

Actas XVIII JENUI 2012, Ciudad Real, 10-13 de julio 2012
I.S.B.N. 10: 84-615-7157-6 | I.S.B.N. 13:978-84-615-7157-4
Páginas 319-326

Programación de prototipos físicos como herramienta formativa en Informática

Jesús Salido,[‡] M^a Gloria Bueno, Óscar Déniz, M^a del Milagro Fernández
Dpto. Ing. Eléctrica, Electrónica, Automática y Comunicaciones, Univ. de Castilla-La Mancha

[‡]E.S. de Informática, P^o de la universidad 4, 13071 Ciudad Real

E.T.S. de Ingenieros Industriales, Avda. Camilo José Cela 3, 13071 Ciudad Real

{Jesus.Salido,Gloria.Bueno,Oscar.Deniz,MMilagro.Fernandez}@uclm.es

Resumen

En este trabajo se presenta una experiencia docente donde los autores emplean prototipos físicos controlados por computador, réplicas a escala de un sistema real, para motivar el estudio y aprendizaje de materias relacionadas con la Informática Industrial. Los autores plasman su experiencia, superior a diez años, empleando el sistema de construcción de prototipos fischertechnikTM en las prácticas de laboratorio de la asignatura de Automatización Industrial integrada en la titulación de Ingeniería en Informática. La exposición establece el contexto de trabajo, motivación y planteamiento de la experiencia docente adaptada al EEES (Espacio Europeo de Educación Superior), para finalizar con los resultados obtenidos, su discusión y las conclusiones.

Summary

This paper presents an educational experience in an undergraduate course on Informatics where the authors use computer controlled physical prototypes, replicas of real systems, to motivate the study and learning of industrial automation topics. The authors describe an experience of over ten years using the fischertechnikTM prototyping system in the lab of Industrial Automation course within a Computer Science degree. The exposition sets the work context, motivation and implementation for the academic experience within the EHEA (European Higher Education Area).

Palabras clave

Informática industrial, robótica, control por computador, programación dirigida por eventos, aprendizaje basado en proyectos.

1. Antecedentes

En este trabajo se expone una experiencia docente, llevada a cabo por los autores, en los últimos diez años (desde el curso académico 1998/99). Sobre todo interesa destacar el uso de prototipos físicos controlados por computador (con un referente real) para realizar prácticas de laboratorio en materias afines a la Ingeniería de Sistemas y Automática (ISA). Para la construcción de los prototipos se emplea el material comercializado por la firma alemana fischertechnik [3] (FT). En este documento se incluyen detalles prácticos que permiten reproducir esta experiencia docente en materias con un perfil similar.

1.1. Contexto docente

Aunque los métodos y experiencias han sido empleadas por los autores en diversas materias, este trabajo se centra en la asignatura de Automatización Industrial impartida en la Escuela Superior de Informática (ESI, Univ. de Castilla-La Mancha) en la titulación de Ingeniería Superior en Informática (ISI). Esta asignatura, de carácter obligatorio, se imparte en el segundo cuatrimestre de 4^o curso de la titulación con una carga lectiva de 60 horas totales de teoría y 30 de prácticas de laboratorio. Por tratarse de una asignatura de un plan de estudios anterior a la implantación del Espacio Europeo de Educación Superior (EEES), su carga docente se expresa en número de horas totales impartidas (9 créditos no ECTS). Sin embargo, en los últimos años ha sido progresivamente adaptada e incluida en un proyecto de innovación docente, financiado por la UCLM, para poner en práctica los métodos docentes impulsados en el EEES [7].

El número promedio de estudiantes de la asignatura en los últimos cinco años ha ido de 38, correspondiendo a un grupo de teoría y dos de laboratorio (con un máximo de 24 alumnos por grupo).

