

# Actas del Simposio-Taller

# JENU I 2011

XVII Jornadas de Enseñanza  
Universitaria de la Informática

Escuela Técnica Superior de Ingeniería Informática  
Sevilla, 5 al 8 de julio de 2011



AENUI  
Asociación de Enseñantes Universitarios  
de la Informática



Vicerrectorado de Docencia  
Vicerrectorado de Investigación  
Servicio de Extensión Universitaria

Actas del SIMPOSIO-TALLER

previo a las XVII Jornadas de Enseñanza Universitaria de la Informática

Sevilla, Julio 2011

[jenui2011.us.es](http://jenui2011.us.es)

Editores de las Actas:

Antonia Chávez

Francisco Gómez

José Ra. Portillo

Agustín Riscos

Responsables académicos del simposio-taller:

**Joe Miró Julia, Universitat de les Illes Balears**

**Juan José Escribano Otero, Universidad Europea de Madrid**

ISBN: 978-84-694-5440-4

Depósito Legal:

# ÍNDICE

<b>Aprendizaje Basado en Proyectos en Ingeniería Informática. Resultados y reflexiones de seis años de experiencia</b>	<b>1</b>
Enric Martí (Universidad Autónoma de Barcelona), Ferran Poveda (Universidad Autónoma de Barcelona), Antoni Gurguí (Universidad Autónoma de Barcelona), Dèbora Gil (Universidad Autónoma de Barcelona)	
<b>Una experiencia: Introducción a la Matemática Discreta en inglés</b>	<b>9</b>
Félix Gudiel Rodríguez (Universidad de Sevilla)	
<b>LSMaker: un proyecto interdisciplinar</b>	<b>13</b>
David Vernet (Universitat Ramon Llull), Xavi Canaleta (Universitat Ramon Llull), Jordi Planas (Universitat Ramon Llull)	
<b>De las ingenierías informáticas a los nuevos grados. Análisis de los contenidos de las asignaturas destinadas al estudio de la ingeniería del software en los antiguos y nuevos planes de estudio en las universidades de Castilla y León.</b>	<b>21</b>
M <sup>a</sup> Dolores Muñoz Vicente (Universidad de Salamanca), María N. Moreno García (Universidad de Salamanca), Vivian López Batista (Universidad de Salamanca)	
<b>Pickaxe: Herramienta para la docencia de minería de datos</b>	<b>27</b>
Javier Verde Velasco (Universidad de Salamanca), María N. Moreno García (Universidad de Salamanca), Vivian F. López Batista (Universidad de Salamanca), María Dolores Muñoz (Universidad de Salamanca)	
<b>Estudio de métodos para fomentar la auto evaluación del alumnado</b>	<b>35</b>
Antonio Egea (Universitat de les Illes Balears), Antoni Jaume-i-Capó (Universitat de les Illes Balears), Joe Miró (Universitat de les Illes Balears), Carlos Guerrero (Universitat de les Illes Balears)	
<b>Combinando Moodle y tecnologías Web 2.0 para trabajar competencias transversales</b>	<b>43</b>
Adelaida Delgado Domínguez (Universitat de les Illes Balears)	
<b>Proceso de Reubicación de las Competencias Generales y Específicas para el Grado en Ingeniería Informática y Definición de un Plan de Métricas de Evaluación de dichas Competencias</b>	<b>51</b>
L.E. Sánchez (Universidad de Castilla-La Mancha), D.G. Rosado (Universidad de Castilla-La Mancha), D. Mellado (Universidad de Castilla-La Mancha), A. Santos-Olmo (Universidad de Castilla-La Mancha), E. Fernández-Medina (Universidad de Castilla-La Mancha)	
<b>Implantación y Orquestación de Contenidos y Competencias en Seguridad y Auditoría acorde a las Certificaciones Profesionales</b>	<b>59</b>
David G. Rosado (Universidad de Castilla-La Mancha), Luis E. Sánchez (Universidad de Castilla-La Mancha), Daniel Mellado (Universidad de Castilla-La Mancha), Eduardo Fernández-Medina (Universidad de Castilla-La Mancha)	
<b>Aplicación de herramientas de e-Evaluación en los nuevos enfoques evaluativos</b>	<b>67</b>
Leire Urcola Carrera (Universidad del País Vasco), Mariano Barrón Ruiz (Universidad del País Vasco)	
<b>Diseño de las actividades formativas y evaluación de la asignatura Fundamentos del Software</b>	<b>75</b>
J. A. Gómez Hernández (Universidad de Gradada), A. León Salas (Universidad de Gradada), P. Paderewski Rodríguez (Universidad de Gradada)	
<b>Evaluación de entornos de programación para el aprendizaje</b>	<b>83</b>
Sonia Pamplona Roche (Universidad a Distancia de Madrid), Nelson Medinilla Martínez (Universidad Politécnica de Madrid)	
<b>Evaluación del trabajo en grupo: ¿café para todos?</b>	<b>91</b>
Manuel Enciso (Universidad de Málaga), Carlos Rossi (Universidad de Málaga), Eduardo Guzmán (Universidad de Málaga)	
<b>Una aproximación metodológica al cambio de paradigma en el profesional de la informática</b>	<b>99</b>
José Lucas Grillo Lorenzo (Universidad de La Laguna), Francisco de Sande (Universidad de La Laguna), Vicente Blanco (Universidad de La Laguna)	



# Aprendizaje Basado en Proyectos en Ingeniería Informática. Resultados y reflexiones de seis años de experiencia

Enric Martí<sup>1,2</sup> Ferran Poveda<sup>2</sup> Antoni Gurgui<sup>2</sup> Dèbora Gil<sup>2</sup>

<sup>1</sup>Grupo GI-IDES- UAB <sup>2</sup>Departamento Ciencias de la Computación, Escuela de Ingeniería  
Universidad Autónoma de Barcelona

Edificio Q – Campus Bellaterra

08193 Bellaterra (Cerdanyola del Vallès) - BARCELONA

enric.marti@uab.es ferran.poveda@uab.es antoni.gurgui@uab.es.com debora.gil@uab.es

## Resumen

En este taller se presenta una experiencia de 6 años en Aprendizaje Basado en Proyectos (ABP) en Gráficos por Computador, asignatura de Ingeniería Informática de la Universidad Autónoma de Barcelona (UAB). Utilizamos un entorno Moodle adaptado para gestionar la documentación generada en ABP.

La asignatura se estructura mediante la definición de dos itinerarios para cursarla: un itinerario clásico de clase magistral y examen y el otro con metodología ABP.

En el itinerario ABP se explica el funcionamiento en grupos, las tareas de tutorización y seguimiento, así como las pautas de evaluación a los alumnos. Como resultados, aportamos algunos de los trabajos realizados por los alumnos, así como los resultados de las encuestas de valoración realizadas a los alumnos durante estos años. Se expone cómo ha ido evolucionando el funcionamiento de ABP a lo largo de los cursos, tanto en su estructuración como en la evaluación.

El taller pretende ser una reflexión sobre las ventajas e inconvenientes del uso de estas metodologías activas en titulaciones de ingeniería como la informática, abriendo una discusión sobre cómo estructurar la metodología ABP y evaluar los aprendizajes de los alumnos de la forma más idónea.

## Summary

In this workshop a 6 years experience in Project Based Learning (PBL) in Computer Graphics, Computer Engineering course at the Autonomous University of Barcelona (UAB) is presented. We use a Moodle environment suited to manage the documentation generated in PBL. The course is organized by means of two

alternative routes: a classic itinerary of lectures and test-based evaluation and another with PBL. In the PBL itinerary we explain the organization in teamgroups, homework tutoring and monitoring and evaluation guidelines for students. We provide some of the work done by students, and the results of assessment surveys carried out to students during these years. We report the evolution of our PBL itinerary in terms of, both, organization and student surveys.

The workshop aims at discussing about on the advantages and disadvantages of using these active methodologies in technical degrees such as computer engineering, in order to debate about the most suitable way of organizing PBL and assessing students learning rate.

## Palabras clave

Aprendizaje Basado en Proyectos, Project Based Learning, Aprendizaje Cooperativo, Recursos Virtuales, Moodle

## 1. Motivación

La búsqueda de nuevas metodologías docentes en el ámbito de la educación superior es un tema de amplio debate en la universidad. Los cambios experimentados en la sociedad de la información han influido en el alumnado que accede a las universidades. Sin embargo creemos que el perfil de los alumnos no es ni mejor ni peor que hace unos años, es diferente. Actualmente la información es mucho más accesible y existen muchas y variadas vías para poder obtenerla y contrastarla. Por otro lado, la sociedad demanda profesionales con conocimientos pero también con competencias y habilidades específicas y transversales. Todo esto motiva un debate abierto en el profesorado de las universidades en la

búsqueda de nuevas metodologías para transmitir y motivar el aprendizaje a nuestros alumnos, con el objetivo de formar profesionales adaptados a esta nueva sociedad. Todo ello toma relevancia con el mandato que tienen las universidades de adaptar los planes de estudio al futuro Espacio Europeo de Educación Superior (EEES). Una de las metodologías activas es la de Aprendizaje Basado en Proyectos/Problemas (ABP), en inglés *Project/Problem Based Learning (PBL)* [1,4].

Algunos autores hemos acumulado experiencia (unos 20 años) en la impartición de la asignatura de Gráficos por Computador en ingeniería informática. Esta experiencia nos ha permitido detectar distintos ritmos e interés en el aprendizaje de nuestra asignatura por parte de los alumnos, que con el método de clase magistral y examen no lográbamos potenciar las aptitudes de los alumnos con un mayor potencial.

Ello nos motivó a iniciar en el curso 2004-05 la aplicación de la metodología ABP en nuestra asignatura. Para no perjudicar a alumnos menos motivados, la asignatura ofrece dos itinerarios: el de ABP y el clásico, basado en clases magistrales y examen. En [6] se detalla la carga docente para el alumno y para el profesorado para cada uno de los itinerarios. Se pretende que el doble itinerario no suponga un esfuerzo excesivo, ni para el alumno ni para el profesor.

En esta comunicación nos centraremos en explicar el funcionamiento y la experiencia en el itinerario ABP en los últimos 6 años.

El artículo se ha estructurado en 5 partes. En la sección 2 se explica el entorno académico de la asignatura: su organización y contenidos. En la sección 3, se expone la organización ABP a nivel de grupos y seguimiento a los alumnos, mostrando algún ejemplo de enunciado de proyecto con los objetivos de aprendizaje. La sección 4 muestra los criterios de evaluación utilizados, tanto a nivel grupal como individual. La sección 5 muestra como resultados el número de alumnos matriculados y resultados de encuestas de opinión. Finalmente exponemos las conclusiones y algunos temas de discusión.

## 2. Entorno académico

Gráficos por Computador 2 es una asignatura optativa de Ingeniería Informática en la Escuela de Ingeniería (EE) de la Universidad Autónoma

de Barcelona (UAB). Para el alumno, es una asignatura de 3 créditos de teoría, 1'5 de problemas y 1'5 de prácticas, en total 60 (30+15+10) horas con profesor.

Durante los cursos 2004-05 y 2005-06 se impartía en tercero de carrera con 150 alumnos, siendo la optativa con más alumnos. Los cursos 2006-07 al 2010-11 la asignatura pasa a cuarto, con lo que se reduce a 75 alumnos, reduciendo también los recursos de profesorado, lo que obligó a reorganizar la asignatura.

Para cursarla, los alumnos han de haber superado Gráficos por Computador 1 donde aprenden los conceptos básicos del área. En nuestra asignatura se profundiza en temas de modelado 3D (modelos espaciales y fractales), realismo (iluminación, texturas, sombras, color) y animación por computador [3].

Los dos principales objetivos de aprendizaje de la asignatura son:

- conocer, saber elegir y aplicar las tecnologías de modelado, realismo y animación por computador para desarrollar una aplicación gráfica que dé respuesta a un problema o proyecto concreto, y
- obtener indicadores de rendimiento para poder validar la aplicación gráfica y si las decisiones tomadas han sido las adecuadas.

Toda la información y documentos de la asignatura están disponibles vía web [5] y en la plataforma Moodle Caronte [2]. En esta última se gestionan los grupos, las entregas y las encuestas.

## 3. Organización en ABP

Los alumnos que cursan este itinerario forman grupos de 4 a 6 personas. Cada grupo asiste a una sesión de seguimiento con profesor de 2 horas cada 15 días. Estos alumnos, además, asisten una hora por semana a clases de problemas y a las sesiones de prácticas (15 horas repartidas en 6 sesiones de 2'5 horas) en las que desarrollan un entorno gráfico.

Podemos dividir la organización ABP en los siguientes puntos:

1. Primera sesión
2. Enunciados ABP
3. Seguimiento
4. Entregables
5. Prácticas

### 3.1. Primera sesión

Durante la primera semana se forman los grupos de ABP. Decidimos que fueran los propios alumnos quienes formaran los grupos de trabajo, y con ello asuman la responsabilidad de desarrollar el proyecto con compañeros que ellos elijan. Para formar estos grupos se utiliza una aplicación Moodle desarrollada por nosotros que permite que los alumnos se apunten a grupos de trabajo y el primero que se apunta a un grupo pueda bloquearlo mediante contraseña, que dará a sus compañeros, evitando “infiltrados”.

En la primera sesión de seguimiento se proporciona a cada grupo tres enunciados de proyecto, de los cuales deben escoger uno.

Se trata de enunciados cortos (entre 10 y 15 líneas). En la mayoría de ellos se les pide que asuman el rol de profesionales, en los que un cliente les pide el desarrollo de una aplicación gráfica para visualizar e interactuar con ciertos contenidos. La propuesta es genérica, en absoluto detallada. Cada proyecto conlleva unos objetivos de aprendizaje que el profesor pretende que los alumnos descubran y aprendan para resolver el proyecto. Creemos que plantear enunciados en los que los alumnos asumen un rol profesional les motiva en el planteamiento de soluciones imaginativas, y en el compromiso de llevar el proyecto a buen fin.

En algunos casos, el grupo propone un proyecto distinto a los que se le ofrecen. Si el proyecto puede cumplir con los objetivos de aprendizaje de la asignatura se acepta, puesto que aumenta la motivación en el aprendizaje y desarrollo si el proyecto es propuesto por el mismo grupo.

### 3.2. Enunciados ABP

Durante el curso 2010-11 hemos propuesto unos 15 enunciados de proyectos, haciendo combinaciones de 3 para ofrecer a cada grupo. En cursos anteriores se han dado casos en los que dos o tres grupos han elegido el mismo enunciado, pero tanto el desarrollo de su proyecto como los resultados, han sido muy distintos.

En cada enunciado se incluye el texto dado a los alumnos y los objetivos de aprendizaje que se pretenden (información que no se da a los alumnos para no condicionar su trabajo). El profesor, en las reuniones de tutorización

quincenales supervisará los objetivos propuestos por los alumnos, y únicamente los reconducirá en el caso de que los alumnos se desvíen en exceso de los objetivos docentes del proyecto.

Los objetivos de aprendizaje pretenden cubrir de un 70% a un 80% del temario de la asignatura. Se han redactado proyectos sobre juegos, movimiento de robots, juego de ajedrez, atracciones de feria, sistema solar, circuitos de carreras, etc. A continuación mostramos dos ejemplos de proyecto: Sistema solar y cruce.

**Enunciado Cruce:** La autoescuela *El Fittipaldi urbano* os pide hacer un simulador gráfico de conducción que reproduzca de la forma más realista posible el comportamiento físico de un vehículo, incorporando diferentes características físicas (potencia motor, peso, frenos, rozamiento asfalto, etc.) como parámetros y que el resultado de la simulación sea gráfica y numérica.

*Objetivos de aprendizaje:*

- **Visualización 3D:** Definición de puntos de vista (conductor, escenario, etc.).
- **Modelado:** Modelado de la escena urbana, coche, interior del automóvil.
- **Iluminación:** Iluminación de la escena, cielo, iluminación urbana.
- **Movimiento:** Trayectoria de los coches, señalización, tratamiento de colisiones, detección de infracciones de conducción.

**Enunciado Sistema Solar:** El Instituto de Estudios Espaciales de Cataluña quiere disponer de una herramienta de visualización gráfica que permita ver con el mayor grado de realismo posible la posición de los principales planetas y satélites del Sistema Solar, así como su movimiento, de forma que sea posible visualizar su posición en una fecha (día, mes, año, hora minuto, segundo) determinada.

*Objetivos de aprendizaje:*

- **Visualización 3D:** Definición de puntos de vista (sol, planetas, satélites, etc.).
- **Modelado:** Modelado de planetas, satélites (cometas), coordenadas astronómicas, definición de órbitas.
- **Iluminación:** Iluminación de los planetas, texturizar planetas y satélites, anillos de planetas, sombras, eclipses.
- **Movimiento:** Trayectoria de los planetas y satélites, posición según fecha concreta, previsión de eclipses.

- *Otros*: Conceptos de astronomía, sistema solar.

### 3.3. Seguimiento ABP

A partir de la primera sesión, los alumnos asisten a las sesiones tutorizadas de dos horas según calendario. En estas sesiones el grupo discute, trabaja, decide y avanza en la realización del proyecto bajo la supervisión del tutor, que por un lado atiende dudas que puedan surgir a los alumnos y por otro lado se fija en la dinámica de trabajo del grupo y de las distintas aptitudes de cada uno de sus miembros.

Básicamente se observan y valoran las siguientes aptitudes a nivel grupal:

- *Dinámica de grupo*: activo (se discute y se consensua el trabajo a realizar), o pasivo (cada uno trabaja de forma individual haciendo su propia tarea, interactuando poco con el resto del grupo).
- *Calidad del trabajo, aprendizaje grupal*: Si el grupo se plantea retos importantes, busca la información de forma activa y la obtiene, es innovador en el planteamiento de soluciones.

A nivel individual se valoran las siguientes aptitudes:

- *Participación*: si el alumno es activo en la discusión o no.
- *Liderazgo*: si el alumno ejerce o no un cierto liderazgo en el grupo, si sus opiniones son consideradas o no.
- *Puntualidad*: si el alumno es puntual a la llegada y se va a la hora estipulada, no antes.
- *Asistencia*: si el alumno asiste o no a las sesiones tutorizadas.

En cada sesión el tutor va tomando anotaciones sobre estas aptitudes, que serán utilizadas en la evaluación grupal e individual del alumno.

Se incentiva que el grupo haga más reuniones de grupo si lo considera necesario, generando para cada reunión una acta que se envía junto al acta de seguimiento que corresponda. Todas las actas que presenta el grupo son indicadores que benefician el seguimiento del proyecto.

Se tolera la colaboración entre grupos, siempre que el grupo que recibe la ayuda (pequeño programa, información, idea, etc.) mencione en la memoria la ayuda recibida y de

qué grupo. No se pretende que haya competitividad y secretismo entre grupos.

La tarea del tutor no ha de ser resolver los problemas que el grupo tenga, si éstos son muy relacionados con la asignatura. Debe proporcionar fuentes de información, o hacer reflexionar al alumno más que dar la solución. En caso de que la línea de trabajo del grupo se desvíe excesivamente de los objetivos de aprendizaje, debe reconducir su línea de trabajo, pero siempre que sea posible hay que influir lo mínimo en la dinámica de trabajo del grupo, aspecto que no siempre es fácil de equilibrar.

### 3.4. Entregables ABP

Se definen cuatro tipos de entregables:

- Actas de reunión
- Controles
- Presentación del proyecto
- Seguimiento del profesor

#### a) Actas de reunión

En cada sesión tutorizada con profesor el grupo debe escribir un acta donde quede reflejado lo que se ha realizado y acordado en la reunión. El patrón de acta dado a los alumnos contiene cuatro puntos:

1. Aprobación del acta anterior
2. *Discusión*: Seguimiento de los acuerdos definidos en la anterior reunión.
3. *Acuerdos*: Tareas a realizar y quién las realiza.
4. Anexos: Información adicional.

Una vez finalizada la sesión tutorizada, el grupo dispone de tres horas para enviar el acta en formato electrónico a la plataforma Moodle Caronte. Pueden adjuntar actas de reuniones anteriores que haya realizado el grupo sin la asistencia del profesor.

Cada grupo define una numeración propia para las actas. Se proporciona un documento patrón. Se valora la presentación de estas actas dentro de la evaluación grupal.

#### b) Controles

Durante el semestre los alumnos han de entregar dos *controles*, uno en la segunda sesión tutorizada y el otro en la quinta. Estos controles consisten en un planteamiento claro de los objetivos y trabajo a hacer en el proyecto, quién se encarga de cada

parte y una previsión de tiempo de realización. En el primer control se aconseja que el grupo sea ambicioso en sus objetivos, y en el segundo, a pocos días de la defensa pública, que sea más realista. Ambos controles se entregan en formato electrónico.

Tras la entrega de cada control de forma electrónica mediante Moodle, se realiza una encuesta de autoevaluación entre los miembros del grupo. Esta encuesta consiste en que cada miembro del grupo conteste un conjunto de preguntas valorando las aptitudes y la participación de cada uno de sus compañeros de grupo y de él mismo. Estas autoevaluaciones también se realizan de forma electrónica.

Un alumno responde a las siguientes preguntas en referencia al compañero X del grupo (X puede ser él mismo):

- 1) X se implica en el trabajo del grupo. Valoración de 0 (nula) a 10 (máxima).
- 2) X se implica en las decisiones del grupo (0-10).
- 3) X aporta ideas interesantes (0-10)
- 4) X motiva a los compañeros (0-10)
- 5) He aprendido trabajando con X (0-10)
- 6) Volvería a trabajar con X en un grupo (Si-No)
- 7) Comentarios sobre X

#### c) Presentación del proyecto

En la última sesión del curso los grupos se juntan para la defensa pública del proyecto. Además entregan la siguiente documentación:

- *Portafolio*. Memoria que recoge los objetivos, trabajo realizado, información consultada y un manual de la aplicación desarrollada. Existe un patrón de documento con indicaciones.
- *Presentación*. Documento de transparencias (máximo 12) para la presentación. Para ello se proporciona un patrón con indicaciones.
- *Aplicación informática* que da respuesta al proyecto. Se entregan todos los ficheros fuente y se proporciona una demo.

Toda esta documentación se entrega en formato electrónico y el portafolio, además, en papel. Se forma un tribunal de tres profesores que valora la calidad del trabajo y la presentación mediante un cuestionario. Esta valoración será importante en la evaluación grupal.

Al final de las presentaciones se pide a cada grupo que valore, no con puntos sino a modo de

ranking, el trabajo que más le ha gustado y el que menos, excepto el suyo. Esta valoración se utilizará en la evaluación grupal.

Tras la defensa del proyecto se abre los siguientes 10 días una última encuesta de autoevaluación entre los miembros del grupo. En total, disponemos de tres encuestas de autoevaluación grupal.

#### d) Seguimiento del profesor

Después de cada control, el profesor rellena una encuesta de seguimiento para cada grupo, con el objetivo que los alumnos conozcan la opinión del tutor. Esta encuesta es rellenada también electrónicamente por el profesor mediante una aplicación de Moodle, y visible para los alumnos. El cuestionario contiene las siguientes valoraciones sobre el grupo:

- 1) Asistencia a las sesiones tutorizadas (suficiente-Insuficiente). Incluye retrasos o marcharse antes de finalizar la sesión.
- 2) Comentarios. Aclaraciones de la valoración anterior.
- 3) Dinámica de trabajo en grupo (Correcta-Suficiente-Insuficiente)
- 4) Comentarios
- 5) Actas de las reuniones (Correctas-Suficientes-Insuficientes)
- 6) Comentarios
- 7) Controles. Presentación (Correctos-Suficientes-Insuficientes)
- 8) Comentarios
- 9) Controles. Trabajo (Correcto-Suficiente-Insuficiente). Valoración si la carga de trabajo es acorde al número de miembros del grupo.
- 10) Comentarios

Este seguimiento es utilizado en la evaluación grupal.

### 3.5. Prácticas

La asignatura programa 6 sesiones de prácticas de 2'5 horas cada una en las que se diseña un entorno gráfico partiendo de cero, al que se le van añadiendo funcionalidades. Las dos primeras sesiones son obligatorias y no evaluables para los grupos ABP. Las siguientes sesiones son recomendadas, puesto que les pueden ayudar a incluir funcionalidades importantes en su aplicación así como optimizaciones. La asistencia y entrega de los trabajos de las 4 últimas sesiones

mejoran la nota individual de los alumnos que cursen ABP.

Se propone definir un entorno de multiplataforma (Windows, Linux o MacOS) programado en C++ y basado en las bibliotecas OpenGL (*Open Graphics Library*) [8] para el desarrollo gráfico y Qt [7] para la creación de interfaces auxiliares. Los enunciados de cada sesión son los siguientes:

- 1) Creación de un entorno de visualización OpenGL
- 2) Principios de modelado y animación en OpenGL
- 3) Iluminación y materiales en OpenGL
- 4) Programación básica de *shaders* en OpenGL
- 5) WebGL y OpenGL ES
- 6) Realidad Aumentada con ARToolkit

Antes de cada sesión, los alumnos disponen de un enunciado para preparar el trabajo de la sesión. Después de la sesión de prácticas, los alumnos entregan el trabajo, respondiendo algunas preguntas sobre el mismo.

## 5. Evaluación ABP

La nota final de la asignatura se obtiene a partir de la *evaluación grupal*, de la actividad realizada por el grupo y la *evaluación individual*, a partir de indicadores obtenidos en las sesiones tutorizadas. Se conceden bonificaciones tanto grupales como individuales que se añaden a la nota.

### 5.1. Evaluación grupal

Corresponde un 70% de la nota final. Los indicadores de esta fase se obtienen fundamentalmente en la presentación del proyecto. Se valoran los siguientes indicadores:

- Complejidad, innovación y riesgo del proyecto (1 pto.)
- Funcionalidades de la aplicación (1 pto.)
- Diseño de la interfície de usuario (1 pto.)
- Calidad de la aplicación. Validación (1 pto.)
- Transparencias. Claridad y presentación (1 pto.)
- Memoria. Claridad y presentación (1 pto.)
- Actas. Claridad y presentación (1 pto.)
- Bonificaciones (máx. 1 pto.): Si han ayudado a otros grupos, aprendizaje importante de temas no relacionados en la asignatura, etc.

### 5.2. Evaluación individual

Corresponde a un 30% de la nota final. La evaluación individual se basa en la observación de las sesiones tutorizadas de dos horas cada quince días. En estas sesiones se admiten un máximo de entre 25-28 alumnos en 5 grupos para poder atender dudas y realizar observaciones en los grupos. Es importante en la tarea de tutorización darles amplia libertad en su aprendizaje y toma de decisiones, siempre que cumplan mínimamente los objetivos de aprendizaje planteados en la asignatura.

Los indicadores de evaluación individual que utilizamos en las sesiones son:

- Asistencia y puntualidad (1 pto.)
- Actitud en el grupo: participativo-pasivo (1 pto.)
- Liderazgo y responsabilidad (1 pto.)
- Bonificaciones:
  - Ponente en la presentación del proyecto (0'5 ptos.)
  - Buena valoración de los compañeros en la autoevaluación (0'5 ptos.)
  - Trabajo realizado en prácticas (0'5 ptos.)

Por último destacar que a los alumnos no se les da todos los indicadores de evaluación ni su cuantificación en la nota final, para no condicionar su trabajo. Genéricamente a principios de curso se les indica en la guía docente que hay una evaluación grupal y una evaluación individual basada en indicadores.

El hacer pública la evaluación a los alumnos es un tema de amplia controversia en ABP, donde hay opiniones favorables y desfavorables. Nuestra decisión va en la línea de condicionar lo mínimo al alumno en su trabajo, potenciando un ambiente de trabajo lo más real y profesional posible.

## 6. Resultados

Como resultados se muestran dos ejemplos de los proyectos Cruce y Sistema Solar realizados el curso 2010-11 (figuras 1 y 2). En ambos se puede apreciar la calidad de la visualización gracias al uso de la librería OpenGL que los alumnos aprenden a utilizar.



Figura 1. Ejemplo del proyecto cruce



Figura 2. Ejemplo de proyecto Sistema Solar.

En la *tabla 1* se muestra la evolución de la matrícula en la asignatura y el número de alumnos que escoge ABP. El número de alumnos ABP que finalmente (última columna) presentan el proyecto es superior al 80%. En la matrícula se aprecia el descenso de alumnos el curso 2006-07 al pasar la asignatura de tercero a cuarto.

Curso	Al. Mat.	# TPPE	# ABP	# FAB
2001-02	152			
2002-03	147	-	-	-
2003-04	150	142	8	8
2004-05	148	88	60	40
2005-06	155	60	95	76
2006-07	65	40	25	25
2007-08	76	42	34	32
2008-09	81	30	51	47
2009-10	68	22	46	40
2010-11	67	17	50	49

Tabla 1. Número de alumnos matriculados y los que han escogido TPPE (#TPPE), los de ABP (# ABP) y los que han finalizado ABP (#FAB).

La tendencia negativa de los últimos años es fruto del descenso de alumnos en la carrera. La asignatura se mantiene entre las 5 optativas de la carrera con más alumnos.

Curso	Tut. Prof	Met.	Val.	#
2001-02	7,9	6,4	6,7	88
2002-03	7,1	6,8	6,9	116
2003-04	8,0	6,9	7,2	91
2004-05 (ABP)	7,6	8,0	8,2	46
2005-06 (ABP)	8,06	8,1	8,35	63
2006-07 (ABP)	8,3	8,35	8,6	20
2007-08 (ABP)	8,04	8,28	8,8	25
2008-09 (ABP)	7,17	8,3	8,44	40
2009-10 (ABP)	8,13	8,41	8,55	29

Tabla 2. Resultados de encuestas a alumnos, sobre 10. (Tut. Prof: Tutorización profesor, Met: Metodología, Val.: Valoración global, #: número de muestras).

Cada año se han realizado encuestas a los alumnos para que valoren su aprendizaje. En los primeros cursos, las encuestas se realizaban minutos antes de empezar el examen, lo que nos daba un número de muestras significativo, pero también hay que tener en cuenta la desviación de los resultados producidos por los nervios del día del examen.

A partir del curso 2005-06 el alumno responde las encuestas de forma electrónica con Caronte [2], pudiendo éste contestar de forma anónima.

Para los alumnos del itinerario TPPE se les hizo la misma encuesta de otros años, mientras que a los alumnos de ABP les hemos pasado una encuesta diferente, con preguntas comunes. Los resultados los mostramos en la *tabla 2*. Se puede ver que los alumnos de ABP valoran positivamente la metodología con puntuaciones superiores a 8, siendo en todos los años mejor valorada que por los alumnos del itinerario TPPE.

Como dato menos tangible, destacar que el número de alumnos a los que hemos dirigido el proyecto de fin de carrera en temas de gráficos por computador ha crecido en los últimos años.

## 7. Conclusiones y discusión

De la experiencia hemos extraído las siguientes conclusiones y temas de discusión:

- Valoramos la oferta de ABP como positiva, pues ha potenciado la iniciativa de los alumnos y ellos también lo han valorado así.

- Durante el curso, los alumnos de ABP han visto el profesor como una figura positiva. Creemos que esta dinámica de clase es más gratificante para el docente.
- Pocos grupos ABP se han disuelto. Sus miembros se han incorporado al itinerario TPPE, recuperando el ritmo con facilidad.
- Al principio los alumnos se ven un poco perdidos en lo que deben hacer, pero en las dos primeras sesiones tutorizadas van tomando iniciativa en su aprendizaje.
- Los alumnos presentan dificultades en escribir las actas, en explicar lo que han discutido y han acordado. Las actas son demasiado resumidas, casi telegráficas, y esto va en detrimento de su nota, al no reflejar el trabajo realizado en las reuniones.
- Ha habido grupos que como alternativa a nuestros proyectos han propuesto ellos uno (usualmente un videojuego). En estos grupos, la motivación y la responsabilidad en la realización del proyecto es mucho más alta, lo que repercute en un buen trabajo. Hay grupos que han enviado mejoras del proyecto después de haberles puesto la nota.
- Estamos estudiando para el próximo año asignar responsables de tareas en el proyecto. Un proyecto real así lo tiene.
- El sistema de evaluación actual califica, a nuestro entender, los conocimientos del alumno, pero no competencias, habilidades o aptitudes trabajadas en ABP. Creemos que debería constar en el expediente una valoración de estas aptitudes.

Obviamente, creemos que esta experiencia no es extrapolable a todas las asignaturas ni a todas las titulaciones. El docente debe conocer metodologías y experiencias siendo su tarea valorar cuáles son las más convenientes para su asignatura y cómo adaptarlas. No hay una metodología única y óptima para todo.

En nuestro caso, creemos que la experiencia continua siendo positiva y enriquecedora, tanto para los alumnos como para el profesorado.

### Agradecimientos

Agradecer a los alumnos **Alex Au bets, Daniel Barea, Jaume Bigas, Javier de Muga, Albert Sapé**, alumnos de la asignatura en el curso 2004-05 por la imagen de su trabajo ABP del Cruce.

Agradecer a los alumnos **Francisco Cobo, Gerard Gilabert, Oscar Guillén y Victor Soler**, alumnos de la asignatura en el curso 2010-11 por la imagen de su trabajo ABP del Sistema Solar.

Queremos agradecer al **Dr. Luis Branda** y al **Dr. Antoni Font**, que nos introdujeron a la metodología ABP y a los miembros del grupo de interés ABP del IDES de la UAB. Sus ideas, y sugerencias inspiraron este trabajo.

Este proyecto ha sido subvencionado por la convocatoria de la Agencia para la Calidad Universitaria (AGAUR) de ayudas para la financiación de proyectos para la mejora de la calidad docente de las universidades catalanas del año 2005 (2005MQD 00246), la convocatoria de la DGU del Ministerio de Educación y Ciencia el año 2007 (EA2007 0286).

### Referencias

- [1] Bigelow, J. *Using problem based learning to develop skills in solving unstructured problems*. Journal of Management Education, 28(5):591-610, 2004.
- [2] <http://caronte.uab.es>, Plataforma Moodle de soporte virtual a la docencia.
- [3] Foley, J.D., van Dam, A., Feiner, S.K., Hughes, J.F. *Introduction to Computer Graphics*, Addison Wesley, 1993.
- [4] Font, A. Branda L. El aprendizaje por problemas y el espacio europeo de educación superior, Plan de formación docente 2003/04, Unidad de Innovación Docente en Educación Superior (IDES-UAB), Enero 2004.
- [5] <http://dcc.uab.es/teach/a25011/c25011.htm>, página web de la asignatura Gráficos por Computador 2 (último acceso: 8 junio 2011).
- [6] Martí, E., Gil D., Julià C., Vivet M., Julià C., *Aprendizaje Basado en Proyectos en la asignatura de Gráficos por Computador en Ingeniería Informática. Balance de cuatro años de experiencia*, XV Jornadas de Enseñanza Universitaria de la Informática (JENUI 09), 1:387-394, ISBN:978-84-692-2758-9, Barcelona, Julio 2009.
- [7] <http://qt.nokia.com/products/>, web con completa documentación de la interface Qt (último acceso: 8 junio 2011)
- [8] Shreiner, D. Woo, M., Neider, J., Davis, T., *OpenGL Programming Guide, 4<sup>th</sup> edition*, Addison Wesley, 2004.

# Una experiencia: Introducción a la Matemática Discreta en inglés

Félix Gudiel Rodríguez  
Departamento Matemática Aplicada I  
Escuela Técnica Superior de Ingeniería Informática  
Universidad de Sevilla  
Avda. Reina Mercedes s/n  
41012 Sevilla  
gudiel@us.es

## Resumen

En esta comunicación se describe la experiencia desarrollada en la asignatura *Introducción a la Matemática Discreta*, con un grupo con docencia en inglés, dentro de los estudios de Grado en Ingeniería Informática. Durante el curso 2010-2011, las asignaturas de primer curso de las nuevas titulaciones de Grado se están impartiendo en la Escuela Técnica Superior de Ingeniería Informática tanto en español como en inglés, dando la posibilidad a los alumnos de elegir en qué grupos se matriculan. Esta iniciativa persigue una mejor preparación del alumno de cara a su futuro profesional, siguiendo las directrices del EEES.

## Summary

This communication describes the experience held teaching *Introduction to Discrete Mathematics* in english for the students of the Bachelor's degree in Computer Engineering. First year subjects are taught in english during the year 2010-2011. This initiative aims to better prepare the student for his professional future, following European Higher Education Area (EHEA) guidelines.

## Palabras clave

matemáticas, inglés, EEES.

## 1. Motivación

El conocimiento de una segunda lengua, y en el caso particular de los estudios de ingeniería informática, de la lengua inglesa, permite adquirir una mayor capacitación laboral y aporta un mayor nivel de competitividad para nuestros estudiantes. Por otra parte, al encontrarnos plenamente inmersos en el Espacio

Europeo de Educación Superior, el dominio de la lengua inglesa se hace imprescindible como medio básico de comunicación intercomunitario. El estudio y perfeccionamiento de la lengua inglesa en los primeros cursos de las titulaciones, facilitará a nuestros alumnos el acceso a estudios en universidades extranjeras acogiendo a becas y convenios internacionales de dobles titulaciones.

Por otra parte, según la normativa vigente, todo estudiante de la Universidad de Sevilla debe acreditar antes de finalizar sus estudios de Grado, un nivel de conocimiento y destreza con la lengua inglesa equivalente, como mínimo, al nivel B1 del Marco Común Europeo de Referencia de las Lenguas (MCERL), que corresponde a un nivel de usuario independiente en una lengua extranjera.

La Escuela Técnica Superior de Ingeniería Informática quiere ofertar a sus alumnos la posibilidad de conseguir este valor añadido, acercándolos al conocimiento de la terminología y el lenguaje propios de la ingeniería informática en lengua inglesa, mientras paralelamente cursan los estudios de Grado y consiguen un perfeccionamiento del idioma inglés. Se apuesta por la posibilidad de desarrollar el conocimiento de la lengua inglesa de forma transversal a lo largo de los diferentes cursos promoviendo el desarrollo de itinerarios curriculares con asignaturas impartidas en inglés.

Con la puesta en marcha de las nuevas titulaciones de Grado en Ingeniería Informática - Ingeniería del Software, Grado en Ingeniería Informática - Ingeniería de Computadores y Grado en Ingeniería Informática - Tecnologías Informáticas, para el curso 2010-2011 y dado que en primer curso las materias se corresponden con 60 créditos de formación básica, comunes a las tres titulaciones, la oferta de la E.T.S.I.I. para impartir docencia en lengua inglesa se concreta en un grupo extra, común para las tres titulaciones. La superación por parte del alumno de cualquier asignatura en este grupo supone el reco-

nocimiento automático de la misma para cualquiera de los tres grados indistintamente. En dicho grupo se han ofertado 36 plazas en total para el grupo de teoría, lo que supone contar con un grupo más reducido que puede redundar en un mejor seguimiento personal del alumno. La división en subgrupos prácticos y de laboratorio será similar a la de los otros grupos en lengua castellana, en función del modelo de docencia de cada asignatura.

Aprovechando esta coyuntura, se ha enfocado la asignatura de forma que el alumno tenga un papel más participativo, que es uno de los principales objetivos del EEES, de forma que pueda mejorar su capacidad de expresarse en dicho idioma. Todas las actividades se han llevado a cabo utilizando el inglés como idioma, lo que ha hecho que los alumnos progresen en su nivel de dominio de dicho idioma, además de desarrollar otro tipo de competencias (como la capacidad de exposición en público, trabajo en grupo, ...) Este enfoque de la asignatura (cuyos contenidos históricamente han resultado difíciles para el alumnado) ha logrado motivar al alumno, que ha pasado a ser el protagonista de las clases. Además, el hecho de que el grupo con docencia en inglés sea más reducido de lo habitual ha permitido hacer un seguimiento más personalizado del aprendizaje del alumno, obteniéndose finalmente buenos resultados tanto a nivel de calificaciones, como de dominio del idioma extranjero en lo que se refiere a la terminología propia de la asignatura.

## 2. La materia

La asignatura en cuestión es "Introducción a la Matemática Discreta", correspondiente al primer cuatrimestre del primer curso del Grado en Ingeniería Informática. Como dicha asignatura se imparte en las tres titulaciones de grado de la Escuela Superior de Informática de la Universidad de Sevilla, el grupo de inglés comprende a alumnos procedentes de todas ellas.

El contenido de la asignatura comprende rudimentos de Matemáticas, Algoritmos, Lógica proposicional y Álgebras de Boole, Combinatoria, Recurrencias, Aritmética entera y Aritmética modular, y está diseñado para familiarizar a los alumnos con principios y técnicas matemáticas de amplio uso en Ingeniería Informática.

La asignatura es especialmente difícil para el

alumnado debido a una serie de circunstancias, a la propia dificultad intrínseca del lenguaje matemático se le une el choque del cambio de etapa educativa, a una enseñanza superior universitaria, y una cierta animadversión a la materia con el argumento de su "presunta" inutilidad. Para contrarrestar esto, la orientación de la asignatura es eminentemente práctica y dirigida a los conceptos que les van a ser de utilidad para sus estudios.

## 3. El grupo

Este año, primero de implantación del Grado y, consecuentemente, primero de impartición de este grupo en inglés, el número de alumnos ha sido reducido, concretamente un total de 9 alumnos se han matriculado en la asignatura.

La procedencia de los alumnos es la siguiente: uno procede del Grado en Ingeniería Informática - Ingeniería de Computadores, cuatro del Grado en Ingeniería Informática - Ingeniería del Software, y otros cuatro del Grado en Ingeniería Informática - Tecnologías Informáticas.

## 4. Las clases

Las clases se han dividido en teórico-prácticas y clases de laboratorio, repartidas en 28 horas de clases teóricas, 14 horas de clases prácticas y 12 horas de clases de laboratorio, además de dedicar 4 horas a exámenes para la evaluación continua y 2 más al examen de laboratorio, lo que comprende las 60 horas de trabajo presencial del alumno (a las que se le deben añadir otras 90 horas de trabajo no presencial hasta completar las 150 horas de trabajo correspondientes a una asignatura de 6 créditos ECTS).

En las clases teórico-prácticas, que se han desarrollado en el aula, se ha procedido a la exposición por parte del profesor de los resultados teóricos estudiados, seguidos de ejemplos ilustrativos de los mismos, así como a la resolución de problemas procedentes del boletín, de exámenes de la antigua asignatura y de ejercicios correspondientes al trabajo personal que han sido considerados especialmente interesantes (especialmente los que han presentado mayor dificultad a la hora de ser resueltos por cada alumno independientemente).

En cuanto a las clases de laboratorio, se han de-

dicado a incidir especialmente en aspectos de la asignatura que son especialmente tediosos o largos para resolver “a mano”, ayudados por un guión-cuestionario para cada sesión, que había que responder utilizando los programas disponibles en los laboratorios, entre los que cabe destacar el programa *DisMat: Experiencias Interactivas en Matemática Discreta* realizado por el profesor Gerardo Vaileiras, del Departamento de Matemática Aplicada I de la Universidad de Sevilla, y que se adapta perfectamente al temario de la asignatura. Dicho programa está disponible para libre descarga por parte del alumnado.

### 5. Trabajo personal

Al término de cada uno de los bloques del temario, un cuestionario ha sido enviado a los alumnos, para ser resuelto y entregado. Dicho cuestionario consta de problemas que refuerzan lo estudiado en clase y sirven como autoevaluación para cada alumno, así como de otros cuya dificultad es algo superior y dan pie a una profundización mayor sobre algún aspecto de la asignatura.

### 6. Recursos

En la página web de la asignatura, los alumnos han dispuesto de un boletín (*problems*) con problemas propuestos (de entre los cuales aproximadamente la mitad son resueltos en clase), los cuestionarios para las prácticas de laboratorio, los problemas propuestos como trabajo personal (*homework*), y un guión de la asignatura (*handouts*) junto con la temporización de la misma.

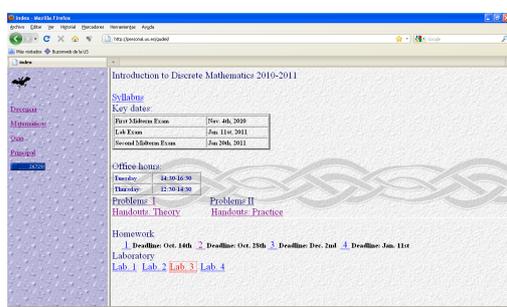


Figura 1: La página web de la asignatura.

### 7. La evaluación

La evaluación continua en esta asignatura estaba prevista de acuerdo al siguiente patrón: un examen escrito durante la sexta semana de curso, con un valor de un 30%, un examen de laboratorio en la semana undécima, con un peso del 20% de la calificación final, y un examen escrito el último día de clase, de peso 50% en la calificación final.

Para aquellos alumnos que no superen la asignatura mediante la evaluación continua, existe la convocatoria de febrero, que constará de un examen escrito calificado sobre 8 puntos y uno de laboratorio de peso 2 puntos (aunque se permite al alumno retener la nota del examen de laboratorio realizado durante el curso, si así lo desea).

### 8. Autoayuda

Aunque no estaba previsto inicialmente, el hecho de que el número de alumnos ha sido pequeño, ha permitido que se hayan organizado por su cuenta, creando un espacio de trabajo común, en el que han colgado material que les ha parecido interesante.

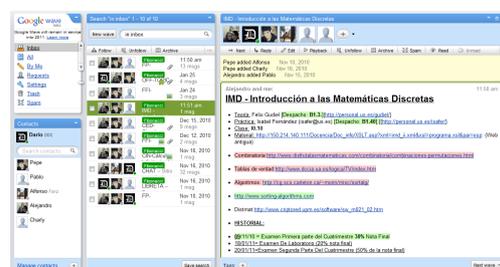


Figura 2: El espacio de trabajo creado por los alumnos.

### 9. Expectativas de los alumnos y percepción del resultado

Preguntados los alumnos por sus expectativas a la hora de escoger la enseñanza en inglés, manifiestan que contemplan ésta como una buena oportunidad para mejorar sus conocimientos y manejo del idioma extranjero (de hecho aunque la Escuela permitía matricularse en algunas asignaturas en inglés y otras en castellano, todos los alumnos de Introducción a la

Matemática Discreta han escogido docencia en inglés para todas las asignaturas del primer cuatrimestre.

