

Enseñanza y Aprendizaje de Ingeniería de Computadores

Revista de Experiencias
Docentes en Ingeniería de
Computadores

Número 1, Marzo 2011
ISSN XXXX-XXXX



Edita: Departamento de
Arquitectura y Tecnología de
Computadores



Colabora: Vicerrectorado para la
Garantía de la Calidad

ENSEÑANZA Y APRENDIZAJE DE INGENIERÍA DE COMPUTADORES
Revista de Experiencias Docentes en Ingeniería de Computadores

TEACHING AND LEARNING COMPUTER ENGINEERING
Journal of Educational Experiences on Computer Engineering

Número 1, Año 2011

**Comité Editorial: Comisión Docente y miembros de los Consejos de Titulación
del Departamento de Arquitectura y Tecnología de Computadores**

Mancia Anguita López
José Luis Bernier Villamor
Antonio Cañas Vargas
Pedro A. Castillo Valdivieso
Miguel Damas Hermoso
Antonio Díaz García
Jesús González Peñalver
Luis Javier Herrera Maldonado
Gonzalo Olivares Ruiz
Julio Ortega Lopera
Héctor Emilio Pomares Cintas
Francisco José Pelayo Valle
Beatriz Prieto Campos
Alberto Prieto Espinosa
Manuel Rodríguez Álvarez
Fernando Rojas Ruiz

ISSN: XXXX-XXXX

Edita: Departamento de Arquitectura y Tecnología de Computadores

Imprime: Copicentro Editorial

Índice

Editorial	1
J. Díaz, C. Morillas, S. Romero, A. Guillén Bolonia ‘for dummies’	5
G. Botella, G. Miñana, S.F. Romero Experiencia interuniversitaria de docencia en un curso clásico de sistemas digitales dentro del EEES	19
F.J. Navarro-Ríos Resultados de una experiencia piloto de implantación del Espacio Europeo de Educación Superior en la Universidad de Granada	29
A.M. Mora, P. García-Sánchez, J.J. Merelo Experiencias en la asignatura Diseño y Evaluación de Configuraciones	35
H. Pomares, I. Rojas, A. Guillén, J. González, O. Valenzuela, J. P. Florido, J. Urquiza, A.B. Cara, L. López-Mansilla, S. Egea-Navarro Desarrollo de un entorno integrado para un computador didáctico elemental, para la asignatura de Fundamentos de Informática del nuevo grado en Ingeniería de Tecnologías de Telecomunicación	43
F. Rojas, H. Pomares, A. Guillén, L.J. Herrera Montaje de los componentes de un servidor para la asignatura del nuevo grado en Ingeniería en Informática: Ingeniería de Servidores	51
J. Fernández, M. Anguita Proyecto Cluster	59
P. García-Sánchez, M.A. López, P.A. Castillo, J. González, M.I. García Arenas Web 2.0: Arquitectura orientada a Servicios en Java	69
G. Botella, D. González, P. Angulo, J. López, F. de A. Martínez, S.F. Romero García Uso de hardware libre en el Máster de Formación del Profesorado de Secundaria (Especialidad Informática)	79
M. Anguita, A. Cañas, F.J. Fernández, J. Ortega, I. Rojas El perfil de Ingeniería de Computadores y las asignaturas de Estructura y Arquitectura de Computadores en el grado de Ingeniería Informática	89
Instrucciones para Autores	99

Editorial

La puesta en marcha en este curso 2010/2011 de los nuevos Planes de Estudios en los que el Departamento de Arquitectura y Tecnología de Computadores de la Universidad de Granada imparte asignaturas (Grados en Telecomunicaciones, Informática, Físicas, y Químicas), y la existencia en alguno de estos planes (Grado en Ingeniería Informática) de perfiles como el de Ingeniería de Computadores, con una participación preponderante del mencionado Departamento, han puesto de manifiesto la necesidad de desarrollar tanto actividades de coordinación de la docencia de las nuevas asignaturas, como de difusión de las perspectivas laborales asociadas a las competencias de esos perfiles.

Se trata de establecer estrategias e iniciativas para conseguir los objetivos de coordinación entre las asignaturas del departamento y garantizar la calidad en la docencia de las mismas. Una de estas actividades ha sido la organización anual de la *Jornada de Coordinación Docente y Empresarial del Departamento de Arquitectura y Tecnología de Computadores (JCDE)*. La primera edición de la JCDE, celebrada en Diciembre de 2010, se ha centrado en dos aspectos importantes y perentorios al iniciarse los planes de estudios de los grados verificados según las directrices emanadas del E.E.E.S. (Planes de Bolonia). Por un lado, hay que difundir, entre todos los profesores del Departamento, las guías de las asignaturas obligatorias en las titulaciones en las que el Departamento participa, contrastar las distintas aproximaciones docentes para extraer conclusiones, alcanzar coherencia y cohesión entre contenidos y objetivos formativos, y asentar las bases sobre las que plantear la docencia de las asignaturas optativas. Por otro lado, se ha buscado la interacción con empresas TIC de nuestro entorno para poner de manifiesto las salidas profesionales de los estudiantes que cursen las especialidades y estudios de postgrado del departamento, a la vez que se difunden, entre las empresas del sector ubicadas en nuestro entorno, las características de la formación que se proporciona a nuestros estudiantes.

Precisamente, cuando en nuestro ámbito universitario se contemplan las relaciones con el tejido empresarial, éstas se enfocan fundamentalmente desde el punto de vista de la transferencia de resultados de nuestra investigación. Se considera que esta puede ser nuestra principal contribución para incrementar la productividad de nuestro país. Sin embargo, ofrecer una docencia de calidad es una de las alternativas más directas que tenemos para aliviar los problemas de baja productividad de nuestra economía. La formación de buenos profesionales puede considerarse la forma de transferencia primordial que la sociedad exige a la Universidad. Pensemos en cuántos estudiantes salen de nuestras aulas cada año.

Por supuesto que hay que desarrollar proyectos de investigación en los que, además de unas cuantas buenas publicaciones, se transfiera tecnología a las empresas, o se consiguen patentes. Pero no se puede renunciar a la semilla constituida por un centenar de ingenieros motivados, rigurosos, y capaces de desarrollar sus proyectos concienciados con la necesidad de alcanzar una sociedad más justa dentro de una economía sostenible. Una forma evidente de transferir conocimiento a la sociedad es, precisamente, la docencia. Un profesor universitario debe ser un buen investigador, pero es esencial que también sea un buen docente.

Dicho esto, es evidente que el estudio y la evolución de la Ingeniería de Computadores se ha producido de la mano de la evolución de los mercados y las aplicaciones que estos demandan, intentando proporcionar dispositivos de cómputo que aprovechen eficientemente la tecnología disponible en cada momento. La ingeniería de computadores no puede entenderse ni enseñarse a espaldas del contexto económico y empresarial. La demanda de ingenieros por parte de las empresas condiciona la demanda de nuestros estudios y la calidad y cantidad de estudiantes que tendremos en nuestras aulas. Como consecuencia de la evolución en el ámbito de las TIC, dentro de las que están incluidas las áreas de nuestro Departamento (Arquitectura y la Tecnología de los Computadores e Ingeniería de Sistemas y Automática), existen varias circunstancias que confieren especial interés a la Ingeniería de Computadores, y determinan un futuro prometedor en cuanto a las salidas profesionales de los graduados en Ingeniería Informática con esta especialidad. Así, podemos citar la utilización generalizada de microprocesadores multinúcleo, la disponibilidad de herramientas que acercan las posibilidades de desarrollo e integración de sistemas embebidos en infinidad de dispositivos y aparatos de consumo, y el paradigma del denominado “cloud computing”, en el que juega un papel importante el acceso eficiente a servidores a través de Internet.

Todas estas circunstancias abren oportunidades de negocio, donde estudiantes emprendedores podrán desarrollar sus competencias profesionales y sus capacidades creativas. Oportunidades que van desde las relacionadas con el desarrollo de aplicaciones que aprovechen los computadores de bajo precio, los sistemas embebidos y los computadores de altas prestaciones, hasta las técnicas de consolidación de servidores basadas en la virtualización. De hecho, en las asignaturas básicas y de rama de Ingeniería de Computadores del grado en Ingeniería Informática se enseña a los estudiantes a programar eficientemente y a diseñar, montar y administrar computadores de gama baja. En la especialidad de Ingeniería de Computadores hay dos materias obligatorias, Sistemas de Cómputo para Aplicaciones Específicas, que enseña a los estudiantes a programar, diseñar, montar y administrar productos embebidos, y Sistemas de Cómputo de Altas Prestaciones, que enseña a programar, diseñar, montar y administrar computadores de gama media y alta. En la especialidad de Tecnologías de la Información se imparte una materia obligatoria en la que el estudiante aprende a diseñar y usar servidores web de altas prestaciones y sistemas basados en virtualización. En todos estos ámbitos, la Ingeniería de Computadores tiene bastante que decir, y debemos promover de activamente que los estudiantes demanden esta especialidad, a la que tenemos que dotar de un marcado carácter innovador.

Por todo lo comentado más arriba, cobra especial sentido promover un medio a través del que puedan recogerse iniciativas, propuestas o experiencias docentes en Ingeniería de Computadores, además de opiniones que pongan de manifiesto el punto de vista de

los estudiantes y las consecuencias que la formación recibida ha tenido en su vida profesional. Al anunciar la celebración de la JCDE se solicitaron colaboraciones que han permitido componer este primer número de esta revista, *Enseñanza y Aprendizaje en Ingeniería de Computadores*, cuya edición inicia el Departamento de Arquitectura y Tecnología de Computadores. Por lo pronto, en este número se recogen diez artículos de profesores, y de antiguos alumnos del departamento. Abarcan aspectos relacionados con la implantación del E.E.E.S., las experiencias docentes y la propuesta de herramientas para distintas asignaturas de grado y posgrado, y las características del perfil de Ingeniería de Computadores. Esperamos que la difusión electrónica y la posibilidad de enviar artículos sin limitaciones temporales, permita que, año tras año, la revista pueda contribuir con ideas interesantes a la mejora de la calidad de la docencia.

Terminamos esta editorial/presentación de esta revista expresando nuestro agradecimiento a todos los participantes en la JDCE de 2010, especialmente a María del Carmen López, directora del Secretariado de Formación y Apoyo a la Calidad, y a Buenaventura Clares, director de la E.T.S.I.I.T. por su participación en el acto de inauguración de la JDCE; a los ponentes en las mesas redondas, los profesores Alberto Prieto, Francisco José Pelayo, Ignacio Rojas, Mancia Anguita, y Héctor Pomares; y a los representantes de empresas Rosa Castellanos (Catón Sistemas Alternativos), Javier Fernández Lorca (Ingeniería y Control Remoto), Eduardo Ros (Seven Solutions), Pedro Romo (Telefónica I+D), y José Fernando López (Trevenque). A todos ellos, muchas gracias por su compromiso y su trabajo. También agradecemos al Vicerrectorado de Garantía de la Calidad de la Universidad de Granada la financiación recibida.

Comité Editorial

Bolonia ‘for dummies’

Javier Díaz¹, Christian Morillas¹, Samuel Romero¹, Alberto Guillén¹

¹ Dpto. Arquitectura y Tecnología de Granada. ETSIT
Granada, España
{jdiaz, cmorillas, sromero, aguillen}@atc.ugr.es

Resumen. El objeto de este trabajo es describir de forma sencilla y clara cómo impartir una asignatura nueva o transformar una asignatura existente, y la metodología docente para su impartición en el marco del Espacio Europeo de Educación Superior (EEES). Hacemos especial hincapié en aquellos aspectos afines a las materias impartidas tradicionalmente en Ingenierías como Informática, Telecomunicaciones o Electrónica, titulaciones relevantes en el área de Arquitectura y Tecnología de Computadores. Enunciamos los principios básicos y el enfoque que motiva y dirige la reforma educativa de Bolonia. Consecuencia de ello, describimos métodos sencillos para transformarnos en unos “nuevos docentes,” alineados con la reforma educativa, tratando a la vez de minimizar el impacto que pueda tener en el profesor y de incluir nuevos enfoques de realización de nuestras tareas docentes.

Palabras Clave: Reforma educativa, EEES, Bolonia.

1 Introducción

“los cursos no valen..., si quiero aprender algo me leo un libro... y hago experimentos... ...deberíamos olvidarnos de la escuela y la universidad,... los niños deberían empezar el doctorado a los 5 años y que dure 20 años...”

Sydney Brenner (Premio Nobel de Medicina 2002). Redes (RTVE, Enero 2005)

“El monólogo frontal en que consiste la clase no sólo refuerza la pasividad del oyente, sino, más grave, suprime el espacio para cuestionar y cuestionarse, que es, justamente, lo que caracteriza a la docencia que entronque con la investigación. La labor del profesor no radica en transmitir los conocimientos adquiridos, en el mejor de los casos en la investigación más reciente, sino en enseñar a preguntar, orientando el trabajo y promoviendo el desarrollo intelectual y científico de los alumnos, lo que únicamente cabe en un diálogo personal”

Ignacio Sotelo, catedrático excedente de Sociología, (El País, 2 Febrero 2005)

Este tipo de declaraciones ilustran cuál es el espíritu del que se nutre la Declaración de Bolonia de 1999, suscrita por 30 estados europeos no sólo de la UE sino países del Espacio Europeo de libre comercio y países del Este y Centro de Europa.

La reforma educativa asociada a la declaración de Bolonia o EEES es hoy en día una realidad (para pesadilla de algunos y deleite de otros). Por ejemplo, en el marco de la ETSIIT de la Universidad de Granada este año se está impartiendo el primero de los cursos de grado de las titulaciones ofertadas en la escuela (Ingeniería Informática y Telecomunicaciones). Pese a que “ya está aquí”, muchos profesores no conocen las nuevas metodologías docentes necesarias para afrontar este proceso y no está claro cómo afectarán a su metodología docente habitual. Dado que a nivel institucional no se está dando formación para el profesorado, éste debe ser responsable e intentar adaptarse lo mejor posible haciendo un aprendizaje cooperando y coordinándose con otros colegas. En esa línea, este documento nace de la participación de algunos de los autores en el Curso de Iniciación a la Docencia impartido en la Universidad de Granada en 2010 [1]. Con él pretendemos, de forma muy básica, ofrecer un resumen de diferentes técnicas y protocolos de actuación convenientes para afrontar esa adaptación del profesorado y de sus asignaturas al EEES de modo que pueda servir de apoyo a quién lo estime oportuno.

Como veremos a lo largo de esta contribución, no se trata de cambiar todo, sino de concretar y cuantificar el esfuerzo del alumno en nuestra materia para conseguir los objetivos que tenemos, y diseñar métodos de aprendizaje que sustituyan las clases magistrales (de modo que se adquieran capacidades y no sólo conocimiento teórico).

Múltiples aspectos y enfoques son vitales para comprender el reto con el que nos encontramos. Entre ellos podemos destacar un enfoque del aprendizaje basado en el alumno y en la adquisición de competencias profesionales (donde dejamos sin respuesta la pregunta de si un profesor de universidad sabe realmente de ello), incorporación de una pluralidad de recursos didácticos y metodológicos y la incorporación clave del proceso de planificación detallada de los contenidos (y temporización) de las distintas materias.

Notemos que los autores sólo pretenden mostrar su experiencia, parcial e incompleta, pero que esperamos pueda ayudar a otros compañeros perdidos como nosotros en esta vorágine de nuevos conceptos, metodologías y nuevos marcos legales y educativos.

Este trabajo está estructurado como sigue: en la Sección 2 damos una revisión de los aspectos principales que rigen el proceso EEES y que servirán como fundamentación del porqué y cómo cambiar nuestra metodología docente. En la sección 3 veremos el proceso de búsqueda de información marco para impartir una asignatura en los nuevos planes de grado. La sección 4 se encargará de mostrar algunos ejemplos de los recursos metodológicos que debemos utilizar para impartir dichas asignaturas. Finalmente, la sección 5 se dedicará a la discusión y conclusiones finales de este trabajo.

2 El espíritu de Bolonia ¿Qué quiere Bolonia de nosotros?

Siguiendo la comunicación [9], debemos entender primero qué es (y qué no es) la reforma educativa promovida por la declaración de Bolonia.

¿QUÉ ES (o debería ser) EL ESPACIO EUROPEO DE EDUCACIÓN SUPERIOR?

- A. Una oportunidad de mejora de la calidad de la docencia universitaria
- B. Un cambio de paradigma: de un sistema basado en la enseñanza del profesor a un sistema basado en el aprendizaje del estudiante
- C. Transparencia: contrato profesor-alumno (crédito ECTS como método de incluir el trabajo del alumno fuera del aula)
- D. Una apuesta por una formación integral: contenidos científicos y capacidades (estudiantes como profesionales y ciudadanos)

¿QUÉ NO ES (o no debería ser) EL EEES?

- A. Un cambio administrativo en la forma de medir la dedicación de los alumnos (¿y los profesores ¿quién y cómo nos evalúan?)
- B. La obligación de preparar una tabla de equivalencias entre horas de clase (crédito tradicional) y horas de trabajo del alumno (crédito ECTS)
- C. Crear nuevas víctimas del sistema:
 - a. Profesor: convertir una asignatura en una “gymkana de capacidades”: “ahora tengo que enseñarles ciencias y, además, inglés, informática, a trabajar en grupo, habilidades de comunicación,…”
 - b. Alumno: la “oportunidad” para recibir menos clases y que los alumnos lo hagan todo por su cuenta

Como se desprende del análisis anterior, el proceso de convergencia europea pretender modificar significativamente (si no revolucionar) los objetivos y métodos docentes denominados clásicos. Por supuesto que muchos profesores ya trabajaban según esta metodología pero ahora se pretende desarrollarla de forma institucional y a nivel de toda Europa. Los principales aspectos de la reforma son:

1. Formación en base a competencias. Esto implica que se pretende dar un enfoque más aplicado al conocimiento. Un ejemplo de esto podría ser el siguiente:

En el marco de un examen de Ingeniero de Caminos, Canales y Puertos, asignatura sobre materiales, solicitarle que realice el diseño de una estructura de hormigón de un puente concreto frente a una pregunta que describiera las propiedades del hormigón es un ejemplo de la orientación a competencias.

Este caso muestra que se le daría mayor importancia al saber hacer (diseñar la estructura usando hormigón del tipo adecuado y demás parámetros de diseño) que al saber (conocer las propiedades mecánicas y físico-químicas del hormigón). Información sobre las competencias podemos encontrarlas en:

- Genéricas (informe Tuning [5])
- Específicas (Libro Blanco [11])
- Titulación (documento VERIFICA [8])

A partir de esta idea, cada asignatura es formulada basada en competencias (qué sabrá hacer el estudiante tras finalizar la asignatura). Es importante destacar que cada

competencia que se incluya deberá ser evaluada, luego habrá que ser cuidadosos para desarrollar un sistema de evaluación que permita asegurar que toda competencia ha sido cubierta y verificada.

Además, notemos que una misma competencia puede (y en muchos casos debe) estar cubierta por varias asignaturas y que, en teoría, podría ocurrir que una asignatura básica y muy necesaria para una titulación no tuviera competencia alguna asociada a la misma. Si bien este último caso es posible, la implantación de los planes de estudio ha forzado que todas las asignaturas tuvieran competencias (algunas veces se ha intentado que una asignatura/materia tenga más competencias con la idea que por ello sería más importante o que sin competencias pudiera desaparecer). Esperemos que en sucesivas modificaciones de los planes este problema desaparezca y las competencias se adecúen realmente a los contenidos que deben impartir las distintas materias.

Queremos destacar que el enfoque inicial basado en competencias profesionales (para saber hacer algo) ha sido modificado para incluir otras capacidades del estudiante de índole intelectual (capacidades de síntesis, análisis, expresión escrita) o de carácter social (actitudes frente a compañeros, capacidades de trabajo en grupo, etc.). Hemos pasado de este enfoque a uno en el que todo se denomina competencias (competencias instrumentales, competencias interpersonales, competencias sistémicas, etc.), independientemente de que tengan relación con el ejercicio de su carrera profesional o no lo que para muchos expertos carece de sentido. Esta discusión queda fuera del alcance de este documento y es más propia de profesionales del área de la didáctica y pedagogía pero es el origen de la “extraña” terminología usada en muchos documentos.

Finalmente, comentar que estas competencias (independientemente del tipo que sean), han sido divididas en los grados en dos apartados, competencias específicas del título y competencias transversales (estas últimas más asociadas a capacidades y sin referente como competencias directas profesionales).

En fin, mucha información en muy poco espacio. Para el lector que quiera ampliar le recomendamos la lectura de [3, 12, 13].

2. Enfoque del proceso de aprendizaje basado en el estudiante. Para ello se deben usar nuevas metodologías docentes: desde herramientas para el aprendizaje autónomo, aprendizaje por indagación (guidado por el profesor pero realizado por el estudiante), colaborativo (en grupos) o la utilización de la clase magistral (pero con la inclusión de elementos que la hagan dinámica, evitando el mero “profesor habla, alumno escucha”. Todos estos elementos requieren el conocimiento y la utilización de múltiples recursos docentes, así como el desarrollo de métodos de evaluación capaces no sólo de determinar el nivel de competencia adquirido por el alumno sino, además, de ayudar en el mismo proceso de aprendizaje. La tutoría se reinventa como método para ayudar a los estudiantes en el proceso de aprendizaje.

3. Nuevas maneras de planificar: uso de las guías didácticas. Estos nuevos objetivos que se le imponen al profesorado sólo son posibles mediante una planificación minuciosa de las clases, identificando qué tipo de contenido y actividades será conveniente abordar en cada una de ellas. Pese a la aparente rigidez de planificar, no debemos sentirlo de manera negativa ya que el objetivo es que sirva de guía y, en base a esa pluralidad de recursos docentes que deberemos poseer,

deberíamos ser capaces de manejar flexiblemente esa planificación. Para más información sobre el proceso de planificación leer [7]. Todo ello se complementa y plasma en el uso de las guías didácticas. Estos documentos irán orientados al alumno (no al profesor como ocurre en el proyecto docente) y deberán ser capaces de guiar a alumno en el mar de actividades y recursos que debe utilizar para conseguir las competencias de la materia en curso. La Universidad de Granada realiza cursos específicos sobre elaboración de Guías didácticas y tiene muchos de los resultados de estos curso disponibles online. Como referencia ver la página web del Vicerrectorado de calidad [2] y como ejemplos consultar [4].

4. Finalmente, destacar que uno de los objetivos del EEES es la **movilidad de estudiantes**, ya sea horizontal (por ejemplo mediante el programa Erasmus) o vertical (mediante la convalidación de titulaciones). Ello se consigue mediante la utilización de “formatos de títulos comunes” (dos ciclos, uso de créditos ECTS, promoción de la movilidad, cooperaciones entre instituciones, suplemento europeo al título, etc.). El análisis de la consecución de este objetivo queda fuera del alcance de este trabajo aunque, como primera aproximación parece que se ha avanzado en los formatos de las titulaciones (grado + máster) y en consideraciones metodológicas de la docencia universitaria, quedando pendiente el hecho que las titulaciones tengan contenidos similares (que es un elemento clave para facilitar la movilidad de los estudiantes).

El resultado de todo este proceso es que los profesores universitarios deberán convertirse en verdaderos expertos en todo tipo de técnicas de comunicación, manejo de herramientas TIC, técnicas psicopedagógicas (dinamización de grupos, motivación del estudiante), etc. además de conocer adecuadamente su materia y aspectos de la misma relativos a las salidas profesionales que tuviera asociada. Pasar de una metodología básicamente magistral a este nuevo enfoque es, sin formación ni ayuda externa, una tarea ingente para el profesorado. Esperemos que con estas líneas podamos dar una pequeña “guía de bolsillo” que nos permita empezar con el proceso. Como lecturas complementarias sugerimos [14, 6].

3 ¿Cómo desarrollo los contenidos de mi asignatura en el grado?

Ya sea porque nos ha tocado impartir una nueva asignatura o bien porque tenemos que “adaptar” una parecida que dábamos en el plan antiguo, debemos prepararnos para esta nueva forma de impartir las clases. Si miramos el marco legislativo, hemos dejado atrás la Ley de la Reforma Universitaria (LRU) y por tanto los contenidos de las asignaturas que venían como descriptores en el BOE carecen actualmente de validez en los nuevos planes. El marco legislativo actual viene dado por la Ley Orgánica 6/2001 de 21 de diciembre, de Universidades (LOU), modificada en la Ley Orgánica 4/2007 de 12 de abril. A nivel local deberemos atender a la Ley 15/2003, de 22 de diciembre, Andaluza de Universidades (LAU) y a la legislación interna de la Universidad de Granada. Información relativa al marco legal LOU/LAU puede ser encontrada en la Tabla 1.

Contenido	LOU	LAU
Servicio a la sociedad	Art. 1, puntos b) y c)	Art. 3, puntos c) y h)
Calidad	Art. 31, puntos a) y c)	Art. 58 Art. 76, puntos 1,2 y 3
Docencia	Art. 33, puntos 1, 2 y 3	Art. 1, punto f) Art. 53, punto 1

Tabla 1: Marco legal para el desarrollo de la labor docente en Andalucía (tabla adaptada de [1]).

No obstante, no nos centraremos en los aspectos legales (¡menos mal!), dejando esta tarea para realización de los interesados (si los hubiere). Desde el punto de vista más aplicado, cuando se nos asigna una nueva asignatura nuestra labor sería obtener la memoria del título de grado verificada positivamente por la ANECA/AGAE. Recordar que ahora los títulos impartidos en los nuevos grados han sido previamente aprobados por las agencias de acreditación mediante el programa VERIFICA [10]. En esta memoria, buscaremos la asignatura que vamos a impartir y tendremos la información básica sobre contenidos, planificación, métodos de evaluación, etc. Los títulos verificados en la Universidad de Granada están disponibles para su consulta en [8]. Notemos que la información ahora disponible sobre qué impartir es ahora mucho más detallado que lo que había en los descriptores del BOE en la LRU. Como consecuencia, estamos mejor guiados y a la vez mucho más coartados en nuestra libertad de cátedra.

Si por algún motivo los títulos de grado hubieran sido desarrollados con mucha rapidez y por personal no correctamente formado en el proceso de adaptación al EEES, la información relativa a los contenidos y metodología de la materia a impartir podrían nos ser los más adecuados. Bueno, hay que vivir con ello y saber superarlo. Notemos que la ANECA ha establecido mecanismos de control (programa Monitor) para realizar un seguimiento del título oficial para comprobar su correcta implantación y resultados. Asociado al mismo cada titulación tendrá responsables encargados de supervisar estos factores (esperemos que para bien) y, con suerte, podrán realizarse modificaciones de los contenidos de la memorias de grado con objeto de mejorarlas.

A partir de aquí, el proceso/tarea a realizar sería la de concretar los contenidos, metodologías docentes y de evaluación, actividades y planificación de la asignatura a impartir.

Por ejemplo, a nivel de contenidos tenemos que pensar en:

- Los descriptores oficiales;
- La acción profesional;
- Las exigencias del contexto: a) institucional (estructura organizativa en la que se inscribe la propuesta); b) socio-profesional (finalidad formativa); c)

curricular (interdisciplinaridad); y d) psicosocial (características de los destinatarios);

- Los estudios de competencias;
- El ámbito estatal y autonómico;
- El propio proceso de desarrollo profesional
- Las respuestas a preguntas de contenido como: ¿Cuántos temas pongo?, ¿Qué puede aportar?, ¿Qué temas son los fundamentales?
- Si vendrán profesionales a explicar el tema.
- Las conexiones con otras asignaturas.
- En qué tipo aulas trabajaremos y de qué medios dispondremos.
- Qué método usaremos para cada tema, por ejemplo ¿en cuáles talleres, cuáles magistral, cuáles indagación?
- El hilo conductor de los temas, ¿a qué interés responde?

Y a nivel de evaluación tenemos:

- Los criterios e instrumentos de evaluación: Evaluar por carpetas (portfolio), exámenes de preguntas cortas, largas o tipo ensayo, pruebas objetivas, trabajos opcionales, por contratos, mapa conceptual, defensa oral o escrita de un libro...
- Las alternativas a los que no aprueban: lecturas, tutoría
- El plan B para alumnos Erasmus, Séneca y de incompatibilidades horarias
- Los objetos de la evaluación que son: aprendizaje, programa de la asignatura, acción práctica del profesor

Como resumen, notemos que todos estos aspectos implican no sólo un profundo conocimiento técnico de la materia a impartir, sino además amplios conocimientos y recursos en el ámbito pedagógico y docente así como de los recursos que ofrece la institución universitaria en que trabajemos. De nuevo muchos aspectos fuera totalmente del alcance de este trabajo pero que esperamos sean abordados por los profesores de la Universidad en pro de una mejora de la docencia en las aulas.

4 Si no doy la lección magistral, ¿cómo lo hago?

El hecho de que el proceso de enseñanza-aprendizaje esté centrado en el estudiante no quiere decir que la lección magistral no sea un método docente válido (¡no nos asustemos!). La reforma educativa promueve que el docente aplique distintas metodologías docentes, eligiendo la más apropiada para los distintos contenidos y actividades de la asignatura, sin embargo, la parte magistral (convenientemente adaptada), sigue siendo un recurso válido.