1.2. Motivación

Ante la propuesta de una asignatura de Automatización Industrial para Ingenieros Informáticos surgió la necesidad de motivar a éstos en una materia que tradicionalmente se ha reservado a otras áreas de ingeniería como la electromecánica, la electrónica, la automática y la ingeniería de sistemas. Así, no era extraño ser cuestionado por los estudiantes sobre la importancia de este tipo de materia en el currículum de un ingeniero informático. Ante la falta de motivación señalada, se optó por un enfoque práctico en el que destacase el papel de las prácticas de laboratorio [2]. Dicho enfoque ya contaba con una buena acogida académica en materias afines [11].

Los objetivos del aprendizaje se centran en la adquisición de conocimientos sobre los elementos principales de un sistema automatizado (actuadores, sensores, controladores, comunicaciones y programación) en paralelo con la experimentación sobre un prototipo físico con un referente real, por ejemplo: una estación automatizada de montaje, mecanización, célula robotizada, etc. De este modo, el estudiante puede relacionar inmediatamente los conceptos estudiados con la realidad.

El planteamiento propuesto requiere un gran esfuerzo inicial para adaptarlo al contexto institucional en el que está prevista su aplicación. De este proceso previo se aprendieron una serie de lecciones, tanto genéricas como específicas, que podemos resumir a continuación:

- Las restricciones temporales son las de mayor impacto en la aplicación de las propuestas docentes.
- Limitar el margen de improvisación. Las actividades de clase y laboratorio deberían estar planificadas con margen para la ocurrencia de imprevistos.
- Orientar las actividades propuestas por objetivos de aprendizaje claros y viables.
- Proporcionar manuales detallados que guíen al estudiante en sus actividades, dejando claros los márgenes para la creatividad e innovación de los alumnos. Siempre se debería incluir guías que establezcan los criterios de evaluación de cada actividad, es decir, las «reglas del juego».
- Simplificar los prototipos físicos propuestos para que sean robustos y sencillos.

1.3. Adaptación al EEES

El proceso de convergencia hacia el EEES supone tanto un replanteamiento del proceso de enseñanza/aprendizaje como de la evaluación de los estudiantes [7]. Así, ahora el modelo está centrado en el aprendizaje, y la evaluación en ofrecer el nivel de las competencias adquiridas por el alumno. Para la asignatura de Automatización Industrial la adaptación mencionada ha supuesto una ligera reestructuración de actividades del curso para su adecuación a la métrica nueva de los créditos ECTS. A continuación se indica cómo quedan redistribuidas las 90 horas lectivas (60 teoría y 30 laboratorio, en el modelo previo al EEES) con que cuenta la asignatura:

- *Lecciones magistrales* (32 hrs.). Clase tradicional en el aula con apoyo de transparencias.
- *Sesiones de laboratorio* (42 hrs.). Sesiones de trabajo en el laboratorio en las que se construye un prototipo físico y se resuelven los problemas propuestos para su automatización.
- *Taller-presentación de trabajo* (5 hrs.). Los alumnos exponen, en parejas, un trabajo (realizado a lo largo de todo el curso) sobre un caso práctico de automatización ligado preferentemente a la realidad socioeconómica de la comunidad.
- *Visita guiada* a una planta automatizada en producción (2 hrs.). Los alumnos acompañados del profesor visitan las instalaciones de COFARCIR (Ciudad Real), una empresa de distribución farmacéutica.
- *Tutorías presenciales en grupo* (5 hrs.). Destinadas a seguir el progreso de las actividades del curso (trabajos, prácticas, etc.).
- *Pruebas de evaluación presencial* (4 hrs.). Durante el curso se plantean dos pruebas escritas similares al examen tradicional.

En este proceso de adaptación descrito también se incluye la utilización de Moodle como plataforma de gestión de aprendizaje (LMS) integrado en la red telemática interna de la UCLM. Moodle se utiliza para publicar información relativa a la asignatura (guía docente, material de estudio y auxiliar, enunciados de las actividades que debe realizar el alumno, etc.) y realizar el seguimiento del proceso de aprendizaje tanto desde la perspectiva del profesor como del alumno.