Por otra parte, como era esperable, siguen percibiendo la asignatura como difícil para ellos, aunque afirman que volverían a matricularse en el grupo de inglés nuevamente.

Encuestados al final del cuatrimestre sobre su valoración de 0 (poca importancia) a 3 (mucha importancia), evalúan los siguientes aspectos de la asignatura:

- Buen ambiente entre compañeros: 3.
- Buen trato con los profesores: 3.
- Aprender también competencias transversales como: trabajar en grupo, exponer en público, planificar mi tiempo, etc.: 2,25
- Que tenga mucho contenido práctico y poco teórico: 2.
- Que pueda aprender conceptos aplicándolos en lugar de memorizándolos: 2,25
- Que me guíen en mis estudios y en saber moverme por la Universidad: 1,75.
- Que el número de alumnos por profesor sea bajo: 1,25.

## 10. Resultados y conclusiones

Aunque el curso aún no ha terminado, un total de 6 de 9 alumnos han superado la asignatura tras la convocatoria de febrero, de los cuales 5 lo hicieron durante la evaluación continua, con lo que los resultados han sido buenos (por encima de los obtenidos en los grupos impartidos en castellano). Por ello, y

además por la propia percepción del profesor de la asignatura, cabe pensar que el hecho de que la docencia se haya desarrollado en inglés no ha supuesto ningún hándicap para el alumnado, sino todo lo contrario; le ha permitido comunicarse y expresarse tanto de manera oral como escrita en inglés demostrando así su competencia lingüística. Sin embargo el alumnado se ha beneficiado claramente del reducido tamaño del grupo, hecho que ha influido también positivamente en el profesor, puesto que le ha permitido individualizar la enseñanza. Se concluye, pues, que la lengua vehicular no ha sido un factor decisivo en el desarrollo de la docencia de esta asignatura.

Para posteriores años, cabría esperar un aumento del número de alumnos que escogen la opción de recibir las clases en inglés, una vez que la experiencia se ha demostrado positiva. De todas formas, el número de alumnos por grupo no debiera ser mayor de 20, de forma que la enseñanza y la evaluación pudieran ser llevadas a cabo de manera individualizada.



Figura 3: Resultados en la convocatoria de febrero.

# LSMaker: un proyecto interdisciplinar

David Vernet, Xavi Canaleta y Jordi Planas

Departamento de Informática  
La Salle, Universitat Ramon Llull  
Sant Joan de la Salle 42  
08022 Barcelona  
{dave,xavic,jordipl}@salleurl.edu

## Resumen

En el presente artículo presentamos un nuevo recurso docente implantado este curso en nuestra facultad. Se trata de un robot autónomo sobre el cual se realizan las prácticas de la mayoría de asignaturas de primer curso. Su carácter interdisciplinar supone una motivación extra para el alumnado pues proporciona un entorno común donde poner en práctica los conocimientos adquiridos en cada asignatura. El proyecto tendrá continuidad y se irá implantando paulatinamente en el resto de cursos, acompañando así al alumno a lo largo de su formación como ingeniero.

Nos centraremos en cómo se han transformado las prácticas de Programación e Introducción a los Ordenadores, adaptándolas al nuevo entorno y mostrando la repercusión de este cambio en alumnos y profesores. También indicaremos cómo las prácticas planteadas en esta asignatura se vinculan de manera indirecta con el resto de materias del curso, creando así un proyecto común para toda la facultad.

## Summary

In this article we present a new teaching resource which we have implanted into our faculty this academic year. It is an autonomous robot on which the practical work of almost all the core first-year subjects can be carried out. Its interdisciplinary form is an additional motivation factor for the students as it gives them the opportunity to put into practice much of the theoretical knowledge gained on each course. This initiative will continue and the robot will be incorporated into the rest of the degree courses and will therefore accompany the student through his/her training as an engineer.

We will focus on the incorporation of the robot in the practical part of the Programming

course and how this has transformed the practical activities as well as studying its repercussions on students and teaching staff. We will also show how the practical work set out in this subject ties in with the rest of the first year subjects, thus creating a common project for the whole faculty.

## Palabras clave

Interdisciplinar, robótica, programación, recurso

## 1. Motivación

Con el objetivo de evitar la falta de entusiasmo que se ha venido observando en los últimos años en los estudios de ingeniería, a finales del curso 2009-10 se decidió crear un proyecto que motivara al alumno a matricularse de los estudios de Informática y a la vez supusiera un recurso sobre el que pudiera practicar los conocimientos que iba adquiriendo a lo largo de su carrera. Así nació el robot *LSMaker*, un robot autónomo, programable y fácilmente ampliable a nivel electrónico.

Con el lema “el robot que crece contigo” se intentaba de esta forma crear un proyecto interdisciplinar, común a la mayoría de las asignaturas de los estudios de Informática, mediante el cual los alumnos podían transferir de manera real los nuevos conocimientos, haciendo evolucionar las prestaciones del robot a medida que avanzaban en los estudios.

Indirectamente, el profesorado también se tenía que sentir motivado con el nuevo recurso pues se le planteaba la posibilidad de reformar sus prácticas y adaptarlas en parte al nuevo entorno. En este curso sólo el profesorado de primer curso estaba implicado, pues la adquisición del robot por parte del alumno se realiza en la primera matrícula de los estudios. Así pues, el objetivo es que se vaya instaurando paulatinamente a lo largo

de los siguientes cursos hasta conseguir su incorporación en todos los niveles.

## 2. Descripción del robot

En este apartado se describirán las principales características del robot. Primero nos centraremos en sus componentes físicos, para luego pasar a especificar las diferentes modalidades de uso y los requisitos necesarios en cada caso.

### 2.1 Descripción física

El proyecto LSMaker está compuesto por dos elementos diferenciados. Por un lado, el robot propiamente dicho, es decir, la plataforma móvil. Por otro lado se encuentra el mando a distancia que actúa como control remoto del robot mediante ondas de radiofrecuencia.



Figura 1. Plataforma móvil del LSMaker

Todos los componentes del robot se alimentan por seis unidades AA en serie. De entre los elementos hardware que forman el robot, el cual se puede observar en la Figura 1, el más importante de ellos es su microprocesador *PIC* de 16 bits, dispositivo que controla el funcionamiento del resto de elementos. Entre las funcionalidades controladas por el procesador destacan el movimiento de los motores tractores (que se traduce en movimiento del robot gracias a sus dos orugas) el control de la pantalla LCD, la recogida de información del acelerómetro, la comunicación vía radio con el mando y la comunicación mediante puerto USB. Estas no son, sin embargo, las únicas funcionalidades que puede llegar a ofrecer el robot LSMaker, puesto que la placa

base del dispositivo cuenta con distintos conectores de expansión debidamente diseñados para disminuir los efectos de cortocircuitos y polarizaciones incorrectas en ellos. Será en estos puertos de expansión donde los alumnos podrán conectar los nuevos elementos hardware que realicen durante la carrera.

El control remoto se puede alimentar ya sea mediante dos unidades AA o a través del cable USB.



Figura 2. El mando a distancia

Este mando a distancia (véase figura 2) se encarga de enviar ondas de radio al robot a un máximo de 12 metros en interiores. Las órdenes que envía el control remoto se pueden originar de dos maneras distintas. Por un lado se puede trabajar con el mando a modo de control remoto de la plataforma móvil. Para tal fin el usuario debe accionar el pulsador y mover la placa como si de un *joystick* se tratara. Por otro lado, cuando el control remoto se conecta al PC mediante el cable USB, este puede actuar de puente entre aplicaciones de ordenador y el robot, reenviando las instrucciones indicadas por las aplicaciones al dispositivo móvil.

### 2.2 Modos de uso

La plataforma del LSMaker se puede mover en base a tres casuísticas diferentes. La primera y más básica de las causas es el modo de control teledirigido y manual, explicado anteriormente, en el que el usuario hace mover el robot usando el control como un *joystick*.

Los otros dos modos de trabajo se basan en la programación de aplicaciones que interactúen con el procesador del robot, ya sea directa o indirectamente. A tal efecto se ha creado una librería de operaciones y funcionalidades (API) en lenguaje C que facilita el trabajo con los distintos

elementos del robot a través del microprocesador. Así pues, dada esta librería de funcionalidades, se pueden crear aplicaciones para el robot de dos maneras distintas:

El primer modo de trabajo automático del robot se basa en programar directamente su microprocesador PIC usando la API facilitada, de manera que el código resida y se ejecute en el robot.

El segundo modo de trabajo programado consiste en realizar aplicaciones nativas de PC que utilicen los recursos ofrecidos en una adaptación de la API, de manera que las llamadas a estas operaciones se redirijan al mando a distancia (debidamente conectado al PC) y éste las transmita vía radio al robot para que las ejecute. A la adaptación de la API se la conoce con el nombre de “API remota” y se desmarca de la API del robot por modificar el comportamiento de algunas operaciones para que trabajen sobre el PC en lugar de sobre el robot. Esta característica conlleva ventajas, como la mejora en la capacidad de cálculo, pero también inconvenientes, como los problemas por interferencias ambientales o las limitaciones de distancia física entre el robot y el mando.

### 2.3 Requerimientos y utilidades

Para poder programar el robot LSMaker es necesario disponer de todo el material hardware que se ha ido citando con anterioridad, así como del código fuente de la API remota y/o la API general, un conjunto de software y las documentaciones y manuales de uso del robot.

Referente al software necesario, por una parte se requieren entornos de desarrollo integrados (IDE) para programar las aplicaciones. Los entornos escogidos para esta tarea son el MPLAB IDE v8.36 [1] para desarrollar con la API general y el Dev-C++ [2] para aplicaciones que usen la API remota. Por otro lado, para comunicar el ordenador con el control remoto (al usar la API remota) o con el robot (para cargar en él las aplicaciones realizadas, calibrar sus motores, etc.) se requiere el uso de la aplicación LSLoader (Figura 3), creada específicamente para este proyecto por parte de la facultad. Entre otras utilidades, esta aplicación contiene una consola para permitir comunicación entre el robot y el usuario de la aplicación, siempre que el primero

esté conectado al PC con el cable USB. Además, es gracias a esta aplicación que el usuario puede cargar o descargar los ficheros de texto internos del robot.

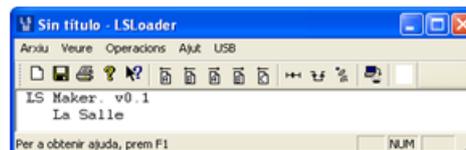


Figura 3. Aplicación LSLoader

Por último, para tener todo el conocimiento básico sobre instalación del software y su uso, así como de las distintas funcionalidades del robot, existe una documentación extendida, así como distintos manuales técnicos.

## 3. Uso del robot en primer curso

Vamos a comentar a continuación las implicaciones que tiene el robot tanto a nivel de curso como a nivel de la asignatura de Programación.

### 3.1 El robot en el primer curso

La introducción del proyecto LSMaker en La Salle ha supuesto la necesidad de adaptar el contenido de las distintas asignaturas impartidas en primer curso, sobre todo en cuanto al material de prácticas se refiere [3]. De esta manera distintas asignaturas han tenido que diseñar un nuevo tipo de prácticas, así como realizar una planificación del calendario de éstas. Las asignaturas que han introducido el LSMaker a su contenido han sido Electrónica, Programación, Introducción a los ordenadores, Cálculo y Álgebra. No hay que olvidar, sin embargo, que no todos los alumnos de primer curso reciben el robot: sólo lo hacen aquellos que se matriculan de todas las asignaturas de este curso. Por este motivo, las asignaturas que han decidido integrar el robot LSMaker al contenido de sus prácticas han tenido que crear diferentes posibilidades de realizarlas para poder atender a las diferentes casuísticas existentes, como prácticas dirigidas a alumnos repetidores o a nuevos estudiantes que llegan con asignaturas convalidadas.

Los nuevos alumnos reciben el LSMaker desmontado en piezas en un maletín, junto con

todo el material necesario y diversa documentación de montaje y de primeros pasos. El montaje del robot se realiza a principios de curso en una de las clases de prácticas de Electrónica. Es a partir de este momento cuando los alumnos podrán empezar a interactuar con él y las otras asignaturas podrán empezar a impartir las prácticas sobre este dispositivo. Con el objetivo de que el robot actuara de elemento común y que las distintas prácticas se relacionaran entre ellas se han establecido dos “puntos de sincronismo”. Estos puntos se materializan en resultados que se obtienen en una práctica y se usan en otras. Por un lado, desde Electrónica los alumnos deben crear una placa de sensores que detecten los colores blanco y negro. Posteriormente, y gracias a esta placa, desde Programación los alumnos tendrán que realizar un código fuente que haga que el robot siga la línea de un circuito cerrado, elemento que servirá para realizar las prácticas de Cálculo y Álgebra, que tratarán de calcular integrales de superficie y detección de formas respectivamente.

Viendo que las prácticas de algunas asignaturas no se podrán realizar si no se implementan aquellas de las que depende se realizó una planificación y un calendario para sincronizar correctamente todas las prácticas afectadas. Como se ha indicado, principalmente son dos las prácticas que generan dependencias con el resto: por un lado la asignatura de Programación necesitará de la realización de la placa de sensores y Álgebra y Cálculo necesitarán el código generado en Programación.



Figura 4. Práctica de Introducción a los Ordenadores

Por lo que se refiere a la práctica de la asignatura de Introducción a los Ordenadores, en

este primer año de implantación se ha realizado una adaptación de la práctica que tradicionalmente se realizaba para que esta pueda ser acoplada al LSMaker. Su resultado se puede apreciar en la figura 4. La práctica consistía en implementar un contador multibase encadenable. Primeramente, los alumnos diseñan el esquema de montaje y revisan que funcione correctamente. Para ello el uso de un simulador se hace imprescindible. Una vez se ha realizado la simulación y el alumno ha comprendido perfectamente el funcionamiento (muy importante para poder detectar y solucionar posibles errores), llegará el momento de implementar el circuito. Finalmente, después de la implementación, se realiza el acoplamiento de la placa con el robot. Llegados a este punto la interacción transversal con otras asignaturas se basa únicamente con Programación. Esto es debido a que el sincronismo necesario para realizar el conteo será producido por un programa que genera los flancos que habilitan el funcionamiento del contador. Si bien este programa se le facilita al alumno, también se le proporciona el código fuente por si desea realizar modificaciones en el mismo y quiere entender la creación del sincronismo.

Por otro lado, queremos comentar que con el fin de motivar tanto al alumnado como al profesorado e incentivar el uso del robot LSMaker, La Salle ha organizado distintos actos y competiciones centradas en el uso del robot. Con ellos se ha pretendido que el público objetivo se familiarizara con él, que lo modificaran a su gusto para competir en combates de sumo, que aprendieran a programarlo para que subiera siempre por la cuesta con mayor pendiente, etc. Estos acontecimientos, junto a diferentes seminarios de uso avanzado (para el momento y los conocimientos que tenían los alumnos), han originado una planificación para motivar en cada momento el aspecto que más interesaba.

Así pues, los primeros acontecimientos simplemente requerían controlar el robot con el control remoto y, posteriormente, se solicitaban ya aportaciones más avanzadas, de manera que a los alumnos les servía para poner los conocimientos adquiridos en práctica y a los profesores para familiarizarse y fomentar el uso de esta herramienta.

### 3.2 El robot en la asignatura de Programación

La asignatura de Programación, concretamente sus prácticas, es la que más cambios ha sufrido respecto al curso anterior debido a que es la asignatura que más ha integrado el robot en su contenido. Así, dos de las tres prácticas que se realizan durante el curso han tenido como elemento de trabajo el LSMaker, manteniendo siempre, en la medida de lo posible, el temario de prácticas indispensable para reforzar los conocimientos teóricos.

Para instaurar el robot como elemento de trabajo de la asignatura se ha tenido que realizar una planificación adecuada respetando el contenido teórico, el contenido deseado en cada práctica y las restricciones de las prácticas de Tecnología. Se ha tenido en cuenta también el hecho de que será necesario impartir clases de prácticas con contenido teórico sobre el LSMaker. Debido a este conjunto de restricciones y al hecho de que durante el periodo de tiempo destinado a la primera práctica los alumnos no tienen conocimiento suficiente sobre el concepto de estructuración de código (funciones y procedimientos), se decidió que la primera práctica no incorporaría el robot y seguiría el mismo patrón que en cursos anteriores.

Así pues, para introducir el robot en las dos últimas prácticas se llevaron a cabo tres tareas principales: primero se definió el calendario de inicio y entregas de prácticas y el tipo de trabajo a realizar en cada práctica. Debido al hecho que no se dispondría de las placas de sensores hasta entrado el segundo semestre, se decidió que la primera de las prácticas usara la API remota y la última la local. La segunda tarea a realizar fue analizar el contenido de la API del robot para decidir las funcionalidades que deberían ser explicadas a los alumnos y, con ayuda del calendario de teoría, se diseñaron cinco sesiones de prácticas. En cada una de ellas se planifica una explicación magistral de una hora de duración y posteriormente se insta a los alumnos a realizar diferentes ejercicios que pongan en práctica los nuevos conocimientos. Estas sesiones se diseñaron para impartirse cuanto antes para que los alumnos tuvieran el máximo tiempo posible para realizar el trabajo. Por último, una vez planificadas las sesiones y situadas éstas en el calendario se procedió a la realización e

implementación, por parte de los monitores, de las prácticas que deberían desarrollar los alumnos, así como de sus normativas y criterios de corrección.

Cabe destacar que para la introducción del LSMaker en las prácticas de Programación se han tenido que destinar más recursos en docencia de prácticas (más monitores por grupo de prácticas y más formación de éstos) así como descartar otros contenidos de prácticas (se ha establecido que se expliquen en cursos posteriores y, por consiguiente, no se presupone su conocimiento para pasar de curso). También se ha recortado una práctica con memoria dinámica para poder dedicar suficiente tiempo a impartir las sesiones y al trabajo personal de cada alumno con la práctica (sobre todo en la que se realiza el seguidor de líneas necesario para las prácticas de álgebra y cálculo).

La segunda práctica de la asignatura unifica la API remota del robot y la librería de gráficos de *Allegro* [4] para generar un simulador/controlador del LSMaker donde, además del trabajo sobre LSMaker, se pretende reforzar el trabajo con ficheros de texto, tipos estructurados y estructuración de código, así como el trabajo en el entorno de desarrollo Dev-C++.

En la tercera práctica se vuelve a reforzar el trabajo con ficheros de texto y se profundiza en las distintas funcionalidades del robot para realizar diferentes ejercicios con dificultad progresiva, relacionados prácticamente todos con el seguimiento de líneas sobre circuitos de mediana escala. Se planificó para la realización de estas prácticas una dedicación de 35 horas para la práctica 2 y 20 horas para la tercera.

Las tareas explicadas hasta el momento son las que se han tenido que realizar para planificar las prácticas del curso 2010/11 de Programación, pero ha habido más trabajo implícito necesario para facilitar el desarrollo de las prácticas por parte de los alumnos con este nuevo dispositivo. Desde la asignatura de Programación, por tratar de algorítmica y por ser la que más trabaja sobre el robot, se ha redactado un manual-documentación extenso sobre las distintas funcionalidades del robot y su correspondiente uso, como material de soporte para los alumnos, al mismo tiempo en que se buscaban defectos o posibles mejores de éstas. También se les ha distribuido varios paquetes de *software* de ejemplos de trabajo sobre las distintas APIs del LSMaker con la finalidad que sirvan

como proyectos de base donde los alumnos puedan empezar a trabajar fácilmente con las prácticas de Programación.

#### 4. Resultados y conclusiones

Debido a que nos hemos encontrado en el primer año de vida de este proyecto en fase *beta*, durante el curso ha habido diferentes aspectos que no han funcionado como se esperaba inicialmente, algunos relacionados incluso con problemas en el funcionamiento del robot. Pese a todo ello, al tratarse de un proyecto incipiente se puede considerar que su implantación y desarrollo han sido un éxito.

Este primer año se ha hecho la introducción del proyecto, pero ha sido esta experiencia la que permitirá establecer y tomar las medidas correctivas para que el próximo curso académico se pueda explotar todavía más el potencial de este robot. Ocasionalmente, se ha visto una falta de motivación por parte de algunos alumnos que, sin embargo, se ha contrarrestado con el entusiasmo con el que han trabajado otros. Los acontecimientos y concursos creados para motivar el uso del LSMaker han tenido una participación más que aceptable, y ha seguido una línea de participación creciente durante el curso. Por lo que se refiere a las prácticas de Programación, los resultados han sido muy parecidos a los de años anteriores: ha habido un 5% de alumnos menos que han entregado la práctica 2 en la fecha de entrega establecida y un 3% más en la tercera práctica.

Para apreciar la opinión de los alumnos al finalizar el calendario regular del curso se les ha pedido una valoración del primer año de este proyecto en los informes de la tercera práctica de la asignatura de Programación. También se les ha pedido que contestaran a una encuesta de valoración general del proyecto. Sobre los comentarios personales, algunos de los más relevantes pueden ser:

“Motiva mucho más trabajar sobre el robot que sobre un terminal, pero hay partes molestas como el hecho de notar que la ejecución nunca es igual de precisa debido al nivel de la batería, el terreno, etc.” (muchos alumnos han apreciado que el mundo físico no es siempre exacto).

“La posibilidad de trabajar sobre un robot es muy positiva ya que se puede ver que todo lo que

se explica en clase se puede aplicar a un proyecto real y nos podrá ser muy útil de cara al futuro.” (otros alumnos observan la relación directa entre teoría y práctica).

Acerca de las encuestas se les preguntaba a los alumnos que valoraran la conveniencia de usar el robot, la calidad del material ofrecido, la atracción de los acontecimientos realizados y, en general, la satisfacción con el robot en este curso. En los resultados se refleja que, en general, los alumnos están satisfechos. A 27 de los 33 alumnos que contestaron la encuesta el robot les ha ayudado a aprender el contenido teórico, 28 están contentos con el material y documentación proporcionados, 22 encuentran interesantes las competiciones lúdicas y, finalmente, 28 de los 33 han visto satisfechas sus expectativas.

Sobre la valoración de este primer año de vida del robot, los profesores y monitores de programación opinan lo siguiente:

“El gran reto que tenemos actualmente con el LSMaker es conseguir que el alumno lo reconozca como un compañero de viaje. Es necesario seguir trabajando en su diseño y en las planificaciones de las diferentes asignaturas para que el alumno lo vea como un recurso que se adapta de manera natural a la docencia de cada asignatura.”

“El LSMaker es una idea interesante que no ha sabido explotar todo el potencial del que dispone.”

“El LSMaker es una herramienta que da un punto de vista diferente a las prácticas de Programación, de manera que los alumnos pueden ver su relación con un elemento físico y que no sólo sirva para realizar aplicaciones de ordenador.”

Así pues, vistas las opiniones de alumnos y profesores, y vista la participación del alumnado durante este primer curso y los resultados las de prácticas en fechas de entrega ordinarias, se puede considerar que el proyecto LSMaker ha sido un éxito. Se ha motivado al alumnado y se han conseguido dos objetivos extras: dar una nueva visión de trabajo y usar tecnología real para plasmar los conocimientos teóricos.

#### Líneas de futuro

Uno de los principales objetivos para el siguiente año académico es instaurar el uso del robot en las prácticas de segundo curso. Cabe recordar que

uno de los objetivos del proyecto es que éste acompañe al alumno a lo largo de todos sus estudios. Para eso actualmente se está planificando el curso con los profesores correspondientes y diseñando en qué puntos de cada asignatura se puede introducir el nexo con el robot.

Por otro lado se está pensando en ampliar los sensores existentes en el robot para poder realizar en Programación otras prácticas diferentes a las realizadas en este curso. Así mismo, se están gestionando cambios en las prácticas de Introducción a los Ordenadores que permitirán obtener más utilidades nuevas del dispositivo y que servirán para la realización de las prácticas de otras asignaturas.

### Agradecimientos

Queremos agradecer especialmente a Francesc Escudero su implicación en este proyecto, pues sin su trabajo, pasión y dedicación nada de lo que

se comenta en este artículo habría visto la luz. Por otro lado agradecemos también a Lisa Kinnear su revisión del texto en inglés incluido en este artículo.

### Referencias

- [1] MICROCHIP (2011) PICKit 3 In-Circuit Debugger [En línea] Disponible en Internet: [http://www.microchip.com/stellent/idcplg?IdcService=SS\\_GET\\_PAGE&nodeId=1406&dDocName=en538340&redirects=pickit3](http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en538340&redirects=pickit3)
- [2] Dev-C++. [En línea] Disponible en Internet: <http://www.bloodshed.net/devcpp.html>
- [3] Planas, Jordi. *Introducció de l'LSMaker a l'entorn docent de programació*. Projecte Final de Carrera, La Salle, Universitat Ramon Llull, 2011.
- [4] Allegro. [En línea] Disponible en Internet: <http://alleg.sourceforge.net/>



# **De las ingenierías informáticas a los nuevos grados. Análisis de los contenidos de las asignaturas destinadas al estudio de la ingeniería del software en los antiguos y nuevos planes de estudio en las universidades de Castilla y León.**

M<sup>a</sup> Dolores Muñoz Vicente\*,

María N. Moreno García y

Vivian López Batista

Departamento de Informática y Automática

Universidad de Salamanca

Plaza de los Caídos s/n

37008 Salamanca

\*mariado@usal.es

## **Resumen**

Se está desarrollando un proyecto de innovación docente subvencionado por la universidad de Salamanca cuyo propósito es analizar los contenidos de las asignaturas destinadas al estudio de la ingeniería del software en los antiguos y nuevos planes de estudio. La mayor parte de la información utilizada ha sido recabada a partir de las webs que las diferentes universidades poseen, o en algunos casos, se ha obtenido a través del profesorado.

En este proyecto se hace un estudio comparativo sobre los programas de las asignaturas en las titulaciones de Ingeniería Superior en Informática, Ingeniería Técnica en las ramas de Gestión y de Sistemas con los nuevos grados. El desarrollo de este proyecto nos ha proporcionado una información que sin duda alguna, nos ayudará a la hora de diseñar los contenidos de las asignaturas para los nuevos grados.

## **Summary**

We are developing a Teaching Innovation Project funded by the University of Salamanca whose purpose is to analyze the contents of the subjects for the study of software engineering in the old and new curricula. Most of the information used was gathered from the web sites of the different Universities, while in some cases it has been obtained through the faculty. In this project, we compare the syllabus of subjects in the Superior Engineering degree in Computer

Engineering in the fields of Management and Systems in the new degrees. The development of this project has provided information which undoubtedly will help us when designing the course content for new degrees.

## **Palabras clave**

Comparativa, Programas, Grados, Ingeniería del Software.

## **1. De los antiguos planes de estudio a los nuevos grados**

En el curso 1997-1998 con la creación de la Licenciatura de Informática comienzan timidamente a implantarse los estudios de informática en la universidad española, convirtiéndose en los años 90 en Ingeniería Informática [3]. La incorporación tanto de este título como los de Ingeniería Técnica en Informática se ha llevado a cabo en todo el territorio nacional de manera gradual impartándose a fecha de hoy, en 45 de las universidades españolas.

El contenido de los estudios ofrecidos por estas titulaciones deriva de la interpretación realizada por cada universidad de las directrices generales aprobadas por el Consejo de Universidades. Sobre la presencia de la ingeniería del software en las universidades españolas, se observa que en muchas de las titulaciones previas al grado se aumenta la troncalidad recomendada y se añaden asignaturas obligatorias u optativas sobre la disciplina.

En relación con los contenidos propuestos se advierte el interés por conjugar la adaptación de los currículos académicos a la realidad social, con el interés de mantener el rigor y el carácter de “persistencia” de los conceptos relacionados con esta disciplina. Éste es un objetivo compatible con la adquisición de conocimientos y destrezas concretas mediante la realización de prácticas y desarrollo de proyectos, contribuyendo así a alcanzar una formación integral del estudiante.

## 2. Los estudios de informática en las universidades de Castilla y León

En la comunidad de Castilla y León se imparten en 6 de las 9 provincias que la constituyen, títulos relacionados con el estudio de la informática. En muchos casos como veremos a continuación, las universidades apuestan por la especialización ofertando diversos itinerarios dentro de un mismo título.

En la universidad de Burgos [2] se puede estudiar el Grado en Ingeniería Informática y una vez superadas las materias de carácter obligatorio, el alumno debe cursar 48 créditos optativos que puede elegir entre cuatro itinerarios (Mención en Computación, Mención en Ingeniería de Computadores, Mención en Ingeniería del Software y Mención en Sistemas de Información).

Como nuestro estudio se centra en la ingeniería del software, el análisis se ha realizado sobre las asignaturas del itinerario de ingeniería del software por el que el alumno conseguirá el título de Grado en Ingeniería Informática en Ingeniería del Software.

La universidad de León [1] oferta un Grado en Ingeniería Informática y la titulación superior de Ingeniería Informática.

En la universidad de Salamanca el alumno puede elegir entre la universidad pública [4] y la privada [5]. Si opta por la primera opción, puede estudiar en dos campus: el campus de Salamanca y el de Zamora y de forma privada lo hará en la universidad Pontificia.

- En el campus Salamanca el alumno podrá elegir entre los títulos siguientes: Grado en Ingeniería Informática, 2º ciclo de Ingeniería Informática, e Ingeniería Técnica en Informática de Sistemas.

- En el campus de Zamora el alumno podrá cursar el título de Grado en Ingeniería Informática en Sistemas de Información e Ingeniería Técnica en Informática de Gestión.
- En la universidad Pontificia el alumno podrá realizar el Grado en Ingeniería Informática y el Grado en Ingeniería Informática mención Ingeniería del Software.

La universidad de Valladolid [6] dispone también de dos campus en los que se puede estudiar titulaciones relacionadas con el estudio de la informática:

- En el campus de Valladolid se ofertan dos grados: el Grado en Ingeniería Informática, y el Grado en Ingeniería Informática de Sistemas además de la titulación de Ingeniería Técnica en Informática de Sistemas.
- En el campus de Segovia se puede cursar el Grado en Ingeniería Informática de Servicios y Aplicaciones y la titulación de Ingeniería Técnica en Informática de Gestión.

## 3. Asignaturas relacionadas con el estudio de la ingeniería del software en los planes de estudios

Si analizamos los planes de estudio vemos que en todas las titulaciones mencionadas en el punto anterior, existen varias asignaturas destinadas al estudio de la ingeniería del software. El hecho de que la puesta en marcha de las nuevas titulaciones se esté haciendo de manera gradual y que muchas de las materias relacionadas con la ingeniería del software se impartan en el segundo año o en cursos superiores, es la causante de que no se disponga de todos los programas.

Afortunadamente, solamente una universidad no muestra en su página web la información que necesitamos para nuestro estudio, nos estamos refiriendo a la universidad de León, en concreto en la titulación de grado, por lo que en esta universidad el análisis se ha realizado sobre los títulos previos al grado.

En el caso de la universidad de Burgos en el Grado de Ingeniería Informática se ofertan en el segundo año de los estudios dos asignaturas de

carácter obligatorio: Ingeniería del software y Análisis y diseño de sistemas. En el Grado de Ingeniería Informática en Ingeniería del Software se deberán cursar cuatro asignaturas en el último año de los estudios: Diseño y mantenimiento del software, Validación y pruebas, Desarrollo avanzado de sistemas software.

En la universidad de León en la titulación de Ingeniería Informática se cursa en el tercer año de los estudios una asignatura de carácter troncal denominada Ingeniería del software.

En la universidad pública de Salamanca en el campus de Salamanca se imparten en el Grado de Ingeniería Informática tres asignaturas de carácter obligatorio: Ingeniería del software I, Ingeniería del software II y Gestión de proyectos en los cursos 2º, 3º y 4º, además de una asignatura optativa denominada Tecnologías de información emergentes. En el segundo ciclo de Ingeniería Informática se imparten dos asignaturas de carácter troncal en los cursos 4º y 5º: Análisis de sistemas y Administración de proyectos informáticos. En la titulación de Ingeniería técnica en Informática de Sistemas se imparte la asignatura Ingeniería del software, asignatura de carácter obligatorio cursada el último año de los estudios.

En el campus de Zamora en la titulación del Grado en Ingeniería Informática en Sistemas de Información es necesario cursar dos asignaturas obligatorias: Ingeniería del Software I y Calidad del software cuya docencia se imparte en el 2º y 3º año de los estudios. En la titulación de Ingeniería Técnica en Informática de Gestión se imparten dos asignaturas de carácter troncal en el segundo y tercer año de carrera: Ingeniería del software I e Ingeniería del software II.

Si se opta por la universidad privada, el alumno en el Grado de Ingeniería Informática debe cursar tres asignaturas obligatorias: Fundamentos de ingeniería del software, Gestión de proyectos e Ingeniería del software web. En el caso de elegir el itinerario de Ingeniería del Software se han de estudiar como complemento, en el último año de los estudios, cuatro asignaturas más: Arquitectura del software, Sistemas de información, Diseño evaluación y desarrollo de interfaces, Servicios y plataformas webs.

En la universidad de Valladolid el alumno también puede estudiar distintas titulaciones en sus dos campus. En Valladolid puede estudiar dos grados: Grado de Ingeniería en Informática en el que deberá cursar cuatro asignaturas obligatorias: Fundamentos de ingeniería del software, Modelado de sistemas software, Diseño de software, Planificación y gestión de proyectos y Grado en Ingeniería Informática de Sistemas en el que deberá cursar dos asignaturas de carácter obligatorio en el 2º y 3º año: Ingeniería del software I e Ingeniería del software II.

En el campus de Segovia en el Grado de Ingeniería Informática en Servicios y Aplicaciones se imparten en el 2º y 4º año de la carrera, dos asignaturas de carácter obligatorio: Procesos de desarrollo software y Análisis de requisitos. En la titulación de Ingeniería Técnica en Informática de Gestión se deberán cursar 3 asignaturas de carácter obligatorio en el último año de los estudios: Ingeniería del software I, Ingeniería del Software II y Calidad del software.

Si estudiamos los programas de las asignaturas anteriormente mencionadas observamos que hay una serie de ítems repetidos: conceptos básicos de la materia, el proceso de ciclo de vida del software, la especificación de requerimientos, el análisis del sistema, el análisis del sistema estructurado, el análisis orientado a objetos, el diseño estructurado, el diseño orientado a objetos, la gestión de configuraciones, la verificación, validación y pruebas, el mantenimiento y evolución del software, la viabilidad, las herramientas Case, la ingeniería inversa, la calidad del software, la gestión de riesgos, la planificación y gestión de proyectos.

Para poder llevar una uniformidad en el estudio se han analizado en que medida los programas de las asignaturas se ajustan a los ítems anteriormente mencionados.

### 3.1. Titulación de ingeniería informática

Esta titulación se puede estudiar en tres universidades: universidad de León, Salamanca y Valladolid.

Se puede afirmar que las tres universidades consideran importante instruir al alumnado en conceptos básicos sobre la ingeniería del

software, el análisis y el diseño orientado a objetos, la calidad del software y el diseño y gestión de proyectos, ya que todas ellas destinan uno o varios temas de sus asignaturas para su estudio.

En cuanto al proceso del ciclo de vida del software, la especificación de requerimientos, el análisis del sistema estructurado, y la gestión de riesgos hay que decir que también dos universidades destinan parte del temario de sus asignaturas a su estudio. Sin embargo se le da menos importancia al diseño estructurado y al uso de herramientas Case ya que en este caso solamente una universidad destina algún tema para su estudio.

### **3.2. Titulación de ingeniería técnica en informática de sistemas**

Las universidades de Valladolid y Salamanca en los campus de Valladolid y Salamanca tienen dentro de su oferta académica estos estudios.

En este caso las dos universidades incluyen en algunas de sus asignaturas conceptos relacionados con el proceso de ciclo de vida del software, la especificación de requerimientos, el análisis y el diseño y la calidad del software. Aspectos relacionados con los conceptos básicos sobre la ingeniería del software, el análisis y el diseño estructurado de sistemas, las herramientas Case y la ingeniería inversa se estudian en alguna asignatura de los dos campus. El mantenimiento y la evolución del software tan sólo se estudian en uno de los campus.

### **3.3. Titulación de ingeniería técnica en informática de gestión**

Esta titulación se estudia en la universidad de Valladolid en los campus de Segovia y Valladolid y en la universidad de Salamanca en el campus de Zamora.

En este caso en los tres campus se incluyen en los programas de sus asignaturas conceptos relativos a la especificación de requerimientos, el análisis y diseño orientado a objetos y la calidad del software.

Por otro lado, hay que decir que a los aspectos relacionados con los conceptos básicos sobre la ingeniería del software, el análisis y el diseño estructurado de sistemas, el uso de herramientas Case, la ingeniería inversa y la

gestión y planificación de proyectos se le da también gran importancia ya que las dos universidades los incluyen en su temario. Sin embargo, se observa que tan sólo en una de las universidades se imparten conceptos relacionados con el mantenimiento y la evolución del software.

### **3.4. Grado en informática**

Todas las universidades de Castilla y León han apostado por el cambio que la universidad española está sufriendo en estos momentos, por ello, en todas ellas se puede estudiar al menos una titulación de grado relacionado con el estudio de la informática. Como se ha comentado en puntos anteriores el hecho de que la implantación del grado se esté haciendo de manera progresiva es la causante de que no se disponga de toda la información sobre los programas de las asignaturas, no obstante, hacemos el análisis con la información que existe en estos momentos en las webs.

Tanto en la universidad de León como en la de Burgos se puede estudiar el Grado de Ingeniería Informática. En este momento no se dispone de la información relativa a las asignaturas relacionadas con el estudio de la materia de Ingeniería del Software.

En la universidad pública de Salamanca se puede estudiar en el campus de Salamanca el Grado en Ingeniería Informática y en el campus de Zamora, el Grado en Ingeniería Informática en Sistemas de Información.

La universidad privada también se ha sumado al cambio que está sufriendo la universidad en su conjunto en estos momentos, ofreciendo la posibilidad de cursar el Grado en Ingeniería Informática y el Grado en Ingeniería Informática mención de Ingeniería del Software.

En el caso de la universidad de Valladolid hay que decir que se ha hecho una apuesta fuerte por estos estudios, ya que en ella se pueden cursar tres grados diferentes: el Grado en Ingeniería Informática de Servicios y Aplicaciones en el campus de Segovia y los grados en Ingeniería Informática y en Ingeniería Informática de Sistemas en el campus de Valladolid.

A continuación vamos a pasar a revisar los contenidos de los programas de los grados

ofertados por las universidades de Salamanca y Valladolid.

Todos los grados confieren gran importancia al estudio de la especificación de requerimientos ya que destinan al menos un tema de los programas de sus asignaturas para su estudio.

A los conceptos básicos sobre la ingeniería del software y al análisis orientado a objetos también se le da bastante importancia ya que se estudian en cuatro de los siete grados.

Conceptos relacionados con el proceso del ciclo de vida del software, el análisis orientado a objetos, la calidad del software y la planificación y gestión de proyectos se estudian en tres de los siete grados.

El análisis estructurado de sistemas y aspectos relacionados con la verificación, validación y pruebas se estudian en solamente dos grados.

Por último hay que decir que apenas se le da importancia al estudio de la viabilidad, las herramientas Case y la gestión de riesgos ya que tan sólo se estudian en uno de los siete grados existentes.

De modo gráfico se ha representado a través de las figuras siguientes el número de grados en las universidades de Castilla y León (Figura 1) y los campus en los que se estudian titulaciones previas al grado (Figura 2).

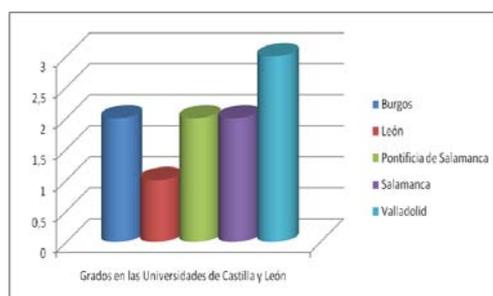


Figura 1. Número de grados en las universidades de Castilla y León

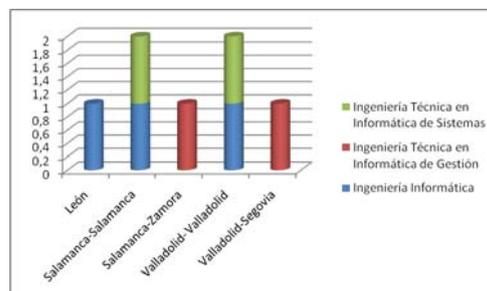


Figura 2. Campus en los que se estudian titulaciones previas al título de grado

### Conclusiones

A tenor de la información mostrada en el artículo se puede afirmar que gran parte de las universidades incluyen en sus programas con peso importante, temas relacionados con el estudio de conceptos básicos sobre la ingeniería del software, el análisis y el diseño orientado a objetos (85%), la calidad del software (70 %), el proceso del ciclo de vida del software, el análisis estructurado y la planificación y gestión de proyectos (62%).

Dándole menos importancia a aspectos relacionados con el diseño estructurado y la gestión de riesgos (38%), las herramientas Case (31%), la verificación, la validación y pruebas, mantenimiento y evolución del software y la ingeniería inversa (15%), la gestión de configuraciones y la viabilidad (7%).

Como era de esperar la especificidad en contenidos aumenta de manera proporcional al número de asignaturas ofertadas en cada universidad.

### Referencias

- [1] Universidad de Burgos, 2011.
- [2] Universidad de León, 2011.
- [3] Matellón, V. *¿Qué tiene que estudiar un informático?*. TodoLinux Número 23 Pág 12-13, 2002.
- [4] Universidad de Salamanca, 2011.
- [5] Universidad Pontificia de Salamanca, 2011.
- [6] Universidad de Valladolid, 2011.



# Pickaxe: Herramienta para la docencia de minería de datos

Javier Verde Velasco, María N. Moreno García\*,  
Vivian F. López Batista y María Dolores Muñoz  
Vicente

Depto. de Informática y Automática. Universidad de Salamanca  
Plaza de los Caídos s/n  
37008 Salamanca  
\*mmg@usal.es

## Resumen

La minería de datos es una disciplina que tiene una amplia presencia en los estudios de posgrado, tanto en másteres y doctorados oficiales como en títulos propios de las Universidades. En el mercado existe una buena oferta de herramientas que podrían ser utilizadas en la enseñanza de dicha materia, sin embargo, la mayoría tienen un coste elevado y no permiten examinar la implementación de los algoritmos. Aunque existen algunas de código abierto, la comprensión del mismo es difícil y se requieren conocimientos elevados para que el usuario pueda incorporar nuevos algoritmos. En este trabajo se presenta una herramienta desarrollada con fines docentes, la cual tiene implementados diversos algoritmos de minería y visualización de datos, permitiendo a su vez una fácil incorporación de nuevos algoritmos.

## Summary

Data mining is a subject with a wide presence in postgraduate studies such as official Master and Doctorate degrees as well as other University titles. There are great offers of commercial tools that can be used in the teaching of this matter, however most of them are very expensive and they do not allow to examine the algorithms' implementation. In spite of the availability of some open source code tools the comprehension of their code is difficult and the incorporation of new algorithms by the user requires a high knowledge level. In this work a tool for teaching is presented, which implements several data mining and visualization algorithms and it allows an easy incorporation of new algorithms.

## Palabras clave

Minería de datos, herramienta, docencia.

## 1. Introducción

La minería de datos ha dado lugar a una paulatina sustitución del análisis de datos dirigido a la verificación por un enfoque de análisis de datos dirigido al descubrimiento del conocimiento. La principal diferencia entre ambos se encuentra en que en el último se descubre información sin necesidad de formular previamente una hipótesis. La aplicación automatizada de algoritmos de minería de datos permite detectar fácilmente patrones en los datos, razón por la cual esta técnica es mucho más eficiente que el análisis dirigido a la verificación cuando se intenta explorar datos procedentes de repositorios de gran tamaño y complejidad elevada.

Los algoritmos de minería de datos se clasifican en dos grandes categorías: supervisados o predictivos y no supervisados o de descubrimiento del conocimiento [9]. Los algoritmos supervisados o predictivos predicen el valor de un atributo denominado etiqueta, de un conjunto de datos, conocidos otros atributos denominados atributos descriptivos. A partir de datos cuya etiqueta se conoce se induce una relación entre dicha etiqueta y otra serie de atributos. Esas relaciones sirven para realizar predicciones en datos cuya etiqueta es desconocida. Esta forma de trabajar se conoce como aprendizaje supervisado. Los métodos no supervisados o de descubrimiento del conocimiento descubren patrones y tendencias en los datos actuales, no utilizan datos históricos. El descubrimiento de esa información sirve para llevar a cabo acciones y obtener un beneficio de ellas.

La intensiva investigación llevada a cabo en los últimos años en el ámbito de la minería de datos ha dado lugar a que esta disciplina se haya

ido incorporando progresivamente a los estudios de posgrado de las áreas de informática tanto a nivel nacional como internacional, estando actualmente presente en gran número de másteres y doctorados oficiales, así como en títulos propios de diferentes Universidades. Los estudiantes que cursan estos estudios tienen por lo general pocos conocimientos en esta materia debido a que ésta no suele formar parte de los planes de estudio de las titulaciones de grado, por lo que resulta imprescindible disponer de herramientas que le faciliten el aprendizaje en estas condiciones. Las herramientas comerciales no cumplen con dicho requisito, además de tener el inconveniente del alto coste de adquisición y mantenimiento. Las herramientas de código abierto, como por ejemplo Weka (University of Waikato, Hamilton, New Zealand), permiten la aplicación de un gran número de algoritmos pero el examen de su lógica interna se ve dificultada por la complejidad del código, hecho que también dificulta la adición de nuevos algoritmos, para lo cual se requieren elevados conocimientos tanto de minería de datos como de programación. Estas circunstancias han motivado el desarrollo de la herramienta que se presenta en este trabajo, la cual representa una plataforma sencilla para su manejo por los estudiantes de posgrado permitiéndoles comprender los algoritmos, visualizar sus resultados e incluso incorporar nuevos algoritmos desarrollados por ellos mismos.

El resto del trabajo se organiza de la manera siguiente: La sección 2 contiene la descripción de las principales características de la herramienta, en la sección 3 se describen brevemente los algoritmos implementados, seguidamente se proporcionan algunos detalles del análisis y diseño; en la sección 5 se comentan algunos aspectos de implementación y se muestra la interfaz de usuario, y finalmente se presentan las conclusiones.

