

¿CÓMO COMPLEMENTAR LA FORMACIÓN MEDIANTE PROYECTOS DE GENERACIÓN DE CONTENIDOS WEB PARA UNA ASIGNATURA?

ANTONIO SARASA CABEZUELO¹ y
JUAN MANUEL DODERO BEARDO²

¹Universidad Complutense de Madrid,

²Universidad Carlos III de Madrid

¹Facultad de Informática
Departamento de Sistemas Informáticos y Programación
C/Juan del Rosal, 8
28040 – Madrid - España
Email: asarasa@sip.ucm.es

²Escuela Politécnica Superior
Edificio Sabatini
Avda. de la Universidad, 30.
28911 - Leganés - Madrid - España
Email: dodero@inf.uc3m.es

Resumen: En este artículo se describen las conclusiones que se han obtenido en las experiencias piloto realizadas durante el curso 2003/04 en el contexto de dos asignaturas distintas de la titulación de Ingeniería en Informática impartidas en dos universidades diferentes de la comunidad de Madrid. El objetivo de las mismas era poner en práctica una metodología de trabajo que complementara la formación presencial que reciben los alumnos, y demostrar por una parte que el uso de las nuevas tecnologías como elemento adicional de la enseñanza presencial puede mejorar el nivel de aprendizaje y la motivación de los alumnos, y por otra parte que estas técnicas se pueden extrapolar con las adaptaciones a las asignaturas de otras titulaciones independiente de la naturaleza de la materia.

Palabras clave: E-Learning, Universidad Virtual, Educación mixta, E-pedagogy, Educación online.

Abstract: In this article we describe the conclusions that we have obtained in an experiences made during the course 2003/04 in the context of two subjects different from the degree of Engineering in Computer science in two universities different from Madrid. The objective was to get work a methodology for complementing the formation from the students, and to demonstrate that the use of the New Technologies like additional element can get better the level of learning and the motivation of the students. Also these techniques can be extrapolated with

adaptations to an others subjets from others degrees with independent of their nature.

Keywords: E-Learning, Virtual Universities, Blended Education, E-pedagogy, Online Education.

1. Introducción

En todas las titulaciones universitarias siempre existe un conjunto de asignaturas que se pueden calificar como conflictivas en vista a los efectos que tienen sobre los alumnos: un alto número de suspensos, abandono de la asignatura, un tiempo excesivo en conseguir aprobar la materia o alumnos que aprueban pero que no han llegado a asimilar los conceptos mostrados y que tienen dificultades en asignaturas posteriores que dan por conocidos estos conocimientos. Las razones que explican el porqué una asignatura llega a tener los problemas comentados son muy diversas, pero esquemáticamente se pueden diferenciar tres posibles orígenes que los causan:

1. Deficiencias del docente.: Falta de formación específica y didáctica para impartir los contenidos de la asignatura. Incapacidad pedagógica para impartir clases y transmitir conocimiento. Metodología errónea para mostrar el tipo de materia que imparte.
2. Deficiencias del alumno: Formación previa insuficiente para la asimilación de nuevos conceptos que suponen el conocimiento de otros. Naturaleza propia del alumno. Falta de hábitos de estudio. Desmotivación del alumno. Naturaleza de la asignatura. Apoyo de la asignatura sobre los conceptos de otras. Descoordinación temporal con otras asignaturas, cuyos conocimientos están relacionados. Necesidad de una alta dedicación de tiempo para su aprendizaje. Alto nivel teórico y lejanía de la realidad práctica. Excesiva materia en un periodo de tiempo corto.

Con la motivación de esta problemática y con el objetivo de poner medios que aportaran soluciones a la misma, se han desarrollado dos experiencias que implementan un conjunto de prácticas metodológicas basadas en técnicas que se usan en Ingeniería del software en la realización de proyectos software. Además de aportar soluciones para los casos particulares de las asignaturas objeto de la experiencia, también se tenía otro objetivo esencial. Ese otro objetivo a alcanzar era que las conclusiones que se obtuvieran fueran significativas y extrapolables a otros casos. Por ello las asignaturas objeto del experimento debían cumplir una serie de requisitos especiales con respecto a:

- Número de alumnos. Una asignatura masificada frente a una asignatura no masificada, con el fin de demostrar la independencia del método aplicado frente al número de alumnos que debe atender un profesor.