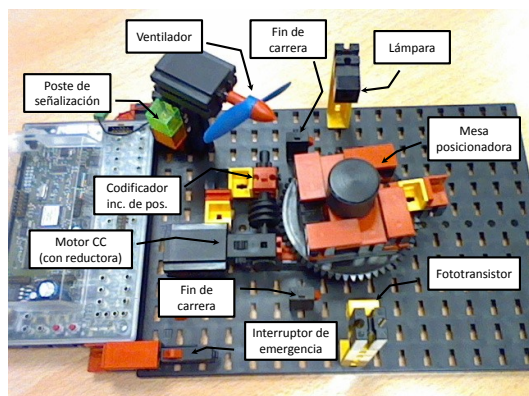


Figura 1: Prototipo de estación de secado

2. Enfoque práctico

Este trabajo se centra en destacar el papel del empleo de prototipos físicos controlados por ordenador como herramienta motivadora que potencia el proceso de aprendizaje de estudiantes en la titulación de Ingeniería Informática. En este caso no sólo se trata de programar prototipos con un referente real, el estudiante también lleva a cabo todas las fases previas a la programación, incluyendo la construcción mecánica y el conexionado eléctrico.

La propuesta pedagógica está basada en la teoría del construccionismo desarrollada por S. Papert [10] y que goza de gran popularidad en materias relacionadas con la Robótica y la Inteligencia Artificial [11]. Los autores plantean un enfoque derivado del aprendizaje basado en proyectos (PBL) [9] aunque en este caso su uso se restringe a una materia, y por tanto sin el carácter holístico que realmente debería tener una orientación PBL.

Durante el curso de Automatización Industrial se propone a los alumnos la automatización de una *estación de secado* (ver fig. 1). Dicha estación se compone de los siguientes elementos:

- Mesa posicionadora. En ella se sitúa la pieza a secar. Cuenta con finales de carrera, codificador incremental de posición y detector de pieza.
- Ventilador de secado.
- Poste de señalización luminosa y acústica.
- Interruptor de emergencia.

La construcción de la estación de secado y otros

prototipos físicos se lleva a cabo empleando las piezas de los kits fischetechnik (FT). Estos incluyen *elementos actuadores* (motores CC, lámparas de filamento y zumbador), *dispositivos sensoriales* (generador de pulsos, interruptores fin de carrera y fototransistores), *sistemas de transmisión de potencia mecánica* (caja reductora y engranajes) y otros *elementos constructivos* (ver fig. 1).

Aunque es posible trabajar con maquetas ya montadas y listas para programar, los autores defienden una metodología donde los propios estudiantes sean quienes realizan el montaje del prototipo propuesto. De este modo el alumno no sólo realiza la programación del prototipo sino que se involucra en su construcción lo que potencia el aprendizaje de conceptos no informáticos (p. ej. sensores, actuadores, conexionado eléctrico, etc.). Esta metodología exige una elección «cuidadosa» del prototipo para que su nivel de complejidad sea apropiado a los objetivos y contexto del programa formativo. En concreto, una restricción «impuesta» a los prototipos es que sea posible su construcción en un máximo de dos sesiones de laboratorio (4 hrs.). En general un usuario con cierta habilidad sería capaz de un montaje completo en menos de una hora, pero debe tenerse en cuenta que los alumnos desconocen por completo este sistema de construcción de prototipos físicos cuando se enfrentan a él y requieren un corto periodo de adaptación. Para renovar el planteamiento del laboratorio se puede emplear otros prototipos como (algunos ya utilizados en ediciones previas): acceso a aparcamiento, estación de estampado, estación clasificadora, ascensor, brazo robotizado, etc.