Conviene hacer hincapié en algunos aspectos clave para que la reforma pueda ser todo un éxito. Tanto el profesorado como el alumnado deben estar concienciados en los cambios que trae la reforma, y el nuevo papel que cada uno debe desempeñar. El profesor debe estar dispuesto a cambiar su esquema de trabajo y convertirse en orientador o facilitador (dejar de ser un ponente), pues ahora el alumno es el protagonista de su propio aprendizaje, por lo que también habría que hacer un

esfuerzo en comunicarles a los alumnos cuál va a ser su papel, antes de comenzar los estudios universitarios.

Entre los cambios que se esperan del profesorado podemos destacar la variedad metodológica, para lo que pasamos a comentar algunas metodologías o estrategias alternativas a la *lección magistral* que tradicionalmente (con sus ventajas e inconvenientes) se ha venido impartiendo en el ámbito universitario, y que está principalmente orientada a la memorización, más que a la comprensión (ver Figura 1) y consultase [1] para más detalles. No obstante, la lección magistral permite la introducción de elementos dinamizadores durante su desarrollo, desde preguntas a debatir con los estudiantes, inclusión de pequeños ejercicios para los alumnos, a sopas de letras y jeroglíficos que rompen con la monotonía del discurso.

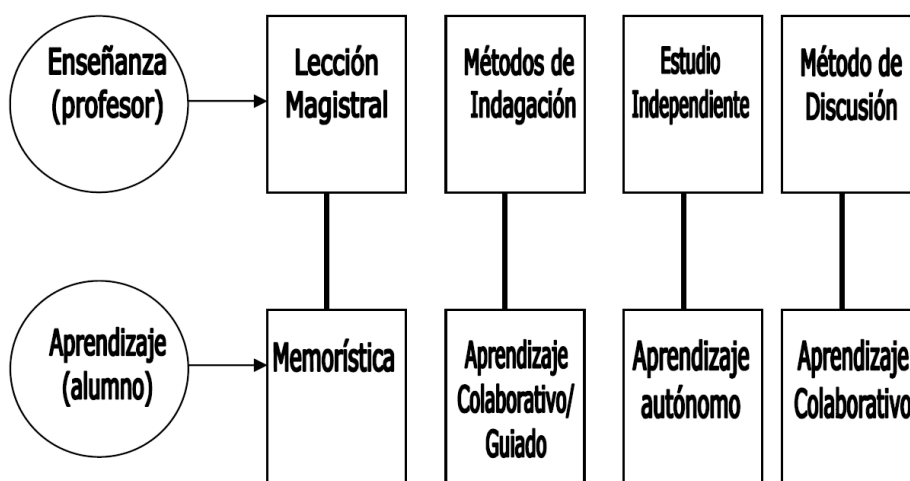


Figura 1: Relación entre los métodos de enseñanza y el tipo de aprendizaje por parte del alumno (figura adaptada de [1]).

En el caso de los *métodos de indagación* el alumno elabora sus conocimientos induciendo o deduciendo reglas a partir de datos o ejemplos que el profesor presenta. Se conoce también como aprendizaje basado en problemas, y el papel del profesor es el de guía en el proceso de aprendizaje, cerciorándose de su consecución. Una actividad dinamizadora para esta metodología es la elaboración de mapas conceptuales por parte de los alumnos, que permite constatar la correcta identificación y relación entre los conceptos clave. Éste es, principalmente, el paradigma que promueve el EEES.

Otra de las novedades que introduce la reforma es la importancia que se da al aprendizaje autónomo por parte del estudiante, mediante el *estudio independiente*. El estudiante es responsable de planificar, desarrollar y evaluar su aprendizaje, permitiéndole aprender a un ritmo personal y adecuado a sus condiciones e intereses. Para el buen desarrollo se hace necesario disponer de una buena biblioteca/hemeroteca, acceso a bases de datos e internet y lugares adecuados para el desarrollo del trabajo personal. Además, el alumnado debe estar preparado en técnicas de trabajo individual y el profesorado debe dedicar tiempo en clases y tutorías a orientar y supervisar este tipo de tareas individuales.

También podemos usar este enfoque basado en aprendizaje autónomo para eliminar contenidos de la lección magistral. Cuando el temario lo permita, podemos indicar en la guía didáctica del alumno que un tema concreto ha de ser estudiado de forma autónoma. Ello, junto con una guía de trabajo autónomo al respecto que, incrementalmente, guíe al alumno en el proceso de aprendizaje y una bibliografía detallada, deberá permitir entender a un alumno un tema por sí mismo. Posteriormente en clase podemos debatir y aclarar las dudas del mismo. Esto tiene como ventaja al profesor que nos permite “no hacer las transparencias” de ese tema. Como contrapartida el diseño de la guía de trabajo autónomo de este tema requiere tanto o más esfuerzo como el de hacer las transparencias. En lo referente al alumno corremos el riesgo que directamente no realice la actividad pero, ¿Cuántas veces un alumno viene a clase y no atiende? Está sólo de cuerpo presente. Este método permite dar dinamismo a las clases y que cada alumno aprenda a su ritmo. Por supuesto tiene inconvenientes y probablemente no es válido para algunos aspectos del temario de nuestras asignaturas pero de nuevo es un recurso didáctico más que podemos explorar.

Un elemento para promover la reflexión, autoevaluación y el análisis crítico por parte del alumno es el portafolio. Podemos definir un portafolio como una colección de los logros individuales del alumno junto a la evaluación del proceso, las estrategias aplicadas, y el análisis sobre las experiencias de aprendizaje, por lo que es mucho más que una colección de tareas realizadas. Notemos que esta carpeta de actividades tiene un doble objetivo. Por una parte de ella podemos extraer las actividades que serán objeto de evaluación del estudiante (no necesariamente todas las actividades promovidas deben ser usadas para la evaluación del mismo). De otra parte puede ser un instrumento más para mejorar el aprendizaje del alumno.

Un gran problema que se encuentra el profesor es el relativo a que algunas tareas no son realizables debido al alto número de estudiantes matriculados (para un profesor que trabaja no más del número de horas exigido en su contrato). Las estrategias de trabajo en grupo pueden ser una herramienta para lidiar con ello. Puesto que en los títulos de grado se han definido actividades para grupo grande (toda la clase) y para grupo pequeño (típicamente lo que eran grupos de prácticas en titulaciones TIC), el tipo de actividades a realizar deberá ajustarse al tipo de grupo con el que contemos en cada momento.

Estas estrategias dan un papel más relevante al alumnado y podemos englobarlas bajo el *método de discusión*. Desarrollan el pensamiento crítico, las habilidades sociales e impulsa la independencia de juicio. En contra, requiere una buena planificación para evitar conflictos de grupo, el trabajo es más lento y hay que cuidar el tamaño del grupo de trabajo para que sea operativo (por ejemplo, elegir un número impar de miembros para evitar empates en la toma de decisiones que paralicen el trabajo). Dependiendo del tamaño de los grupos que realicemos y el tiempo que se le pueda/quiera dedicar a este tipo de actividades, podemos encontrar una amplia variedad de estrategias, entre las que destacamos el debate, el ‘role playing’, el foro, la mesa redonda, el simposio, el panel o la conferencia, especialmente destinadas a grandes grupos de participantes. Para grupos medianos podemos trabajar la lluvia de ideas o el grupo T. En grupos pequeños podemos aplicar técnicas más específicas como Phillips 6/6 (grupos de 6 personas discutiendo un problema durante 6 minutos), el cuchicheo (discusión de un problema en voz baja por parejas), la entrevista o el

mantel de papel (papel de grandes dimensiones en el que cada alumno o sub-grupo escribe sus impresiones o decisiones sobre un tema concreto).

La aplicación de unos métodos u otros exige, en diferentes momentos del proceso, la organización de los alumnos en grupos de distinto tamaño. La lección magistral está dirigida al gran grupo, todos los alumnos de una clase; los métodos de discusión exigen que los grupos de trabajo sean más pequeños; en el caso del aprendizaje autónomo el alumno debe trabajar solo mientras que los métodos de indagación permiten diferentes configuraciones. Otra exigencia muy importante es la planificación, que debe estar presente en todos los aspectos que intervienen en el proceso de enseñanza-aprendizaje, desde los contenidos, metodologías y estrategias didácticas y de evaluación, hasta las tutorías [7].

Toda esta marabunta de nuevas técnicas y métodos nos llevan a otra idea, ¿cómo evaluar de forma eficaz todo el trabajo que realizan los estudiantes sin que suponga un conflicto con el resto de encargos que tiene el profesorado? En el EEES se apuesta firmemente por la evaluación continua (el 60% de la calificación debería estar fijada antes del examen), lo que puede llevarnos a una dedicación prácticamente por completo a la corrección de trabajos, supervisión de tareas, etc. Algunas ideas que pueden facilitarnos el trabajo serían la de (a) incorporar la evaluación entre iguales, en la que los alumnos evalúan a sus propios compañeros de forma responsable mediante técnicas como la rúbrica, la lista de control, o la escala de valoración, (b) definir tareas de autoevaluación o incluso co-evaluación, en la que la evaluación se establece como un acuerdo al que se llega entre profesor y alumno en una entrevista (c) usa de las herramientas TIC para automatizar el proceso de evaluación continua (ver por ejemplo [16]).

Notemos que un problema típico que se indica con el tema de la evaluación entre iguales es la de objetividad o capacidad de un alumno para evaluar al compañero. Si bien es necesario realizar una planificación adecuada por parte del profesor y dejar muy claro a los alumnos los criterios de evaluación (posiblemente consensuados con ellos), la experiencia dice que generalmente es un método que funciona muy bien. Además siempre podemos utilizar técnicas disuasorias (si algún alumno protesta re-evaluamos nosotros y, si la nota puesta por el compañero no es acorde a los objetivos pactados, podemos puntuar negativamente al estudiante que ha hecho una evaluación incorrecta). Como conclusión operativa, o tenemos un número pequeño de alumnos (como en optativas de los últimos cursos) y corregimos todo nosotros, o podemos optar por corregir un número ingente de trabajos/actividades, o seguimos con clases Magistrales. Como mejor alternativa, os animamos a probar estas técnicas de evaluación entre iguales y a sorprenderos de su utilidad.

Otro elemento que interviene en el proceso de enseñanza-aprendizaje que, sin ser nuevo, adquiere mucha importancia en el EEES es la tutoría. Podemos decir que la tutoría se reinventa, deja de ser un tiempo para la mera consulta de dudas por parte del estudiante a ser un nuevo espacio temporal para el que podemos planificar actividades de seguimiento, de orientación o de cualquier tipo, tanto individuales como de grupo. Todo ello ha dado lugar incluso a proyectos específicos para mejora de las tutorías, los conocidos como planes de acción tutorial o PAT. Para más información ver por ejemplo el material disponible en [15].

Finalmente toda esta metodología debe apoyarse en el uso de las nuevas herramientas docente que, principalmente de la mano de las TIC, están disponibles.

blogs, plataformas docentes virtuales, herramientas de consulta, material para presentaciones, etc. Todas estas herramientas deben conocerse y manejarse con soltura para, de esta forma, mejorar la comunicación entre profesor-alumno, hacer los contenidos más atractivos y, en resumen, facilitar el proceso de aprendizaje del alumno.

5 Resumen y conclusiones

Esperamos que en este breve espacio hayamos podido describir, de forma básica y clara qué requiere de nosotros el proceso de adaptación al EEES y, con un poco de suerte, motivar al lector a profundizar en los temas aquí descritos. Como efecto adicional, esperamos que aquellos compañeros que no tenían claro cómo adaptar sus materias en este nuevo marco puedan, al menos, cumplir con la normativa vigente y que sus asignaturas sean “un poquito más a la Bolonia” y, aunque no cambien drásticamente su metodología docente, sí puedan dar pasos en la dirección que las instituciones y la política nos demandan.

Dicho todo esto, permitidnos un espacio para la reflexión personal. Surge la duda de si este nuevo enfoque promulgado por el EEES es el medio de mejorar la calidad de la docencia universitaria. Según se desprende de algunos trabajos como [6] y [13], muchos de los elementos son positivos (aprendizaje basado en el estudiante, aumentar nuestras herramientas metodológicas y docentes, etc...). No obstante, surgen otras cuestiones más asociadas al día a día, ¿proporciona el marco institucional los medios y la formación necesaria para aplicar estos cambios? Como es bien sabido, estos nuevos enfoques requieren un trabajo muy intensivo del profesor y son difíciles de llevar a cabo con grupos grandes. Además, ¿qué tanto por ciento de los profesores universitarios ha recibido formación para este cambio docente? La realidad dice que los grupos actuales son grandes y que la mayoría de los profesores (y alumnos) no tienen la formación adecuada para afrontar este cambio de enfoque docente.

En nuestra opinión, la filosofía actual basada en hacer cambios en la que nos hallamos sumidos, que muchas veces consiste simplemente en llamar con nombre distinto a las cosas y cuyo principal requisito es que sea a ‘coste 0’, tiene altísimas probabilidades de fracaso. Aunque de esta forma las estadísticas arrojen resultados positivos, el problema de base que consiste en tener universitarios que reciban la mejor formación posible seguirá sin solución. Si la docencia, y la universitaria en particular, es tan importante, ¿por qué no se apuesta firmemente por ella? ¿Por qué no se favorece la promoción del profesor que demuestra un alto grado de competencia en aspectos docentes, mientras que la investigación es el elemento principal que determina la promoción? Todo esto sólo puede salvarse (al menos parcialmente) apelando a la profesionalidad del cuerpo de los docentes universitarios que, de forma gratuita y en sus “ratos libres”, deberán realizar un cambio mucho más profundo que el mero cambio terminológico.

Para concluir, un último aspecto a considerar es que el uso de las fichas Verifica y la obligación de usar esta metodología del EEES, no necesariamente es el camino hacia una docencia de calidad. Como indica [13], los mejores profesores son los que, de forma flexible y con gran nivel de conocimiento de la materia que imparten, son

capaces de adaptarse a los alumnos e innovar. Tal y como vemos la reforma, parece que todo el profesorado debe andar prácticamente por los mismo itinerarios, seguir las mismas metodologías e impartir unos contenidos y/o planificación que en muchos casos no soportarían una revisión profunda de los mismos. Con el EEES se ha pretendido forzar la mejora de la docencia a coste de perder, en buena parte, aspectos relativos a la libertad de cátedra. Como consecuencia, se nivela al profesorado, obligando al profesor menos preocupado por aspectos docentes a enredarse en mil y una técnicas nuevas. Por otra parte, al profesor “excelente”, se le impone un corsé que puede llegar a dificultar su faceta innovadora.

En el marco de la Universidad, es precisamente la singularidad de sus individuos y su complementariedad lo que le aporta un valor diferencial a todo el conjunto. El papel de las instituciones debería permitir mejorar la calidad de la educación, motivando (u obligando) al profesorado que lo requiera pero dejando libertad para el ejercicio de los “buenos” profesores, (cualesquiera que estos sean, tema no fácil de determinar). Durante estos años de adaptación veremos cómo el EEES afecta a unos y a otros y, en todo caso, apelaremos al sentido común de la comunidad universitaria para que, pese a todo, el resultado final sea satisfactorio.

Referencias

- [1] Curso de Iniciación a la Docencia Universitaria (3ª Edición), Múltiples autores. Universidad de Granada 2010-2011. Material e información disponible en: http://calidad.ugr.es/pages/secretariados/form_apoyo_calidad/programa_iniciacion_docencia/iniciacion3index
- [2] Cursos de formación para el profesorado del Vicerrectorado para la garantía de la Calidad, Universidad de Granada: http://calidad.ugr.es/pages/secretariados/form_apoyo_calidad/index#_doku_programa_de_ayuda_a_la_formacion
- [3] Curso de Planificación de la docencia universitaria por competencias y elaboración de Guías Didácticas (2ª ed), http://calidad.ugr.es/pages/secretariados/form_apoyo_calidad/programa-de-formacion-permante/curso_planificacion/2edicion/guiasdidacticas
- [4] Materiales resultantes del Curso de Planificación de la docencia universitaria por competencias y elaboración de Guías Didácticas (2ª ed). Disponibles en http://calidad.ugr.es/pages/secretariados/form_apoyo_calidad/programa-de-formacion-permante/curso_planificacion/2edicion/guiasdidacticas
- [5] Proyecto Tuning, desde <http://tuning.unideusto.org/tuningeu/>, (2004)
- [6] Zabalza Beraza, Miguel Ángel (2003) Competencias docentes del profesorado universitario, Calidades y desarrollo profesional. Narcea. Madrid.
- [7] Zabalza, M.A. y Zabalza, M. (2010). Planificación de la docencia en la Universidad. Madrid: Narcea.

- [8] Relación de memorias de títulos de Grado verificadas positivamente por la ANECA/AGAE <http://vicengp.ugr.es/pages/grados-verificados>
- [9] Juan Freire, Estrategias para la adaptación al Espacio Europeo de Educación Superior en titulaciones de Ciencias (Cáceres y Badajoz, 8 y 9 Marzo 2005). Plan de Formación Docente del Profesorado Universitario de la UEX. Cursos sobre Diseño de Planes Docentes. Organizado por la Oficina de Convergencia Europea de la Universidad de Extremadura.
- [10] Agencia Nacional de Evaluación de la Calidad y Acreditación ANECA: <http://www.aneca.es/>
- [11] Libros blancos: <http://www.aneca.es/Documentos-y-publicaciones/Libros-blancos2>
- [12] Riesgo, M., El enfoque por competencias en el EEES y sus implicaciones en la enseñanza y el aprendizaje, Tendencias pedagógicas, 13, pp. 79-105, (2008).
- [13] Aristimuño, A., Las competencias en la educación superior: ¿demonio u oportunidad? Departamento de Educación, Universidad Católica de Uruguay, Uruguay (2004).
- [14] Bain, K., Lo que hacen los mejores profesores (2ª Ed), Universitat de Valencia, Valencia, España, (2007).
- [15] Ejemplos PAT UGR: http://www.ugr.es/~vic_plan/formacion/ato/index.html
- [16] Romero, S. y Guillén, A., Evaluación y autoevaluación continuas automatizadas mediante herramientas de software libre. Congreso de Docencia Universitaria, Vigo (2009).

Experiencia interuniversitaria de docencia en un curso clásico de sistemas digitales dentro del EEES.

¹ Guillermo Botella, ¹Guadalupe Miñana, ²Samuel F. Romero ,

¹ Departamento de Arquitectura de Computadores y Automática, Universidad Complutense de Madrid, 28040, Madrid, España

² Departamento de Arquitectura y Tecnología de Computadores, Universidad de Granada, Granada, España

{gbotella,guamiro}@fdi.ucm.es; sromero@atc.ugr.es

Abstract. Reflexionamos respecto a los cursos clásicos de sistemas digitales impartidos dentro del Espacio Europeo de Educación Superior (EEES), contextualizamos dos casos prácticos de estudio: La Universidad Complutense de Madrid y la Universidad de Granada. Finalizamos el presente trabajo con los datos de una encuesta docente diseñada al efecto que nos permita correlacionar la opinión del estudiante con la distribución temática y organizativa de la asignatura. Creemos que esta primera experiencia puede ser utilidad para otros Centros de Educación Superior que aborden la misma temática.

Keywords: Sistemas Digitales, Fundamentos de Computadores, Docencia, Estudios de Grado, EEES, Encuesta docente.

1 Introducción

Esta contribución pretende presentar la experiencia docente y reflexionar sobre la percepción del alumnado sobre los conocimientos adquiridos en un curso clásico de introducción a los sistemas digitales. Este curso pertenece al primer cuatrimestre dentro de los estudios de Grado (EEES), se denomina Fundamentos de Computadores (nos centramos en el grupo más numeroso de todos) y se imparte en la Facultad de Informática de la Universidad Complutense de Madrid.

Si contextualizamos brevemente la misma, podemos afirmar que tiene 12 créditos BOE , 12 Créditos ECTS (teniendo 6 créditos correspondiente a sistemas digitales en el primer cuatrimestre y 6 a estructura de computadores en el segundo cuatrimestre), cada semana tiene 2 horas teóricas y 2 horas de contenido práctico: problemas o laboratorio alternos. El departamento responsable es el de Arquitectura de Computadores y Automática de la UCM.

Como objetivos en este primer cuatrimestre tenemos: Introducir la especificación y diseño digital de sistemas hardware y como conocimientos y destrezas que se adquieren: Aritmética Elemental; Especificación de sistemas digitales; Diseño de sistemas digitales síncronos; el método docente consta de la enseñanza presencial teórica, de problemas y de laboratorio.

2 Contextualización de la UCM

La Universidad Complutense de Madrid, como versa en su página web [1], fue pionera en España en el área de la Informática al introducirla como tema de investigación y docencia hace más de medio siglo. Ya en los años 50 existía un grupo de investigación de la UCM en estrecha colaboración con otros grupos de Cambridge y Harvard, universidades donde se estaba gestando lo que hoy conocemos como Informática. En la década de los 60, se impartieron cursos de doctorado en estos temas, así como cursos internacionales en el ámbito de la UNESCO. A principios de los 70, antes de la creación de las primeras Facultades de Informática del país, la UCM instaure especialidades de Informática en las Licenciaturas de Ciencias Matemáticas y Físicas.

En octubre de 1991 se crea en la UCM la Escuela Superior de Informática que después, en abril del año 2000, cambia su nombre por el de Facultad de Informática. En ella se han venido impartiendo las tres titulaciones oficiales de nivel universitario: Ingeniería en Informática, Ingeniería Técnica en Informática de Gestión e Ingeniería Técnica en Informática de Sistemas. Esta facultad ha sido el centro universitario de la Comunidad de Madrid más demandado para las tres titulaciones impartidas.

A partir del curso 2010-2011 se ponen en marcha tres nuevas titulaciones de grado dentro del Espacio Europeo de Educación Superior [2, 3] y que cumplen las especificaciones del Acuerdo del Consejo de Universidades, (B.O.E. 4 de agosto de 2009), sobre los títulos oficiales en el ámbito de la Ingeniería Técnica Informática. Las tres titulaciones dan acceso directo al máster en Ingeniería Informática. Estas titulaciones son: Graduado en Ingeniería Informática, Graduado en Ingeniería de Computadores y Graduado en Ingeniería del Software. También se imparten en la Facultad de Informática estudios oficiales de posgrado y doctorado (con mención de calidad ANECA) y diversos títulos propios de especialización profesional.

El edificio de la Facultad de Informática fue inaugurado en marzo de 2003 y alberga actualmente un total de 2200 alumnos.

En la asignatura Fundamentos de Computadores [4] (1er cuatrimestre, Grupo E) objeto de estudio de este trabajo, se tiene alumnado matriculado en Grado en Ingeniería de Computadores, Grado en Ingeniería Informática y Grado en Ingeniería del Software.

La Ingeniería de Computadores es una disciplina que se ocupa del diseño, desarrollo y administración de sistemas de proceso de información en sus aspectos hardware, software y de comunicaciones, con un conocimiento global de todas las áreas relacionadas con estos sistemas y con capacidad para liderar el desarrollo de proyectos y adaptarse de manera eficiente a un entorno de rápida evolución. El título de Graduado en Ingeniería de Computadores supone la adaptación, al Espacio Europeo de Educación Superior, del título existente en Ingeniería Técnica en Informática de Sistemas que, en el caso de la Universidad Complutense de Madrid, lleva impartándose desde 1991.

La Ingeniería Informática es la ciencia y la tecnología del diseño, implementación y mantenimiento de las componentes software y hardware que forman los modernos sistemas informáticos. Está sólidamente fundamentada en teorías y principios de computación, matemáticas, física e ingeniería y aplica todos ellos a la resolución de problemas técnicos que requieran el desarrollo de arquitecturas software, hardware y de red que presenten un equilibrio entre diferentes requisitos y objetivos contrapuestos. El título de Graduado en Ingeniería Informática supone la adaptación, al Espacio Europeo de Educación Superior, del título existente con anterioridad a la Ley Orgánica de Universidades 4/2007, de 12 de abril, por la que se modifica la Ley 6/2001, de 21 de diciembre en Ingeniería en Informática que, en el caso de la Universidad Complutense de Madrid, lleva impartándose desde 1991.

La Ingeniería del Software es una disciplina que se ocupa de los sistemas de información en un entorno empresarial, con énfasis en el desarrollo de sistemas de información, bases de datos, modelos de gestión y gestión de la empresa y con capacidad para liderar el desarrollo de proyectos y adaptarse de manera eficiente a un entorno de rápida evolución. El título de Graduado/a en Ingeniería del Software supone la adaptación, al Espacio Europeo de Educación Superior, del título existente en Ingeniería Técnica en Informática de Gestión que, en el caso de la Universidad Complutense de Madrid, lleva impartándose desde 1991.

3 Metodología Docente y encuesta

La asignatura se desarrolla en tres tipos de sesiones, como se ha apuntado previamente:

Clases de teoría: Exposición de los temas de programa y resolución de ejemplos ilustrativos; se realizan conjuntamente en el aula.

Clases de problemas: Resolución de problemas y dudas relacionados con los conceptos explicados en las sesiones de teoría; se realizan en grupos y la asistencia es obligatoria.

Clases de laboratorio: Realización de prácticas que consisten en el montaje de circuitos digitales y su estudio con herramientas de diseño y simulación; se realizan en grupos y la asistencia es obligatoria. Como bibliografía para la asignatura se recomienda [5-8].

ENCUESTA SOBRE LA ASIGNATURA “FUNDAMENTOS DE COMPUTADORES” 1er CUATRIMESTRE Grupo E.

CONOCIMIENTOS ADQUIRIDOS:

- He asistido a clase :

Teórica → Nada Poco Algo Bastante Mucho
Problemas → Nada Poco Algo Bastante Mucho
Laboratorio → Nada Poco Algo Bastante Mucho

- Teniendo en cuenta esta asistencia, considero que he adquirido un conocimiento teórico global de la asignatura:

Teórica → Nada Poco Algo Bastante Mucho
Problemas → Nada Poco Algo Bastante Mucho
Laboratorio → Nada Poco Algo Bastante Mucho

- He hecho un esfuerzo en esta asignatura :

Teórica → Nada Poco Algo Bastante Mucho
Problemas → Nada Poco Algo Bastante Mucho
Laboratorio → Nada Poco Algo Bastante Mucho

- Pienso que voy a obtener una calificación en la asignatura :

Teórica → Nada Poco Algo Bastante Mucho
Problemas → Nada Poco Algo Bastante Mucho
Laboratorio → Nada Poco Algo Bastante Mucho

CONTENIDOS Y ORGANIZACIÓN DE TEORÍA/PROBLEMAS/LABORATORIOS [4, 5, 6, 7]:

Teniendo en cuenta el programa de la asignatura siguiente:

- A. Puntuar de 1 a 5 los conocimientos adquiridos
- B. Distribuir 5 prácticas que refuercen los conocimientos de teoría (P1,..P5)
- C. Repartir (en %) las horas de teoría que el profesor dedicaría al temario
- D. Repartir (en %) las horas de problemas que el profesor dedicaría al temario

1. Representación de la información.

- Representación de números en un computador. Sistemas de numeración
- Conversión entre bases. Conversión a potencias de la base.
- Representación de números naturales y enteros
- Representación de la información alfanumérica. Códigos.

2. Especificación de sistemas combinatoriales.

- Especificación mediante funciones de conmutación.
- Especificación mediante expresiones de conmutación.
- Algebra de Boole. Manipulación algebraica de expresiones de conmutación.
- Formas canónicas de expresiones de conmutación.

Mapas de Karnaugh. Simplificación de expresiones de conmutación.

3. Implementación de sistemas combinacionales.

Puertas lógicas.

Análisis y síntesis de redes de puertas AND-OR-NOT.

Conjuntos universales de módulos.

4. Módulos combinacionales básicos.

Descodificador.

Codificador.

Multiplexor.

Read-Only memories (ROM)

Programmable Logic Arrays (PLA)

Sumador/restador.

5. Especificación de sistemas secuenciales.

Concepto de estado y diagrama de estados.

Sistemas síncronos y asíncronos.

Máquinas de Mealy y de Moore.

Ejemplos: Diseño de contadores, generadores y reconocedores de secuencias.

6. Implementación de sistemas secuenciales.

Biestables RS y D

Método de obtención de una especificación binaria.

Ejemplos: Implementación de contadores, generadores y reconocedores de secuencias.

7. Módulos secuenciales básicos

Registros y Registros de desplazamiento

Contadores

Diseño con Registros y Contadores

4 Resultados de la encuesta

Encuesta realizada con 68 personas: Se adjuntan valores medios.

