

Una aproximación a la evaluación de laboratorio en programación de sistemas

C. Glez-Morcillo, F. Jurado, D. Vallejo
Departamento de Tecnologías y Sistemas de Información
Universidad de Castilla-La Mancha
Paseo de la Universidad, 4
13071 Ciudad Real

{Carlos.Gonzalez, Francisco.Jurado, David.Vallejo}@uclm.es

Resumen

Tras la implantación del proceso de Bolonia en España, los nuevos planes de estudio asociados a las titulaciones de Informática cuentan con asignaturas relacionadas con la programación de sistemas (ej. Sistemas Operativos). La adquisición de determinadas competencias prácticas, específicas de dichas asignaturas, resultan esenciales para que el estudiante sea capaz de asimilar y manejar los conceptos fundamentales asociados a las mismas.

Los mecanismos de evaluación y calificación clásicos pueden plantear ciertas deficiencias a la hora de validar la correcta adquisición de dichas competencias, especialmente en la parte vinculada a los aspectos prácticos. En este contexto, el presente trabajo describe una aproximación para la realización de pruebas prácticas evaluables en el laboratorio, así como una descripción sobre la metodología empleada en la construcción de dichas pruebas.

La exposición y discusión de los resultados, obtenidos como consecuencia de su aplicación en los últimos tres años en una asignatura obligatoria vinculada al estudio de los Sistemas Operativos en *Ingeniería en Informática*, ha permitido obtener conclusiones que contribuyen a afianzar la adquisición de competencias por parte del estudiante.

Summary

After the implantation of the Bologna Process in Spain, the new undergraduate studies related to Computer Science integrate courses that cover Operating Systems, Distributed Systems and Concurrent Programming. Acquiring and developing the specific practical competences of these subjects is essential so that the students are able to understand and apply the basic concepts.

Classical evaluation and grade methods may su-

ffer from some limitations when validating the adequate acquisition of the previously mentioned competences, specially those related to the practical aspects. Within this context, this work discusses the employed approach to carry out the practical assessment of the students in the laboratory and describes the adopted methodology.

The obtained results, after having carried out this approach during the last three years in a required course of Operating Systems, show how this approach has contributed to guarantee the acquisition of essential competences on behalf of the students.

Palabras clave

Sistemas Operativos; Evaluación del aprendizaje; TIC en el proceso de enseñanza-aprendizaje.

1. Introducción

Los Sistemas Operativos resultan un componente clave en cualquier sistema de cómputo. Es por esto que las titulaciones asociadas con estudios de Informática deben garantizar que los estudiantes adquieran aquellas competencias que les permitan no sólo manipularlos, sino entender tanto sus mecanismos internos de funcionamiento como el diseño e implementación de los mismos. Es más, muchas de las cuestiones abordadas en los Sistemas Operativos tienen una amplia aplicación en otros campos de la Informática, como son la programación concurrente y distribuida, el diseño e implementación de algoritmos, el desarrollo de nuevos dispositivos de interacción, la construcción de entornos virtuales, la seguridad informática, etc. [1][2].

Las metodologías de enseñanza/aprendizaje usualmente empleadas para permitir a los estudiantes adquirir los conceptos teóricos y desarrollar las habilidades prácticas asociadas con estas com-

petencias incluyen sesiones de laboratorio que posibiliten a los estudiantes experimentar con Sistemas Operativos [1][2]. Es decir, emplean el método de Aprendizaje Basado en Problemas [6] a fin de promover el aprendizaje activo centrado en el estudiante, combinando actividades teóricas y de laboratorio [5][7][9].

Por su parte, el Espacio Europeo de Educación Superior, con la introducción de los nuevos Estudios de Grado bajo las directrices del Tratado de Bolonia¹, promueve el uso de metodologías educativas y técnicas de evaluación activas, potenciando el empleo de mecanismos de calificación y certificación de habilidades y competencias que eliminen los exámenes finales como única fuente de evaluación del estudiante.

