

PERCEPCIÓN DE LOS ESTUDIANTES DE INGENIERÍA DEL SOFTWARE FRENTE A LA INCLUSIÓN DE OBJETOS VIRTUALES DE APRENDIZAJE EN SU FORMACIÓN

Gustavo Martínez Villalobos¹
Jesús Herney Lozano Olivares²
Juan Manuel Rodríguez Guzmán³

Recibido: 5 de octubre de 2010 - Aceptado: 20 de noviembre de 2010

RESUMEN

Este artículo es un reporte de caso que plantea una nueva manera de mirar el aprendizaje de la Ingeniería de Software en universidades de modalidad presencial y a distancia, con el desarrollo e inclusión de objetos virtuales de aprendizaje que promuevan la interacción, la construcción de conocimiento y la generación de habilidades en los estudiantes que les permita una mayor autonomía y conectividad para el manejo del cambio en las empresas de hoy y del futuro. Para los autores, son relevantes las experiencias y los resultados obtenidos sobre percepción estudiantil ante el uso apropiado de la tecnología como herramienta para el aprendizaje.

Palabras claves: Conectivismo, Ingeniería de *software*, LMS, Objetos Virtuales de Aprendizaje, Software social.

PERCEPTION OF SOFTWARE ENGINEERING STUDENTS ON THE INCLUSION OF VIRTUAL LEARNING OBJECTS IN THEIR TRAINING

Abstract

This article is a case report which presents a new way to look at learning of Software Engineering in universities offering in-person and distance education modalities, with the development and inclusion of Virtual Objects of Learning that promote the interaction, the construction of knowledge and the generation of skills in students, that allows them a greater autonomy and connectivity for the management of change in present and future companies. For the authors, the experiences and results obtained from student perceptions regarding the appropriate use of technology as a learning tool are relevant.

Key words: Connectivism, Software Engineering, LMS, Virtual Learning-objects, Social software.

¹ Gustavo Martínez Villalobos, cc.14.243.915, Ingeniero de Sistemas, MSc. – vgus@bundenet.com, gustavo.martinez@unibague.edu.co, Profesor Asistente, integrante grupo de investigación GINNOVA, Universidad de Ibagué.

² Jesús Herney Lozano Olivares, cc.80.873.745, Estudiante programa Ingeniería de Sistemas – jeus2@live.com, asistente de investigación, Universidad de Ibagué.

³ Juan Manuel Rodríguez Guzmán, cc. 1.110.464.388, Estudiante programa Ingeniería de Sistemas – juroguz@hotmail.com, asistente de investigación, Universidad de Ibagué.

PERCEPÇÃO DOS ALUNOS DE ENGENHARIA DE SOFTWARE DIANTE DA INCLUSÃO DE OBJETOS VIRTUAIS DE APRENDIZAGEM EM SUA FORMAÇÃO

Resumo

Este artigo é um relato de caso que apresenta uma nova forma de olhar a aprendizagem da Engenharia de Software em universidades dos tipos presencial e à distância, com o desenvolvimento e inclusão de Objetos Virtuais de Aprendizagem que promovam a interação, construção de conhecimento e geração de habilidades nos alunos que lhes permitam uma maior autonomia e conectividade para o gerenciamento da mudança nas empresas de hoje e do futuro. Para os autores, são relevantes as experiências e os resultados obtidos sobre percepção estudantil diante do uso apropriado da tecnologia como ferramenta para a aprendizagem.

Palavras-Chave: Conectivismo, Engenharia de Software, LMS, Objetos Virtuais de Aprendizagem, Software social.