- Lugar de impartición. Asignaturas impartidas en universidades distintas, con el fin de demostrar la independencia frente al plan de estudios, al profesor y al entorno en el que se aplica.
- Temporalidad de su realización. Asignaturas impartidas en distintos ciclos del plan de estudios.
- Carácter de la problemática presente en cada asignatura. Asignaturas que presenten diferentes aspectos problemáticos.
- Naturaleza de los contenidos. Contenidos de distintas naturaleza y con distinto grado de aplicabilidad en la realidad.

En base a estos condicionantes se eligieron las asignaturas de “Estructuras de Datos y de la Información” y “Programación Avanzada”. Ambas asignaturas pertenecen a la titulación de Ingeniería en Informática, y cumplen los requisitos comentados, que pueden verse en la tabla 1.

Asignatura	“Estructuras de Datos y de la Información”	“Programación Avanzada”
Número de alumnos	141	30
Lugar de impartición	Universidad Complutense de Madrid. Departamento de Sistemas Informáticos y Programación.	Universidad Carlos III de Madrid. Departamento de Informática
Temporalidad de su realización	Primer ciclo. Segundo curso.	Segundo ciclo. Cuarto curso.
Problemática de la asignatura	Alto contenido matemático.	Densidad de la materia, y de las habilidades requeridas.
Naturaleza de los contenidos	Troncal. Teórica.	Optativa.Práctica.

Tabla 1. Características de las asignaturas objeto de estudio.

Así en este artículo describimos los pormenores de los experimentos realizados en cada asignatura, y las conclusiones a las hemos llegado. Para ello el artículo se ha estructurado en tres apartados en los que relatamos el experimento en la asignatura de “Estructuras de Datos y de la Información”, el experimento en la asignatura de “Programación Avanzada” y por último establecemos unas conclusiones.

2. La metodología

En ambas asignaturas se ha aplicado una misma metodología, aunque adaptada a las particularidades presentes en cada una de ellas. Las características de esta metodología son:

- a) Generación de contenidos educativos Web. Uno de los objetivos que se quiere alcanzar es complementar la formación del alumno, y ayudarle a aprender. Por ello

creemos que una forma de conseguirlo es mediante la realización de trabajos relacionados con la creación de recursos educativos para la asignatura de estudio, de forma que los trabajos realizados puedan ser usados por otros alumnos para aprender la asignatura. Esta afirmación se basa en el siguiente razonamiento: al tratar de hacer recursos sobre la asignatura, los alumnos se ven obligados a estudiar la asignatura, ya que tienen que conseguir transmitir en esos trabajos el conocimiento necesario para que otros aprendan la asignatura. Se estimula su orgullo, ya que toman el papel de “profesores”, ya que otros van a usar lo que ellos hagan. El nivel de calidad exigido les obliga a imponerse un hábito de trabajo, y de investigación al tener que consultar distintos libros y recursos para llevar a cabo su trabajo. Otra característica esencial de los contenidos generados es la accesibilidad a los mismos. En este sentido un requisito de los trabajos es que se realicen en formato Web (Sugrue, 2000), de forma que disponiendo únicamente de acceso a internet y de un navegador se pueda acceder a la información, con lo que se asegura un acceso universal desde cualquier lugar.

b) Enseñanza a los alumnos de las nuevas tecnologías. En relación con el apartado anterior, un requisito ineludible si se quieren realizar los trabajos comentados es que los alumnos tengan una pequeña cultura sobre nuevas tecnologías (McFarlane, 2001). Concretamente lo mínimo que un alumno debe conocer es como crear documentos HTML. En este sentido creemos que uno de los primeros pasos para empezar a trabajar en nuevas tecnologías es que los alumnos aprendan HTML. Y no hay mejor forma que motivarles a que hagan su propia página Web, lo cual además de parecerles una estupenda idea, les permite plasmar toda su creatividad con toda libertad. Además también les hace aprender las herramientas necesarias que van a tener que usar al realizar sus trabajos.