2.1. Desarrollo de las prácticas de laboratorio

Al plantear una asignatura en el EEES, una de las lecciones aprendidas por los autores es la necesidad de una planificación «generosa» en el tiempo destinado a las actividades que debe realizar el alumno. Una planificación demasiado «exigente» suele provocar pérdida de motivación del alumno cuando no es capaz de cumplir con los hitos marcados. Por contra, se ha observado que es mucho más realista y motivador un planteamiento basado en pocos hitos con flexibilidad en las fechas de los entregables. En la automatización de la estación de secado se han previsto cuatro hitos:

1. *Elección de herramientas software* de desarro-

llo tras el análisis general del problema. Este hito se plantea en primer lugar para asegurar que los alumnos conozcan lo antes posible las herramientas informáticas que tendrán que emplear y realicen las primeras pruebas de programación.

2. *Conexión física* de elementos (actuadores y sensores) a la interfaz FT y prueba de un subsistema del prototipo final (p. ej. célula de presencia y ventilador de secado).
3. *Programación del algoritmo de control* de la estación de secado. Aquí los alumnos ya se enfrentan al prototipo completo en el que se plantea un problema de automatización «realista» donde aplicar conceptos clave como: programación concurrente, control dirigido por eventos, temporización de eventos, etc.
4. *Integración de la estación individual en un sistema automatizado mayor*. En este caso se pide que los alumnos proporcionen, junto con los programas de control, interfaces de usuario de mayor nivel y la posibilidad de realizar la gestión de la estación de secado desde un ordenador diferente a aquel en el que se ejecuta. De este modo se motiva la importancia, en los sistemas automatizados, de las interfaces hombre máquina (HMI) y los sistemas de control supervisado y adquisición de datos (SCADA).

Los objetivos para estos cuatro hitos son concretos, públicos y conocidos por los alumnos en las primeras semanas del curso (no más allá de la 4ª) con una definición concreta de los entregables que deben prepararse. Sin embargo, aunque se proponen una fechas de entrega no se impide los envíos posteriores.

2.2. Control de prototipos fischertechnik

Para el control de prototipos se emplea una tarjeta de interfaz conectada mediante puerto serie (USB o RS-232), al ordenador que ejecuta el programa principal de control. Los prototipos FT están preparados para trabajar con cualquier interfaz capaz de suministrar los 9 V (1 A) que alimentan a los actuadores, y captar la información de los sensores disponibles [4]. También es posible recurrir a elementos actuadores con tensiones de trabajo de 24 V para construir prototipos controlados mediante un autómata programable comercial (PLC) [14].

Para las prácticas de Automatización Industrial se ha escogido el control de prototipos mediante las interfaces proporcionadas por FT [12]. Aunque son válidas varias interfaces de las comercializadas por FT, se recomienda el uso de las que permitan conexión USB con el ordenador (como la ROBO interfaz, ref. 93293, y todas las posteriores). De este modo se garantiza que los alumnos puedan emplear sus ordenadores portátiles, sin conexión RS-232, para realizar la programación de los prototipos.

Las interfaces FT permiten dos modos de programación diferentes. En el *modo conectado (online)* el programa de control de usuario se ejecuta en el ordenador auxiliar conectado a la interfaz FT. Este modo es el habitual cuando se trabaja con prototipos estáticos en su lugar de instalación (*in place*) como sucede con los empleados en las prácticas de Automatización Industrial. Por el contrario en el *modo autónomo (download)* el programa de control se descarga a la memoria *flash* de la interface y se ejecuta en ésta sin intervención del ordenador auxiliar. Este modo de programación es más apropiado para prototipos móviles como robots autónomos, aunque también en ellos es posible trabajar en modo conectado empleando conexión inalámbrica como *bluetooth* (no disponible en todas las interfaces FT). Para la programación de prototipos, en ambos modos, se comercializa el entorno de desarrollo ROBO Pro que incluye un lenguaje de programación visual del mismo nombre. Afortunadamente, para la programación en modo conectado es posible emplear otros lenguajes de programación más extendidos y conocidos por los estudiantes de Informática como: Java, C/C++, C#, Python, etc.