Tema I	A	B	C	D
Representación de números en un computador. Sistemas de numeración	4		3%	3%
Conversión entre bases. Conversión a potencias de la base.	4		3%	4%
Representación de números naturales y enteros	5		5%	5%
Representación de la información alfanumérica. Códigos.	4		1%	0%

TOTAL	4,25	No Práct.	12%	12%
--------------	------	-----------	-----	-----

Tema 2	A	B	C	D
Especificación mediante funciones de conmutación.	3		3%	1%
Especificación mediante expresiones de conmutación.	3		3%	1%
Algebra de Boole. Manipulación algebraica de expresiones de conmutación.	4		5%	7%
Formas canónicas de expresiones de conmutación.	4		2%	1%
Mapas de Karnaugh. Simplificación de expresiones de conmutación.	5		6%	8%
TOTAL:	3,8	No Prácticas.	19%	18%

Tema 3	A	B	C	D
Puertas lógicas.	4		2%	1%
Análisis y síntesis de redes de puertas AND-OR-NOT.	4		2%	1%
Conjuntos universales de módulos.	3		3%	1%
TOTAL:	3,7	No Practicas.	7%	3%

Tema 4	A	B	C	D
Descodificador.	4	P1(propuesta)	4%	3%
Codificador.	4	P1(propuesta)	4%	5%
Multiplexor.	4	P2(propuesta)	4%	5%
<i>Read-Only memories</i> (ROM)	3		2%	1%
<i>Programmable Logic Arrays</i> (PLA)	3		2%	1%
Sumador/restador.	2	P2(propuesta)	6%	3%
TOTAL:	3,4	3 Prácticas.	22%	18%

Tema 5	A	B	C	D
Concepto de estado y diagrama de estados.	4		4%	7%
Sistemas síncronos y asíncronos.	5		1%	0%
Máquinas de Mealy y de Moore.	5		3%	5%
Ejemplos: Diseño de contadores, generadores y reconocedores de secuencias.	5		6%	9%
TOTAL:	4,75	No Prácticas	14%	19%

Tema 6	A	B	C	D
Biestables RS y D	4		5%	3%
Método de obtención de una especificación binaria.	4		3%	4%
Ejemplos: Implementación de contadores, generadores y reconocedores de secuencias.	4	P4(propuesta)	5%	8%
TOTAL:	4	1 Práctica	11%	15%

Tema 7	A	B	C	D
Registros y Registros de desplazamiento	4		5%	1%
Contadores	4		6%	6%
Diseño con Registros y Contadores	4	P5(propuesta)	4%	8%
TOTAL:	4	1 Práctica	15%	15%

5 Universidad de Granada

En la Escuela Técnica Superior de Ingenierías Informática y de Telecomunicaciones de la Universidad de Granada, el departamento de Arquitectura y Tecnología de Computadores imparte materias directamente relacionadas con los contenidos detallados para el caso de la Universidad Complutense de Madrid.

En concreto, en las titulaciones de Ingeniería Informática (a extinguir), así como en las Ingenierías Técnicas en Informática de Gestión y de Sistemas, se contemplan las materias de Tecnología de Computadores I [9] y Tecnología de Computadores, respectivamente se imparten contenidos asimilables a los descritos en

apartados anteriores.

Sin embargo, en la fase actual de implantación de los nuevos títulos de grado, las mencionadas asignaturas se encuentran ya extintas por pertenecer a los primeros cursos, y por tanto, no existe docencia presencial de la misma.

Estas asignaturas encontrarán (parcialmente) cobertura en la nueva materia de Tecnología y Organización de Computadores, que en el presente curso 2010/2011 se impartirá en el segundo cuatrimestre.

Por tanto, en la fase en la que actualmente nos encontramos, no es posible la aplicación de dicha experiencia, si bien, los resultados de la misma en el caso de la UCM nos permiten anticipar mecanismos similares que prevemos ensayar en la puesta en marcha de nuestra nueva asignatura de Grado en Ingeniería Informática. Los resultados que entonces se recogerán nos permitirán modelar y reajustar la docencia en la materia en su momento inicial, de forma que podamos obtener un mejor rendimiento académico.

6 Conclusiones y trabajo Futuro:

Se observa dentro del apartado A, un conocimiento por encima del valor medio exceptuando en el caso del sumador/restador donde se ha recibido una calificación de 2/5.

Si analizamos el apartado B el alumnado propone prácticas en el tema 4. Aquí tenemos sendas propuestas de prácticas empatadas en porcentaje: Uso de Codificadores y uso de Descodificadores por un lado y uso de multiplexores y sumador /restador por otro. En el tema 6 tenemos una propuesta mayoritaria de diseño de sistemas secuenciales y en el tema 7 la mayoritaria es el diseño con contadores.

Las prácticas realizadas este curso consistieron en:

- 1ª Sesión: Diseño y simulación de un circuito combinacional usando puertas lógicas,
- 2ª Sesión: Montaje de un circuito combinacional usando puertas lógicas,
- 3ª Sesión: Diseño, simulación y montaje de un Conversor de código Gray a Binario,
- 4ª Sesión: Diseño, simulación y montaje de un reconocedor de patrón,
- 5ª Sesión: Diseño y montaje de un registro.

Vemos como exceptuando la última donde el alumnado propone usar contador, registro en lugar de diseñar uno, las prácticas coinciden con las realizadas este curso.

Los apartados C,D dedicación de horas de teoría y problemas están bastante repartidos predominando los siguientes epígrafes:

Manipulación Algebraica (7%), Mapas de Karnaugh (8%), Contadores (6%), Diseño con registros y contadores (8%), Implementación de Contadores, reconocedores de secuencia (9%). Respecto a problemas sólo estos puntos suponen casi un 40% del tiempo total de problemas. Sin embargo respecto a horas de teoría, por bloques podemos decir que predominan los temas 2 y 4 con un 41 % de contenido propuesto.

Este último balance es bastante curioso ya que el alumnado considera más importante aumentar el tiempo que se dedica a clases prácticas en sistemas secuenciales y simultáneamente aumentar las horas docentes en teoría de sistemas combinatorios.

Esta experiencia ha servido para obtener unos resultados visibles que impulsen nuestra mejora como docentes dentro del EEES, como trabajo futuro correlacionaremos estos resultados con los obtenidos en la evaluación de Febrero y Junio y compararemos los resultados con otras universidades.

7 Bibliografía

1. <http://www.ucm.es>
2. Bologna Working Group on Qualifications Frameworks. A Framework for Qualifications of the European Higher Education Area (2005), [online] en http://www.bologna-bergen2005.no/Docs/00- Main doc/050218_QF_EHEA.pdf
3. De Miguel, M. Cambio de paradigma metodológico en la educación superior. Exigencias que conlleva. Cuadernos de Integración Europea, 2, 16-27. (2005).
4. Asignatura de Fundamentos de Computadores en la UCM, Guía Docente [online] http://www.fdi.ucm.es/Guia_Docente/ver_prog_asig.asp?Titulo=0846&Asignatura=803265&fdicurso=2010-2011
5. Hermida, R., del Corral, A.M., Pastor, E., Sánchez, F., "Fundamentos de Computadores", Ed. Síntesis. 1998.
6. Daniel D. Gajski, "Principios de Diseño Digital", Ed. Prentice Hall.
7. A. Cuesta, J.I. Hidalgo, J. Lanchares, J.L. Risco, "Problemas de Fundamentos y estructura de Computadores", Ed. Prentice Hall, 2009".
8. Baena, C.; Bellido, J.; Molina, A.; Parra, M.; Valencia, M., "Problemas de Circuitos y Sistemas Digitales", Ed. McGraw-Hill, 1997.
9. Asignatura de Tecnología de Computadores I en la UGR, Guía Docente [online] <http://etsiit.ugr.es/planes/index.php?id=3&id2=12>

Resultados de una Experiencia Piloto de Implantación del Espacio Europeo de Educación Superior en la Universidad de Granada

Francisco José Navarro-Ríos

Computer and Communication Service Center, University of Granada
18071 Granada, Spain
francisco@ugr.es

Resumen. Este trabajo explica qué es el Espacio Europeo de Educación Superior (EEES) y cómo se ha llevado a cabo una experiencia piloto de su implantación en la Universidad de Granada, mediante los créditos ECTS, el Suplemento Europeo al Título, las Guías ECTS, la nueva estructuración de las enseñanzas, la nueva metodología docente integrada por distintas actividades: clases magistrales, clases prácticas, seminarios, tutorías, lecturas obligatorias, visitas programadas a distintas instituciones, pruebas y prácticas autoevaluables, etc. y la variación de los criterios de evaluación del estudio y trabajo del alumno.

Palabras Clave: EEES, ECTS, educación, Bolonia, Europa, crédito.

1. Introducción

La Declaración de Bolonia fue firmada en Junio de 1999 por los Ministros de Educación de veintinueve países europeos para conseguir en el año 2010 un Espacio Europeo de Educación Superior (EEES), es decir, un sistema educativo europeo de calidad que permita a Europa fomentar su crecimiento económico, su competitividad internacional y su cohesión social a través de la educación y formación de los ciudadanos a lo largo de la vida y su movilidad. Después, se han estado firmando otras declaraciones: Berlín (2001), Praga (2001), Graz (2003), Bergen (2005), Londres (2007), Lovaina (2009) y Budapest-Viena (2010). La próxima será en Bucarest en 2012.

Los objetivos estratégicos del EEES son impulsar la movilidad de estudiantes, profesores y personal de administración y servicios de las universidades y de otras instituciones de educación superior europeas, promocionar la cooperación europea en materia de garantía de calidad y desarrollo de criterios y metodologías comparables, conseguir mayores oportunidades de trabajo, fomentar la dimensión europea en la educación superior como condición necesaria para el logro de los objetivos del EEES, disponer de un sistema fácilmente comprensible y comparable de titulaciones basado en dos niveles y tres ciclos: grado y posgrado (máster y doctorado), adoptando un sistema de acumulación y transferencia de créditos que favorezca la movilidad (créditos ECTS).

2. Créditos ECTS y Suplemento Europeo al Título

El Sistema Europeo de Créditos ECTS (European Credit Transfer System) es un sistema que permite medir el trabajo que tienen que llevar a cabo los alumnos para la adquisición de los conocimientos, capacidades y destrezas necesarias para superar las materias de su plan de estudios. La actividad de estudio (entre 25 y 30 horas por crédito) incluye el tiempo dedicado a clases lectivas, horas de estudio, tutorías, seminarios, trabajos, prácticas o proyectos, así como las exigidas para la preparación y la realización de exámenes y evaluaciones. La finalidad es facilitar la movilidad de los estudiantes mediante la utilización de un sistema de créditos común en todo el EEES. Este sistema se ha generalizado a partir de los programas de movilidad de estudiantes, fundamentalmente Erasmus.

El Suplemento Europeo al Título (SET) es un documento personal para cada titulado, que añade información al título, mediante una descripción de su naturaleza, nivel, contexto y contenido, con el objetivo de incrementar la transparencia de las diversas titulaciones impartidas en Europa, y facilitar su reconocimiento académico y profesional por las instituciones.

3. Guías ECTS y estructura de las enseñanzas

En la Universidad de Granada, para converger hacia el EEES, se establecieron dos fases: una de planificación, y una segunda de implantación. La fase de planificación finalizó con la publicación de las llamadas Guías ECTS, que pueden ser para la titulación o para la asignatura. Las primeras, cumpliendo con los requisitos de información establecidos por la Comisión Europea, contienen todos los elementos para su uso en la elaboración del SET (Suplemento Europeo al Título). Las guías pertenecientes a la asignatura tienen como misión expandir la información de la titulación, y alimentan una base de datos informática que permite la búsqueda pormenorizada de información.

Las enseñanzas universitarias se estructuran en dos niveles (grado y posgrado) y en tres ciclos. El primer ciclo conduce a títulos de grado. El posgrado comprende el segundo y el tercer ciclo, conducentes a los títulos de máster y doctor, respectivamente. El primer ciclo comprende enseñanzas básicas y de formación general junto a otras orientadas al ejercicio de actividades profesionales, y el título obtenido debe tener relevancia en el mercado laboral nacional y europeo. El segundo ciclo está dedicado a la formación avanzada, multidisciplinar o especializada. El tercer ciclo tiene como finalidad la formación avanzada del estudiante en las técnicas de investigación. El grado debe comprender 240 créditos, y la cantidad total de créditos de las enseñanzas y actividades académicas conducentes a la obtención de un título de máster estará comprendido entre 60 y 120 créditos. Para la obtención del título de doctor, se exige la elaboración y la presentación de una tesis doctoral, consistente en un trabajo científico con resultados de investigación originales.

4. Experiencia en la implantación

El crédito europeo, según el artículo 3 del Real Decreto 1125/2003, es la unidad de medida del haber académico que representa la cantidad de trabajo del estudiante para cumplir los objetivos del programa de estudios y que se obtiene por la superación de cada una de las materias que integran los planes de estudios de las diversas enseñanzas conducentes a la obtención de títulos universitarios de carácter oficial y validez en todo el territorio nacional. En esta unidad de medida se integran las enseñanzas teóricas y prácticas, así como otras actividades académicas dirigidas, con inclusión de las horas de estudio y de trabajo que el estudiante debe realizar para alcanzar los objetivos formativos propios de cada una de las materias del correspondiente plan de estudios.

Sin embargo, según el sistema de créditos LRU, cada uno de los créditos que integran la carga lectiva de las asignaturas de un plan de estudios se corresponde con diez horas de docencia efectiva del profesor. Esto significa que el crédito LRU no toma en consideración ni el desarrollo de otras actividades alternativas a la clase tradicional por parte del profesor ni tampoco, las horas de trabajo y estudio que los estudiantes deben invertir para la superación de las asignaturas. Por otra parte, en conexión con la carga lectiva del ECTS, debe añadirse también que, de conformidad con el artículo 4.5 del mismo cuerpo reglamentario, el número mínimo de horas por crédito será de 25 y el máximo de 30. Para realizar una conversión aproximada entre el crédito LRU y el nuevo crédito europeo ECTS, se procederá a multiplicar el número de créditos actuales por veinticinco (créditos LRU x 25 horas), que son las horas que, como mínimo, deben empeñar en estudio y trabajo los estudiantes por cada crédito de la asignatura de que se trate. Esto significa que una asignatura que posea 4 créditos LRU deberá tener una carga aproximada de 100 horas para el alumno.

En definitiva, con el crédito europeo, se ha de considerar no sólo la labor llevada a cabo por el profesor, sino también las horas de trabajo y estudio que los estudiantes deben invertir para la superación de las asignaturas, lo que requiere de una nueva metodología docente, innovadora, que permita incrementar la calidad de la docencia, y hacer que los alumnos sean los verdaderos protagonistas de su experiencia de aprendizaje. La docencia tradicional, en la que el profesor era el principal protagonista, ha sido sustituida por una nueva docencia, donde las actividades son dirigidas y supervisadas por el profesor, y el trabajo del alumno es lo más destacado.

En algunos centros de la Universidad de Granada, se ha llevado a cabo una experiencia piloto, donde las asignaturas han desarrollado una metodología docente integrada por las siguientes actividades:

- Clases magistrales, destinadas a la exposición teórica de las distintas lecciones que integran el programa de la asignatura.
- Clases prácticas, dirigidas al planteamiento, discusión, y resolución de diferentes supuestos prácticos relacionados con la asignatura, como, por ejemplo, la búsqueda de recursos electrónicos que suministren el material bibliográfico necesario para ello.

- Seminarios monográficos relacionados con diferentes aspectos de la asignatura en cuestión. Eventualmente se celebrarán también seminarios multidisciplinares con profesores de distintas asignaturas sobre temas que trasciendan de un solo área de conocimiento.
- Tutorías individualizadas o en pequeños grupos destinadas a la resolución de dudas específicas sobre los puntos más dudosos y problemáticos del programa.
- Otras actividades dirigidas como lecturas obligatorias, visitas programadas a distintas instituciones, pruebas y prácticas autoevaluables, etc.

En esta experiencia piloto, la carga lectiva determinada por la impartición de las clases teóricas y prácticas era, como mínimo, de un 70 % del total de la asignatura. El resto de actividades alternativas no podría superar, en consecuencia, el 30 % de la carga lectiva global de la asignatura en cuestión.

Además, el impulso de una nueva metodología docente que pretende la implantación del sistema europeo de créditos comporta necesariamente una variación de los criterios de evaluación del estudio y trabajo del alumno. En consecuencia, una valoración adecuada del esfuerzo y rendimiento del estudiante exige diferenciar si el objeto de dicha evaluación viene determinado por los conocimientos de que deben haberse adquirido durante la asistencia a las clases teóricas y prácticas o si, por el contrario, la valoración se refiere al resto de las actividades alternativas. Por lo tanto, la calificación global de cada asignatura puede determinarse por la ponderación de las siguientes partes:

- Calificación de los conocimientos adquiridos a través de las clases teóricas y prácticas, que se lleva a cabo a través de un examen final, y en su caso, también parcial, en el que el alumno demuestre haber asimilado el contenido de las mismas.
- Es posible y recomendable la sustitución de tal examen o exámenes por un sistema de pruebas periódicas o de evaluación continua. El resultado de esta evolución supondrá como mínimo un 70 % de la nota final.
- Calificación de los conocimientos adquiridos en los seminarios, trabajos dirigidos, control de las lecturas obligatorias, cuya calificación variará según la naturaleza de la actividad de que se trate. Normalmente se conjugarán los controles de asistencia con la evaluación concreta de la función desarrollada. Esta parte de la evaluación no podrá superar el 30 % de la calificación final.

De acuerdo con el sistema de calificaciones prescrito por el artículo 5 del Real Decreto 1125/2003, la calificación final se fijará numéricamente en una escala del 1 al 10 donde, según lo fijado anteriormente, la calificación de los conocimientos adquiridos en las clases teóricas y prácticas se puntuará como mínimo del 1 al 7, y los adquiridos en el resto de actividades se puntuará, como máximo, con 3 puntos de la calificación final.

5. Conclusiones

Se ha puesto en práctica un nuevo enfoque metodológico que ha transformado nuestro sistema educativo (basado en la enseñanza) a otro basado en el aprendizaje. La mejora es interactiva, y se basa en una mayor implicación y autonomía de los alumnos mediante el uso de metodologías más activas (tutorías, seminarios, trabajo en equipo, tecnologías multimedia, casos prácticos, etc.) y en el papel de los profesores, como creadores de entornos de aprendizaje que estimulen a los estudiantes.

Los alumnos, con su capacidad de adaptación, esfuerzo y confianza en las ventajas que la convergencia implica para su educación formal y aprendizaje a lo largo de la vida, son los verdaderos beneficiarios del nuevo modelo educativo, y los principales protagonistas del cambio.

Con su doble misión tradicional de investigación y docencia, la universidad se está colocando en la vanguardia del pensamiento, y está siendo referente de soluciones para nuestra plena incorporación a la sociedad del conocimiento.

Se está consolidando un sistema de becas y ayudas que garantice el acceso a los estudios universitarios de grado y posgrado a todos los ciudadanos en igualdad de oportunidades, y que elimine los obstáculos para la movilidad.

Referencias

1. Real Decreto 1044/2003, de 1 de agosto, por el que se establece el procedimiento para la expedición por las universidades del Suplemento Europeo al Título.
2. Real Decreto 1125/2003, de 5 de septiembre, por el que se establece el sistema europeo de créditos y el sistema de calificaciones en las titulaciones universitarias de carácter oficial y validez en todo el territorio nacional.
3. Real Decreto 55/2005, de 21 de enero, por el que se establece la estructura de las enseñanzas universitarias y se regulan los estudios universitarios oficiales de Grado.
4. Real Decreto 56/2005, de 21 de enero, por el que se regulan los estudios universitarios oficiales de Posgrado.
5. Directiva 36/2005 del Parlamento Europeo y del Consejo de la Unión Europea, de 7 de septiembre, relativa al reconocimiento de cualificaciones profesionales.
6. Real Decreto 1509/2005, de 16 de diciembre, por el que se modifican el Real Decreto 55/2005, de 21 de enero, por el que se establece la estructura de las enseñanzas universitarias y se regulan los estudios universitarios oficiales de grado y el Real Decreto 56/2005, de 21 de enero, por el que se regulan los estudios universitarios oficiales de posgrado.
7. Real Decreto 188/2007, de 9 de febrero, por el que se modifica el Real Decreto 774/2002, de 26 de julio, por el que se regula el sistema de habilitación nacional para el acceso a Cuerpos de Funcionarios Docentes Universitarios y el régimen de los concursos de acceso respectivos, modificado por el Real Decreto 338/2005, de 1 de abril.
8. Real Decreto 189/2007, de 9 de febrero, por el que se modifican determinadas disposiciones del Real Decreto 56/2005, de 21 de enero, por el que se regulan los estudios universitarios oficiales de posgrado.
9. Real Decreto 1393/2007, de 29 de octubre, por el que se establece la ordenación de las enseñanzas universitarias oficiales.

Experiencias en la Asignatura Diseño y Evaluación de Configuraciones*

Antonio M. Mora¹, Pablo García-Sánchez², y Juan J. Merelo²

¹ Depto. de Lenguajes y Sistemas Informáticos

² Depto. de Arquitectura y Tecnología de Computadores
Universidad de Granada

{amorag, pgarcia, jmerelo}@geneura.ugr.es

Resumen En este trabajo se exponen las experiencias docentes que han tenido durante los últimos cinco años los profesores de la asignatura Diseño y Evaluación de Configuraciones, asignatura optativa de las Ingenierías Técnicas en Informática en la Universidad de Granada. Igualmente, también se comentan las herramientas utilizadas para gestionar dicha asignatura, intentando facilitar, tanto a los alumnos, como al profesor, todas las tareas propias de la misma. El principal objetivo de esta asignatura es dar una metodología para la evaluación de prestaciones (o rendimiento) de un ordenador. Se divide, a grosso modo, en tres partes: una parte dedicada a los monitores, o herramientas encargadas de medir la carga de un ordenador, otra parte dedicada a la mejora de prestaciones, y otra parte dedicada a la reproducción de la carga de un ordenador, los llamados benchmarks.

1. Introducción

La asignatura Diseño y Evaluación de Configuraciones tiene como objeto de estudio los *sistemas informáticos*, es decir, cualquier sistema que use medios informáticos, abarcando, todos los sistemas situados en diferentes niveles del modelo de capas OSI [1]: desde la más baja, la física, hasta la más alta, la de aplicación; junto con las diferentes capas de un sistema operativo. Desde el punto de vista de esta asignatura, un sistema puede ser tanto un chip, una tarjeta de red, o una red completa, como un programa que ofrezca servicios en esa red, o el programa junto con todo el sistema necesario para ejecutarse.

Durante el ciclo de vida de un sistema informático, resulta muchas veces necesario evaluar sus prestaciones o rendimiento, habitualmente con el objetivo de mejorarlas o bien de comparar diversos sistemas informáticos entre sí. Esa evaluación de prestaciones se debe hacer de forma objetiva, a fin de poder comparar distintos valores a lo largo del tiempo o bien el mismo valor para diversos sistemas informáticos. Tales mediciones pueden servir también para identificar los problemas que tiene un sistema informático, con el objetivo de solucionarlos.

Esta asignatura se ha convertido en los últimos años en una de las más exitosas en cuanto al número de alumnos matriculados dentro de las asignaturas

* Financiado por el proyecto EvOrq (TIC-3903) y la Beca FPU AP2009-2942.

optativas de las Ingenierías Técnicas Informáticas en la Universidad de Granada. Gran parte de su éxito se basa tanto en la novedosa forma de evaluación a los alumnos, no fundamentada en exámenes, sino en su trabajo en clase, como en la orientación que se ha dado a la misma, primando la parte práctica sobre la teórica. Otro factor relevante ha sido la forma de trabajo, centrada en el uso de herramientas web ampliamente conocidas y extendidas, a la par que llamativas para los alumnos (como Sistemas web de ayuda a la docencia, Wikis, Blogs o Foros).

En este trabajo comentaremos la citada forma de evaluación, así como el modo de funcionamiento e integración de las herramientas dentro de la asignatura.

2. Diseño y Evaluación de Configuraciones

Esta sección presenta los contenidos teóricos y prácticos de la asignatura.

2.1. Teoría

Tema 1: Sistemas informáticos y su evaluación. En este tema se analizan las diferentes magnitudes observables de un sistema informático y como se pueden medir, con una serie de ejemplos que indican como se hace en diversos sistemas operativos, como UNIX y Windows.

Inicialmente se realiza una introducción a los sistemas informáticos y las fases en la evaluación de estos sistemas. El siguiente apartado explica cómo elegir las *métricas* [2] de prestaciones y las distintas técnicas de evaluación. El último apartado de este tema explica cómo monitorizar la carga de un sistema y explica varias formas de hacerlo utilizando varios monitores, tanto para monitorizar el hardware (ver Figura 1), como el software (mediante un *profiler*) [3].

Tema 2: Representación gráfica de las prestaciones de los sistemas.

Para presentar y analizar los resultados obtenidos de la ejecución de un monitor sobre un sistema, o una comparativa entre varios sistemas, normalmente se usa algún tipo de gráfico, como por ejemplo gráficos de barras de evolución temporal (las llamadas *strip chart*). Más habitualmente, para resumir el rendimiento de todo el sistema se suele usar un diagrama de Gantt [4,5], o bien un gráfico de Kiviatt [5].

En esta parte de la asignatura también se explican algunos de los errores comunes en la representación gráfica de resultados, así como unas reglas y consejos para realizar dichas representaciones gráficas.

Tema 3: Mejora de prestaciones. Este tema está principalmente dedicado a examinar qué elementos pueden fallar (o estar sujetos a error) en un equipo informático [6], principalmente Unix o Windows NT, qué herramientas se usan para el diagnóstico, y una vez diagnosticado, qué medidas hay que tomar para mejorar las prestaciones.

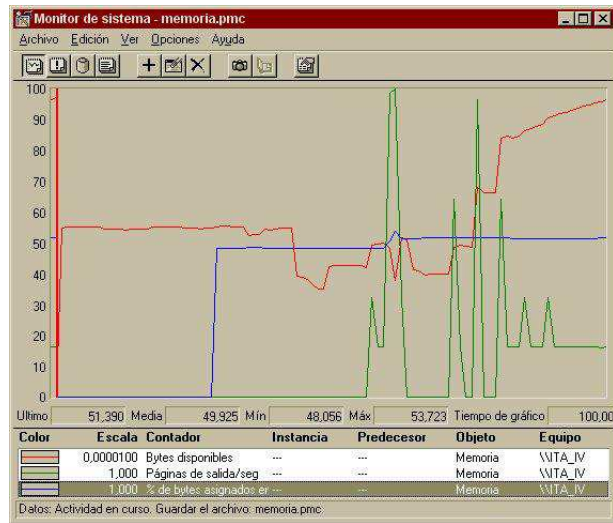


Figura 1. Monitor de Windows NT. Este monitor da información del sistema en tiempo real.

- Gestión de carga y prestaciones en el sistema operativo
- Políticas de gestión del sistema
- Mejora de prestaciones de la CPU
- Sintonización de la memoria
- Mejora de prestaciones en entrada/salida
- Optimización de un servidor web

Tema 4: Caracterización de la carga: benchmarks. En este tema se explica qué es lo que se tiene que tener en cuenta para medir la carga de un equipo, y qué es un *benchmark* [3], el cual es un programa o conjunto de programas que evalúan las prestaciones de un sistema informático, reproduciendo una carga de trabajo genérica o específica en dicho sistema informático. Al proceso de comparar dos o más sistemas mediante la obtención de medidas se le denomina *benchmarking*.

En general, para evaluar las prestaciones de un sistema informático es necesario conocer y caracterizar previamente cuál es la carga de trabajo, como se habrá visto en temas anteriores. Sin embargo, en muchos casos tal carga no se conoce de antemano, es difícil de caracterizar o es suficientemente amplia como para considerarla una carga genérica.

En esta sección se explican los pasos para diseñar o escoger un buen benchmark, siguiendo lo propuesto por [7], así como los tipos que existen y las métricas y errores más comunes. Finalmente se presentan algunos ejemplos de benchmarks, como son los propuestos por la SPEC¹ (Standard Performance Eva-

¹ <http://www.spec.org>

uation Corporation) o LINPACK, que son los utilizados para medir la lista de los computadores más potentes o Top 500².

2.2. Prácticas

Esta sección explica las distintas prácticas que los alumnos deben realizar para superar la asignatura. Por regla general, para realizar estas prácticas hay que seguir la metodología de la asignatura, es decir proponer un objetivo, hacer medidas, diagnosticar, proponer una solución, volver a tomar medidas, y comparar el sistema viejo con el sistema nuevo.

Práctica 1: Búsqueda de recursos relacionados con la asignatura. Esta primera práctica consiste en buscar sitios web o referencias bibliográficas (libros, revistas, artículos, noticias, etc) o cualquier otra fuente donde se pueda encontrar información útil relacionada con los temas tratados en la asignatura.

Práctica 2: Instalación y configuración de sistemas de medición de prestaciones. Utilizando los recursos de la práctica anterior, esta práctica consiste en descargar de Internet monitores (de prestaciones), y configurarlos e instalarlos para medir las prestaciones del ordenador personal del alumno o bien de los ordenadores de las aulas de prácticas.

Práctica 3: Uso de un profiler. Se hará uso de uno de estas herramientas para evaluar las prestaciones de un programa propio realizado en cualquier lenguaje, y ver en qué puntos se está empleando más tiempo, para que, a partir de ahí, se busquen modos de mejorarlo. Se tendrá que buscar cómo configurarlo, como hacerlo funcionar, y finalmente, usarlo. La Figura 2 muestra la ejecución de un profiler sobre un programa Java realizado por un alumno.

Práctica 4: Uso de programas de monitorización de un sistema. Esta práctica consiste en la definición de una carga computacional, y la utilización de herramientas de monitorización del sistema para visualizar cómo ejercita al sistema esas cargas. Estas herramientas presentan medidas del mismo en tiempo real, permitiendo medir y cuantificar su carga, medir su evolución y predecir su comportamiento. La herramienta de monitorización la elegirá el alumno en función de lo hecho en otras prácticas (especialmente la segunda) y de su adecuación para las magnitudes que desee medir.