c) Trabajo en grupo. Debido al tipo de trabajo que deben realizar, su extensión no es asumible por una sola persona, y además tampoco es el objetivo, ya que lo que se busca, es que al trabajar en grupo, el grupo ponga distintas ideas en juego (Tao y Gunstone, 1999), enriqueciendo así la formación de todos aquellos que participan en el trabajo. Por otra parte el trabajo en grupo desarrolla en los alumnos el sentido de la responsabilidad, al depender el trabajo de cada uno de que los otros miembros del grupo cumplan con sus partes.

d) Motivación mediante una filosofía Premio/Castigo. Si bien es cierto que la motivación antes comentada basada en el orgullo que pueden sentir de ser “profesores” de sus compañeros es importante, sin embargo no es suficiente para que de manera “gratuita” hagan este tipo de trabajos (Taylor, 1980). Su trabajo, igual que cualquier otro trabajo, debe ser premiado si se realiza adecuadamente o castigado en caso contrario. Es por ello que este tipo de trabajos deben ser evaluados con una nota adicional a la que pudieran obtener en los exámenes de la asignatura, e incluso planteamos que pudieran convertirse en una forma de evaluación, por lo que antes argumentábamos de que la realización de estos trabajos obliga al alumno a aprender de verdad la asignatura, por lo que en cierta

forma se consigue el objetivo del aprendizaje, siendo por tanto innecesario su evaluación mediante un examen.

e) Gestión basada en la Ingeniería del Software. Gestionar distintos grupos de alumnos, en cualquier asignatura normal, supone hacerse cargo de un conjunto demasiado grande para un solo profesor. Es por ello que se ha acudido al uso de algunas de las técnicas que se usan en la Ingeniería del Software (Sommerville, 2002) para gestionar proyectos software. Los proyectos software comparten muchas de las características del tipo de trabajo que van a tener que realizar: se trata de un trabajo en equipo, participan muchas personas que necesitan ser coordinadas, el trabajo que realiza una parte del equipo depende del trabajo realizado por otros miembros del equipo, hay que cumplir unas fechas de entrega del trabajo que se va realizando, hay un cliente, hay un pago por el trabajo realizado. En la tabla 2 se puede ver una comparación de las similitudes comentadas.

Proyecto Software	Recursos educativos.
Cliente	Profesor
Dinero	Puntos
Muchas personas. Trabajo coordinado en equipo	Muchas personas. Trabajo Coordinado en equipo
Plazos de entrega. Planificación.	Plazos de entrega. Planificación.
Proyectos y subproyectos.	Trabajos y tareas.
Jefe de proyecto y jefe de subproyecto	Profesor y alumno jefe del equipo.
Especificaciones	Especificaciones

Tabla 2. Similitudes entre los proyectos software y la generación de recursos educativos.

Así los trabajos se han configurado como la realización de un proyecto software (Guzdial, 1998), en la que la asignatura toma el papel de un proyecto en general, y cuyo jefe de proyecto es el profesor de la asignatura. Este proyecto principal se divide a su vez en varios subproyectos que llevan a cabo diferentes tareas. Cada subproyecto se caracteriza por:

- Recibe unas especificaciones técnicas de qué debe hacer, en que plazos, con que tecnologías, y una planificación de las distintas reuniones que se deben realizar con el profesor para controlar el progreso del trabajo.
- Uno de los miembros actúa como jefe del subproyecto, y tiene como misión coordinar el subproyecto, y servir de nexo entre el equipo que forma el subproyecto y el profesor.
- A su vez cada subproyecto, se divide en pequeños equipos de trabajo que realizan labores independientes pero que después tendrán que enlazarse y que el jefe del subproyecto debe supervisar.

- El jefe de proyecto para conseguir que el trabajo fructifique, puede usar técnicas de planificación, gestión de riesgos, que le facilitarán la gestión de sus equipos de trabajos, y el cumplimiento de los plazos establecidos por el profesor.