Para la realización de las prácticas de Automatización Industrial se considera muy importante ofrecer a los estudiantes la posibilidad de emplear los lenguajes de programación que ya conocen, y así poner en práctica los conocimientos adquiridos en otras materias ya cursadas. Esto crea sinergias muy importantes entre los contenidos nuevos que se ofrecen y los de otras asignaturas de la titulación como: programación, ingeniería del software, comunicaciones, interfaces de usuario, programación concurrente, sistemas de tiempo real, etc.

El uso de lenguajes de propósito general para la programación de las interfaces FT exige el empleo de librerías externas como `FTLib.DLL` (de Knobloch GmbH [6]) para Windows y `libft` (de Eric Andre-

sen [1]) para Linux. Los alumnos pueden elegir libremente el lenguaje de programación y el SO para desarrollar su trabajo, pero la mayoría se decanta por Windows como plataforma de desarrollo (el 93%).

El trabajo en Windows con la librería `FTLib.DLL` presenta cierto grado de dificultad debido a su bajo nivel y la escasez de documentación, únicamente disponible en inglés y alemán. Para facilitar su empleo se recurre a otras librerías propuestas por U. Muller [8] que añaden una capa de abstracción sobre `FTLib` proporcionando interfaces de programación más sencillas. En la fig. 2 se resume el esquema de trabajo empleando las librerías mencionadas para los lenguajes Java, Python y C#.

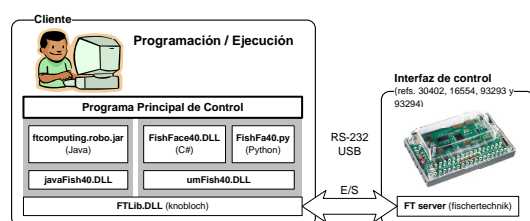


Figura 2: Programación *on-line* mediante librerías externas

Mayoritariamente los alumnos recurren al lenguaje de programación Java (66%) mientras el resto se reparte por igual entre C# y C/C++. En los dos últimos cursos se ha observado un crecimiento en popularidad de Python empezando a restar protagonismo, entre los alumnos, a lenguajes como C# y C/C++.

2.3. Herramientas desarrolladas

Para facilitar el trabajo de los alumnos e incluso ampliar las posibilidades de programación de los prototipos controlados mediante interfaces FT, los autores han desarrollado varias herramientas:

- *FTPack* [12]. Está constituida por un *plugin* para el entorno integrado de desarrollo (IDE) Eclipse que facilita la programación de prototipos FT sobre plataforma Windows empleando las librerías de U. Muller (sólo para lenguajes Java, Python y C#). Proporciona ejemplos de proyectos, documentación en español y un simulador software que permite el desarrollo y test de programas sin necesidad de tarjeta FT. Se encuentra en estado beta pero es usado por

todos los alumnos que eligen Java como lenguaje de programación.

- *CtrlWeb* [13]. Lleva el modo conectado de programación un paso más allá al proporcionar una API para la programación de aplicaciones distribuidas de control mediante el denominado *modo remoto*. Así es posible programar aplicaciones para controlar prototipos que se encuentran en un laboratorio remoto. Esta herramienta es plenamente operativa pero su uso se ha restringido a experiencias piloto.
- *EduPLC* [5]. Su objetivo es proporcionar capacidad de programación más próxima a la realidad profesional de la automatización empleando lenguajes compatibles con el estándar IEC 61131-3 (diagrama de contactos y lista de instrucciones) habituales en la programación de PLC. Actualmente continua en desarrollo y aún no ha sido empleada por los alumnos.

2.4. Recursos materiales necesarios

A continuación se realiza un breve balance del material y recursos materiales requeridos en la experiencia docente descrita en este trabajo.

Se parte de la necesidad de un aula de laboratorio equipada con puestos en los que existan tomas de corriente accesibles para conectar tanto la fuente de alimentación de la interfaz FT como el ordenador auxiliar. Es conveniente que los puestos de laboratorio dispongan de una cajonera en la que pueda almacenarse el material de laboratorio entre distintas sesiones. El laboratorio de Automatización Industrial dispone de mobiliario adaptado para 12 puestos completos (2 alumnos por puesto), reservando uno para el profesor instructor.