Por tanto, evaluar si los estudiantes son capaces de analizar, diseñar y construir aplicaciones de sistemas de forma robusta, segura y eficiente, resulta una tarea fundamental en el proceso de enseñanza de las asignaturas asociadas con estas competencias.

Del mismo modo, dado que el objetivo de cualquier metodología de enseñanza/aprendizaje es permitir a los estudiantes alcanzar niveles cognitivos de pensamiento altos, los mecanismos de evaluación deben tener en cuenta criterios que posibiliten la evaluación de dichos niveles cognitivos [3][4].

En este contexto, el presente trabajo describe una aproximación para la realización de pruebas prácticas evaluables en el laboratorio llevada a cabo en la Escuela Superior de Informática de la Universidad de Castilla-La Mancha, exponiendo y analizando los datos recopilados de su aplicación durante los últimos tres años en una asignatura obligatoria vinculada al estudio de los Sistemas Operativos en *Ingeniería en Informática*.

El resto del documento está estructurado de la siguiente forma. La sección 2 describe el contexto en el que se ha aplicado la metodología propuesta, haciendo especial hincapié en los procesos de evaluación y de implantación. A continuación, la sección 3 expone, desde un punto de vista general, las primeras conclusiones relativas a la aplicación del enfoque discutido en este trabajo en términos de resultados finales en la evaluación de los estudiantes. La sección 4 discute en profundidad estos resultados, prestando especial atención a su evolución du-

rante los cursos académicos en los que se ha aplicado. Finalmente, en la sección 5 se concluye el trabajo realizando una reflexión general sobre el enfoque planteado.

2. Descripción del método propuesto

En esta sección se comenzará con una descripción de las características del curso y los estudiantes sobre los que se ha aplicado a fin de mostrar la problemática a abordar, para pasar a continuación a adentrarnos en describir el método de evaluación propuesto y el modo en que ha sido implantado.

2.1. Curso y estudiantes

La propuesta que se describe en esta contribución ha sido aplicada a la asignatura de Ampliación de Sistemas Operativos de la Universidad de Castilla-La Mancha. Esta es una asignatura de tercer curso de *Ingeniería Informática e Ingeniería Técnica en Informática de Sistemas*. Antes de cursarla, los estudiantes deben haber adquirido competencias asociadas a programación, algoritmos, estructuras de datos, y conocimientos previos sobre el funcionamiento, estructura y diseño de Sistemas Operativos.

En esta asignatura debe garantizarse la adquisición de determinadas competencias prácticas. Así, el estudiante debe adquirir competencias específicas de las características, funcionalidades y estructuras de los Sistemas Operativos, destacando aspectos fundamentales de la gestión de procesos de un núcleo, desarrollo de programas multiproceso empleando un lenguaje de programación de sistemas, manejo de herramientas de desarrollo de compilación, enlace y depuración de aplicaciones de sistemas y diseño e implementación de aplicaciones que utilicen los mecanismos de gestión de concurrencia que proporciona el sistema operativo.

La acreditación y evaluación de las competencias prácticas por parte del estudiante puede resultar una tarea complicada, atendiendo al volumen de estudiantes y el tiempo requerido para la realización del seguimiento que debería realizarse según las directrices del Tratado de Bolonia.

La siguiente subsección detalla el procedimiento propuesto para analizar el progreso y evaluar los avances de los estudiantes en materia de programación de sistemas.

¹ec.europa.eu/education/policies/educ/bologna/bologna.pdf

2.2. Método y procedimiento de evaluación

Para analizar el progreso de los estudiantes se realizarán diversas sesiones evaluables de prácticas de una hora y media de duración, donde éstos deberán resolver problemas acotados de programación de sistemas (sólo un problema por sesión). Para resolver dichos problemas, deberán emplearse aquellos mecanismos y estructuras específicos cuya habilidad y manejo se desean evaluar.

