos, consiste en asignar un porcentaje de la puntuación de la actividad —normalmente marginal— para evaluar estos aspectos. Porcentajes significativos de puntuación asignados a evaluar aspectos asociados a las habilidades comunicativas, tales como correcta ortografía, sintaxis adecuada, claridad expositiva, buena organización de conceptos e ideas, lenguaje corporal activo... suelen acarrear conflictos con el alumnado que no entiende que se penalice una buena solución técnica de este modo.

Con el objetivo de abordar este problema, se han propuesto algunas ideas en la bibliografía [1, 3, 4, 5], siendo tal vez el modelo de presentaciones controladas conocido como *Pecha Kucha*¹ la más interesante. La estrategia principal que subyace en casi todas las propuestas consiste en convertir el acto de comunicación en la *forma de entrega* de la actividad. De esta manera, pasamos de centrarnos exclusivamente en el desarrollo de la competencia técnica —lo que hace que la memoria o presentación que acompaña la entrega de la actividad sea algo complementario o secundario— a convertir el acto de comunicación en protagonista de la presentación de los resultados de la actividad que demuestran el logro de la competencia técnica.

Con este cambio de enfoque se consigue que el alumnado se centre en el acto de comunicación como parte del desarrollo de la actividad y, consecuentemente, como parte importante en el logro de la competencia técnica asociada. El profesorado, ahora, se centra en el acto de comunicación como una parte más de la evaluación de la competencia técnica. Si la comunicación es deficiente influirá, sin duda, en la calificación obtenida. Una buena comunicación servirá para que el profesorado detecte errores en el proceso de aprendizaje y pueda, al mismo tiempo que evalúa, proporcionar la retroalimentación adecuada.

En esta línea, el modelo *Pecha Kucha* resulta muy interesante pues permite, al mismo tiempo que se evalúa una competencia técnica, evaluar los siguientes aspectos inherentes a todo proceso de comunicación oral y escrita: capacidad de síntesis, organización de conceptos, correcta expresión escrita, capacidad de comunicación oral, lenguaje gestual, dominio de la materia y planificación del tiempo de exposición.

No obstante, este modelo, al estar centrado en la exposición audiovisual, prima los aspectos relacionados con la presentación propiamente dicha sobre otros aspectos, tal como el empleo de la sintaxis adecuada.

¹<http://www.pechakucha.org/>.

Además, un aspecto fundamental a la hora de medir el nivel de la capacidad de comunicación de una persona, como es el *desarrollo de argumentos*, no queda reflejado en actividades presentadas mediante esta técnica.

La idea que proponemos en este trabajo sigue centrada en el *acto de comunicación como medio para la entrega* de (los resultados) de la actividad. Básicamente, nuestra propuesta consiste en plantear un escenario donde competirán dos alternativas contrapuestas. Cada grupo de trabajo ha de elaborar un documento donde resumirá brevemente (una página) los “pros” y los “contras” de cada postura. En un segundo apartado, igualmente en una página como máximo, cada grupo elaborará un esquema o hilo argumental donde se estructurará la “defensa” de cada una de las dos posturas basándose en el resumen previo de pros y contras. Para concluir este documento, se puede proponer (opcionalmente, en función de la competencia técnica a evaluar) la elaboración de una tercera página en la que se planteen sendos escenarios donde se impone, respectivamente, cada una de las posturas planteadas. Finalmente, los grupos de trabajo se enfrentarán en un debate² donde se sorteará la postura que cada grupo debe defender —es decir, el grupo no sabe de antemano qué papel le tocará adoptar, por lo que debe preparar con igual esmero ambas soluciones propuestas.

En las secciones siguientes desglosaremos con mayor detalle cada uno de los puntos que conforman nuestra propuesta, explicando la motivación y los aspectos de la competencia de comunicación oral y escrita que se pueden evaluar en cada punto. Para finalizar el artículo, expondremos los principales inconvenientes que vemos para el desarrollo de actividades basadas en nuestra propuesta, fundamentalmente centrados en el elevado trabajo de corrección que debe realizar el profesorado, así como una propuesta de evaluación basada en estrategias de autoevaluación y co-evaluación que permitiría superar muchos de los inconvenientes planteados para poder llevarla a cabo.

1.1. Experiencia previa y ejemplos

El esquema propuesto se puso en práctica en una de nuestras asignaturas —una asignatura de formación básica de primer curso del grado en ingeniería informática, para más señas— en el curso 2010/11. La experiencia fue interesante para el alumnado, ya que, por un lado, aprendieron una rutina que les permitía no sólo analizar sistemáticamente un problema sino también poner en orden sus ideas para la elaboración de un hilo argumental y, por otro, aprendieron a plantear un escenario en el que debían “vencer” dialécticamente al adversario y convencer a la audiencia.