Para el almacenamiento del material se usan estuches de plástico con compartimentos que permiten la ordenación del material para su conveniente almacenaje. Si la adquisición de las piezas e interfaz FT se realiza separadamente de los kits disponibles es posible conseguir todo lo necesario por un coste/puesto inferior a 500€. Todo el software necesario es de libre disposición y puede obtenerse en páginas web públicas [8].

3. Evaluación al alumno

El proceso de evaluación a los alumnos permite obtener una calificación final total en la que se pondera por un lado, la evaluación de las actividades de laboratorio (25%), y por otro el resto de actividades (75%). Estas últimas incluyen 6 cuestionarios (15%, uno por sección del temario), 2 controles de evaluación (37,5%), trabajo de investigación y exposición (15%), y otras tareas (7,5% comentarios sobre lecturas propuestas, visita, etc.). Todo el proceso tiene lugar a lo largo del curso y persigue ajustarse al modelo de evaluación continua impulsado en el EEES [7]. También se mantiene el examen final como una alternativa no excluyente a la evaluación continua, aunque su ponderación sobre la calificación final es del 75% (con la evaluación final no es posible obtener una calificación total del 100%).

Es importante destacar el papel de Moodle en el proceso de evaluación continua como herramienta de seguimiento tanto desde la perspectiva del profesor como del alumno. De todos los recursos disponibles en Moodle, los más útiles han sido:

- *Foro de novedades.* Empleado como método de comunicación de profesor al conjunto de la clase. Por su parte los alumnos prefieren la comunicación directa hacia el profesor o bien el uso del correo electrónico.
- *Consultas.* Empleadas cuando es preciso recoger la opinión de la clase sobre una elección concreta (horarios, fechas de entrega, etc.).
- *Módulo de asistencia.* Gestiona la asistencia de los alumnos en las actividades presenciales.
- *Páginas web.* En ellas se publica material del curso (guía docente, páginas con enlaces a material de referencia y estudio, guiones de prácticas, etc.).
- *Directorios.* Con material del curso para descargar (apuntes, manuales, transparencias, ejemplos, etc.)
- *Cuestionarios.* Permiten realizar pruebas de test, en línea, con gran flexibilidad.
- *Actividades.* Empleadas para proponer tareas con fecha de entrega o realización.

4. Resultados y discusión

¿Cómo evaluar los resultados de la experiencia docente realizada? Esta pregunta admite una contestación desde dos perspectivas: la del profesor y la del alumno. La primera se refiere a los resultados obtenidos por los alumnos en la evaluación de los objetivos de aprendizaje, mientras que la segunda persigue obtener la valoración de los alumnos respecto del proceso formativo seguido en el curso. A continuación se discuten los resultados más relevantes desde ambos puntos de vista.

Desde su puesta en vigor, el modelo de evaluación continua ha sido cada vez más aceptado por los alumnos hasta dejar el examen final relegado a casos singulares y constituir una mera «obligación legal». El porcentaje de alumnos suspensos en la convocatoria ordinaria se ha ido reduciendo hasta obtener cifras del 100% de aprobados entre los alumnos que siguen el modelo de evaluación continua. Además, entre estos, la calificación promedio se ha elevado desde el entorno del 6 (sobre 10) al 7,5 decreciendo también la desviación estándar de las calificaciones (desde el entorno del 2 a 1). La Fig. 3 muestra la evolución de las calificaciones (promedios) y su dispersión (desviación estándar). En resumen *aprueban más alumnos, con mejores calificaciones y se obtiene unos resultados más homogéneos para el grupo.*

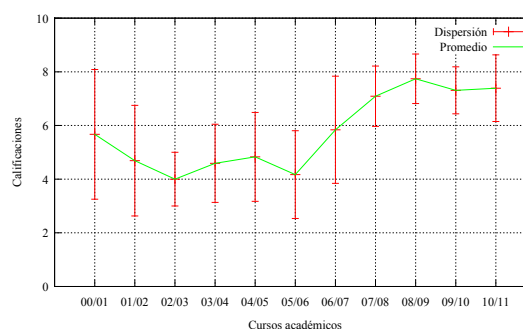


Figura 3: Evolución de calificaciones

La mejora en las calificaciones se atribuye tanto a la mejora de los procedimientos implantados tras la experiencia de sucesivos cursos como a la reducción del número total de alumnos matriculados, lo que facilita su seguimiento más individualizado.