Uno de los requisitos fundamentales exigidos a los problemas planteados a los estudiantes es que éstos deben proporcionar una salida determinista, es decir, independiente del orden de ejecución de las instrucciones y concurrencia de los procesos que intervengan en la solución implementada. Esto permite agilizar el procedimiento de evaluación, dado que estudiantes y profesores conocen en el acto si se ha alcanzado una solución al problema, ya sea óptima o no.

Además de la descripción del problema a resolver mediante el correspondiente enunciado, al estudiante se le proporciona gran parte del sistema a implementar (en torno al 75-80% de la codificación total). Entender el comportamiento y la finalidad del código proporcionado es otra de las habilidades evaluables.

Durante toda la sesión, los estudiantes tienen a su disposición las páginas del manual del sistema (aplicaciones, funciones de biblioteca estándar, llamadas al sistema, etc.). Al terminar la realización de la prueba práctica, los estudiantes anotan sus soluciones en papel con el objetivo de facilitar futuras consultas y reclamaciones sobre dichas soluciones.

Con esto, se pretende que el procedimiento de evaluación considere diferentes niveles cognitivos de pensamiento. Así, tomando como referencia la taxonomía de Bloom [4], los niveles abordados son:

- **Conocimiento y comprensión:** el empleo de las páginas del manual del sistema implica conocer o recordar determinadas funciones y llamadas al sistema de una manera aproximada. Saber manipularlas del modo idóneo para la resolución del problema implica haber alcanzado el nivel de comprensión.
- **Aplicación:** para resolver el problema planteado, los estudiantes deben demostrar que saben utilizar los conocimientos adquiridos aplicándolos a nuevas situaciones.

- **Análisis:** entender el código proporcionado, identificar las partes en las que se descompone el problema, etc., también constituye un componente evaluable. Ello supone que los estudiantes están trabajando en el nivel de análisis de la taxonomía.
- **Síntesis:** añadir el código necesario para alcanzar una solución al problema, integrándola con el resto del código proporcionado, implica haber alcanzado este nivel cognitivo.

Para que todos los estudiantes tengan disponibles un entorno de trabajo concreto y específico para la realización de las pruebas, así como el enunciado y el código necesarios, debe disponerse de un mecanismo que permita tener controlado tanto el entorno de trabajo como la distribución del material. En la siguiente subsección se pasarán a describir los mecanismos, servicios y procedimientos necesarios para su implantación.

2.3. Implantación

La Escuela Superior de Informática de la Universidad de Castilla-La Mancha cuenta con 207 ordenadores para las clases prácticas, repartidos en 10 laboratorios docentes². Cada aula cuenta con un hardware de ejecución específico, pero todos están basados en arquitectura x86. Los requerimientos mínimos que debe tener cada computador para pertenecer al sistema son una memoria RAM de 500 MB, una partición Swap de 1 GB, y tener una conexión de al menos 100Mbps/s (todos los ordenadores están conectados a la red utilizando *switches* de 100Mbps/s). La figura 1 muestra un esquema general de la configuración del sistema utilizado para la gestión de los exámenes.

Para el desarrollo del examen se utiliza una imagen de GNU/Linux basada en *Slackware* sin servidor X, con las herramientas básicas de programación de sistemas instaladas (*make*, *gcc*, *gdb*...), así como las páginas del manual *man*. A esta versión del sistema operativo se le ha eliminado el soporte de red, de forma que una vez cargada la imagen del sistema operativo no es posible acceder a la red (tanto a Internet como a la red local).