## 2. Características de Pickaxe

El diseño de Pickaxe se ha realizado teniendo presente en todo momento la finalidad docente. Además, se ha considerado la adaptabilidad como un requisito imprescindible para dar respuesta a posibles cambios en los enfoques o contenidos de las asignaturas. En la actualidad sus principales características son las siguientes:

- Implementación de algoritmos de minería: la herramienta dispone varios algoritmos de minería de datos, de los tres tipos fundamentales presentes en la disciplina (clasificación, asociación y agrupamiento), así como filtros y formas de preprocesamiento para los conjuntos de datos iniciales, de forma que puedan adecuarse a las condiciones de entrada de los algoritmos.
- Visualización de datos: la aplicación proporciona una forma de visualización gráfica de los conjuntos de datos introducidos y de los resultados.
- Exportación de gráficos: Permite exportar en formato estándar de imagen las posibles formas de mostrar los datos de las que dispone.
- Estadísticas: el sistema genera estadísticas de tiempo y rendimiento de los algoritmos.
- Resultados: la herramienta permite exportar y almacenar los resultados obtenidos tras la aplicación de un algoritmo, para evitar la necesidad de recalcular si fuera necesario volver a analizarlo. Asimismo permite la generación de informes en formato pdf.
- Facilidad de implementación de algoritmos adicionales: la aplicación está diseñada de forma que un desarrollador pueda ampliarla con nuevos algoritmos a posteriori de una manera sencilla, lo que facilita su extensión futura.
- Multiplataforma: la aplicación proporciona funcionamiento completo en los principales sistemas operativos actuales: Windows XP, Windows Vista, Linux y Mac OS X, además de utilizar *el look & feel* apropiado en cada sistema.
- Internacionalización: Se ha hecho uso de las utilidades que proporciona Java para la internacionalización de la aplicación. Aunque la versión actual solo está disponible en español e inglés, puede traducirse fácilmente a cualquier idioma reconocido por Java sin necesidad de recompilar, únicamente añadiendo un fichero de recursos al JAR.

## 3. Algoritmos implementados

Se han implementado los algoritmos más representativos de las diferentes categorías de la

minería de datos. Dichos algoritmos se describen a continuación.

### 3.1. Asociación

En el grupo de las técnicas no supervisadas una de las más utilizadas es el análisis de asociación, que persigue el establecimiento de relaciones entre registros individuales o grupos de registros de la base de datos. Dos especializaciones del análisis de asociación son las reglas de asociación y el descubrimiento de patrones secuenciales. El propósito de las reglas de asociación es descubrir artículos que implican la presencia de otros artículos en la misma transacción. Aplicando esta técnica a un conjunto de transacciones se descubren afinidades entre la colección de artículos. El algoritmo Apriori [1] es el algoritmo de reglas de asociación implementado en Pickaxe debido a que es el más representativo. El algoritmo Apriori sólo trabaja con conjuntos de datos discretos, por lo que es necesario discretizar los atributos numéricos antes de aplicarlo.

Otro de los algoritmos que incorpora la herramienta es AprioriAll [3], el cual intenta obtener reglas de asociación secuenciales a partir de un conjunto de datos. Este tipo de reglas revelan información sobre la secuencia en que los consumidores adquieren bienes. Los datos con los que trabaja este algoritmo tienen un formato muy concreto: en primer lugar, un atributo ID que sirva como identificador único de un determinado cliente; un atributo fecha que identifique el momento en el que se realizó la transacción estudiada, y un conjunto de enteros que identifiquen las acciones realizadas por ese cliente en ese momento de tiempo.

### 3.2. Agrupamiento

Los métodos de agrupamiento (*clustering*) son también técnicas no supervisadas utilizadas para encontrar afinidades subyacentes en los datos mediante el descubrimiento de subpoblaciones homogéneas. Los más usados son los denominados de agrupación demográfica, los cuales utilizan medidas de distancia para distribuir los registros de datos en grupos con dispersión mínima. Los algoritmos incorporados en la herramienta se describen a continuación.

El algoritmo K-Medias (*K-Means*) [6], es un algoritmo de agrupamiento por vecindad, en el que se parte de un conjunto de ejemplos a agrupar

y de un número determinado de agrupaciones. Una variante de este algoritmo es el K-Medias iterativo. En lugar de especificar un determinado número de grupos, se especifica un intervalo con mínimo y máximo. El algoritmo aplicará el algoritmo *K-means* para todos los valores posibles del intervalo, y retornará el mejor. Como es evidente, esta variedad disminuye el rendimiento en el entrenamiento, pero garantiza una mejor selección de grupos.

COBWEB [4] y CLASSIT [5] son dos algoritmos de agrupación jerárquica que ordenan un conjunto de datos en un árbol de agrupamiento, construyéndolo de forma progresiva, comenzando por el conjunto total de ejemplos y subdividiéndolo hasta llegar a los elementos individuales. Su funcionamiento es similar, diferenciándose únicamente en la naturaleza de los atributos tratados: COBWEB trata únicamente con atributos discretos, y CLASSIT con atributos continuos con distribución normal.

### 3.3. Clasificación

En los métodos supervisados el proceso de aprendizaje requiere la división del conjunto de datos con etiqueta conocida en dos subconjuntos disjuntos, generalmente uno de mayor tamaño que el otro. El proceso se desarrolla en dos fases:

- Entrenamiento: construcción de un modelo de predicción usando el mayor de los subconjuntos.
- Prueba: se comprueba la validez del modelo inducido probándolo sobre el resto de los datos.

Si las pruebas demuestran que el modelo es válido, éste puede ser utilizado para predecir la etiqueta en datos no etiquetados.

Cuando la etiqueta toma valores discretos, cada uno de estos valores representa una clase, por lo que el proceso de aprendizaje se denomina clasificación.

Los árboles de decisión son una familia de algoritmos de clasificación que emplea una técnica de partición para generar un árbol a partir de los datos ejemplo. Todos los algoritmos de esta clase tienen la misma estructura general. Existen muchos ejemplos de algoritmos de esta familia, entre los que podemos destacar el CART, el ID3 o el C4.5 [8].

Naive Bayes es un algoritmo de clasificación basado en redes bayesianas muy conocido y

utilizado. Es una buena introducción al funcionamiento de las redes bayesianas. El fundamento principal de este algoritmo se basa en la premisa de que una vez conocido el valor de la variable clase, el resto de atributos son independientes entre sí. Aunque es una suposición muy fuerte, e incorrecta en algunos casos, el clasificador obtiene resultados excelentes en muchos casos, superando a otros algoritmos con la misma función.

#### 4. Modelado del sistema

Como ya se ha comentado anteriormente, la aplicación se ha modelado de forma que su implementación posterior permita adicionar fácilmente nuevos algoritmos tanto de preprocesamiento como de minería, además de permitir la modificación de los existentes. En el modelo de casos de uso presentado en la figura 1 se puede observar la manera en que se han organizado las diferentes funcionalidades del sistema.

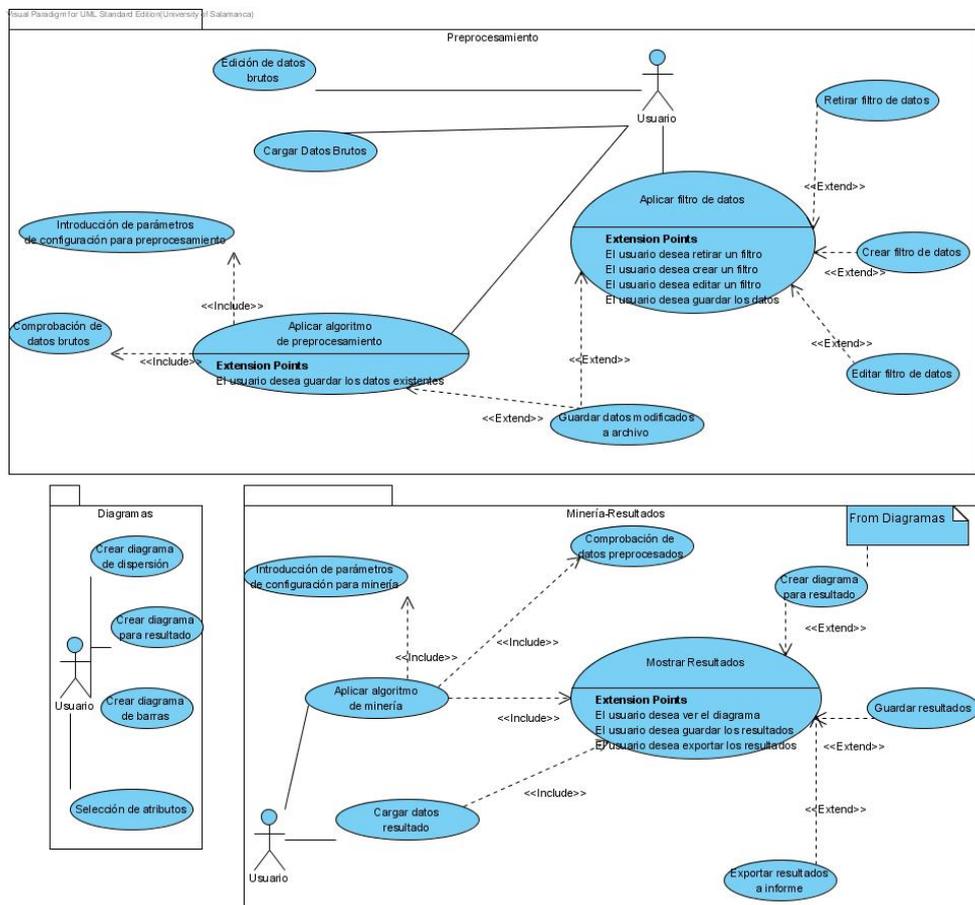


Figura 1. Modelo de casos de uso

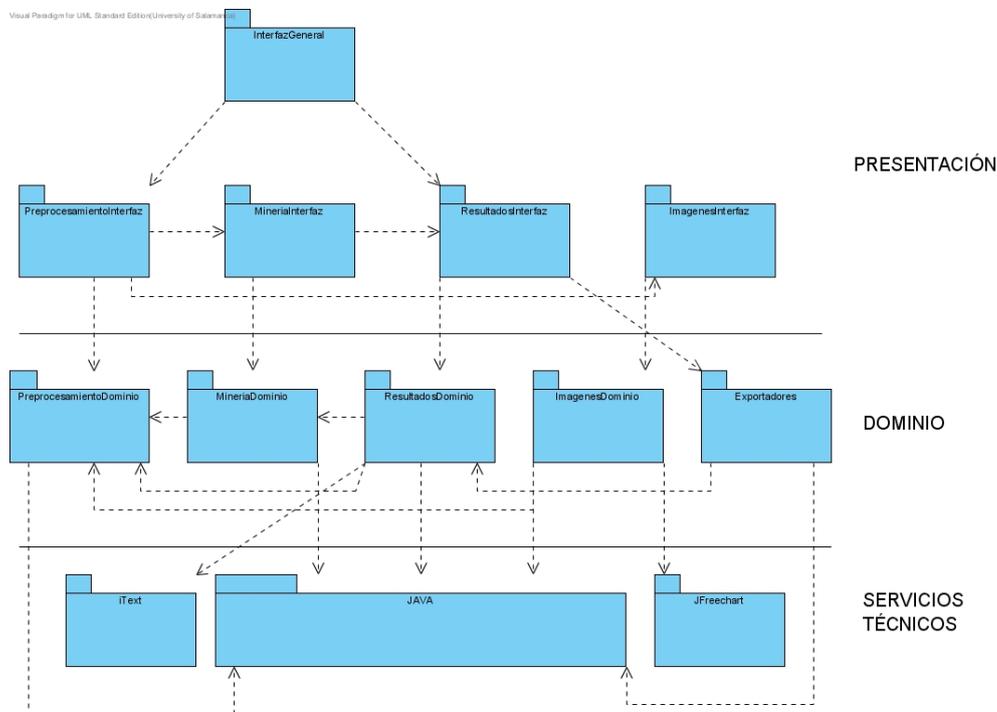


Figura 2. Arquitectura de capas

El modelado de análisis se realizó sin tener en cuenta los algoritmos concretos que iban a ser implementados, sino que fueron tratados genéricamente de forma abstracta. Durante el diseño ya se trataron los detalles de implementación de cada uno de los algoritmos, cada uno de los cuales requería estructuras de datos propias, bastante complejas por lo que fue necesario modelar una gran cantidad de tipos de datos y sus correspondientes clases. Aún así, se ha mantenido el objetivo de poder añadir algoritmos, filtros y resultados sin modificar demasiadas clases. Por lo tanto, se ha buscado minimizar el acoplamiento reduciendo el número de clases que necesitan conocer el funcionamiento de los algoritmos, y reducir al mínimo el conocimiento del funcionamiento interno de los algoritmos que necesitan las clases. Se han aplicado diferentes patrones de diseño para conseguir dicho objetivo. En la figura 2 se muestra la arquitectura de capas del sistema con las relaciones de uso entre los paquetes de diseño.

## 5. Implementación

La implementación se ha realizado en lenguaje Java, se ha utilizado la versión 6.0 de la máquina virtual. Además se ha hecho uso de algunas bibliotecas:

- JFreeChart: permite la utilización de gráficos de muchos tipos en una aplicación Java. Es compatible con la biblioteca Swing, con formatos estándar (JPG y PNG, entre otros) y su uso es sencillo. Esta biblioteca requiere el uso de la biblioteca JCommons. La versión utilizada es la 1.0.11.
- JCommons: Implementa algunas funciones no presentes en las bibliotecas estándar de Java, pero de uso corriente entre los desarrolladores, como generación de código, o creación de fábricas de objetos.
- iText: Añade a Java funcionalidades para exportar archivos de tipo PDF, pudiendo añadir a éstos texto o imágenes. Es compatible con Java desde su versión 1.4. La versión de la biblioteca utilizada es la 2.1.5.

Una de las primeras decisiones relativas a la implementación fue la elección del formato de entrada de los datos. Se necesitaba un formato fácil de utilizar, al que pudieran transformarse los datos a importar por el programa de forma sencilla, y que no requiriera demasiado tiempo de procesamiento. Finalmente nos decidimos por la solución más sencilla y bastante similar a la empleada en la mayoría de las herramientas, texto plano con un carácter especial como separador. El archivo debe disponer de unas líneas de cabecera que permiten nombrar los datos y asignarles un tipo.

En la figura 3 se muestra la visualización de los datos de entrada en forma tabular. Existe también la opción de representarlos gráficamente mediante diagramas de barras o de dispersión. Si es de barras, solo hay que elegir una componente (de cualquier tipo) y se representará la proporción de elementos. Si hay demasiados elementos distintos y el campo es de tipo numérico, Pickaxe los agrupará automáticamente en intervalos de igual tamaño para su presentación. La manera de seleccionar y configurar un determinado diagrama se muestra en la figura 4, cuyo resultado aparece en la figura 5.

La herramienta permite aplicar diferentes tipos de filtro a los datos de entrada. Uno de los más útiles es el de discretización de atributos continuos. La figura 6 muestra un ejemplo de discretización en intervalos de igual tamaño.

Una vez preparados los datos de entrada se podrán aplicar los algoritmos de minería de datos. En la figura 7 se presenta la ventana de selección de los algoritmos de agrupamiento, en la que además se pueden introducir los parámetros de configuración del algoritmo seleccionado.

Nombre	Tipo	Intervalo	Activo
Nombre	Cadena	n/a	<input checked="" type="checkbox"/>
Tipo	Cadena	n/a	<input checked="" type="checkbox"/>
Precio	Numérico	8.0-40.0	<input checked="" type="checkbox"/>
Menú	Booleano	n/a	<input checked="" type="checkbox"/>
Capacidad	Numérico	10.0-100.0	<input checked="" type="checkbox"/>
Recomendado	Booleano	n/a	<input checked="" type="checkbox"/>

Figura 3. Datos de entrada



Figura 4. Selección y configuración de diagramas

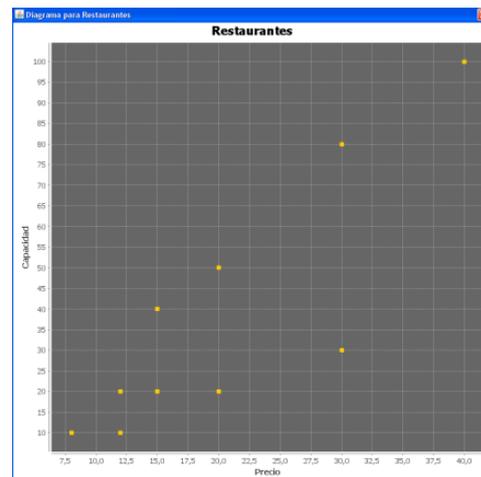


Figura 5. Diagrama de dispersión

The dialog box shows the following configuration:
 

- Realizar intervalos: nominales
- con: igual tamaño

Nombre	Tipo	Intervalo	Activo
Nombre	Cadena	n/a	<input type="checkbox"/>
Tipo	Cadena	n/a	<input type="checkbox"/>
Precio	Numérico	8.0-40.0	<input type="checkbox"/>
Menú	Booleano	n/a	<input type="checkbox"/>
Capacidad	Numérico	10.0-100.0	<input type="checkbox"/>
Recomendado	Booleano	n/a	<input type="checkbox"/>

Figura 6. Discretización de atributos



Figura 7. Selección de algoritmos de agrupamiento

Los resultados producidos por los algoritmos de minería de datos se presentan de forma diferente en función de sus características. Todos ellos se pueden visualizar en forma textual (figura 8) y además para algunos existe la posibilidad de obtener una representación gráfica, como es el caso de los árboles de decisión (figura 9)

### 6. Conclusiones

En este artículo se presenta una herramienta de minería de datos desarrollada con fines didácticos.

La herramienta cumple con los dos objetivos principales que se fijaron cuando fue concebida. El primero de ellos fue implementar una herramienta fácil de utilizar por personas sin experiencia que pudiese ayudar a comprender los algoritmos de minería de datos. El segundo objetivo consistió en desarrollar una aplicación fácilmente extensible y modificable que fuera adaptándose a los cambios en las asignaturas e incluso a la incorporación de nuevas asignaturas de mayor nivel de especialización como podrían ser “Minería Web”, “Minería de textos”, etc.

En la actualidad la herramienta cuenta con un grupo de algoritmos que podrían considerarse como los más representativos de las diferentes categorías de la minería de datos, pero en el futuro se incorporarán nuevos algoritmos. Además, la aplicación está dotada con funcionalidades de preprocesamiento de datos, de representación gráfica, evaluación de rendimiento, exportación y almacenamiento de resultados y generación de informes. Otras características que cabe destacar son la internacionalización de la interfaz y las facilidades multiplataforma.

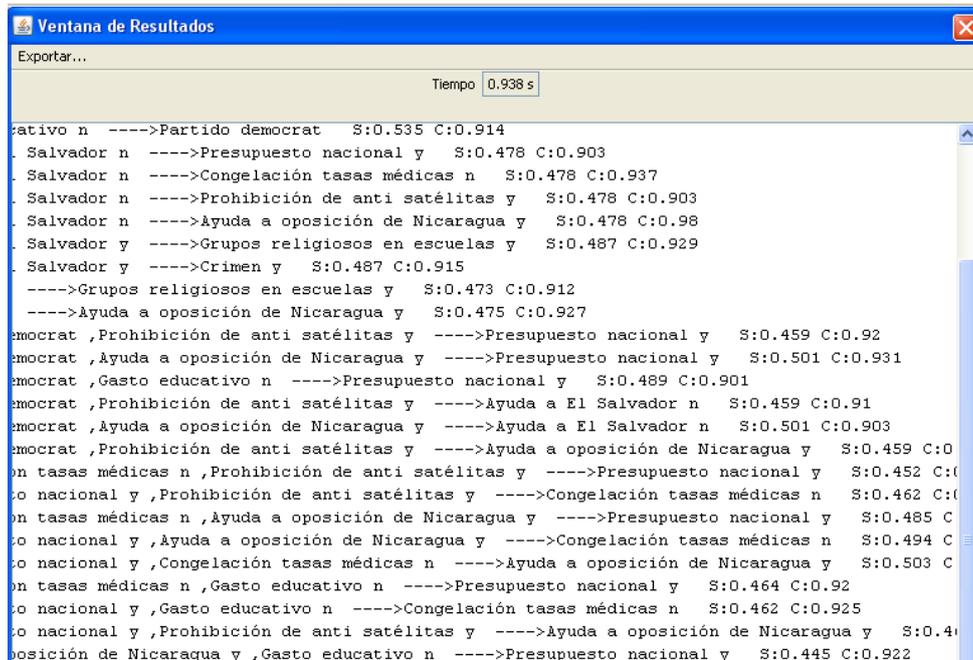


Figura 8. Reglas de asociación obtenidas mediante el algoritmo A priori

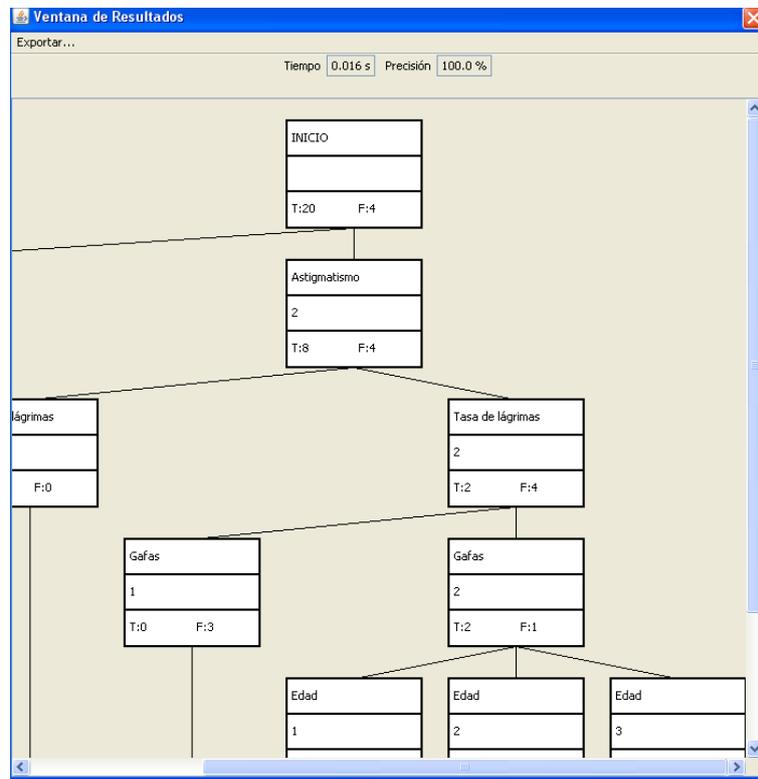


Figura 9. Representación de un árbol de decisión

## Referencias

- [1] Agrawal, R. Imielinski, T., Swami, A. Mining associations between sets of items in large databases. Proc. of ACM SIGMOD Int. Conference on Management of Data, Washinton, D.C., 207-216, 1993.
- [2] Duda, R.O. and Hart, P.E. Pattern Classification and Scene Analysis. John Wiley and Sons, 1973.
- [3] Dunham, M.H. Data Mining Introductory and Advanced Topics. Beijing: Tsinghua University Press, 195-220, 2003.
- [4] Fisher, D. Knowledge acquisition via incremental conceptual clustering, Machine Learning, 2, 139-172, 1987.
- [5] Gennari, J.H., Langley, P., Fisher, D. Models of incremental concept formation. Journal of Artificial Intelligence, 40:11-61, 1989.
- [6] Moody, J, Darken, C. Fast Learning in Networks of locally tuned processing Units en *Neural Computation*, 1, 281-294, 1989.
- [7] Pearl, J. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Representation and Reasoning Series San Francisco, California: Morgan Kaufmann, 1988.
- [8] Quinlan, J.R. Induction of Decision Trees. Machine Learning, 1, 81-106, 1986.
- [9] Weiss, S.M., Indurkha, N. Predictive data mining. A Practical Guide. Morgan Kaufmann Publishers, San Francisco, 1998.

# Estudio de métodos para fomentar la autoevaluación del alumnado

Antonio Egea, Antoni Jaume-i-Capó, Joe Miró, Carlos Guerrero

Departament de Matemàtiques I Informàtica  
Universitat de les Illes Balears  
Ctra. Valldemossa km 7,5 - Ed. Anselm Turmeda  
07122 Palma

{antonio.egea,antoni.jaume,joe.miro,carlos.guerrero}@uib.es

## Resumen

En este trabajo se incluye un estudio preliminar de como responden los alumnos a distintos niveles de motivación en la realización de actividades de autoevaluación. Los procesos de autoevaluación se han realizado basándose en el uso de rúbricas. Estas rúbricas han sido pasadas en distintos grupos de alumnos, motivando a cada uno de ellos de forma distinta. Se ha estudiado el nivel de realización (número de alumnos que realizaron la autoevaluación) y se ha buscado si existe alguna correlación con el tipo de motivación.

## Summary

This paper includes a preliminary study of how students respond to different levels of motivation in self-assessment activities. Self-assessment processes have been based on the use of rubrics. These items have been passed in different groups of students, encouraging each of them differently. We have studied the level of achievement (number of students taking the self-assessment) and we have analyzed if there is any correlation with the type of motivation.

## Palabras clave

Autoevaluación, rúbricas, motivación alumnado, EEES, estudio experimental.

## 1. Introducción

El aprendizaje continuo y la retroalimentación de las actividades realizadas por los alumnos tiene un papel fundamental dentro de los nuevos planes de estudio y del nuevo espacio europeo de educación superior. Esto supone que el profesorado ha de llevar a cabo una mayor cantidad de trabajo basado en la redac-

ción y corrección de actividades, trabajos y ejercicios.

Se ha demostrado que la autoevaluación es una técnica beneficiosa para los alumnos y que puede sustituir perfectamente al trabajo de corrección del profesor incluso, en algunos casos, llegando a ser preferible. Muchos han sido los trabajos que se han llevado a cabo sobre los beneficios de la autoevaluación, pero en muchas ocasiones se duda de los mejores métodos para motivar al alumno a que la realice.

El estudio que se presenta aquí realiza un análisis de tres posibles caminos para motivar al alumno a su autoevaluación. Estos caminos se diferencian en la estrategia del planteamiento a los alumnos:

- En un primer lugar se intenta fomentar la autoevaluación mediante la asignación de una calificación con peso en la nota final de la asignatura.
- Una segunda estrategia se basa en la insistencia constante por parte del profesor en la importancia de este tipo de actividades.
- En una última estrategia, se confía en la propia madurez del alumnado para llevar a cabo estas autoevaluaciones

El estudio está enfocado a analizar con cual de estos métodos el alumno es más reactivo y participa más en la autoevaluación. Por este motivo, lo que se ha estudiado es el grado de participación del alumnado, en otras palabras, la métrica que estudiamos es el porcentaje de alumnos que han realizado autoevaluación para cada una de las tres estrategias utilizadas.

El experimento se ha llevado a cabo con cinco profesores. Estos profesores han motivado la autoevaluación en distintos grupos de alumnos mediante cada una de las estrategias planteadas inicialmente. Estas estrategias se han planteado sobre grupos de distintas asignaturas intentando evitar los efectos de otros parámetros como son el nivel de los alumnos o la madurez de los mismos. Todas las asignaturas for-

man parte de planes de estudios de carreras de informática. La herramienta utilizada para que los alumnos se autoevaluarán han sido las rúbricas. Los datos extraídos del experimento han sido el porcentaje de alumnos que han completa las rúbricas.

Resumiendo, la contribución de este trabajo es un estudio inicial y estadístico de la participación de los alumnos usando tres formas de fomentar la autoevaluación.

## 2. Estado del arte

Durante los últimos años la co-evaluación y la auto-evaluación están recibiendo un interés creciente, seguramente al vernos obligados a trabajar en el aula competencias genéricas como el pensamiento crítico y la capacidad de análisis, debido a la reciente implantación del EEES. Pueden parecer nuevas formas de evaluación, pero ya entre 1774 y 1826, el profesor George Jardine de la Universidad de Glasgow las incluía en su plan pedagógico [6].

Al delegar en los alumnos toda o parte de la evaluación se obtienen mejoras en su proceso de aprendizaje, con el correspondiente aumento del trabajo para el profesorado. Gracias a las herramientas en línea existentes, podemos reducir un poco este trabajo [3, 13, 15], sin olvidar otras herramientas mas específicas para los estudios de informática [7, 16].

Sobre la utilidad de este tipo de evaluaciones, la mas destacada es la retroalimentación, al no ser el alumnado un elemento pasivo en el proceso y tener que corregir, confirmar y/o hacer sugerencias sobre el trabajo que evalúa. Además se añade un factor formativo en el proceso de evaluación, mucho mas importante que el aditivo, al ser consciente el alumnado de sus puntos fuertes y sus puntos débiles [3, 14]. Por otra parte, está demostrado que el alumnado mejora en la gestión de su aprendizaje, dedicándose a profundizar sobre sus nuevos conocimientos, y le permite trabajar las competencias de reflexión y generalización de nuevas situaciones, trabajo en grupo, autoaprendizaje y gestión del tiempo, entre otras [13].

En referencia a la fiabilidad, al trabajar en investigación todos estamos habituados y damos por buena la revisión entre iguales (co-evaluación) y la revisión de nuestros propios trabajos (autoevaluación). De hecho si no fuese así, nunca los enviaríamos a congresos y revistas, y no daríamos por buenos los

resultados de la revisión recibida. Aún así, muchos artículos publicados recientemente comparan la correlación de este tipo de evaluaciones con la evaluación del profesor [10, 4, 9, 11, 12]. Eso si, todas las referencias consultadas suelen tener un punto en común, incluso cuando los resultados no son satisfactorios, para que este tipo de evaluaciones sean fiables es muy importante conocer que se quiere evaluar, y dar a los alumnos los conocimientos y las herramientas necesarias para que puedan hacerlo correctamente.

Delegar la evaluación a los alumnos, para obtener un nivel aceptable de fiabilidad, implica que el profesorado debe definir con gran claridad y precisión los criterios de evaluación, de la misma forma que los editores lo hacen con los revisores [11, 14, 5, 2]. Entre otras cosas, requiere mucha mas preparación de entrada, sobre todo de un modelo de soluciones detallado [3], y con el consiguiente seguimiento durante el proceso de evaluación [14]. Las experiencia que tengan nuestros alumnos en este tipo de evaluaciones, en los instrumentos y criterios de evaluación, y la naturaleza de la materia son variables muy importantes que afectan la fiabilidad y validez [13]. A pesar de todas estas variables, según la literatura los aspectos mas importantes para asegurar el éxito, resumidos en [13], son:

- Explicar el porque se ha decidido aplicar la auto-evaluación y/o co-evaluación a todas las partes interesadas.
- Involucrar a todos los participantes y explicar los criterios de evaluación de forma detallada.
- Facilitar la comunicación entre los participantes.
- Proporcionar un entrenamiento de calidad, con ejemplos y practicas.
- Proporcionar directrices y formas de comprobación.
- Especificar claramente la planificación y el calendario de actividades.
- Realizar un seguimiento del proceso y con el consiguiente apoyo.
- Evaluar las evaluaciones y retroalimentar al alumnado para que mejore.

Analizando estos aspectos, la fiabilidad de este tipo de evaluaciones depende en gran medida de la formación, la motivación, las herramientas y el soporte que se ofrezca al alumnado.

La autoevaluación y la co-evaluación obligan al alumnado tener que corregir, confirmar y/o hacer sugerencias sobre el trabajo asignado, convirtiendo la evaluación en formativa además de aditiva. La fiabilidad de estos tipos de evaluaciones depende de muchos factores, entre los que cabe destacar una buena definición de los parámetros de evaluación, y dar al alumnado los conocimientos, las herramientas y el soporte necesario para que las puedan realizar de forma correcta.

### 2.1. Rúbricas

Una rúbrica, o matriz de valoración, es una herramienta de calificación formada por un conjunto de criterios ligados a los objetivos de aprendizaje usados para evaluar la actuación de los alumnos en la realización de determinadas tareas que pretendamos evaluar.

La rúbrica permite detallar los criterios específicos que se tendrán en cuenta a la hora de evaluar al alumno, en definitiva, hacer más transparente y fácil el proceso de evaluación.

Algunas de sus características más destacadas pueden ser:

- Facilita la evaluación al profesor.
- Define lo que se espera de un alumno en la realización de un trabajo.
- Reduce la sensación de ambigüedad que pueda tener un alumno cuando es evaluado.
- Permite ordenar los conocimientos en base a un juicio de valor que se establece y, por tanto, garantiza una evaluación más equitativa.

En el proceso de elaboración de una rúbrica se realizan un grupo de tareas muy diversas. Entre ellas es importante buscar las características que definen el trabajo que se ha de realizar y los contenidos sobre los que los alumnos han de evaluar su aprendizaje. Igualmente, deben quedar claramente detallados los objetivos, desempeños, competencias y comportamientos que han de demostrar los alumnos.

Finalmente, se han de establecer unos niveles de consecución de esos objetivos y competencias. Cada uno de estos niveles debe estar claramente detallado para que el alumno sepa establecer, por sí solo, el nivel al que él ha conseguido llegar. Toda esta información acabará mostrándose en forma de matriz donde las filas son los aspectos a evaluar, y

		Has de millorar	Satisfactori	Bé	Excel·lent
8	Sap cercar, seleccionar i organitzar la informació més rellevant sobre la història de la guerra d'Afganistan	No és capaç de distingir ni seleccionar la informació més rellevant.	Reconeix alguns punts rellevants, però mescla la informació important amb dades irrelevantes.	La informació és rellevant, però no està prou organitzada.	La informació seleccionada és la més rellevant. La presenta de manera molt clara i organitzada.
9	És capaç de sintetitzar la informació i escriure correctament textos guiats	Hi ha faltes ortogràfiques i el text està escrit sense coherència. La informació està copiada, no sintetitzada.	El text té certa coherència, però hi ha moltes faltes ortogràfiques. La informació no està ben resumida.	El text és coherent i hi ha poques faltes ortogràfiques. La informació està sintetitzada.	El text té coherència i cohesió. No hi ha faltes ortogràfiques. La informació està correctament sintetitzada.
10	Sap penjar la informació correctament al blog, de manera ordenada, amb les etiquetes que facilitin trobar la informació.  És capaç de corregir les deficiències de presentació més bàsiques	Ha penjat la informació de manera confusa. És difícil trobar els elements importants.	El blog té una presentació útil, però hi ha un excés d'informació. Només alguns elements són fàcils de trobar.	El blog té una presentació útil. Hi ha elements decoratius que ajuden a comprendre l'explicació. La majoria dels elements són fàcils de trobar.	El blog resulta molt atractiu i té una presentació molt útil. És fàcil trobar tots els elements importants. Els espais estan ben distribuïts i les etiquetes són emprades de manera eficaç per organitzar el material.

Figura 1: Ejemplo de una matriz de rúbrica.

las columnas el nivel conseguido. En la Figura 1 se puede ver un ejemplo de este tipo de matriz.

El proceso de realización de rúbricas es un trabajo cíclico, ya que una vez probadas las matrices con los alumnos, se deben revisar los criterios seleccionados para la graduación de los conocimientos. En cada uno de estos ciclos de revisión-prueba, se irá refinando y mejorando la matriz de la rúbrica.

Las rúbricas es el método considerado como el más adecuado para la realización de las actividades de autoevaluación. Existen otros métodos (como por ejemplo la elaboración de tests), pero no encajan tan bien, de forma general, en las actividades de este tipo ni, de forma particular, en el contexto de las asignaturas y trabajos que se han llevado a cabo en nuestros experimentos.

### 3. Metodología del experimento

El objetivo del estudio es conocer la respuesta del alumnado ante distintos niveles de motivación para la realización de tareas de autoevaluación. A continuación definimos más detalladamente el contexto de los experimentos llevados a cabo.

La autoevaluación de los alumnos se ha basado en el uso de rúbricas. El número de rúbricas utilizadas han sido tantas como grupos de distintos cursos estudiados. Cada uno de estos grupos pertenecen a materias distintas, por lo que las rúbricas no pueden ser,

en ningún caso, compartidas entre ellos, pero sí que estas han sido supervisadas por la misma persona. Por esto, podemos suponer que las mismas presentan ciertos niveles de homogeneidad, por lo que el factor referente a la calidad de las rúbricas no afectará al resultado del estudio.

En ningún momento se ha intentado analizar si el proceso de autoevaluación de los alumnos se ha llevado a cabo de forma correcta. Tampoco está dentro de los objetivos del estudio el estudiar si existe alguna correlación entre las notas de los alumnos y el hecho de haber realizado las autoevaluaciones.

Por tanto, el único parámetro estudiado es la respuesta, por parte de los alumnos, a distintos escenarios o actividades de autoevaluación planteados. Esta respuesta se ha medido en términos de número de alumnos que han llevado a cabo estas tareas. Únicamente, y a raíz de los resultados de una de las pruebas, se ha añadido un parámetro más al estudio: el período en el que se realiza la autoevaluación. Se ha estudiado si existe una relación directa entre el éxito de participación en las autoevaluaciones y la proximidad del periodo de exámenes.

En los experimentos han participado cinco profesores del Departamento de Matemáticas e Informática. Estos profesores han llevado a cabo las tareas de autoevaluación en seis grupos distintos de alumnos, todos ellos enmarcados en estudios técnicos, principalmente en la rama de informática (cinco de los seis grupos). El tamaño de los grupos es bastante variable, cubriendo casos de grupos muy reducidos (9 alumnos) hasta grupos de un número considerable (casi 50 alumnos). El total de alumnos que se han visto involucrados en los experimentos ha sido de 227. En el Cuadro 1 se puede ver más detalladamente los grupos que han intervenido.

Todos los experimentos se han llevado a cabo durante el segundo cuatrimestre del curso 2010/2011. Las autoevaluación se ha llevado a cabo sobre actividades que ya estaban contempladas y planificadas anteriormente en el desarrollo de las asignaturas. Añadiendo la realización de la rúbrica como una actividad adicional. La forma que el profesorado ha utilizado para motivar a los alumnos ha variado en los distintos grupos, y se han llevado a cabo tres tipos de motivación distinta.

La primera de ella, a la que llamaremos motivación única, consiste en comentar una única vez durante las clases de los alumnos la importancia, jus-

tificación y bondades de la autoevaluación. Se han explicado todos los aspectos importantes comentados previamente en la Sección 2. Este incentivo se ha llevado a cabo únicamente en una ocasión, a la hora de presentar la actividad, y se fijó una fecha de entrega. Tras esta primera sesión, no se volvió a recordar ni motivar al alumnado a llevar a cabo la autoevaluación.

Un segundo tipo de motivación, motivación periódica, está basada en los mismos términos que la anterior, pero se ha ido recordando de forma periódica, entre el momento de presentar la actividad y la fecha de entrega de la misma, la importancia de los procesos de autoevaluación y las ventajas que los alumnos pueden tomar de ellos.

Finalmente se utilizó una motivación incentivada a base de mejorar la calificación de la asignatura, en la que el hecho de realizar la actividad de autoevaluación se ve recompensada con un porcentaje de la nota final.

#### 4. Análisis de los resultados

El análisis de los resultados se ha realizado dividiendo los 6 grupos de alumnos estudiados en tres bloques. Los grupos que forman parte de cada bloque se han determinado por el tipo de motivación llevada a cabo (única, periódica o incentivada). Cada bloque cuenta con dos grupos experimentales (o grupos de alumnos) para controlar que los resultados no se desvirtúen si encontramos un grupo que pueda actuar de forma distinta a la habitual. Los grupos vendrán identificados por el mismo nombre utilizado para el tipo de motivación. Por lo que tendremos el bloque de motivación única (grupos B1 y C1), el bloque de motivación periódica (grupos B2 y C2) y el bloque de motivación incentivada (grupos B3 y C3).

Para realizar el análisis de los datos se tendrá en cuenta los datos siguientes: el número de trabajos entregados por cada grupo experimental; el número de autoevaluaciones llevadas a cabo. En el Cuadro reftabla:entregas podemos ver esta información.

Se ha realizado un análisis preliminar de la información mediante la comparación de los porcentajes de los trabajos entregados por grupos experimentales y bloques. El Cuadro 3 muestra información, resumiendo los porcentajes de entrega para los dis-

Id. grupo	Núm. alumnos matriculados	Curso	Estudios	Tipo asignatura	Id. profesor	Tipo de motivación
B-único	19	3	Inf. Sistemas	Obligatoria	W	Única
B-periódico	9	4	Ing. Informática	Troncal	W	Periódica
B-incentivado	43	5	Ing. Informática	Optativa	W	Incentivada
C-único	23	3	Inf. Sistemas, Inf. Gestión y Ing. Tec. Industrial	Optativa	X	Única
C-periódico	22	3	Inf. Sistemas	Obligatoria	Y	Periódica
C-incentivado	111	2	Inf. Gestión	Troncal	Z	Incentivada

Cuadro 1: Grupos en los que se ha llevado a cabo el experimento.

Grupo experimental	Cantidad	Bloque único	Bloque periódico	Bloque incentivado
B	Entregas	10	9	39
B	Autoevaluaciones	4	8	31
C	Entregas	15	12	333
C	Autoevaluaciones	0	0	289

Cuadro 2: Número de trabajos entregados y autoevaluaciones realizadas en cada uno de los grupos experimentales.

tintos grupos experimentales, por un lado, y para los bloques, por otro.

Posteriormente se ha realizado un test de Chi-cuadrado ( $\chi^2$ ) y de coeficiente de contingencia de Pearson [1]. Mediante estos dos métodos se puede comprobar si las variables estudiadas son estadísticamente independientes. El test de  $\chi^2$  nos ofrece la probabilidad de concurrencia, y el test de coeficiente de contingencia de Pearson normaliza los resultados entre 0 y 1. Con esos valores se puede comprobar la asociación o no de las variables.

Para el caso de  $\chi^2$  tenemos que el valor límite, generalmente aceptado, para considerar significativo o no un resultado es  $P=0,05$ . Cuando el P obtenido es menor que este valor, existen diferencias entre las variables estudiadas y, por tanto, la asociación es significativa. Para el caso del test de coeficiente de contingencia, el valor que marca la asociación se encuentra en 0,333. Se consideran variables diferentes cuando el valor, de nuevo, es menor [1].

El Cuadro 4 muestra los resultados de estos estudios. La primera columna de esta tabla nos muestra, para cada uno de los bloques, el valor resultante del test de  $\chi^2$ . El segundo valor numérico corresponde a la probabilidad con la que se acepta que la corre-

lación entre las variables es significativa, y por último, se muestra el coeficiente de contingencia de Pearson. La herramienta utilizada para realizar todos cálculos ha sido el paquete estadístico SPSS 18 [8].

Las hipótesis a comprobar mediante el uso de estas herramientas estadísticas son tres:

- La existencia de diferencias de comportamiento del alumnado entre los tres bloques estudiados. Según la hipótesis inicial, los tres bloques deben de mostrar diferencias a la hora de realizar las autoevaluaciones.
- La existencia de diferencias entre los grupos experimentales del mismo bloque. Nuestra hipótesis determina que si dos grupos experimentales del mismo bloque, se ven sometidos a una misma variable independiente, estos deben responder de la misma forma.
- La existencia de diferencias en la entrega de las autoevaluaciones a medida que se acerca la fecha de los exámenes. La hipótesis inicial considera la correlación entre la proximidad de los exámenes y el número de entregas de autoevaluaciones.

Bloque único		Bloque periódico		Bloque incentivado	
Grupo B	Grupo C	Grupo B	Grupo C	Grupo B	Grupo C
40%	0%	88,9%	0%	97,5%	86,8%
16%		38,1%		87,1%	

Cuadro 3: Porcentajes de autoevaluaciones entregadas.

Bloque	$\chi^2$	Correlación significativa	Coefficiente de contingencia
Único	7,143 (P=0,008)	P <0,01	0,008
Periódico	17,231 (P=0,000)	P <0,01	0,000
Incentivado	3,863 (P=0,049)	P <0,05	0,049

Cuadro 4: Resultados de los test  $\chi^2$  y de coeficiente de contingencia de Pearson para los bloques.

$\chi^2$	Correl. significativa	Coef. de contingencia
106,027 (P=0,00)	P <0,01	0,000

Cuadro 5: Resultados de los test  $\chi^2$  y de coeficiente de contingencia de Pearson para los grupos experimentales.

En el Cuadro 4 podemos ver que, para los tres bloques, la probabilidad de concurrencia (P) obtenidas en el test de  $\chi^2$  son menores que 0,05. Por tanto, podemos decir que no hay diferencia entre los tres bloques y, en consecuencia, la asociación entre los tres tipos de motivación es significativa. Estos resultados vienen contrastados con el segundo test realizado, el de coeficiente de contingencia de Pearson. Se puede ver que el valor obtenido con este test es menor a 0,333 en los tres casos, con lo que el segundo método contrasta los resultados obtenidos con el primero.

Las diferencias entre los distintos bloques parece que puede ser explicada por el tipo de incentivo que han recibido los alumnos. De esta forma quedaría claro que los alumnos, por sí solos, no realizarían las autoevaluaciones y que estás sólo las llevarían a cabo cuando se les reitera su importancia o cuando se les premia por su realización. Esto dejaría de manifiesto que los alumnos no tienen interiorizados los beneficios de la autoevaluación, y que necesitan de una motivación externa para llevarla a cabo. De esta forma, la primera de nuestras hipótesis estaría comprobada.

El análisis de los datos también nos muestra que las diferencias entre los grupos experimentales de un mismo bloque también existen y que, por tanto, la asociación entre los grupos experimentales también es significativa (Cuadro 5).

La primera explicación posible para este hecho podría venir dada por los profesores que han realizado los experimentos y el tipo de actividades que llevan a cabo a lo largo de la asignatura. Parece lógico pensar que en asignaturas donde se lleva a cabo una evaluación constante de las actividades y trabajos del alumno, estos están más acostumbrado a este tipo de actividades y están más acostumbrados a una docencia basada en el aprendizaje. Este hecho podría afectar a la tasa de autoevaluaciones realizadas. Los grupos experimentales B han sido evaluados por el mismo profesor, y todos ellos muestran unas tasas de autoevaluación mayores. En cualquier caso, la segunda de nuestras hipótesis no quedaría validada por los resultados.