Práctica 5: Mejora de las prestaciones de un sistema. Se trata de poner en práctica lo aprendido en el Tema 3 de la asignatura sobre mejora de prestaciones de sistemas informáticos y aplicarlo a un caso determinado. Por ejemplo, modificar un programa para que consuma menos recursos o cambiar la

² <http://www.netlib.org/benchmark/top500.html>

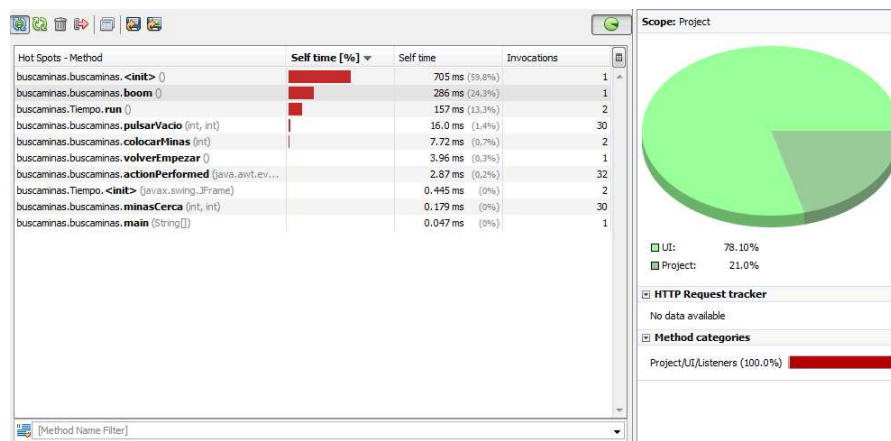


Figura 2. Captura de pantalla del profiler para Java incluido en el editor NetBeans.

configuración de un servidor web para que procese mejor la cola de peticiones. Una vez establecido el objetivo, se tomarán medidas de, al menos, un aspecto, antes de realizar ningún tipo de diagnóstico, con la carga de trabajo con la que se vaya a hacer la práctica. Éste será el sistema base. En función de esto, se decidirá qué medidas tomar para mejorar las prestaciones del sistema.

Práctica 6: Programación de un benchmark portable. En ella se hace énfasis en la programación independiente de la máquina, y se trata de, utilizando un lenguaje de alto nivel, comparar las prestaciones de diferentes máquinas o de la misma con diferentes sistemas operativos. En resumen, el alumno realizará un programa en un lenguaje de alto nivel que permita medir y comparar las prestaciones de dos sistemas. Comparar, por ejemplo, diferentes equipos con diferentes sistemas operativos, o el mismo equipo con dos sistemas operativos diferentes. Como en todas las prácticas, hay que establecer claramente cuál es el objetivo del benchmark, obtener resultados, analizarlos, presentarlos y obtener un índice que indique qué sistema es mejor para los objetivos planteados.

2.3. Evaluación

La evaluación de la asignatura se realiza como sigue:

- 1/6 de la nota será nota de clase (participación en actividades y ejercicios de autoevaluación)
- 2/6 de la nota corresponderá a la media ponderada (por sesiones) de la nota obtenida en prácticas
- 3/6 de la nota será la correspondiente a un trabajo final, entregado al término de las clases, que agrupará todo lo aprendido para evaluar y mejorar un sistema a elección del alumno, pero de cierta 'entidad'

Como se puede ver, no existe ningún tipo de examen a lo largo de la asignatura y únicamente se valora el trabajo de los alumnos (tanto en clase, como en casa). Este tipo de evaluación resulta bastante exitosa y, a tenor de lo demostrado por los alumnos, éstos retienen gran parte de los contenidos (conocimientos adquiridos).

3. Herramientas utilizadas en la asignatura

Esta asignatura ha sido pionera en varias facetas de compaginación de herramientas y utilidades web dentro de las clases, permitiendo, e incluso fomentando el uso del ordenador portátil en las aulas (llegando incluso a proveer de varios de estos portátiles a los alumnos).

El objetivo que se persigue es el de involucrarlos en mayor medida en las clases, haciendo que tomen apuntes y notas en un *Wiki de la asignatura* (página web dinámica en la que se generan contenidos entre los usuarios de la misma) (<http://dyec-ugr.wikispaces.com/>), que todos los demás (incluido el profesor) podrán consultar y completar posteriormente. De esta forma, el conocimiento generado (siguiendo la filosofía de los wikis) va siendo cada vez más completo y riguroso, uniendo además el punto de vista que los alumnos le quieran dar para facilitar su comprensión. Los alumnos encargados de estas labores reciben parte de la nota de participación en clase como premio.

El profesor además, hace uso de dicho Wiki para poner ejemplos y ejercicios, que los alumnos realizan, comentan y corrigen.

Por otra parte, otro aspecto fomentado es la creación de *Blogs de alumnos*, en los que se describan las inquietudes, intereses, enlaces y trabajos de los mismos en relación a la asignatura. Para ello, se insta a que éstos hagan la entrega de sus prácticas a modo de página web, consiguiendo igualmente un formato legible (y bonito) en todos los sistemas operativos. Esta medida suele tener un gran acogimiento entre los alumnos, manteniendo muchos de los cuales dicho blog en adelante.

La otra herramienta utilizada es *SWAD: Sistema Web de Apoyo a la Docencia* (<http://swad.ugr.es>) [8], una plataforma libre de teleformación desarrollada y utilizada en la Universidad de Granada en los últimos 11 cursos académicos. SWAD integra diversas funciones de apoyo al aprendizaje, a la docencia y a la gestión de los datos de los estudiantes. Entre ellas el acceso a información sobre las asignaturas (guía docente, horarios, bibliografía,...), la descarga de documentos (transparencias, relaciones de problemas), las listas y fichas de alumnos y profesores, los foros de discusión, la asignación de actividades, la autoevaluación mediante exámenes interactivos o la consulta individual de calificaciones.

En dicha plataforma se hace el seguimiento de los alumnos, la publicación de los temarios y guiones de prácticas, la entrega de ejercicios, trabajos y prácticas, y la publicación (con consulta individual) de la notas.

Respecto al software a utilizar en la asignatura, todo el propuesto y presentado en secciones anteriores es software libre (o tiene alternativa libre), de hecho

se fomenta también la búsqueda de nuevas utilidades de libre distribución, con lo que el coste para el alumno es nulo.

Todas estas herramientas y las peculiaridades o ventajas del sistema de evaluación, hacen de la asignatura una de las más populares entre las optativas, contando generalmente con un gran número de alumnos en cada curso académico.

4. Conclusiones

En este trabajo se ha descrito una de las asignaturas pioneras en la utilización y el fomento de uso de herramientas de software libre, así como utilidades web que ayudan a los alumnos a hacer más amenas e interesantes las clases. Se trata de Diseño y Evaluación de Configuraciones, asignatura optativa de las Ingenierías Técnicas Informáticas en la Universidad de Granada.

Entre dichas herramientas se incluyen SWAD, una plataforma web para facilitar las tareas de docencia, un Wiki de la asignatura, que los alumnos completan y gestionan entre todos (junto con el profesor) y diversos blogs de alumnos que versen sobre la asignatura y sus contenidos.

Los años de experiencia impartíendola nos han demostrado que el sistema funciona, y que los alumnos adquieren y afianzan conocimientos, aún sin tener que realizar ningún examen.

Toda la información, software y ejemplos de la asignatura se encuentra en la web <http://geneura.ugr.es/~jmere10/DyEC/>, a disposición de los alumnos y el resto de personas interesadas en la temática.

Referencias

1. Zimmermann, H.: Osi reference model. *IEEE Transactions on Communications* **28**(4) (1980) 425–432
2. Molero, X., Juiz, C., Rodeño, J.: *Evaluación y modelado del rendimiento de los sistemas informáticos*. Pearson Education (2004)
3. Loukides, M.: *System performance tuning*. O'Reilly & Associates, Inc. Sebastopol, CA, USA (1996)
4. Wikipedia: Diagrama de gantt (1910) http://es.wikipedia.org/wiki/Diagrama_de_Gantt.
5. Jain, R.: *The art of computer systems performance analysis: techniques for experimental design, measurement, simulation, and modeling*. Volume 491. Wiley New York (1991)
6. Lilja, D.: *Measuring Computer Performance*. Cambridge University Press United Kingdom; (2000)
7. Puigjaner, R., Serrano, J., Rubio, A.: *Evaluación y explotación de sistemas informáticos*. Síntesis (1993)
8. Cañas, A., Ortigosa, E., Aragón, Y.: La plataforma swad como recurso docente para la innovación educativa. In: *Congreso internacional sobre el profesorado ante el reto de las nuevas tecnologías en la sociedad del conocimiento*. (2005)

Desarrollo de un Entorno Integrado para un Computador Didáctico Elemental, para la Asignatura de Fundamentos de Informática del Nuevo Grado en Ingeniería de Tecnologías de Telecomunicación

H.Pomares, I.Rojas, A.Guillén, J.González, O.Valenzuela, J.P.Florido, J.Urquiza,
A.B.Cara, L.López-Mansilla, S.Egea-Navarro

Departamento de Arquitectura y Tecnología de Computadores. E.T.S. Ing. Informática y de
Telecomunicación, Universidad de Granada
Granada, España
hpomares@atc.ugr.es

Resumen. El objetivo principal de este trabajo es la potenciación y mejora de la metodología docente en la asignatura de Fundamentos de Informática, asignatura obligatoria correspondiente al nuevo Grado en Ingeniería de Tecnologías de Telecomunicación de la Universidad de Granada. Para ello, se ha desarrollado un entorno integrado didáctico que engloba, en una sola aplicación, tanto el simulador como el ensamblador de un Computador Didáctico Elemental desarrollado en nuestro departamento, CODE-2, y que, además, incorpora un programa editor y un ensamblador en línea con el que el alumno puede visualizar directamente el código máquina generado por el código ensamblador que está escribiendo en tiempo real. De esta forma, conseguimos facilitar que los alumnos puedan adquirir los conocimientos necesarios para comprender el funcionamiento interno de un computador.

Palabras Clave: Fundamentos de Informática; Ingeniería de Tecnologías de Telecomunicación; Computador Didáctico Elemental; CODE-2.

1 Introducción

En octubre de 2010 ha comenzado la implantación de los estudios del nuevo Grado en Ingeniería de Tecnologías de Telecomunicación por la Universidad de Granada. Como consecuencia de la visión de las necesidades del mercado laboral, y teniendo en cuenta el Libro Blanco de la titulación y los recursos humanos y materiales disponibles, se decidió proponer un único título de Graduado en Ingeniería de Tecnologías de Telecomunicación, con tres especialidades profesionales diferenciadas y elevado atractivo en el entorno socioeconómico, regional, nacional y europeo: 1. Sistemas de Telecomunicación, 2. Telemática, 3. Sistemas Electrónicos. Dentro de la materia básica *Fundamentos Tecnológicos y Empresariales*, común a todas las especialidades, se encuentra la asignatura *Fundamentos de Informática*.

Según la memoria de verificación de grado (disponible en <http://etsiit.ugr.es>), el objetivo principal de la asignatura es que el alumno adquiriera conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

2 Temario de la asignatura

Teniendo en cuenta todo esto, los responsables de la asignatura han decidido el siguiente temario para la misma:

- Tema 1.- Introducción a la Informática: Conceptos básicos y definiciones. Soporte lógico y soporte físico. Tipos de ordenadores. Tipos de software: herramientas informáticas de aplicación en Ingeniería.
- Tema 2.- Representación de la Información: Codificación. Representación de texto, datos numéricos, sonido e imágenes.
- Tema 3.- Estructura funcional de los ordenadores: Niveles conceptuales de descripción de un computador. Esquema de funcionamiento de un computador. Estructuras básicas de interconexión.
- Tema 4.- Elementos de programación: Tipos de lenguajes. Elementos básicos de los lenguajes de programación. El proceso de traducción y ejecución de programas. Fundamentos de algorítmica.
- Tema 5.- Fundamentos de Sistemas Operativos: Definición. Gestión del procesador, de la memoria y E/S. Seguridad.
- Tema 6.- Bases de Datos: Conceptos básicos. Bases de Datos Relacionales. Diseño de Bases de Datos. Sistemas Gestores de Bases de Datos.

Igualmente, se han propuesto, como complemento docente, los siguientes seminarios:

- Seminario 1: Estructura y montaje de un PC.
- Seminario 2: Instalación de un Sistema Operativo.
- Seminario 3: Búsqueda de información en Internet y en Bibliotecas Digitales.
- Seminario 4: Software Libre y Software Propietario Sistemas operativos y software específico para servidores.

Finalmente, la parte práctica de la asignatura está compuesta por las siguientes cuatro prácticas:

- Práctica 1: Uso del Sistema Operativo.
- Práctica 2: Funcionamiento a bajo nivel de un ordenador.
- Práctica 3: Herramientas informáticas con aplicación en Ingeniería.
- Práctica 4: Uso básico de un Sistema Gestor de Bases de Datos.

En las siguientes secciones, se presenta el entorno integrado que se ha desarrollado para la segunda de estas prácticas: *Funcionamiento a bajo nivel de un ordenador*.

3 El Computador Didáctico Elemental CODE-2

CODE-2 es una máquina de tipo von Neumann que contiene todas las unidades típicas de este modelo: entradas, salidas, unidad de control, unidad de procesamiento y memoria de datos e instrucciones. CODE-2 es de 16 bits de longitud de palabra y tiene tan sólo 16 instrucciones máquina.

A través del lenguaje máquina de CODE-2 tenemos acceso a los siguientes elementos:

- Banco de registros: compuesto por 16 registros (r0 .. rF). Todos son de propósito general, aunque el registro rE se suele utilizar como puntero de pila y rD como registro de dirección.
- Unidad aritmético lógica (ALU): podremos hacer sumas y restas en C2, la operación NAND y desplazamientos.
- Bistables indicadores: cero (Z), signo (S), acarreo (C) y desbordamiento (V), según la operación realizada en la ALU.
- Memoria principal: compuesta por 64Kpalabras de 16 bits (128KB).
- Puertos de entrada: se pueden usar hasta 256 puertos de entrada (desde el IP00 a IPFF).
- Puertos de salida: se pueden usar hasta 256 puertos de salida (desde el OP00 a OPFF).

En la siguiente tabla se describe el repertorio completo de instrucciones de este computador didáctico elemental.

Nombre	Nemónico	Parámetros	Explicación
Cargar	LD	rx,[v]	$rx \leftarrow M(rD+v)$
Almacenar	ST	[v],rx	$M(rD+v) \leftarrow rx$
Carga inmediata baja	LLI	rx,v	$rx(15:8) \leftarrow H'00;$ $rx(7:0) \leftarrow v$
Carga inmediata alta	LHI	rx,v	$rx(15:8) \leftarrow v$
Entrada	IN	rx,IPv	$rx \leftarrow IPv$
Salida	OUT	OPv,rx	$OPv \leftarrow rx$
Suma	ADDS	rx,rs,ra	$rx \leftarrow rs+ra$
Resta	SUBS	rx,rs,ra	$rx \leftarrow rs-ra$
NAND	NAND	rx,rs,ra	$rx \leftarrow (rs \cdot ra)'$
Desplaz. izqda	SHL	rx	$C \leftarrow rx(15), rx(i) \leftarrow rx(i-1), i=15, \dots, 1; rx(0) \leftarrow 0$
Desplaz. dcha	SHR	rx	$C \leftarrow rx(0), rx(i) \leftarrow rx(i+1), i=0, \dots, 14;$ $rx(15) \leftarrow 0$
Desplaz. aritmétic. dcha	SHRA	rx	$C \leftarrow rx(0), rx(i) \leftarrow rx(i+1), i=0, \dots, 14$
Salto	B-	cnd	Si cnd se cumple, $PC \leftarrow rD$
Subrutina	CALL-	cnd	Si cnd se cumple, $rE \leftarrow rE-1, M(rE) \leftarrow PC,$ $PC \leftarrow rD$
Retorno	RET	-	$PC \leftarrow M(rE); rE \leftarrow rE+1$
Parar	HALT	-	Parar

4. Descripción del entorno desarrollado

Antes del desarrollo del proyecto actual, se disponía del siguiente conjunto de herramientas para CODE-2:

- Un simulador básico de CODE-2.
- Un programa ensamblador que traduce un programa escrito en lenguaje ensamblador de CODE-2 a código máquina.

Para mejorar el potencial pedagógico de estas herramientas, nuestro equipo ha desarrollado, a partir de estas dos aplicaciones independientes, un entorno integrado didáctico que engloba, en una sola aplicación tanto el simulador como el ensamblador y que, además, incorpora un programa editor y un ensamblador en línea con el que el alumno pueda visualizar directamente el código máquina que está generando el código ensamblador que está escribiendo en tiempo real. De esta forma, el alumno sólo tendrá que utilizar una única aplicación en sus prácticas de la asignaturas que hagan uso de dicho Computador Didáctico Elemental.

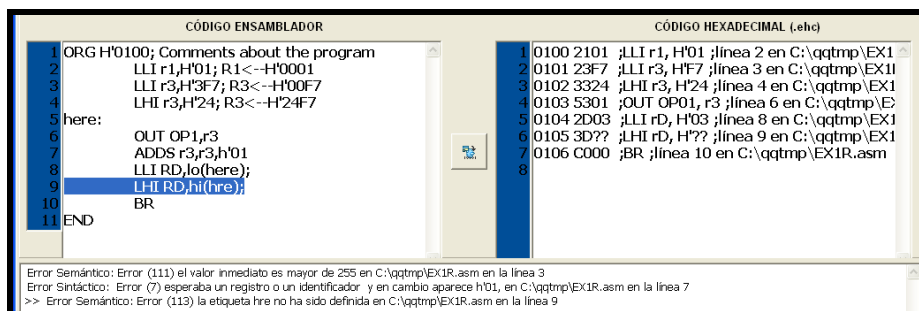


Figura 1. Ensamblador en línea dentro del entorno integrado desarrollado.

En la Figura 1 se muestra el aspecto del ensamblador en línea que se ha desarrollado como primer elemento del entorno integrado. Esta herramienta está compuesta por:

- a) Un editor (panel izquierdo) en el que el estudiante escribe sus programas mediante lenguaje ensamblador de CODE-2.
- b) Una ventana de código máquina (panel derecho) donde se traduce cada instrucción en lenguaje ensamblador a código máquina. Esta traducción se realiza en tiempo real cada vez que el alumno escribe una nueva instrucción. De esta forma, el estudiante puede comprobar cómo se va generando el programa final que debe cargarse en la memoria de CODE-2.
- c) Una ventana de mensajes de error (panel inferior). Conforme el alumno va escribiendo su programa, en esta ventana van apareciendo los diversos mensajes de error (si los hubiere) existentes en el mismo. Con un simple doble clic, el cursor se posiciona automáticamente en la línea de código que ha generado cada error.

Junto con el ensamblador en tiempo real, hemos desarrollado un multi-editor (Figura 2) con el que los estudiantes pueden abrir, modificar y guardar diversos ficheros tanto en formato ensamblador como en código máquina. El editor está equipado con las siguientes herramientas:

- Un conjunto de pestañas para poder navegar entre los diferentes ficheros abiertos.
- Una barra de estado con el nombre del fichero actualmente seleccionado, la línea y columna del cursor, y el número total de líneas del fichero.
- Cada fichero puede ser abierto, editado, imprimido y cerrado.
- Se pueden modificar las fuentes y su tamaño.
- Se puede copiar, cortar, pegar, seleccionar todo, deshacer, rehacer, buscar y reemplazar.
- Se pueden ensamblar cada fichero directamente desde el editor. En ese caso, aparece una ventana adicional (ver Figura 2) donde se indican los mensajes de error de igual forma que en el caso del ensamblador en línea.

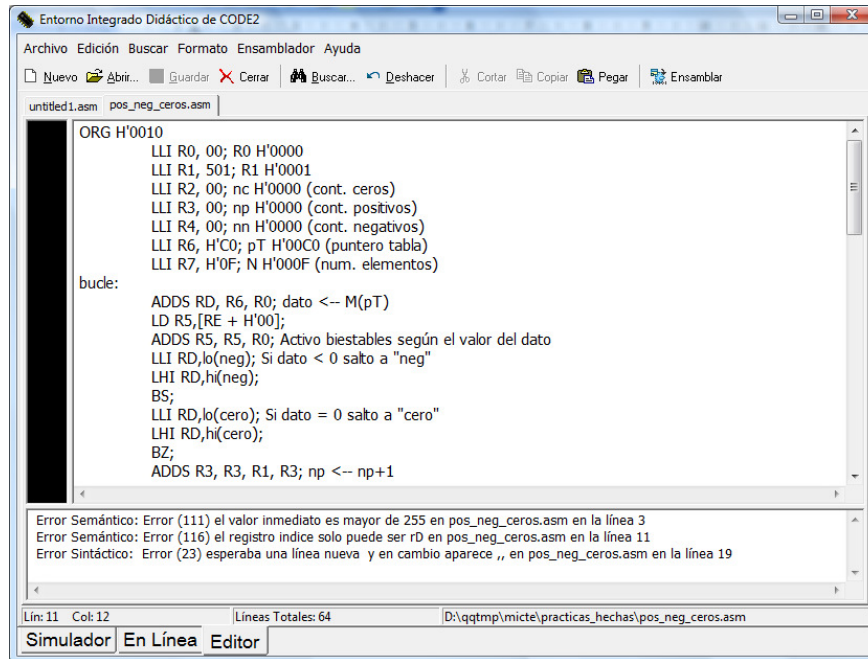


Figura 2. Multi-editor para la edición de programas para CODE-2

Finalmente, en la Figura 3 se muestra el aspecto del simulador de CODE-2 ya integrado en el entorno desarrollado. A través del simulador, los alumnos pueden cargar los programas ensamblados mediante cualesquiera de los editores anteriores y ejecutarlos paso a paso, comprobando en cada momento el efecto de cada instrucción ejecutada.

Por lo tanto, podemos resumir la labor realizada diciendo que hemos desarrollado una única herramienta didáctica que consta de:

- Un programa simulador completo de CODE-2, que permite visualizar fácilmente los contenidos de todos los registros, posiciones de memoria y puertos de E/S.
- Un editor de programas en ensamblador para CODE-2.
- Un ensamblador de CODE-2 integrado en la aplicación, de modo que se permite, con un simple clic del ratón editar la línea que ha generado cada uno de los códigos de error del ensamblador.
- Un programa intérprete que ensambla en tiempo real el programa que va escribiendo el alumno, de tal forma que le permite observar el código máquina generado cada vez que se escribe una línea de código.

De entre todas las características de la aplicación informática desarrollada, nos gustaría destacar su fiabilidad, robustez, facilidad de uso, y apariencia estética. Atributos que se han considerado indispensables para que el alumno pueda trabajar cómodamente con la herramienta, le sirva de ayuda, y no la deje de lado por su complejidad, por contener errores o cualquier otro motivo.

Finalmente, consideramos que la herramienta desarrollada constituye una mejora significativa en el desarrollo tanto de la parte teórica como de las prácticas de laboratorio, ya que con ella los alumnos podrán corroborar la correcta realización de los programas que deban escribir para dicho computador didáctico y, en caso de haber cometido un fallo, podrán de forma muy sencilla encontrar la fuente del mismo.

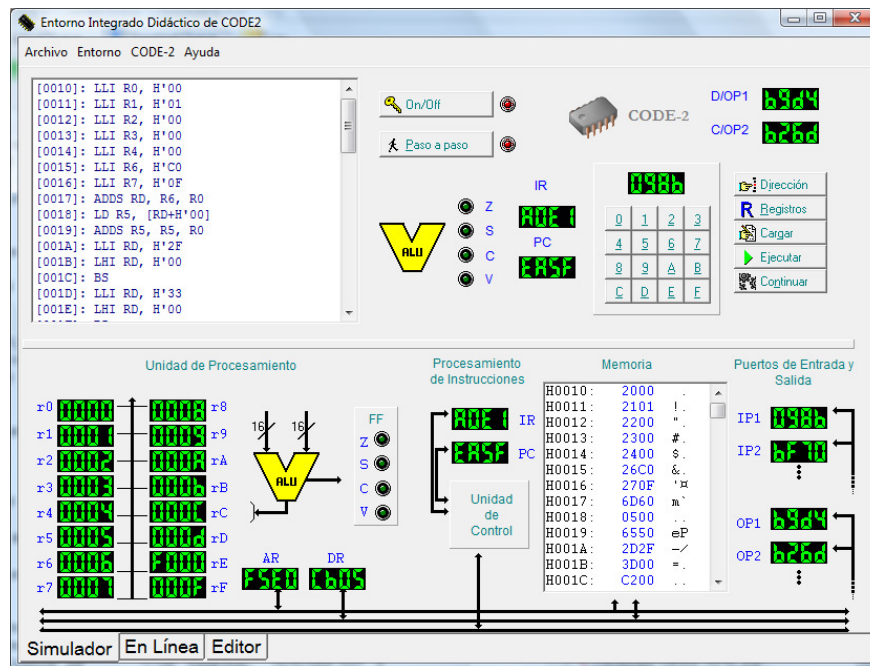


Figura 3. Simulador de CODE-2 dentro del entorno integrado desarrollado

4. Conclusiones

En este trabajo se ha presentado una herramienta para facilitar el aprendizaje del funcionamiento de un computador, dirigida a alumnos de la asignatura de Fundamentos de la Informática, asignatura obligatoria correspondiente al nuevo Grado en Ingeniería de Tecnologías de Telecomunicación de la Universidad de Granada. Dicha herramienta está compuesta por un ensamblador en línea para CODE-2, un multi-editor y un simulador de CODE-2, todo ello integrado en un mismo entorno. Esta herramienta está libremente disponible a través de Internet para todos los alumnos de la Universidad de Granada y de otras universidades. Consideramos que la herramienta diseñada constituye una mejora significativa en el desarrollo tanto de la parte teórica como de las prácticas de laboratorio, ya que con ella los alumnos podrán comprobar la correcta realización de los programas que deban escribir para dicho computador didáctico y, en caso de haber cometido un fallo, podrán de forma muy sencilla encontrar la fuente del mismo.

Agradecimientos. Este trabajo ha sido subvencionado parcialmente por el proyecto de innovación docente 10-22 de la Universidad de Granada.

Referencias

1. A. Prieto and A. Lloris, Introducción a la Informática, 3th edition, McGraw Hill, 2004.
2. J.Díaz, Prototipo hardware de CODE-2, proyecto fin de carrera, Universidad de Granada, 2002.
3. C.Hamacher, Z.Vranesic and S.Zaky, Organización de Computadores, McGraw Hill, 3th edition, 2000.
4. W.Stallings, Computer Organization & Architecture: Designing for Performance, Prentice-Hall, 6th edition, 2003.
5. H.Pomares et al, Novel Assembler to Facilitate the Understanding of How Processors Work, Current Developments in Technology-Assisted Education, Seville, Spain, 2006.

Montaje de los Componentes de un Servidor para la Asignatura del Nuevo Grado en Ingeniería en Informática: Ingeniería de Servidores

F.Rojas, H. Pomares, A.Guillén, L.J. Herrera

Departamento de Arquitectura y Tecnología de Computadores. E.T.S. Ing. Informática y de Telecomunicación, Universidad de Granada
Granada, España
hpomares@atc.ugr.es

Resumen. En este trabajo, se presenta una visión general de la nueva asignatura Ingeniería de Servidores, del nuevo plan de estudios del Grado en Ingeniería Informática de la Universidad de Granada, así como una nueva metodología interactiva para que el alumno aprenda a montar un servidor de gama baja. A través de este aprendizaje práctico, que el Espacio Europeo de Educación Superior promueve activamente, tratamos de que el alumno descubra cómo asociar la arquitectura de un servidor con los componentes de los computadores con los que ellos trabajan a diario.

Palabras Clave: Ingeniería de Servidores; Montaje de un servidor de gama baja; Componentes de un servidor de gama baja.

1 Introducción

En octubre de 2010 ha comenzado la implantación de los estudios del nuevo Grado en Ingeniería Informática por la Universidad de Granada. Como consecuencia de la visión de las necesidades del mercado laboral, y teniendo en cuenta el Libro Blanco de la titulación y los recursos humanos y materiales disponibles, se decidió proponer un único título de Graduado en Ingeniería Informática, con cinco especialidades profesionales diferenciadas y elevado atractivo en el entorno socioeconómico, regional, nacional y europeo: 1. Computación y Sistemas Inteligentes, 2. Ingeniería de Software, 3. Ingeniería de Computadores, 4. Sistemas de Información, 5. Tecnologías de la Información. Una de las asignaturas obligatorias de rama, es decir, asignatura que deben cursar todos los alumnos que quieran obtener el título de graduado en ingeniería informática por la Universidad de Granada, independientemente de la especialidad, es la asignatura *Ingeniería de Servidores*.

Ingeniería de Servidores tiene como objetivo principal que el alumno sepa *diseñar, instalar y configurar un servidor de gama baja a partir de las necesidades de un cliente*. Según la memoria de verificación de grado (disponible en <http://etsiit.ugr.es>), sus contenidos generales son los siguientes: *Componentes de un servidor. Diseño y configuración de un servidor de gama baja. Almacenamiento. Montaje e instalación.*

Administración y evaluación de prestaciones de un servidor. En cuanto a los distintos niveles de profundización, capacidades, competencias y destrezas que debe adquirir un alumno tras la superación de la asignatura, a continuación se detallan las que son específicas a la asignatura:

- 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 planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.
- Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.
- Conocimiento, administración y mantenimiento de sistemas, servicios y aplicaciones informáticas.
- Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.