De esta forma no es posible que los estudiantes compartan ningún tipo de material por red. Por la

²<http://webpub.esi.uclm.es/curso/actual/espacios-docentes>

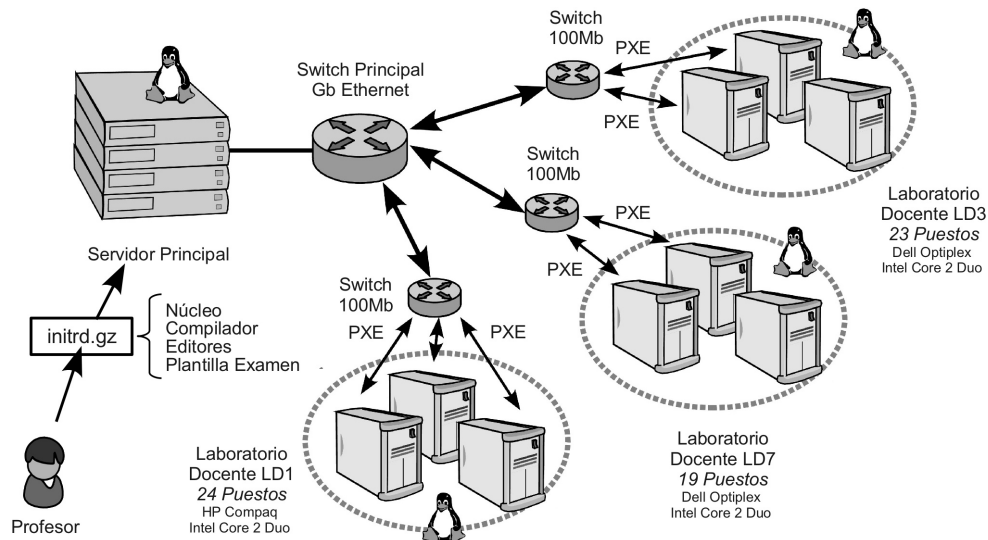


Figura 1: Esquema general de implantación del sistema de distribución de exámenes de laboratorio.

misma razón, tampoco es apropiado instalar la imagen de forma permanente en las máquinas, de modo que la imagen del sistema operativo no utiliza ninguna partición del disco duro del ordenador sobre el que se carga. Todo el sistema de archivos se monta en memoria RAM de modo que, cuando se finaliza el examen y se apagan los ordenadores del aula, no queda ningún resultado de la realización del mismo.

Cada nodo del sistema se configura para obtener los parámetros de configuración de la red; empleando la extensión PXE (*Pre eXecution Environment*) de la BIOS. En nuestro caso estos datos se encuentran en la imagen del sistema operativo que será ejecutado. De esta forma, sin necesidad de instalar ningún tipo de software, se carga la imagen del sistema operativo (que está formada únicamente por los servicios básicos, las herramientas de compilación, edición y depuración, las páginas del manual y la plantilla de código fuente correspondiente al examen concreto que se va a realizar). Para garantizar que el sistema de archivos pueda montarse adecuadamente en los ordenadores de todas las aulas, la imagen del mismo una vez descomprimida debe ser de un tamaño inferior a 512MB.

El servidor tiene dos procesos claves para atender las peticiones PXE; el DHCPD y el TFTPd. El protocolo DHCPD (*Dynamic Host Configuration Pro-*

ocol Daemon) se utiliza para informar a cada cliente de la IP que le corresponde y la imagen del sistema operativo que debe cargar. Por su parte, el TFTPd (*Trivial Transfer Protocol Daemon*) se encarga de enviar el archivo al nodo consultando las tablas de configuración correspondientes.

Como se ha comentado anteriormente, el sistema de archivos completo que usará el sistema operativo se encontrará montado en memoria. De esta forma es posible utilizar el disco RAM inicial (*initrd*) para que contenga el sistema de archivos completo (sin necesidad de una posterior carga del sistema de archivos raíz).

En el sistema de arranque *initrd*, los archivos accesibles por el núcleo en la etapa de arranque se encuentran en un sistema de archivos que puede montarse como un dispositivo de bucle. En el caso de la imagen del sistema operativo utilizado para la evaluación de prácticas (*initrd.gz*) tiene un tamaño base comprimido de 88MB, al que hay que incorporar las plantillas de código y datos de prueba concretos para cada examen (que habitualmente no superan los 500KB).