Una segunda posibilidad es que los periodos en los que se han llevado a cabo la entrega de las autoevaluaciones influyan en su tasa de éxito. La recogida de las autoevaluaciones se ha llevado a cabo en diferentes fechas durante el segundo cuatrimestre del año, y no han coincidido entre ellas. De ahí viene la justificación del estudio realizado para contrastar nuestra tercera hipótesis.

Los datos del grupo experimental C-incentivado corresponden a la suma de tres actividades de autoevaluación sobre el mismo grupo de alumnos. De ahí los números tan elevados que se muestran (y que no corresponden con el número de alumnos de la

Entrega 1	Entrega 2	Entrega 3
92,8 %	86,5 %	81,1 %

Cuadro 6: Porcentajes de autoevaluaciones entregadas para las tres actividades del grupo C-incentivado.

$\chi^2$	Correl. significativa	Coef. de contingencia
6,652 (P=0,036)	P <0,05	0,038

Cuadro 7: Resultados de los test  $\chi^2$  y de coeficiente de contingencia de Pearson para el grupo C-incentivado.

asignatura). Para comprobar la validez de esta última hipótesis vamos a estudiar la evolución de la participación a lo largo de las tres autoevaluaciones realizadas. En los Cuadro 6 y 7 se pueden ver los datos correspondientes a las tres entregas.

De nuevo existen diferencias en la entrega de las autoevaluaciones por parte de un mismo grupo, por lo que se puede decir que la asociación en la entrega de las autoevaluaciones por parte del mismo grupo es significativa. Los resultados parecen mostrar que a medida que el período de exámenes se acerca, la tasa de éxito de entregas disminuye. Esto podría estar asociado, de nuevo, a que los alumnos no tienen interiorizados los beneficios de la autoevaluación, y se dedican a afrontar los exámenes de una forma más tradicional basada en el estudio. De esta forma, nuestra hipótesis sobre la existencia de una relación directa entre la época en que está programada la actividad de autoevaluación y la proximidad de los exámenes quedaría contrastada.

## 5. Conclusiones

En este artículo se muestra los resultados de un estudio sobre la influencia de los métodos de motivación utilizados por los profesores sobre las tasas de éxito de participación de los alumnos. Para ello se han propuesto actividades a distintos grupos de alumnos, y se les ha facilitado la posibilidad de autoevaluarlas. Los distintos grupos han sido motivados de distintas formas para que realizaran la autoevaluación. En algunos casos esta motivación ha sido mínima y única en el tiempo, en otros se ha realizado de forma periódico-

ca, y en otros grupos se ha incentivado añadiendo esta actividad a las evaluables dentro de la asignatura.

De los datos extraídos se han estudiado fundamentalmente dos aspectos: la relación entre el método utilizado para motivar a los alumnos, y el éxito de la actividad de autoevaluación; y la influencia del período en el que se han realizado las actividades sobre la tasa de éxito de la misma.

Las conclusiones extraídas son que los alumnos incentivados con un incremento de la calificación final de la asignatura llevan a cabo con más frecuencia las actividades de autoevaluación. También se muestra más probables en aquellos grupos a los que se les motiva periódicamente, pero con menor eficacia que en el caso de la utilización de incentivos. Del mismo modo se ha visto que la proximidad del período de exámenes influye negativamente en la realización de actividades de autoevaluación.

Como consecuencias del estudio también se ha visto que las actividades que lleva a cabo el profesor durante el año, y la frecuencia de actividades entregables que deben realizar los alumnos podría influir en el éxito de la tasa de entrega de autoevaluaciones. Esto es así porque se han mostrado diferencias entre grupos motivados mediante el mismo método, pero con distintos profesores.

A raíz de este trabajo se abre un gran número de posibilidades de exploración y de trabajos futuros. De entrada, el número de grupos y el número de alumnos no es todo lo grande como cabría desear. Pero las limitaciones de tiempo y de profesorado han restringido enormemente la obtención de más datos.

Al haberse centrado todo el trabajo en el segundo semestre del año, no se disponen de datos sobre el primero, y podría resultar interesante estudiar si existe alguna relación en este sentido. De la misma forma que resultaría interesante, si se dispusiera de datos suficientes, estudiar la relación entre la madurez de los alumnos (curso de las asignaturas) y la realización de este tipo de actividades de autoevaluación.

Resultaría igualmente interesante, ver si existen diferencias entre los resultados cuando estos son recogidos en el mismo período de tiempo, y cuando estos se recogen en distintos momentos del curso. La influencia del profesor también sería un punto a tener en cuenta y a estudiar más profundamente.

Finalmente, sería interesante estudiar la relación

entre la *cantidad* del incentivo recibido y la participación del alumnos. En otras palabras, hasta que punto participan más los alumnos de su autoevaluación a medida que el beneficio sobre la calificación final se va incrementando.

### Agradecimientos

Este trabajo ha sido financiado parcialmente por el Departamento de Matemáticas e Informática de la Universitat de les Illes Balears y por el proyecto "Estudi d'autoevaluació i coevaluació de l'EEES" financiado por la convocatoria de "Projectes d'innovació i millora de la qualitat docent 2010-2011" del Vicerectorado de Profesorado e Innovación Docente de la Universitat de les Illes Balears. Igualmente agradecer el trabajo de elaboración de rúbricas y recolección de datos a los profesores Isaac Lera y Carmen Erice.

### Referencias

- [1] Ballester, Lluís. *Bases metodològiques de la investigació educativa*. Edicions Universitat de les Illes Balears. Segunda edición. España. 2004.
- [2] Bauer, C., Figl, K., Derntl, M., Beran, P. P., and Kabicher, S. *The student view on online peer reviews*. In Proceedings of the 14th Annual ACM SIGCSE Conference on innovation and Technology in Computer Science Education (Paris, France, July 06 - 09, 2009). ITiCSE '09. ACM, New York, NY, 26-30.
- [3] Boud, David J.; Holmes, W. Harvey. *Self and Peer Marking in an Undergraduate Engineering Course*. IEEE Transactions on Education, vol. 24, issue 4, pp. 267-274.
- [4] Falchikov, Nancy; Goldfinch, Judy. *Student Peer Assessment in Higher Education: A Meta-Analysis Comparing Peer and Teacher Marks*. Review of Educational Research 70 (3): 287-322.
- [5] Fallows, S.; Chandramohan, B., *Multiple Approaches to Assessment: reflections on use of tutor, peer and self-assessment*, Teaching in Higher Education, 6(2), 229-246, 2001.
- [6] Gaillet, L. I. *A foreshadowing of modern theories and practices of collaborative learning: The work of the Scottish rhetorician George Jardine*. Paper presented at the 43rd Annual Meeting of the Conference on College Composition and Communication, Cincinnati OH, March 1992.
- [7] Gehringer, E. F. *Electronic peer review and peer grading in computer-science courses*. SIGCSE Bull. 33, 1 (Mar. 2001), 139-143.
- [8] IBM. *SPSS, Statistical Package for Social Sciences*. URL: <http://www.spss.com/>, (20/Mayo/2011).
- [9] Mirielli, E. J. *Using peer-evaluation in a website design course*. J. Comput. Small Coll. 22, 4 (Apr. 2007), 14-21.
- [10] Oliver Bernal, Javier; Canivell Castillo, Verónica. *Evaluación entre compañeros: estudio de su correlación con la evaluación del profesor*. Actas Jornadas de Enseñanza Universitaria de la Informática JENUI 2009. Barcelona 2009.
- [11] Riesco Albizu, M; Díaz Fondón, M. *La revisión entre iguales como herramienta de aprendizaje y evaluación en la asignatura de sistemas operativos*. Actas Jornadas de Enseñanza Universitaria de la Informática JENUI. Barcelona 2009.
- [12] Sitthiworachart, J. and Joy, M. *Effective peer assessment for learning computer programming*. SIGCSE Bull. 36, 3 (Sep. 2004), 122-126.
- [13] Sitthiworachart, Jirarat; Joy, Mike. *Self and Peer Assessment in School and University: Reliability, Validity and Utility*. In Book: Optimising New Modes of Assessment: In Search of Qualities and Standards: 55-87. 2003.
- [14] Sitthiworachart, J.; Joy, M., *Web-based peer assessment in learning computer programming*, Advanced Learning Technologies, 2003. Proceedings. The 3rd IEEE International Conference on , vol., no., pp. 180-184, 9-11 July 2003.
- [15] Sondergaard, H. *Learning from and with peers: the different roles of student peer reviewing*. In Proceedings of the 14th Annual ACM SIGCSE Conference on innovation and Technology in Computer Science Education (Paris, France, July 06 - 09, 2009). ITiCSE '09. ACM, New York, NY, 31-35.
- [16] Wolfe, W. J. *Online student peer reviews*. In Proceedings of the 5th Conference on information Technology Education (USA, October 28 - 30, 2004). CITC5 '04. ACM, New York.

# Combinando Moodle y tecnologías Web 2.0 para trabajar competencias transversales

Adelaida Delgado Domínguez

Departament de Ciències Matemàtiques i  
Informàtica  
Universitat de les Illes Balears  
Ctra de Valldemossa km 7,5  
07122 Palma  
adelaida.delgado@uib.es

## Resumen

Este artículo presenta la experiencia docente, llevada a cabo durante los dos últimos cursos en dos asignaturas, en la que se combinan utilidades, actividades y recursos de Moodle, con tecnologías y herramientas Web 2.0, para trabajar competencias transversales, especialmente el trabajo autónomo y el colaborativo, favoreciendo además la comunicación en línea.

En Moodle se han empleado las utilidades de calendario, calificaciones, grupos, y ficheros; los recursos de inserción de etiquetas, composición de páginas web, enlace a ficheros o sitios web y visualización de directorios; y las actividades de foros, tareas y wiki.

Las tecnologías y herramientas web 2.0 utilizadas son: blogs, wikis, suites ofimáticas, galerías de imágenes, de vídeos, de presentaciones, mapas personalizados, sindicación de contenidos y creación de páginas de inicio personalizadas.

## Summary

This paper presents a teaching experience based on two courses, where activities, resources and utilities of Moodle are combined with Web 2.0 technologies and tools to work generic competences, especially self-and collaborative learning and online communication .

In Moodle we used the calendar, qualifications, groups, files, tags, web pages, links to files or web sites and view directories, and the activities tasks, forums and wiki.

The web 2.0 technologies and tools used are: blogs, wikis, office suites, image galleries, videos, presentations, personalized maps, content syndication and personalized homepages.

## Palabras clave

Moodle, Web 2.0, competencias transversales, técnicas aprendizaje autónomo, trabajo en grupo.

## 1. Introducción

En la mayoría de los planes de estudios adaptados al EEES figuran competencias transversales (aquellas relacionadas con la formación de un estudiante universitario que se reconocerá en el mundo laboral cuando se deba enfrentar a una situación como profesional o investigador), las cuales se pueden categorizar en de 3 tipos: instrumentales, interpersonales y sistémicas [1].

La siguiente experiencia docente está encaminada a llevar a cabo un aprendizaje basado en un conjunto de competencias transversales [4] haciendo uso de utilidades, recursos y actividades Moodle, combinadas con tecnologías y herramientas Web 2.0.

Las competencias transversales que se han tenido en cuenta principalmente en esta experiencia son las siguientes:

**Instrumentales:** herramientas para el aprendizaje y la formación:

- Técnicas aprendizaje autónomo
- Análisis y síntesis
- Organización y planificación
- Comunicación escrita
- Capacidad de gestión de la información
- Toma de decisiones

**Interpersonales:** capacidades que permiten mantener una buena relación social:

- Trabajo en equipo
- Comunicación interpersonal

**Sistémicas:** relacionadas con la gestión de la totalidad de la actuación (visión de conjunto):

- Preocupación por la calidad

Para ello, en Moodle se han empleado las utilidades de calendario, calificaciones, grupos, informes y ficheros; los recursos de inserción de etiquetas, composición de páginas web, enlace a ficheros o sitios web y visualización de directorios; y las actividades de foros, tareas y wiki.

Las tecnologías y herramientas web 2.0 utilizadas son: blogs (Blogger), wikis (Google Sites y Google Docs), galerías de imágenes (Flickr y Picasa Web Albums), de vídeos (Youtube), de presentaciones (SlideShare), mapas personalizados (Google Maps), sindicación de contenidos (Google Reader) y creación de páginas de inicio personalizadas (iGoogle).

## 2. Contexto y aplicabilidad

Esta experiencia docente se ha llevado a cabo, durante los dos últimos cursos de manera similar, en 2 asignaturas de informática, una optativa para los estudios de la Diplomatura de Maestro (Introducción a la Informática), de 4'5 créditos con 31 alumnos, y otra obligatoria para la licenciatura de Administración de Empresas (Informática Aplicada), con 101 alumnos divididos en 2 grandes grupos.

Ambas asignaturas se imparten en modalidad semipresencial (con un 30% de no presencialidad), utilizándose Moodle como plataforma de teleeducación.

En ambas asignaturas, la Web 2.0 forma parte tanto de los contenidos como de la metodología. No sólo se trabaja a nivel conceptual e instrumental sino también a nivel actitudinal, teniendo en cuenta el lema "compartir, colaborar, comunicar, confiar", y fomentando el trabajo colaborativo y la creación de conocimiento social. Además también se analiza cómo aplicar las tecnologías y herramientas Web 2.0 a la Educación (en la asignatura de Magisterio) y en las empresas (en la asignatura de Administración de Empresas). El tiempo dedicado a todo ello equivale a un 36% de la asignatura Informática Aplicada (el 70% de manera presencial y el 30% no presencial). En el caso de la asignatura de Magisterio ocupa el 100% de las clases

presenciales, dado que es una asignatura que consta de sólo 15 sesiones presenciales, interrumpidas a la mitad del cuatrimestre durante dos meses, porque los alumnos han de realizar prácticas en centros de Educación Primaria o Infantil (*practicum*).

En ambas asignaturas, los temas dedicados a la Web 2.0, se complementan con otros temas adicionales más específicos de sus estudios. De entre esa lista de temas cada grupo ha de elegir 2, y sus contenidos han de ser desarrollados por los propios estudiantes de manera no presencial. En la asignatura de Informática Aplicada, como además hay todo un bloque obligatorio de ofimática (equivalente al 46% de la asignatura), se utilizan el procesador de textos y la herramienta de presentaciones, para editar uno de esos temas (con un formato especificado por el profesor que contemple las diversas posibilidades de edición).

En ambas asignaturas, por tanto, como parte de la evaluación, los alumnos han de realizar diversos trabajos en pequeños grupos (de 3 a 5 personas), además de otros individuales, y para ello utilizarán tecnologías y herramientas Web 2.0, así como la plataforma Moodle.

A pesar de tratarse de asignaturas de planes antiguos, se ha querido aplicarles la idea de competencias transversales y de evaluación continua, a fin de ir ensayando metodologías, de manera voluntaria para adaptarlas a las asignaturas de grado, cuando llegue el momento.

A continuación se mostrarán diversas estrategias seguidas para el aprendizaje autónomo del alumno (y algunas competencias transversales relacionadas), y también las utilizadas para el trabajo colaborativo y la creación de conocimiento social (y otras competencias transversales relacionadas).

## 3. Estrategias para el aprendizaje autónomo

### 3.1. Trabajando las competencias de gestión del tiempo, y de organización y planificación

Para organizarse el trabajo autónomo, los alumnos cuentan con el calendario de Google[2], donde el profesor informa de qué se va a hacer cada día de clase presencial y las fechas de entrega de las actividades no presenciales. En el calendario se

incluyen los enlaces a los recursos necesarios, tanto para el seguimiento de las clases presenciales (de esta manera si algún alumno no puede asistir un día determinado, puede realizar la sesión de manera autónoma), como para llevar a cabo las actividades en línea.

Los enlaces del calendario pueden llevar a visualizar:

- un determinado directorio del servidor de Moodle para examinar ficheros (por ejemplo el directorio de las prácticas entregadas el curso anterior)
- páginas web creadas expresamente para la asignatura en Moodle (por ejemplo conteniendo diversos vídeo tutoriales de Youtube, o explicaciones específicas de una actividad concreta)
- ficheros enlazados para su descarga (en especial dossiers de los temas integrantes de la Web 2.0 en PDFs con el enunciado de la práctica asociada, presentaciones realizadas por el profesor, ficheros de trabajo para las prácticas de ofimática, etc.)
- páginas externas a Moodle, (por ejemplo recursos complementarios de la asignatura, o las páginas de las herramientas en línea

El alumno puede utilizar también ese calendario para hacer sus propias anotaciones y planificarse el estudio de la asignatura.

### 3.2. Trabajando las competencias de análisis, síntesis, y comunicación escrita

Cada alumno lleva un diario de la asignatura, editado en un blog con Blogger. En este blog los alumnos han de introducir como mínimo una entrada semanal, en la cual se informe de qué van aprendiendo en las clases presenciales y cuáles son los adelantos en los trabajos en grupo.

Escribir ayuda a estructurar el pensamiento. De esta manera los alumnos analizan lo que se ha hecho en clase, así como la documentación que se les ha proporcionado en línea y escriben un resumen de manera sintetizada. También cabe la posibilidad de que se hagan preguntas, o de que aporten ellos mismos más recursos.

En el blog pueden insertar presentaciones (por ejemplo de SlideShare), o vídeos, (por ejemplo de Youtube), o documentos, (por ejemplo de Scribd), e imágenes (por ejemplo de Picasa Web Albums). En caso de que alguno de estos recursos sea de creación propia, han de asignarles una licencia

Creative Commons, que también incorporan al blog para ceder derechos que deseen. Con todo ello complementan los contenidos vistos en clase y crean su propio conocimiento.

También pueden realizar volcados de pantalla del trabajo realizado en clase e incorporarlo como imágenes en el blog. Para ello se les recomienda utilizar complementos como el ScreenGrab o el FireShot para Firefox.

Los alumnos que no hayan asistido a las clases presenciales durante esa semana, tienen igualmente que crear la entrada correspondiente del blog tras realizar la/s sesión/es de manera autónoma.

Las entradas del blog no han de contener faltas de ortografía, han de contener un título descriptivo, el desarrollo del tema tratado en clase, y además etiquetas asociadas que sintetizen, a modo de palabras clave, el contenido.

### 3.3. Trabajando individualmente la competencia de gestión de la información

Durante el curso los alumnos tienen que buscar información de los temas a desarrollar o hacer el seguimiento de noticias de actualidad sobre tecnologías de la información y comunicación, y en particular éstas ligadas a la Educación o a las empresas (según la asignatura). Para gestionar esa información utilizan 3 tipos de herramientas Web 2.0:

- Un agregador o lector de canales RSS (Google Reader) en el que reciben las novedades de las webs o blogs a los que se subscriben (véase Figura 1)



Figura 1. Ejemplo de suscripciones a canales RSS con Google Reader, de un alumno de Administración de Empresas

- Un servicio de creación de páginas de inicio personalizadas (iGoogle), en el que se organizan los sitios que suelen visitar a menudo, los enlaces a las diversas herramientas que utilizamos en el curso, las suscripciones a canales RSS (si les es más cómodo tenerlas aquí que en Google Reader), y multitud de gadgets como por ejemplo notas autoadhesivas, listado de tareas pendientes, un traductor, un convertidor de moneda, etc., todo ello clasificado en pestañas, para una mejor organización (véase Figura 2).



Figura 2. Página de inicio personalizada (en iGoogle) de un alumno para organizarse la información más utilizada y las suscripciones RSS

- Un servicio de gestión de marcadores sociales (en Mister Wong) donde incorporan en línea los enlaces a los recursos de su interés que van encontrando, en su usuario particular (véase Figura 3).



Figura 3. Ejemplo de almacenamiento en línea de los marcadores individuales de una alumna (Mister Wong)

### 3.4. Entrega y evaluación del trabajo autónomo

Algunas de las actividades individuales se entregan a través de entradas del blog, redactando una entrada o aportando volcados de pantalla de la actividad realizada. El profesor se puede suscribir vía RSS a los blogs de los alumnos para realizar un seguimiento de su evolución y anotar la retroalimentación en los comentarios de las entradas. En caso de ser una clase muy numerosa, en vez de suscribirnos a las novedades de sus blogs, se puede pedir a los alumnos que anoten la dirección de su blog en una tabla (editada en un wiki de Moodle) [3] y avisarles con qué periodicidad o en que fechas se entrará a hacer la retroalimentación de su trabajo para que la tengan a punto.

Para otras actividades individuales, se requiere la utilización de la actividad *Tareas* de Moodle, cuya fecha de entrega va ligada al calendario. Para realizar la retroalimentación se emplean las *Calificaciones* que van asociadas a la tarea, pudiendo dar una calificación numérica, o cualitativa mediante un mensaje que se puede enviar directamente al correo electrónico.

## 4. Estrategias para el trabajo en grupo

### 4.1. Formación de los grupos

Para empezar, los alumnos han de escoger dos temas, de una lista de unos 25 propuestos por el profesor, para desarrollarlos y darles formato, a uno con una herramienta colaborativas en línea (Google Sites) y al otro con una herramienta ofimática en línea (Google Docs), en el caso de los estudiantes de Magisterio, y en Word y Powerpoint en el caso de los estudiantes de Administración de Empresas.

Para formar los grupos, se utiliza una tabla de comunicación bidireccional, editada en la herramienta wiki de Moodle. Así los alumnos se pueden apuntar desde cualquier lugar y constituir el grupo de acuerdo con el tema elegido. Inicialmente sólo deben poner en la tabla su nombre en una celda, junto al tema elegido, más adelante utilizarán esa misma tabla para informar de la ubicación de sus trabajos en grupo.

### 4.2. Trabajando la competencia de comunicación interpersonal y la toma de decisiones entre los miembros dentro de cada grupo

Para los alumnos de los estudios de Magisterio el profesor se encarga de crearles una red social de tipo vertical y privada (en el servicio Grou.ps) para que se puedan comunicar entre sí y tomar decisiones acerca de la organización del trabajo, especialmente durante los 2 meses en que las clases presenciales se suspenden debido a sus prácticas en centros de Primaria e Infantil. Los alumnos que se incorporan a esta red no pueden entablar relación más que con los miembros de la propia red educativa (eliminando así el ruido existente en redes horizontales tipo Facebook). Y al ser privada, la privacidad e intimidad de sus miembros está asegurada. Además en Grou.ps, los alumnos pueden crear sus propios subgrupos en la red social para comunicarse con los miembros de su grupo de trabajo (véase Figura 4).

Un miembro del grupo se encarga de crear el subgrupo e invitar a participar al resto de miembros. También le ponen un logotipo identificativo de uno de los temas escogidos y un nombre. Aquí disponen de herramientas para compartir imágenes y vídeos, crear foros de discusión, escribir en un muro y hablar por chat.

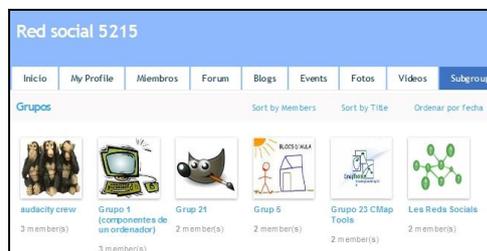


Figura 4. Red social privada con subgrupos para la comunicación interpersonal en línea de los alumnos

Los alumnos de los estudios de Administración de Empresas, en lugar de la red social, disponen en Moodle de foros de discusión separados por grupos.

La red social ha demostrado ser mucho más rica y motivadora como herramienta de comunicación y para compartir recursos que un simple foro en Moodle

### 4.3. Trabajando en grupo la competencia de gestión de información

Para desarrollar los temas elegidos, los alumnos deben buscar información en la red y anotar los enlaces que encuentren interesantes. Para compartir esta información utilizamos marcadores sociales, con la herramienta Mister Wong. Cada alumno puede tener allí sus marcadores personales con un usuario propio; pero además, deben crearse ellos mismos un grupo al cual se han de subscribir todos los miembros del equipo para compartir los enlaces del tema que deben desarrollar. Dentro de cada grupo de Mister Wong disponen de un foro de discusión que les permite coordinarse en este proceso de búsqueda de información (véase Figura 5).

### 4.4. Trabajando en grupo la competencia de comunicación escrita y la preocupación por la calidad: edición colaborativa

La edición del desarrollo de uno de los temas también se ha de realizar de manera conjunta, en forma de página web, en un wiki (con la herramienta colaborativa en línea Google Sites).

En el caso de los alumnos de Magisterio utilizan una segunda herramienta de edición colaborativa (Google Docs).

Ambos tipos de documento quedan públicos (para contribuir al conocimiento social), por lo

que han de preocuparse por la calidad de sus publicaciones.

#### 4.5. Evaluación de los trabajos en grupo

Una vez finalizado el trabajo en grupo, los alumnos notifican al profesor su URL, y la lista de enlaces utilizada a modo de fuentes de documentación, anotando estos datos en el wiki de Moodle que habían utilizado inicialmente para escoger el tema.

La nota de los trabajos en grupo tiene una componente grupal, en base a la calidad del contenido del tema desarrollado, que es responsabilidad de todos los miembros, y otra componente individual, según el grado de implicación en el grupo (cantidad de enlaces aportados al grupo de Mister Wong, y grado de participación en la escritura colaborativa del wiki de Google Sites y Google Docs).

- En el caso de Mister Wong, pulsando sobre un miembro de un grupo se puede ver de manera agrupada todos los enlaces aportados al mismo.



Figura 5. Seguimiento de la aportación de los marcadores de un usuario a un grupo

- En Google Sites se puede ver el historial *Actividad Reciente del Sitio* para saber qué ha aportado cada alumno. Este tipo de información también se puede saber página por página del sitio.

- Y en Google Docs, desde el menú Archivo se puede acceder al historial de revisiones y saber quién ha realizado cada una de ellas e incluso restaurarla.

## 5. Resultados y conclusiones

Se han presentado diversas estrategias y herramientas para fomentar principalmente el trabajo autónomo y el colaborativo en línea, que además permiten trabajar otras competencias transversales como el análisis y síntesis, la organización y la planificación, la toma de decisiones, la comunicación escrita, la gestión de la información, la preocupación por la calidad o la comunicación interpersonal.

De las tablas 1 y 2, que muestran, para cada asignatura, un extracto de las encuestas de evaluación docente realizadas a los alumnos durante el curso 2008/09, en el que se aplicó una metodología más tradicional, y las del curso 2090/10 en el que se aplicó la metodología descrita en el artículo (no se dispone todavía de los resultados del curso 2010/11) se pueden extraer las siguientes conclusiones:

- El nuevo ritmo del desarrollo del programa de la asignatura facilita más el aprendizaje
- Los criterios y procedimientos de evaluación están mejor presentados y explicados
- Se fomenta más el espíritu crítico
- Los alumnos se sienten más participativos con el nuevo estilo de actividades
- Los materiales proporcionados a los alumnos les ayudan más en su aprendizaje
- Los recursos audiovisuales y otros medios utilizados por el profesor favorecen más el aprendizaje (comparado con otros profesores), aunque esa característica se mantiene bastante similar de un curso a otro
- La carga del alumno es asumida como mayor para los alumnos de "Informática Aplicada" pero menor para los de "Introducción a la Informática", que ya están acostumbrados a utilizar tecnologías de la información en su aprendizaje.

Por otra parte, también se han podido llevar a cabo actividades en grupo sin la necesidad de reunirse físicamente, utilizando las herramientas de edición en línea y las estrategias de formación de grupos y de gestión de la información.

Asignatura <b>"Introducción a la Informática"</b> Estudios: Maestro	Curso 2008/09		Curso 2009/10		Mejora media del profesor
	Media del profesor	Media del Dpto	Media del profesor	Media del Dpto	
El programa se desarrolla a un ritmo que facilita el aprendizaje	<b>4.91</b>	6.44	<b>7.33</b>	6.65	<b>+2.42</b>
Presenta y explica los criterios y procedimientos de evaluación	<b>6.04</b>	6.94	<b>8.05</b>	7.06	<b>+2.01</b>
Fomenta el espíritu crítico	<b>4.77</b>	6.22	<b>6.14</b>	6.46	<b>+1.37</b>
Incita al estudiante a participar en las actividades del curso	<b>5.36</b>	6.43	<b>6.71</b>	6.67	<b>+1.35</b>
Los materiales de estudio ayudan en el aprendizaje	<b>6.41</b>	6.67	<b>7.75</b>	6.79	<b>+1.34</b>
La utilización que hace de los recursos audiovisuales y de otros medios favorece el aprendizaje	<b>7.27</b>	6.30	<b>8.14</b>	6.50	<b>+0.87</b>
Por los créditos que tiene esta asignatura, la carga de trabajo es elevada	<b>6.24</b>	6.85	<b>5.75</b>	6.78	<b>-0.49</b>

Tabla 1. Comparación de las encuestas de evaluación docente de los cursos 2008/09 y 2009/10 para la asignatura "Introducción a la Informática"

Asignatura <b>"Informática Aplicada"</b> Estudios: Administración y Dirección de Empresas	Curso 2008/09		Curso 2009/10		Mejora media del profesor
	Media del profesor	Media del Dpto	Media del profesor	Media del Dpto	
El programa se desarrolla a un ritmo que facilita el aprendizaje	<b>6.13</b>	6.44	<b>6.90</b>	6.65	<b>+0.77</b>
Presenta y explica los criterios y procedimientos de evaluación	<b>7.50</b>	6.94	<b>7.77</b>	7.06	<b>+0.27</b>
Fomenta el espíritu crítico	<b>4.74</b>	6.22	<b>6.29</b>	6.46	<b>+1.55</b>
Incita al estudiante a participar en las actividades del curso	<b>5.83</b>	6.43	<b>6.89</b>	6.67	<b>+1.06</b>
Los materiales de estudio ayudan en el aprendizaje	<b>6.88</b>	6.67	<b>7.77</b>	6.79	<b>+0.89</b>
La utilización que hace de los recursos audiovisuales y de otros medios favorece el aprendizaje	<b>8.08</b>	6.30	<b>8.03</b>	6.50	<b>-0.05</b>
Por los créditos que tiene esta asignatura, la carga de trabajo es elevada	<b>6.58</b>	6.85	<b>7.16</b>	6.78	<b>-0.07</b>

Tabla 2. Comparación de las encuestas de evaluación docente de los cursos 2008/09 y 2009/10 para la asignatura "Informática Aplicada"

### Agradecimientos

A todos mis alumnos por su participación activa en mis asignaturas.

### Referencias

- [1] González, Julia; Wagenaar, Robert. *Tuning Educational Structures in Europe*. Universidad de Deusto y de Gronigen, 2003. [[http://www.relint.deusto.es/TUNINGProject/spanish/doc\\_fase1/Tuning%20Educational.pdf](http://www.relint.deusto.es/TUNINGProject/spanish/doc_fase1/Tuning%20Educational.pdf)]
- [2] Universitat de les Illes Balears. *Manuales. Calendario*. [[http://www.uib.es/ca/infosobre/serveis/complementaris/campusextens/\\_doc/bonespract/proyec-manuales/castellano/calendari1\\_9\\_castellano.pdf](http://www.uib.es/ca/infosobre/serveis/complementaris/campusextens/_doc/bonespract/proyec-manuales/castellano/calendari1_9_castellano.pdf)]
- [3] Universitat de les Illes Balears. *Manuales. Wikis*. [[http://www.uib.es/ca/infosobre/serveis/complementaris/campusextens/\\_doc/bonespract/proyec-manuales/castellano/wiki.pdf](http://www.uib.es/ca/infosobre/serveis/complementaris/campusextens/_doc/bonespract/proyec-manuales/castellano/wiki.pdf)]
- [4] Villa, A; Poblete, M. *Aprendizaje basado en competencias. Una propuesta para la evaluación de las competencias genéricas*. Universidad de Deusto, 2008.

# Proceso de Reificación de las Competencias Generales y Específicas para el Grado en Ingeniería Informática y Definición de un Plan de Métricas de Evaluación de dichas Competencias

L.E. Sánchez, D.G. Rosado, D. Mellado, A. Santos-Olmo, E. Fernández-Medina  
Universidad de Castilla-La Mancha, Grupo de Investigación GSyA. Ciudad Real.  
{Luise.Sanchez; David.GRosado; Daniel.Mellado; Antonio.Santos-Olmo; Eduardo.FdezMedina}@uclm.es

## Resumen

El momento actual en el que Europa se encuentra inmersa en el proceso de convergencia de la educación superior es fundamental para el futuro de algunas carreras como la Ingeniería Informática, y por ello es muy importante ser capaces de adaptar los nuevos planes de estudio a las necesidades reales del mercado en este sector. Lo que se plantea en este artículo es desgranar las competencias generales y específicas del Grado en Ingeniería Informática y obtener métricas para valorar dichas competencias, de modo que se ofrezca un acercamiento mucho más concreto y detallado con las asignaturas, y consecuentemente pueda justificarse adecuadamente la forma en que las asignaturas permiten alcanzar parcial o completamente las competencias para el grado, ofreciendo a las empresas un mayor conocimiento sobre las competencias reales que se han obtenido y el grado de obtención de las mismas.

## Summary

The present moment in which Europe finds itself at the convergence of higher education is critical to the future of some races, such as Computer Engineering, and is therefore very important to be able to adapt new curricula to the needs actual market in this sector. At issue in this article is scrutinizing the general and specific skills Degree in Computer Engineering and obtain metrics to assess these skills, so as to provide a much more concrete and detailed subjects, and consequently it can be justified adequately how the subjects can partially or completely achieve the competencies for the degree, giving companies a greater understanding of the real powers that have been obtained and the degree of obtaining them.

## Palabras clave

EEES, Grado en Ingeniería Informática, Competencias Generales, Competencias Generales Específicas, Métricas, Certificaciones profesionales.

## 1. Introducción

Europa se encuentra actualmente en un momento crucial, en que se decidirá el futuro de muchos aspectos de su sistema productivo e innovador, y en el que el éxito o fracaso dependerá de la correcta definición de los nuevos planes de estudio. Actualmente, Europa se encuentra inmersa en el proceso de convergencia de la educación superior, que es fundamental para el futuro de algunas carreras como la Ingeniería Informática, y por ello es muy importante ser capaces de adaptar los nuevos planes de estudio a las necesidades reales del mercado en este sector. En el caso de la Ingeniería Informática, las empresas y los profesionales están demandando perfiles cada vez más especializados y que se adapten a una o varias certificaciones profesionales internacionales. Por lo tanto, es muy importante que los nuevos estudios estén muy enfocados a las necesidades profesionales, sin perder el rigor científico exigible en una ingeniería, y para conseguir este objetivo es fundamental que estos nuevos planes de estudio tengan una orientación que facilite la obtención de certificaciones profesionales. En este artículo se hace un análisis sobre la orientación profesional aplicada en el perfil de Ingeniería del Software del Grado en Ingeniería Informática de la UCLM, que resulta aplicable a otros muchos grados en informática en el panorama europeo.

El proceso de elaboración de las memorias de grado (en concreto en el caso del Grado en Ingeniería Informática), se ha basado en un

conjunto de competencias generales y específicas, que en la mayoría de los casos entrañan un alto nivel de abstracción y ambigüedad. Este hecho ha dificultado considerablemente la forma de asociar asignaturas con competencias y, en todo caso, el nivel de ambigüedad de las competencias ha quedado compensado con el margen que ofrece el hecho de definir las asignaturas en términos de descriptores generales que después, en la fase de implantación del plan de estudio, serán concretados adecuadamente.

Lo que se plantea en este artículo es crear unas nuevas bases que permitan desgranar las competencias generales y específicas del Grado en Ingeniería Informática, de modo que se ofrezca un acercamiento mucho más concreto y detallado con las asignaturas, y consecuentemente que pueda justificarse adecuadamente la forma en que las asignaturas permiten alcanzar parcial o completamente las competencias para el grado. Esto ayudará a especificar los contenidos y estrategias docentes de las asignaturas a medida que se va implantando el grado, de forma que se plantee un aprendizaje real basado en competencias en lugar de un planteamiento poco riguroso en el trato de las mismas.

Junto al trabajo de concreción de las competencias, se pretende plantear un plan de métricas de evaluación con dos objetivos fundamentales. En primer lugar, nos permitirán afinar en la definición detallada de las asignaturas (contenidos, actividades, prácticas, técnicas docentes, etc.) para garantizar que todas las competencias son abordadas adecuadamente por una o varias asignaturas, garantizando por lo tanto su correcto tratamiento. En segundo lugar, dichas métricas, convenientemente tratadas, pueden ser utilizadas para evaluar la consecución de las competencias por parte de los alumnos, es decir, se podrán utilizar como un componente que permita guiar la evaluación de las asignaturas y, de manera transitiva, la evaluación de las competencias.

La investigación que se está realizando se está basando en planteamientos relacionadas con aspectos académicos y complementados con importantes conexiones con el mundo profesional. Con ello se pretende mejorar la conexión entre el origen de las competencias, que aborda objetivos basados en las necesidades profesionales, y la aportación académica que ofrece la Universidad.

Esta investigación, ha supuesto la creación de equipos docentes multidisciplinares, con experiencia académica pero también con experiencia profesional. Los resultados están siendo aplicados y validados en el Grado de Ingeniería Informática del Campus de Ciudad Real de la Universidad de Castilla-la Mancha.

El artículo estará formado por cinco secciones: En la primera pondremos en contexto el momento actual de creación de planes de estudio. En la segunda sección se analizará la estructura general del plan de estudio para la Ingeniería Informática propuesta en la Universidad de Castilla-la Mancha. En la tercera sección se analizarán los métodos que se han utilizado para la investigación. En la cuarta sección se analizarán en detalle los objetivos que se han perseguido durante la investigación. Finalmente, en la última sección describiremos las principales conclusiones obtenidas hasta el momento durante la investigación.

## 2. Estado del arte

El Espacio Europeo de Educación Superior (EEES) se inicia con la Declaración de la Sorbona de 1998, que destacó el papel de las Universidades en el desarrollo de la dimensión cultural y de la Europa del conocimiento, y se amplía con las Declaraciones de Bolonia (Junio de 1999), de Praga (2001) y de Berlín (Septiembre de 2003) y Bergen (Mayo de 2005). En ellas se acordó promover y desarrollar en los países participantes la reforma de la estructura y la organización de las enseñanzas universitarias para estimular la construcción de un Espacio Europeo de Educación Superior con el objetivo de favorecer la movilidad y las oportunidades de empleo, y además hacer que estos nuevos planes de estudio y su implantación, se adaptasen a las demandas de las empresas [1], de forma que sirvan para hacer que los nuevos profesionales aumenten la productividad del tejido empresarial Europeo [2].

Actualmente, la mayor parte de los grados ya se han definido, pero gran parte de las Universidades Europeas se encuentran en pleno proceso de implantación de los nuevos planes de estudio del Grado en Ingeniería Informática, basándose para ello en las intensificaciones propuestas por la ACM [3], las cuales están muy

orientadas a competencias excesivamente complejas y difusas.

En el caso del grado en ingeniería informática, los nuevos planes,+ se han orientado a la existencia de un grado único con cinco especialidades o intensificaciones. Estas cinco intensificaciones se corresponden con las Tecnologías Específicas de la Resolución de 8 de junio de 2009 de la Secretaría General de Universidades, por la que se da publicidad al Acuerdo del Consejo de Universidades que se establece recomendaciones para la propuesta por las Universidades de memorias de solicitud de títulos oficiales del ámbito de la Ingeniería Técnica Informática (BOE Num. 187 del 4/8/2009), y las propuestas por la ACM [3], y que son: Ciencias de la Computación [4], Ingeniería del Software [5], Ingeniería de Computadores [6], Sistemas de Información [7] y Tecnologías de la Información [8].

Actualmente muchas instituciones e investigadores están trabajando para unificar y complementar el grado de ingeniería informática, tomando como base el modelo USA [9] o el modelo Europeo [10]. Algunas investigaciones han considerado que el problema no estaba tanto en el contenido de los dominios sino en el mecanismo de aprendizaje, centrándose en buscar metodologías de enseñanza ágiles [11]. Pero uno de los grandes problemas es que casi todas las investigaciones se han centrado en la definición de los planes, dejando de lado el proceso de implantación.

### 3. Grado en Ingeniería Informática en la UCLM

En el caso de la UCLM (Universidad de Castilla-La Mancha), la nueva propuesta del plan de estudios (ver figura 1) está dividida en un conjunto de bloques, orientados a la obtención de un título que por una parte se centrará en aspectos generalistas, haciendo que el estudiante adquiera al menos las competencias transversales de formación básica, comunes a la rama de informática, y por otra parte las competencias de al menos una de las especializaciones recomendadas por la ACM (de las 4 ofertadas en la UCLM).

ECTS	Estructura del Título				Mod
12	Trabajo fin de grado				
24	Optatividad				
48	Ing. Del Software	Tecnologías Información	Ing. De Computador	Computación	
36	Formación complementaria para la rama de Ingeniería Informática				3
60	Formación común para la rama de Ingeniería Informática				2
60	Formación básica para la Ingeniería				1

Figura 1. Estructura del Título de Ingeniero en Informática de la UCLM

Para poder entender mejor la problemática planteada en esta investigación, podemos centrarnos en la intensificación de Ingeniería del Software propuesta para el nuevo grado en ingeniería informática, que está basada en la "Guía para la creación del Cuerpo de Ingeniería de Software para el Conocimiento (SWEBOK)" [12, 13], donde se definen las competencias y conocimientos que según el IEEE un Ingeniero del Software debería haber obtenido al finalizar los estudios (ej: proyectos de Ingeniería del Software, Seguridad y Auditoría,...). El problema de esas competencias es que son complejas y difusas a la hora de poder aplicarlas, tanto para los alumnos como para las empresas, y tampoco permiten responder a preguntas como ¿Los conocimientos asociados con estas competencias han sido realmente obtenidos por el alumno? ¿En qué medida han sido obtenidos? ¿Son las competencias que necesitan realmente las empresas?

Y es en este proceso de cambio, definición e implantación de los nuevos planes de estudio donde está el punto crítico para el futuro de algunos estudios tan nuevos, tan cambiantes y de los que depende tanto el progreso de la sociedad como es el caso de la Ingeniería Informática [14]. Por lo tanto, es muy importante ser capaces de adaptar los nuevos planes de estudio a las necesidades reales del mercado [15], siendo capaces de implantarlos de una forma correcta que permita alinearlos con las competencias a las que se orientan y a las necesidades de las empresas. En el caso de la Ingeniería Informática, las empresas y los profesionales están demandando perfiles cada vez más especializados [16], por lo que es deseable que los futuros graduados cuenten con una o varias certificaciones profesionales [17,

18] que les permitan reforzar las competencias obtenidas y que éstas tengan además un carácter internacional [19], pero también que las empresas sean capaces de identificar claramente las competencias que han obtenido los alumnos y en qué medida han sido obtenidas. Por lo tanto es muy importante que los nuevos estudios estén muy enfocados a poder obtener una serie de competencias objetivas y medibles, y que estas estén alineadas con las necesidades profesionales [20], sin perder el rigor científico exigible en una ingeniería. Para conseguir este objetivo es fundamental que la implantación de estos nuevos planes de estudio tenga una orientación que facilite la identificación de sub-competencias medibles, que puedan vincularse con los contenidos de las asignaturas, de forma que pueda determinarse en qué medida una asignatura contribuye al cumplimiento de la competencia, y en qué medida un alumno consigue dicha competencia.

Por ello, es muy importante que durante el proceso de implantación del grado seamos capaces de pasar de unas competencias excesivamente complejas y genéricas a un conjunto de sub-competencias perfectamente definidas, que puedan relacionarse con los contenidos de las asignaturas y que sean medibles a partir de un conjunto de actividades asociadas a los contenidos de las mismas, que permitirán mostrar a las empresas que las asignaturas que componen las nuevas áreas de especialización están ofreciendo realmente lo que el mercado demanda [21], estableciendo mapas de relación entre los contenidos de estas asignaturas, las competencias del grado y los contenidos de las principales certificaciones profesionales que el mercado está demandando. Conseguir enlazar en la práctica con las necesidades formativas reales es un reto que ofrece muchas ventajas, tanto para los profesionales, que obtendrán una inserción laboral más directa y con mejor proyección, como para las propias empresas, que verán satisfechas sus necesidades mucho más rápidamente, y además en un contexto internacional. Existen estudios centrados en las TIC que demuestran que las personas que completaron su currículo con certificaciones profesionales, que les permitían explotar aquellas competencias para las que se habían mostrado más aptos, obtuvieron mejores salarios, principalmente en los casos en que

complementaron las certificaciones técnicas con certificaciones empresariales [22, 23].

Por último, no debemos olvidar que el ayudar a los alumnos a comprender en detalle los requerimientos de obtención de una competencia, que además ha sido validada con las necesidades reales de la empresa, supone una ventaja competitiva, ya que permite al alumno tomar una mejor orientación laboral al finalizar sus estudios, que también se traduce en una mejora de la productividad de la empresa al poder contratar a los alumnos más adecuados para las competencias buscadas, lo que se traduce en mejoras salariales que pueden superar el 10% [24].

#### 4. Metodología para la investigación

Para la realización de la investigación se va a utilizar el método de investigación denominado “Investigación-Acción”, que nos permitirá alinear en todo momento la investigación con un objetivo práctico y realista que, además, tenga un impacto directo en la mejora del sector privado. Y por otro lado, utilizando el enfoque GQM (Goal-Question-Metric) para obtener métricas adecuadas. A continuación se exponen ambos métodos de forma detallada.

##### 4.1. Investigación en Acción:

Considerando que en este proyecto hay un componente importante vinculado a conocer las necesidades reales de las empresas privadas y, por tanto, donde uno de los aspectos más valiosos es contar con una conexión directa con las necesidades reales provenientes de la empresa a la hora de seleccionar un candidato adecuado para cubrir una competencia determinada, el método de investigación que, a nuestro juicio es el más adecuado es una aproximación a “investigación-acción”. Uno de los objetivos que tenemos en el proyecto de innovación es intentar reducir la barrera tan habitual existente entre la teoría y la práctica, haciendo converger el aprendizaje que se realiza en las universidades con necesidades reales demandadas por las empresas.