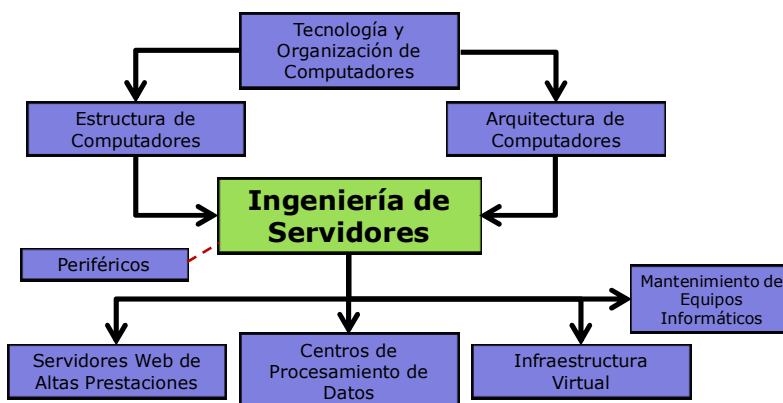


Figura 1. Relación de la asignatura con otras asignaturas de la titulación

2 Temario de la asignatura

Teniendo en cuenta todo esto, y los contenidos adquiridos por los estudiantes en otras asignaturas relacionadas (ver Figura 1), los responsables de la asignatura hemos decidido el siguiente temario para la misma:

- Tema 1.- Introducción a la Ingeniería de Servidores: Clasificación de computadores utilizados comercialmente. Concepto de servidor. Niveles de empaquetamiento/conexión en un servidor. Identificación de componentes en los niveles de tarjeta, placa y chasis.
- Tema 2.- Microprocesadores para Servidores de Gama Baja: Historia de los microprocesadores. Caracterización de un microprocesador. Evaluación de prestaciones de un microprocesador. Análisis de microprocesadores de gama baja actuales.
- Tema 3.- Memorias: Tipos de memoria. Tecnologías de memoria actuales. Evaluación de prestaciones de una memoria.
- Tema 4.- Buses y Entrada/Salida: Tipos de buses. Jerarquía de buses. Medidas de rendimiento de un bus.
- Tema 5.- Almacenamiento: Tipos de dispositivos de almacenamiento permanente. Configuración de dispositivos de almacenamiento. Evaluación de prestaciones.
- Tema 6.- Placas Base y Chasis: Estructura de una placa base. Chipsets y placas base comerciales. Características adicionales: factores de forma y fuentes de alimentación. Configuración de la placa. Elección de la caja/chasis
- Tema 7.- Diseño y Configuración de Servidores: Metodologías de diseño. Criterios de selección. Evaluación de prestaciones de un servidor. Montaje, instalación y configuración: casos de estudio. Introducción a los servidores de gama media y alta.

Igualmente, se han propuesto, como complemento docente, los siguientes seminarios:

- Seminario 1: Sistemas operativos y software específico para servidores.
- Seminario 2: Consejos prácticos para la administración de un servidor.
- Seminario 3: Planteamiento de un pliego de condiciones técnicas para el diseño de un servidor de acuerdo a la normativa.

Finalmente, la parte práctica de la asignatura está compuesta por las siguientes cuatro prácticas:

- Práctica 1: Montaje de los Componentes de un Servidor.
- Práctica 2: Instalación Software y Evaluación de Prestaciones de un Servidor.
- Práctica 3: Administración de un Servidor.
- Práctica 4: Optimización de Prestaciones de un Servidor.

En las siguientes secciones, se presentará de forma más específica la metodología prevista para el desarrollo de la primera de estas prácticas.

3 Metodología propuesta

En la sesión práctica se pondrán de manifiesto los componentes reales de un servidor de gama baja real de la forma más práctica posible: ensamblando nosotros

mismos los elementos del servidor. La planificación de la sesión se divide en los siguientes bloques:

- Mostrar las diferentes partes del servidor tal como se suministrarían por el fabricante.
- Informar sobre las normas de seguridad al manipular dispositivos eléctricos y electrónicos.
- Comenzar el ensamblaje del servidor invitando a los estudiantes a participar, proporcionando soluciones, opiniones, alternativas, etc.
- Una vez finalizado el montaje del servidor, comprobar su correcto funcionamiento.

4. Desarrollo de la práctica

Para favorecer la visibilidad de los componentes una vez ensamblados, se recomienda adquirir una carcasa de metacrilato transparente. Antes de comenzar con el ensamblado del servidor, es importante que los estudiantes sean conscientes de las normas básicas de seguridad al tratar con elementos eléctricos y electrónicos:

- Asegurarse de que todos los dispositivos estén completamente desenchufados de cualquier fuente de alimentación.
- Antes de tocar cualquier tarjeta o la placa base, el individuo debe descargar su electricidad estática. Dicha energía puede dañar algunos componentes delicados (tales como memorias, circuitos, etc.). Para descargar su electricidad estática, basta con estar en contacto con algún elemento conductor que actúe como tierra o masa, a través de una pulsera conductora o por contacto directo.
- Emplear un lugar cómodo para el ensamblaje: debe disponer de suficiente iluminación y espacio, así como de las herramientas adecuadas (destornillador con punta magnética americana y pinzas de precisión).

Durante la realización de la práctica, los principales pasos a seguir serán los siguientes:

- a) Desembalaje de los componentes del servidor: evidentemente, la primera parte del proceso debe ser desembalar los componentes que formarán el servidor. Los estudiantes deben prestar atención para guardar los manuales y las garantías de los productos.
- b) Montaje de la placa base: La placa base se distingue bajo varios nombres en computación: placa base, placa principal, placa madre. En ella se insertan (entre otros): el procesador, los módulos de memoria, las tarjetas de control y expansión y los buses de comunicación de las distintas unidades. En este punto, es importante mostrar a los estudiantes el esquema general de la placa base, identificando cada una de las ranuras de inserción de los componentes más representativos y chips integrados. Este esquema se encuentra normalmente en la documentación de la placa base (ver Figura 2).

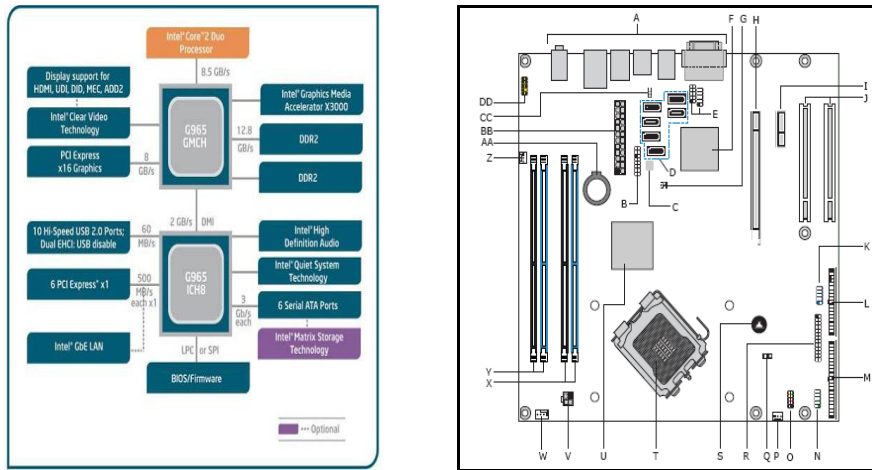


Figura 2. Chipset Intel® Q965 Express y principales componentes de la placa base Intel® DQ965CO

- c) Montaje del procesador: El procesador es el elemento central de un computador, puesto que es el dispositivo que interpreta las instrucciones máquina. Para instalar el procesador en la placa base procedemos de la siguiente forma: 1) Localizar el zócalo en que se debe insertar en la placa base. 2) Desbloquearlo levantando la palanca en el lateral. 3) Colocar la CPU sobre el zócalo de forma que la esquina marcada coincida con la base de la palanca del zócalo. 4) Empujar hacia abajo la palanca para fijar la CPU. 5) Fijar el disipador a la placa base. Se suele usar cola térmica para favorecer la disipación de calor. 6) Conectar el cable de alimentación del disipador a la placa base.
- d) Montaje de la fuente de alimentación: La fuente de alimentación provee alimentación a la placa base (2 conectores de 20 pines y 4 pines, 12V), a las unidades de disco duro y CD (12V) y disqueteras (5V). La instalación consiste en atornillar la fuente a la carcasa y fijar las conexiones a la placa base y unidades. En este punto, los estudiantes deben reflexionar sobre la potencia necesaria para el correcto funcionamiento del servidor, en función del número y requerimientos de potencia de cada dispositivo.
- e) Montaje de la tarjeta gráfica: Algunas placas base proporcionan una tarjeta gráfica integrada. Si este no es el caso, o si se prefiere disponer de una tarjeta independiente con mejores prestaciones, se debe instalar a través del bus correspondiente (AGP, Accelerated Graphics Port; PCI- Express o cualquier estándar futuro).
- f) Montaje de los módulos de memoria: Las memorias DIMM (Dual In-line Memory Module) son de 168 contactos. DDR (Double Data Rate) es un tipo de memoria síncrona RAM (SDRAM), que permite la transmisión de datos por dos canales distintos en un mismo ciclo de reloj. Se deben mostrar a los alumnos y comparar los diferentes tipos de memorias volátiles SDRAM (SIMM, DIMM), y

RDRAM (RIMM). La instalación del módulo es sencilla, aunque los estudiantes deben asegurarse de que el módulo esté completamente instalado en la ranura.

- g) Montaje de los dispositivos de almacenamiento masivo: el disco duro y otros dispositivos ópticos (si son IDE o S-ATA) tienen un montaje similar. 1) Seleccionamos la ranura de la carcasa en la que lo vamos a fijar. 2) Lo atornillamos (4 tornillos, 2 por lateral) a la carcasa. 3) Conectamos el cable de alimentación que proviene de la fuente (12V) al disco. 4) Conectamos el disco al conector correspondiente.
- h) Finalización del montaje: atornillado de la caja, monitor, teclado y ratón. Finalmente, tras instalar los componentes internos, se cierra y atornilla la caja. Se conectan los dispositivos externos: monitor TFT (conexión VGA al servidor y alimentación a la corriente eléctrica), teclado (puerto PS2) y ratón (puerto USB o PS2).

En la Figura 3 se muestra una secuencia de imágenes que resume el proceso seguido en todas estas etapas durante la práctica.

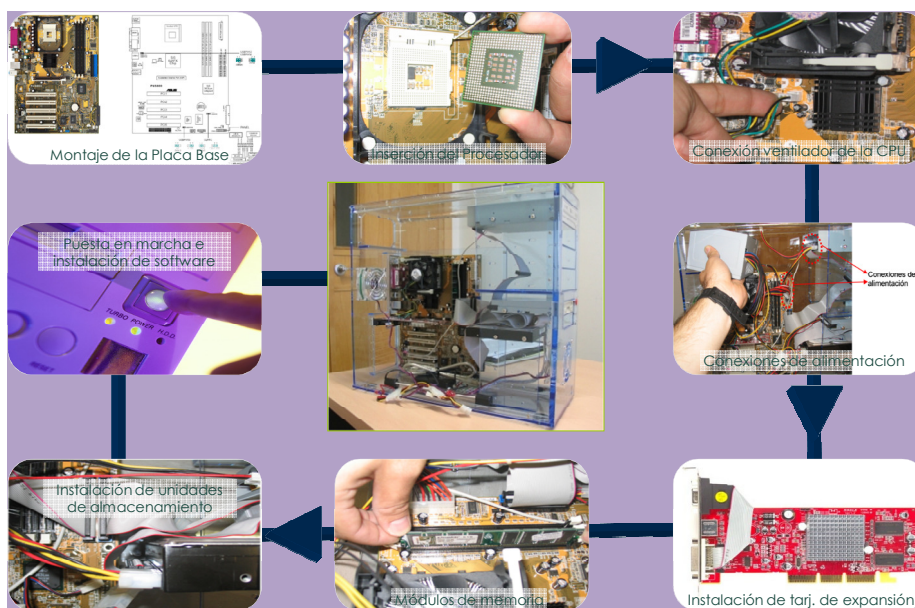


Figura 3. Secuencia de etapas a seguir para el montaje del servidor.

4. Evaluación de la práctica

Una vez finalizado el montaje del servidor, se procede a conectarlo a la corriente eléctrica y se pulsa el botón de encendido. En este punto, si no se había instalado previamente un sistema operativo en el disco duro, se debe instalar uno. Por razones de limitación de tiempo, esta sección no se lleva a cabo en esta sesión práctica sino en

uno de los seminarios posteriores en los que se comprobarán los componentes instalados mediante software especializado (ver Figura 4). Las tareas que el profesor debe evaluar en esta práctica son:

- El interés y la habilidad del alumno en el montaje del servidor.
- La capacidad del alumno para utilizar los recursos electrónicos para encontrar una alternativa al servidor montado en prácticas con un presupuesto ajustado.

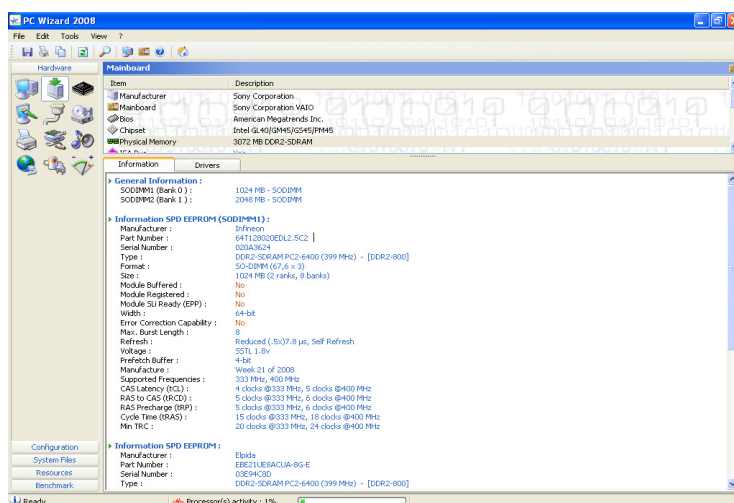


Figura 4. Programa de ejemplo para comprobar los componentes instalados

5. Conclusión

El objetivo principal de este trabajo es doble. Por un lado se ha presentado una visión general de la nueva asignatura *Ingeniería de Servidores*, del nuevo plan de estudios del Grado en Ingeniería Informática de la Universidad de Granada. Por otro lado, dentro del marco de dicha asignatura, se ha presentado la propuesta de una práctica para que el alumno realice un proceso de aprendizaje activo a través de la participación interactiva en el diseño y montaje de un servidor, así como un análisis crítico de las características y eficiencia del mismo.

Referencias

1. M. Ujaldón. Arquitectura del PC. Ciencia-3. 2003.
2. Messmer, Hans-Peter, The Indispensable PC HARDWARE book, Addison-Wesley.
3. W. Stallings, Computer organization and architecture: designing for performance, 6th Ed., Pearson Education, 2003.
4. S.Mueller, Manual de actualización y reparación de PCs, Prentice Hall, 12ª Edición, 2001.

Proyecto Cluster

Javier Fernández¹, Mancia Anguita¹

¹ Depto. Arquitectura y Tecnología de Computadores.
ETS Ingenierías Informática y de Telecomunicación
Universidad de Granada
C/ Periodista Daniel Saucedo Aranda s/n, 18071-Granada
Granada, España
{jfernand, manguita}@ugr.es

Resumen. En el marco de un Proyecto de Innovación Docente de la Universidad de Granada, aplicado a la asignatura de 5º curso “Arquitectura de Computadores IP” (ACII) en el curso 2008-09, realizamos el diseño, compra, montaje, instalación y utilización de un mini-cluster de computadores orientado a aplicaciones científicas. En el curso 2009-10, disponiendo ya del equipo montado, la actividad se ha replanteado como instalación y utilización del cluster, habiéndose concurrido al Programa de Apoyo a Docencia Práctica para la adquisición de hardware adicional con el cual explotar las prestaciones del cluster (discos adicionales para servidor de disco, infraestructura Infiniband, etc). Esta contribución describe el método aplicado para desarrollar y evaluar la actividad, el cluster diseñado por los estudiantes y su configuración software, así como los resultados de encuestas de opinión de los estudiantes que participaron, posibles variaciones de la experiencia que podrían probarse, y las conclusiones alcanzadas con la realización del proyecto.

Palabras Clave: Cluster, Linux, Beowulf, aprendizaje por proyectos.

1 Introducción

El Aprendizaje por Proyectos [1] es un método docente ampliamente reconocido por su capacidad para estimular la participación de los estudiantes y mantenerlos motivados hacia la asignatura. Como características básicas distintivas del método se pueden destacar el menor énfasis en la enseñanza mecánica y memorística para dedicarse a un trabajo más retador y complejo, el enfoque interdisciplinario (apropiado a un trabajo complejo) en lugar de orientado al área o asignatura, y el trabajo cooperativo frente al individual.

Adoptar el método puro de Aprendizaje basado en Proyectos implicaría una fuerte coordinación entre el profesorado de las distintas asignaturas implicadas [2,3,4], y en último término una reforma completa en la estrategia docente de los estudios. En este trabajo hemos optado por introducir un pequeño proyecto voluntario en la asignatura ACII, con la intención obvia de mejorar la motivación de los estudiantes, y en segundo término para observar la actitud de los estudiantes ante lo que podría ser una práctica de las nuevas asignaturas en los próximos planes de estudio de los Grados.

2 El proyecto y su forma de evaluación

En el contexto del Plan de Innovación Docente de la UGR en convocatoria 2008-09 [5], se nos concedió un presupuesto de 3.650€ que se puso a disposición de los estudiantes para diseñar, comprar, montar, instalar, evaluar y utilizar un pequeño cluster para la ejecución de aplicaciones científicas. Esta finalidad se justifica por el tipo de prácticas que se vienen realizando en la asignatura ACII (integración numérica, tratamiento de imágenes, etc.).

El enunciado de la actividad se proporcionó anticipadamente a los estudiantes a través de la web de los profesores [6]. Básicamente, se proponía diseñar el cluster en un plazo de 5 semanas (acabando antes de Semana Santa), para posteriormente instalarle el S.O., pasarle benchmarks (para comprobar si las prestaciones eran las esperadas según el diseño) y que diera tiempo a ejecutar en él las prácticas de la asignatura antes de que acabara el curso. Las normas de evaluación de la asignatura se alteraron de la forma indicada en la Tabla 1. Las ventajas para los estudiantes que participaran eran un menor umbral para aprobar, y un tope de nota algo mayor (11p).

Tabla 1. Puntuación de la asignatura ACII en el curso 2008-09.

Actividad	Resto de la clase		Grupo Cluster	
	máximo	umbral	máximo	umbral
Examen final	7,0	3,5	5,0	1,5
Prácticas	3,0	1,5	3,0	1,0
Proyecto Cluster			3,0	1,0
Suma	10,0	5,0	11,0	3,5

Se decidió añadir 3p adicionales, por el trabajo realizado en el Proyecto Cluster (2p) y un cuestionario final sobre el mismo (1p), mantener los 3p de prácticas, y prorratear a 5p el examen de teoría. Obsérvese que se podrían obtener 11p, y que se rebajaron los umbrales de aprobado al 30% (frente al 50% “normal”), pudiéndose aprobar holgadamente con 1.5p en el examen final de Teoría.

Las encuestas del curso **2008-09** nos han permitido detectar la predilección de los estudiantes por los umbrales bajos, no por los prorrateos, de forma que en el curso **2009-10** hemos preferido plantear el proyecto (instalación, configuración, evaluación y explotación del cluster) como alternativa a las prácticas normales, con los mismos máximos y umbrales (Tabla 2). La alternativa se denominó “Centro de Proceso de Datos”, dando a entender que esta actividad u otras similares podrían practicarse en la futura asignatura de Grado de dicho nombre.

Tabla 2. Puntuación de la asignatura ACII en el curso 2009-10.

Actividad	máximo	umbral								
Examen final	7,0	3,0								
<table style="display: inline-table; vertical-align: middle;"> <tr> <td rowspan="2" style="font-size: 2em; vertical-align: middle;">{</td> <td>Test</td> <td>3,5</td> <td rowspan="2" style="font-size: 2em; vertical-align: middle;">{</td> <td>0,7</td> </tr> <tr> <td>Ejercicios</td> <td>3,5</td> <td>-</td> </tr> </table>	{	Test	3,5	{	0,7	Ejercicios	3,5	-		
{		Test	3,5		{	0,7				
	Ejercicios	3,5	-							
Prácticas	3,0	1,25								
Suma	10,0	4,25								

(a)

Foros de discusión

Todos los foros Foros de este sitio
Arquit. Comp. 1 [5 hilos, 348 mensajes]

Pág 1/21

Asunto	Primer mensaje	Último mensaje	Nº mens.	No Escritos	Leídas
Dinámica de trabajo, evolución del proyecto y normas de foro	Javier 01.03.09 Fernand... 12.02	Javier 22.06.09 Fernand... 19.20	10	0	3 82
Empate en materia del examen del Proyecto Cluster [Pág. 1 2]	Javier 10.06.09 Fernand... 30.03	José Luis 20.06.09 Trapano... 20.30	19	0	11 40
Propuesta configuración de los servidores [Pág. 1 2 ... 5 ... 8]	Luis 01.03.09 Queredo... 22.30	Javier 07.05.09 Fernand... 19.38	28	0	15 61
Benchmarks [Pág. 1 2 3]	Ignacio 02.03.09 Robles... 19.47	Javier 03.05.09 Fernand... 20.59	22	0	9 50
Elección del procesador [Pág. 1 2 ... 4 ... 8]	José Al... 02.03.09 Jiménez... 13.14	Javier 14.04.09 Fernand... 12.07	34	0	17 50
Utilización de CPUs para procesamiento [Pág. 1 2 3 4]	Francis 04.03.09 García Roa 14.20	Javier 12.04.09 Fernand... 23.00	34	0	11 41
Sistemas Operativos [Pág. 1 2 3 ... 8]	Ignacio 02.03.09 Robles... 12.49	Paula 12.04.09 Ureña D... 17-12	47	0	13 50
Comunicación entre máquinas [Pág. 1 2]	José Al... 03.03.09 Jiménez... 14.36	Javier 08.04.09 Fernand... 10.37	13	0	11 53
Otros factores a tener en cuenta	Isidro 02.03.09 López Cos 16.30	Javier 08.04.09 Fernand... 10.36	9	0	7 56
Red de interconexión [Pág. 1 2]	Manuel 06.03.09 Martín... 22.01	Javier 07.04.09 Fernand... 21-17	11	0	9 32

Pág 1/21

(b)

Zona común (grupo: Trabajo Cluster)
(accesible para lectura por alumnos del grupo y profesores de la asignatura)
3 niveles; 12 carpetas; 38 archivos; 29,2 MiB (2,8% de 1,0 GiB)

Nombre	Fecha	Tamaño	Usuario
comun	10.06.09 21:16	4,0 KB	
ClusterAC2.pdf	16.04.09 13:13	292,3 KB	Lopez Cos, Isidro
Diseño_final.pdf	16.04.09 11:35	2,2 MB	Orantes Pozo, Pablo
fotos	16.05.09 00:51	4,0 KB	Martín Salvador, Manuel
Lista compra	11.05.09 11:02	4,0 KB	Fernández Ba..., Javier
meetings	22.05.09 13:14	4,0 KB	Fernández Ba..., Javier
Trabajos	15.06.09 19:16	4,0 KB	Fernández Ba..., Javier
Francisco Joaquín	15.06.09 19:21	4,0 KB	García Roa, Francis...
benchmarking en windows.pdf	15.06.09 19:21	592,3 KB	García Roa, Francis...
Ignacio	10.06.09 13:38	4,0 KB	Robles Páiz, Ignacio
HPL.pdf	10.06.09 13:34	266,9 KB	Robles Páiz, Ignacio
maui-admin.pdf	10.06.09 13:38	226,2 KB	Robles Páiz, Ignacio
maui-instalacion.pdf	10.06.09 13:34	290,0 KB	Robles Páiz, Ignacio
Jose Maria	10.06.09 14:36	4,0 KB	González Guiles, José María
skum.pdf	10.06.09 14:36	81,3 KB	González Guiles, José María
Luis	10.06.09 13:44	4,0 KB	Quesada Torres, Luis
livecd.pdf	10.06.09 13:44	104,1 KB	Quesada Torres, Luis
Manuel	09.06.09 19:06	4,0 KB	Martín Salvador, Manuel
gcc_icc_mpi.pdf	09.06.09 19:06	168,7 KB	Martín Salvador, Manuel
Pablo	10.06.09 21:09	4,0 KB	Orantes Pozo, Pablo
C3Tools.pdf	10.06.09 21:06	271,5 KB	Orantes Pozo, Pablo
Presentacion.ppt	10.06.09 21:09	2,2 MB	Orantes Pozo, Pablo
sge-tokyo.pdf	10.06.09 21:08	1,9 MB	Orantes Pozo, Pablo
Systemimager.pdf	10.06.09 21:06	291,9 KB	Orantes Pozo, Pablo
Paula Urena	14.06.09 23:33	4,0 KB	Ureña Delgado, Paula
FEDORA CORE 10.pdf	14.06.09 23:33	43,5 KB	Ureña Delgado, Paula
Victor	09.06.09 12:56	4,0 KB	Corcoiba Magaña, Victor
Benchmark mpiBlast.pdf	09.06.09 12:56	35,1 KB	Corcoiba Magaña, Victor
Guia de instalacion de SGE.pdf	09.06.09 12:53	145,4 KB	Corcoiba Magaña, Victor
instalacionMpiBlast.pdf	09.06.09 12:55	61,5 KB	Corcoiba Magaña, Victor
Manual de usuario de SGE.pdf	09.06.09 12:54	74,1 KB	Corcoiba Magaña, Victor

Figura 1. Foro SWAD usado para el diseño, y Zona Común de trabajos (2008-09).

Para supervisar el desarrollo de la actividad se ha utilizado el sistema SWAD de la UGR [7,8,9], aprovechando el foro para comunicación profesores-estudiantes, y la zona común de archivos para entregar documentos de trabajo (Figura 1). La web de la actividad [6] sugería unas cuantas actividades que podían realizar autónomamente los estudiantes (Tabla 3), apuntaba algunas webs de fabricantes y vendedores de donde obtener información para realizar los diseños, y ha servido también para ir anotando comentarios y resumiendo las aportaciones realizadas a lo largo del proyecto.

Tabla 3. Posibles actividades del proyecto (curso 2008-09)

Actividad	puntos
Aportación de componente, precio y características técnicas (en el foro)	0,1 - 0,3
Rebatir o comentar aportación previa con precios, detalles técnicos o conceptos de clase	0,1 - 0,3
Encontrar defectos de diseño en aportaciones previas	0,1 - 0,2
Encontrar diseño similar al considerado, y comparar precios	0,1 - 0,2
Encontrar relaciones entre documentación componentes y conceptos estudiados en clase	0,1 - 0,2
Documentar la instalación / configuración de un software	0,3
Tarea de instalación / configuración de software	0,3
Otras	según

La experiencia del curso **2008-09** ha permitido aquilatar de forma más precisa la cantidad de trabajo que los estudiantes consideran apropiada para 3,0p de calificación, de forma que en el curso **2009-10** se planteaba únicamente la instalación, configuración, evaluación y explotación del cluster ya montado. Las actividades se realizaban presencialmente en horario de prácticas y tutorías, sobre el cluster real, anotando los profesores las tareas en las que cada estudiante participaba activamente. Los estudiantes han preferido realizar su informe [10] como documento GoogleDocs, también enlazado en la web del Proyecto Cluster [6].

3 Desarrollo del proyecto

Al tratarse de una actividad voluntaria adicional (curso 2008-09) o alternativa (2009-10) a las prácticas normales, sólo una pequeña fracción de los estudiantes se decanta por escogerla, razonando que implica realizar más trabajo que las prácticas normales para obtener la misma nota. En el curso 2008-09 participaron 15 estudiantes de un total de 150 (10%), y en el 2009-10, 7 de un total de 134 (5%).

Como incidencias destacables, hay que comentar que en el curso 2008-09 la etapa de diseño colaborativo en el foro no convergió a un diseño consensuado, por lo que hubo que reunir a los estudiantes, agruparlos en 2 grupos (7-8 estudiantes) y encargarles que realizaran el diseño por escrito durante Semana Santa para cumplir los plazos. Paradójicamente, los dos diseños entregados fueron virtualmente idénticos. La etapa de instalación de software estuvo plagada de incidencias, lo cual sirvió de argumento para dedicarle mayor atención (exclusiva, de hecho) a esta etapa en el curso siguiente.

En el curso 2009-10 se procuraron redactar las instrucciones de participación de forma que no llevaran a confusión a los estudiantes, resaltando que el proyecto consistía exclusivamente en instalar, configurar y evaluar el cluster, y administrarlo durante el periodo de explotación (las últimas semanas del curso) para que los restantes compañeros pudieran ejecutar en él sus prácticas. Como ya se ha indicado, las actividades se han realizado presencialmente, anotando el profesor la participación de cada estudiante. Aparentemente no ha habido incidencias, aunque a posteriori algunos estudiantes han mostrado discrepancias con el desarrollo del proyecto en la encuesta anónima que se les ha realizado (no durante las sesiones presenciales).