²http://csdf-fcde.ca/UserFiles/File/resources/teacher_debate_guide.pdf.

Actividad 1

- Tipo de asignatura: Formación básica.
 Dirigida a: Estudiantes de primer curso.
 Competencia técnica: Desglosar los componentes de un equipo informático en función de su adecuación a una tarea dada.
 Planteamiento general: Se plantea un problema de configuración de un equipo informático sencillo en función de la idoneidad de sus componentes para cumplir una tarea determinada y a un precio razonable.
 Ejemplo: Plantear dos configuraciones diferentes de un ordenador personal destinado a realizar tareas de servidor web y FTP con una carga de trabajo de nivel medio. Se trataría de mostrar la idoneidad de cada configuración y su relación calidad/precio frente a la otra alternativa.

Actividad 2

- Tipo de asignatura: Estructuras de datos o programación avanzada.
 Dirigida a: Estudiantes de segundo curso.
 Competencia técnica: Elección de la estructura de datos adecuada para resolver un problema.
 Planteamiento general: Se plantea un problema de programación o diseño de un algoritmo, en el cual las alternativas a debatir se centran en la elección entre dos posibles estructuras de datos.
 Ejemplo: Reparto de escaños en una circunscripción electoral mediante la denominada Ley de Hondt empleando un diccionario de prioridad o un *heap*.

Actividad 3

- Tipo de asignatura: Algoritmia o similar.
 Dirigida a: Estudiantes de tercer curso.
 Competencia técnica: Elección del algoritmo más eficiente para resolver un problema según las características de los datos de entrada.
 Planteamiento general: Se plantea un problema clásico que puede resolverse mediante dos algoritmos diferentes cuya eficiencia depende de las características de los datos de entrada.
 Ejemplo: Calcular el mínimo número de kilómetros de vía ferroviaria que se necesita adaptar para comunicar una serie de ciudades mediante trenes de alta velocidad empleando el algoritmo de cálculo del MST de Kruskal o de Prim. Se proporcionan dos posibles grupos de ciudades mediante sendos grafos (uno disperso y otro denso).

Cuadro 1: Ejemplos de posibles actividades que podrían plantearse siguiendo el esquema propuesto en diferentes cursos de un grado en ingeniería informática.

Desgraciadamente, la carga de trabajo que supuso para el profesorado el excesivo trabajo de corrección y supervisión de la actividad hizo que esta propuesta se descartara en estos dos últimos cursos. Sin embargo, a la luz de las recientes estrategias de autoevaluación y co-evaluación sugeridas para la evaluación continuada [7], hemos decidido retomar la propuesta, explicando en este artículo cómo podría plantearse reduciendo la carga de trabajo del profesor y potenciando la parte formativa y de auto-reflexión para el alumnado.

En el Cuadro 1 esbozamos tres ejemplos de posibles actividades que podrían plantearse con el esquema propuesto con el fin de ilustrar nuestro discurso.

2. Análisis de las alternativas

En primer lugar, el grupo de trabajo tiene que elaborar la respuesta a la actividad vista como ejercicio de la competencia técnica en sí. Al mismo tiempo que se efectúa la actividad, se prepara una lista con los “pros” y los “contras” de cada una de las dos posibles soluciones propuestas que se van a enfrentar en el debate.

Por ejemplo, si consideramos la *Actividad 3* que se plantea en el Cuadro 1, el grupo de trabajo debería implementar ambos algoritmos —el de Kruskal y una versión de Prim—, preparar una rutina para leer los datos de entrada en el formato descrito por el profesorado y proporcionar el resultado (un número entero que repre-

sentaría la cantidad de km. de vía ferroviaria que habría que adaptar).

Para evaluar la competencia técnica, el profesorado se limitaría a ejecutar los programas proporcionados y comprobar el resultado obtenido. Si se quiere, se podría realizar la entrega mediante una tutoría o entrevista con el fin de comprobar la autoría del trabajo y descartar copias. En una aproximación tradicional, la entrega se acompañaría de una memoria que daría respuesta a las preguntas *que formula el profesor* —en contraposición a los detalles o cuestiones que realmente hayan llamado la atención del alumnado.