3. Experiencia en la asignatura de “Estructuras de Datos y de la Información”.

La asignatura de “Estructuras de Datos y de la Información”, es una asignatura troncal de primer ciclo del plan de estudios de Ingeniero en Informática, que se cursa en el segundo curso en la Universidad Complutense de Madrid, y que es impartida por el Departamento de Sistemas Informáticos y Programación. Las características que la definen son:

- Asignatura numerosa. Existe una gran cantidad de alumnos que se matriculan en la asignatura. Actualmente hay 3 grupos abiertos, 2 por la mañana y uno por la tarde, con una media de 140 alumnos en cada grupo. En estas condiciones la atención que puede tener el profesor con cada alumno es muy limitada.
- Dificultad conceptual. La materia de que versa la asignatura tiene un alto grado de abstracción y un carácter muy teórico, siendo necesaria una buena base de “Lógica matemática” y “Matemática Discreta”. En general los alumnos rehuyen de todas aquellas asignaturas en la que la componente práctica es mínima o inexistente.
- Alta tasa de suspensos y abandono de la asignatura. Debido a su alto contenido matemático y a la débil base matemática con la que muchos de los alumnos llegan a la asignatura, las tasas de suspensos y abandonos son bastantes altas.
- Rechazo generalizado por los alumnos. Por las razones anteriormente esgrimidas se trata de una asignatura que presenta un alto rechazo entre los alumnos, considerándola como un “asignatura hueso”.

Ante este contexto el profesor se enfrenta al reto de motivar a unos alumnos que parten desde el principio y antes de conocer la asignatura con un alto grado de desmotivación, rechazo y miedo a la dificultad que presenta la misma. Así la puesta en práctica de esta metodología de trabajo se realizó de forma que se cubrieran dos objetivos, por una parte complementar la formación de los alumnos a nivel de conocimiento, y por otra elevar la motivación y el interés de los alumnos por la asignatura.

Aunque se tomaron como base las prácticas metodológicas antes comentadas, sin embargo en la adaptación a las particularidades propias de la asignatura, se introdujeron dos elementos novedosos respecto a los comentados, y que van a ser los factores en que se diferencien ambas experiencias:

- Carácter voluntario en la participación del experimento. Un punto importante de la experiencia en esta asignatura es no exigir a los alumnos la participación

obligatoria en el experimento. En una asignatura de las características mencionadas, en general el alumnado se siente reacio a realizar más del trabajo necesario, pues piensa que el trabajo que tiene que hacer para aprobar cubrirá con creces el tiempo que dedique a la asignatura. De esta forma los trabajos se realizaron en una labor de equipo y colaboración entre profesor y alumnos voluntarios que quisieran participar en la misma.

- Apoyo a la docencia mediante una herramienta de teleformación. Los servicios, y en particular las herramientas de comunicación tales como foros, correo electrónico o Chat, que ofrecen este tipo de herramientas abren a los alumnos y al profesor un canal de comunicación entre ellos que les permiten intercambiar opiniones, conocimientos o dudas que enriquece positivamente (Marra y Jonassen, 2001) el ambiente de trabajo y la sensación de formar un grupo, que debería existir en cualquier asignatura. Por otra parte hay que señalar el papel clave (Marcelo, Puente, Ballester y Palazón, 2001) que desempeñan las herramientas de teleformación para implementar este tipo de trabajos, ya que ofrecen la posibilidad de gestionar grupos de trabajo entre los alumnos dados de alta. Disponiendo estos de su propio espacio local de almacenamiento, así como de herramientas privadas para el grupo. En nuestro caso se ha utilizado la herramienta de WebCT (WebCT) que proporciona el campus virtual de la Universidad Complutense de Madrid.