Introducción

El sistema educativo actual en Colombia se encuentra inmerso en un proceso de cambio. Las transformaciones sociales propiciadas por las Nuevas Tecnologías de Información y Comunicación (NTIC), traen consigo sus propias dinámicas educativas, transformando los libros en objetos de aprendizaje, los salones de clase en plataformas de gestión del aprendizaje –LMS– tipo Dokeos/Moodle, los “corrillos de pasillo” en foros y la conversación en *chat*. Estas transformaciones no son un simple cambio de forma, representan un cambio de los roles del profesor y del estudiante, en las interacciones comunicativas y en especial en los procesos de enseñanza-aprendizaje. Para comprender estas transformaciones y analizar el impacto que ellas producen, se deben realizar investigaciones y poner los hallazgos a la luz de los grupos académicos.

En los procesos de enseñanza y de aprendizaje los problemas educativos no son únicamente de naturaleza técnica, no se trata sólo de introducir innovaciones técnico-pedagógicas y organizacionales. La educación es de naturaleza social y cultural, y la simple introducción o aplicación de tecnología

educativa, por más eficiente que sea, no soluciona todos los problemas del sistema educativo. Un área prioritaria de la ciencia y la tecnología en nuestra región, tiene que ser imprescindiblemente la educación. Por eso, pretender hacer tecnología educativa colombiana con reflexión, inventiva y cooperación de todos los agentes educativos debe ser una gran meta.

Hay mucho por hacer en la línea de investigación en tecnología y calidad educativa; mucho por explorar, construir, medir, evaluar, etc. Y se necesita investigar no por el simple hecho de conocer nuestra realidad educativa, sino conocerla, pero para transformarla positivamente. Una tecnología educativa renovada admite y practica nuevos lineamientos: libertad, actividad, comunidad, valores, proyección y desarrollo social, trabajo colaborativo, auto aprendizaje, etc.

Este trabajo del grupo de investigación GINNOVA se enmarca en esa línea, y su ejecución busca dar respuesta fundamentalmente a: ¿qué relevancia tienen las nuevas tecnologías como soporte al proceso de generación de competencias en el área de ingeniería de *software*?, y ¿cómo lograr

que estas competencias sean las que demanda la sociedad actual?

El término Ingeniería del *Software* (IS) ha sido definido de varias maneras, por diferentes autores. El organismo IEEE (IEEE, 1990) expone oficialmente lo siguiente: “*Ingeniería de Software: (1) la aplicación de un enfoque sistemático, disciplinado y cuantificable hacia el desarrollo, operación y mantenimiento del software; es decir, la aplicación de la ingeniería al software. (2) El estudio de enfoques como en (1)*”. Según la ACM, Ingeniería del *Software* es la disciplina del desarrollo y mantenimiento de sistemas computacionales que se comportan de manera confiable y eficiente y que su costo de desarrollo y mantenimiento puede ser pagado (ACM, 1989).

La IS también aparece dentro de la Ciencia de la Computación, pero en los últimos años está siendo considerada como una entidad curricular independiente, aunque con profundas raíces en la Ciencia de la Computación y las Matemáticas (Peñalbo et al, 2004). Efectivamente, se ha definido un cuerpo de conocimiento para la Ingeniería del *Software* (*Software Engineering Body of Knowledge* – SWE-BOK), y existen borradores públicos del *Computing Curriculum – Software Engineering* por parte de la acción conjunta entre IEEE-CS y ACM.

Para lograr una buena formación en IS hay múltiples puertas de entrada y uno de los medios constituyen los Objetos de Aprendizaje (OA), ya que contienen diversos lenguajes (visual, sonoro, gráfico) que estimulan el sistema cognitivo para generar conocimientos, convirtiendo así los saberes teóricos en saberes prácticos, o en saber hacer. Este concepto del ámbito educativo es aplicado a materiales digitales creados como pequeñas piezas de información, material educativo digital, auto contenido y re-utilizable (Wiley, 2002), a fin de maximizar el número de situaciones educativas en las que el recurso pueda ser utilizado.

Hoy, en ausencia de contenidos estructurados por sus propios programas académicos, los estudiantes

hacen uso masivo de la Internet, donde no siempre encuentran los contenidos bajo la calidad, la forma o la orientación requerida para su formación, lo que se traduce en un ineficiente uso de este recurso.