4 Cluster diseñado por los estudiantes

En el curso 2008-09 se realizó una reunión tras los diseños de Semana Santa para pulir algunos detalles, tras la cual se realizaron los siguientes encargos de compra:

Tabla 4. Componentes del cluster ACII.

Primer encargo		Segundo encargo	
componente	precio	componente	precio
3x Intel Core i7-920	734,97	3x Guías laterales telescópicas	91,11
3x ASUS P6T SE	662,31	1x Panel de parcheo 24x RJ-45	63,28
3x 6GB DDR3 1066 KVR-N7K3	284,97	1x Herramienta Impacto	27,42
1x Switch GbE TL-SG1016	76,99	4x cable parcheo 1m	4,84
1x Tarjeta GbE D-Link DGE-530T	18,99	1x bolsa tortillería M6	26,26
4x cable RJ-45 Cat6 3m	15,08	1x Termostato Digital 1U	137,47
1x KVM Level-One KVM-0410	89,00	1x Regleta 19" 6 tomas	49,96
4x cable KVM PS/2 1.8m	16,00	2x Pasacable 1U cepillo/peine	50,34
3x HDD 500GB SATA-II	143,97	8x Anilla guiacable 1U horizontal	31,68
1x Grab. DVD SATA	24,44	8x Anilla guiacable 1U vertical	31,68
2x Lector DVD SATA	27,84	1x Cable TwisTies 30m	2,96
3x Fuente Alimentación 500W	84,87		
3x VGA PCIe GF7200 256MB	76,77		
1x Monitor LCD 19" LG W1941S	99,99		
1x Teclado Logitech PS/2	7,54		
1x Ratón Logitech óptico PS/2	5,40		
1x Armario 19" 22U	434,90		
3x Caja Rack 19" 4U	236,70		
1x Regleta 19" 8 tomas	48,60		
Total	3089,33	Total	517,00

La parte izquierda de la lista es el diseño de los estudiantes, con un precio de 3000€ aprox. La parte derecha se compró posteriormente, tras haber montado el cluster y comprobado su correcto funcionamiento. El colchón de 500€ hubiera permitido reponer algún componente vital si se hubiera estropeado durante el montaje. Al no surgir ninguna contingencia, se aprovechó para darle un acabado quasi-profesional al equipo. Seis estudiantes se animaron a participar en el montaje del cluster. En la Figura 2 se muestra el cluster ACII tras esa sesión (izq.), y con todos los componentes instalados, las puertas retiradas y los nodos parcialmente extraídos.



Figura 2. El cluster ACII tras el montaje (izq.) y tras la 2ª orden de compra (der.).

5 El software

En el curso 2009-2010 los estudiantes que escogieron el proyecto realizaron la instalación del Sistema Operativo Debian con servidor DHCP y firewall IPTables, clonado de nodos con CloneZilla, paso de mensajes con Open-MPI, sistema de colas Torque, monitorización Ganglia y autenticación LDAP. No se probó ningún benchmark, ni se probó el switch Infiniband, debido fundamentalmente a la cantidad de tiempo dedicada a instalar Torque y LDAP. Los profesores indicaron previamente que podían aportar su experiencia con otros Sistemas Operativos, aunque si los estudiantes estaban dispuestos a organizarse entre ellos, repartirse el trabajo equitativamente y comprometerse a tener el cluster operativo en los plazos indicados, no se les obligaría a rechazar la opción Debian. El cluster estuvo operativo en la fecha indicada y sin incidencias durante las sesiones presenciales, aunque la encuesta anónima realizada posteriormente revelara disconformidades con el desarrollo del proyecto que permanecieron ocultas durante todo el cuatrimestre.

6 Encuestas de opinión

En ambos cursos, 2008-09 y 2009-10, se invitó a los estudiantes a rellenar un cuestionario anónimo destinado a evaluar en qué medida se habían alcanzado los objetivos del proyecto, y si se debía mantener el próximo curso, con o sin

modificaciones, o se debería eliminar. Estos objetivos del cuestionario se les comunicaban explícitamente a los estudiantes en el propio formulario. Las preguntas del cuestionario se ofrecen en la siguiente Tabla.

Tabla 5. Cuestionario anónimo del Proyecto Cluster.

Pregunta	Formato
¿Es el primer año cursando la asignatura?	s/n
El proceso de diseño del cluster le ha permitido aprender	-2...2
Considera que se debe mantener este trabajo en la asignatura	-2...2
Justifique 2 anteriores respuestas	libre
¿Qué es lo que más le ha gustado? ¿Por qué?	libre
¿Qué cambiaría?	libre
En la forma de calificar al estudiante ¿qué cambiaría? ¿qué conservaría?	libre
Cualquier otro comentario que desee hacer	libre

De los 12 estudiantes que entregaron la encuesta en el curso **2008-09**, casi la mitad eran matriculados de primera vez (5/12), frente al 100% (6) del curso **2009-10**. En la Tabla 7 se resumen las respuestas más frecuentes, junto con su frecuencia. En general, los estudiantes opinan que la actividad les ha permitido aprender y se debe mantener.

Tabla 6. Resultados de la encuesta anónima.

	2008-2009	2009-2010
Universo	12	6
Primer año	41.7% (5 de 1 ^{er} año / 7 repetidores)	100% (6 de 1 ^{er} año)
Aprender	1.42 (-2:1 -1:0 0:0 1:3 2:8)	1.50 (-2:0 -1:0 0:0 1:3 2:3)
Mantener	1.33 (-2:1 -1:0 0:0 1:4 2:7)	1.33 (-2:0 -1:0 0:0 1:4 2:2)
	frec. respuesta	frec. respuesta
justificación	5x aplicar teoría 4x temas no cubiertos otras asign, motiva, interesa, es voluntario 3x aprender mediante la práctica, trabajar en grupo	2x trab. satisfactorio, interesante 2x trabajo diferente, no cubierto en otras asignaturas 2x se aprende de los compañeros
gustado +	6x el clúster ha sido real foro (nivel, formato, ambiente) 3x los grupos aprender sobre componentes 2x libertad para todo (escoger tarea, proponer comp)	3x desarroll.capacidades profesion útil para un futuro empleo 2x ver el cluster funcionando ir resolviendo problemas instal. 2x entorno distendido
cambiaría	6x que haya más participación mucho trabajo para poca nota 4x que quede + claro al principio objetivos, puntuación, lista tareas, y que no es competitivo	5x organizar mejor: falta experien, material apoyo, difícil aprender que sea + guiado, - libertad 2x grupos + pequeños: 7→5 máx. sesiones todas en Laboratorio no en tutorías (probl. espacio) 2x instalar Infiniband
forma calif.	4x en general, correcta 2x bajar + umbrales, o quitarlos 2x se debe conocer la nota sobre la marcha, tras la aportación	4x + nota, injusto respecto otra Pr. 4x puntuar no según resultados, sino según esfuerzo/aprendizaje

comentario	3x	estoy satisfecho	satisfecho con nuestro trabajo, he aprendido bastantes cosas agradecer dedicación profesor y flexibilidad desarrollo trabajo hay que ser más realista, no da tiempo, montar 2 SO es de locos, o -objetivos o +nota obligar: Debian, Torque... tanta libertad da quebraderos cabeza, obligando ahorraríamos tiempo
	2x	espero que otros estudiantes puedan beneficiarse también	
	2x	se debe conocer la nota sobre la marcha, tras la aportación	

En atención a las opiniones **2008-09**, en **2009-10** se bajaron los umbrales, se redujo drásticamente la cantidad de trabajo a realizar (a la parte software exclusivamente) y se planteó como unas Prácticas alternativas, a pesar de lo cual se apuntaron aún menos estudiantes (7). Los 6 estudiantes que entregaron el cuestionario anónimo opinaron mayoritariamente que debería darse más nota, siendo la puntuación injusta en comparación con las otras Prácticas.

En esta segunda promoción, resulta más claro el efecto *progresivo* de la encuesta, notándose una menor satisfacción y mayor contundencia verbal (“*es injusto*”, “*es de locos*”) conforme se van rellenando apartados.

Lejos de abandonar en este punto, los profesores aún tenemos curiosidad por encontrar en qué forma se pueden proponer prácticas al estilo de este proyecto, haciendo comprender a los estudiantes que instalar y configurar (sin llegar a evaluar con benchmarks) un cluster no es trabajo suficiente para aprobar una asignatura troncal universitaria de 4.5 créditos. Las calificaciones concedidas a los estudiantes se resumen en la Tabla 7, para mostrar que no se trata de un caso de calificación cicatera por parte del profesorado.

Tabla 7. Calificaciones del Proyecto Cluster.

Curso	notas										media
2008-09	3,0	3,0	3,0	3,0	3,0	2,9	2,6	2,5	2,4	2,4	2,41
						1,9	1,8	1,8	1,4	1,4	
2009-10	3,0	2,9	2,8	2,7	2,6	2,5	2,5				2,71

Las visitas web del Proyecto Cluster se resumen en estas analíticas Google:



Figura 3. Informe Google Analytics de la web del Proyecto Cluster [6] (español).

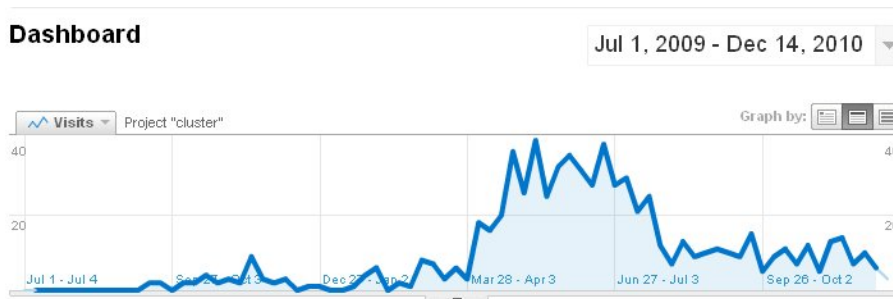


Figura 4. Informe Google Analytics de la web del Proyecto Cluster [6] (inglés).

Se puede observar una distinta estacionalidad en las versiones española e inglesa.

7 Posibles variaciones del proyecto

En vista a futuras variaciones del proyecto, se puede argumentar (mediante las encuestas recolectadas) que los estudiantes desean una planificación rigurosa con hitos marcados en donde se obtenga puntuación que se conozca sobre la marcha, sobre un plan de trabajo previamente fijado por los profesores. El pequeño porcentaje de estudiantes que disfruta de un trabajo con más libertad y no se lamenta del resultado obtenido no compensa el abrumador consenso acerca de cambiar la organización de la experiencia (5/6 en 2010) y la paradójica poca satisfacción con la nota obtenida (sólo 4/12 en 2009, sólo 2/6 en 2010, siempre 1/3 satisfecho, 66% no).

La variación que quedaría por intentar sería realizar unos diarios detallados de instalación, temporizados por el profesor, para eliminar radicalmente el argumento de que es demasiado trabajo para tan poca nota. Los estudiantes que demostraran haberlos leído previamente (por ejemplo, respondiendo preguntas verbales o un test), tendrían derecho a intentar realizar la instalación y configuración descrita en ellos, en las fechas indicadas en la temporización. Se podría ofrecer la posibilidad de abandonar el proyecto durante el curso, si consideran que la nota que van obteniendo no es suficiente, y otra práctica alternativa les resultara más interesante.

8 Conclusiones

Durante los cursos 2008-09 y 2009-10 hemos conseguido poner en contacto a un total de 22 estudiantes con el tipo de multicomputador más habitual actualmente, el cluster de computadores, salvando las diferencias en cuanto a categoría de los componentes (nodos sobremesa en lugar de servidor, red GbE en lugar de Infiniband o similar, etc.). Las actividades realizadas han ido desde el propio diseño y montaje hasta la instalación y configuración del software de sistema y middleware asociado. Los estudiantes consideran que en grado razonable la actividad les ha servido de aprendizaje y se debe mantener, salvando opiniones sobre el desarrollo de la misma.

Agradecimientos. Agradecemos la ayuda del Plan de Innovación Docente de la UGR, que nos ha permitido diseñar un cluster que podremos reutilizar en cursos futuros para seguir intentando motivar a los estudiantes en esta asignatura y otras del área que pueden beneficiarse del equipo adquirido. Agradecemos igualmente la ayuda del Plan de Apoyo a la Docencia Práctica, que nos ha permitido adquirir la infraestructura Infiniband añadida posteriormente a dicho cluster.

Agradecemos especialmente a los estudiantes Pablo Orantes, Ignacio Robles, Luis Quesada y Manuel Martín (2008-09), y Juan Pablo China y Rubén Ramos (2009-10) su honesta colaboración y genuino interés por el buen término del proyecto.

Referencias

1. Railsback, J.: Project-Based instruction: Creating excitement for learning. Northwest Regional Educational Laboratory, "By Request" series (2002). Disponible online en http://educationnorthwest.org/webfm_send/460, adaptado a español en Eduteka <http://www.eduteka.org/AprendizajePorProyectos.php>
2. Pou Amérigo, R., Ochando Gómez, L. E., García Lopera, R.: La coordinación del profesorado a través de proyectos interdisciplinarios en la Licenciatura en Química de la Universitat de València. In: ReVisión, vol.2, no.2, AENUI (2009). Disponible online en <http://www.aenui.net/ojs/index.php?journal=revision&page=article&op=view&path%5B%5D=25>
3. Comellas, F., González-Cinca, R., Santamaría, E.: Simulación: Un curso innovador en los estudios de Aeronáutica. In: ReVisión, vol.2, no.2, AENUI (2009). Disponible online en <http://www.aenui.net/ojs/index.php?journal=revision&page=article&op=view&path%5B%5D=24>
4. Valero García, M., Navarro,J.J.: FAQ sobre la adaptación de asignaturas al EEES: docencia centrada en el aprendizaje del estudiante. In: ReVisión, vol.1, no.2, AENUI (2008). <http://www.aenui.net/ojs/index.php?journal=revision&page=article&op=view&path%5B%5D=8>. Consultar en concreto el penúltimo párrafo en la sección §3.10, sobre la eventualidad de que todos los profesores impartieran docencia basada en proyectos.
5. Fernández Baldomero, F. J., Anguita, M., et al: Memoria descriptiva del Proyecto Clúster. http://serin.ugr.es/unidad_innovacion_docente/memorias/08-08.doc. Buscador de proyectos también disponible online en la web: Convocatoria de Proyectos de Innovación Docente 2008: <http://innovaciondocente.ugr.es/pages/convocatoria-2008>.
6. Web del Proyecto Cluster: http://atc.ugr.es/~javier/docencia/Proyecto_Cluster.html
7. Cañas, A., Díaz, A. F., Prieto, A.: Sistema de servicios web de apoyo a la docencia y gestión de una asignatura. In: Actas de las VIII Jornadas de Enseñanza Universitaria de la Informática (JENUI'2002), pp. 611-614, Cáceres, 10-12 Julio 2002. Disponible online en http://bioinfo.uib.es/~joemiro/aenui/procJenui/Jen2002/Cac635_638.pdf
8. Cañas, A., Ortigosa, E. M., Fernández, F. J., Anguita, M., et al: SWAD (Sistema Web de Apoyo a la Docencia). In: Actas del 6º Simposio Internacional de Informática Educativa (SIEE'04), Cáceres, 16-18 Noviembre 2004.
9. Cañas, A., Calandria, D. J., Ortigosa, E. M. et al: SWAD: Web System for Education Support. In: Fernández-Manjón et al (Eds.): Computers and Education: E-learning - from Theory to Practice, 241 pages, Chapter 12, pp. 133-142, ISBN 978-1-4020-4913-2, Springer, (2007). <http://www.springer.com/computer/general/book/978-1-4020-4913-2>
10. China, J. P., Ramos, R. et al (estudiantes ACII 2009-10): Informe ACII: Puesta en marcha y configuración de un clúster con Debian 5.0. Disponible como documento GoogleDoc <https://docs.google.com/Doc?docid=0Aane43447qy1ZGRxMmp6MjdfNWRwMjNkcWZr>

Web 2.0: Arquitectura Orientada a Servicios en Java^{*}

Pablo García-Sánchez¹, Miguel A. López², Pedro A. Castillo¹, Jesús González¹,
y María I. García Arenas¹

¹ Dept. de Arquitectura y Tecnología de los Computadores, Universidad de Granada

² Fundación I+D del Software Libre

{pgarcia, pedro, jesus, maribel}@atc.ugr.es, malopez@fidesol.org

Resumen Este trabajo presenta los contenidos del curso “Web 2.0: Arquitectura Orientada a Servicios en Java” de la Escuela de Posgrado de la Universidad de Granada. El objetivo del curso es familiarizar al alumno con la programación de Servicios Web. Dada la gran variedad de técnicas disponibles para utilizar Arquitectura Orientada a Servicios, se presentan las siguientes técnicas: utilización de protocolos bien definidos para comunicación y contrato (SOAP y WSDL), creación de Web Services con JAX-WS, orquestación de Servicios Web con BPEL. Al final del curso, el alumno será capaz de crear, utilizar y mantener Servicios Web para el desarrollo de aplicaciones interempresariales, utilizando servicios ya disponibles en la web, así como la orquestación lógica de los mismos.

1. Introducción

Los problemas más comunes en el desarrollo del software suelen ser la incompatibilidad entre aplicaciones, modelos de datos, lenguajes de programación y sistemas de comunicación, lo que obliga a rediseñar todas las aplicaciones y reescribirlas para que operen entre sí. Por lo tanto una aplicación que desee crecer en un futuro debería obviar características restrictivas y partir de un buen diseño que permita la extensibilidad y la comunicación con el mayor nivel de abstracción posible. De esta idea surge la Arquitectura Orientada a Servicios (SOA) [1,2], ya que se hace necesaria una forma de comunicación eficiente y escalable, independiente del lenguaje de programación y plataforma de cada una de las aplicaciones que deseen comunicarse. Los elementos básicos que conforman SOA son:

- Proveedores de servicios: Una aplicación expone operaciones que cualquier otra puede usar
- Consumidores de servicios: Utilizan las operaciones de los proveedores para obtener información
- Bus de servicios empresariales: para integrar los servicios de forma lógica y ampliable.

^{*} Financiado con los proyectos AmIVital (CENIT2007-2010), EvOrq (TIC-3903) y Beca FPU AP2009-2942.

Estos servicios, llamados Servicios Web (Web Services), son un sistema software diseñado para soportar interacción Máquina a Máquina sobre una red, es decir, son interfaces (APIs) que pueden ser accedidas remotamente. Utilizan protocolos para comunicación bien definidos, como SOAP y sus interfaces se publican utilizando el formato WSDL (donde se indican las operaciones y tipos de dato que se pueden intercambiar), siendo su implementación realizada cualquier lenguaje. Esto permite por ejemplo que una aplicación escrita en Java reciba información generada por otra aplicación realizada con C++, PHP o cualquier otro lenguaje de programación.

Debido al auge de estas tecnologías en el mundo empresarial se proyectó la realización del curso “Web 2.0: Arquitectura Orientada a Servicios en Java”, organizado por la Escuela de Posgrado de la Universidad de Granada, para familiarizar a los estudiantes de carreras técnicas (sobre todo los de Ingeniería en Informática o Telecomunicación)³ en estas tecnologías, ya que no están presentes en el plan de estudios de la Universidad de Granada.

El resto del artículo se estructura como sigue: inicialmente se introduce el concepto de Arquitectura Orientada a Servicios. A continuación, en la Sección 3, se presentan los contenidos del curso y las tecnologías utilizadas (Java, XML, PHP, JAX-WS y BPEL), para finalmente mostrar la recepción del curso por parte de los alumnos y las conclusiones a este trabajo.

2. Introducción a los Servicios Web

Actualmente las Arquitecturas Orientadas a Servicios (*Service Oriented Architecture*, SOA) están en auge, debido a los beneficios que proporcionan a la hora de desarrollar e integrar aplicaciones distribuidas o modulares. El principal concepto de SOA es el de *servicio*. Podemos ver un servicio como una llamada a una función, que se ejecutará local o remotamente, y que es independiente del lenguaje de programación y plataforma en la que se ejecuta. Este servicio consta de una interfaz bien definida y que depende de la tecnología que se desea utilizar para implementar SOA.

El resto de elementos básicos que conforman SOA, y cuyas relaciones pueden verse en la Figura 1, son los Proveedores, Consumidores y Publicadores de Servicios. El *Proveedor de Servicios* es un ente (nodo, clase, programa, etc.) que brinda un servicio en respuesta a una llamada o petición desde un *Consumidor de Servicios*. Éste utiliza el *Publicador de Servicios* para obtener información sobre los servicios que estén disponibles para su uso y sobre las interfaces (*Descripción del servicio*) para invocarlos.

A la hora de desarrollar sistemas software se hace necesario que sean compatible con las implementaciones SOA más extendidas, como por ejemplo los *Servicios Web (Web services)* [2]. Su arquitectura está diseñada para soportar interacción máquina a máquina sobre una red, utilizando sobre todo el protocolo

³ Cualquiera se puede apuntar, pero por los conocimientos previos que se recomiendan, está orientado especialmente a alumnos de esas dos titulaciones.

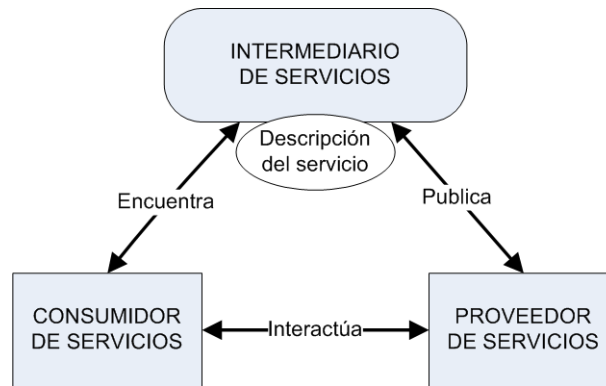


Figura 1. Esquema de interacción de servicios. El proveedor de servicios publica una descripción del servicio que es utilizada por el consumidor para encontrar y usar servicios.

SOAP (*Simple Object Access Protocol*) [3] para transmitir mensajes entre los diferentes computadores. Las interfaces de los servicios están descritas en WSDL (*Web Service Description Language*) [4], un lenguaje basado en XML que proporciona un modelo para describir Servicios Web y la manera de comunicarse utilizando éstos (equivalente a *Descripción del servicio* en la Figura 1). Estos servicios pueden ser listados usando UDDI (Universal Description, Discovery, and Integration) [5], un registro basado en XML independiente de la plataforma (*Intermediario de servicios* en la Figura 1).

En un ambiente SOA, los nodos de la red suelen poner disponibles sus recursos a otros participantes en la red como servicios independientes, a los que tienen acceso de un modo estandarizado. La mayoría de las definiciones de SOA identifican la utilización de Servicios Web (empleando SOAP y WSDL) en su implementación, pero no obstante se puede implementar SOA utilizando cualquier tecnología basada en servicios, como por ejemplo, OSGi [6].

3. Temario

Esta sección presenta los contenidos teóricos y prácticos del curso. El curso dura 50 horas, divididas en cinco bloques: Introducción a los Servicios Web y Java (10 horas), XML (5 horas), Creación de servicios web en Java utilizando Jax-WS (15 horas), PHP (5 horas), Orquestación de Servicios Web con BPEL (10 horas) y finalmente un día para presentar otras arquitecturas SOA y examen (5 horas).

3.1. Introducción a los Web Services y a Java

Como introducción al curso se le presenta al alumno de forma sucinta el funcionamiento de los Servicios Web, para luego profundizar en una introducción

al lenguaje Java, ya que será el que se utilizará durante el curso. En esta sección se explican las peculiaridades del lenguaje y se realizan distintos ejercicios simples para comprender el funcionamiento y uso. Asimismo se explica el uso de NetBeans ⁴, entorno de desarrollo libre usado para los ejercicios en el curso y algunas de las funcionalidades básicas (compilación, depuración, creación de proyectos...).

3.2. XML

En la segunda parte del curso, se les muestra a los alumnos en qué consiste XML (Extensible Markup Language) junto con herramientas para su manejo y ejemplos prácticos variados de utilización en diversos ámbitos de la programación web.

El orden del temario que se incluye en esta parte es el siguiente:

- Introducción: Simplemente se presenta al docente, sus datos, datos de donde encontrar la documentación necesaria para esta parte y se sitúa esta parte del curso dentro de la totalidad del temario.
- XML: Donde se introduce en qué consiste XML, su utilidad, ventajas frente a otros lenguajes de marcas y cada uno de las partes que lo componen.
- Protocolos XML: Donde se mencionan varios de los protocolos de comunicación más utilizados en la actualidad y que están basados en XML, como son: XML-RPC, SOAP y RSS.
- XML y Java: Java es la herramienta que se utiliza en todas las partes del curso para programar, por lo que en esta parte se introduce lo que ofrece Java para tratar documentos XML. Se mencionan aspectos básicos para el tratamiento de la información como el análisis de la formalidad de los documentos, la aplicación de “plantillas” a los documentos para comprobar si están correctamente formados, etc. También es en esta parte donde se mencionan la gran cantidad de herramientas que Java proporciona para tratar XML. Y concretamente, se detalla el funcionamiento básico de dos de ellas en los siguientes apartados del curso.
- SAX: Es una de las APIs disponibles en Java para el tratamiento de documentos XML. Durante el curso se ve la estructura general de SAX, así como diversos ejemplos de utilización que comienzan siendo sencillos para ir complicándose a lo largo del temario.
- DOM: Es la segunda API que se muestra a los alumnos puesto que el funcionamiento es bastante diferente a la anterior, dando así dos enfoques totalmente distintos a los asistentes al curso de cómo tratar documentos XML con Java.

A lo largo de todo el temario se le propone a los alumnos la realización de numerosos ejemplos relacionados con la materia que se está introduciendo y son ellos los que los realizan aunque al final del curso se les proporciona la dirección web donde pueden encontrarlos ya resueltos.

⁴ <http://www.netbeans.org>

Entre estos ejercicios se encuentra realizar un programa Java que se conecte a un servidor remoto donde se encuentran disponibles varios servicios web ya programados y en funcionamiento. El alumno puede aprender como realizar con Java una conexión remota de estas características, cómo acceder a los servicios web disponibles y cómo obtener la respuesta de estos servicios a la petición que ellos formulan.

Para esta propuesta se utiliza una herramienta adicional denominada SoapUI (<http://www.soapui.org/>). Se trata de una herramienta de software libre que nos permite testear el funcionamiento de servicios webs y que los alumnos comienzan a utilizar en esta parte del curso y continúan en las partes siguientes.

3.3. Desarrollo de Servicios Web con Jax-WS

Sabiendo que la teoría sobre Java, XML y los Servicios Web ya se han impartido en los primeros capítulos del curso, este apartado se centra estructurar conocimientos que permitan relacionar conceptos cuando estemos desarrollando servicios web.

Para realizar un desarrollo de servicios web haciendo uso de Netbeans es necesario explicar nuevos conceptos de la herramienta Netbeans (ya que se usarán nuevas funcionalidades). Además, se explica el uso del servidor de aplicaciones web donde nuestros servicios se desplegarán para su ejecución, que en este caso es el servidor Glassfish ⁵. El alumno aprende conceptos básicos del servidor como son: arrancar, parar, cambiar a modo de depuración, desplegar nuevos proyectos y sincronizarlo con el Netbeans.

El temario comienza con la explicación del desarrollo de un servicio web sin utilizar la ayuda de un entorno de desarrollo moderno. En este tema se introduce al alumno en el desarrollo de un WSDL (xml de descripción de un servicio web) escribiendo cada uno de los elementos necesarios para que el sistema pueda comprender y levantar un servicio web. Es decir, a partir de un WSDL se crea automáticamente el código a rellenar con su comportamiento. También se explica el paso contrario: a partir de una clase Java se genera automáticamente el WSDL que la representa.

El siguiente tema presenta la creación de clientes para web services, introduciendo al alumno en las técnicas de programación necesarias para la invocación de servicios web desde cualquier programa Java (una aplicación de escritorio, una aplicación web u otro servicio web). Durante la explicación de este apartado el alumno puede crear clientes para todos los ejemplos realizados en el curso y comparar los resultados por los ofrecidos por un cliente genérico como es SoapUI.

También se profundiza en la creación de Servicios Web y clientes más completos, con estructuras de datos reales y más complejas.

Como conclusión de esta sección se introduce al alumno en el desarrollo de una aplicación final haciendo uso de servicios web desarrollados por ellos mismo que realizan operaciones simples. Como ejemplos de proyecto se propone el desarrollo de interacciones de personajes de una conocida serie de televisión,

⁵ <http://www.glassfish.org>

utilizando objetos complejos, como Usuarios, Listas de Usuarios y otros objetos compuestos.

3.4. PHP

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas.

Es un lenguaje interpretado de propósito general ampliamente usado y diseñado especialmente para desarrollo web que puede ser incrustado dentro de código HTML. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. Veamos un ejemplo sencillo:

```
<html>
<body>
  <?php
    echo ‘‘Hola. Esto es un script PHP’’;
  ?>
</body>
</html>
```

Cuando el cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado, que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. Así, al ejecutar el script anterior, el cliente recibirá sólo los resultados de la ejecución por lo que es imposible para el cliente acceder al código que generó la página.