Si consideramos nuestra propuesta, es el propio grupo de trabajo el que daría cuenta de los puntos fuertes y débiles de cada algoritmo: cuál es más sencillo de implementar, cuál conlleva mayor tiempo de ejecución con los datos proporcionados, cuál consume más memoria, análisis de la complejidad (teórica) temporal y espacial... El grupo de trabajo *debe* dar cuenta de todo esto en la primera página del documento que acompaña la entrega (véase la Sección 1) porque tiene que *preparar* —y “ganar”— un debate. Nótese, además, que el grupo de trabajo debe preparar las dos alternativas, pues no sabe de antemano cuál le tocará defender cuando se efectúe el debate.

Bajo nuestro punto de vista, esto resulta mucho más enriquecedor para el alumnado, el cual ejercitará con esta actividad, aparte de la competencia técnica a eva-

luar, los aspectos de *capacidad de síntesis* (debe sintetizar puntos fuertes y débiles de cada alternativa), *resumen de ideas* (debe organizarlos adecuadamente y no solaparlos), *correcta expresión escrita y dominio de la materia* (debe consultar la bibliografía para entender los aspectos que debe considerar a la hora de comparar algoritmos y cómo calcularlos). Obviamente, todos estos aspectos se ejercitarán, en mayor o menor grado, con independencia de la actividad a desarrollar y según el nivel requerido (no será igual para un estudiante de primer curso que para uno de tercero).

3. Preparación de argumentos

El siguiente paso en la preparación del debate será la elaboración de un esquema o hilo argumental que, considerando los puntos fuertes y débiles de cada alternativa desglosados en la fase anterior, proporcione una lista de argumentos a los que se puede recurrir en la realización del debate y en función de los argumentos que ponga sobre la mesa el adversario.

Continuando con nuestro ejemplo (*Actividad 3* del Cuadro 1), el grupo de trabajo podría preparar una lista de argumentos en la línea que se expone a continuación para apoyar la utilización del algoritmo de Prim en la resolución del problema planteado:

- El algoritmo de Prim es más fácil de implementar pues la estructura de datos más complicada que requiere usar es el conjunto³. Además es una extensión directa y natural del método del crecimiento del árbol a la hora de construir un árbol de recubrimiento a partir de un grafo.
- Las estructuras de datos que emplea se encuentran implementadas en la mayoría de lenguajes de programación modernos (Python, Java, C#...).
- Con los datos que se proporcionan, su tiempo de ejecución, en promedio, es el menor.
- Teóricamente, su complejidad temporal asintótica es mejor en la mayoría de las situaciones que podemos encontrarnos, mientras que su complejidad espacial es del mismo orden que la de Kruskal.

Análogamente, el grupo tendría que preparar otra lista de argumentos para poder apoyar la utilización del algoritmo de Kruskal. El objetivo que se persigue es transmitir la idea al grupo de trabajo de que sólo el conocimiento profundo de ambas posturas les podrá permitir argumentar a favor o rebatir adecuadamente en el desarrollo posterior del debate. Ambas listas de argumentos se expondrían en la segunda página del documento descrito en la Sección 1.

Creemos que, de esta forma, aparte de la competencia técnica que se desea ejercitar, el alumnado podrá

³Nos referimos a la versión más simple de Prim.

practicar los aspectos de *capacidad de síntesis* (debe agrupar uno o varios puntos fuertes en un único argumento), *elaboración de argumentos* (debe razonar acerca de la idoneidad de sus afirmaciones basándose en los puntos fuertes y preparar argumentaciones que oculten o maten los débiles), *correcta ortografía y sintaxis, correcta expresión escrita y dominio de la materia* (debe consultar la bibliografía para encontrar ejemplos o reputadas afirmaciones que apoyen sus argumentos).

Nuevamente, la intensidad con la que se ejercitarán estos aspectos necesarios para una eficiente comunicación escrita dependerá del nivel de la actividad planteada.

4. Preparación del debate

Para finalizar, el grupo de trabajo tendría que abordar la preparación del debate en sí, ya que el trabajo que se ha venido realizando en las dos fases anteriores (Secciones 2 y 3) se podría considerar como el trabajo de documentación y recopilación de datos previo que debe realizar todo buen ponente. De hecho, puede apreciarse que, de manera natural, el alumnado ya dispone de las herramientas que le van a permitir intentar convencer a la audiencia de que la alternativa que va a defender es la adecuada.

Lo que se pretende es que el grupo de trabajo prepare un escenario en el cual, a partir de una exposición inicial que desglose los principales argumentos elaborados en el apartado anterior, se puedan encadenar una serie de réplicas a los argumentos dados por la parte contraria —se trataría, fundamentalmente, de atacar los puntos débiles de la otra alternativa— y de contrarreplicas para rebatir los argumentos del adversario dirigidos a poner en evidencia los puntos débiles de la postura que les toca defender.