Para abordar este problema cada vez es más habitual utilizar métodos de investigación cualitativos, que ofrecen ciclos iterativos de aplicación y refinamiento de las teorías construidas en entornos reales que permiten

obtener interesante información para mejorar las propuestas. Típicamente un proceso de investigación que emplea Investigación-Acción se halla compuesto de grupos de actividades organizadas formando un ciclo característico. Padak y Padak identifican los siguientes pasos, que deben seguirse en las investigaciones que utilicen este método [27]: Planificación, Acción, Observación y Reflexión.

Es fundamental para el buen éxito de esta investigación utilizar el método de investigación-acción, ya que mediante la realización de encuestas y entrevistas en empresas privadas y a profesores de la titulación, obtendremos una realimentación constante de información que nos permitirá determinar si los contenidos de las asignaturas y las competencias están alineadas con las necesidades de las empresas.

#### 4.2. GQM (Goal-Question-Metrics):

Para la realización del plan de métricas de esta investigación se utilizará el enfoque GQM (Goal-Question-Metric) [25], que proporciona una manera útil para definir mediciones tanto de procesos como de los resultados de un proyecto. Este enfoque considera que un programa de medición puede ser más satisfactorio si es diseñado teniendo en mente las metas (objetivo perseguido, que para nosotros serán las competencias y sub-competencias). Para determinar si hemos alcanzado el objetivo perseguido, se realizarán preguntas que ayudarán a medir si se está alcanzando en forma exitosa la meta definida. GQM define un objetivo, lo refina en preguntas y define métricas que intentan dar información para responder a estas preguntas. Puede ser utilizado por los miembros individuales de un equipo de proyecto para: i) Enfocar su trabajo; ii) Determinar su progreso hacia la realización de sus metas específicas.

GQM fue originariamente definido por Basili y Weiss [28] y extendido posteriormente por Rombach [29] como resultado de muchos años de experiencia práctica e investigación académica. Está basado en el principio básico de que “la medición debe ser realizada, siempre, orientada a un objetivo”. Ejemplos de objetivos pueden ser: mejorar la calidad, confiabilidad, reducción de costes, riesgos, etc.

#### 5. Objetivos perseguidos en la investigación

Una vez que se hemos analizado la importancia del momento actual en que se están implantando los nuevos grados, y analizado los métodos de investigación más afines, se ha planteado una serie de objetivos que permitan garantizar que dicha implantación será lo más acertada posible.

- El primero de los objetivos perseguidos en este proyecto es conseguir descomponer las competencias (que se describen actualmente de manera excesivamente compleja), en varias sub-competencias, que sean más fáciles de manejar. Con esa descomposición hecha, se tendría un mecanismo mucho más objetivo a la hora de definir relaciones entre dichas sub-competencias y las asignaturas. Para la obtención de este objetivo se plantea, en primer lugar, conseguir un conjunto objetivo y preciso de sub-competencias derivadas de las competencias definidas en cada intensificación, y posteriormente relacionar cada una de estas sub-competencias con los contenidos de las asignaturas y las actividades de las mismas.
- El segundo de los objetivos perseguidos consistirá en la definición de un plan de métricas que permitan medir el cumplimiento detallado de las competencias, permitiéndonos conocer en qué medida una asignatura contribuye al cumplimiento de una competencia (a través del cumplimiento de una o varias sub-competencias) y poder evaluar en qué medida los alumnos cumplen las competencias una vez cursadas las asignaturas. Para la obtención de este objetivo se plantea, en primer lugar, establecer un conjunto de métricas basadas en el estándar GQM, que permitan medir de forma objetiva el nivel de adecuación de las asignaturas que componen la intensificación con las competencias que se pretenden conseguir.
- El tercero de los objetivos perseguidos es poder comprobar si estas competencias

están alineadas con las necesidades de la empresa, mediante la realización de encuestas en empresas privadas y a profesores de la titulación. La empresa privada requiere tener ese conocimiento de cara a poder realizar una adecuada selección del perfil profesional, lo que redundará en elevar la satisfacción del empleado y los resultados obtenidos por la empresa.

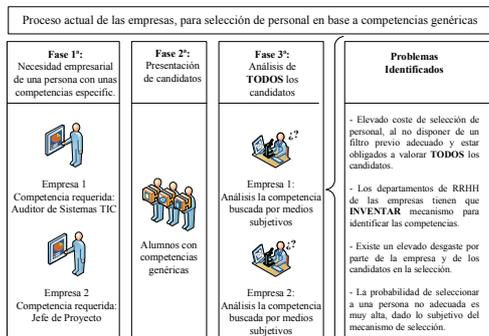


Figura 2. Proceso actual de las empresas para selección de personal en base a competencias genéricas.

Orientar la implantación de los planes de estudios a competencias que sean fácilmente medibles e identificables, y que estén alineadas con el sector privado, resuelve varios de los problemas que actualmente tienen las empresas a la hora de poder determinar el candidato más adecuado para una competencia (ver figura 2).

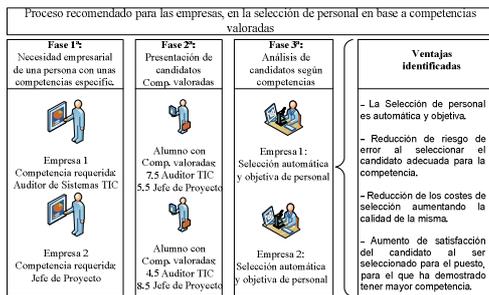


Figura 3. Proceso recomendado en el proyecto de innovación docente para selección de personal en base a competencias valoradas.

Los objetivos propuestos en este proyecto de innovación resuelven los problemas planteados en la Figura 2, y generan un nuevo paradigma de selección de personal en las empresas (ver Figura 3) mucho más eficiente que se traduce en mejoras productivas directas y en mejores salarios para los profesionales [24].

La unión de estos factores y el buscar el alineamiento de los conocimientos, tanto del sector público como el privado, tiene un valor añadido para este proyecto y garantizan la viabilidad del mismo.

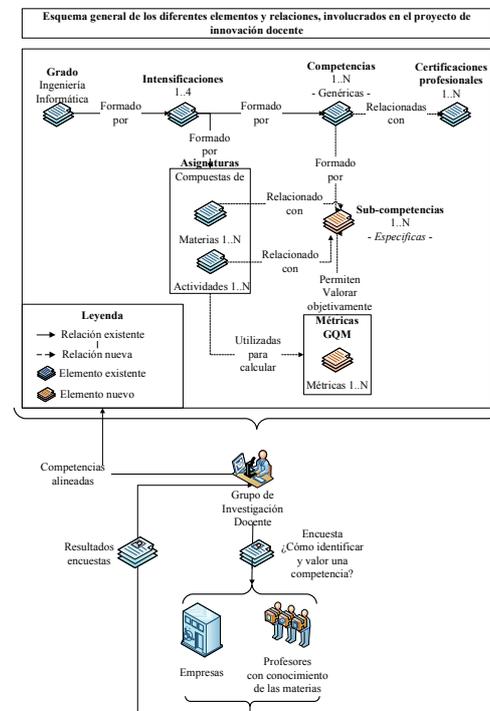


Figura 4. Esquema general de los diferentes elementos involucrados en el proyecto.

En la Figura 4 se puede ver el esquema general que se seguirá para conseguir los objetivos propuestos en el proyecto de innovación docente. Cada competencia se dividirá en un conjunto de sub-competencias, y para cada una de estas sub-competencias se definirán una serie de métricas derivadas del estándar GQM [25, 26] y de un conjunto de actividades asociadas a las asignaturas de la intensificación. A la vez, estas

competencias estarán alineadas con los requerimientos de la empresa privada, mediante la realización de encuestas del tipo “¿Cómo identifican una competencia a la hora de seleccionar un recurso para un tipo de proyecto específico?”, siendo esta información procesada y analizada por el equipo de investigación con el objetivo de mejorar las competencias definidas en la intensificación y determinar las actividades necesarias para la valoración del alumno en cada competencia.

De esta forma, el futuro graduado en Informática no sólo contará con un conjunto abstracto de competencias asociadas a una intensificación, sino que tendrá unas competencias unidas a un conjunto de métricas y actividades alineadas con las asignaturas de dicho grado y con los requerimientos del sector privado, lo que permitirá aumentar el grado de satisfacción y eficacia de los nuevos ingenieros y las empresas que los contraten, ayudando al entorno empresarial a ser más competitivo y a los nuevos ingenieros a comprender mejor las actividades asociadas con cada una de las competencias alcanzadas.

Por otro lado, aunque el proyecto inicialmente se centrará en obtener un conjunto de métricas que permita validar las competencias del grado de Ingeniería Informática, consideramos que una vez establecidas las bases y el conjunto de métricas, éstas podrán ser fácilmente extrapolables en otros grados de otras carreras.

## 6. Conclusiones

En este artículo hemos mostrado la importancia que tiene el momento actual de reestructuración de los planes de estudio para su adaptación al EEES, tanto para el futuro profesional de los alumnos como para el crecimiento estable a medio y largo plazo del tejido empresarial europeo.

Para ello, es fundamental realizar unos planes de estudio acordes a las necesidades reales del mercado, de forma que estos planes sirvan como catalizador de un aumento de la productividad y del tejido empresarial. Por lo tanto, estos planes de estudio tienen que ser acordes a la demanda del mercado.

Los primeros resultados obtenidos de la investigación y contrastados en empresas reales demuestra la importancia de la misma, y que

actualmente para las empresas no es suficiente con saber que una persona ha obtenido un título, sino que necesita tener un mapa de conocimiento global de las competencias obtenidas por esa persona y el grado de obtención de las mismas.

Dado el enorme interés que está despertando la investigación en las empresas privadas, consideramos que debemos profundizar mucho más en la obtención del mapa de competencias, con el objetivo de poder aportar a los alumnos no sólo un título académico, sino también un mapa con el grado de obtención de las diferentes competencias que complementen dicho título, y que haga mucho más fácil una adecuada valoración de sus conocimientos de forma que pueda obtener el trabajo más acorde a sus características y en el que pueda aportar la máxima productividad al tejido empresarial.

## Agradecimientos

Esta investigación es parte de los proyectos de innovación docente “Proceso de Reificación de las Competencias Generales y Específicas para el Grado de Ingeniería Informática y Definición de un Plan de Métricas de Evaluación de dichas Competencias”, y “Implantación y Orquestación de los Contenidos de Seguridad en el Grado en Ingeniería Informática que Favorezca en Acercamiento a las Principales Certificaciones Profesionales de Seguridad y Auditoría” concedidos dentro de la 6ª Convocatoria de Ayudas para Proyectos de Innovación Docentes promovidos por el Vicerrectorado de Ordenación Académica y Formación Permanente de la Universidad de Castilla-la Mancha.

## Referencias

1. Pereira, C., et al. *The European Computer Science Project: A Platform for Convergence of Learning and Teaching*. in *DLC&W 2006*. 2006. Lisbon, Portugal: October 2006.
2. Forbes, N.M., P. *Computer science today in the European Union*. *Computing in Science & Engineering*, 2002. 4(1): p. 10-14.
3. ACM, *Computer science curriculum 2008: An interim revision of CS 2001*, in *Review Task Force*, R.f.t. Interim, Editor. 2008, ACM.

4. CC2001, *Computing Curricula 2001. Computer Science*, I.C.S.a.A.f.C. Machinery, Editor. 2001.
5. SE2004, *Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering*, I.C.S.A.f.C. Machinery, Editor. 2004.
6. CE2004, *Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering*, I.C.S.A.f.C. Machinery, Editor. 2004.
7. Gorgone, J., et al., *MSIS 2006: Model Curriculum and Guidelines for Graduate Degree Programs in Information Systems*. Communications of AIS, 2006. **38**(2): p. 121-196.
8. Lunt, B., et al., *Curriculum Guidelines for Undergraduate Degree Programs in Information Technology*, in *Association for Computing Machinery (ACM)*, I.C. Society, Editor. 2008.
9. Pyster, A., et al., *Master's Degrees in Software Engineering: An Analysis of 28 University Programs*. IEEE Software, 2009: p. 95-101.
10. Lago, P., et al. *Towards a European Master Programme on Global Software Engineering*. in *20th Conference on Software Engineering Education & Training (CSEET'07)*. 2007.
11. Rico, D. and H. Sayani. *Use of Agile Methods in Software Engineering Education*. in *Agile Conference, 2009*. 2009. Chicago, USA.
12. Tripp, L., *SWEBOK: Guide to the Software Engineering Body of Knowledge*, I.C. Society, Editor. 2004: Los Alamitos, California.
13. Lavrischeva, E.M. *Classification of Software Engineering Disciplines*. in *Kibernetika i Sistemnyi Analiz*. 2008.
14. Lethbridge, T., et al. *Improving software practice through education: Challenges and future trends*. in *Future of Software Engineering(FOSE'07)*. 2007.
15. Thompson, J. *Software Engineering Practice and Education An International View*. in *SEESE'08*. 2008. Leipzig, Germany.
16. García García, M.J. and L. Fernández Sanz, *Opinión de los profesionales TIC acerca de la formación y las certificaciones personales*, in *Certificaciones profesionales en las TIC*. 2007, mayo-junio 2007: Novática. p. 32-39.
17. Seidman, S. *The Emergence of Software Engineering Professionalism*. in *IFIP International Federation for Information Processing*. 2008: Springer.
18. Suarez, B. and E. Tovar. *Accreditation in engineering*. in *Plenary Sessions of Int. Conf. Engineering Computer Education 2005 (ICECE05)*. 2006.
19. Povalej, R. and P. Weib, *Investigación de los sistemas de certificación TIC para profesionales en Europa*, in *Certificaciones profesionales en las TIC*. 2007: mayo-junio 2007. p. 24-31.
20. Seidman, S.B. *Software Engineering Certification Schemes*. in *Computer*, 2008. 2008.
21. Lutz, M.J.B., D., *Introduction: Software Engineering Curriculum Development*. Software, IEEE, 2006. **23**(6): p. 16-18.
22. Global\_Knowledge, *2010 IT Skills and Salary Report. A Comprehensive Survey from Global Knowledge and TechRepublic*, G.K.T. LLC, Editor. 2010.
23. Santiago, R. *Certificaciones personales, o como ser más competitivo*. 2010 [cited 2010 18/08/2010 00]; Available from: <http://www.rhhdigital.com/ampliada.php?sec=45&id=71019>.
24. Willmer, D. *Today's Most In-Demand Certifications*. 2010 [cited 2010 26 July 2010].
25. Basili, V.R., G. Caldiera, and H.D. Rombach, *The Goal Question Metric Approach*, in *Encyclopedia of Software Engineering*, G.C.a.D.H. Rombach, Editor. 1994, Jhon Wiley and Sons: New York. p. 528-532.
26. Basili, V.R., ed. *Applying the GQM paradigm in the experience factory*. Software quality assurance and measurement, ed. N. Fenton, Whitty, R., and Iizuka, Y. 1995, London, 1995: Thomson Computer Press. 23-37.

# Implantación y Orquestación de Contenidos y Competencias en Seguridad y Auditoría acorde a las Certificaciones Profesionales

David G. Rosado   Luis E. Sánchez   Daniel Mellado   Eduardo Fernández-Medina

Grupo Investigación GSyA. Escuela Superior de Informática. Dpto. Tecnologías y Sistemas de Información  
Universidad de Castilla-La Mancha  
Paseo de la Universidad,4  
13071 Ciudad Real

David.GRosado@uclm.es   Luise.Sanchez@uclm.es   dmellado@gmail.com   Eduardo.FdezMedina@uclm.es

## Resumen

Con la implantación del Grado en Ingeniería Informática, se ha elaborado una memoria donde se recogen, entre otras cosas, información sobre su organización en módulos, que a su vez contiene materias, y que éstas están formadas por asignaturas, que son definidas en términos de unos descriptores generales. Para estas asignaturas, se incluye también información sobre las competencias a las que da cuenta, las prácticas docentes, métodos de evaluación, etc., y en todo caso, queda para el momento de la implantación de las asignaturas, el trabajo de definir detalladamente los contenidos de las mismas. De entre todas las asignaturas y competencias definidas en el grado, hay varias dedicadas exclusivamente a seguridad y auditoría, y hay otras asignaturas que definen implícitamente aspectos de seguridad ya sea en las competencias a alcanzar o en los descriptores a desarrollar. De cualquiera de las maneras, hay que detallar el contenido de seguridad y auditoría de todas estas asignaturas que se ajusten a sus competencias y descriptores de forma coordinada, y que se acerquen lo máximo posible a las necesidades que demanda la sociedad a través de las principales certificaciones profesionales de seguridad y auditoría.

## Palabras clave

Seguridad, Auditoría, certificaciones profesionales, competencias, contenidos.

## 1. Motivación

La construcción de una Europa del conocimiento ha dado lugar a un movimiento importante, el cual

tiene como objetivo el desarrollo de un Espacio Europeo de Educación Superior (EEES) [1]. El Grado en Ingeniería Informática de la Universidad de Castilla-La Mancha surge como el nuevo reto de adaptación de los estudios de Informática al EEES, dando lugar a la definición, actualización y establecimiento de unos nuevos planes de estudios para adaptarlos hacia el sistema europeo de créditos (ECTS) [2].

Por otra parte, con el auge de las nuevas tecnologías de la información, servicios Web, comercio electrónico, etc., las organizaciones no se sienten seguras, sólo se generará confianza cuando podamos demostrar que el sistema global es seguro [3, 4]. Por ello, existe la necesidad de contar con nuevos profesionales en seguridad para mejorar la productividad de las organizaciones, asegurar su supervivencia, e incluso, cambiar nuestra forma de vida. Por tanto, queda justificada la gran importancia que tiene la implantación de la seguridad en nuestra sociedad moderna y conectada, implantación que se consigue desde la propia Universidad, insertando el contenido de seguridad en los nuevos planes de estudio.

La respuesta por parte de la Universidad de Castilla-La Mancha, a esta gran demanda y basándose en las recomendaciones de los principales currículos internacionales [5-11], es el nuevo plan de estudios para el grado en Ingeniería Informática donde se han establecido 4 intensificaciones o ramas de especialización en las cuales se han incorporado 4 asignaturas exclusivas de seguridad y auditoría en diferentes campos de la informática, como son las asignaturas obligatorias de “Seguridad de Sistemas Software”, “Seguridad en redes” y “Seguridad de los Sistemas Informáticos”, y la asignatura optativa de “Auditoría de Sistemas de Información”.

Además, más contenidos en seguridad y auditoría aparecen implícita o explícitamente en otras asignaturas obligatorias y optativas del grado, como por ejemplo, en el descriptor de seguridad y auditoría de redes inalámbricas dentro de la asignatura optativa de “Dispositivos y Redes Inalámbricas”, o el descriptor de seguridad en bases de datos dentro de la asignatura obligatoria “Desarrollo de Bases de Datos”, por nombrar algunos ejemplos. Además aparecen aspectos de seguridad de forma explícita e implícita dentro de las competencias generales y/o específicas de algunas asignaturas que debemos analizar y tener en cuenta.

Las empresas y los profesionales están demandando perfiles cada vez más especializados, por lo que es deseable que los futuros graduados cuenten con una o varias certificaciones profesionales internacionales en seguridad y auditoría informática [12-14], o al menos que tengan el camino preparado para conseguirlas. Por lo tanto es muy importante que los nuevos estudios estén muy enfocados a las necesidades profesionales [15, 16], sin perder el rigor científico exigible en una ingeniería, y para conseguir este objetivo es fundamental que estos nuevos planes de estudio tengan una orientación que facilite el acercamiento hacia las certificaciones profesionales [17-19].

Aunque no es posible, y tampoco aconsejable, que las principales certificaciones profesionales en seguridad y auditoría se obtengan directamente desde los estudios de grado universitario, creemos que es fundamental que estos estudios universitarios impulsen lo máximo posible estas certificaciones que tan directamente representan las necesidades reales de la industria de las TIC.

Aprovechando que nos encontramos definiendo y detallando los contenidos de las asignaturas del grado, es un buen momento para proponer y recomendar ciertos contenidos de seguridad y auditoría que encajen perfectamente con las competencias y objetivos de las asignaturas y con los temas de las asignaturas a cursar, que estén encaminadas a las necesidades reales de la industria estableciendo relación con los contenidos de las principales certificaciones profesionales en seguridad y auditoría, y que queden bien acopladas, ajustadas y coordinadas para el fin que se persigue, que es que el alumno adquiera gradualmente los

conocimientos básicos de seguridad y auditoría y tenga una clara progresión conforme vaya avanzando en el grado con un contenido que se ajusta a las necesidades reales que la sociedad demanda.

## **2. Competencias específicas de Seguridad y Auditoría en el Grado**

El contenido que se pretende incorporar debe estar enlazado y cubrir la mayor parte de los conceptos básicos requeridos por las certificaciones, de forma que contenidos incorporados en ciertas asignaturas nos sirvan de base para profundizar en otros aspectos que tienen relación con el contenido de seguridad y auditoría que se ha adquirido en asignaturas cursadas anteriormente. De esta forma, conforme el alumno progrese en el grado, también progresará en sus conocimientos sobre seguridad y auditoría, incluso podrá realizar prácticas o ejercicios que profundicen sobre aspectos aprendidos en asignaturas previamente cursadas. Se pretende que sea un aprendizaje continuo en temas de seguridad, de forma que nunca se repitan contenidos y que la suma de todos los contenidos defina un amplio conocimiento en aspectos de seguridad y auditoría que se acercan mucho al conocimiento exigido por las certificaciones profesionales de seguridad y auditoría.

Toda esta distribución e incorporación de contenidos en el grado debe realizarse de acuerdo a las competencias específicas de seguridad y auditoría que el grado en Ingeniería Informática define de forma explícita, y también analizar y tener en cuenta los aspectos de seguridad que se definen de forma implícita, ya que en muchas de las competencias, el aspecto de la seguridad está presente debido a su actual importancia. Las competencias que definen aspectos relacionados con la seguridad, calidad, auditoría, etc., son las siguientes:

- Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.
- Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta,

segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

- Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software.
- Capacidad de identificar, evaluar y gestionar los riesgos potenciales asociados que pudieran presentarse.
- Capacidad para comprender, aplicar y gestionar la garantía y seguridad de los sistemas informáticos
- Capacidad para determinar los requisitos de los sistemas de información y comunicación de una organización atendiendo a aspectos de seguridad y cumplimiento de la normativa y la legislación vigente.
- Capacidad para comprender y aplicar los principios de la evaluación de riesgos y aplicarlos correctamente en la elaboración y ejecución de planes de actuación.

Además, la coordinación entre las competencias se debe realizar entre materias, por un lado, se debe coordinar el contenido de seguridad y auditoría a incorporar dentro de las asignaturas que forman parte de la misma materia para llegar a alcanzar las competencias específicas definidas para esa materia, y por otro lado, se debe coordinar el contenido de seguridad y auditoría de las asignaturas de un mismo curso o cuatrimestre que forman parte de diferentes materias que definen distintas competencias de seguridad y auditoría, para que puedan acoplarse de forma coordinada y se enriquezca y amplíe el conocimiento a adquirir dentro de un mismo curso o cuatrimestre.

### 3. Objetivos que se persiguen

El objetivo de este proyecto es establecer una guía transversal para la implantación de contenidos relacionados con la seguridad informática en todas las asignaturas, materias y módulos del nuevo grado de Ingeniería Informática, que encajen perfectamente con las materias que se cursan, que

sirvan de elemento enriquecedor y que sirva a los alumnos para adquirir los conocimientos básicos de seguridad mínimos que a cualquier ingeniero informático se le exige desde el principio de su formación. Además, esta integración debe asegurar un camino que les lleve a adquirir la base del conocimiento y materias relacionadas y exigidas en las diferentes certificaciones profesionales de seguridad.

Además, estos contenidos deben estar bien definidos y orquestados con las materias que se cursan, con el curso donde se imparta y con el nivel y competencias exigidas a los alumnos. Por tanto, los contenidos deben repartirse por todo el grado de tal forma que no se repitan entre asignaturas afines, que vayan de niveles básicos a más avanzados conforme se vaya progresando en el grado, y que estén relacionados unos con otros dentro de la misma materia e incluso dentro del mismo curso.

Lo que se pretende es que una vez superado los tres primeros cursos del grado, los alumnos hayan adquirido los conocimientos básicos de seguridad necesarios dentro de todos los ámbitos y campos de la informática (sistemas, software, redes, bases de datos, web, programación, arquitecturas, tecnologías, etc.) antes de iniciar la especialización, donde, dependiendo de ésta, se profundice con más detalles sobre seguridad para un ámbito específico de la informática (con alguna asignatura exclusiva de seguridad), como pueden ser los sistemas software, redes, etc. Así, nos aseguramos que aunque sea experto en, por ejemplo, seguridad software, también haya adquirido los conocimientos básicos para el resto de ámbitos de la informática. Tanto las materias de la especialización como las del resto del grado con contenidos de seguridad deben encajar perfectamente con las materias exigidas por las certificaciones profesionales o al menos sentar las bases y principios para conseguirlas.

De esta forma, un futuro graduado en Informática tendrá las nociones y conocimientos básicos sobre Seguridad, con amplios conocimientos sobre un área en concreto y con las bases necesarias para optar a alguna de las acreditaciones profesionales de seguridad que existen y que son demandadas en la industria de las TIC.

#### 4. Actividades para alcanzar objetivos

Para conseguir los objetivos establecidos, vamos a dividir el plan de trabajo en 4 actividades bien diferenciadas y que son necesarias para la consecución de los objetivos. Las actividades son:

1. **Coordinación.** Se establece un plan de seguimiento y coordinación entre todos los involucrados en el proyecto, para establecer plazos de entrega y conseguir resultados que alimenten al resto de actividades, de forma ordenada, coherente y a tiempo.
2. **Análisis de las certificaciones profesionales:** Se debe realizar un estudio profundo y análisis de las diferentes certificaciones profesionales existentes en temas de Seguridad y del contenido establecido para cada una de ellas. Se deberá seleccionar los aspectos claves del contenido de las certificaciones profesionales e identificar los aspectos comunes a ellas.
3. **Análisis de las asignaturas del grado:** Se debe realizar un estudio y análisis de todas las asignaturas y materias del grado, así como conocer los diferentes descriptores y contenido para cada una de ellas con el fin de poder identificar temas y contenidos afines a las certificaciones profesionales para poder incorporar cierto contenido en la siguiente actividad.
4. **Establecer guía de implantación en el grado:** En esta actividad se hace el trabajo importante dónde detallamos los contenidos más apropiados a ser incorporados en las asignaturas del grado e indicamos en qué asignaturas deben ser incorporados, estableciendo una guía de implantación de los contenidos de seguridad acorde a las certificaciones profesionales dentro de las asignaturas y/o materias del grado, además de actualizar las competencias, tareas, contenidos, objetivos y planificación de cada asignatura modificada. También se define y detallan coherentemente los contenidos, prácticas, actividades docentes, ejercicios, etc. de las asignaturas involucradas. Además, se establece relaciones entre asignaturas con contenidos de seguridad y auditoría con el fin de evitar repetir contenidos y con el propósito de que el aprendizaje de los contenidos sea de

forma continua y progresiva conforme se vaya avanzando en el grado.

5. **Definir mapas de conocimiento orientados a certificaciones:** En esta actividad se hace un análisis y estudio de los contenidos de seguridad y auditoría incorporados, de las asignaturas involucradas y de las competencias que definen aspectos de seguridad dentro del plan de estudios. Con todo esto se hace un mapa indicando qué contenido es cubierto por cuales asignaturas y si cumplen con las competencias establecidas en el plan de estudios. Así, sabremos el mapa de asignaturas que cubren la mayor parte de los contenidos definidos por qué certificación, de forma que el alumno sepa las asignaturas que debe cursar para estar mejor preparado para una certificación u otra y qué aspectos no son cubiertos y debería reforzar.

#### 5. Metodología aplicada

Para abordar todo este planteamiento, vamos a utilizar una aproximación y adaptación del método “investigación-acción” que es un método de investigación cualitativo, que ofrecen ciclos iterativos de aplicación y refinamiento de las teorías construidas en entornos reales que permiten obtener interesante información para mejorar las propuestas. Padak y Padak identifican los siguientes pasos, que deben seguirse en las investigaciones que utilicen este método [20]:

- **Planificación:** Identificar las cuestiones relevantes, que guiarán el trabajo, que deben estar directamente relacionadas con el objeto que se está investigando y ser susceptibles de encontrarles respuesta. En esta actividad se estudian todas las posibilidades para encontrar los contenidos más relevantes en seguridad y auditoría de las principales certificaciones profesionales y se estudian las distintas alternativas de incorporar el contenido en las diferentes asignaturas del grado.
- **Acción:** Variación de la práctica, cuidadosa, deliberada y controlada. Se efectúa una simulación o prueba de la solución. Es cuando el participante de la iniciativa interviene sobre la realidad. Aquí es dónde tenemos que plasmar los contenidos en seguridad y

auditoría seleccionados en las diferentes asignaturas del grado, definir una guía de implantación y describir cuidadosamente las prácticas docentes, temas, formas de evaluación, competencias específicas, etc., de las asignaturas del grado que contendrán los contenidos de seguridad y auditoría implantados.

- **Observación:** Recoger información, tomar datos, documentar lo que ocurre. También se conoce como “evaluación”. En esta fase debemos ver y comprobar que la implantación se está llevando a cabo conforme a lo previsto, que los contenidos no se repiten, que se incorporan a las asignaturas correctas, que sigue una línea gradual de incorporación y relacionando contenidos entre asignaturas, etc.
- **Reflexión:** Compartir y analizar los resultados con el resto de interesados, de tal manera que se invite al planteamiento de nuevas cuestiones relevantes. Lo que se intenta es mejorar y actualizar la guía de implantación con nuevos contenidos, nuevos aspectos que no se habían considerado inicialmente y que han ido apareciendo conforme se implantan los contenidos de seguridad y auditoría en el grado.

Con estas características, el proceso adaptado y aproximado a “Investigación-Acción” que hemos utilizado es iterativo, de forma que se va avanzando en soluciones cada vez más refinadas mediante la compleción de ciclos, en cada uno de los cuales se ponen en marcha nuevos contenidos y asignaturas, que son puestas en práctica y comprobadas en el ciclo siguiente, tal como se muestra en Figura 1.

Este ciclo caracteriza Investigación-Acción como un proceso reflexivo de aprendizaje y búsqueda de soluciones. El carácter cíclico supone volver a reevaluar o replantear las acciones o caminos a seguir ponderando diagnóstico y reflexión.

Para el buen éxito de esta propuesta, la adaptación al método de investigación-acción es muy interesante, ya que mediante la realización de encuestas y entrevistas en las empresas de certificaciones y a profesores de la titulación, obtendremos una realimentación constante de información que nos permitirá determinar si los contenidos en seguridad y auditoría de las

asignaturas y las competencias, están alineadas con las necesidades reales de las empresas.

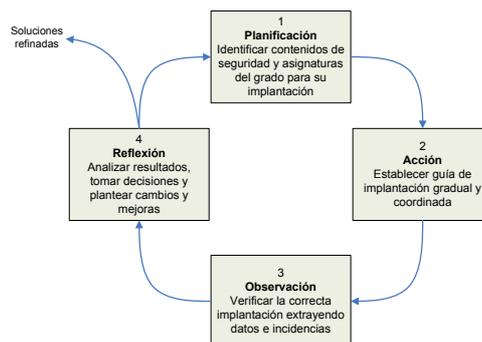


Figura 1. Carácter cíclico de Investigación-Acción.

## 6. Tareas a desarrollar

Dentro de las actividades en las que hemos dividido la propuesta, se han definido un conjunto de tareas que facilitan el entendimiento de lo que hay que hacer, la forma de planificar la propuesta y qué se debe realizar en cada una de las actividades.

- Actividad de Coordinación:
  1. Coordinar los trabajos de los distintos participantes y gestionar el proyecto (tiempo y recursos) para establecer comunicación y “feedback” entre actividades, tareas y participantes.
  2. Realizar la organización y seguimiento del proyecto, indicando los pasos a seguir, los plazos a cumplir y verificando que se vaya consiguiendo los objetivos planteados.
  3. Integrar los distintos trabajos realizados en cada actividad para que se vaya progresando, mejorando, actualizando y se pueda llevar a cabo una progresión correcta y fiable en cada actividad.
  4. Servir de interfaz entre los distintos participantes del equipo y resto de profesores involucrados en las asignaturas que requieran actualización, así como coordinar los resultados conseguidos.
- Actividad de Análisis de las certificaciones profesionales:

1. Estudiar las distintas certificaciones profesionales existentes en Seguridad y Auditoría y seleccionar aquellas más relevantes y demandadas por las empresas.
  2. Analizar el contenido exigido para la obtención de las certificaciones profesionales con el propósito de obtener una amplia lista de contenidos y seleccionar los más interesantes y convenientes, así como para identificar los aspectos comunes a todas ellas.
  3. Extraer de la lista de contenidos, aquellos más apropiados y que cubran la mayor parte de temas de seguridad y auditoría para ser incorporados a las asignaturas del grado.
- Actividad de Análisis de las asignaturas del grado:
    1. Estudiar y analizar el contenido de todas las asignaturas del grado con el fin de establecer relaciones entre ellas, prerrequisitos y post-requisitos de asignaturas afines.
    2. Seleccionar las asignaturas apropiadas a las que incorporar el contenido de seguridad y auditoría con el fin de obtener aquellas asignaturas candidatas a ser actualizadas con cierto contenido de seguridad y auditoría.
  - Actividad de Establecer guía de implantación en el grado:
    1. Relacionar el contenido a implantar extraído de la actividad 1 con las asignaturas identificadas en la actividad 2 y analizar qué contenido debe ser asignado a qué asignatura.
    2. Definir cómo y dónde se incorpora el contenido en la asignatura de tal forma que se defina los temas de seguridad y auditoría que se quieren incorporar y se añada en la guía docente correspondiente.
    3. Verificar que todos los cambios realizados sean correctos, que todo el contenido seleccionado haya sido incorporado, y que exista una coordinación entre contenidos dependiendo del curso donde nos encontremos.
  - Actividad de Definir mapas de conocimiento orientados a certificaciones:
    1. Establecer caminos formativos de conocimiento de seguridad y auditoría en el grado incluyendo las asignaturas

obligatorias, optativos y las de la especialización.

2. Relacionar los caminos formativos de conocimiento en seguridad y auditoría con las diferentes certificaciones profesionales definiendo el contenido de seguridad y auditoría de ese camino y comparándolo con el contenido total exigido por la propia certificación profesional.

## 7. Resultados

Los resultados esperados de esta propuesta, de los cuales ya se han conseguido una gran parte, son los siguientes:

1. Un informe de seguimiento que nos indique cómo se está desarrollando el proyecto, si los entregables están siendo definidos correctamente, si los participantes están desarrollando sus tareas convenientemente, etc.
2. La lista de contenidos de seguridad y auditoría, extraídos de las principales certificaciones profesionales, es un resultado importante porque a partir de aquí tenemos que definir la gran parte del contenido que será incorporado en el grado en Ingeniería Informática.
3. La lista de asignaturas del grado que son candidatas a ser actualizadas y/o modificadas para incorporarle nuevo contenido de seguridad tiene un valor importante porque de esta lista depende el éxito de la propuesta.
4. Una guía de implantación donde se indique qué contenido de seguridad y auditoría extraído de las certificaciones profesionales es incorporado a qué asignatura del grado, y cómo realizar dicha incorporación actualizando la guía docente de las asignaturas involucradas. Además, debemos indicar la composición de temas, relación de ejercicios apropiados, las prácticas a realizar, pruebas y exámenes, y definir posibles trabajos globales que se pueden ir realizando por partes y en distintas asignaturas y/o cursos que tienen cierta relación de contenidos.
5. Mapas de conocimientos orientados a las certificaciones profesionales de forma que el alumno pueda tener claro el conjunto de

asignaturas que debe cursar y el contenido de seguridad y auditoría definido en esos caminos que más se ajustan al perfil de alguna certificación profesional. Es importante este resultado porque con él podemos ver la lista de asignaturas que cubren cierto contenido de seguridad y auditoría de las certificaciones y qué competencias se cumplen, para así valorar el camino a seguir que más se acerca a adquirir el mayor conocimiento posible definido para alguna certificación concreta.

El procedimiento para comprobar el éxito de la propuesta es construir una tabla de asociación (que se está llevando a cabo actualmente) donde se relacionen, por un lado los contenidos de seguridad seleccionados y que son más apropiados y más generalizados por las principales certificaciones de seguridad; por otro lado las competencias que se deben adquirir definidas en el plan de estudios; y por último, las asignaturas del plan de estudio del grado en Ingeniería Informática. Así, se irá identificando qué contenido de seguridad es cubierto por cuáles asignaturas del plan de estudios y si cumple las competencias especificadas, indicando el porcentaje de cumplimiento.

De esta forma, tendremos la certeza de comprobar si el contenido más apropiado, que se ajustan a las necesidades reales de las empresas, ha sido debidamente incorporado e implementado en el grado, y si con dicha incorporación se cubre con alto porcentaje de competencias especificadas en el plan de estudios para las asignaturas. Además, con esta asociación, se puede extraer información de los puntos débiles en cuanto a contenidos y lo que el alumno tendría que reforzar para optar a alguna de las acreditaciones profesionales en seguridad y auditoría.

## 8. Conclusiones

Los contenidos de seguridad y auditoría dentro del grado en Ingeniería Informática deben estar perfectamente acoplados y organizados de forma que sea una progresión de conocimientos conforme se vaya avanzando en el grado, tengan una relación directa entre contenidos, estén ajustados a las competencias y objetivos de las

asignaturas y estén orientados a las necesidades más demandadas por la sociedad.

Las certificaciones profesionales internacionales son un excelente recurso para medir la demanda existente de profesionales en seguridad y auditoría que el mercado requiere. Estas certificaciones definen un contenido especializado en seguridad y auditoría que podemos utilizar para incorporarlos en el grado ajustándolos y adaptándolos a las competencias, descriptores y objetivos de cada asignatura del grado.

Por lo tanto, con esta propuesta pretendemos definir y detallar los contenidos, competencias, objetivos, prácticas docentes, etc. de cada asignatura donde se definan implícita o explícitamente temas de seguridad y auditoría descritos en el plan de estudios del grado, intentando que ese contenido se acerque lo máximo posible a los contenidos y competencias definidas en las principales certificaciones profesionales en seguridad y auditoría, de forma que haya una relación entre los contenidos de seguridad del grado y los contenidos de seguridad exigidos por las certificaciones profesionales que marcan las necesidades del mercado. Esto se debe hacer sin condicionar excesivamente la implantación del grado, pero de modo que se favorezca un acercamiento a estas certificaciones, tanto para que el alumno tenga una mejor formación, como para que opte a conseguir los certificados.

Además, pretendemos definir un mapa de conocimientos donde se describen el contenido de seguridad de cada asignatura, la relación con las competencias y objetivos de la asignatura y la relación con el contenido de las certificaciones profesionales. De esta forma, construimos un mapa indicando qué conjunto de asignaturas se acercan más al contenido de cierta certificación y qué asignaturas deben ser cursadas o qué contenidos deben ser ampliados por parte del alumno para acercarse más a las competencias exigidas para cada certificación. Así el alumno puede optar por una especialidad u otra, elegir unas optativas o de libre configuración u otras, dependiendo de los conocimientos que quiera adquirir referentes a seguridad y auditoría que más se acerquen a los requeridos por las certificaciones profesionales, a la vez que se le da una visión de los contenidos no cubiertos y que necesitan ser

reforzados y adquiridos para optar a las distintas certificaciones profesionales.

### Agradecimientos.

Esta investigación es parte de los proyectos de innovación docente titulados “Implantación y Orquestación de los Contenidos de Seguridad en el Grado en Ingeniería Informática que Favorezca en Acercamiento a las Principales Certificaciones Profesionales de Seguridad y Auditoría” y “Proceso de Reificación de las Competencias Generales y Específicas para el Grado de Ingeniería Informática y Definición de un Plan de Métricas de Evaluación de dichas Competencias”, concedidos dentro de la 6ª Convocatoria de Ayudas para Proyectos de Innovación Docentes promovidos por el Vicerrectorado de Ordenación Académica y Formación Permanente de la Universidad de Castilla-la Mancha.

### Referencias

- [1] EEES. Espacio Europeo de Educación Superior. Available from: <http://www.eees.es/>.
- [2] ECTS. European Credit Transfer System. Available from: <http://www.ects.es/>.
- [3] ANECA, Libro Blanco de Informática. 2005.
- [4] COPIITI - Conferencia de la Profesión de Ingeniero e Ingeniero Técnico en Informática, Perfil de la profesión de Ingeniero en informática y definición del currículo académico. 2003.
- [5] ACM/AIS, MSIS 2006: Model Curriculum and Guidelines for Graduate Degree Programs in Information Systems. 2006.
- [6] ACM/IEEE, Computer Engineering 2004. Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering. 2004.
- [7] ACM/IEEE, Software Engineering 2004. Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering. 2004.
- [8] ACM/IEEE, Computing Curricula 2005. The Overview Report. 2005.
- [9] ACM/IEEE, Computer Science Curriculum 2008. 2008.
- [10] ACM/IEEE, Information Technology 2008. Curriculum Guidelines for Undergraduate Degree Programs in Information Technology. 2008.
- [11] ISACA, ISACA Model Curriculum for Information Security Management. 2008.
- [12] Seidman, S., The Emergence of Software Engineering Professionalism, in IFIP International Federation for Information Processing. 2008, Springer.
- [13] Crowley, E., Information system security curricula development, in 4th conference on Information technology curriculum. 2003. p. 249-255.
- [14] Suarez, B. and E. Tovar, Accreditation in engineering, in Plenary Sessions of Int. Conf. Engineering Computer Education 2005 (ICECE05). 2006.
- [15] Seidman, S.B., Software Engineering Certification Schemes in Computer. 2008.
- [16] Batchman, T. and E. Tovar, Advantages and challenges which the accreditation process with ABET offers to engineering and computer science programs. Perspective of the engineering college, in Plenary Sessions of Int. Conf. Engineering Computer Education 2005 (ICECE05). 2005.
- [17] (ISC)2. The International Information Systems Security Certification Consortium, Inc., (ISC). Available from: <http://www.isc2.org/>.
- [18] GIAC. GIAC –Global Information Security Assurance Certification. Available from: [www.giac.org](http://www.giac.org).
- [19] ISACA. Information Systems Audit and Control Association. Available from: [www.isaca.org](http://www.isaca.org).

# Aplicación de herramientas de e-Evaluación en los nuevos enfoques evaluativos

Urcola Carrera, Leire

Departamento de Economía Aplicada V  
Facultad de Ciencias Económicas y  
Empresariales F. CC.EE. y EE.  
Universidad del País Vasco (UPV-EHU)  
[leire.urcola@ehu.es](mailto:leire.urcola@ehu.es)

Barrón Ruiz, Mariano

Departamento de Ingeniería de Sistemas  
y Automática  
Esc. Universitaria de Ingeniería Industrial  
Universidad del País Vasco (UPV-EHU)  
[mariano.barron@ehu.es](mailto:mariano.barron@ehu.es)

## Resumen

El objetivo de esta contribución es presentar dos experiencias relativas a métodos de evaluación orientados al aprendizaje implementadas en la Universidad del País Vasco (UPV-EHU). Esta universidad, junto a otras diez universidades de España, está participando en la actualidad en el desarrollo de un proyecto<sup>1</sup> enfocado hacia la implantación de estrategias de evaluación formativas, utilizando como soporte tecnológico la herramienta EvalCOMIX que ha sido integrada en la plataforma de software libre Moodle.

Los autores del presente artículo han puesto en práctica los conocimientos adquiridos en el proyecto a través de la implementación de distintos métodos de evaluación en las asignaturas "Informática de Gestión" y "Técnicas Digitales" en el transcurso del primer cuatrimestre del curso académico 2010-2011.

En las siguientes secciones se describen las principales características de las intervenciones educativas realizadas, en las que se especifica la metodología utilizada, el sistema de calificación fijado, así como los procedimientos y tareas de evaluación desarrolladas.

La última sección presenta los resultados globales y las principales conclusiones extraídas de ambas intervenciones, y concluye con un conjunto de propuestas de mejora y deja abiertas algunas preguntas para su discusión de cara a la implantación de nuevas metodologías evaluativas y su integración en entornos virtuales de aprendizaje.

---

<sup>1</sup> Proyecto Re-Evalúa: "Reingeniería de la e-Evaluación, Tecnologías y Desarrollo de Competencias en Profesores y Estudiantes Universitarios". Financiado por la Secretaría General de Universidades, Investigación y Tecnología. Referencia P08-SEJ-03502.

## Summary

The aim of this paper is to present two experiments based on methods of learning-oriented assessment carried out in the University of the Basque Country (UPV-EHU). This university, along with ten other universities in Spain, is currently involved in a project<sup>1</sup> focused on the implementation of formative assessment strategies, and it is supported by a technology called EvalCOMIX that has been integrated into an open source virtual learning environment, such as Moodle.

The authors of this paper have put into practice the knowledge acquired during the project through the implementation of different assessment methods in the courses of "Computer Science Management" and "Digital Techniques" during the first semester of the current academic year.

The following sections describe the main features of the educational interventions carried out, which specify the methodology used, the rating system established and the assessment procedures and tasks carried out.

The final section presents the overall results and key findings from both interventions, and concludes with a set of proposals for improvement and leaves opened some questions for discussion in order to the implementation of new assessment methodologies and their integration into virtual learning environments.

## Palabras clave

Evaluación, instrumentos de evaluación, evaluación formativa, evaluación entre iguales, aprendizaje cooperativo.

8:

## 1. Intervención educativa en la Escuela Universitaria de Ingeniería Industrial

La presente intervención educativa se ha llevado a cabo en una asignatura obligatoria de segundo curso de la especialidad electrónica industrial durante el primer cuatrimestre del presente curso académico. Los datos del grupo experimental se muestran en la tabla 1.

Universidad	Universidad del País Vasco (UPV-EHU)
Centro	Escuela Universitaria de Ingeniería Industrial de Eibar
Titulación	Ingeniería Técnica Industrial. Rama de Electrónica Industrial
Asignatura	Técnicas Digitales
Créditos	4,5 (3 Teóricos + 1,5 Prácticos)
Nº alumnos	27 matriculados

Tabla 1. Datos del grupo de alumnos

### 1.1. Metodología

El desarrollo de la asignatura contempla tareas presenciales (Exposición oral del profesor, Resolución de problemas en el aula, Seminarios y Prácticas de Laboratorio, Examen final) y tareas no presenciales. (Trabajo individual, Trabajo en grupo, Preparación de examen).