Este proceso de actualización de cada examen puede resumirse en los siguientes pasos (que son automatizados empleando un script en *bash*):

1. **Descompresión** de *initrd.gz* sobre un sistema de archivos local (máquina del profesor).
2. **Montaje** de la imagen descomprimida (*initrd*) sobre un directorio local del sistema en modo bucle.
3. **Copia** de la plantilla del examen al directorio */home/alumno* del sistema montado.
4. **Desmontaje** del directorio.
5. **Compresión** de la imagen *initrd*.
6. **Subida** de *initrd.gz* al servidor de arranque PXE mediante sftp.
7. **Actualización** de enlaces simbólicos para el arranque de la misma imagen del sistema operativo en todas las aulas de laboratorio (automatiza empleando un script del lado del servidor).

En la versión actualmente implantada de este esquema, el arranque del núcleo del sistema operativo se realiza de forma manual antes de proceder a la realización del examen de prácticas. Los profesores de la asignatura deben arrancar cada máquina y seleccionar en el menú de arranque el nombre de la imagen. Este proceso conlleva aproximadamente 5 minutos por aula docente.

Actualmente se encuentra en pruebas un mecanismo automático para activar o desactivar los ordenadores del sistema haciendo uso de una extensión de las BIOS actuales llamada WOL (*Wake On Lan*), que ha sido exitosamente utilizada en otros proyectos de *Rendering* Distribuido realizados en la ESI [8]. Esta extensión se utiliza con la ayuda de la placa madre y el paquete software *Ether-Wake*. Incluso si un ordenador se encuentra apagado, la tarjeta de red continúa escuchando posibles peticiones. Cuando el paquete generado por *Ether-Wake* llega a la tarjeta, el ordenador se enciende y, por medio de una petición PXE, se carga la imagen del sistema operativo. Finalmente, para apagar por completo los ordenadores es necesario configurar correctamente el interfaz ACPI. Para poder hacer uso de esta funcionalidad se implementó un sencillo script en el lado del servidor el cual establece una conexión ssh con cada nodo y les envía el comando *shutdown* para apagarlos.

3. Primeros datos de su aplicación

El mecanismo de evaluación de laboratorio descrito en el presente trabajo se ha utilizado en la asignatura

de **Ampliación de Sistemas Operativos** desde el curso 2008/2009. La evaluación de las competencias relativas a las prácticas de laboratorio en cursos anteriores se realizaba mediante la resolución de un examen escrito en aula de teoría, y un test de conocimientos previos sobre ANSI C y System V.

La figura 2 muestra las calificaciones obtenidas por los estudiantes de la asignatura de ambas titulaciones (*Ingeniería en Informática* e *Ingeniería Técnica en Informática de Sistemas*) en los últimos 4 años.

Los datos correspondientes al curso 2007/2008 son similares a los obtenidos en convocatorias de los tres cursos anteriores. Antes de la incorporación de la nueva metodología de evaluación de las prácticas de laboratorio, la tasa de rendimiento era del 23 % de los estudiantes (con una tasa de abandono - no presentados - del 37 %).

El primer año de incorporación del mecanismo de evaluación de prácticas presentado en este trabajo mejoró los resultados de aprobados (con una tasa de rendimiento del 30 %), aunque empeoró la tasa de abandono (casi un 50 % sobre matriculados).

En los siguientes cursos (2009 y posteriores) continúa el crecimiento en el porcentaje de aprobados sobre matriculados, y se disminuye la tasa de abandono frente al curso 08/09.

Los exámenes completos compuestos por las plantillas suministradas a los estudiantes (enunciado y código fuente), junto con las soluciones de todas las pruebas realizadas en el curso académico 2010/2011 se encuentran disponibles³ para su estudio y evaluación.