En suma, se trata de preparar una estrategia para “ganar” el debate y convencer a la audiencia (formada por el resto de la clase y el profesorado) de que la alternativa que le ha tocado defender al grupo de trabajo resulta la más adecuada. El grupo debería preparar un escenario para defender cada una de las dos alternativas propuestas. Ambos escenarios quedarían plasmados en la tercera página del documento solicitado en la Sección 1. Por ejemplo, el escenario que podría desarrollar el alumnado para apoyar el uso del algoritmo de Prim podría ser parecido a éste:

- Exposición inicial: el algoritmo de Prim es más fácil de implementar pues supone una extensión directa del método del crecimiento del árbol que resulta de aplicar el criterio voraz que minimiza el peso de las aristas elegidas. Además, su tiempo de ejecución en promedio con los datos propor-

cionados es más bajo.

- Si se cuestiona la facilidad de implementación, tenemos que rebatir con el argumento de que las estructuras de datos que emplea (conjuntos) se encuentran presentes en lenguajes como Python y Java.
- Si se cuestiona el consumo de memoria, debemos tomar como referencia siempre la complejidad asintótica, pues en la práctica Prim va a acabar consumiendo más memoria que Kruskal.
- Insistir en la complejidad de implementación que supone la estructura de datos conocida como MFSET para programadores no avezados.
- Exponer la complejidad temporal asintótica teórica. Puesto que Prim es mejor con grafos densos, hemos de dar por sentado que éste va a ser el tipo más frecuente con el que nos encontremos en el tipo de problemas que nos puedan plantear.

Los aspectos comunicativos que creemos que se ejercitarían en esta fase serían los mismos que ya se han comentado en las Secciones 2 y 3.

5. Realización del debate

La realización del debate sería la última fase del esquema propuesto y, en realidad, comprendería las fases de exposición pública y defensa del trabajo realizado que, tradicionalmente, se han venido considerando en el sistema universitario, sobre todo en lo que se refiere a la presentación de proyectos o tesis.

En nuestra experiencia del curso 2010/11, preparamos 3 actividades diferentes para los 32 grupos de trabajo formados⁴. Se asignó aleatoriamente cada grupo a una de las actividades propuestas (11, 11 y 10 grupos asignados a cada actividad, respectivamente). Como paso previo al debate, se debía entregar el documento cuyos contenidos hemos comentado en las Secciones 2, 3 y 4 en el espacio *Aula Virtual* de la universidad (Moodle). En una clase magistral anterior, se les explicó a los alumnos las bases de cómo hacer un buen debate: Gestualización, lenguaje corporal, asertividad y respeto.

El debate se realizó en clase de teoría, la cual tenía una hora y media de duración. Dividimos el tiempo de clase en tres periodos de media hora, asignando cada franja a una actividad distinta. En cada franja horaria debían estar presentes *solamente* los grupos asignados a la realización de la actividad correspondiente. El debate lo llevaban a cabo dos grupos elegidos por sorteo entre los presentes en el aula. La alternativa a defender por cada grupo y el orden de intervención también se sorteaban en ese momento. El resto de los grupos pre-

sentes en el aula debían seguir el debate y entregar un documento —en el *Aula Virtual* igualmente— donde resumiesen en una página sus conclusiones acerca del debate, la actuación de los grupos en el mismo y su postura personal sobre la actividad realizada.

El profesorado actuaba como moderador en el debate y tenía que puntuar la memoria entregada con anterioridad, la actuación de los grupos implicados en el debate —ofreciendo la pertinente realimentación al respecto de lo que habían hecho bien y lo que no— y el resumen que del mismo realizaban, después, el resto de grupos asistentes al debate.

Nuestra propuesta actual contempla que todos los grupos participen en el debate, lo cual puede crear ciertos problemas logísticos (tiempo, fundamentalmente). Además, esto nos llevaría a un escenario en el que la audiencia queda reducida al moderador del debate (el profesorado), ya que no queremos que el resto de grupos puedan verse influenciados en sus planteamientos por el transcurso de los debates previos a los que han asistido. En cualquier caso, este detalle puede quedar a criterio del profesorado y del valor que quiera dar a la parte de comunicación oral en la evaluación de la actividad. Al igual que hicimos hace dos años sorteáramos entre los dos grupos implicados en el debate la alternativa a defender y el orden de intervención. Con el objetivo de llevar a cabo el debate en un tiempo máximo de media hora aproximadamente, el esquema que sugerimos es el siguiente:

1. Turno de exposición inicial: cada uno de los grupos que tiene que debatir dispone de cinco minutos para llevarla a cabo. Aunque no es estrictamente necesario, se sugiere que el alumnado deba realizar una presentación con unas pocas diapositivas para ello. En función del curso, el nivel de la actividad propuesta y de las habilidades técnicas que se presumen en el alumnado se puede obligar o no a usar un programa para presentaciones tipo *Impress* de *LibreOffice*.
2. Dos turnos para cada grupo de réplica y contraréplica, con un máximo de tres minutos en cada intervención. Esto nos da un total de doce minutos como máximo en esta fase.
3. Conclusiones: cada grupo dispone de un turno de tres minutos máximo de duración para resaltar sus conclusiones con el objetivo de convencer a la audiencia.

Nótese que este mismo esquema propuesto también se puede aplicar, sin ningún tipo de modificación, si se prefieren realizar debates con el profesorado —es decir, el profesorado es el adversario— en actividades desarrolladas en seminarios o tutorías.

Siguiendo el esquema propuesto, en una hora se habrían celebrado debates realizados por cuatro grupos

⁴Grupos en su mayoría compuestos por tres alumnos. El resto, unos pocos, estaban formados por dos alumnos.

de trabajo distintos. Suponiendo una clase formada por unos 90 alumnos, en la cual habría aproximadamente unos 30 grupos con tres estudiantes cada uno, habríamos terminado con todos los debates en *ocho* horas aproximadamente. En el curso 2010/11, los debates (con sólo seis grupos) nos llevaron hora y media, pero el trabajo de corrección se cuantificó aproximadamente entre siete y ocho horas para los documentos previos a la preparación del debate y sobre unas tres horas para los documentos que exponían las conclusiones de los grupos que no habían participado en el debate.

Pensamos que en la realización del debate se ejercitan los aspectos de *adecuada capacidad de comunicación oral* (se debe ser capaz de expresar los argumentos de manera comprensible y seguir las pautas marcadas en el escenario previsto), *lenguaje gestual* (se tiene que adoptar una actitud activa y segura, “agresiva” en algunos momentos, para poder convencer a la audiencia, al contrario de la tendencia natural a adoptar una actitud pasiva cuando se realiza una exposición con diapositivas), *dominio de la materia* (se debe ser capaz de responder a argumentos fuera del guión previsto y eso implica conocer muy bien todos los detalles de la actividad realizada) y *control del tiempo de exposición* (obviamente, teniendo en cuenta que el moderador cronometra el tiempo y los turnos son estrictos).

En este punto, se podría objetar que no todos los alumnos van a poder ser evaluados de los aspectos mencionados, todos relacionados con la capacidad de comunicación oral. Esto es evidente ya que en nuestra propuesta siempre hemos hablado de una *actividad de grupo*. La solución es muy sencilla y ya se adoptó en el curso 2010/11 cuando se probó esta idea: uno de los alumnos se encarga de realizar la exposición inicial, otro se encarga de los dos turnos de réplica y contrarréplica y el otro de sintetizar las conclusiones del debate.

6. Inconvenientes

En esta sección queremos comentar los principales inconvenientes y dificultades que podemos encontrarnos a la hora de llevar adelante la propuesta que hemos presentado en este artículo. Hemos de decir que algunos de estos problemas los hemos padecido cuando la llevamos a cabo en el curso 2010/11, siendo fundamentalmente el esfuerzo requerido por parte del profesorado el principal óbice que nos llevó a abandonar su puesta en práctica en los dos cursos posteriores.

Hemos decidido resucitar la idea a la luz de las estrategias docentes centradas en métodos de autoevaluación y evaluación entre pares (o co-evaluación), con las que hemos tenido contacto durante el último año y que hemos conocido en mayor profundidad. Pensamos que estas estrategias pueden ayudar a resolver (casi en su totalidad) el principal problema que nos atenazaba: el

enorme trabajo de corrección que tuvo que realizar el profesorado en el pasado (véase la Sección 7).