Así la puesta en marcha de la experiencia se llevo a cabo en dos fases:

Fase 1. Acercamiento al entorno de la herramienta. Esta primera fase se ha implementado usando varias estrategias:

- Realización de sesiones presenciales en un laboratorio de programación, para mostrar a los alumnos el funcionamiento de la herramienta y los servicios que puede ofrecer a sus necesidades.
- Creación de una página Web personal para colgar en un espacio personal que ofrece la herramienta en las cuentas de que disponen cada alumno.
- Puesta a disposición de los alumnos de herramientas de comunicación de utilidad para ellos tales como: foros de discusión de distinta temática donde los alumnos pudieran dejar sus dudas, comentarios,..., y que bien el propio profesor, o bien otros alumnos pudieran contestarle, correo electrónico interno de la herramienta, de forma que los participantes puedan comunicarse entre sí, pizarra virtual, en donde varios participantes pueden compartir un espacio común, en el cual pueden escribir o dibujar, y de manera simultánea pueden ver lo que hace cada uno de ellos, chat, en donde pueden conversar varios participantes, y enlaces, en donde se colocaron enlaces que pudieran ser interesantes para el desarrollo de la asignatura.
- Entrega a los alumnos de diversos manuales introductorios sobre algunas de las tecnologías que debían usar en sus trabajos.

Fase 2. Generación de contenidos educativos para la asignatura. Se han planteado dos convocatorias distintas de trabajos, una durante el primer parcial y otra durante el segundo parcial.

Subproyecto	Nº alumnos	Subproyectos	Nº alumnos
Complejidad Algorítmica	8	Trabajo 1	8
Especificación formal	8	Trabajo 2	8
Diseño recursivo	8	Trabajo 3	8
Diseño iterativo	8	Trabajo 4	8
Programa Tutor	10	Trabajo 5	8

Tabla 3. Trabajos planteados durante el primer parcial

En la primera convocatoria se planteó a los alumnos la realización de 10 trabajos (Tabla 3) que versaban sobre los contenidos explicados durante la primera parte de la asignatura, "Diseño de algoritmos recursivos e iterativos" (Peña, 1998). La realización de cada trabajo se planteó como un subproyecto siguiendo las directrices antes comentadas. Cada subproyecto tenía asignado entre 8 y 10 personas y podían obtener como máximo 2 puntos. La estructuración de los trabajos fue:

- Material teórico/práctico básico. En este tipo entran los trabajos que aparecen en la tabla denominados: Complejidad algorítmica, Especificación formal, Diseño recursivo y diseño iterativo. El objetivo de estos trabajos es crear recursos educativos autocontenido en forma de documentos HTML sobre cada uno de los temas. Constan de: Apuntes, ejercicios resueltos, cuestionarios de autoevaluación, y exámenes junto a su resolución. El resultado final debe ser un recurso que permita la navegación lógica sobre cada uno de los apartados comentados.
- Problemas aplicados. En este tipo entran los trabajos que aparecen en la tabla denominados: Trabajo 1, Trabajo 2, Trabajo 3, Trabajo 4 y Trabajo 5. El objetivo de estos trabajos es la generación de recursos educativos que ilustren el desarrollo completo de un supuesto práctico del que únicamente se conoce el enunciado del problema. El material debe ser autocontenido en forma de documentos HTML navegable que permitan ver los distintos desarrollos, la ejecución de un applet de Java con el programa implementado así como las explicaciones de cómo se han llegado a estos desarrollos.
- Programa de ayuda al estudio. El objetivo de este trabajo es la creación de un programa que actúe como un tutor artificial más o menos

inteligente que ayude a los alumnos a comprender los pasos que tienen que dar para poder diseñar un algoritmo.

En la segunda convocatoria se planteó a los alumnos la realización de 8 trabajos (Tabla 4) que versaban sobre los contenidos explicados durante la segunda parte de la asignatura, "Estructuras de datos" (Martí, Mallén y Verdejo, 2003). La realización de cada trabajo tenía asignado entre 8 y 10 personas con un valor de 2 y 3 puntos como máximo, según la dificultad del trabajo.