4. Valoración de los resultados obtenidos

Una de las principales conclusiones obtenidas tras haber analizado los resultados detenidamente es que el número final de estudiantes que superan la asignatura es mayor con respecto al método más *clásico*, basado en pruebas escritas, utilizado durante los cursos 2007/2008 y anteriores. Además, la tendencia al alza respecto al número de aprobados se confirma con el paso del tiempo. En este contexto, la adaptación de los estudiantes al nuevo método de evaluación supone un factor relevante.

También se puede apreciar cómo en el cur-

³www.esi.uclm.es/www/cglez/downloads/docencia

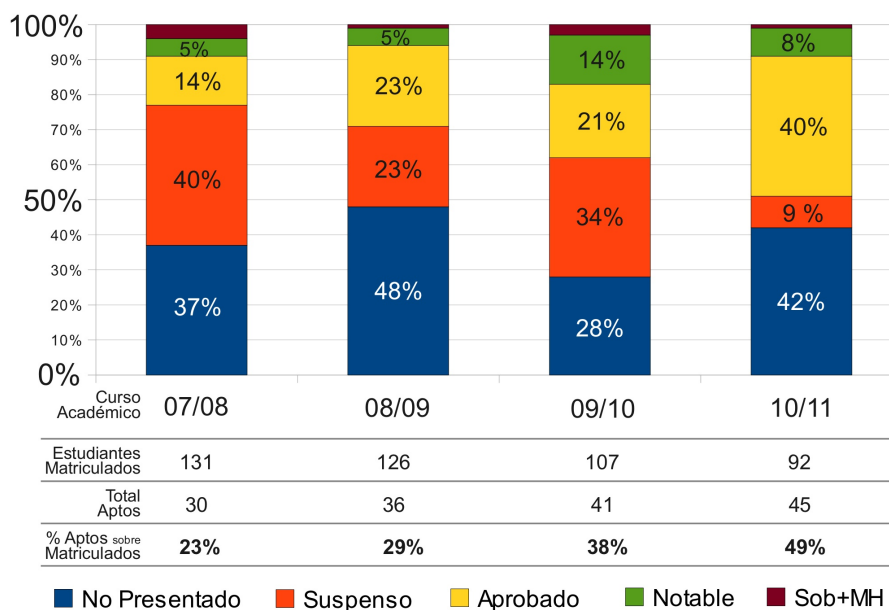


Figura 2: Tasa de Rendimiento en Ampliación de Sistemas Operativos (Cursos 2007-2011).

so 2008/2009 los resultados no son especialmente buenos según la tasa de rendimiento, probablemente debido a la propia adaptación a la nueva metodología de evaluación en el laboratorio. La realización de los exámenes delante de un ordenador supone un cambio importante con respecto a la realización de un examen escrito ya que, por ejemplo, el resultado correcto del proceso de compilación es un requisito necesario para la posterior ejecución del problema. Sin embargo, en un examen escrito suele existir más flexibilidad respecto a aspectos de este tipo.

Es importante destacar cómo la tasa de abandono del curso 2010/2011 creció. Este hecho se debe a que la asignatura de Sistemas Operativos, del curso 2º, se impartía de manera presencial por última vez debido al cambio de plan de estudios de la Escuela Superior de Informática de Ciudad Real. Así, hubo un número importante de estudiantes que focalizó sus esfuerzos en superar dicha asignatura, optando por no presentarse en la asignatura de Ampliación de Sistemas Operativos del curso 3º. Sin embargo, la tasa de estudiantes suspensos en esta última es considerablemente baja durante el curso 2010/2011.

Desde una perspectiva general, la realización

de pruebas presenciales en el laboratorio conlleva una mayor implicación del estudiante a la hora de afrontar problemas reales vinculados a la asignatura de Ampliación de Sistemas Operativos, típicamente relacionados con la sincronización de procesos mediante semáforos y paso de mensajes. El hecho de tener que programar una solución que compile y se ejecute correctamente, además de generar la solución correcta, hace que los estudiantes se impliquen directamente en el uso del computador como herramienta final, reforzando así la adquisición de las competencias básicas de la asignatura y planteando un esquema fiable para su evaluación.