Para intentar analizar la percepción de los alumnos sobre el proceso formativo se ha realizado al

final de curso una encuesta voluntaria en la que se pide la valoración personal sobre los siguientes aspectos de la asignatura:

1. *Actividad del curso con mejor/menor valoración.* Los estudiantes califican mayoritariamente al proyecto de laboratorio como la parte más interesante y provechosa. Por el contrario las clases magistrales son las que reciben la valoración más baja.
2. *Sugerencias de mejoras a introducir.* En general, las sugerencias de los alumnos van dirigidas a cambios metodológicos particulares. Sin embargo, se advierte que sus principales inquietudes giran en torno a una programación integral de actividades en la titulación que tenga en cuenta el conjunto de asignaturas del curso. Otra observación recurrente se refiere a la valoración positiva del uso de Moodle. Sin embargo, muestran su preocupación por las garantías de fiabilidad que pueda ofrecerse para su utilización, sobre todo en la realización de actividades evaluables realizadas en línea, como los cuestionarios.
3. *Importancia de la materia en la titulación de Ingeniería Informática.* La valoración que los alumnos hacen de la asignatura en el contexto de la titulación, aumenta de manera considerable, desapareciendo la idea de que se trata de un área accesoria o sin relación directa con la profesión del Ingeniero Informático.

5. Conclusiones

En este trabajo se presenta la experiencia docente de los autores en la asignatura de Automatización Industrial para Ingenieros Informáticos, destacando la evaluación continua del proceso de enseñanza/aprendizaje y un enfoque constructor en la realización de las prácticas de laboratorio que pasan a jugar el papel de «hilo conductor» en el desarrollo del programa orientado a competencias.

Los «frutos» principales del modelo docente seguido (ver §4) se pueden resumir en los puntos siguientes:

1. Se potencia la adquisición del conocimiento a través del ejercicio de competencias ya que mejora la valoración de los aspectos prácticos

orientados al «saber hacer» frente a los puramente teóricos.

2. Mejora la percepción que el alumno posee sobre la importancia de la materia en el contexto de la titulación cursada y la conexión «real» con el resto de materias.
3. Mejora los resultados académicos de los alumnos, descendiendo el número de «fracasos» y consiguiendo un nivel más uniforme en cuanto a la evaluación calificativa.

Respecto a las modalidades y métodos de enseñanza, en esta experiencia se destaca los siguientes aspectos:

- Mayor importancia a las clases prácticas desarrolladas en el laboratorio y al método de aprendizaje basado en proyectos. Este hecho queda reflejado en el aumento del 40% de horas destinadas a las clases prácticas y la disminución del 50% en las horas destinadas a lecciones magistrales, respecto de un planteamiento anterior al EEES.
- En las clases prácticas el alumno construye y programa un prototipo físico, con referente real. Este procedimiento motiva al alumno y le ayuda tanto a adquirir las competencias específicas de la materia como a poner en práctica competencias ya adquiridas en otras materias.
- Evaluación continua y formativa del proceso de enseñanza/aprendizaje en la que se emplean múltiples estrategias y procedimientos evaluativos: pruebas presenciales, cuestionarios en línea, memoria de laboratorio, trabajo/exposición, etc.