La idea principal de los OVA radica en la posibilidad de que los estudiantes y profesores adapten los recursos didácticos de acuerdo con su disciplina, con sus propias necesidades, inquietudes y estilos de aprendizaje y enseñanza, generando así una educación flexible y personalizada. Al respecto, Sicilia (2005) expresa que esta concepción ha evolucionado en los últimos años como la noción subyacente (tácita o explícita) alrededor de la cual se estructura un conjunto de tecnologías y estándares que se presupone conducen a una “industria del aprendizaje” más eficiente y evolucionada, o si se prefiere, a un nuevo panorama en la educación basada o apoyada en la *Web*. Los OA permiten orientar el aprendizaje hacia el desarrollo de competencias, y cada objeto enfrentado por el estudiante, es muy explícito frente a los objetivos a cumplir por este una vez asimilado el objeto.

Un objetivo relevante del aprendizaje es que los alumnos deben aprender a aprender, y ser autónomos, capaces de tomar sus propias decisiones con respecto a lo que aprenden. Por ejemplo, aunque a menudo son vistas de forma negativa, las redes sociales y la “blogosfera” son, sin embargo, los espacios de interacción social donde los alumnos pueden optar por explorar facetas de su propia identidad, además de dedicarse al diálogo, la auto-expresión y el intercambio de conocimientos con los demás. Al participar en estas formas de conversación e interacción, los estudiantes exploran y desarrollan las facetas de su propia identidad y estilos personales de aprendizaje. De esta manera, ellos pueden dar forma a sus propias trayectorias de aprendizaje informal, así como participar en las de los demás.

Los escenarios de conversación e interacción mencionados anteriormente han sido generados mediante el *Software Social* (*social software*), definido como “*software* que soporta la interacción

grupal” (Martínez, 2010). Éste se refiere a un conjunto de herramientas de *software* o programas que permiten conectarse con otras personas a través de la Internet y socializar o trabajar de manera colaborativa en proyectos comunes. El resultado de esta interacción es la creación de espacios interactivos compartidos. El *Software Social* tiene ya mucho tiempo en la Red, con diferentes nombres y esquemas; esto existe desde antes de que apareciera la *www*, pero como una categoría en sí misma sólo recientemente se le reconoce.

Como herramientas de *software* social cada vez más comunes, son exploradas por los docentes e investigadores, que observan las tendencias cambiantes y el poder de la *Web 2.0* para la conexión de los profesores, los estudiantes y los recursos. Muchas de las aplicaciones actuales de *software* social se sitúan en el mundo social virtual y real, ya que implica la conectividad de interacciones tanto en línea y visual/verbal. Por ejemplo, *YouTube*, *Slideshare* y *Flickr*, que facilitan el intercambio de videos y fotos tanto con “amigos virtuales” como del “mundo real”; sitios de redes sociales como *Twitter* y *Facebook* permite a los usuarios crear una identidad en línea mediante la personalización de sus cuentas, perfiles con una gama de elementos multimedia, así como interactuar con los contactos existentes y crear relaciones nuevas (McLoughlin, 2007).

Es evidente que el *Software Social* rodea nuestras actividades diarias; en general, el *Software*, desde aviones, barcos y carros, hasta los bancos, supermercados, la salud, las comunicaciones, educación, entretenimiento, etc. Hay más y más demanda de *software* que controla elementos esenciales de nuestra vida, a la vez que cada vez es más complejo. Las nuevas tecnologías de desarrollo de *software* buscan conectar los elementos, más que crearlos. Es necesario analizar y validar estos sistemas. De acuerdo con Sommerville (2008), el reto más importante está en conseguir un desarrollo de *software* adecuado para los sistemas modernos que son cada vez más complejos y realizan tareas más importantes.