6.1. Profesorado

Nuestra propuesta pretende realizar tanto evaluación formativa, centrada fundamentalmente en los comentarios que el profesor pueda hacer sobre el debate realizado —no sólo técnicos sino también desde el punto de vista de los aspectos implicados en la comunicación oral y escrita—, como sumativa, otorgando una calificación tanto a los aspectos técnicos como a los inherentes al proceso de comunicación realizado. Por tanto, la tarea del profesor, tal como se planteó en el curso 2010/11 (véase la sección anterior) consiste en:

- Preparar la actividad (cuestión que abordaremos en la siguiente sección).
- Corregir la respuesta a la parte técnica y el documento con la preparación del debate (con los contenidos descritos en las Secciones 2, 3 y 4), centrándose en el estilo, ortografía, sintaxis, capacidad de síntesis y elaboración de argumentos.
- Organizar los debates, moderarlos y corregir los aspectos implicados en el proceso de comunicación oral.
- Corregir los informes de los grupos asistentes al debate.

A la vista de esto, puede entenderse que el esfuerzo del profesor es excesivo. Además, se centra fundamentalmente en la parte de evaluación más que en la de orientación —que sería lo deseable. De hecho, hemos de resaltar que el volumen de trabajo fue tal que se descartó la realización de este tipo de actividad en los dos últimos cursos. Creemos que estos inconvenientes pueden superarse si el proceso de evaluación se lleva a cabo de la forma propuesta en la Sección 7.

6.2. Preparación de actividades

Bajo nuestro punto de vista, el trabajo de diseño y preparación de la actividad requiere algo más de trabajo por parte del profesorado, pero no mucho más. Por ejemplo, la preparación de la *Actividad 3* planteada en el Cuadro 1 implicaría que el profesor pensase en un problema para cuya resolución hubiese que emplear o adaptar algunos de los algoritmos conocidos. El trabajo del profesor consistiría, además, en elegir los datos de entrada adecuados en función de las situaciones con las que desea que se enfrenten los alumnos. Como se puede apreciar, no habría que hacer mucho más trabajo que el requerido para el diseño de cualquier actividad en el contexto del tipo de asignaturas que hemos puesto como ejemplo.

Obviamente, no todas las actividades admiten el enfoque sugerido: dos alternativas contrapuestas que se

van a debatir, en resumen. Tampoco pretendemos que todas las actividades de una asignatura se orienten de esta manera. Creemos que bastaría con plantear una única actividad en este sentido —dos, si se quiere, considerando la primera como trabajo de preparación de la segunda y, por tanto, no siendo evaluada— si en nuestra asignatura tenemos que evaluar la competencia de comunicación oral y escrita.

En el caso de las actividades que pueden plantearse de la manera propuesta, debe tenerse en cuenta que lo aconsejable es que las dos alternativas escogidas estén *parejas* para que exista debate. Pudiera ocurrir —como es el caso de la *Actividad 2* propuesta en el Cuadro 1— que hubiese una solución mejor que otra, pero se recomienda que sólo lo fuera parcialmente, y que, en algún sentido, se pudiese compensar con alguna característica que sólo es propia de la otra alternativa.

6.3. Alumnado

El alumnado no sólo debe realizar las tareas necesarias para poder dar una respuesta correcta a la competencia técnica que se desea ejercitar, sino que también tiene que pensar y documentarse sobre ella. Se pretende que el alumnado lleve a cabo el necesario proceso de reflexión que requiere la compleción de cualquier actividad [2].

Se podría argüir que la actividad que tiene que desarrollar el alumnado es excesivamente compleja, a la vista de las reflexiones que es necesario realizar. Sin embargo, esto no es realmente así, ya que la complejidad de la actividad se puede modular tanto en función del problema propuesto como de las alternativas seleccionadas para competir. Además, al contrario que con el modelo del *Pecha Kucha*, no necesariamente se requiere el uso de programas para realizar presentaciones —aunque resulta conveniente emplearlos para realizar la exposición inicial.

Igualmente, se podría argumentar que el alumnado debe realizar mucho trabajo para completar una actividad diseñada siguiendo las pautas de nuestra propuesta. Conviene recordar, no obstante, que realmente no se requiere más trabajo que el que supone la alternativa tradicional, es decir, la entrega de la actividad acompañada de una memoria. Si se quiere realizar una memoria de calidad resulta obligado efectuar la mayor parte de los procesos reflexivos que pautamos al alumnado en nuestra propuesta. El alumnado ha de ejercitar igualmente la competencia técnica y lo único que hacemos es guiar el desarrollo de la memoria, obligándole a realizar el necesario proceso de reflexión sobre la tarea efectuada que resulta clave en el proceso de aprendizaje [2].

Frente a aproximaciones más vanguardistas, tales como presentaciones de actividades basadas en el modelo *Pecha Kucha* o en el uso de video-píldoras, pensa-

mos que el esfuerzo que hay que realizar para llevar a cabo actividades según el modelo propuesto es comparable. Creemos que el trabajo de diseño, preparación y ensayo de la presentación o del vídeo (y su grabación) es similar al esfuerzo que implica preparar el debate, con la ventaja de que nosotros proporcionamos el esquema de antemano.