La experiencia docente adaptada al EEES que se ha descrito ha supuesto para los autores una fuente de aprendizaje respecto a los métodos y procedimientos más provechosos de cara a conseguir la mayor satisfacción de los estudiantes y por añadido de los frutos recogidos por éstos. A continuación se resume las principales lecciones aprendidas:

- Es muy importante una planificación integral del plan formativo que contemple todas las materias del curso. La planificación, independiente y aislada de cada asignatura conduce a desajustes en la carga de trabajo del alumno incidiendo negativamente en su motivación y desempeño.

- La planificación de actividades debería permitir cierto grado de flexibilidad contemplando la posibilidad de «incidencias» sobrevenidas que requieran reajuste tanto del calendario de actividades (entregas, pruebas presenciales, etc.), como de objetivos particulares.
- Es necesario impulsar el seguimiento particularizado del alumno por parte del profesor/es para identificar de inmediato aquellas deficiencias a subsanar. Los mejores resultados de la experiencia docente se han obtenido con grupos de trabajo en clases prácticas en torno a 20 alumnos.
- Es muy recomendable la elaboración de guías detalladas para el alumno que describan los métodos y objetivos del programa formativo.
- Para llevar a cabo este tipo de experiencia docente no se precisan grandes recursos materiales. En concreto, el material empleado en la experiencia aquí descrita no alcanza los 500€. Es importante tener presente el empleo de software libre y la reutilización de recursos de libre disposición como estrategias para reducir los costes de implantación.

Agradecimientos

Los autores desean expresar su agradecimiento a la Escuela Superior de Informática (ESI) y al Departamento de Ingeniería Eléctrica, Electrónica, Automática y Comunicaciones (IEEAC) de la Universidad de Castilla-La Mancha (UCLM) por su apoyo y soporte técnico y material en la realización de su labor académica y especialmente en lo relativo a la puesta en práctica de la propuesta docente objeto de este trabajo.

Referencias

- [1] Andresen, E., *Libft index page*, <http://defiant.homedns.org/~erik/ft/libft/index.html>
- [2] Feisel, L. D. and Rosa, A. J., *The role of the laboratory in undergraduate engineering education*, Journal of Engineering Education, 94(1):121-130, 2005.
- [3] fischertechnik, <http://www.fischertechnik.de/>
- [4] fischertechnik, *ROBO Interface. Instrucciones de manejo*, 2004.
- [5] Franco, A., *EduPLC: Entorno de programación para interfaces fischertechnik*, Proyecto Fin de Carrera, E. S. de Informática (UCLM), 2009.
- [6] Knobloch GmbH, <http://www.knobloch-gmbh.de/>
- [7] de Miguel, M. (Dir.), *Modalidades de enseñanza centradas en el desarrollo de competencias. Orientaciones para promover el cambio metodológico en el espacio europeo de educación superior*, Universidad de Oviedo, 2005.
- [8] Müller, U., *ftComputing*, <http://www.ftcommunity.de/ftComputingFinis/fishneu.htm>
- [9] Northwest Regional Educational Laboratory, *Project-Based Instruction: Creating Excitement for Learning*, 2002.
- [10] Papert, S., *The children's machine: rethinking school in the age of the computer*, BasicBooks, 1993.
- [11] Salido, J. y Sanz, J., *La enseñanza en Ingeniería de Sistemas y Electrónica mediante el laboratorio de robots autónomos*, VIII Jornadas de Enseñanza Universitaria de la Informática (JENUI), 2002.
- [12] Salido, J., Déniz, O., Bueno, G. y Fernández, M., *FTPack: un framework para la programación de sistemas físicos educativos basados en kits fischertechnik*, VI Jornadas CEA de Enseñanza a través de Internet-Web de la Ing. de Sistemas y Automática (EIWISA), 2010.
- [13] Salido, J., Lillo, A., Déniz, O. y Bueno, G., *CtrlWeb: Una herramienta de programación para telecontrol de sistemas físicos educativos*, Revista Iberoamericana de Automática e Informática Industrial (RIAI), 8(1):89-99, 2011.
- [14] Staudinger GmbH, <http://www.staudinger-est.de/>