Por otra parte, y debido a la propia naturaleza del método de evaluación, los estudiantes hacen especial hincapié en la implementación real de las soluciones planteadas en la parte más teórica de la asignatura, normalmente discutidas en pseudocódigo, con el objetivo de prepararse de una manera más adecuada para las pruebas de laboratorio.

Desde el punto de vista del profesor, el enfoque planteado facilita la evaluación práctica de los estudiantes, ya que el modelo de evaluación es totalmente determinista. En otras palabras, una solución

es correcta o no lo es. Si es correcta, entonces el profesor evaluará la calidad de la misma. Sin embargo, y en función del número de estudiantes involucrados en este tipo de pruebas, la coordinación de las mismas se puede convertir en una desventaja si dicho número es elevado. En tal caso, son necesarios más recursos humanos para controlar el correcto funcionamiento de las pruebas.

5. Conclusiones

En el presente trabajo se ha descrito una aproximación para la realización de pruebas prácticas evaluables en el laboratorio de la Ampliación de Sistemas Operativos, asignatura obligatoria para las titulaciones *Ingeniero en Informática* e *Ingeniero en Informática de Sistemas* en la Escuela Superior de Informática de la Universidad de Castilla-La Mancha. Así mismo, se ha discutido la metodología empleada para la construcción de dichas pruebas.

En el contexto de la Adaptación al Espacio Europeo de Educación Superior se promueve el uso de metodologías educativas y técnicas de evaluación activas, potenciando el empleo de mecanismos de calificación y certificación de habilidades y competencias que eliminen los exámenes finales como única fuente de evaluación del estudiante.

La metodología planteada en este trabajo, basada en la evaluación práctica del estudiante, permite evaluar y seguir de una manera más estrecha la evolución del mismo, ofreciendo mecanismos más fiables para llevar a cabo dicha evaluación y garantizar la adquisición de las competencias prácticas más importantes.

Los resultados obtenidos muestran cómo el número de estudiantes que superan la asignatura ha crecido en los tres cursos de aplicación de la metodología propuesta, demostrando así la viabilidad de este tipo de esquemas y fomentando el aprendizaje activo por parte del estudiante en los aspectos más prácticos de asignaturas vinculadas a la programación de sistemas.

Referencias

- [1] ACM/IEEE, Computer Science Curriculum 2008, *An Interim Revision of CS 2001*. Report from the Interim Review Task Force, Technical report, Association for Computing Machinery (ACM) IEEE Computer Society, 2008.
- [2] ACM/IEEE, *Curriculum Guidelines for Undergraduate Degree Programs in Information Systems*, Technical report, Association for Computing Machinery (ACM) Association for Information Systems (AIS), 2010.
- [3] Anderson, L., Krathwohl, D., et al, *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*, New York: Longman, 2001.
- [4] Bloom, B. S., *Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain*, New York: David McKay Co Inc., 1965.
- [5] Bonwell, C., Eison, J., *Active Learning: Creating Excitement in the Classroom*, ASHEERIC Higher Education Report No. 1, Technical report, The George Washington University, School of Education and Human Development, 1991.
- [6] Dewey, J., *How We Think, A Restatement of the Relation of Reflective Thinking to the Educative Process*, D.C. Heath and company, New York, 1922.
- [7] Eden, H., Einsenberg, M., Fischer, G. Repeing, A., *Making learning a part of life*, Communications of the ACM, 39(4), 40-42, 1996.
- [8] González, C., Weiss G., Vallejo, D., Jiménez L. Albusac J; *3D Distributed Rendering and Optimization using Free Software*, Upgrade (The European Journal for the Informatics Professional), Vol 9, 45-53, 2007.
- [9] McConnell, J.J., *Active learning and its use in computer science*, SIGCUE Outlook 24(1-3), 52-54, 1996.