6.4. Debates

Precisamente, a la hora de realizar el debate es donde, desde nuestro prisma, nos encontramos con los mayores inconvenientes:

1. El tiempo que conlleva realizar todos los debates. En un grupo de alrededor de 90 alumnos, con 30 grupos de trabajo formados, nos llevaría cerca de ocho horas. Obviamente, si hablamos de grupos más reducidos (unos treinta alumnos o menos) este inconveniente ya no sería tal.
2. La audiencia del debate. Pensamos que para que la actividad sea lo más enriquecedora posible, todos los alumnos deberían asistir al debate, con el profesorado actuando de moderador —y cronometrador— del mismo. Sin embargo, esto plantea algunos problemas, ya que, sin duda, los asistentes a un debate se verán influenciados, en cierta forma, por los argumentos y actitudes empleados por los protagonistas, lo cual condicionará, en alguna medida, su papel cuando les toque ser los actores del debate.

El primer inconveniente puede obviarse planteando el tiempo de debate como horas de trabajo presencial del alumnado. Desde el punto de vista del profesorado, serían las horas que emplearía en la evaluación o, mejor, en un papel de guía u orientador del aprendizaje. Siguiendo la propuesta de evaluación descrita en la siguiente sección, el papel del profesorado en el debate sería meramente orientador y facilitador del aprendizaje —lo deseable. El segundo inconveniente resulta más difícil de soslayar, a no ser que sean pocos grupos. Proponemos tres posibles soluciones⁵:

1. Aplicar la misma solución empleada en el curso 2010/11: el debate lo realizan sólo los dos grupos escogidos por sorteo y el resto (audiencia) se limita a enviar un informe con sus conclusiones sobre el debate. El inconveniente de esta propuesta es que, entonces, la evaluación de los aspectos asociados a la comunicación oral sólo se realiza con los dos grupos participantes.
2. No hay audiencia, tan sólo están presentes los dos grupos que van a debatir y el profesorado como

⁵Descartamos la idea de proponer más de una actividad diferente, ya que multiplicaríamos el trabajo preparatorio del profesorado.

moderador. El inconveniente es que restamos trascendencia al proceso comunicativo en sí, que en este caso coincide con el de mayor relevancia y sobre el cual centramos la presentación de la actividad, además.

3. Dividir la asistencia a los debates en franjas horarias. Por ejemplo, podríamos asignar cinco franjas horarias distintas a las que asistirían sólo seis grupos en cada una de ellas (partimos de 90 alumnos, 30 grupos). En cada debate participarían dos grupos y los otros cuatro constituirían la audiencia. Cabe esperar que, al asistir a menos debates como público —los que intervienen en último lugar sólo habrían asistido a dos debates previos—, los participantes en los últimos debates se vean menos mediatizados. Particularmente, esta solución es la que nos gusta más.

7. Propuesta de evaluación

Para finalizar, vamos a comentar los puntos fundamentales de nuestra propuesta en cuanto a evaluación se refiere. Los objetivos que perseguimos son dos; por un lado descargar de trabajo de corrección al profesorado para que pueda centrarse en la orientación del aprendizaje y, por otro, que el alumnado reflexione sobre su propio trabajo y su aprendizaje [2].

Para evaluar la competencia técnica, pensamos que se podría aplicar un mecanismo de autoevaluación mediante rúbrica o bien, si se trata de proporcionar una (o varias) respuestas concretas (como sería el caso de la *Actividad 3* del Cuadro 1), algún tipo de evaluación automática, por ejemplo introduciendo las respuestas en un cuestionario del *Aula Virtual*. Otra posibilidad sería que el profesorado evaluase la parte técnica mediante rúbrica. Esta evaluación sería sumativa.

En lo que concierne al documento de preparación del debate, cabe resaltar que en su evaluación confluyen tanto algunos aspectos técnicos como los relacionados con la competencia de comunicación escrita. Nuestra propuesta consiste en realizar una evaluación entre pares (co-evaluación) mediante rúbrica. La rúbrica centrada en la evaluación de los aspectos técnicos sería sumativa, y pensamos que debería ser supervisada por el profesorado. La rúbrica relativa a los aspectos de la comunicación escrita tendría dos partes, una sumativa y otra formativa. Creemos que esta última no tendría que ser supervisada, ya que cada grupo de trabajo podría verla e indicar si está conforme o no con ella. El profesorado simplemente se limitaría a ejercer de árbitro en las posibles disputas que pudiesen surgir.