Desde hace varios años ya se venían utilizando las tareas no presenciales como recurso para aplicar la evaluación formativa en la citada asignatura. Las tareas no presenciales, se realizaban de forma individual o en grupos de 2 ó 3 personas, y se gestionaban a través de la plataforma Moodle. Estas tareas se programaban fijándose fechas de entrega. Asimismo, el profesor establecía otra fecha con el objeto de examinar los trabajos que habían entregado los alumnos para solicitar explicaciones adicionales oportunas, realizar valoraciones constructivas de sus trabajos y, finalmente, para mostrar la resolución de las tareas encomendadas.

Estas tareas no presenciales tenían una ponderación en la calificación final de la asignatura de un 20%, correspondiendo el 80% restante a la nota obtenida en el examen final. En el proceso de calificación intervenía sólo el profesor y las notas de cada tarea se publicaban en Moodle en un plazo corto (inferior a una semana) tras la fecha límite para la entrega de la tarea.

En la situación descrita, la implantación de la nueva metodología no parecía tarea complicada, tan sólo se requerían algunas modificaciones, entre las que se incluían las siguientes: diseñar instrumentos de evaluación, incorporar distintos métodos de evaluación (como la autoevaluación y la co-evaluación) y, consensuar los criterios de evaluación con los alumnos.

### 1.2. Sistema de calificación

La calificación final de la asignatura se calcula de la siguiente manera: las tareas no presenciales suponen un 6% de la nota final (se propone la realización de 4 tareas no presenciales), otro 6% se otorga por la asistencia y destreza observada en el Laboratorio y, el 70% restante se asigna al examen final. Hay que notar que en el supuesto de que los alumnos no entreguen una o más tareas, se añade su peso correspondiente al examen final.

### 1.3. Procedimiento de evaluación

El procedimiento de evaluación de la presente intervención educativa se realizó siguiendo la propuesta del Curso-Taller Re-Evalúa [5], que ofrece un modelo de procedimiento que permite concretar los elementos fundamentales que se deben contemplar para poder diseñar un procedimiento de evaluación contextualizado en asignaturas universitarias.

En este contexto, la tabla 2 recoge las competencias a desarrollar y los resultados de aprendizaje que se esperan alcanzar con la presente intervención.

En base a la especificación de estos elementos, se han diseñado las tareas de evaluación, explicitándose además, los criterios, la modalidad y los instrumentos de evaluación empleados.

Competencia general o básica:	
CG.1	Analizar y Diseñar circuitos digitales secuenciales.
Competencias específicas:	
CE.1	Explicar el funcionamiento de los circuitos integrados digitales MSI interpretando su símbolo lógico normalizado por el IEEE, y sus características eléctricas y temporales.
CE.2	Analizar máquinas de estado síncronas de tipo Mealy o Moore.
CE.3	Diseñar circuitos digitales utilizando dispositivos lógicos convencionales y dispositivos lógicos programables.
CE.4	Dominar un software de PC de captura de esquemas y de simulación de circuitos usándolo en el análisis y en la síntesis de los circuitos digitales.
CE.5	Manejar con destreza un lenguaje de descripción de hardware (HDL) diseñando circuitos digitales programables.
Resultados de aprendizaje	
RA.1	Obtener el diagrama de transición de estados, la tabla de transición de estados y el cronograma de las variables internas, de un circuito secuencial síncrono.
RA.2	Diseñar circuitos secuenciales síncronos con dispositivos lógicos de función fija.
RA.3	Diseñar circuitos secuenciales síncronos con dispositivos lógicos programables.
RA.4	Simular en un PC los circuitos diseñados para comprobar si su funcionamiento es conforme con las especificaciones.
RA.5	Montar un circuito secuencial compuesto por un PLD, y conseguir que funcione igual que el circuito simulado.

Tabla 2. Competencias a desarrollar y Resultados de aprendizaje

#### 1.4. Tareas de evaluación

Las tareas no presenciales sometidas a evaluación corresponden a trabajos destinados a verificar el nivel competencial del estudiante y están ligadas a los cuatro primeros resultados de aprendizaje reflejados en la tabla 2. Más concretamente, las tareas no presenciales consisten en ejercicios de análisis y diseño de circuitos digitales que se realizan con un programa de Simulación y un Compilador de Lógica Programable utilizando los ordenadores del centro o los ordenadores personales de los propios alumnos. Por su parte, el resultado de aprendizaje RA-5 señalado en la tabla 2, puede verificarse en el transcurso de las sesiones prácticas de Laboratorio.

Con relación a estas tareas, se solicita la entrega de un documento a través de la plataforma Moodle, que contenga la memoria del trabajo desarrollado, así como todos los ficheros que hayan sido generados a la hora de realizar la simulación o compilación de los diseños. Hay que señalar que para la evaluación de las tareas no presenciales se utiliza una Escala de Valoración con tres sub-dimensiones que ponderan el aspecto formal, la redacción, y el contenido de los trabajos de los alumnos con pesos del 10%, 10% y 60% respectivamente; además, existe un apartado de valoración global, con un peso de un 20%, reservado para estimar otras circunstancias no previstas que pudieran aparecer en el desarrollo de los trabajos. Cada una de las sub-dimensiones posee un conjunto de atributos (en total, 22) que pueden tener distinto peso dentro de cada sub-dimensión.

Por su parte, la opinión de los alumnos en la evaluación de cada tarea tendrá un peso de un 15%, quedando el resto del peso de la evaluación a criterio del profesor. Así, por ejemplo, en la tarea 1 que tiene carácter individual, la autoevaluación del alumno tendrá un peso del 15% y la del profesor un 85%. Mientras que en el resto de las tareas realizadas en grupos de dos alumnos, tanto la autoevaluación como la evaluación del compañero tendrán asignado el mismo peso (un 15%), y un 70% la evaluación del profesor.

## 2. Intervención educativa en la Facultad de CC.EE. y EE

La profesora que describe esta experiencia, ha desarrollado su intervención en una asignatura que imparte en cuarto curso de la Licenciatura de Administración y Dirección de Empresas, a lo largo del primer cuatrimestre del curso académico 2010/2011. Las características básicas del grupo experimental se exponen en la tabla 3.

Universidad	Universidad del País Vasco (UPV-EHU)
Centro	Facultad de Ciencias Económicas y Empresariales (FCCEE, Unidad Delegada en San Sebastián)
Titulación	Administración y Dirección de Empresas
Asignatura	Informática de Gestión
Créditos	9 (3 Teóricos + 6 Prácticos)
Nº alumnos	110 matriculados, 75 alumnos experimentales, 75 alumnos de control

Tabla 3. Datos del grupo de alumnos

### 2.1. Metodología

La presente asignatura se ha impartido combinando clases magistrales para la exposición de las distintas unidades temáticas que conforman la asignatura, la resolución de casos de estudio, así como la realización de prácticas en el laboratorio de ordenadores.

Como soporte de la asignatura presencial se utiliza la plataforma virtual de aprendizaje Moodle. Esta plataforma se emplea no sólo como repositorio de los materiales de estudio, sino también como medio de interacción entre el profesor y los alumnos, a través de la utilización de diversos recursos: foros, entrega de actividades, consultas, cuestionarios, etc. En este sentido, el entorno virtual de aprendizaje ha permitido implementar una metodología *Blended Learning* que combina el proceso de enseñanza-aprendizaje formal e informal, facilitando el

seguimiento y la gestión efectiva del mismo, así como llevar a cabo la tutorización y la evaluación de sus participantes. En relación con este aspecto, en la asignatura se implementan distintos tipos de evaluación; junto al sistema tradicional en el que sólo interviene el profesor, se ha incluido la evaluación entre pares, la co-evaluación y la autoevaluación de los estudiantes a lo largo de diversas fases, que tienen por objeto el diseño y la creación de una base de datos que pudiera formar parte de un sistema de información empresarial (de un establecimiento, negocio u organización, tanto real como virtual). Atendiendo al objetivo definido, las tablas 4 y 5 resumen las competencias específicas y los resultados de aprendizaje que se espera que alcancen los estudiantes:

Competencia general o básica:	
CG.1	Gestionar y organizar la información de forma eficiente como soporte a la adopción de decisiones
Competencias específicas:	
CE.1	Proporcionar los conocimientos necesarios sobre los sistemas de información empresariales y las correspondientes aplicaciones informáticas.
CE.2	Utilizar software específico para el tratamiento de grandes volúmenes de información apoyándose en una metodología formal de diseño.
CE.3	Estructurar y organizar la información susceptible de almacenamiento y de posteriores tratamientos en una estructura informacional del tipo de base de datos relacional.
CE.4	Interrogar bases de datos utilizando lenguajes estándar de consulta.

Tabla 4. Competencias a desarrollar

RA.1	Identificar un conjunto de informaciones interrelacionadas que puedan ser susceptibles de almacenamiento y de posteriores tratamientos en una estructura informacional del tipo base de datos relacional.
RA.2	Partiendo de la definición del dominio de aplicación y del análisis de requerimientos inicial, diseñar el esquema conceptual para la organización de los datos y su posterior almacenamiento y tratamiento.
RA.3	Partiendo de las consideraciones semánticas pertinentes, y aplicando el proceso de normalización de datos, diseñar el diagrama relacional de la base de datos.
RA.4	A partir del diseño, crear las estructuras de datos, especificando todas las características o propiedades que deben satisfacer sus atributos.
RA.5	Utilizando los lenguajes de interrogación de bases de datos relacionales (QBE y SQL), implementar consultas y otros objetos de la base de datos

Tabla 5. Resultados de aprendizaje

El trabajo propuesto de *Análisis, Diseño, Creación y Explotación de una Base de Datos* se realiza en grupos de tres o cuatro alumnos, que tienen que presentar, como si de una idea de negocio se tratara, una propuesta de “asunto” que después les permita materializarla a lo largo de distintas fases o etapas.

El número de participantes contemplado se considera idóneo para garantizar un equilibrio entre las dificultades que entraña el trabajo propuesto y la necesidad de controlar un reparto equitativo de las tareas entre los miembros del grupo, así como para poder realizar una tutorización más estrecha por parte del profesor y, desarrollar técnicas colaborativas por parte del estudiante, aspecto que, también se ha pretendido impulsar.

Como es sabido, en los procesos de enseñanza-aprendizaje el trabajo en equipo es una metodología pertinente para llegar a metas comunes, a través del diálogo, la comunicación y la participación de todos los miembros del grupo. Trabajar en equipo supone descubrir las fortalezas y debilidades de las personas que lo integran. Supone, además, analizar los mecanismos para mejorar continuamente la dinámica de todo el grupo. En este sentido, el trabajo en equipo es un proceso cooperativo, en el que cada miembro realiza las tareas para las que tiene mayor capacidad y habilidad [1], [3] y [4].

Toda la documentación relativa a las distintas etapas del trabajo, desde la definición del dominio y el análisis de requerimientos inicial, hasta la explotación final de la base de datos, se debe incluir en una Wiki, al considerarse que el uso de esta herramienta colaborativa facilitará la gestión, la coordinación y el desarrollo del proyecto de forma cooperativa. En definitiva, la Wiki se utiliza como repositorio web de toda la documentación asociada a la base de datos, disponiendo cada grupo de alumnos de su propia Wiki.

## 2.2. Sistema de calificación

Los alumnos que cursan esta asignatura pueden optar por uno de los siguientes tipos de evaluación: evaluación final (principalmente orientado a aquellos alumnos que no pueden asistir a clase con regularidad) y, evaluación continua (dirigido a alumnos que asisten a clase con regularidad). Pues bien, estos últimos alumnos forman parte del grupo experimental del proyecto que se ha puesto en marcha a lo largo del primer cuatrimestre. Por su parte, la evaluación de la asignatura se realiza mediante un examen final, que supone el 75% de la nota total y la realización del proyecto de diseño y creación de una base de datos del dominio de aplicación elegido por los estudiantes, que tiene una valoración del 25% de la nota total de la asignatura.

## 2.3. Procedimiento de evaluación

La tabla 6 resume las distintas fases que contempla el proyecto, especificándose las principales tareas que se deben realizar en cada una de ellas, así como los agentes responsables de la evaluación en cada caso (el profesor, entre compañeros, el propio alumno) y los instrumentos que han sido diseñados para ello.

Tal y como queda reflejado en dicha tabla, cada fase del proyecto tiene asignado un porcentaje del total del proyecto (el total del proyecto, es un 25%). Esta valoración está ponderada en base a las distintas sub-dimensiones (subtareas) y atributos (criterios) establecidos en cada uno de los instrumentos que han sido implementados para la evaluación.

Con el objeto de facilitar la gestión del procedimiento de evaluación, se configuran los distintos grupos de trabajo en la plataforma Moodle. Cada grupo nombra a una persona responsable que se encargará de la administración general de la Wiki del grupo, y además, deberá enviar en el plazo estimado las tareas exigidas en cada fase del proyecto. Así, se establece un único flujo de información y *feedback* entre la profesora y la persona responsable del grupo. Por su parte, para poder efectuar una evaluación intergrupala se realiza una asignación aleatoria de grupos, así cada grupo conocerá los grupos a los que tiene que evaluar y aquellos que lo van a evaluar. Se ha establecido un límite de tres o cuatro evaluaciones por grupo. Ahora bien, es necesario señalar que la evaluación intergrupala se realiza ciertamente a través de individuos, ya que se debe tener presente que para poder evaluar y ser evaluado, todo individuo debe enviar "su" trabajo a la plataforma.

En este sentido, los instrumentos de evaluación diseñados incluyen preguntas orientadas tanto a la evaluación del grupo en su conjunto, como a las aportaciones del individuo que se evalúa. Con este proceder, todos los alumnos toman parte en la evaluación intergrupala, obteniéndose efectos muy positivos (mayor responsabilidad en el cumplimiento de los compromisos adquiridos, desarrollo de capacidades críticas, se comparte el conocimiento, etc.).

La herramienta Wiki se considera muy adecuada para la realización y evaluación de trabajos en grupo, al quedar registradas todas y cada una de las aportaciones de los participantes, así como la frecuencia de las mismas. Por lo tanto, esta herramienta permite calificar de manera más objetiva al grupo en su conjunto, y a cada uno de sus integrantes en particular, en base al empleo de ciertos indicadores, como pueden ser la cantidad, la calidad y la frecuencia de las aportaciones realizadas, entre otras.

Productos de aprendizaje	Participan en la evaluación	Instrumentos de evaluación
Fase I. Análisis de requerimientos y diseño de la BD (10%)	Profesor, 100%	Escala de valoración
Fase II. Edición en un espacio web (Wiki) (5%)	Co-evaluación (profesor 50% y entre iguales 40%) y Autoevaluación, 10%	Escala de valoración
Fase III. Creación y Explotación de la base de datos (10%)	Co-evaluación (profesor 60% y entre iguales 30%) y Autoevaluación, 10%	Lista de control + Escala de valoración
Examen final (75%)	Profesor, 100%	

Tabla 6. Resumen de las fases de evaluación

### 3. Resultados globales

La valoración de los resultados globales de las intervenciones llevadas a cabo se realiza desde dos perspectivas distintas. En primer lugar, desde el aspecto pedagógico y, en segundo lugar, desde la valoración técnica del propio software utilizado para realizar las tareas de evaluación.

Considerando el aspecto pedagógico y formativo del proyecto Re-Evalúa, las dos intervenciones que han sido descritas coinciden en calificar de exitosa, en términos generales, la implementación de los objetivos establecidos en el marco del proyecto. La formación y los materiales facilitados a través del curso-taller *Re-Evalúa* han permitido sistematizar todo el procedimiento de evaluación con un enfoque orientado a la evaluación por competencias, que es uno de los retos a los que se enfrenta la Universidad ante el Espacio Europeo de Educación Superior.

Sin duda, la participación de los alumnos en el proceso de evaluación en sus distintas modalidades les permitirá desarrollar ciertas competencias que muy probablemente deberán aplicar en su vida profesional.

En este sentido, las metodologías empleadas en ambas intervenciones, han favorecido el desarrollo de competencias tan importantes en los estudiantes como son: la reflexión, la exploración, la comunicación, el espíritu crítico y el trabajo en equipo, favoreciendo en el estudiante un aprendizaje responsable y autónomo y permitiéndole, además, aprender nuevos conceptos, conocer nuevas tecnologías y experimentar con herramientas novedosas para desarrollar otras competencias específicas relacionadas con la propia materia de estudio.

Con todo, la respuesta de los estudiantes se ha correspondido de una forma mayoritaria con las expectativas creadas. Los estudiantes se han sentido parte integrante de un ambicioso proyecto, con lo que la motivación y la responsabilidad para alcanzar sus objetivos han sido mayores, así como la calidad de los resultados obtenidos. Este hecho corrobora los positivos efectos que la experimentación y el uso de herramientas innovadoras parecen tener en las actitudes y motivación de los estudiantes [6].

Sin embargo, desde el aspecto puramente técnico con relación al software empleado para el diseño de los instrumentos de evaluación (EvalCOMIX), hay que notar que se han detectado deficiencias técnicas importantes, que han impedido llevar a cabo todo el procedimiento de evaluación de una forma totalmente automatizada. Este hecho, sin duda, ha restado eficacia al proceso.

Ciertamente, el desarrollo de este software forma parte de una experiencia piloto que tiene por objeto diseñar instrumentos de evaluación y utilizarlos en el proceso de evaluación de forma integrada en una plataforma virtual de aprendizaje [2], pero a la vista de los errores detectados, hoy por hoy, no puede considerarse todavía un producto maduro para su empleo en el ámbito educativo. Ahora bien, una vez solventadas y depuradas las mejoras necesarias, consideramos que el software empleado puede ofrecer gran potencialidad como herramienta de evaluación desde un entorno virtual de aprendizaje.

En este sentido, a continuación, se enumeran algunas propuestas de mejora que deberían tenerse en consideración:

- Es necesario depurar las opciones de personalización que se incluyen en cada uno de los instrumentos de evaluación (lista de control, escalas de valoración, rúbricas, instrumentos mixtos, etc.), de forma que se puedan configurar pesos distintos a los que establece la propia herramienta para poder evaluar correctamente las dimensiones y sub-dimensiones con los criterios que se establezcan.
- Asimismo, se requiere una mayor flexibilidad a la hora de configurar los propios instrumentos de evaluación, por ejemplo posibilitando cambios en los desplazamientos de las dimensiones, sub-dimensiones, y atributos de los mismos.
- Se recomienda ofrecer formatos más simples a la hora de guardar/imprimir los instrumentos (formularios) de evaluación, ya que se ha detectado que cuando se guarda o imprime un instrumento en formato PDF, se cambian los formatos de configuración, originando que el instrumento ocupe más tamaño del que debiera y ralentizando su visualización.
- Sería conveniente ofrecer la posibilidad de importar instrumentos de evaluación para agilizar la creación de instrumentos mixtos.
- Por último, se echa en falta la posibilidad de realizar una evaluación inter-grupal, no sólo entre agentes individuales, sino de todo el conjunto, de forma que puedan asignarse calificaciones grupales además de individuales (aunque consideramos que esta limitación sería más achacable a la propia plataforma de aprendizaje que al software empleado para el diseño de los instrumentos de evaluación).

Para concluir, podemos afirmar que en el tiempo que llevamos participando en el proyecto, son varias las preguntas que nos han surgido en relación a la utilidad de integrar la herramienta EvalCOMIX en un entorno virtual de aprendizaje para facilitar la puesta en práctica de las nuevas tendencias en la evaluación.

Recogemos, a continuación, algunas de estas preguntas de forma sintética, como elementos de debate y reflexión:

- ¿Qué ventajas ofrece EvalCOMIX frente a otras herramientas de software libre existentes en la Red?
- ¿Podrá integrarse EvalCOMIX en otros entornos virtuales de aprendizaje además de Moodle?
- Teniendo en cuenta la tendencia hacia la utilización de entornos de aprendizaje personalizados o PLEs (*Personal Learning Environments*) ¿tiene sentido desarrollar este tipo de herramientas para ser integrados en entornos cerrados de aprendizaje?

#### 4. Agradecimientos

Agradecemos al Grupo de Excelencia de la Universidad de Cádiz la oportunidad que ha brindado a la Universidad del País Vasco (UPV/EHU) para participar en el proyecto Re-Evalúa, permitiendo que se incorpore a un grupo puntero, a nivel nacional, en el desarrollo de estrategias de evaluación de competencias en el ámbito universitario.

#### Referencias

- [1] Hadfield, J. *Classroom dynamics*. Oxford University Press, 1992.
- [2] Ibarra, M.S. *EvalCOMIX: Evaluación de competencias en un contexto de aprendizaje mixto*. Universidad de Cádiz, 2008.
- [3] Johnson, D.W. *El aprendizaje cooperativo en el aula*. Paidós, 1999.
- [4] Putnam, J.W. *Cooperative Learning and Strategies for Inclusion. Celebrating Diversity in the Classroom*. Paul H. Brookes, 1993.
- [5] Rodríguez, G. e Ibarra, M.S. *Re-Evalúa. La sistematización de la evaluación. Unidad temática III*. Curso-Taller., 2010.
- [6] Rubia, B.; Ruiz, I.; Anguita, R.; Jorrín, I. y Rodríguez, H. "Experiencias colaborativas apoyadas en e-learning para el EEES: Un estudio de seis casos en la universidad de Valladolid". *Relatec*, Nº 8, 2009.

# Diseño de las actividades formativas y evaluación de la asignatura Fundamentos del Software

J. A. Gómez Hernández    A. León Salas    P. Paderewski Rodríguez

Departamento de Lenguajes y Sistemas Informáticos

Universidad de Granada

C/ Periodista Daniel Saucedo Aranda s/n

18071 Granada

[jagomez@ugr.es](mailto:jagomez@ugr.es)

[aleon@ugr.es](mailto:aleon@ugr.es)

[patricia@ugr.es](mailto:patricia@ugr.es)

## Resumen

Con la puesta en marcha del Grado en Informática en la Universidad de Granada durante el curso 2010-11 nos hemos enfrentado a la necesidad de diseñar la asignatura Fundamentos del Software de primer curso. El principal objetivo que nos planteamos fue establecer un equilibrio entre las actividades formativas, adecuadas para que los alumnos alcancen las competencias establecidas por las guías, y un sistema de evaluación integrado en la metodología que realimente el proceso de aprendizaje, además de calificar. La presente ponencia muestra como se han materializado estas propuestas genéricas en la asignatura con el objetivo de mejorar el aprendizaje de nuestros alumnos de manera eficiente.

## Summary

With the implementation of the Degree in Computer Science at the University of Granada during the academic year 2010-11 we have faced the need to design the course Foundations of Software in the first year. The main goal was to establish a balance between training activities suitable for students to achieve the tasks set by the guidelines, and an evaluation system built into the methodology that feedback the learning process, and also assessment. This paper shows how generic these proposals have materialized in the subject with the aim of improving our students' learning effectively.

## Palabras clave

Enseñanza, aprendizaje, metodología, actividades, evaluación.

## 1. Motivación

El binomio enseñanza-aprendizaje es un proceso complejo y en numerosas ocasiones la experiencia nos ha puesto de manifiesto la inexistencia de una correlación entre ambos procesos, de manera que hemos realizado procesos de enseñanza que no han tenido necesariamente como consecuencia un proceso de aprendizaje equiparable.

En ocasiones, al planificar nuestra enseñanza creemos que la puesta en práctica de la misma trae necesariamente como consecuencia que los alumnos a los que va dirigida, llevarán a cabo su

aprendizaje, y eso no tiene por qué ser una realidad. Si bien nuestra enseñanza es la misma para todos los alumnos, el aprendizaje que produce es diferente para cada uno de ellos [8].

De lo expuesto se deduce que es necesario trabajar en un diseño del proceso de enseñanza que sea capaz de producir como resultado el mayor aprendizaje posible en cada alumno, teniendo en cuenta las variables que inciden en el aprendizaje, y adaptar, en la medida de lo posible, el proceso de enseñanza al perfil de cada uno, con el objetivo de obtener la máxima eficacia posible.

Uno de los aspectos que nos replanteamos en el binomio enseñanza-aprendizaje fue no centrar tanto la acción en los resultados de los aprendizajes como en el proceso y los medios con los que se alcanzan, de cara a desarrollar en el estudiante la capacidad para aprender ahora y a lo largo de su vida.

En este sentido, es primordial mejorar y optimizar los procesos de aprendizaje de los alumnos a través de la adecuada elección de las actividades de aprendizaje (entendiendo por actividad de aprendizaje cualquier evento dentro de un programa, cuya realización exija la participación del alumno y facilite la consecución de un objetivo previamente definido). Teniendo presente que la variación en el tipo de actividades facilita el aprendizaje.

Para que las actividades de aprendizaje sean consideradas como tales, deben formar parte de un programa de formación, ser realizadas por el alumno (para que sean eficaces), y su realización debe facilitar la consecución de alguno o varios objetivos de aprendizaje.

Respecto al enfoque tradicional, debemos replantearnos también la naturaleza de las actividades de aprendizaje, contemplando no solo las actividades clásicas de conocimiento, sino también los contenidos procedimentales y actitudinales, que deben ser objeto de aprendizaje y evaluación, como establece el Título. Nuestro diseño no se orienta a que el alumno adquiera multitud de conocimientos sino a que desarrolle la capacidad de hacer un uso competencial de los elementos fundamentales que necesita saber para resolver problemas reales.

De forma general, indicar que cada actividad tiene una estructura formal que consta de: título de

la actividad, objetivos de aprendizaje (ya sean generales, u operativos), los recursos necesarios, una descripción (que expresa de forma clara y operativa en qué consiste la actividad prestando especial atención a lo que el estudiante debe “producir” como resultado de la realización de la actividad), orientaciones, sugerencias y pistas, presentación de resultados, los criterios de evaluación, y una valoración por el estudiante.

En este contexto, cobra especial importancia la evaluación de los distintos aprendizajes. Ya no se trata solo de acreditar lo aprendido (evaluación sumativa), sino también y más importante, ayudar a la mejora del aprendizaje (evaluación formativa). Para ello, será necesario incorporar a los mecanismos de evaluación tradicionales nuevas técnicas de evaluación centradas en valorar y mejorar la actividad del alumno y no solo en juzgar los conocimientos adquiridos.

En consecuencia, nos enfrentamos a un replanteamiento de binomio enseñanza-aprendizaje en el cual tenemos en cuenta los siguiente elementos:

1. Una concepción de la docencia basada en una visión integral del proceso enseñanza-aprendizaje donde se programa por actividades de carácter comprensivo e integrador.
2. Un enfoque metodológico basado en la adquisición de conocimientos por la integración de la realidad donde se aplican.
3. Un sistema de evaluación basado en el análisis de la ejecución de actividades próximas a la realidad.

## 2. La asignatura y su contexto

La asignatura que nos ocupa se denomina Fundamentos del Software (FS) y se imparte en el primer cuatrimestre de primer curso de los nuevos estudios de Grado en Ingeniería Informática en la Universidad de Granada. En la memoria del título, esta asignatura tiene encomendadas las siguientes competencias: “Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería” dentro del módulo de formación básica que establecen las recomendaciones del Consejo de Universidades.

En este contexto, la guía docente de la asignatura [11] establece los objetivos de la misma: conocer el entorno de ejecución de programas, el proceso de construcción de los mismos, y su interacción con los sistemas persistentes de datos. Estos objetivos se materializan en el programa que podemos encontrar en [10]. El hecho de ser una asignatura de primer cuatrimestre impone dos fuertes condicionantes: uno, debe impartirse en paralelo con la asignatura Fundamentos de Programación, y dos, es previa a la asignatura Tecnología y

Organización de Computadores, que desarrollan competencias relacionadas con nuestra asignatura.

La memoria del título establece una serie directrices que afectan a la metodología, la evaluación y programación de las asignaturas. El Apartado 3, muestra como se han materializado estas propuestas genéricas en la asignatura.

## 3. Metodología

La metodología seguida en FS tiene su base en la motivación realizada en el Apartado 1, y se apoya en la propuesta realizada en [9], donde se persigue una integración de los tres elementos claves: el modelo de aprendizaje, las actividades planificadas, y la forma de evaluación propuesta, como se muestra en la Figura 1. Con este esquema, se establecen los pasos a seguir:

- 1º Establecer competencias y subcompetencias.
- 2º Diseñar las tareas de acuerdo con el modelo de aprendizaje, la modalidad de enseñanza y los medios y recursos disponibles.
- 3º Determinar los instrumentos de evaluación de esas tareas.

Siguiendo estos pasos, se establecen para la asignatura las competencias y unidades de competencia que aparecen en la Tabla 1. De ella se observa como hay competencias que abarcan varios temas. Esto se debe principalmente a que todos los componentes de un sistema de computo forman parte de un único sistema mostrando dependencias entre sí más o menos fuertes.

### 3.1 Métodos de enseñanza

Las actividades formativas junto con su contenido en ECTS, la metodología de enseñanza y aprendizaje, y su relación con las competencias que deben adquirir los estudiantes se establecen en el título de Grado y en la respectiva Guía docente.

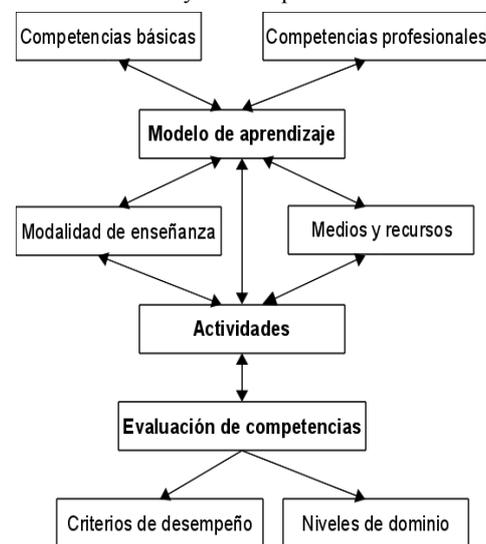


Figura 1.- Esquema de planificación docente [8]

Vamos a comentar brevemente estos métodos y a explicar el uso particular que de ellos hemos realizado.

El elemento principal para la adquisición de conocimientos ha sido el *trabajo no presenciales individual* (estudio y trabajo autónomo) que contempla actividades (guiadas y no guiadas) propuestas por el profesor a través de las cuales se adquiere, mediante el estudio del material bibliográfico y la realización de ejercicios sencillos, el conocimiento básico que anteriormente se adquiría a través de las lecciones magistrales. Desde el punto de vista organizativo, esto tiene la ventaja de que es fácil estimar el tiempo que el alumno dedica fuera del aula a la asignatura. Desde el punto de vista metodológico, se deja al alumno la responsabilidad de realizar el trabajo que puede acometer perfectamente de forma individual y en el que no necesita interacción con los demás.

Además, las actividades no presenciales permiten el estudio individualizado o grupal de las actividades realizadas en las clases presenciales.

Su propósito es favorecer en el estudiante la capacidad para autorregular su aprendizaje, planificándolo, diseñándolo, evaluándolo y adecuándolo a sus especiales condiciones e intereses.

La *lección magistral* (clases teóricas-expositivas) en nuestro caso supone un aspecto diferenciador respecto a la metodología tradicional. Al no ser el elemento básico de transmisión de conocimientos, la clase presencial se utiliza principalmente como punto de encuentro entre profesor y alumnos donde se resuelven dudas, se indaga, se dialoga e intercambias ideas a través de actividades propuestas a tal fin. La filosofía subyacente se basa en dedicar el tiempo presencial, y por tanto colectivo, a realizar actividades que conllevan interacción personal. Su propósito es principalmente el desarrollo del alumno de las competencias cognitivas, procedimentales y actitudinales, de la materia motivando al alumno a la reflexión para que descubra las relaciones entre conceptos y forme una mentalidad crítica y analítica.

Indicar que en algunos casos se siguen utilizando para ilustrar conceptos complejos con enfoques alternativos/complementarios a los de los libros de texto, o dar una visión general de los temas, haciendo uso de metodología expositiva y medios audiovisuales. También se utilizan para las pruebas de evaluación de las capacidades adquiridas.

Las *actividades prácticas* (clases prácticas de laboratorio) pretenden mostrar al alumnado cómo debe actuar para aplicar los conocimientos adquiridos. Se utilizan también para evaluar las capacidades adquiridas. Su propósito es el desarrollo en el alumnado de las habilidades instrumentales de la materia.

Las *actividades no presenciales grupales* (estudio y trabajo en grupo) propuestas por el profesor permiten de forma grupal se profundiza en aspectos concretos de la materia posibilitando a los estudiantes avanzar en la adquisición de determinados conocimientos y procedimientos de la materia. Su propósito es favorecer en los estudiantes la generación e intercambio de ideas, la identificación y análisis de diferentes puntos de vista sobre una temática, la generalización o transferencia de conocimiento y la valoración crítica del mismo.

Las *tutorías académicas grupales* complementan el proceso de enseñanza al existir una interacción directa entre el estudiante y el profesor. Se integran en la metodología haciendo que su uso se obligó un par de veces en el cuatrimestre. Su propósito es orientar el trabajo autónomo y grupal del alumnado, profundizar en distintos aspectos de la materia y orientar la formación académica integral del estudiante.

### 3.2 Metodología de enseñanza y aprendizaje

Las actividades formativas propuestas se desarrollarán desde una metodología participativa y aplicada que se centra en el trabajo del estudiante (presencial y no presencial/individual y grupal). Las clases teóricas, las clases prácticas, las tutorías, el estudio y trabajo autónomo y el grupal son las maneras de organizar los procesos de enseñanza y aprendizaje de los diferentes módulos.

Experiencias propias previas [5,6] permiten evidenciar que la utilización de un variado abanico de métodos de enseñanza favorece el aprendizaje de los alumnos. Además de los recogidos en el título, podemos indicar que hacemos un uso intensivo de los siguientes métodos [4]: aprendizaje colaborativo y aprendizaje basado en problemas (ABP). En lugar de seguir una metodología ABP "pura", utilizamos un la versión propia que pretende reducir el trabajo autónomo del alumno dado que son de primer curso. Para ello, el profesor es quién divide el problema complejo en subproblemas más simples y fáciles de resolver de forma aislada.

### 3.3. Técnicas e instrumentos de evaluación

Los criterios de evaluación nos indican la forma en que debemos evaluar el aprendizaje de cada una de las competencias asignadas a la asignatura. Como ejemplo, la Tabla 2 muestra los criterios de evaluación para la competencia 2 "Conocimientos básicos sobre sistemas operativos".

Fijados los criterios de evaluación, antes de mostrar las técnicas que se utilizarán para evaluar el aprendizaje de las competencias establecidas, indicar que hemos apostado por un sistema de

9:

Tabla 1.- Asociación de conocimientos a competencias.

Competencia	Unidad de competencia	Conocimientos asociados
Conocimientos básicos sobre el uso y programación de los ordenadores	1.- Conocimiento de los componentes de un sistema de computo.	Tema 1: Sistema de cómputo Tema 2: Introducción a los SOs.
	2.- Conocimiento de la estructura y funcionamiento de un computador	Tema 1: Sistema de cómputo
Conocimientos básicos sobre sistemas operativos	3.- Conocimiento de la estructura y funcionamiento de un SO.	Tema 1: Sistema de cómputo Tema 2: Introducción a los SOs
	4.- Uso y programación de un ordenador mediante un interprete de órdenes	Tema 2: Introducción a los SOs Tema 3: Compilación y enlazado de programas
...		

evaluación continuada [12] para favorecer el aprendizaje ya que permite que el alumno tenga una realimentación constante sobre su propio proceso de aprendizaje.

Entre las técnicas utilizadas encontramos: pruebas objetivas, cuestionarios teóricos y prácticos, resúmenes, resolución de problemas, cuaderno de laboratorio, portafolio, y rúbrica.

A continuación describimos los instrumentos de evaluación y la forma de utilizarlos:

- Para cada tema se propondrán al alumno un conjunto de actividades/problemas para su resolución. Estos problemas se resuelven de forma cooperativa. Una vez resueltos en el tiempo estipulado, un miembro del grupo elegido por el profesor expone su solución en clase de forma oral con ayuda de los medios técnicos necesarios (pizarra, proyector, ordenador, etc.). La exposición es evaluada por el profesor (heteroevaluación) y por el resto de los grupos (co-evaluación) mediante una escala de valoración. Así mismo, cada grupo realizará una autoevaluación de su trabajo mediante otra escala de valoración.
- El cuaderno de laboratorio refleja el trabajo realizado por el alumno en las clases de laboratorio en la resolución de los supuestos prácticos propuestos en la guía de prácticas, así como el trabajo previo a las mismas. Este cuaderno es único, contendrá un apartado para cada una de las competencias desarrolladas, y se integrará en el portafolio.
- Cada alumno confecciona un portafolio con los resúmenes, soluciones de los problemas, cuaderno de laboratorio, cuestionarios teóricos y prácticos, que se entregará al finalizar cada tema y será evaluado por el profesor mediante rúbricas.

La corrección de todas estas evidencias se realiza al finalizar cada una de las actividades lo que proporciona al alumno una realimentación inmediata sobre su proceso de aprendizaje. Es

decir, se han planteado principalmente como una evaluación formativa.

Analizando el coste desde el punto de vista del profesor de realización de los citados instrumentos puede parecer elevado, pero en realidad no lo es tanto ya el profesor solo corrige las pruebas objetivas (cuyo coste de corrección es el equivalente al de un examen final con un par de preguntas por tema), un trabajo en grupo por tema, y el portafolio. Otras evidencias, como los cuestionarios teóricos, prácticas, etc. son corregidos entre compañeros.

Por último, hemos definido el sistema de calificación, establecido en parte en el epígrafe "Sistema de evaluación" del Título de Grado, que pondera como contribuye cada instrumento a la calificación final del alumno. En este sentido, comentar que los cuestionarios teóricos y los prácticos, los resúmenes se integran en el portafolio. Éste no es calificable por temas sino globalmente, al objeto de simplificar la corrección. Su realización es obligatoria dado su inminente carácter formativo, siendo su objetivo ayudar al alumno a realizar un seguimiento de su progreso en la asignatura. Por ello, se realiza una revisión intermedia a mitad de cuatrimestre, y cada vez que el alumno asiste a tutorías. Su calificación se realizará mediante una rúbrica.

Para finalizar indicar que la evaluación sumativa es la suma ponderada de una prueba individual por tema (teoría y prácticas), un problema de grupo por tema, y el portafolio.

### 3.4 Evaluación de la labor docente

Es importante destacar que como se establece en [2] es necesario evaluar el proceso docente para establecer cuales son los puntos fuertes y los aspectos que hay que mejorar de nuestra propia labor de cara a mejoras a aplicar en los cursos siguientes. Para ello, se ha diseñado y realizado una encuesta destinada a recoger las impresiones del alumno sobre la metodología y evaluación

Tabla 2.- Criterios de evaluación para la competencia 2.

<b>Competencia 2: Conocimientos básicos sobre sistemas operativos</b>	
Criterios para la evaluación de la <b>realización de actividades</b>	Razonar sobre el funcionamiento de sistema operativo en diferentes circunstancias.
Criterios para la evaluación de los <b>resultados de actividades</b>	Utilizar los servicios de un sistema operativo a través de órdenes y utilidades del sistema.
Criterios para la evaluación de <b>conocimientos adquiridos</b>	Describir las funciones de los componentes básicos de un sistema operativo multiprogramado y sus relaciones.
Criterios para la evaluación de <b>actitudes y valores</b>	Expresarse con claridad, exactitud y coherencia. Asistir y participar en las actividades propuestas en los plazos previstos.

empleada, cuyos resultados se comentan en el Apartado 5.

#### 4. Diseño y evaluación de actividades

Vamos a concretar la aplicación de la metodología docente propuesta en la sección anterior utilizando una de las actividades que se han desarrollado durante el curso: el diseño e implementación de un sistema operativo multiprogramado “en papel”. Esta actividad se ha planteado como un proyecto de envergadura (ABP) que se ha troceado en sub-actividades que involucran a los temas 1 y 2.

La competencia que se pretende que adquieran los estudiantes en esta unidad didáctica es el diseño y construcción de los elementos estructurales de un sistema operativo multiprogramado. Pensamos que, de cara al aprendizaje del alumno, es necesario acompañar los conceptos asociados a la multiprogramación de una serie de problemas prácticos. Cada ejercicio práctico permitirá fijar uno o varios conceptos y elaborar un elemento necesario para la construcción del SO multiprogramado. Al final los diferentes elementos se combinan obteniendo el proyecto completo.

Para entender la estructura y funcionamiento de un SO multiprogramado el alumno debe conocer y saber aplicar una serie de elementos previos:

- a) Cómo funciona un procesador abstracto.
- b) Cómo se manejan las interrupciones.
- c) Cómo implementar un proceso.
- d) Cómo un proceso realiza una operación de entrada/salida.
- e) Cómo puedo conmutar entre procesos.
- f) La necesidad de introducir el concepto estado de un proceso.
- g) Necesidad de un planificador de procesos.

Todos estos pasos se han planteado como actividades separadas que se unen finalmente para que el alumno comprenda la estructura y funcionamiento de un sistema operativo multiprogramado.

#### 4.1 Arquitectura “de papel”

Para la elaboración de las distintas relaciones concepto(s)-ejercicio práctico es muy útil trabajar con una arquitectura de ordenador abstracta (“en papel”) sobre la que se podrá programar con un lenguaje máquina simplificado. Trabajar con una arquitectura abstracta nos permite fijar los conceptos teóricos básicos obviando gran parte de los elementos de una arquitectura real concreta, evitando de esta forma que el alumno se pierda con multitud de detalles técnicos de arquitecturas reales y que verán en asignaturas posteriores.

El primer paso es pues caracterizar esta arquitectura “de papel” que se hace en el Tema 1. La arquitectura presenta los siguientes elementos:

- Memoria de palabras con un tamaño fijo para cada palabra y direccionamiento por palabra. Elegimos esta opción porque nos interesa que el estudiante entienda el concepto de direccionamiento de una palabra de memoria y obtención de la información alojada en la palabra direccionada.
- CPU con registros de propósito general (R0 a R7), un contador de programa (PC), un registro de instrucción (IC), un registro de pila para modo usuario (SP1), registro de pila para modo kernel (SP2), un registro palabra de estado del procesador (PSW) que contiene bits para el resultado operación igual o menor que 0 o el modo de operación del procesador.
- Dispositivo de entrada/salida con interrupciones con un puerto (registro) para control de operación (lectura/escritura) y uno de datos.
- Descripción de un lenguaje máquina simplificado para manejar los tipos de instrucciones básicas y su significado. Las instrucciones utilizadas son: de transferencia de información, operaciones aritméticas, de ruptura de flujo de ejecución, llamada a subrutina y, por último, una instrucción que permite realizar una llamada al sistema, SYSCALL, para que el alumno entienda la necesidad de que el SO controle la E/S sobre dispositivos.

Una vez que disponemos de las descripciones tanto de la arquitectura abstracta como del lenguaje ensamblador, pasamos a definir cada concepto y su correspondiente ejercicio práctico. Como se comentó previamente, el objetivo es que el estudiante fije los conceptos mediante actividades formativas.

De esta forma, el alumno en el trabajo no presencial individual adquiere los conocimientos relativos al funcionamiento de un computador. En clase, se resuelven las dudas que hayan surgido.

Normalmente el estudiante entiende los conceptos teóricos pero no sabe como aplicarlos. Para ello, el profesor planifica una serie de actividades a resolver de forma cooperativa sobre aquellos temas que la experiencia dicta son difíciles de entender. Estas actividades son para este punto: construir pequeños programas (sumar, escribir en un puerto de dispositivo, hacer un manejador de interrupciones, etc.) que permiten aplicar los contenidos estudiados a situaciones prácticas, y por tanto desarrollar las competencias establecidas. Todas estas actividades son corregidas y discutidas en clase inmediatamente a su realización. De esta forma el alumno obtiene una realimentación inmediata sobre su trabajo. Sólo uno de estos ejercicios es corregidos por el profesor, de forma que como máximo se corrige un ejercicio en grupo por tema. El resto de ejercicios se pueden corregir entre compañeros.

#### 4.2 Concepto de multiprogramación.

Una vez que se ha asimilado como funciona un computador elemental, y realizadas las lecturas de introducción a los sistemas operativos. Se plantean al alumno diferentes actividades destinadas a asimilar los conceptos y poner en prácticas competencias transversales como la capacidad para resolver problemas, capacidad de análisis y síntesis, de trabajo en equipo, etc.

Por ejemplo, se propone un ejercicio (Listado 1) con el objetivo de que el alumno vea la necesidad de que las operaciones de entrada/salida estén proporcionadas por el SO y no disponibles directamente para el programa de usuario

Listado 1.- Error lógico en E/S.

```
1000 LOAD R0, 10 # Programa
1001 LOAD R1, 20
1002 ADD R3,R1,R2
1003 IN R4,3
1004 ADD R0,R3,R4
```

Se pide al alumno que describa el comportamiento del programa ejecutándolo ficticiamente sobre la arquitectura “en papel”. El alumno debe entonces darse cuenta que el sistema trabaja con E/S guiada por interrupciones por lo que la instrucción “IN R4,3” que requiere una entrada de datos del puerto 3 permite que el programa continúe con la siguiente instrucción “ADD R0,R3,R4” provocando un error lógico. De esta forma, reflexiona sobre el hecho de que el poder realizar

instrucciones en espera del envío de la solicitud de interrupción por parte del dispositivo no es siempre bueno y que es necesario que este tipo de instrucciones solamente puedan ser ejecutadas por el SO y solicitadas a éste mediante el uso de llamadas al sistema.

Otro aspecto a considerar en la adquisición de conceptos y habilidades que nos ha mostrado la experiencia, es que es difícil que los alumnos entiendan determinadas soluciones a problemas, si no a ellos mismos no se les ha planteado el problema. Por ello, ciertas actividades están más orientadas a que se planteen el problema para que entiendan el por qué de la solución.