Subproyecto	Nº alumnos	Subproyectos	Nº alumnos
Pilas	8	Árboles de búsqueda y Tablas Hash	8
Listas	8	Grafos	8
Colas	8	MiniLenguaje de Especificación	10
Árboles binarios y generales	8	Colas de Prioridad y Montículos	8

Tabla 4. Trabajos planteados durante el primer parcial

La estructuración de los trabajos fue:

- Material teórico/práctico básico+ Problemas aplicados. De este tipo son todos los trabajos que aparecen en la tabla salvo el trabajo denominado "Minilenguaje de Especificación". En estos trabajos se pretende conseguir dos objetivos. Por una parte se pretenden crear recursos educativos autocontenidos en forma de documentos HTML sobre cada uno de los temas. Constan de: Apuntes, ejercicios resueltos, cuestionarios de autoevaluación, y exámenes junto a su resolución. El resultado final debe ser un recurso que permita la navegación lógica sobre cada uno de los apartados comentados. Y por otra parte deben incluir la resolución de un problema aplicado que ilustre el desarrollo completo de un supuesto práctico del que únicamente se conoce el enunciado del problema. El material debe ser autocontenido en forma de documentos HTML navegable que permitan ver los distintos desarrollos, la ejecución de un applet de Java con el programa implementado así como las explicaciones de cómo se han llegado a estos desarrollos.
- Minilenguaje de Especificación. El objetivo de este trabajo es la creación de un programa que actúe como un tutor artificial más o menos inteligente que ayude a los alumnos a comprender los pasos que tienen que dar para poder diseñar un algoritmo.

Todos los trabajos realizados se han dejado al acceso libre de todos los alumnos de la asignatura a través de la herramienta WebCT, y además en cada trabajo es posible bajarse una copia del mismo para su ejecución en local, de forma que no haya que estar conectado, y pueda realizarse su estudio de manera off-line.

4. Experiencia en la asignatura de “Programación Avanzada”.

La asignatura de “Programación Avanzada”, es una asignatura optativa de segundo ciclo del plan de estudios de Ingeniero en Informática, que se cursa en el cuarto curso en la Universidad Carlos III de Madrid, y que es impartida por el Departamento de Informática. Las características que la definen son:

- Asignatura poco numerosa. Existe un solo grupo de 50 alumnos. En estas condiciones la atención que puede tener el profesor con cada alumno puede ser relativamente personalizada.
- Dificultad por el número de habilidades a conocer y usar. La materia de que versa la asignatura hace necesario el dominio de una amplia variedad de conocimientos como por ejemplo: comunicaciones entre redes, base sólida de java, programación distribuida, aplicaciones web o bases de datos. Sin embargo en general el nivel con que los alumnos llegan a esta asignatura es medio, teniendo una curva de aprendizaje larga, hasta que pueden usar todo el potencial de las herramientas que se les explican.
- Agrado generalizado de los alumnos. En general los alumnos se sienten motivados por la asignatura, al tener un perfil muy práctico. Considerándolo como una “asignatura útil” para cuando aterricen en el mundo empresarial.
- Alta tasa de aprobados pero con unos conocimientos muy débiles. La motivación a la que hacíamos referencia hace que los alumnos se interesen por la asignatura, y le dediquen tiempo, lo cual se ve reflejado en las tasas de aprobados. Sin embargo ese número de aprobados no muestra la realidad sobre el entendimiento real de la asignatura, sobre su asimilación de cómo aplicar las técnicas aprendidas en la realidad a casos prácticos. Esto se debe al número limitado de horas de clase de que se dispone y a la densidad de la materia que hay que explicar, que en muchas ocasiones hace necesario incluir recordatorios de conocimientos ya enseñados, o de conocimientos que se deberían conocer y no se conocen. De esta forma el número de horas teóricas reduce el número de horas prácticas posibles.

Ante este contexto el profesor se enfrenta al reto de tratar de no solo enseñarles unos conocimientos teóricos, sino también enseñarles cómo y donde se utilizan, de forma que puedan aprovechar toda su potencialidad. Y todo ello dentro del tiempo limitado del que disponen, y que normalmente suele ser consumido en horas de clases de teoría. Así la puesta en práctica de la metodología de trabajo comentada se realizó con el objetivo de complementar la formación de los alumnos a nivel de conocimiento, y mostrarles esa visión aplicada de los conocimientos teóricos aprendidos.