Pensamos que existen dos posibilidades para realizar la evaluación del debate, ambas efectuadas mediante rúbrica: evaluación entre pares o realizada por el profesorado [6] (véanse también los documentos “Tea-

cher Mini-Debate Guide”⁶ y “Class Debate Rubric”⁷). En su papel de moderador del debate, el profesorado podría cubrir dos aspectos: la orientación que proporciona la evaluación formativa y la puntuación otorgada por la sumativa. La co-evaluación plantea problemas de organización. Por ejemplo, ¿quién de los grupos asistentes al debate se encarga de evaluar a los que han intervenido en el mismo? Además, este proceso dependería de la solución adoptada a los inconvenientes presentados en la Sección 6.4. En el caso de que se optase por la primera solución planteada en dicha sección —que es la que adoptamos en el curso 2010/11— habría que evaluar el informe entregado por los asistentes. Nuestra propuesta sería no evaluarlo y, simplemente, comprobar si se ha entregado o no. Alternativamente, se puede comprobar si ha sido correctamente redactado.

Agradecimientos: Financiado por la «Unitat de Suport Educatiu» de la Universitat Jaume I en el marco del Proyecto de Innovación Educativa 10G136-329.

Referencias

- [1] Isabel Gallego, José Manuel López, Eva Rodríguez, Esther Salamí, Eduard Santamaría y Miguel Valero. Presentaciones orales a un coste razonable. En *JENUI*, páginas 25–32, 2010.
- [2] Graham Gibbs y Claire Simpson. Condiciones para una evaluación continuada favorecedora del aprendizaje. Número 13 en *Cuadernos de docencia universitaria*. OCTAEDRO, 1ra. edición, enero 2009.
- [3] Joaquín Gracia y M. Ángeles Pinar. Un caso práctico de evaluación de competencias lingüísticas en informática. *Revista de Formación e Innovación Educativa Universitaria*, 5(2):99–111, 2012.
- [4] David López y Alex Ramírez. Marco para el desarrollo de la competencia transversal “comunicación eficaz”. En *JENUI*, páginas 213–220, 2011.
- [5] Joe Miró. La técnica de la escritura técnica, abril 2008. Disponible en <http://bioinfo.uib.es/~joemiro/RecEscr/protolibro.pdf>.
- [6] Ricardo Olanda, Máximo Cobos y Paloma Moreno. Evaluación por compañeros de exposiciones orales. En *JENUI*, páginas 113–120, 2012.
- [7] Miguel Valero-García y Luis M. Díaz de Cerio. Autoevaluación y co-evaluación: estrategias para facilitar la evaluación continuada. En *Simposio Nacional de Docencia en la Informática*, páginas 25–32, 2005.

⁶http://www.proquestk12.com/productinfo/pdfs/MiniDebate_Teachers.pdf.

⁷<http://course1.winona.edu/shatfield/air/classdebate.pdf>.

Índice de autores

Alonso Pardo, Miguel Ángel, 35
Alonso Ríos, David, 35
Amengual, Juan Carlos, 93

Baró, Xavier, 27
Barreira Rodríguez, Noelia, 35
Benlloch Dualde, José V., 3
Buendía García, Félix, 3

Cabrero Canosa, Mariano J., 35
Coltell, Oscar, 61

Escribano Otero, Juan José, 19

Flich, José, 69

Gómez Rodríguez, Carlos, 35
Gómez, Crispín, 75
Gómez, María E., 75
García García, María José, 19
Gaya López, María Cruz, 19
Goñi, Alfredo, 43
Granell, Ximo, 61
Guijarro Berdiñas, Bertha, 35

Hernández Pereira, Elena, 35

Ibáñez, Jesús, 43

Igual, Laura, 27
Iturrioz, Jon, 43

Latorre, Pedro, 61
Lemus Zúñiga, Lenin Guillermo, 3
Lizán, Luís, 61
Llorens, Faraón, 87

Marqués Andrés, Mercedes, 93
Martínez, José María, 69
Montañana Aliaga, José Miguel, 3

Pérez Sánchez, Beatriz, 35
Perez Poch, Antoni, 83
Polo Márquez, Antonio, 11

Recatalá, Gabriel, 53
Roca, Antoni, 69

Sánchez, José Salvador, 61
Sahuquillo, Julio, 75
Sales, Jorge, 53
Sanchez Maroño, Noelia, 35

Tosca, Ricardo, 61

Vadillo, José Ángel, 43
Vilares Ferro, Jesús, 35