Por ejemplo, la actividad mostrada en e Listado 2 se encamina a ver la necesidad de que exista un código de sistema operativo que realice la función de selección del siguiente programa a ejecutar en CPU y de salvar registros del procesador (mantienen estado de ejecución) del programa que deja de ejecutarse.

Listado 2.- Planificador de procesos.

```
100 LOAD R7,1 # Gestor de llamadas, PSW=1
101 CMP RSC,R7 # operación de lectura?
102 JZ 106 # Saltar a sys_read
103 LOAD R7,2
104 CMP RSC,R7 # operación de escritura?
105 JZ 109 # Saltar a sys_write
106 IN RD,RP # Entrada de dato del puerto
RP al registro RD
107 # codigo para permitir ejecutar otro
programa
108 IRET
109 OUT RD,RP # Salida de dato RD al
puerto 3.
110 # codigo para permitir ejecutar otro
programa
111 IRET
```

```
1000 LOAD R0, 10 # Programa
1001 LOAD R1, 20
1002 ADD R3,R1,R2
1003 LOAD RSC,1 # Llamada al sistema para
lectura
1004 LOAD RP,3 # Puerto de E/S
1005 SYSCALL # Salto a rutina de llamadas
1006 ADD R0,R3,RP
```

En ella, se pide al grupo que describa el comportamiento del programa y de la rutina de gestión de llamadas al sistema sobre la arquitectura “en papel”. Esta permite que el alumno reflexione sobre el siguiente hecho: a partir de este momento, podemos hacer trabajo útil si conseguimos que el SO ceda el control a otro programa cargado en memoria principal. Seguidamente, se le plantean los siguientes interrogantes: ¿Cómo identificamos el programa al que ceder el control? ¿Cómo podemos salvaguardar el contexto de ejecución (registros del procesador) del programa que se está ejecutando en este momento?

Como vemos, estos ejercicios permiten proponer al alumno suficientes interrogantes (problemas) como para dar paso al siguiente contenido teórico: implementación de la abstracción de proceso a través del descriptor de

Tabla 3.-Resumen de algunos resultados de la encuesta realizada por los alumnos.

Aspectos cuestionados	Valoración media obtenida en el curso 2010/2011
Metodología basada en el trabajo autónomo y cooperativo	Adecuación, contenidos, etc. : 3 puntos (sobre 4) Calificación general: 6.2 puntos (sobre 10)
	Le parece más interesante para su formación: 52,83% Exige más trabajo para el estudiante : 60,38% Exige más trabajo para el profesor: 45,28% Es más adecuada para el nivel universitario: 37,74% Los hace más responsable de su formación: 62,26%
Evaluación continua	2,88 puntos (sobre 4)
Interés por la asignatura/estudios	2,99 puntos (sobre 4)
Mejora en capacidades generales	2,8 puntos (sobre 4)

proceso (Bloque de control de proceso - PCB) y la utilidad de un modelo de estados de procesos. La estructura de este contenido teórico sería similar a la que acabamos de exponer.

Como podemos observar, la metodología docente se basa en la presentación de un contenido teórico y a continuación la presentación de ejercicios que ayuden a contextualizar la teoría en nuestra máquina abstracta y, por tanto, que el alumno no solo sepa los contenidos si también como aplicarlos con el rigor y análisis necesario como para poder implementarlos realmente. De esta forma, se fomenta la aparición de dudas del tipo “¿pero cómo se llevaría a cabo ...?”, lo que permite ir encadenando el aprendizaje de conceptos y la adquisición de las correspondientes competencias.

Una vez trabajado un tema, se realiza una evaluación individual del mismo en una hora de clase. En ella, se plantea al alumno que resuelva un supuesto práctico similar a los trabajados en el tema. Como se pretende que se alcancen determinadas competencias, se ha optado, no por hacer preguntas teóricas, si no cuestiones aplicadas para ver como aplica el alumno las habilidades adquiridas a un problema práctico.

## 5. Resultados

Para evaluar la metodología y el sistema de evaluación seguidos en la asignatura en el primer año de su impartición, se pasó a los estudiantes una encuesta en la que se preguntó su opinión, sobre todo, por cuatro aspectos que consideramos importantes:

1. La metodología utilizada.
2. La evaluación continua realizada.
3. La visión que tienen de la carrera que comienzan y que se introduce con esta asignatura que contiene conceptos propios de informática.
4. La mejora durante el curso de una serie de capacidades claves en los estudios universitarios.

Respecto a la metodología seguida se le han preguntado cuestiones como la adecuación de la metodología, los contenidos seleccionados, las lecturas propuestas, la utilidad de las clases presenciales para aclarar las dudas. Las respuestas se valoran positivamente ya que en los cuatro primeros, con una puntuación máxima de 4, se llega a una media de 3 puntos, y se da una valoración general (que se puntúa sobre 10) de 6,92 puntos. Creemos que vamos bien encaminados ya que estos datos son positivos en general, pero más aún teniendo en cuenta que es la primera vez que se imparte esta asignatura, es de primer cuatrimestre y son estudiantes que vienen directamente de la enseñanza secundaria. Estos factores añaden dificultades adicionales ya que la asignatura, sus contenidos y la forma de impartirla se ha ido construyendo poco a poco durante el cuatrimestre. Nos hemos encontrado, por ejemplo, con problemas del tipo: no saben cómo funciona un computador y, antes de hablar del software, hemos tenido que darle unos principios básicos del funcionamiento de los computadores.

Además, se recogieron también cuestiones relacionadas con la visión que tienen los estudiantes sobre si la metodología seguida es más interesante para su formación (52,83%), si les requiere más trabajo (60,38%), si le exige más trabajo al profesor (45,28%), si es más adecuada para el nivel universitario (37,74%) y si la consideran importante para su formación (62,26%). Una vez finalizada la evaluación continua sumativa que hemos llevado a cabo, los datos también nos indican que ésta ha sido bien aceptada por los estudiantes, sobre todo teniendo en cuenta que esta asignatura es de las primeras en las que se les evalúa en la Universidad.

Se les han preguntado cuestiones relacionadas con su interés por la asignatura y por los estudios de informática en general. Creemos que tanto la metodología como la forma de evaluar y de hacerles trabajar de forma continuada influye positivamente en su visión de los estudios que acaban de comenzar. Entre estas cuestiones

podemos destacar preguntas sobre si han llegado a dominar los principios básicos de la asignatura (50,9 % han contestado 3 y 15,1% han contestado 4), si han aumentado su vocabulario técnico (67,9 % han contestado 3 y 22,6% han contestado 4) y si han aumentado su interés por los estudios de informática (52,8 % han contestado 3 y 24,5% han contestado 4). La puntuación media está cercana a los 3 puntos sobre 4 puntos (2,99).

Finalmente, es importante también poder comprobar si los estudiantes tienen la impresión de haber mejorado su capacidad para interpretar la información, para resolver problemas, para extraer conclusiones y se ha mejorado su actitud crítica. Todos estos ítems también se han valorado sobre 4 puntos, y en todos se han superado los 2.5 puntos, lo cual indica que los estudiantes han sacado provecho de la metodología propuesta para mejorar sus capacidades generales que son importantes y que les serán útiles durante toda la carrera (y en su trabajo profesional). Su puntuación media es de 2.8 puntos.

En la Tabla 3 podemos ver resumidos estos cuatro aspectos y las puntuaciones medias obtenidas en los grupos de preguntas relacionados con cada uno. Esta comparativa nos demuestra que los estudiantes valoran bien tanto la metodología como el tipo de evaluación seguidos y que también ven un cambio positivo en otros aspectos como el interés por la asignatura y la mejora de ciertas capacidades.

### Conclusiones

De los resultados obtenidos, podemos concluir que los alumnos ven positivamente la metodología con evaluación formativa. En especial valoramos positivamente cómo los alumnos indican su aumento por el interés por la asignatura y los estudios de informática y como ven mejorar sus capacidades generales.

El aspecto más importante a mejorar es la motivación de los alumnos a la hora de realizar las lecturas dada la importancia de éstas en la metodología. La encuesta ha revelado que casi un 40% contestan un 2 sobre 4 a la pregunta sobre si “realizan las lecturas en el momento indicado” dada la importancia de éstas en la metodología.

Para ello estamos trabajando en complementar las lecturas con un cuestionario que incida sobre los puntos importantes de la lectura y sirva de guía para el debate que se realiza en clase presencial. Además, se pueden formalizar todas las

actividades en una guía de trabajo autónomo que, además de recoger los objetivos de las mismas, de una visión general del objetivo final de todas y cada una de las lecturas y actividades.

### Referencias

- [1] Agudo Martínez, M<sup>a</sup> Josefa (2009/10), Guía de evaluación de la asignatura Análisis Gráfico Arquitectónico, disponible en <http://proyectoformcom.org>
- [2] Álvarez Rojo, V., García Jiménez, E., Gil Flores, J. y Romero Rodríguez, S. (Coords.) (s.a.), Guía para la planificación y desarrollo de la docencia en el área de enseñanzas técnicas, Universidad de Sevilla.
- [3] Benjumea Mondéjar, Jaime (s.a.), Guía de evaluación de la asignatura Ingeniería de Protocolos, disponible en <http://proyectoformcom.org/>
- [4] De Miguel Díaz, Mario (2005). Modalidades de enseñanza centradas en el desarrollo de competencias. Orientaciones para promover el cambio metodológico en el Espacio Europeo de Educación Superior. Ediciones Universidad de Oviedo.
- [5] J. A. Gómez Hernández. *Guía didáctica y de trabajo autónomo de Sistemas Operativos I*, Editorial Copicentro, 2007.
- [6] J. A. Gómez Hernández, *Guía didáctica y de trabajo autónomo de Sistemas Operativos II*, Editorial Copicentro, 2009.
- [7] González Romano, José M. (2008-9), Guía de evaluación para Introducción a la Programación 1, Universidad de Sevilla, disponible en <http://proyectoformcom.org/>
- [8] Lozano Rodríguez, A. (2005), *Estilos de aprendizaje y enseñanza*, Editorial MAD.
- [9] Salmerón Pérez, Honorio (2010), La evaluación integrada en un modelo de aprendizaje para la educación superior, Curso de evaluación de competencias en la universidad, Vicerrectorado para la Garantía de la Calidad, 14 junio-15 julio, Granada.
- [10] Universidad de Granada (2010), Título: Grado en Ingeniería Informática.
- [11] Universidad de Granada (2010). Guía docente de la asignatura Fundamentos del Software.
- [12] Valero-García, M. y Díaz de Serio, Luis M., Autoevaluación y co-evaluación: estrategias para facilitar la evaluación continuada, *Actas del I Congreso Español De Informática (CEDI)*, Granada, 2005.

# Evaluación de entornos de programación para el aprendizaje

Sonia Pamplona Roche  
Departamento de Ciencias e Ingeniería  
  
Universidad a Distancia de Madrid  
Carretera de la Coruña  
(Vía de Servicio, nº 15), KM.38,500  
28400 Collado Villalba (Madrid)  
sonia.pamplona@udima.es

Nelson Medinilla Martínez  
Departamento de Lenguajes y Sistemas  
Informáticos e Ingeniería del Software  
Universidad Politécnica de Madrid  
Facultad de Informática  
Campus de Montegancedo  
28660 Boadilla del Monte (Madrid)  
nelson@fi.upm.es

## Resumen

El presente trabajo tiene como objetivo la evaluación en el nivel universitario de cuatro entornos de programación que han sido diseñados para edades más tempranas. Los entornos seleccionados son *LOGO*, *Scratch*, *Alice* y *Greenfoot* y todos ellos comparten el objetivo de motivar al estudiante y facilitar el aprendizaje.

El resultado ha sido una caracterización de cada uno de los entornos. El objetivo es que pueda servir como guía al profesor o al estudiante para seleccionar un entorno o para la creación un nuevo entorno a partir de las ventajas y debilidades encontradas.

## Summary

The objective of the present work is the evaluation at the university level of four programming environments designed for younger ages. The selected environments are *LOGO*, *Scratch*, *Alice* and *Greenfoot*. All of them share the goal of motivating and facilitating student learning.

The result has been a characterization of each environment in order to serve as a guide to the teacher or the student, as well as for creating a new environment based on the strengths and weaknesses found.

## Palabras clave

*Greenfoot*, *Alice*, *LOGO*, *Scratch*, entornos de programación, aprendizaje.

## 1. Introducción

La programación de computadores puede ser una forma de aprendizaje y no un fin en sí mismo en el nivel escolar, según Seymour Papert y Alan Kay. [15, 6].

El presente trabajo constituye un primer paso en la aplicación de esta idea a la enseñanza universitaria, en el campo de la informática. Se han evaluado cuatro entornos de programación que comparten el objetivo de motivar al estudiante y facilitar el aprendizaje. Los entornos evaluados son *LOGO*, *Scratch*, *Greenfoot* y *Alice*. El propósito principal de *LOGO* y *Scratch* es potenciar nuevas maneras de aprender y de pensar, aplicables a varias materias. *Greenfoot* y *Alice* fueron concebidos con el fin más específico de facilitar el aprendizaje de la programación de computadores.

Nuestra evaluación realiza dos aportaciones principales. En primer lugar, se lleva a cabo desde el punto de vista del aprendizaje y no de la enseñanza, poniendo como evaluadores a estudiantes del Grado de Ingeniería Informática, en lugar de docentes como es lo habitual. En segundo lugar, completa la información existente sobre cada entorno, que era insuficiente para poder tomar decisiones con respecto a su uso. Para llevar a cabo la evaluación de los entornos, se diseñó un cuestionario que los estudiantes cumplimentaron después de una sesión de trabajo con cada uno ellos en la que se valoró su funcionalidad, usabilidad e interés. Los datos obtenidos fueron analizados cuantitativa y cualitativamente.

Como resultado se obtuvo una caracterización de cada uno de los entornos. Esto constituye un primer paso en la aplicación de las ideas de Paper y Kay en dos direcciones. La primera, servir como guía al profesor o al estudiante para seleccionar un entorno adecuado para un determinado tipo de aprendizaje. La segunda, constituir un punto de partida para la creación un nuevo entorno que aúne ventajas y mejore debilidades encontradas.

## 2. Entornos evaluados

A continuación se realiza una breve presentación de los entornos usados en el trabajo.

### paper

El entorno de programación *paper*<sup>1</sup> [16] es una implementación actual del lenguaje *LOGO* (Figura 1). La primera versión de *LOGO* fue creada en 1967 por el equipo de Seymour Papert en el Instituto Tecnológico de Massachusetts (MIT) con la finalidad de que éste fuera una herramienta para el aprendizaje. El símbolo del lenguaje *LOGO*, la tortuga, originalmente fue una criatura robótica situada en el suelo que se movía tecleando comandos en un ordenador. Más tarde, la tortuga migró a las pantallas de los ordenadores y fue usada para dibujar formas y diseños. Hasta el día de hoy se conocen más de doscientas versiones de *LOGO* [9], y muchas de ellos están disponibles para los sistemas operativos actuales. Hemos seleccionado el entorno *paper* por su simplicidad de uso y por su simplicidad de acceso, ya que es una aplicación web.

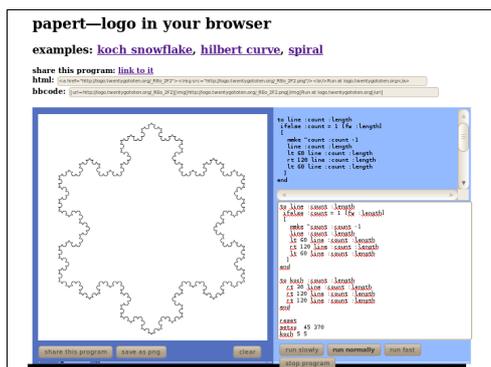


Figura 1. Entorno de programación *paper*.

<sup>1</sup> El nombre del entorno evaluado es *paper*, todo con minúsculas. Se distingue así del nombre del creador del lenguaje *LOGO*: Seymour Papert.

### Alice

*Alice* [2] fue originalmente desarrollado en 1995 como una herramienta de prototipado rápido para animaciones de realidad virtual [17] (Figura 2). A finales de los noventa el equipo de *Alice* se propuso transformar el entorno en una herramienta educativa para enseñar programación de computadores. Su objetivo fue crear una nueva versión para aprender conceptos sobre programación orientada a objetos usando un entorno gráfico 3D que contenía objetos concretos cuyo comportamiento podía modificar el usuario [4].

Una de las características del entorno *Alice* es que dispone de un editor “arrastrar y soltar” para prevenir los errores de sintaxis que suelen frustrar a los programadores novatos.

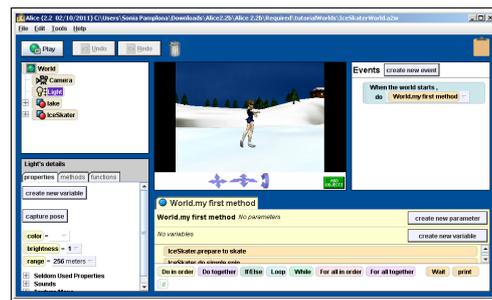


Figura 2. Entorno de programación *Alice*.

Existen diferentes versiones de *Alice* en este momento:

- *Alice 2.2*. Es la última versión estable del entorno.
- *Alice 3 beta*. Es una versión beta de la nueva versión del entorno cuya principal característica es que incluye dos interfaces, una en el lenguaje natural de las anteriores versiones y otra en lenguaje Java. Además incluye también una biblioteca de personajes del conocido juego: los Sims2, cedidos por la compañía Electronic Arts.
- *Storytelling Alice*. Es una versión para edades más tempranas.

La versión beta de *Alice 3* consume muchos recursos y no dispone todavía de tutoriales. Por tal motivo, en este trabajo se ha seleccionado la versión *Alice 2.2*. que tiene un funcionamiento

similar y no tiene los inconvenientes mencionados.

### Scratch

*Scratch* [20] fue creado en el año en 2004 por el MIT Media Lab con el objetivo de mejorar el desarrollo de las habilidades informáticas en jóvenes de 10 a 18 años pertenecientes a comunidades económicamente desfavorecidas y con diversidad cultural (Figura 3). Está profundamente influenciado e inspirado en el trabajo de Seymour Papert y Alan Kay [13].

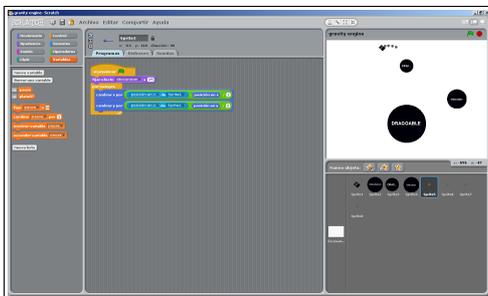


Figura 3. Entorno de programación *Scratch*.

Sus principales características incluyen:

- La programación se realiza mediante construcción de bloques que encajan cuando la sintaxis es correcta.
- Manipulación de distintos medios: imágenes, animaciones, películas y sonido.
- Permite compartir los proyectos con el exterior.
- Soporta múltiples idiomas.

### Greenfoot

El entorno *Greenfoot* [8] (Figura 4) fue creado en 2004 por Poul Henriksen y Michael Kölling (University of Southern Denmark) para introducir la programación orientada a objetos en edades de 14 a 18 años [10].

Para diseñar *Greenfoot*, se realizó un análisis de otras herramientas similares combinando sus mejores aspectos.

Entre sus objetivos principales estaban la visualización de objetos y clases en tiempo de ejecución y conseguir un rendimiento aceptable incluso en hardware antiguo.

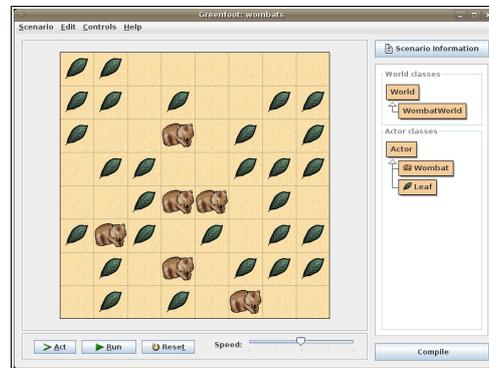


Figura 4. Entorno de programación *Greenfoot*.

## 3. Trabajos relacionados

A continuación se exponen brevemente algunas experiencias recientes de la aplicación de estos entornos en el nivel universitario de la enseñanza informática en los últimos años.

Las experiencias documentadas de *Scratch* son en general positivas. En 2007, la Universidad de Harvard, usó *Scratch* junto con Java en un curso de verano titulado “Computer Science S-1: Great Ideas in Computer Science, the summertime version os a course at Harvard College.”. Como resultado los estudiantes pudieron familiarizarse con los conceptos de la programación sin la distracción de la sintaxis y el 76% de ellos consideró que *Scratch* había tenido una influencia positiva en el aprendizaje de Java. [12]

Un año más tarde, la Universidad ORT Uruguay realizó una experiencia similar concluyendo que los estudiantes que habían usado *Scratch* expresaron una mayor motivación. Sin embargo, no se encontraron evidencias estadísticas ni en las tasas de abandono ni en las calificaciones. [5].

Una última experiencia significativa con *Scratch* fue una propuesta conjunta de dos áreas: informática y música, en la Universidad de Massachusetts Lowell. En el curso “Sound Thinking” se ha pasado de usar HTML y JavaScript a usar el entorno de programación *Scratch* por su mayor adecuación para el aprendizaje tanto de conceptos musicales como computacionales. [19]

En cuanto a *Alice*, hay experiencias documentadas tanto positivas como negativas.

La Universidad del Pacífico, en 2008, decide dejar de utilizar *Alice* junto con Java en los primeros cursos de programación debido al alto número de estudiantes (40%) que pensaron que *Alice* no les ayudó a aprender Java [3].

Más positivos son los resultados de la experiencia en la Universidad de Monmouth que comprobaron que el uso de *Alice* incrementó de forma significativa la probabilidad de éxito en el curso denominado “Programación I” [11].

Para terminar con las experiencias de *Alice*, en el año 2010 la Universidad de North Texas comparó los resultados del uso de *Alice* frente al uso de pseudocódigo antes de comenzar con el lenguaje de programación Java. Sorprendentemente, los estudiantes que usaron *Alice* obtuvieron calificaciones más bajas [7].

Los resultados obtenidos con el uso de *Greenfoot* son similares a los de *Alice*.

La Universidad de Denver no obtuvo resultados concluyentes acerca del uso de *Greenfoot* en un curso de Programación en Java. Aunque los estudiantes mostraron una actitud más positiva con la tecnología en general, fueron pocos los estudiantes que en las pruebas construyeron de forma correcta una sentencia if o un bucle, a pesar de haberlos usado correctamente en los juegos desarrollados durante el curso [1].

Como última experiencia con *Greenfoot* citaremos su uso en el primer curso universitario de la Universidad de Guelph-Humber en Toronto. En este caso, el entorno *Greenfoot* se considera una técnica efectiva para aprender Java. Además, se llama la atención sobre la excelente adecuación de este entorno para la enseñanza a distancia. [18].

#### 4. Metodología de trabajo

La evaluación fue realizada por 26 estudiantes de segundo curso del Grado en Ingeniería Informática de la Universidad a Distancia de Madrid (UDIMA) a través de una actividad de evaluación continua de la asignatura Interacción Persona-Ordenador. Debido al carácter exclusivamente online de la UDIMA, la edad media de los estudiantes que han participado en la evaluación es aproximadamente de 30 años, ya

que muchos de ellos han cursado otros estudios anteriores.

El procedimiento comunicado al estudiante para realizar la evaluación fue el siguiente:

- Obtener la aplicación que se evalúa (una de ellas fue una aplicación web y el resto son aplicaciones de escritorio que fue necesario descargar e instalar).
- Seguir los pasos de un tutorial propuesto para cada entorno con el objetivo de asegurar la uniformidad de la experiencia. En el caso de *Scratch*, *Greenfoot* y *Alice* se usó uno de los tutoriales proporcionados con la aplicación. Para *papert*, se confeccionó un tutorial ad hoc inspirado en un trabajo de Seymour Papert [14].
- Exploración libre del entorno acerca de las cuestiones interesantes para el estudiante durante el tiempo que éste estimara oportuno.
- Evaluar cada entorno mediante un breve cuestionario particular y cumplimentar un cuestionario general, una vez finalizada la evaluación de todos los entornos.

Los cuestionarios para cada uno de los entornos incluían las siguientes preguntas:

1. ¿Qué característica o función del entorno te han gustado más?
2. ¿Qué característica o función del entorno te han gustado menos?
3. Evalúa la facilidad de aprendizaje del entorno. (Escala de 1 a 5).
4. Evalúa la satisfacción general acerca del entorno (Escala de 1 a 5).
5. ¿Qué grado de interés ha despertado en tí este entorno de programación? (Escala de 1 a 5).

El cuestionario general planteaba las siguientes cuestiones:

1. ¿Te parece interesante aplicar esta idea del aprendizaje mediante la programación a alguna asignatura del Grado de Informática de la UDIMA?
2. ¿Por qué motivo?
3. ¿Cuál es el entorno que mejor impresión te ha causado en general?
4. ¿Por qué motivo?

#### 4. Resultados

En primer lugar se presentan los resultados acerca de la facilidad de aprendizaje, satisfacción e interés de los cuatros entornos.

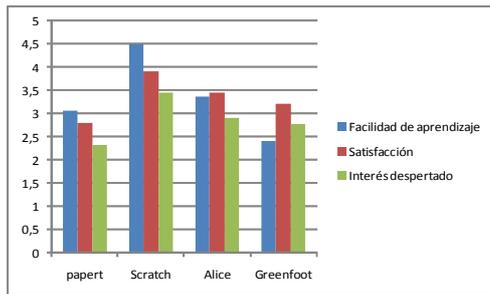


Figura 5. Evaluación de la facilidad de aprendizaje, satisfacción e interés.

Como se puede observar en la figura 5, el entorno que más interés y satisfacción ha causado es también el que tiene una mayor facilidad de aprendizaje: *Scratch*. Este resultado coincide con los resultados positivos descritos en los trabajos citados. Es curioso observar que a pesar de que está dirigido a edades más tempranas haya tenido tan buena acogida en el segundo curso del Grado con estudiantes con una edad media de 30 años.

Del análisis de los comentarios de los estudiantes se deduce que *Scratch* produce satisfacción e interés principalmente por su facilidad de uso y sus múltiples posibilidades. Algunos ejemplos son los siguientes: “Es rápido de entender, con un sólo ejemplo casi se aprende el manejo de la aplicación, es muy intuitivo.”, “Todo es sencillísimo y se obtienen resultados desde el primer momento.”, “La forma en que encajan las condiciones, las estructuras de repetición y las distintas opciones para los personajes. Permite crear animaciones o juegos bastante complejos de una manera intuitiva.”

Después de *Scratch*, el entorno considerado de más fácil aprendizaje es *Alice*, seguido de *papert* y *Greenfoot*.

El motivo de que los estudiantes hayan percibido una mayor dificultad en *Alice* es el uso del paradigma de orientación a objetos y el aspecto similar a entornos profesionales, como se puede comprobar en el siguiente comentario: “Tiene un aspecto demasiado “profesional”. No es

como *Scratch* que e invita a experimentar con él.”, “Lo peor es que si empiezas con procedimientos, métodos y funciones la cosa se complica bastante.”.

El siguiente entorno en facilidad de aprendizaje es *papert*. No obstante, a pesar de su reducido conjunto de instrucciones y su simplicidad en el diseño, el entorno *papert* se considera difícil de aprender debido a la necesidad de conocer la sintaxis para programar. Presentamos a continuación un comentario acerca de este problema: “Tener que aprender toda la sintaxis de un lenguaje de programación nuevo hace que el nivel de satisfacción sea bajo”.

Por último, *Greenfoot* es considerado el más difícil de aprender debido al uso del paradigma de orientación a objetos y a la necesidad de usar Java para crear o modificar métodos. Este hecho se puede comprobar en el siguiente comentario: “La programación se complica un poco más, exige conocimientos sobre programación de objetos y Java.”

Por otra parte, en cuanto a satisfacción e interés el que ha obtenido una puntuación menor es *papert*, seguido de *Greenfoot* y *Alice*.

En cuanto al análisis de las preguntas del cuestionario general, no hubo una preferencia clara sobre qué entorno era considerado el mejor (Figura 6). Tres de los entornos fueron votados de forma similar: *Scratch* y *Greenfoot* fueron seleccionados por 8 estudiantes y *Alice* por 7. Por último, el entorno *papert* fue seleccionado sólo por 2 estudiantes.

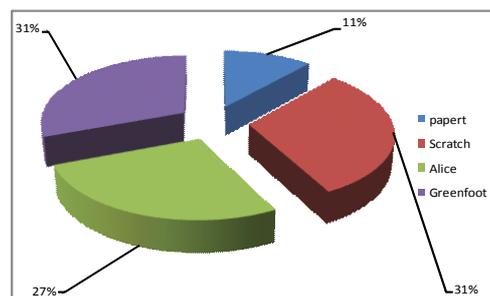


Figura 6. Entornos preferidos por los estudiantes.

Siguiendo con el análisis del cuestionario general, fue sorprendente conocer que 24 de los 26 estudiantes se mostraron interesados en el uso de este tipo de entornos de programación para el

::

aprendizaje dentro del grado en Ingeniería Informática. Sólo 2 estudiantes no consideraron apropiado este tipo de aprendizaje en el nivel universitario.

Algunos de los comentarios de los estudiantes que apoyan esta idea son los siguientes: "Las asignaturas de programación a veces se complican con el tema de la sintaxis, los compiladores, instalaciones, etc. Estas aplicaciones permiten ir al grano y con ejemplos muy concretos. Pienso que si se asimilan estos conocimientos luego en el futuro resultará más sencillo abordar proyectos más complicados y con aplicaciones profesionales.", "Me parecería bastante interesante profundizar en el aprendizaje de estas aplicaciones, sobre todo *Alice* y *Greenfoot*, ya que podrían aportar al estudiante un buen medio para aprender a programar de una manera más atractiva ya que se podrían obtener resultados más vistosos con un menor esfuerzo."

Con el fin de analizar mejor las preferencias de los estudiantes se ha realizado una tabla (Tabla 1) que se puede encontrar en la página siguiente en la que se detallan las características más valoradas por los estudiantes.

En la tabla aparecen las características ordenadas de mayor a menor preferencia. Para ello se ha definido la siguiente métrica: número de veces que la característica ha sido mencionada por los estudiantes en la pregunta del cuestionario de cada entorno: ¿Qué característica te ha gustado más?

La tabla indica el valor de la métrica para cada característica en general y también en particular para cada uno de los entornos.

De esta tabla se puede deducir información importante. Las características de los entornos que más han gustado a los estudiantes son la programación mediante "arrastrar y soltar" (mencionada 14 veces) y la posibilidad de ver proyectos de otras personas y aprender de ellos (mencionada 10 veces).

Esto explica el interés despertado por el entorno *Scratch* que posee ambas características. Además, hay que tener en cuenta que la posibilidad de arrastrar y soltar de *Scratch* se ve reforzada por el código de colores y la metáfora de construcción mediante bloques. Como ya

hemos mencionado, los bloques sólo encajan cuando la sintaxis es correcta (Figura 8).



Figura 8. Ejemplo de código Scratch.

Las siguientes características más importantes son el uso del paradigma de orientación a objetos. (mencionada 9 veces) y los gráficos 3D (mencionada 8 veces).

Estas cuatro primeras características justifican que los entornos más votados hayan sido *Scratch*, *Alice* y *Greenfoot*. Lo que más ha gustado de *Scratch* es el modo de programar; de *Alice* los gráficos 3D; y de *Greenfoot* el paradigma de orientación a objetos.

Otras características también importantes para los estudiantes, pero en menor medida que las anteriores son la introducción de código mediante texto, el uso del lenguaje de programación Java y la posibilidad de trabajar con funciones con parámetros. Todas ellas han sido mencionadas 4 veces por los estudiantes.

Por último, el parecido con entornos de programación profesional y la conexión del entorno con el exterior mediante sensores se han mencionado 3 veces.

## Conclusiones

Aunque estos entornos fueron concebidos para el aprendizaje en edades tempranas, se ha demostrado que es potencialmente posible utilizarlos en estudios universitarios. Un 98% de los estudiantes han mostrado interés en la aplicación de estos entornos en el Grado en Ingeniería Informática. Esto incluye a los estudiantes adultos que también se han visto atraídos por la facilidad de aprendizaje de *Scratch* y la inmediatez de los resultados gráficos de los entornos *Alice* y *Greenfoot*.

Características y funciones	<i>papert 1.0</i>	<i>Scratch 1.4</i>	<i>Alice 2.0</i>	<i>Greenfoot 2.1.0</i>
Programar mediante arrastrar y soltar (14)	No	Sí (9)	Sí (3)	Sí (2)
Visualizar proyectos de otras personas (10)	Sí. (7)	Sí (2)	No	Sí (1)
Paradigma de programación orientada a objetos (9)	No	No	Sí (4)	Sí (5)
Gráficos 3D (8)	No	No	Sí (8)	No
Introducción de código mediante texto (4)	Sí	No	No	Sí (4)
Lenguaje de programación Java (4)	No	No	No	Sí (4)
Funciones con parámetros (4)	Sí (2)	No	Sí (2)	Sí
Parecido con entornos de programación profesionales (3)	No	No	Sí (2)	Sí (1)
Conexión al exterior mediante sensores (3)	No	Sí (3)	No	No
Las características indicadas en la tabla están ordenadas por el número de veces que han sido mencionadas por los estudiantes en la pregunta del cuestionario particular de cada entorno: ¿Qué característica te ha gustado más?				

Tabla 1. Características y funciones de los entornos de programación más valoradas por los estudiantes.

El entorno que ha provocado un mayor interés y satisfacción es *Scratch*, probablemente debido a su gran facilidad de aprendizaje.

*Alice* y *Greenfoot* se han considerado también interesantes para su aplicación en los estudios universitarios de informática debido a su parecido con entornos profesionales y al uso del paradigma de orientación a objetos. Se ha comprobado también que los gráficos 3D de *Alice* han ejercido una gran motivación en los estudiantes.

La caracterización de los entornos (Tabla 1) supone una aportación que puede ser utilizada como para que el profesor o el estudiante seleccione un entorno en función de los objetivos del aprendizaje. Asimismo, la selección de las características más valoradas por los estudiantes, puede constituir un punto de partida en la creación de nuevo entorno.

Es importante reseñar que los resultados de la evaluación se corresponden con los de los trabajos descritos anteriormente.

Por último, hay que tener en cuenta que esta evaluación está realizada en el contexto de una actividad de evaluación continua y las respuestas podrían contener algún sesgo favorable al uso de los entornos.

## Referencias

1. Al-Bow M., Austin D. et al. (2008). Using Greenfoot and games to teach rising 9th and 10th grade novice programmers. In Proceedings of the 2008 ACM SIGGRAPH symposium on Video games (Sandbox '08). ACM, New York, NY, USA, 55-59.
2. Alice. <http://www.alice.org/>
3. Cliburn, D. C. (2008). Student Opinions of Alice in CS1. In: 2008 IEEE Frontiers in Education Conference, Vols 1-3 (pp. 192-197).
4. Dann, W., & Cooper, S. (2009). Alice 3: Concrete to Abstract. Communications of the Acm, 52(8), 27-29.
5. de Kereki, I. F. (2008). Scratch: Applications in Computer Science 1. In: 2008 IEEE Frontiers in Education Conference, Vols 1-3 (pp. 198-202).
6. Feldman S.. 2004. A Conversation with Alan Kay. Queue 2, 9, 20-30.
7. Garlick, R., Cankaya, E. C. (2010). Using Alice in CS1-A Quantitative Experiment.
8. Greenfoot. <http://www.greenfoot.org>.
9. Logo Tree Project [www.elica.net/download/papers/LogoTreeProject.pdf](http://www.elica.net/download/papers/LogoTreeProject.pdf). 2011.
10. Henriksen P., Kölling M. (2004). Greenfoot: combining object visualisation with interaction, Companion to the 19th annual ACM SIGPLAN conference on Object-oriented programming systems, languages, and applications, 24-28.
11. Johnsgard, K., & McDonald, J. (2008). Using Alice in overview courses to improve success rates in programming I. In H. Saiedian & L. Williams (Eds.), 21st Conference on Software Engineering Education and Training, Proceedings (pp. 129-136).
12. Malan, D. J., & Leitner, H. H. (2007). Scratch for Budding Computer Scientists. Sigcse 2007: Proceedings of the Thirty-Eighth Sigcse Technical Symposium on Computer Science Education, 223-227.
13. Maloney, J., Burd, L., Kafai, Y., Rusk, N., Silverman, B., & Resnick, M. (2004). Scratch: A sneak preview. Second International Conference on Creating, Connecting and Collaborating through Computing, Proceedings, 104-109.
14. Papert, S. (1972). Teaching children to be mathematicians vs. teaching about mathematics. International Journal of Mathematics Education and Science Technology.
15. Papert, S. (1981). Mindstorms: Children, computers, and powerful ideas. Basic Books.
16. papert. <http://logo.twentygototen.org/>
17. Pausch, R., Burnette, T., Capehart, A. C., Conway, M., Cosgrove, D., Deline, R., White, J. (1995). Alice - rapid prototyping for virtual-reality. IEEE Computer Graphics and Applications, 15(3), 8-11.
18. Randy J. Gallant and Qusay H. Mahmoud. (2008). Using Greenfoot and a Moon Scenario to teach Java programming in CS1. In Proceedings of the 46th Annual Southeast Regional Conference on XX (ACM-SE 46). ACM, New York, NY, USA, 118-121.
19. Ruthmann, A., Heines, J. M., Greher, G. R., Laidler, P., & Saulters, C. (2010). Teaching Computational Thinking through Musical Live Coding in Scratch.
20. Scratch. <http://scratch.mit.edu/>.

# Evaluación del trabajo en grupo: ¿café para todos?

Manuel Enciso

Carlos Rossi

Eduardo Guzmán

Departamento de Lenguajes y Ciencias de la Computación

E.T.S.Informática

Universidad de Málaga

Campus de Teatinos

29071 Málaga

enciso@lcc.uma.es, rossi@uma.es, eduardo.guzman@lcc.uma.es

## Resumen

Este trabajo tiene como objetivo exponer nuestra experiencia docente en asignaturas basadas en proyectos y en la competencia de trabajo en grupo. Partiendo de nuestro convencimiento de la idoneidad de este tipo de diseño docente para asignaturas relacionadas con el desarrollo de software, analizamos los problemas asociados tanto con la evaluación del alumno como con la estimación de esfuerzo en este tipo de asignaturas, y describimos nuestra aproximación a la resolución de dichos problemas. Finalmente, discutimos sobre una serie de tópicos relacionados con el trabajo en grupo.

## 1. Introducción

Uno de los ejes fundamentales del EEES es el diseño de las asignaturas en base al esfuerzo que ha de desempeñar el alumno. Es, por tanto, esencial que el docente sepa estimar de la más forma precisa posible la carga de trabajo que exigirá a sus alumnos.

Esta estimación es especialmente complicada cuando en la metodología de aprendizaje se marcan técnicas que se alejan del diseño básico basado en clases magistrales y clases de laboratorio/problemas. Así, por ejemplo, hemos de poner especial cuidado en las asignaturas basadas en el método del caso [4], en las que el alumno ha de realizar tareas de cierta duración y complejidad de forma relativamente autónoma. Del mismo modo, las asignaturas fuertemente basadas en el trabajo en equipo precisan de una estimación de esfuerzo que tenga en cuenta la dinámica del trabajo en grupo como un componente más del aprendizaje. Estos métodos docentes, que destacan por fomentar el desarrollo de competencias transversales, han de ser aplicadas con precisión para no requerir un sobreesfuerzo del alumno que pueda provocar conflictos con otras asignaturas.

No hemos seleccionado aleatoriamente estos dos

métodos para apoyar nuestra afirmación, pues ambos métodos suelen venir dados de la mano o, al menos, aprovechan al máximo su potencial cuando son combinados en una misma asignatura. El método del caso y, en general, el aprendizaje basado en proyectos (PBL, Project Based Learning) propone modelos de instrucción centrados en tareas complejas basadas en pequeños retos (problemas o preguntas) que los alumnos deben resolver. Con este fin, los alumnos deben diseñar soluciones, tomar decisiones, llevar a cabo actividades de investigación que les permiten trabajar de forma relativamente autónoma durante periodos de tiempo considerables, y que finalizarán con productos, documentos o presentaciones [6]. Esta estrategia didáctica supone un cambio radical en la instrucción más tradicional basada en un currículo de conceptos que representaban el centro del proceso didáctico. Con el aprendizaje basado en proyectos, el centro del proceso es el propio alumno, el cual debe ir descubriendo de forma progresiva y por sus propios medios los conceptos del currículo; es decir, los alumnos encuentran y aprenden los conceptos utilizando como vía el proyecto.

Diversos autores han puesto de manifiesto que los métodos docentes basados en PBL acostumbran a ser más motivadores para los estudiantes, lo cual repercute en aspectos tales como el rendimiento académico, y la persistencia en los estudios. De hecho, el objetivo fundamental de la introducción generalizada de PBL en algunas universidades ha sido el mejorar ostensiblemente el bajo rendimiento académico de los estudiantes (e.g. [2]).

Por otro lado, si el proyecto propuesto queremos que tenga la entidad necesaria para ser un punto importante en la evaluación de la asignatura, debemos de apoyarnos en el trabajo en equipo. Esta es una de las vías principales para asegurar, como hemos dicho antes, que nuestra asignatura basada en proyectos no se convierte en una asignatura "tiburón" que devora las horas que el alumno debe dedicar a otras

asignaturas de su currículum.

Pero no debemos olvidar que el diseño de las asignaturas en el EEES deben tener en cuenta tanto la carga de trabajo del alumno, como la dedicación del profesor. Como es bien sabido, para que un diseño de asignatura sea sostenible, más allá de la necesaria sobrecarga que supone la implantación de la asignatura por primera vez, las tareas de dedicación deben tender a estabilizar el tiempo que consumen. Este planteamiento es difícil de cumplir en las asignaturas basadas en proyectos y en el trabajo en grupo. A menudo la principal dificultad que los profesores plantean para su implantación es precisamente el tiempo que ha de emplearse en la evaluación. Las tareas entregadas por los alumnos en cada hito deben ser revisadas por el profesor, para que el proceso de aprendizaje sea incremental, como lo es el propio desarrollo del trabajo. Esto exige un seguimiento muy continuado y la revisión de trabajos (documentación o código) que no es absoluto una tarea trivial.

Pero el problema de la evaluación no acaba en la corrección de las entregas. Además, el profesor debería poner énfasis en realizar una evaluación personalizada del alumno. Esto exige desgranar el trabajo que cada uno de los miembros del equipo realiza en cada una de las entregas y disponer de elementos que motiven y razonen calificaciones diferentes dentro de un mismo grupo de trabajo. Es decir, la evaluación ya de por sí costosa en un método basado en proyectos, se hace más costosa si queremos asignar a cada alumno una calificación equitativa, acorde con su esfuerzo y aprendizaje. En estas circunstancias, a menudo muchos docentes deciden que una asignatura así no es sostenible y optan por un diseño más clásico con una evaluación que no presente esta sobrecarga de trabajo.

## 2. Objetivos

El objetivo de este trabajo es presentar nuestra experiencia de ocho cursos de docencia de asignaturas diseñadas en base a dos pilares comunes: el aprendizaje basado en proyectos y la competencia transversal de trabajo en grupo. Además de la propia descripción de la experiencia, nuestra intención es enfatizar la forma en que hemos resuelto el proceso de evaluación en dichas asignaturas, apoyándonos en una herramienta de autoevaluación diseñada ad hoc.

La elección de los dos pilares básicos indicados

anteriormente viene determinada por las características de las asignaturas implicadas en la experiencia: asignaturas de último curso y de ámbitos muy cercanos al desarrollo profesional de la Informática (Bases de Datos, Ingeniería del Software). Nuestra trayectoria de veinte años en la docencia de estas asignaturas nos dice que el aprendizaje del alumno es muy limitado si no son diseñadas con una fuerte componente práctica, y ésta debe ser lo más cercana posible a la realidad profesional del informático, que no es posible concebir hoy día sin cooperación (presencial o virtual) e integración en un equipo de trabajo.

### 2.1. Evaluación del grupo vs. evaluación individual

Como se indicó anteriormente, uno de los principales handicaps de asignaturas basadas en trabajo en grupo es la extracción de una evaluación individual a partir de la evaluación del grupo. El handicap es tanto para el profesor, preocupado por otorgar una calificación justa a cada estudiante, como para el alumno, preocupado por que su calificación no se vea “arrastrada” a la baja por la de sus compañeros de grupo.

Para la extracción de la evaluación individual nos apoyamos en tutorías y sesiones de seguimiento del trabajo técnico realizado por grupos. Sin embargo consideramos que esto no es suficiente, y que es necesaria la utilización de rúbricas para extraer información de características como la cantidad y calidad del trabajo, iniciativa, compromiso, puntualidad en las entregas, etc. de cada uno de los miembros del grupo. Para ello, precisamos que esas rúbricas sean completadas en un proceso de autoevaluación de cada alumno, y de “evaluación mutua” con sus compañeros de grupo. Esto genera un volumen de información considerable, teniendo en cuenta que las rúbricas deben ser cumplimentadas varias veces durante el curso (normalmente una vez por cada entrega de una fase del proyecto).

### 2.2. Herramienta de evaluación

Para facilitar el manejo y explotación de la información incluida en las rúbricas hemos desarrollado una herramienta denominada Puzzle, actualmente en su segunda versión. Sin ella, consideramos que el

proceso de evaluación mediante rúbricas no es sostenible.

La funcionalidad básica de Puzzle es la edición y gestión de informes de actividad (que pueden ser configurados según las necesidades del profesor, con distintos formatos de pregunta/respuesta), la gestión de asignaturas, profesores, alumnos y grupos. Puzzle permite, además de la consulta y evaluación de los informes de cada alumno, la generación de estadísticas y gráficas resumidas por grupo y por alumno, que a nuestro entender es una de las aportaciones clave de Puzzle para que el profesor pueda realizar de forma eficiente y en un tiempo razonable el proceso de evaluación de una asignatura basada en el trabajo en grupo.

En el apéndice A se muestran algunas pantallas de esta herramienta y se comentan algunas de sus funcionalidades.