De igual forma que en el caso anterior se tomaron como base las prácticas metodológicas antes comentadas, pero con las adaptaciones necesarias a las particularidades propias de la asignatura, e introduciendo dos elementos novedosos respecto a los comentados, y que van a representar la diferencia con respecto a la anterior experiencia:

- Carácter obligatorio en la participación del experimento. Un punto importante de la experiencia en esta asignatura y que la diferencia de la anterior es el carácter obligatorio de la misma. Todos los alumnos deben participar en la experiencia. Existen dos diferencias con el caso anterior que hacen más factible el realizar esta prueba en estas condiciones, y que son, por una parte el menor número de alumnos, y por otra la motivación y predisposición de los alumnos a intentar aprender los conocimientos de la asignatura al considerarlos útiles.
- Una misma práctica de carácter competitivo. El planteamiento en esta experiencia, a diferencia de la anterior (en la que se trabajaban sobre diferentes tipos de trabajo y temática), va a ser realizar un mismo proyecto desde el comienzo de la asignatura. Este proyecto se va realizando según se van aprendiendo los conocimientos necesarios para realizarlo, es decir se explican técnicas guiadas por las necesidades del proyecto a realizar. Pero existe otro elemento diferenciador que es el carácter competitivo de la realización del proyecto. Los alumnos se dividen en grupos de trabajo como en el caso anterior, pero las notas no pueden ser iguales, y se puntúa de forma competitiva. Es decir que los trabajos se van a ordenar por la calidad de los mismos, y a más calidad más nota. Estas evaluaciones se realizan cada semana, y es en una de las sesiones de clase durante las cuales se va a llevar a cabo el trabajo. De esta forma el profesor va a actuar como un cliente que cada semana revisa el trabajo que van haciendo, premiando siempre al mejor. Con este carácter competitivo se espera que la calidad de los trabajos mejore (Schank, 1990), los alumnos se motiven al tener que trabajar por ser los mejores, y poder conseguir el premio (la mejor nota posible). Además el hecho de realizar el trabajo cada semana junto al profesor, permite llevar un seguimiento directo del aprendizaje de los alumnos, y a estos les obliga a trabajar para poder presentar cada semana resultados, ya que tienen un cliente, el profesor, "ansioso" por ver resultados y dar "premios".

Con respecto al trabajo que han tenido que hacer todos los grupos, se trata de un sistema de gestión del aprendizaje (*Learning Management Systems*). Un sistema de tales característica provee(Wiley,2000) una plataforma integrada para el contenido, entrega y gestión del aprendizaje, así como también facilita acceso a un amplio rango de usuarios que pueden incluir participantes, creadores de contenido y administradores, de forma que el sistema de gestión de aprendizaje actúa como un elemento central de un sistema de aprendizaje. Además un LMS debe ser capaz de mantener varios modos de entrega: online, dirigido por instructor, colaborativo, facilitado, no facilitado, autocontrolado o bien al gusto; debe automatizar el proceso del participante respecto a inscripción, registro, historial, transcripción, planificación y reporte, y debe incorporar facilidades de evaluación,

valoración y testeo. Para facilitarles la comprensión del trabajo que debían hacer se les facilitó el código fuente de un LMS de prueba de SCORM (Advanced Distributed Learning Initiative), para que pudieran ver cómo estaba hecho, y la funcionalidad básica de que disponían.

Los conocimientos que se imparten en esta asignatura son funcionalidades avanzadas de Java tales como Servlets, JSPs, RMI o Web Services. Así el objetivo es que el LMS se implementará usando algunas de éstas técnicas, pero siempre bajo la supervisión del profesor que les aconseja que técnica es la mejor para usar cada una de las funcionalidades a realizar.