A lo largo del curso se muestran diversos ejemplos, partiendo de algunos muy sencillos para estudiar la sintaxis básica, definición de tipos de datos y uso de estructuras de control. Más adelante se estudian conceptos y ejemplos avanzados, tales como orientación a objetos, tratamiento de formularios web, despliegue y uso de servicios web, y por último, acceso a bases de datos MySQL desde PHP. Con esta sección el alumno comprende que los Servicios Web son independientes del lenguaje, pudiendo llamar a servicios hechos en Java desde programas PHP.

3.5. Orquestación de Servicios Web con BPEL

BPEL (*Business Process Execution Language*, Lenguaje de Ejecución de Procesos de Negocio) es un lenguaje basado en XML diseñado para el control centralizado de la invocación de diferentes servicios Web, con cierta lógica de negocio añadida que ayudan a la programación en gran escala (programming in the large).

Su objetivo es definir procesos de negocio que interactúen con entidades externas mediante operaciones de un servicio Web definidas usando WSDL y que

se manifiestan a si mismas como un servicio Web. Para ello utiliza un lenguaje basado en XML. BPEL sirve para orquestar servicios web, ya que consume servicios existentes para crear nuevos servicios de grano grueso. Un ejemplo de un proceso BPEL puede verse en la Figura 2. Al estar definido en XML existen muchos programas que permiten desarrollar procesos BPEL de forma gráfica, como por ejemplo NetBeans.

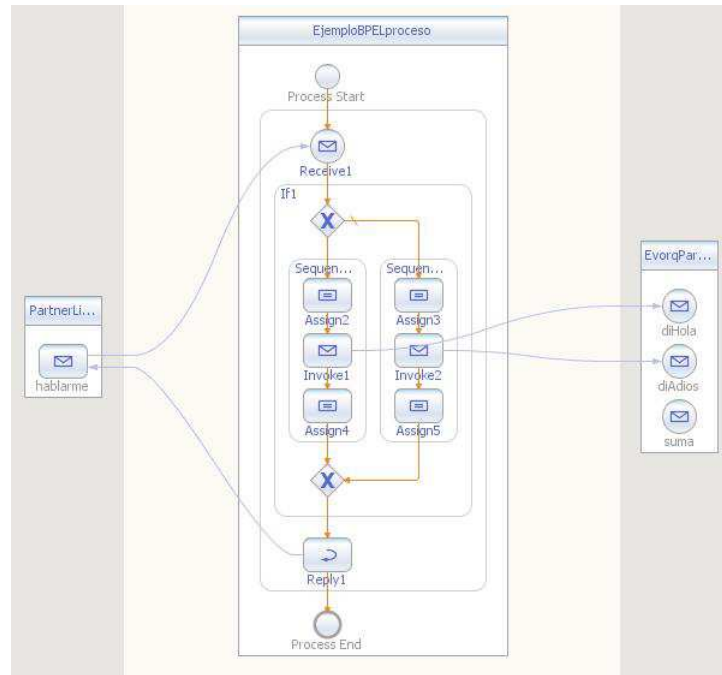


Figura 2. Ejemplo de proceso BPEL realizado en NetBeans. Este proceso llama a las operaciones diHola o diAdios dependiendo de un booleano, y devuelve un saludo

En la parte teórica de este curso se explica también el concepto de ESB (*Enterprise Service Bus*). Un ESB es un contenedor de aplicaciones en el que cualquier aplicación que entienda XML puede ser emisora/receptora de mensajes dentro del bus. Permite una enorme flexibilidad a la hora de diseñar la arquitectura de nuestro sistema, ya que se puede utilizar desde una arquitectura cliente/servidor hasta una arquitectura orientada a eventos. Los componentes son “enchufables”, esto da la flexibilidad de añadir funcionalidad a un sistema en producción sin tener que pararlo y sin comprometer el funcionamiento de otros componentes del sistema. Por ejemplo, un componentes básico para ESB es un motor que ejecuta BPEL. Un ejemplo de ESB libre es OpenESB [7], incluido dentro del servidor Glassfish de Sun y utilizado en el curso.

En esta parte del curso se explican los distintos conceptos del lenguaje BPEL y se propone al alumno realizar varios ejercicios utilizando los servicios web creados en secciones anteriores del curso. Para ello el alumno utilizará el programa NetBeans para desarrollar procesos BPEL y desplegarlos dentro de OpenESB. Los primeros ejercicios son iterativos, los alumnos van añadiendo nuevos servicios simples ya desplegados en el servidor del curso a su proyecto BPEL para ir aprendiendo el uso de todas las actividades que se usan en este lenguaje (*for*, *if*, *invoke*, *reply*...). Una vez comprendidos estos conceptos, se les presenta un proyecto más completo en el que tienen que utilizar otros servicios desplegados más complejos para crear distintos procesos BPEL que los orquesten.

3.6. Otras arquitecturas y metodologías SOA

Como cierre al curso se presentan de forma teórica otras implementaciones de SOA y metodologías de desarrollo. Entre las arquitecturas orientadas a servicio más extendidas está OSGi [6], que es una SOA para máquinas virtuales en Java, o ebXML [8], orientada sobre todo al intercambio de datos empresariales. Además, se presentan algunas metodologías para desarrollar sistemas basados en SOA, como BCM [9] o SOMA [10].

Finalmente se presentan brevemente algunos proyectos basados en SOA en los que los docentes del curso han trabajado tanto en el ámbito académico como el empresarial, como por ejemplo eIntegra [1], GAD, AmIVital⁶ [11] u OSGiLiath [12].

4. Respuesta de los alumnos y Conclusiones

El curso ha sido impartido en tres ediciones en la ETS. de Ingenierías en Informática y Telecomunicación de la Universidad de Granada. En todas ellas se ha cubierto el cupo de 27 alumnos. Al final de cada edición se pasó un cuestionario para evaluar el curso, siendo las notas 4,08, 4,5 y 4,8 (sobre 5) en las tres ediciones respectivamente. Además, se propuso a los alumnos qué mejoras habrían de añadirse en ediciones posteriores, siendo aplicadas en tales ediciones, como por ejemplo el aumento de la parte de prácticas respecto a teoría, y orientando la enseñanza a la realización de ejercicios iterativos.

Todo el software utilizado en el curso es libre, con lo que el coste para el alumno es nulo. Toda la información, software y ejemplos del curso se encuentran en la web <http://evorq.ugr.es/cursows>, a disposición de los alumnos y el resto de personas interesadas en la temática.

Referencias

1. García-Sánchez, P., Merelo, J., Castillo, P., Sevilla, J., Martín, M., López, M.: Plataforma de integración de servicios para la administración basada en BPEL y

⁶ <http://www.amivital.es>

- SOA. In: Actas de las III Jornadas en Servicios Web y SOA (JSWEB 2007). (2007) 111–118
2. Papazoglou, M.P., Van Den Heuvel, W.: Service oriented architectures: Approaches, technologies and research issues. *VLDB Journal* **16**(3) (2007) 389–415
 3. World Wide Web Consortium: SOAP Version 1.2 Part 1: Messaging Framework (Second Edition) (2007)
 4. World Wide Web Consortium: Web Services Description Language (WSDL) 1.1. (2001) Disponible en: <http://www.w3.org/TR/wsdl>.
 5. Oasis Open: UDDI Specification. (2006) Disponible en: <http://www.uddi.org/specification.html>.
 6. OSGi Alliance: OSGi service platform release 4.2. (2010) Disponible en: <http://www.osgi.org/Release4/Download>.
 7. Salter, D., Jennings, F.: Building SOA-Based Composite Applications Using NetBeans IDE 6. Birmingham-Mumbai: Packt Publishing (2008)
 8. Patil, S., Newcomer, E.: ebXML and Web Services. *IEEE Internet Computing* **7**(3) (2003) 74–82
 9. OASIS BCM TC: Business-Centric Methodology for Enterprise Agility and Interoperability. Executive White Paper. (2003) <http://businesscentricmethodology.com/>.
 10. Arsanjani, A., Ghosh, S., Allam, A., Abdollah, T., Ganapathy, S., Holley, K.: SOMA: A method for developing service-oriented solutions. *IBM Systems Journal* **47**(3) (2008) 377–396
 11. García-Sánchez, P., González, J., Castillo, P., Prieto, A.: Using UN/CEFACT'S Modelling Methodology (UMM) in e-Health Projects. *Bio-Inspired Systems: Computational and Ambient Intelligence* (2009) 925–932
 12. García-Sánchez, P., González, J., Castillo, P., Merelo, J., Mora, A., Laredo, J., Arenas, M.: A Distributed Service Oriented Framework for Metaheuristics Using a Public Standard. *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)* (2010) 211–222

Uso de Hardware Libre en el Máster de Formación del Profesorado de Secundaria, (Especialidad Informática) .

Guillermo Botella¹, Daniel González^{2,4}, Pablo Angulo³,
Jordi López⁴, Francisco de Asís Martínez⁴ and Samuel F. Romero García⁵

¹ Departamento de Arquitectura de Computadores y Automática, Facultad de Informática, Universidad Complutense de Madrid, 28040, Madrid, España

² Departamento de Tecnología, Instituto de Enseñanza Secundaria Francisco de Quevedo, 28037, Madrid, España

³ Departamento de Matemáticas, Facultad de Ciencias, Universidad Autónoma de Madrid, Madrid, España

⁴ Alumnos del Máster de Formación del Profesorado de Educación Secundaria (Especialidad Informática, promoción 2009-2010) Universidad Complutense de Madrid, 28040, Madrid, España

⁵ Departamento de Tecnología y Arquitectura de Computadores, Escuela Técnica Superior de Ingeniería Informática y de Telecomunicación, Universidad de Granada, 18071, España

Correspondencia : gbotella@fdi.ucm.es

Abstract. En esta contribución exponemos la experiencia del uso de plataformas digitales de código y esquemáticos abiertos como una parte importante dentro del temario del Máster de formación del profesorado de secundaria (especialidad informática) en la Universidad Complutense de Madrid. Se describen y ordenan temáticamente un número de iniciativas de hardware libre, motivando esta experiencia dentro del constructivismo más clásico. Posteriormente, se describe el trabajo a realizar y se presenta un breve cuestionario, comentando los resultados obtenidos en el marco de las implicaciones del uso de este tipo de plataformas en centros de enseñanza.

Keywords: Robótica, Hardware Libre, Arduino, Lego, Constructivismo, MFPS, Educación Secundaria, EEES.

1 Introducción y motivación de este trabajo

Esta contribución pretende presentar la experiencia sobre el uso de hardware libre como uno de los puntos fuertes dentro del contenido de la asignatura Didáctica de la Informática II [1], perteneciente a los estudios del Máster de Formación del Profesorado (Especialidad Informática).

Si contextualizamos brevemente la misma, podemos especificar que tiene 5 créditos ECTS, es de carácter obligatorio y de duración semestral. Se diseña el descriptor de la ficha docente como Organización de un aula-taller de informática. Innovación en educación. Hardware y Software educativos.

No se requieren requisitos previos y como Objetivos tenemos: Competencias Generales: G1, G2, G3 Y G4. Transversales y Específicas CE16, CE17, CE18, CE19, CE20, CE21, CE22, CE23, CE24 Y CE25.

El denominado hardware libre se compone de los dispositivos cuyas especificaciones y diagramas esquemáticos son de acceso público, ya sea bajo algún tipo de pago o de forma gratuita.

Hoy en día encontramos una gran cantidad de proyectos basados en hardware libre, por ejemplo:

- Elphel [2], es una empresa especializada en dispositivos de visión mediante red basados en GNU/Linux y FPGAs. El código que proporcionan es FreeSoftware y todas las disposiciones PCB, diagramas de socket y fuentes de FPGA Verilog están disponibles bajo la licencia de GNU/GPL.
- Proyectos de impresoras tridimensionales libres y autorreplicantes, como ejemplo tenemos RepRap [3], Clanking Replicator Project [4] y Fab@Home [5].
- Proyectos de microprocesadores libres como OpenSPARC [6] basado en un chip multinúcleo UltraSPARC T1 [7] de Sun Microsystems. OpenRISC [8] trabaja para producir un CPU RISC libre de muy alto rendimiento. LEON [8] es una CPU de 32 bits creada por la ESA [9] y la CPU estándar para la industria espacial europea.
- Arduino es una plataforma libre de computación física basada en una tarjeta de entrada/salida y un entorno de desarrollo que implementa el lenguaje libre “Processing / Wiring”. Será la plataforma objeto de esta contribución [10].
- Bug Labs [11] es una compañía tecnológica cuyo principal producto es el BUG. Esto es una plataforma abierta de dispositivos electrónicos de consumo que abarca el computador BUGbase mini-linux y complementos como cámara, acelerómetro, detector de movimiento, teclado táctil, GPS, etc.
- Proyectos de computadores libres, como Open OEM [12], OpenBook [13] que estaría situado entre la computadora portátil de 100 dólares (OLPC) y la Tablet PC. Simputer [14], siendo éste último un computador de mano dirigido a los países en desarrollo.
- Proyectos de tarjetas específicas, como Open Graphics Project [15a] que apunta a diseñar una arquitectura abierta y estándar para tarjetas gráficas. BalloonBoard [16] produce tarjetas de desarrollo basadas en el procesador ARM, dirigidas a los OEMs y Further Education. ECB ATmega32/644 [17] – está basado en una tarjeta que lleva el ATmega32/644 de Atmel (de 20 MHz) con funcionalidad de servidor web y un consumo de energía inferior a 100mA.
- Proyectos cooperativos inalámbricos, como OpenPCD [18] sobre un lector/escritor RFID, usando el microcontrolador AT91SAM7S128. SquidBee - Open Mote [19], basado en Arduino para desarrollar redes de sensores.

- Proyectos generales de hardware abierto, como Open Hardware (OH) [20], donde los diseñadores de hardware comparten los esquemáticos y los drivers usados en sus diseños. Este sistema fomenta la cooperación mutua, la solución de problemas de diseño y la propuesta de ideas. OpenCores [21] es otro ejemplo de fundación que forma una comunidad de diseñadores para apoyar los diseños libres para CPU, periféricos y otros dispositivos. OpenCores da mantenimiento a un bus de conexión libre llamado Wishbone.
- Proyectos de dispositivos de telefonía libre como Opencellphone [22], también llamado 'TuxPhone'. OpenMoko [23], Astfin - Free Telephony Project, [24] hardware de telefonía libre.
- Proyectos de vehículos autónomos ecológicos que siguen los principios del movimiento libre, como "c,mm,n" [25], Oscar [26]: es el primer intento de diseñar un coche entero. EVProduction club es una organización con un wiki para diseñar y producir vehículos eléctricos y complementos libres. OSMC [27] , como proyecto de control de motor libre, pensado principalmente para la robótica, pero aplicable a vehículos eléctricos de baja potencia y otros usos.
- Dispositivos multimedia, como Source Device, un dispositivo libre tipo set-top box diseñado para servir como un "media center" de bajo costo en Linux. Daisy - Un reproductor de MP3 libre [28]. GP2X [29], una consola de mano de videojuegos y reproductor multimedia libre basada en Linux, creada y vendida por GamePark Holdings de Corea del Sur.

Si indagamos acerca de la robótica educativa, vemos que uno de los principales promotores fue Seymour Papert, un matemático de la década de 1960, interesado en la ciencia, en la adquisición del conocimiento y en el desarrollo de la mente infantil. Es uno de los creadores del Lenguaje de programación LOGO, que fue ampliamente usado para la enseñanza de la programación.

Papert, firmemente partidario del constructivismo, defiende que el niño sea "constructor de sus propias ideas mentales". El niño es creador de su conocimiento y la educación tiene que proporcionarle herramientas para ello.

Papert escribió un libro que se llama "Mindstorms: Children, Computers, and Powerful Ideas" [30] . El presidente de la compañía LEGO leyó el libro y decidió emprender el camino hacia la construcción de la robótica educativa, precisamente de ahí surgió la idea de la creación del ladrillo programable, que más tarde llevaría el nombre de Paper. Son famosas las palabras de los creadores de LEGO (Resnick y Steve Ocko) y emanan de aquí: "Diseñar cosas que permitan a los estudiantes diseñar cosas".

En esta contribución nos vamos a centrar en una plataforma que permita seguir las ideas constructivistas, pero al mismo tiempo de bajo coste (libre) como el Boarduino. El boarduino es una versión simplificada del arduino, siendo a su vez una plataforma de hardware libre basada en una placa de entradas y salidas simple y un entorno de desarrollo que implementa el lenguaje de programación *Processing/Wiring* .

2. Construcción del sistema con el Boarduino



Fig. 1. Placa Boarduino, con un coste aproximado de 15 euros. El Arduino se puede conseguir por 30 euros[10].

En la asignatura Didáctica de la Informática II (MFPS, Informática, UCM) se realizó un taller de cómo conseguir unos dibujar unos LED en las ruedas de una bicicleta. Esta aplicación está suficientemente documentada y expuesta en [31] de donde extraemos la siguiente imagen.



Fig. 2. Aplicación donde se usa la rueda giratoria de la bicicleta, basada en un banco de leds que van cambiando de color según la velocidad angular de giro.

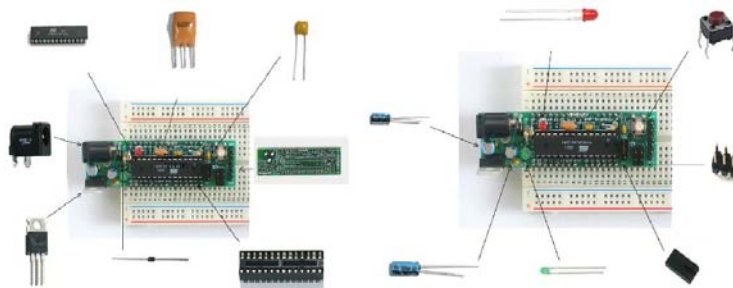


Fig. 3. Componentes que se usarán en esta práctica, donde se aprecia el propio microprocesador, condensadores, la placa de prototipado, zócalos, diodos, diodos led, condensadores, sensor de efecto Hall, interruptores.

Se procede al ensamblado del sistema, apareciendo información de la soldadura en [32]. Así mismo se prescinde del conector de alimentación debido a que se usará una batería pequeña. En este caso particular también se prescinde de la placa “breadboard” debido a la capa que se diseñará con los led y los registros de desplazamientos. Las entradas analógicas, vienen marcadas con una *a*, las digitales como *d* y *tx, rx* pueden servir como entradas digitales o salidas digitales. Se dispone de 14 salidas y 20 entradas.

La capa de leds se conectará con la parte superior del procesador para facilitar las conexiones de entrada y salida. Al final de los cables encontramos un sensor de posición de efecto hall, que proporciona respecto al valor inicial, un incremento en presencia de un campo magnético (producido por imán en este caso) y decremento en presencia de un imán polaridad cambiada.

El sensor de posición de efecto hall se montará en un radio de la rueda posicionando un imán en la pastilla de freno de la rueda, ahora cada revolución se puede detectar. El programa a ejecutar utilizará esta información para cuadrar los dibujos dependiendo de la velocidad. Como alternativa se han usado circuitos mecánicos activados por el paso del radio por un determinado sitio, sin embargo suelen tener problemas de calibración y de desgaste. Los dispositivos como el sensor de posición de efecto hall o los acelerómetros no se ven afectados por este inconveniente.

A continuación construimos dos placas (una para cada radio) con 16 leds/placa.



Fig. 4. Esquema de la “breadboard” donde aparecen los puertos de entrada y salida del sistema.

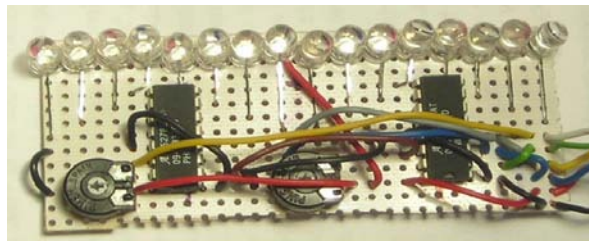


Fig. 5. Esquema del circuito donde van colocados los leds.

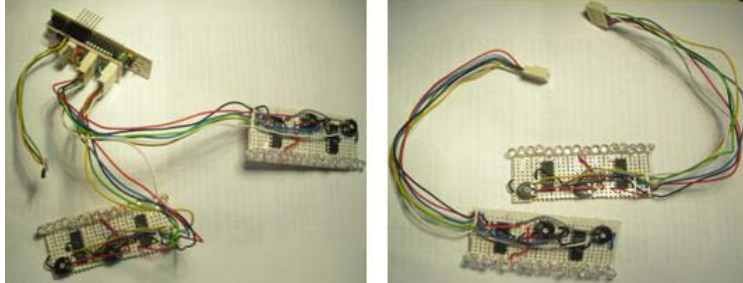


Fig. 6. Esquema general de los circuitos de los dos bancos de leds y el procesador .

Se usarán 4 registros de desplazamiento, para el manejo de los 32 leds. El cable de color amarillo activa/desactiva los leds con los bits que tenga almacenado en ese momento los registros, una vez encendidos o apagados, podremos cambiar los bits que están en los registros sin que cambie el estado de los leds, hasta el nuevo uso del cable amarillo. El cable azul es el reloj, va escribiendo una secuencia periódica 10101010..., cada vez que cambia entrará un nuevo ciclo. El resto de colores son para los datos y la corriente.

Una vez hecho esto unimos las tres partes como se muestra en la figura 7. Se usará una fuente de alimentación (2 pilas de 1,5 V) y se monta en el radio de la bicicleta. Como nota importante se aconseja dejar los pines accesibles para conectar el PC y cambiar los programas.



Fig. 7. Esquema del sistema montado en el radio de la bicicleta. Funcionamiento final.

Acera del software a usar se requiere la máquina virtual de Java y las bibliotecas avr-libc. Una vez realizado este paso desde la página de Arduino [4], se instala el software siendo el editor muy sencillo. Dispone de la posibilidad de obtener la salida por pantalla mientras se ejecuta el Arduino. Si hacemos girar la rueda finalmente vemos un ejemplo de programa en la figura 7.

3 Resultados obtenidos y cuestionario

Esta experiencia piloto ha servido para obtener unos resultados visibles que motivan al futuro profesorado de secundaria en su paso por el Máster obligatorio de Educación Secundaria. El hecho de tener herramientas de prototipado rápido, de bajo coste y programación sencilla hace que podamos crear sistemas fieles a los que nos hemos imaginado, siguiendo los paradigmas del constructivismo más ortodoxo.

Si pensamos actividades como un huerto solar, robot sencillos o proyectos cooperativos, vemos que se pueden trabajar las competencias que aparecen en los Boletines Oficiales que regulan los currículos de las diferentes Comunidades Autónomas.

A continuación mostramos el grado de confianza del futuro profesorado de Educación Secundaria en plataformas hardware no propietarias con una breve encuesta, realizada a los alumnos de la asignatura Didáctica II y profesores de IES.

“Valore (1-5) Las siguientes afirmaciones, (1) total desacuerdo, (5) total acuerdo.

P1. ¿Cree que esta experiencia sirve para plantear nuevas formas de enseñar en Ed. Secundaria?

P2. ¿Estaría dispuesto a aplicar este sistema en un instituto y montar un laboratorio?

P3. Valore la dificultad de diseño de aplicaciones prácticas reales, (1 muy difícil, 5 muy fácil) después de haber visto el entorno de diseño, documentación y de acuerdo a su formación inicial.

P4. Valore la importancia de la robótica educativa (1 poca importancia, 5 mucha importancia) dentro del temario propuesto para la asignatura Didáctica de la Informática II (se adjunta):

- a) Entorno de trabajo en el aula-taller y su influencia en el rendimiento académico. Diseño de Aula Taller de Informática.
- b) Organización del aula-taller, problemas y soluciones teniendo en cuenta arquitectura y estructura del PC. Componentes. Configuración y diagnóstico de averías.
- c) SW educativo. Introducción. Generalidades y casos de estudio concretos.
- d) Proyectos educativos.
- e) Hardware Libre. Robótica educativa.
- f) Informática en educación aplicada a la discapacidad.
- g) Investigación en educación. Innovación. Casos prácticos.
- h) Máquinas virtuales y acceso controlado a Internet en sesiones de trabajo de alumnos. Seguridad. “

Encuesta realizada por 41 personas:

Cuestionario	P1	P2	P3	P4
1	19%	25%	15%	15%
2	10%	15%	20%	14%
3	28%	23%	27%	21%
4	29%	35%	27%	36%
5	13%	7%	11%	15%

4 Referencias

1. Máster en Secundaria UCM, Especialidad Informática, [online] en <http://www.ucm.es/centros/webs/m5057/index.php?tp=Especialidades&a=dir3&d=18729.php>
2. [online] <http://www.elphel.com>
3. [online] http://reprap.org/wiki/Main_Page
4. [online] <http://3dreplicators.com/>
5. [online] <http://www.fabathome.org/>
6. [online] <http://www.opensparc.net/marketplace/>
7. [online] <http://datasheets.chipdb.org/Sun/UltraSparc-III.pdf>
8. [online] <http://datasheets.chipdb.org/Sun/UltraSparc-III.pdf>
9. [online] <http://www.esa.int/esaCP/index.html>
10. [online] <http://www.arduino.cc/>
11. [online] <http://www.buglabs.net/>
12. [online] <http://www.innovationstage.com/openoem/>
13. [online] http://www.opencircuits.com/Open_source_electronics
14. [online] <http://www.simputer.org/>
15. [online] <http://wiki.opengraphics.org/tiki-index.php>
16. [online] <http://www.balloonboard.org/>
17. [online] http://en.wikipedia.org/wiki/ECB_ATmega32/644
18. [online] <http://www.openpcd.org/>
19. [online] http://www.libelium.com/squidbee/index.php?title=Main_Page
20. [online] <http://www.openhardware.net/>
21. [online] <http://opencores.org/>
22. [online] http://www.opencellphone.org/index.php?title=Main_Page
23. [online] <http://www.openmoko.com/>
24. [online] http://www.rowetel.com/blog/?page_id=457
25. [online] <http://www.cmmn.org/>
26. [online] <http://www.theoscarproject.org/>
27. [online] http://www.robotpower.com/osmc_info/
28. [online] <http://teuthis.com/daisy/index.html>
29. [online] <http://en.wikipedia.org/wiki/GP2X>
30. S. Papert, “ Mindstorms: children, computers and powerful ideas”, pp. 244, Basic Books, Inc. New York, 1980. Isbn: 0-465-04627-4 appears in ACM-CBS: ACM Classic Books Series
31. Monkeylectric.com, empresa distribuidora de displays LED para ruedas de bicicleta, [online] http://www.monkeylectric.com/m464q_gallery.htm

32. Guía de soldadura y montaje del programador USBtinyISP. [online] en <http://www.ladyada.net/make/boarduino/solder.html>

El perfil de Ingeniería de Computadores y las asignaturas de Estructura y Arquitectura en el Grado de Ingeniería Informática

Anguita, M.; Cañas, A.; Fernández, F.J.; Ortega, J.; Rojas, I.

Departamento de Arquitectura y Tecnología de Computadores
E.T.S.I.I.T., Universidad de Granada
[\[manguita,acanas,javier,julio,ignacio}@atc.ugr.es](mailto:{manguita,acanas,javier,julio,ignacio}@atc.ugr.es)

Resumen. En este trabajo se pone de manifiesto el interés de un perfil de Ingeniería de Computadores a pesar de las características de nuestro entorno socio-económico. Además, se proponen preguntas relacionadas con la Estructura y Arquitectura del Computador, que son útiles y pertinentes para los distintos perfiles de grado en Ingeniería Informática. Por tanto, se argumenta a favor de una enseñanza de la Estructura y la Arquitectura de los Computadores centrada en los aspectos que ponen de manifiesto la interrelación de los distintos niveles de descripción del computador. Esta perspectiva permite motivar a estudiantes menos interesados en aspectos relacionados con los niveles tradicionalmente hardware.

Palabras Clave: Ingeniería de computadores, Niveles de descripción del computador, IEEE/ACM *Computing Curricula*.

1 Introducción

Los niveles de descripción o de abstracción del computador [1] permiten organizar su estudio como sistema complejo (Figura 1). Como sabemos, cada nivel ofrece una interfaz que permite su uso por parte de los niveles superiores. En cada nivel se estudian los bloques que, a su vez, se utilizan como componentes en el nivel inmediatamente superior. Esta aproximación ha contribuido a la separación entre distintas áreas de conocimiento y ha justificado aproximaciones parciales a la ingeniería informática apoyadas en estudios acerca de la situación socio-económica de nuestro entorno y en previsiones sobre demanda de profesionales que concedían mayores oportunidades a los niveles de descripción superiores relacionados con el desarrollo de aplicaciones informáticas. Esta perspectiva afecta de forma considerable al aprendizaje de asignaturas obligatorias dentro del plan de estudios de Ingeniería Informática como Estructura y Arquitectura de Computadores y a las optativas relacionadas con ellas, y puede condicionar la demanda de acceso al perfil de Ingeniería de Computadores en los nuevos planes de estudios de Grado en Ingeniería Informática.