### 3. Tópicos

El objetivo de esta sección es analizar, uno por uno, diversos tópicos sobre el trabajo en grupo. Para rebatir cada uno de ellos, nos hemos basado en nuestra propia experiencia docente y en los diversos estudios que hemos ido realizando durante nuestros años de docencia.

#### 3.1. Uniformidad en las calificaciones de los miembros del grupo

Nuestra reflexión se centra en la siguiente pregunta: ¿todos los miembros del grupo suelen sacar calificaciones similares en la asignatura? En otras palabras, intentamos saber si el hecho de que una asignatura esté basada en proyectos, hace que los miembros del grupo obtengan calificaciones parecidas también en otras actividades de la asignatura (exámenes o actividades individuales).

La respuesta es **sí**: aproximadamente en el 75% de los grupos la variación de las calificaciones es de sólo un nivel en la escala habitual (suspense, aprobado, notable, etc.). En este estudio se ha considerado un esquema en el que la calificación del proyecto es un 50% de la calificación de la asignatura, siendo el 50% restante el resultado de la evaluación de un examen (individual). La conclusión que obtenemos es que el trabajo en grupo genera un cierto efecto de “arrastre”, que hace que los miembros de un equi-

po obtengan también calificaciones similares en los exámenes.

#### 3.2. Evolución en competencias

En este tópico reflexionamos sobre la evolución de los alumnos en diferentes aspectos relacionados con la competencia de trabajo en grupo. En este sentido, hemos de destacar el proceso de refinamiento de las rúbricas que hemos ido empleando a lo largo de los sucesivos cursos. Inicialmente partimos de una evaluación básica en la que únicamente teníamos en cuenta la cantidad y la calidad del trabajo realizado por cada miembro del grupo. Pronto detectamos que era necesario disponer de rúbricas más precisas, que nos permitiesen obtener información más determinante para extraer la evaluación individual a partir de la calificación del grupo. En este sentido, estamos empleando actualmente una rúbrica que evalúa los siguientes aspectos:

- Calidad del trabajo realizado por el alumno
- Cantidad, habitualmente medida en número de horas
- Proporción de trabajo realizada por el alumno en relación al esfuerzo total desarrollado por el grupo
- Cumplimiento, es decir, si entrega las tareas completas o incompletas, y si éstas necesitan seguimiento o revisión. Esta información nos sirven para contrastar las anteriores mediciones de calidad y cantidad de trabajo.
- Actitud del alumno, es decir, si es activo y con iniciativa, o pasivo.
- Puntualidad, para saber si el alumno entrega sus tareas a tiempo para su integración en el trabajo global del grupo.

La utilización de rúbricas más complejas naturalmente genera un volumen de información, cuya gestión no sería sostenible de no contar con una aplicación como Puzzle. Además, la herramienta nos ha facilitado poder sacar conclusiones respecto a la reflexión que planteábamos al inicio de este apartado: ¿los alumnos evolucionan en estas competencias como consecuencia del trabajo en grupo?

### 3.2.1. La formación del grupo

#### 3.2.2. ¿Cuándo se forma el grupo?

A lo largo de estos años hemos observado también que un factor muy influyente en los resultados académicos de un trabajo en grupo viene dado también por el momento la gestación del grupo.

En varias de nuestras asignaturas son los propios alumnos los que forman los grupos, que indican al profesor en los primeros días del curso los componentes de cada uno de ellos. Es más, en asignaturas que ya tienen “tradición” de docencia basada en grupos, los alumnos nos comunican que en muchos casos forman los grupos ya en el curso anterior al que van a matricularse en la asignatura.

En cambio, siempre hay también un número de alumnos “rezagados” o poco integrados que no se incorporan a ninguno de los grupos. En estos casos, suele ser el profesor el que mediante los foros de la asignatura los pone en contacto para que formen un grupo.

Hemos observado que, en una gran mayoría de los casos estos grupos suelen tener peores calificaciones que el resto. Nuestra conclusión es que el problema no viene dado por problemas de comunicación o por un menor grado de “amistad” entre los miembros del grupo, sino por el menor grado de compromiso con la asignatura que presentan habitualmente los alumnos con este perfil.

#### 3.2.3. ¿Quién forma el grupo?

Otra reflexión hace referencia a la influencia que puede tener en los resultados de un grupo el hecho de que sus componentes tengan una relación de amistad. En este sentido disponemos de datos subjetivos y anónimos recogidos en encuestas realizadas al final de cada curso.

En general la mayoría de las opiniones coincide en la importancia de conocer previamente a los miembros del grupo, pero no en el sentido de amistad, sino en tener la certeza de que son responsables, con capacidad de trabajo y comprometidos con la asignatura.

### 3.3. Los alumnos copian el proyecto

Este no es absoluto un tópico infundado. Cuando el desarrollo de la asignatura se produce en el aula

pero también en combinación con actividades fuera de ella y la entrega de dichas actividades se considera una parte importante de la nota, sería irresponsable por nuestra parte no prever que se pueden dar situaciones de plagio. Añadimos además que en todas nuestras experiencias, el proyecto o caso propuesto es el mismo para todos los alumnos de la asignatura. Incluso si un alumno cursa varias de las asignaturas involucradas en nuestros proyectos de innovación educativa, se encontrará con el mismo problema en todas ellas, aunque naturalmente tendrá que desarrollar tareas diferentes según la propia temática de cada asignatura.

Esta circunstancia de prevención de plagio ha sido reflejada convenientemente en experiencias de otras universidades. Cabe destacar quizá por su profundo desarrollo y también por su visión temprana del problema la experiencia de la Universidad de Nottingham [8]. Desde entonces se han desarrollado algunas herramientas para detectar plagio entre documentos y también, más específicamente, entre trabajos de desarrollo de software en el ámbito universitario [11, 1].

Sin menoscabar la utilidad de estas herramientas y de la necesidad de que el docente prevenga y detecte estas situaciones, este tópico se ha visto desmontado en nuestras experiencias. En ninguna de ellas hemos detectado un plagio en la entrega de proyectos. Desde nuestra perspectiva, esta situación se debe a dos factores, la camaradería dentro del equipo y la competitividad generada *ad extra*.

La dinámica de equipo desarrollada dentro de los grupos de trabajo, reforzada en muchos casos por relaciones de amistad dentro de ellos, hace que el trabajo generado sea visto como un bien del propio equipo y por tanto no es posible considerarlo un bien propio del que hacer mal uso. De hecho muchos alumnos consideran las filtraciones del código común como una falta de identificación con el grupo y una cierta traición al mismo. Este sentimiento de equipo sirve así como primer nivel de prevención de plagio.

En alguna de las asignaturas el desarrollo del trabajo culmina, además de con la calificación final, con la entrega de un galardón al mejor trabajo. En las asignaturas donde esto se hace la competitividad entre los equipos se dispara. Los equipos no solo aspiran a sacar una buena calificación, sino que aspiran a ser destacados como el mejor proyecto de ese año.

En dicha elección se promueve la presentación de los trabajos ante un tribunal ajeno a la asignatura en el que además intervienen algunos profesionales del sector. Este galardón no tiene reconocimiento académico en la nota (por tratarse de un tribunal mixto), pero si lleva consigo un reconocimiento oficial por parte del departamento mediante certificado.

Para el mejor seguimiento del caso, creamos un espacio en el campus virtual donde poder debatir diferentes dudas y aproximaciones del mismo. Como evidencia de esta competitividad, en el transcurso de los años hemos notado un fuerte decrecimiento de la actividad en dichos foros. La razón principal es que los alumnos no muestran sus progresos al resto de los equipos en pro de conseguir sorprender al resto en la puja por ser el mejor proyecto del curso.

### 3.4. Es importante que el grupo sea uniforme

Cuando se propone a los alumnos el trabajo en equipo, se hace necesario dotar a los mismos de unas ciertas herramientas que promuevan un trabajo eficaz. De otro modo el trabajo en equipo podría ser un obstáculo más en el aprendizaje en lugar de una ventaja. Si proponemos esta metodología como medio para ahorrar tiempo en el estudiante, no podemos permitir que la dinámica del trabajo en equipo sea un agente más de consumo del tiempo.

Habitualmente proporcionamos a los alumnos unas nociones básicas del trabajo en equipo [5] y de la gestión de tiempo [3]. Como se observa por las referencias, no se trata de incluir en el temario de las asignaturas este apartado como un tema más, por lo que nuestra intención no es realizar una profunda formación en dinámicas de grupos sino dotar de herramientas básicas que ayuden a planear el trabajo sabiendo que este se va a realizar en equipo y se use adecuadamente el entorno que lo rodea (reuniones, comunicación, traza de documentación, etc).

En estas experiencias hemos detectado que cuando la asignatura está vertebrada en torno al proyecto, la propia diversidad de tareas dentro del mismo hace muy recomendable que los grupos no sean uniformes. En un desarrollo de software completo hay tareas tan diversas como el diseño siguiendo una metodología de software, la programación del mismo, el diseño de la interfaz, la exposición del mismo, etc. Pero la dinámica propia del grupo también exige una serie de habilidades sociales como la facilidad de

comunicación, el saber trasladar y convencer de las ideas que queremos aportar, el liderazgo del equipo, etc. Este escenario, como bien se presenta en [10] es un escenario perfecto para el desarrollo del aprendizaje en el EEES pues el alumno puede encontrar siempre un tipo de tarea donde mostrará sus habilidades y, por tanto, no se verá limitado en unas pocas actividades del mismo carácter.

Sin embargo, en este punto nos hemos encontrado con un hecho que debemos mencionar. Hemos constatado a lo largo de los cursos una mayor dificultad para abordar este tipo de asignaturas en aquellos grupos donde el número de alumnos extranjeros es significativo. Achacamos este hecho a la necesidad de enfrentarse a un problema que no está enunciado en una página sino que se presenta habitualmente en forma de charla presencial desde donde se debe hacer la captura de requisitos [9]. Aunque la charla puede ser reproducida más tarde al disponer los alumnos del audio de las entrevistas, hemos observado una gran dificultad para realizar esta tarea sin un dominio fluido del castellano.

Al mismo tiempo, notamos que a los diferentes caracteres del equipo, hay que unir diferentes culturas que aportan riqueza pero al mismo tiempo hacen más compleja la dinámica de equipo. Ante esta situación vemos interesante que el equipo dispusiera de algún tipo de herramienta básica relacionada con la interculturalidad, lo que ayudaría a hacer más fluida la comunicación y el trabajo de este tipo de equipos.

### 3.5. La evaluación en grupos es injusta

En muchas ocasiones, los alumnos se quejan de que en los trabajos en grupo, la cantidad y calidad del trabajo realizado difiere notablemente entre los miembros de cada grupo. Sin embargo, si la evaluación se hace de forma global, todos los alumnos recibirían la misma nota sin ponderar su trabajo real. Con este fin, resulta muy útil el uso de la *evaluación por pares*. A través de este tipo de evaluación, cada miembro del equipo evalúa al resto de sus compañeros de acuerdo con un conjunto de rúbricas especificadas por el profesor. Éstas cuantifican y cualifican el trabajo realizado por el compañero. De esta forma, la responsabilidad de la "justicia" de la evaluación recae en parte en el propio alumno que deberá realizar este proceso de forma honesta e imparcial.

### 3.6. El alumno evalúa siempre bien a sus compañeros

Un tópico recurrente en el aprendizaje basado en proyectos, que suele proceder de profesores que no han aplicado nunca este tipo de metodologías, es que ningún alumno evaluará negativamente a sus compañeros de grupo. Precisamente, esta afirmación es empleada frecuentemente como justificación para no incluir dinámicas de trabajo en grupo en el aula.

Si bien es cierto que muchos estudiantes se muestran inicialmente reticentes a poner una mala calificación a sus compañeros, hay tres factores que fomentan la sinceridad en la evaluación. El primero de ellos es el carácter confidencial de este tipo de rúbricas. Es esencial que el alumno que desempeña el rol de evaluador se sienta libre de poder calificar a sus compañeros sin que éstos conozcan (de forma detallada) ni las evaluaciones concretas ni quién es responsable de ellas.

El segundo factor importante a la hora de fomentar la sinceridad y objetividad de la evaluación es que ésta se lleve a cabo de forma iterativa. En las dinámicas de trabajo en grupo basado en proyectos, hemos observado que es importante que cada grupo realice entregas parciales del proyecto. Esto permite realizar un seguimiento cuantitativo del trabajo desarrollado en un determinado periodo a la vez que obliga a los alumnos a dosificar el esfuerzo que aplican a la asignatura a lo largo del cuatrimestre (o del curso en el caso de asignaturas de carácter anual). Así, si a estas entregas les sigue un proceso de evaluación por pares del trabajo desarrollado en esa iteración, la evaluación suele ser más sesgada en las primeras iteraciones a favor del alumno evaluado. Sin embargo, conforme se suceden las diversas entregas, el evaluador se vuelve más objetivo y suele perder el reparo a poner una baja calificación a uno de sus compañeros.

Finalmente, la inmediatez de la evaluación por pares es también importante. Si se implementa una dinámica de entregas parciales durante el transcurso de la asignatura, es esencial que los alumnos realicen la evaluación de sus compañeros una vez hecha la entrega del trabajo en grupo. Normalmente, la mayoría de conflictos entre los miembros de un grupo suelen aflorar cuando se acerca la fecha de entrega. Sin embargo, conforme más tiempo transcurre desde que la entrega tuvo lugar, más sesgada se vuelve la evaluación, normalmente a favor de aquellos miem-

bros del equipo que menos trabajo han realizado y que menos interés han mostrado.

### 3.7. La docencia basada en proyectos no mejora la tasa de abandonos

Uno de los grandes problemas de la docencia universitaria, especialmente en las carreras de perfil técnico, es el número de alumnos que dejan una asignatura, sobre todo la primera vez que la cursan. Este problema es especialmente significativo en los primeros cursos, en los que además, estas pautas suelen llevar a que un porcentaje notable de alumnos abandone la carrera.

Nuestra experiencia personal es que en las asignaturas en las que empleamos el aprendizaje basado en proyectos, la tasa de abandonos es ínfima y en muchos casos inexistente. Las dinámicas participativas de este tipo de docencia provocan que el alumno se involucre, de forma más o menos activa, en su propio proceso de aprendizaje. Un aspecto que afianza notablemente esta implicación es hacer que el alumno se vea inmerso en una simulación (más o menos real) de lo que va a ser su futuro profesional. Cuando el aprendizaje basado en proyectos se centra en casos reales y se somete a los alumnos a situaciones que se asemejan a las que tendrán que vivir en su vida profesional, se despierta en ellos un interés especial en este tipo de asignaturas.

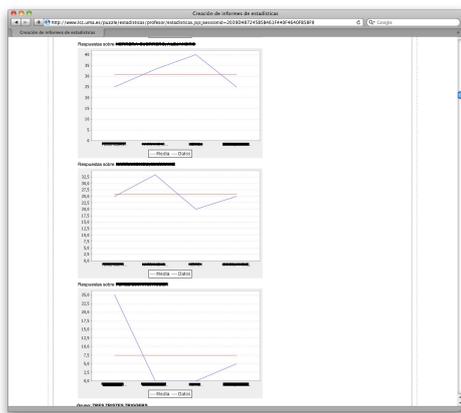
Además, el hecho de no trabajar de forma aislada, sino cooperar y colaborar con otros estudiantes, suele provocar en la mayoría de los alumnos un sentimiento de compromiso con el grupo, reforzado por la interdependencia que se establece con ellos. Igualmente, el proceso de aprendizaje se ve enriquecido por esa colaboración y por ese proceso de descubrimiento del conocimiento tanto individual como grupal.

#### A. Puzzle: herramienta de informes de actividad

En este apéndice queremos mostrar algunas de las funcionalidades que hacen de Puzzle una herramienta muy importante para hacer posible un seguimiento de la actividad de los grupos de trabajo. Las características de Puzzle permiten que esta tarea de recopilación de información sobre el progreso de los grupos y su evaluación se convierta en una tarea acotada



son numéricas (número de horas, calidad de la entrega, etc). De esta forma detectamos rápidamente los casos de contestaciones incoherentes en un grupo. En este ejemplo podemos ver como el último alumno obtiene una ponderación muy pobre del resto de sus compañeros.



Informe con gráficas

## Referencias

- [1] A. Aiken, *MOSS: A System for Detecting Software Plagiarism*, <http://www.cs.berkeley.edu/~aiken/moss.html>. Oxford University Press. Accessed 10th August 2004.
- [2] J. Alcober, S. Ruiz y M. Valero, (2003). *Evaluación de la implantación del aprendizaje basado en proyectos en la Escuela Politécnica de Superior de Castelldefels*. Universidad Politécnica de Cataluña.
- [3] B. Eilam y I. Aharon, *Students' planning in the process of self-regulated learning*, *Contemporary Educational Psychology*, Vol. 28, No. 3, pp. 304-334, 2003.
- [4] J. Erskine, M. Leenders y L. Mauffette, *Learning with cases*. Richard Ivey School of Business. The University of Western Ontario. 2001.
- [5] S. B. Fiechtner y E. A. Davis, *Why Some Groups Fail: A Survey of Students' Experiences with Learning Groups.*, en A. Goodsell, M. Maher, V. Tinto & Associates (eds.), *Collaborative Learning: A Sourcebook for Higher Education*. University Park: National Center on Postsecondary Teaching, Learning, and Assessment, Pennsylvania State University. 1992.
- [6] T. Markham, J. Mergendoller, J. Larner y J. Ravitz, J. *Introduction to project based learning*. Project Based Learning Handbook (2nd revised/special edition, pp. 3-8). 2003.
- [7] B. Oakley, R.M. Felder, R. Brent y I. Elhadj, *Coping with Hitchhikers and Couch Potatoes on Teams. Turning Student Groups into Effective Teams*, *Journal of Student Centered Learning* Vol. 2, No. 1, 2004/9.
- [8] PBLE: Project Based Learning in Engineering, *A Guide to Learning Engineering Through Projects*, University of Nottingham. Nov 2003.
- [9] C. Rossi, E. Guzmán, M. Enciso y Francisco Durán, *Desarrollos Cuasi-profesionales: La empresa en el aula*, Actas de las XIV Jornadas de Enseñanza universitaria de la Informática, JENUI 2008.
- [10] M. Valero, *Cómo nos ayuda el Tour de Francia en el diseño de programas docentes centrados en el aprendizaje*, *Novática*, Vol. 170, pp. 42-47, 2004.
- [11] P. Vamplew y J. Dermoudy, *An anti-plagiarism editor for software development courses*, Proceedings of the 7th Australasian conference on Computing education - Volume 42. ACE '05. Australian Computer Society, Inc. pp. 83-90. 2005.

# Una aproximación metodológica al cambio de paradigma en el profesional de la informática

José Lucas Grillo Lorenzo, Francisco de Sande, Vicente Blanco  
Dep. de Estadística, I. O. y Computación  
Universidad de La Laguna  
Avda. Astrofísico F. Sánchez, s/n  
38271. La Laguna. S/C de Tenerife  
jlucas.gl@gmail.com, fsande@ull.es

## Resumen

El profesional de la informática que el tejido empresarial demanda en la actualidad posee una serie de capacidades que tradicionalmente no se fomentan en las Escuelas de ingeniería informática. La implantación en nuestro país del Espacio Europeo de Educación Superior es una oportunidad inmejorable para reflexionar sobre la práctica docente que los educadores en esta materia venimos desarrollando. A pesar de que pedagogos y psicólogos del aprendizaje llevan ya tiempo indicando que los métodos tradicionales de enseñanza/aprendizaje deben evolucionar para incorporar lo que hoy sabemos respecto a cómo aprendemos, es frecuente observar en nuestro entorno el “cambio para que todo siga igual”.

En este trabajo se presenta una metodología de enseñanza/aprendizaje basada en actividades que utilizamos en la asignatura *Compiladores*. Las principales características de la metodología son: alto grado de autonomía por parte del estudiante, trabajo colaborativo, realimentación rápida y participación en el aprendizaje de todo el grupo.

Los resultados que hemos recogido a través de encuestas al alumnado nos indican un moderado grado de aceptación. Hemos conseguido disminuir los niveles de abandono de la asignatura a través de una metodología en la que el estudiante se siente por una parte obligado, pero también implicado en su propio proceso de aprendizaje.

## Palabras clave

Actividades, metodología, *Compiladores*, feedback, competencias.

### 1. El cambio de paradigma: del informático 1.0 al Informático 2.0

En la actualidad se ha producido un cambio en lo que el mercado y la sociedad demandan de un profesional de la informática, y podemos afirmar que hemos completado la transición del informático 1.0 al informático 2.0.

El informático 1.0 era un profesional destinado a trabajar la mayor parte del tiempo en los sótanos del edificio empresarial, ante una pantalla de fósforo verde y con pocas oportunidades de relación con su entorno. Su indumentaria no era una cuestión relevante en el desempeño de sus funciones, y el dominio de la programación era su competencia más definitoria. El Informático 2.0 es un nuevo profesional que posee grandes cualidades comunicativas, lidera y coordina equipos de trabajo, viaja frecuentemente y cuya imagen pública corresponde a la de un alto ejecutivo.

Así pues, las Escuelas de informática se enfrentan hoy en día al desafío de formar profesionales con este nuevo perfil. Si se analiza la última versión del Computing Curricula de la ACM [2] descubrimos que el Informático 2.0 precisa de una serie de competencias que difícilmente son compatibles con una enseñanza centrada casi

exclusivamente en la clase magistral. Entre estas competencias destacamos las siguientes:

- Capacidad de liderazgo y de trabajo en equipo
- Capacidad expresarse en público de forma efectiva y convincente
- Preparación para la gestión de las relaciones interpersonales
- Capacidad de síntesis
- Habilidad para ejercer la crítica constructiva

Motivados por este cambio de paradigma, en la asignatura de *Compiladores* que se imparte en la Escuela Técnica Superior de Ingeniería Informática (ETSII) de la Universidad de La Laguna (ULL) hace ya cinco años decidimos dar un giro a la forma tradicional de estudiar la asignatura para enfatizar el desarrollo de estas nuevas capacidades. La excusa perfecta para implantar estos cambios fue la implantación del EEES, que propone un mayor énfasis en el papel del discente en relación a su propio aprendizaje.

La docencia de *Compiladores* se basa en una serie de actividades [4] en las que la participación del alumno es el elemento central y que estructuran el tiempo de dedicación del alumno. En la mayor parte de los casos el profesor juega el papel de orientador del alumnado.

En este trabajo exponemos una visión general de la metodología utilizada en la asignatura, tratando de incluir los puntos de vista de profesorado y alumnado. En la Sección 2 se describe la organización de la asignatura, para posteriormente exponer en profundidad (Sección 3) cada una de sus actividades pedagógicas. En la Sección 4 se analizan los resultados de las encuestas realizadas al alumnado para valorar su participación en la asignatura. Finalmente, en la Sección 5 se presentan unas conclusiones a partir de las cuales pretendemos mejorar el planteamiento docente de la asignatura.

## 2. Organización de la asignatura

En la ULL, *Compiladores* es una asignatura de 6 créditos (3 teóricos y 3 prácticos) obligatoria en el tercer curso de la titulación de Ingeniería Técnica en Informática de Sistemas. La asignatura se centra en el estudio del diseño de compiladores y es la única asignatura con este perfil de contenidos que estudian los ingenieros técnicos. Se estudia en el primer cuatrimestre y es una asignatura que se impartirá por última vez el curso 2011-2012 ante la implantación del nuevo título de Graduado en Informática.

En el primer tema de la asignatura se introduce el concepto de compilador y las fases del proceso de compilación. Se avanza luego en el estudio de las etapas de la compilación, partiendo de la entrada en código fuente hasta las últimas etapas de optimización y generación del código. El trabajo de la asignatura está guiado a través del aula virtual de la asignatura que está soportada por el Campus virtual de la ULL [?], basado en la plataforma Moodle.

En *Compiladores* se usa un proceso de evaluación continuada, uno de cuyos objetivos es [5, “mantener bien informado al alumno sobre el estado de su aprendizaje”]. En este sentido propiciamos que el alumno obtenga una retroalimentación razonablemente rápida del nivel de calidad de sus ejercicios, y en el que el habitual *examen final* de la asignatura ha sido eliminado. Cada actividad diferente realizada a lo largo del curso conlleva su evaluación separada, y la calificación final de la asignatura se obtiene mediante la combinación ponderada de las calificaciones de cada actividad. Con esta metodología pretendemos valorar positivamente el esfuerzo y la constancia de los alumnos para alcanzar los objetivos propuestos.

Las actividades que se realizan en la asignatura, junto con la proporción en que las calificaciones de cada una contribuyen a la nota final de la asignatura son: Resúmenes escritos 15 %, Presentaciones 10 %, Prácticas 40 %, Cuestionarios 25 % y Participación en Foros 10 %.

Teniendo en cuenta que *Compiladores* se estructura en 9 temas, que se realizan 4 sesiones de clase presencial a la semana y que el cuatrimestre suele durar unas 14 semanas sin tener en cuenta los días festivos, la Tabla 1 detalla una estimación del tiempo mínimo que un estudiante debería dedicar a la asignatura. La columna *Rep.* indica el número de repeticiones de la actividad correspondiente.

En la evaluación de Resúmenes y cuestionarios así como parcialmente en la de Presentaciones, y siguiendo las ideas expuestas en [5] utilizamos un sistema de co-evaluación. Para ello, en cada tipo de actividad evaluada los alumnos disponen de una rúbrica: un documento que les sirve de guía a la hora de evaluar el trabajo de un compañero. En las rúbricas se indican los criterios que, a juicio del profesor, deben utilizarse a la hora de evaluar cada trabajo. La evaluación de Prácticas, presentaciones y participaciones en foros se realiza directa y exclusivamente por parte del profesorado.

### 3. Actividades de la asignatura

Las actividades son el eje central del aprendizaje del alumno en *Compiladores*. Asociadas con cada tema de la asignatura, se realizan siguiendo la secuencia que muestra la Figura 1. Inicialmente el profesor emplaza a los alumnos a estudiar un nuevo tema y a elaborar un resumen escrito del mismo. Se trata en este caso de trabajo no presencial. Previamente a que la mayoría de estudiantes comiencen a trabajar en el nuevo tema, un

equipo de trabajo al que llamaremos *grupo especializado* del tema es encargado de un estudio más profundo del mismo, con el apoyo del profesor. El *grupo especializado* ha de preparar adicionalmente una presentación oral que expondrá a sus compañeros.

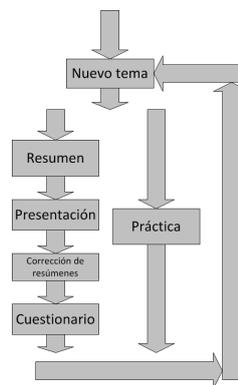


Figura 1: Secuenciación de las actividades

Discrecionalmente, dependiendo de la complejidad del tema estudiado, así como de la calidad del trabajo realizado por el *grupo especializado*, el profesor dedica una sesión presencial con el formato de clase magistral a aclarar posibles dudas. Después de esta actividad, se dedica otra sesión presencial a la corrección de resúmenes por parte de los propios alumnos, con la asistencia del profesor y apoyo de rúbricas.

Asociada con cada tema de la asignatura hay al menos una práctica de laboratorio que los alumnos preparan de forma autónoma, no presencial y que es evaluada semanalmente en el laboratorio de prácticas.

Los cuestionarios suele ser la última actividad que se realiza asociada con cada tema. Se trata de una actividad eminentemente evaluativa en la que se pretende contrastar la calidad del aprendizaje.

En todas las actividades se utilizan los foros del aula virtual para resolver dudas del alumnado. Este elemento es particularmente útil en la resolución de dudas relativas a la implementación de las prácticas de laboratorio. También se usa intensivamente el

	Horas	Rep.	
Prácticas	10	12	120
Presentaciones	10	1	10
Resúmenes escritos	3	9	27
Asistencia a clase	4	14	56
TOTAL			213

Cuadro 1: Horas estimadas que un alumno dedica a estudiar *Compiladores*

aula virtual como punto de encuentro donde suministramos al alumnado todo el material con que ha de trabajar.

Describimos a continuación, con mayor grado de detalle cada una de las actividades.

### 3.1. Resúmenes de temas de la asignatura

El estudio de cada tema comienza con una indicación por parte del profesor de las referencias fundamentales. Los alumnos estudian la lección usando esta bibliografía, otras fuentes adicionales, y materiales complementarios disponibles en el aula virtual.

Como resultado de su estudio, los alumnos preparan un entregable en el que resumen los conceptos básicos del tema estudiado. El resumen se entrega escrito a mano con la limitación de un folio de extensión. En el entregable los alumnos plasman también las dudas halladas en el estudio del tema. Esta actividad representa el primer contacto con cada lección, y se realiza de forma individual fuera del horario presencial.

El objetivo que perseguimos al pedir que el entregable se escriba a mano es doble. En primer lugar propiciamos que los estudiantes tomen conciencia de la calidad de su ortografía y caligrafía, así como de sus capacidades a la hora de presentar trabajos escritos. En segundo lugar, evitamos la copia de trabajos.

Previo a su valoración por parte del profesor, estos resúmenes son evaluados de forma cooperativa por los estudiantes de la clase: en una sesión presencial, cada estudiante corrige de forma anónima los trabajos de cuatro o cinco compañeros. Para realizar correcciones, los alumnos disponen de una rúbrica que les orienta respecto a los criterios de evaluación.

La rúbrica dirige a los alumnos a evaluar positivamente (con diferentes factores de ponderación) las características que el profesor considera positivas: originalidad, calidad de la presentación, capacidad de síntesis, corrección, etc. Originalmente el alumno evaluado parte de una puntuación de 100, de la que se pueden descontar puntos por diferentes aspectos evaluados. Uno de los

factores clave de estas correcciones consiste en que el alumno que actúa como evaluador ha de justificar por escrito (mediante comentarios al margen del ejercicio que corrige) la razón por la que ha descontado puntuación al trabajo evaluado.

### 3.2. Exposiciones orales en clase

Todos los alumnos del curso tienen al menos una oportunidad de realizar una presentación oral de un tema de la asignatura ante el resto de sus compañeros, que actúan como público de esa exposición. El alumnado-público ya posee un cierto conocimiento del tema, a través del trabajo realizado para el entregable que han elaborado con anterioridad.

El profesor encarga al *grupo especializado*, generalmente un par de alumnos, la preparación de una exposición oral que realizarán con apoyo de transparencias. Este encargo se realiza con suficiente antelación para que el grupo pueda realizar una preparación concienzuda. El *grupo especializado* dispone de una sesión presencial (50 minutos) para su exposición, dejando tiempo para un breve intercambio de preguntas con toda la clase. Previo a la preparación de la exposición, los alumnos realizan sesiones de trabajo con el profesor para conocer criterios de calidad, metodología, selección de contenidos, etc. En el aula virtual de la asignatura se encuentran publicados los criterios de calidad para la realización de este tipo de trabajos, y se utilizan asimismo los foros del aula virtual para aclarar todas las dudas que puedan surgir en la preparación de la exposición.

Además de la evaluación que realiza el profesor, simultáneamente a la presentación del tema por parte de un grupo de alumnos, otro grupo (de 3 a 4 alumnos, generalmente los que harán en breve un trabajo similar) se encarga de la evaluación de la exposición a la que están asistiendo. Al igual que a la hora de evaluar resúmenes escritos, cuentan con una rúbrica que les sirve de guía de la evaluación. Este informe de evaluación se entrega al finalizar la presentación al grupo

que la ha realizado, sirviendo así de “*feedback*” inmediato.

### 3.3. Laboratorio de prácticas de Compiladores

A lo largo del curso, y a la vez que se van introduciendo las técnicas del diseño de compiladores, los alumnos desarrollan un compilador para Pascal- [1], una versión reducida de Pascal, usando para el desarrollo algún lenguaje de alto nivel (hemos utilizado Pascal, C, C++, Java y Python). Este proyecto de programación supone una de las principales ocupaciones del alumnado de *Compiladores*. Es posiblemente la primera ocasión en que los estudiantes tienen la oportunidad de enfrentarse a un proyecto de desarrollo de software de amplitud y dificultad similares a los que abordarán en su carrera profesional.

Las prácticas de *Compiladores* son una actividad que se desarrolla en pequeño grupo (generalmente 3 estudiantes) al que denominamos *equipo*. Con una semana de antelación se propone a todos los equipos del curso la realización de una práctica que consiste en la implementación de un módulo del compilador de *Pascal*. El trabajo a realizar se describe a través de un documento de especificación que se hace disponible al alumnado mediante el aula virtual. Utilizamos en esta actividad una aproximación de aprendizaje basada en problemas [3], de modo que la práctica que se describe en esta especificación queda un tanto abierta, de modo que posibilite al alumnado cierta flexibilidad a la hora de su resolución. En el aula virtual se dispone de un foro de discusión donde los alumnos discuten entre sí y con los profesores todo tipo de ideas relativas al proyecto que han de realizar.

Las prácticas se evalúan semanalmente en sesiones presenciales a las que asisten 2 ó 3 *equipos* simultáneamente. El profesor suministra a cada equipo tests de validación para comprobar la calidad del trabajo realizado. El *equipo* dispone de cierto tiempo de reacción (en el marco de la sesión presencial) para corregir anomalías en el

funcionamiento del programa que presentan. Alternativamente, el profesor de prácticas puede solicitar una modificación en el trabajo del *equipo* para acreditar su dominio del tema de estudio así como su capacidad como programadores.

### 3.4. Cuestionarios

Al finalizar el estudio de cada tema se utiliza una sesión presencial en cuya primera parte los alumnos responden a un breve cuestionario con formato de diez preguntas de respuesta corta (mini-examen) sobre conceptos fundamentales del tema estudiado. Todas las respuestas han de incluirse en un único folio que cada estudiante marca de modo que sea identificable sin hacer público en el propio folio el nombre del autor del trabajo. Se dedica entre uno y tres minutos por pregunta para que el alumno responda a las cuestiones planteadas.

Al finalizar el cuestionario, en la segunda parte de la sesión, los alumnos se intercambian los cuestionarios para proceder a su co-evaluación, de modo que cada estudiante evalúa de forma anónima dos o tres cuestionarios de sus compañeros, elegidos al azar. El profesor revela posibles respuestas correctas a las cuestiones planteadas, y discute con el alumnado las diferentes cuestiones que se suscitan a la vista de los trabajos que los alumnos tienen en sus manos. Los estudiantes disponen también en este caso de una rúbrica con pautas que sirven de guía para la co-evaluación. En este caso el grueso de la nota se asigna lógicamente a que la respuesta sea correcta, suponiendo hasta el 80% de cada respuesta, pero se deja también margen para valorar aspectos como calidad de la expresión escrita, legibilidad u ortografía.

Esta actividad es la que pudiéramos considerar más parecida a un examen clásico, pero los estudiantes no se examinan de toda la materia sino del tema que acaban de estudiar. La evaluación de cada mini-examen se obtiene combinando la co-evaluación que realizan los estudiantes con la llevada a cabo por el profesor. La finalidad primordial del proceso de co-evaluación vuelve a ser el “*feedback*” a

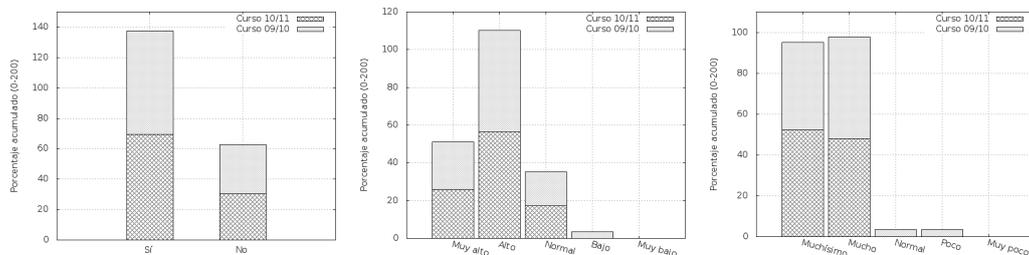


Figura 2: **Izquierda:** Respuestas a la pregunta *¿Prefiere la metodología que se ha usado a una metodología “clásica”?* **Centro:** Respuestas a la pregunta *Indique el grado de aprendizaje que ha conseguido en la asignatura Derecha:* Cuantificación del grado de esfuerzo que le ha conllevado el cursar *Compiladores* con respecto a otras asignaturas

los alumnos, puesto que pocos días más tarde se les entrega el resultado de la evaluación realizada por sus compañeros.

### 3.5. Participación en los foros

Los foros del aula virtual de *Compiladores* son un lugar de encuentro entre profesores y alumnos. Allí los estudiantes pueden proponer y compartir noticias, herramientas, enlaces web a páginas relacionadas, y en general, recursos relacionados con la asignatura. Los foros de prácticas se usan asimismo intensivamente para discutir diferentes posibilidades e interpretaciones a la hora de resolver las prácticas propuestas. Los profesores evalúan y califican cada una de las intervenciones significativas que se producen en los foros. Estas calificaciones se normalizan al conjunto de toda la clase, de modo que se valora no solo la calidad de las intervenciones sino la cantidad de las mismas.

Se pretende con esta actividad que cada estudiante sea capaz de implicarse en la asignatura de manera activa y que transmita conocimientos a sus compañeros, fomentando la comunicación y el interés por la materia de estudio. De esta forma, mediante el uso de los foros, los alumnos y profesores comparten experiencias y puntos de vista que contribuyen a construir una guía de referencia de valor superior al alcanzable de modo particular por cada individuo.

## 4. Análisis de resultados

Sintetizaremos a continuación algunos de los datos que consideramos más significativos de los obtenidos a través de las encuestas al alumnado realizadas en los dos últimos cursos una vez finalizado el cuatrimestre durante el cual se impartió la asignatura. La encuesta del curso 2010-11 constaba de 61 preguntas y participaron en la misma 23 de los 24 alumnos que cursaron la asignatura. La realizada el curso anterior constaba de 56 preguntas y fue respondida por 28 alumnos.

La Figura 2 (izquierda) muestra que de forma mayoritaria el alumnado prefiere la metodología de enseñanza/aprendizaje activo utilizada en *Compiladores* a los métodos tradicionales que mayoritariamente se utilizan en otras asignaturas en la ETSII de la ULL. En el caso concreto del curso 2010-11, 16 alumnos prefieren la metodología usada (69.57%) frente a 7 que prefieren otra (30.43%). Posiblemente un mayor éxito de la metodología pasaría por la aceptación de la carga de trabajo que conlleva. Una de las quejas más frecuentes entre los alumnos y que se refleja en la Figura 2 (derecha) es la de la carga de trabajo que entraña nuestro método de aprendizaje. Entendemos que el alumnado no tiene completamente asumida la parte no presencial y de trabajo autónomo que este tipo de metodologías entraña. En cuanto a la oportunidad del método utilizado, entre un 40% y un 60% de los encuestados lo

considera adecuado mientras que de un 15 a un 25 % lo considera poco o nada adecuado.

Al valorar el grado de aprendizaje alcanzado en el estudio de la asignatura (Figura 2, centro) una clara mayoría de estudiantes lo consideran alto o muy alto. En el curso actual las calificaciones obtenidas por el alumnado fueron: 1 sobresaliente, 14 notables y 9 aprobados. Ninguno de los alumnos matriculados en la asignatura suspende la misma. Se producen al principio del cuatrimestre uno o dos casos de abandono de la asignatura (anulación o cambio de matrícula).

La asistencia a las sesiones presenciales es valorada como positiva o muy positiva por 8 de cada 10 estudiantes encuestados. Este resultado resultaba esperable puesto que con la evaluación continua sólo puede superar la asignatura quienes mantienen un cierto grado de constancia en el estudio de la misma.

La organización del temario y la coordinación de la asignatura es considerada buena o muy buena por un 70 % de los alumnos y entre un 70 y un 85 % de los estudiantes consideran que su participación con preguntas y respuestas, y la retroalimentación de los trabajos ha sido bastante o mucha.

La metodología de enseñanza a través de actividades produce un elevado número de evidencias objetivas acerca de la evolución del alumnado en la asignatura. La Figura 3 (derecha) refleja la evolución temporal de las calificaciones acumuladas de tres alumnos: el que obtuvo la máxima calificación en la asignatura (MAX), el que obtuvo la calificación mínima (MIN) y el alumno cuya nota era la mediana de calificaciones (MED). En los estudiantes con buen desarrollo en la asignatura (es el caso de MAX y MED) hay una evolución positiva que se traslada de forma inmediata a las calificaciones que obtienen en sus actividades.

La Figura 3 (izquierda) muestra la evolución en Prácticas y Cuestionarios de los alumnos designados como MAX y MED, mientras que la Figura 3 (centro) muestra la evolución de las notas medias de las actividades con mayor ponderación.

Como colofón de las encuestas al alumnado se les pide que resuman en pocas palabras lo que piensan de la asignatura y el modo en el que han aprendido. Resumimos a continuación algunos de los comentarios que nos parecen más relevantes: *“Interesante, muy trabajosa pero fácil de aprobar si se tiene constancia.”* *“[...] es una asignatura interesante, pero muy dura.”* *“La asignatura es muy interesante y por la forma de impartirla se aprende mucho más que en otras.”* *“Es una asignatura en la que se aprende mucho, pero creo que debería tener más créditos porque me parece mucho contenido para 4 horas de clase a la semana [...]”* *“Al tratarse de una evaluación continua debes estar con las pilas puestas cada día”*

A la vista de estos resultados podemos decir que los alumnos en general se encuentran satisfechos y consideran la experiencia como positiva e interesante. Cabe destacar la percepción por parte del alumnado de un elevado esfuerzo e intensa dedicación a la hora de cursar *Compiladores*. Este esfuerzo es considerado por los estudiantes superior al resto de asignaturas de la titulación. Es la componente práctica de la asignatura la que se señala como la que más trabajo requiere al estudiar la asignatura. Se trata, como se ha explicado, de la actividad que mayor tiempo consume al margen de las sesiones presenciales.

La totalidad de los encuestados considera que la realización de prácticas de laboratorio en equipos de trabajo es un mecanismo muy apropiado. Un 70 % de los encuestados está satisfecho con su calificación en las prácticas. Como ventajas alegan aquellas con las que el profesorado está de acuerdo: que *“En el mundo laboral se trabaja en equipo”*, se fomenta el *“Análisis de las soluciones en grupo, respaldo en caso necesario, conocer nuevos compañeros”*, *“Se suelen encontrar mejores soluciones a los problemas encontrados, y en general, al compartir opiniones se aprende más”*, *“Ayuda a unificar criterios a la hora de programar y a trabajar en equipo”*, etc.

Una solicitud que el alumnado realiza de forma reiterada es *“Que se impartan más clases magistrales, para entender así mejor*

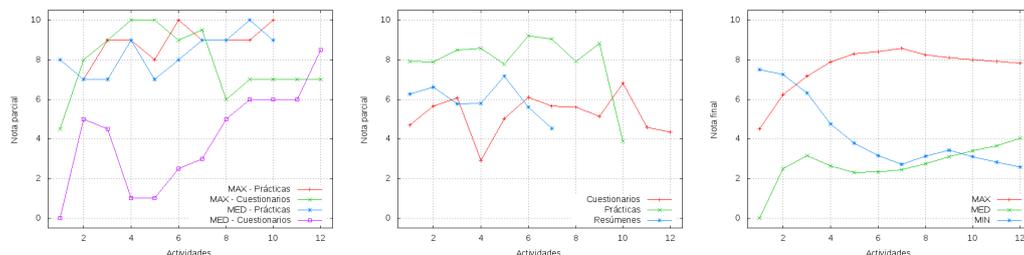


Figura 3: **Izquierda:** Evolución de las calificaciones parciales en Prácticas y Cuestionarios de dos alumnos tipo **Centro:** Evolución de la media de calificaciones en las actividades con mayor ponderación **Derecha:** Evolución de las calificaciones acumuladas en Cuestionarios de dos alumnos tipo

los temas”. Entendemos que esta petición viene motivada por la fuerte tradición de aprendizaje guiado en detrimento de aprendizaje autónomo.

## 5. Conclusiones

La metodología de enseñanza/aprendizaje basada en actividades pedagógicas es muy adecuada para ser utilizada en muchas de las nuevas asignaturas del título de Graduado en Informática que ya ha comenzado a impartirse en la ETSII de la ULL. Como hemos constatado a través de las encuestas al alumnado, el grado de satisfacción por parte de los estudiantes es elevado, así como el nivel de éxito alcanzado. Mediante el uso de esta metodología, la totalidad de los alumnos que cursan al completo la asignatura consiguen aprobar la misma, y no hemos detectado diferencias sustanciales ni en cualificación ni en resultados académicos con respecto a la metodología “clásica” que usábamos hace cinco años.

Las actividades acaban siendo experiencias enriquecedoras y se crea un clima de aprendizaje muy ágil. Es común entre los alumnos una constante necesidad de conocer las actividades que van a realizarse, y el creciente interés por los temas que quedan por tratar.

Con el nuevo modelo que supone el EEES también los estudiantes han de cambiar el rol que históricamente venían asumiendo

respecto a su formación para pasar a tener una actitud mucho más activa y participativa en su propia formación. De acuerdo a nuestra experiencia, este rol no está todavía asumido por parte del alumnado, y es crucial para que se adopte que el número de asignaturas que utiliza metodologías de aprendizaje activo sea relevante, al contrario de lo que ocurre en la actualidad.

## Referencias

- [1] P. B. Hansen. *Brinch Hansen on Pascal Compilers*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1985.
- [2] ACM IEEE. Computing curricula 2001. computer science volume, 2001. <http://www.sigcse.org/cc2001/>.
- [3] L. Prieto. Aprendizaje activo en el aula universitaria: el caso del aprendizaje basado en problemas. *Miscelánea Comillas: Revista de teología y ciencias humanas*, 64(124):173–196, 2006.
- [4] M. Valero-García. Cómo nos ayuda el tour de Francia en el diseño de programas docentes centrados en el aprendizaje. *Novática*, (170):42–56, 2004.
- [5] M. Valero-García and L. Díaz. Evaluación continuada a un coste razonable. In AENUI, editor, *Actas de las IX Jornadas sobre la Enseñanza Universitaria de la Informática JENUI 2003*, pages 183–190, 2003.

## Organizadores



## Patrocinadores



[jenui2011.us.es](http://jenui2011.us.es)