5. Conclusiones y trabajo futuro.

Las conclusiones que se han obtenido de las experiencias realizadas se pueden agrupar en tres apartados diferentes:

1. Sobre la metodología. En vista a los resultados obtenidos en las asignaturas, creemos que las pautas que proponemos son una buena metodología de trabajo que combina las buenas características del trabajo en equipo, la investigación y las nuevas tecnologías. Además tiene unos beneficios directos sobre los alumnos los cuales aprenden la asignatura sin darse cuenta, incorporan a sus conocimientos una pequeña cultura sobre nuevas tecnologías y se les acostumbra a trabajar en equipo, pudiendo vivir una experiencia similar a la que vivirán en las empresas.
2. Sobre la asignatura de “Estructuras de Datos y de la Información”. Una vez realizados los exámenes parciales, se ha podido comprobar en los resultados un alta tasa de aprobados. Además la motivación del alumnado se ha visto incrementada, afirmaciones argumentadas en los trabajos realizados y en las iniciativas propias de los alumnos. Con respecto a los trabajos se ha podido comprobar que los alumnos han hecho en la mayoría de los casos más de lo que se les pedía, se han preocupado por buscar información en más de un libro, han cuidado el diseño de las páginas Web, y han creado un código HTML puro. Con respecto a las iniciativas propias de los alumnos, algunos de ellos han realizado trabajos personales que han solicitado compartir con el resto de compañeros tales como apuntes, o pequeños programas.
3. Sobre la asignatura de “Programación Avanzada”. A la vista de los trabajos entregados se puede concluir que el factor competencia, es un factor adecuado para un aumento progresivo de la calidad del trabajo realizado. Y concretamente los trabajos que se han entregado muestran que efectivamente los alumnos le han dedicado bastantes horas. Y de igual forma se puede afirmar que estos trabajos son beneficiosos para ellos pues han aprendido a usar las tecnologías sin tener que estudiárselo a nivel abstracto.

Con respecto al trabajo futuro nos proponemos en el próximo curso académico realizar: (a) mejorar las prácticas metodológicas, incluyendo nuevas actividades; (b) en la Universidad Complutense de Madrid, se aplicará a una nueva asignatura de características similares al experimento actual, "Metodología y Tecnología de la Programación" y (c) en la Universidad Carlos III de Madrid se continuará el experimento con la misma asignatura.

6. Referencias bibliográficas

- Advanced Distributed Learning Initiative. Sharable Courseware Object Reference Model (SCORM). <http://www.adlnet.org/>
- Guzdial, M.(1998). Soporte tecnológico para el aprendizaje basado en proyectos. Barcelona: Paidós.
- Ian Sommerville.(2002). Ingeniería del Software. Addison Wesley.
- Marra, R. M. y Jonassen, D.H (2001). Limitations of online courses for supporting constructive learning. Quarterly Review of Distance Education, University of Cambridge. 302-317.
- Marcelo C., Puente D.,Ballesteros M.A, Palazón A.,(2001).ELearning Teleformación. Diseño, Desarrollo y Evaluación de la formación a través de Internet. Editorial Gestión2000.
- Martí Oliet, N., Ortega Mallén, Y., Verdejo López, J. A.(2003), Estructuras de datos y métodos algorítmicos: Ejercicios resueltos; Colección Prentice Practica, Pearson/Prentice may.
- McFarlane, A.(2001). El aprendizaje y las tecnologías de la información. Madrid: Santillana.
- Peña, R. (1998); Diseño de programas. Formalismo y abstracción; Segunda edición. Prentice Hall.
- Schank, R.C. (1990). Teaching architectures. Technical Report. Northwestern University.
- Sugrue, B. (2000). Cognitive approaches to Web-based Instruction. En Lajoie, S. P: Computers as cognitive tools. Vol. II. Hillsdale: Erlbaum
- Tao, P.K. y Gunstone, R.F. (1999). Conceptual change in science learning through collaborative learning at the computer. International journal of Science Education, 21,39-57.
- Taylor, R.(1980). The computer in the school: Tutor, tool, tutee. New York: Teachers College Press.
- WebCT, University British Columbia. <http://www.webct.com/>
- Wiley D.A (2000), Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy <http://www.reusability.org/read>