Los modelos más aceptados actualmente indican que el aprendizaje se construye [2]. Así, los estudios recientes acerca del aprendizaje han puesto de manifiesto que la memoria no es un almacén donde se introducen cosas que luego se podrán sacar sin más. Desde nuestro nacimiento interconectamos sensaciones y construimos patrones que nos ayudan a entender la realidad. Al enfrentarse al aprendizaje de una asignatura los estudiantes poseen modelos mentales que han ido forjando y que serán los que usen para acercarse a dicha asignatura. Por tanto, es importante tener en cuenta las ideas preconcebidas de los estudiantes y estimularles a que construyan modelos adecuados para llegar al espíritu de la asignatura, al aprendizaje profundo que les permita recordar y enfrentarse a problemas, más allá del tiempo necesario para aprobar un examen. Para conseguir esto habría que mostrarles situaciones en las que su modelo mental se muestre insuficiente, y ofrecerles alternativas que contrarresten sus prejuicios e ideas preconcebidas. La existencia de prejuicios respecto a asignaturas, como las Arquitecturas de Computadores I y II de la Ingeniería Informática de la Universidad de Granada (B.O.E de 25/10/2000), que se imparten en el último curso de una titulación muy escorada hacia los niveles de descripción más altos, condiciona la forma en la que los estudiantes se acercan al aprendizaje en estas asignaturas y hacen imprescindible un esfuerzo de motivación intenso por parte del profesorado. Estas situaciones ponen de manifiesto la importancia de una correcta ubicación temporal de las asignaturas, y de proporcionar información correcta acerca de la relación entre las materias que deben cursar los estudiantes.

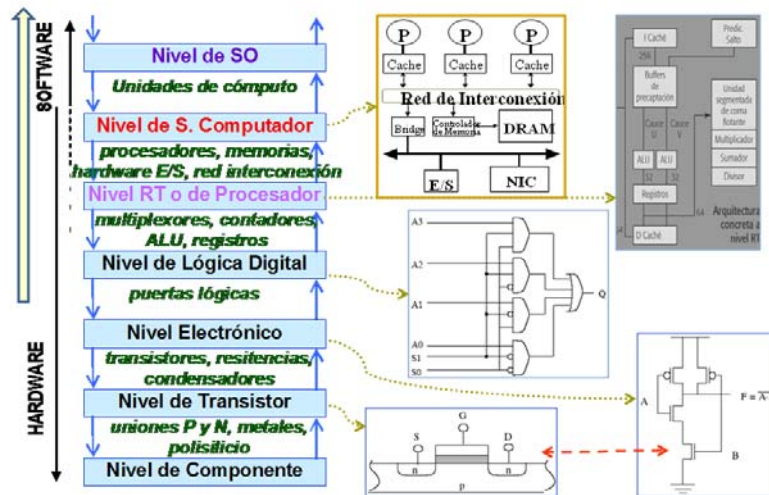


Figura 1. Niveles de abstracción de un computador

Desde esta perspectiva, conocer la interacción entre los niveles de descripción de un computador es fundamental puesto que en un mercado global en el que la competitividad es despiadada, no basta con que las cosas funcionen sino que deben hacerlo de la forma más eficiente posible. En el aprendizaje de cada asignatura, debe fomentarse el estudio interrelacionado de sus contenidos junto con los de otras asignaturas, y plantear prácticas y problemas que pongan de manifiesto la

interrelación existente entre los distintos niveles y la importancia de tenerlos en cuenta en el diseño óptimo de plataformas y aplicaciones.

Por otra parte, existe otra razón más para fomentar el aprendizaje interrelacionado de asignaturas en los distintos cursos de una carrera. La elaboración de los Planes de Estudios no tiene en cuenta el número de asignaturas entre las que los estudiantes deben distribuir su tiempo. El número de exámenes y de actividades a realizar puede ser tan elevado que dificulte la posibilidad de profundizar en los conceptos y asentar lo aprendido. Por ejemplo, se podrían hacer estimaciones de tiempo mínimo de cómputo y perfil de uso de la memoria (fallos de cache, localidad de acceso a memoria, etc.) en asignaturas de Arquitectura de Computadores impartidas en quinto curso de la Ingeniería Informática, utilizando alguno de los algoritmos de procesamiento de imágenes estudiados en la asignatura Modelos de la Inteligencia Artificial. La interrelación entre asignaturas debería ser un objetivo importante desde el punto de vista de la coordinación docente, y no solo entre asignaturas impartidas por una misma área o departamento.

La Sección 2 de este trabajo se proporciona una descripción resumida de los perfiles propuestos por ACM e IEEE para los estudios de Informática. En la Sección 3 se analiza la presencia de las asignaturas de Estructura y Arquitectura de Computadores en dichos perfiles. Finalmente, la Sección 4 justifica el interés del perfil de Ingeniería de Computadores y la Sección 5 es la conclusión de este artículo.

2 Los perfiles de la Informática según ACM e IEEE

En el Computer Curricula de 2005 de ACM e IEEE [3] se consideran cinco perfiles (o disciplinas de computación según la propia terminología del informe) como significativos actualmente. El criterio para establecerlos ha sido la existencia de guías curriculares publicadas para el perfil correspondiente por una o más sociedades científicas o profesionales. Estos perfiles eran los de *Ingeniería de Computadores (CE, de sus siglas en inglés)*, *Ingeniería del Software (SE)*, *Tecnología de la Información (IT)*, *Sistemas de Información (IS)* y *Ciencia de la Computación (CS)*. El título de Grado en Ingeniería Informática de la Universidad de Granada (aprobado oficialmente en 2010), incluye las especialidades de *Ingeniería de Computadores*, *Ingeniería del Software*, *Tecnologías de la Información*, *Sistemas de Información*, y *Computación y Sistemas Inteligentes*. Como se puede ver, salvo por el uso del plural en el perfil de *Tecnología de la Información* y por el añadido de *Sistemas Inteligentes* y el uso de *Computación* en lugar de *Ciencia de la Computación*, las cinco especialidades del Grado en Ingeniería en Informática de la Universidad de Granada coinciden con los perfiles de la ACM/IEEE. A continuación se describe brevemente el ámbito de cada uno de estos perfiles, de los que puede encontrarse una información más amplia en [3].

La *Ingeniería de Computadores* abarca el diseño y fabricación de computadores y sistemas basados en computador, e implica el estudio del software, el hardware, las comunicaciones y su interacción. En [3] también se hace referencia explícita a que el desarrollo de sistemas embebidos en dispositivos como teléfonos móviles, cámaras,

lectores de DVD, sistemas de alarma, instrumental médico, etc. integra elementos hardware y software y forma parte del ámbito de la ingeniería de computadores.

Los *Sistemas de Información* se centran en la integración de las soluciones proporcionadas por las tecnologías de la información en los procesos empresariales, para la consecución eficiente de sus objetivos. Desde este perfil, se hace énfasis en la información y se contempla la tecnología como un instrumento para generar, procesar, y distribuir información. Por tanto, un ingeniero informático con este perfil se encargaría de la determinación de los requisitos de los sistemas de información de una organización y de su especificación, diseño e implementación.

La *Tecnología de la Información* se centra en los aspectos específicos de la implementación de los sistemas de información de las empresas, como son su funcionamiento correcto y seguro, su mantenimiento, y su actualización. Surge como complemento al perfil de Sistemas de Información, para centrarse más en los aspectos de la tecnología que en los relativos a información. Los especialistas en Tecnologías de la Información son los encargados de la selección de hardware y del software adecuado para su empresa u organización, de integrarlo, adaptarlo y mantenerlo. Como ejemplos de actividades incluidas en este perfil, en [3] se citan la instalación de redes y componentes de comunicación, el diseño de páginas web, el desarrollo de recursos multimedia, y la planificación y gestión del ciclo de vida de la tecnología utilizada.

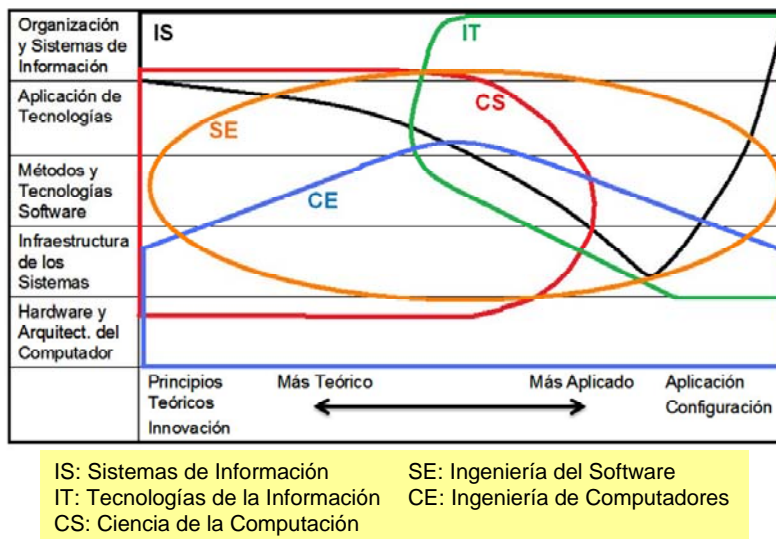


Figura 2. Aproximación gráfica del ámbito de los distintos perfiles (adaptada de [3]).

La *Ingeniería del Software* se encarga del desarrollo y mantenimiento del software, asegurando su funcionamiento ajustado a las especificaciones, fiable, eficiente, y fácil de mantener. Usualmente deben tenerse en cuenta factores como el tamaño, el coste, y el comportamiento seguro del software en aplicaciones críticas.

La *Ciencia de la Computación* abarca desde los fundamentos teóricos y algorítmicos hasta los avances en robótica, visión por computador, sistemas

inteligentes o bioinformática. Este perfil incluye facetas que van desde el diseño e implementación de software desde nuevas aproximaciones, hasta la búsqueda de nuevas formas de utilizar los computadores y de resolver eficientemente problemas de computación como la forma de almacenar información en las bases de datos, de enviar información a través de las redes, etc. La base teórica que se supone en esta especialidad les permitiría encontrar las aproximaciones que proporcionan mejores prestaciones, y adaptarse a nuevas ideas y tecnologías.

La Figura 2, realizada a partir de información proporcionada en [3], ilustra los aspectos comunes y diferenciales de cada uno de los perfiles. En el eje vertical se indican los distintos niveles desde los que se puede contemplar el computador: el hardware y la arquitectura; la infraestructura de los sistemas; las tecnologías y métodos del software; las aplicaciones; y los aspectos organizativos y de sistemas de información. El eje horizontal pone de manifiesto la existencia de actividades más relacionadas con los fundamentos y las bases teóricas frente a las más aplicadas.

Según el punto de vista que se refleja en la Figura 2, el perfil de Ingeniería de Computadores es el que abarca de forma más profunda y completa el estudio del hardware y la arquitectura del computador y la de la infraestructura de los sistemas. Según [3], el interés en el software y en las aplicaciones de este perfil se concentra en los aspectos más directamente relacionados con el desarrollo de dispositivos integrados. El perfil de Ciencias de la Computación se concentra en los aspectos teóricos (generan capacidades pero no gestionan su uso) y solo abarca una pequeña parte de los niveles relativos a sistemas de información y al hardware y la arquitectura. El perfil de Sistemas de Información abarca los aspectos de organización e información en toda su amplitud y baja hasta el nivel de infraestructura de los sistemas en sus aspectos más relacionados con la aplicación y puesta en marcha de los mismos. Por ejemplo, a menudo los especialistas en sistemas de información configuran bases de datos o desarrollan aplicaciones basadas en productos disponibles para satisfacer las necesidades de su empresa. Los especialistas en Sistemas de Información se solapan bastante con los especialistas en Tecnologías de Información, aunque estos se concentran más en los aspectos relacionados con la configuración y mantenimiento de la tecnología informática de que dispone una empresa u organización. La Ingeniería del Software debe abarcar todos los aspectos relacionados con los métodos y las tecnologías software, pero en su objetivo de proporcionar software eficiente y al estar involucrada en el desarrollo de grandes proyectos software también debe abarcar áreas muy amplias de las aplicaciones y de la infraestructura de los sistemas.

Según esta descripción extraída de [3], los perfiles descritos abarcan aspectos relacionados con las plataformas de cómputo, ya sea desde un punto de vista teórico o más bien desde una perspectiva aplicada, aunque solo en el nivel de infraestructura del sistema (únicamente el perfil de Ciencia de la Computación toca aspectos relacionados con los fundamentos del hardware y la arquitectura). Desde nuestro punto de vista, eso no significa que en esos perfiles no se deban incluir contenidos relacionados con el hardware y la arquitectura del computador. Así, la sección siguiente proporciona más detalles de los elementos propios de la Estructura y la Arquitectura del Computador que deben abordarse en los distintos perfiles.

3 La estructura y arquitectura en Ingeniería Informática

En la Tabla 3.1 que se muestra en [3], se proporcionan los pesos que, en cada perfil deben dedicarse a un total de 40 tópicos que incluye la Informática. Sobre un peso máximo de 5, a la Estructura y Arquitectura del Computador deben dedicarse pesos entre 1 y 2 en los perfiles de Tecnologías de la Información y Sistemas de información, y entre 2 y 4 en los de Ciencia de la Computación e Ingeniería del Software. A continuación se indican alguno de los aspectos que consideramos que atañen al estudio de la Estructura y Arquitectura del computador en cada uno de los perfiles descritos en [3]:

Ciencia de la Computación. Una de las facetas del trabajo del especialista en este ámbito consiste en la búsqueda de la mejor forma de almacenar información en bases de datos, o de enviar datos a través de redes. Cualquier referencia a las prestaciones de una tarea supone tener en cuenta la forma en que el computador la implementa, y por tanto requiere utilizar ciertos conceptos del hardware y la arquitectura de los computadores. Así, hay que tener nociones acerca de la forma de almacenar y acceder a la información en discos, de las alternativas para realizar las E/S en el computador, de la interfaz de red a través de la que se llevan a cabo las comunicaciones, etc.

Sistemas de Información. Puesto que este perfil contempla la tecnología como un instrumento para procesar y distribuir la información, el conocimiento de las oportunidades que ofrecen las plataformas de cómputo y las tendencias en la evolución de las mismas es esencial para este perfil. Planteamientos como la consolidación de servidores a través de técnicas de virtualización o las posibilidades que abre el “cloud computing” sin duda conciernen a los sistemas de información.

Tecnologías de la Información. Los especialistas en este perfil son los encargados de configurar, mantener, actualizar, y garantizar el funcionamiento seguro de las plataformas de cómputo en las empresas. Por tanto, deben tener conocimientos, fundamentalmente prácticos, de la Estructura y la Arquitectura de los computadores, principalmente de servidores. Tópicos como los consolidación de servidores a través de técnicas de virtualización, o las posibilidades que abre el “cloud computing” también conciernen a los especialistas en tecnologías de la información.

Ingeniería del Software. La evaluación de la eficiencia del software implica ser consciente de la forma en que los programas utilizan los recursos del computador. En el caso de aplicaciones críticas, donde se tienen requisitos de seguridad o de tiempo real, es imprescindible el conocimiento de la Estructura y la Arquitectura del computador.

Para concretar más la necesidad de estudiar las plataformas de cómputo en ámbitos distintos al de Ingeniería de Computadores, en la Tabla 1 proporcionamos preguntas que deberían ser capaces de contestar los egresados de los perfiles contemplados en el Grado de Ingeniería Informática en la Universidad de Granada. Los estudiantes podrán responderlas gracias a los contenidos de las asignaturas de Estructura y Arquitectura de Computadores. Definir un conjunto de preguntas pertinentes es esencial para el aprendizaje y la modificación de los modelos mentales erróneos con que los estudiantes se enfrentan a una asignatura. De hecho, muchos especialistas en cognición consideran que no se puede aprender hasta que no se han formulado las preguntas adecuadas debido a que dichas preguntas permiten indexar la información que tenemos en memoria cuando buscamos la correspondiente respuesta [2].

Algunas de las preguntas incluidas en la Tabla 1 se refieren a predicciones acerca de la situación futura de la tecnología. Para un ingeniero es importante disponer de un modelo del cambio plausible. Conocer los paradigmas de su disciplina. Éstos definen lo comúnmente aceptado como posible y lo que se puede esperar en un futuro. El resultado de cualquier proyecto de ingeniería debe competir en un mercado futuro y, por tanto, se necesitan estimaciones del porvenir. Muchas de las preguntas que deben ser capaces de responder los estudiantes están relacionadas con esta situación. ¿Cuántos Teraflops proporcionarán los microprocesadores dentro de cinco años?. ¿Qué ancho de banda de comunicación proporcionarán las redes?. ¿Qué barreras hay que superar para alcanzar estos objetivos?. ¿Qué arquitecturas serán las dominantes?. ¿Existirán herramientas eficientes para generar códigos optimizados en ese tipo de máquinas?. Pero no sólo es importante conocer los entresijos de la propia especialidad. Es importante mantener el contacto con la realidad externa porque hay que prever las necesidades que generarán demandas de aparatos significativas y definirán las “killer applications” y las arquitecturas de cómputo más eficientes. En la mesa redonda de empresas celebrada en la Jornada de Coordinación Docente y de Empresas del Departamento de Arquitectura y Tecnología de Computadores de 2010 todos los representantes de las empresas participantes coincidieron en que los perfiles de ingenieros más solicitados debían incluir los elementos que caracterizan una inteligencia emocional que facilite el trabajo en equipo, y actitudes positivas en cuanto a la resolución de problemas. También pusieron de manifiesto la necesidad de conectar con lo que demandan los mercados.

Tabla 1. Algunas preguntas relacionadas con el hardware, la arquitectura y los sistemas que, aparte de al perfil de CE, conciernen a otros perfiles

Preguntas	Perfiles
¿Cuánto cuesta una hora de CPU? ¿Es mejor comprar un equipo o alquilar su uso?	IT IS
¿Es posible ejecutar una aplicación dada dentro de los límites de tiempo prefijados? ¿Dónde están los cuellos de botella? ¿Es posible conseguir los requisitos establecidos mejorando el software o hay que pensar en otra plataforma?	SE IT CS
¿Qué capacidad de disco es necesaria para las bases de datos de una empresa dada? ¿Qué previsiones de crecimiento de necesidades existen para el futuro? ¿Qué sistemas de ficheros y organización de E/S pueden satisfacer los requisitos de tiempo de acceso y ancho de banda?. ¿Qué anchos de banda se esperan en cinco años?. ¿Qué tiempos mínimos de lectura de datos de disco son usuales actualmente?. ¿Cuáles se podrán alcanzar en cinco años?	IS IT
¿Qué arquitecturas de procesadores podrán aprovechar las mejoras tecnológicas futuras? ¿Qué características de las arquitecturas paralelas se pueden aprovechar en nuevos paradigmas de cómputo?. ¿Qué nuevos paradigmas serán más convenientes para programar las nuevas arquitecturas?	CS
¿Es capaz el compilador disponible de aprovechar las características de la arquitectura del procesador para conseguir un código eficiente en una aplicación determinada?	CS SE
¿Es capaz la interfaz de red utilizada por el sistema operativo de aprovechar el ancho de banda de red que interconecta los equipos de la empresa? ¿Dónde están los cuellos de botella que limitan las prestaciones de comunicación de mi sistema?	SE IT

A continuación, en la Sección 4 ofrecemos algunos argumentos para motivar la elección de la especialidad de Ingeniería de Computadores para los estudiantes del Grado en Ingeniería Informática.

4 Motivos para estudiar Ingeniería de Computadores

La evolución tecnológica y de la arquitectura nos ha llevado hacia microprocesadores con varios núcleos de procesamiento. La necesidad de mantener cuota de mercado doblando las prestaciones cada dos años para seguir con lo prefijado con la ley de Moore, y las dificultades para mantener ese ritmo aprovechando el paralelismo entre instrucciones en un núcleo han determinado que el incremento en el número de transistores en los circuitos integrados se aproveche más para aumentar el número de núcleos de procesamiento en el microprocesador que para diseñar núcleos de procesamiento más complejos que permitan terminar más instrucciones por ciclo. Por un lado, es difícil diseñar microarquitecturas que puedan doblar el número de instrucciones finalizadas por ciclo cada dos años. Esto es debido a las limitaciones en la potencia que pueden consumir y la relación entre frecuencia a la que funcionan los transistores y retardo mínimo entre módulos que actúan consecutivamente. Aparte de las limitaciones en la frecuencia a la que pueden funcionar los procesadores para mantener unos consumos razonables, a medida que se utilizan transistores capaces de conmutar a mayores frecuencias, el número de ciclos de retardo entre etapas consecutivas aumentaría en microarquitecturas complejas, donde los retardos no disminuyen al mismo ritmo que las dimensiones de los transistores. El intento de resolver estos problemas transfiriendo al compilador la responsabilidad de extraer el paralelismo entre instrucciones para evitar la complejidad del hardware asociado a la planificación dinámica de instrucciones de los superescalares no parece que haya sido totalmente efectiva para mantener el ritmo de mejora de prestaciones en los microprocesadores.

No obstante, aunque con los microprocesadores con arquitecturas multinúcleo mantener el ritmo de mejora de las prestaciones que fija la ley de Moore ha pasado a ser un problema tecnológico (mientras se mantenga el ritmo adecuado en el incremento de transistores por circuito integrado, se podrá mantener el incremento de núcleos de procesamiento por microprocesador y con ello el incremento de prestaciones) en realidad estamos hablando de las prestaciones máximas que se pueden alcanzar. Otra cosa es conseguir que las aplicaciones puedan aprovechar las posibilidades de los microprocesadores multinúcleo, que además caminan hacia arquitecturas heterogéneas con núcleos de propósito específico como las GPU (“Graphic Processing Units”) o los NP (“Nework Processors”). Este es, precisamente, un reto al que nos enfrentamos ahora, constituye un tópico importante dentro de la Ingeniería de Computadores, y determinará una parte importante del trabajo en Informática.

Además de la exigencia de que una aplicación aproveche el paralelismo del microprocesador multinúcleo en el que se ejecuta, el uso eficiente de los recursos hardware se contempla como algo crucial dentro de una economía sostenible. Internet y las tecnologías de virtualización que permiten distribuir los recursos de una plataforma entre distintas aplicaciones clientes también son cruciales y han generado una línea de negocio muy prometedora. Se trata del “cloud computing”. La posibilidad de acceder a través de Internet a los recursos de cómputo y almacenamiento que se necesitan en cada momento, definiendo un entorno virtual dinámico que aprovecha servidores remotos compartidos con otros usuarios es atractiva, tanto desde el punto de vista del coste, como del respeto con el medio

ambiente, dado que compartir ordenadores podría conseguir una reducción del 70% en el gasto energético mundial de las TIC. Todas las actividades relacionadas con la implantación eficiente del “cloud computing”, entre las que es crucial la seguridad de acceso y mantenimiento de la información, también definirán gran parte del negocio informático. Existen previsiones de crecimientos del 28% para el mercado del “cloud computing” a nivel mundial, hasta alcanzar volúmenes de 126000 millones de dólares en 2012 (IBM), y niveles del mercado español en torno a los 1000 millones de euros en 2014 (IDC). Todas estas cuestiones también se abordan en el perfil de Ingeniería de Computadores.

Muchas aplicaciones tienen unos requisitos de movilidad, tamaño, consumo, prestaciones, y precio, tan exigentes, que precisan arquitecturas de cómputo específicas. Por ejemplo, los teléfonos móviles de cuarta generación necesitarían una capacidad de cómputo superior a los Tops (10^{12} operaciones por segundo) con un consumo de menos de 1 vatio. En [4] se muestra cómo una arquitectura programable embebida podría aproximarse a estos niveles. Las mejoras en las tecnologías de circuitos integrados, incluyendo los circuitos reconfigurables, y la disponibilidad de potentes y asequibles herramientas de diseño han cambiado el panorama en el ámbito de la Ingeniería de Computadores, permitiendo el diseño y desarrollo de sistemas embebidos en dispositivos muy diversos. El diseño de sistemas de cómputo en este ámbito puede escapar del dominio de las grandes empresas de hardware, de forma que pequeñas y medianas empresas más cercanas, y en algunos casos más estrechamente relacionadas con la aplicación demandada, puedan aprovechar su capacidad de innovación. Por tanto, el trabajo en el diseño y la implementación de arquitecturas de cómputo y comunicación que se aborda en el perfil de Ingeniería de Computadores, también podría generar riqueza en nuestro contexto socio-económico.

Tabla 2. Correlación entre perfiles según competencias
(descritas en la Tabla 3.3 de [3])

Correlación	CE	CS	IS	IT	SE
CE	1.00	0.05	-0.68	-0.57	0.09
CS	0.05	1.00	-0.18	-0.27	0.75
IS	-0.68	-0.18	1.00	0.87	-0.11
IT	-0.57	-0.27	0.87	1.00	-0.20
SE	0.09	0.75	-0.11	-0.20	1.00

Para terminar, en [3] se proporciona una tabla (la Tabla 3.3. concretamente) en la que se relacionan las capacidades relativas de los graduados con los diferentes perfiles descritos en la Sección 2, asignando un peso que va desde 0 a 5 a un total de 59 competencias agrupadas en 11 áreas que van desde los algoritmos, a las bases de datos, pasando por el hardware y los dispositivos, la interfaz humano-computador, los sistemas inteligentes, las redes y las comunicaciones, etc. Entre todos los perfiles, el de Ingeniería de Computadores es el que tiene ceros en menos competencias. Concretamente, solo lo tiene en la de “diseño de planes corporativos de información”. Además, si se determina la correlación entre los distintos perfiles en cuanto a competencias, se puede construir la Tabla 2, donde se puede comprobar que el perfil de Ingeniería de Computadores es el que menos correlacionado está con cualquier otro perfil. Se observa que existen mayores correlaciones entre los perfiles de

Ingeniería del Software y Ciencia de la Computación, y entre los de Sistemas de Información y Tecnologías de la Información.

5 Conclusión

Desde el punto de vista profesional, el perfil de la Ingeniería de Computadores es actualmente uno de los mejor definidos y más claramente identificado a nivel académico. En Europa, y sobre todo en EE.UU. se tiene una idea bastante clara de qué Universidades tienen un programa de formación de Ingenieros de Computadores [3]. Como se puede observar en la Figura 2, es el único perfil que abarca totalmente el ámbito del hardware y la arquitectura del computador (el perfil de ciencia de la computación sólo incluye algunos aspectos de este ámbito). Esta circunstancia es una ventaja que debería incentivar la demanda de este perfil. Sin embargo, puede parecer que, actualmente, el tejido económico que define nuestro contexto no parece que sea capaz de crear un número considerable de puestos de trabajo con este perfil. Esto no significa que no existan posibilidades de negocio e innovación para los titulados con este perfil en nuestro entorno socio-económico. La economía global en que nos encontramos y la accesibilidad a herramientas hacen posible el desarrollo de nuevas actividades empresariales que pueden contribuir al desarrollo de nuestra tierra. Este es uno de los perfiles que pueden resultar más atractivos para los estudiantes emprendedores que quieren desempeñar una actividad profesional basada en el desarrollo y la explotación de dispositivos de cómputo (y de sus aplicaciones). En el artículo “Profesiones para la próxima década” (Manuel Ángel Méndez, El País, 2/1/2011) [5] se citan una serie de perfiles profesionales para los que se prevén demandas significativas a medio y largo plazo (entre estos está precisamente el de desarrollador de interfaces robóticas). En su mayor parte requieren una formación que integre conocimientos de nichos diversos y currículos personalizados. El estudio de la interacción entre tecnología, aplicaciones, hardware, organización y arquitectura del computador, continuamente presente en la formación del Ingeniero de Computadores, participa de ese paradigma.

Referencias

1. Ortega, J.; Anguita, M.; Prieto, A.: “Arquitectura de Computadores”. Thomson-Paraninfo, 2005.
2. Bain, K.: “Lo que hacen los mejores profesores universitarios”. Publicacions Universitat de València, 2006.
3. ACM/AIS/IEEE-CS (The Joint Task Force for Computing Curricula 2005): “Computing Curricula 2005. The Overview Report”. Septiembre de 2005. (<http://www.acm.org/education/curricula-recommendations>)
4. Woh, M.; Mudge, T.; Chakrabarti, C.: “Mobile Supercomputers for the Next-Generation Cell Phone”. IEEE Computer, pp.81-85. Enero, 2010.
5. http://www.elpais.com/diario/negocios/index.html?d_date=20110102

Instrucciones para Autores

Enseñanza y Aprendizaje de Ingeniería de Computadores es una revista que se publica anualmente y se difunde tanto electrónicamente como en papel. Los artículos remitidos para su evaluación pueden estar escritos en castellano o inglés y tendrán un máximo de 12 páginas con el formato descrito en la dirección web: http://atc.ugr.es/noticias/formato_contribucion_jcde_2010.doc.

El correspondiente fichero *.pdf* debe enviarse a la dirección de correo electrónico jortega@ugr.es.

Los artículos deben abordar, tanto contenidos relacionados con la docencia universitaria en general, como con la docencia de asignaturas específicas impartidas por las áreas de conocimiento involucradas en estudios relacionados con la Ingeniería de Computadores, y también pueden aspectos relativos a las competencias profesionales y la incidencia de estos estudios en el tejido socio-económico de nuestro entorno.

En particular, se anima a antiguos alumnos de los estudios de Informática y a estudiantes de grado y posgrado a que envíen colaboraciones relacionadas con sus experiencias al cursar asignaturas relacionadas con la Ingeniería de Computadores, sugerencias, propuestas de mejora, etc.

Teaching and Learning Computer Engineering

Journal of Educational
Experiences on Computer
Engineering

March 2011, Number 1
ISSN XXXX-XXXX