Lo anterior implica que el proceso de formación en la disciplina Ingeniería del *Software* debe garantizar un balance entre la teoría y la práctica. No es fácil mejorar la ausencia de conocimiento práctico en una disciplina donde la tecnología es tan cambiante y dinámica. Por eso se necesita un marco conceptual sólido más que la formación en manejo de herramientas. Según Villalobos (2007), un ingeniero, más que tener un conjunto de conocimientos (leyes, teorías, fórmulas), debe contar con las habilidades necesarias para usarlos en la solución de problemas. El reto para las universidades es ir más allá del simple enunciado de las definiciones, y generar en el estudiante las habilidades necesarias para poder utilizar de manera efectiva aquello que sabe.

También debe existir un balance entre las necesidades de hoy de las empresas y las necesidades del futuro ingeniero de *software* y, sobre todo, integrarse de manera adecuada a la empresa, entendiendo el negocio y la manera como la tecnología se incorpora a ella, como un medio y no como un fin en sí mismo. Esto último implica que el ingeniero debe tener bases más sólidas y profundas, y a la vez una visión más amplia de la disciplina, con la capacidad de ver el mundo con una aproximación multidimensional y sistémica, la que le será muy útil en su desempeño profesional.

Existen varios modelos de proceso de *software*, los cuales están documentados en un amplio catálogo de libros y documentos. Algunos autores, incluso, han dedicado gran parte de su trabajo a recopilar y actualizar toda la información existente al respecto y han publicado varias versiones de libros de IS considerados unos “clásicos” hoy en día; como los conocidos Roger S. Pressman (2006), Ian Sommerville (2005). Diferentes modelos de proceso han sido propuestos y puestos en práctica con mayor o menor éxito.

Sin embargo, es frecuente que en la práctica diaria profesional no se incluyan las recomendaciones más elementales de la ingeniería del *software*. La industria no suele utilizar estos modelos. Muchos

ejecutivos los ven como un aumento del proceso de desarrollo sin repercusión en las ventas. También, muchos gerentes de proyecto, diseñadores y desarrolladores de *software* ven la IS simplemente como una asignatura académica. Así, la ingeniería de *software* no es todavía una historia feliz de éxito.

En Colombia se han desarrollado diversos OVA, que son utilizados en las plataformas de educación virtual de varias universidades importantes, pero carecen de estrategias como lecciones aprendidas y estudio de casos, además de estrategias de estimación y valoración del esfuerzo de *software*.

Las plataformas educativas actuales cuentan con grandes servicios y diversas orientaciones, pero se requiere de miradas críticas y sobre todo creativas, para diseñar y apropiar plataformas que creen nuevos espacios virtuales acordes con modelos pedagógicos innovadores y de vanguardia.

A partir de la necesidad teórica de analizar con dimensión pedagógica la presencia y uso de las Nuevas Tecnologías de la Información y Comunicación (TIC) en los procesos educativos, esta iniciativa propone reconocer, desde una perspectiva sociocultural, un modelo de aprendizaje orientado a la era digital y aplicarlo en la creación de Objetos Virtuales de Aprendizaje compatibles con Sistemas de Gestión del aprendizaje (LMS) y con módulos virtuales que utilicen elementos y herramientas colaborativas y comunicativas de *software* libre para la Web 2.0 (*OpenAPIs*, *Wikis*, *Redes sociales*, *javaFx*, *Googledocs*...), como componentes que agregan a la actividad educativa nuevas oportunidades, motivaciones y estrategias como condición de aprendizaje de la disciplina Ingeniería de *Software*. El proyecto tiene las siguientes ideas-fuerza: a) la verdadera función de un docente no es dictar clases, sino lograr que los estudiantes aprendan, y b) el proceso educativo debe estar centrado en el estudiante y el aprendizaje.

Metodología

Este estudio de caso pertenece a un proyecto de investigación aplicada, explorando la posible aplicación de teorías de aprendizaje para la era digital y de tecnologías *Web* en la generación de OA, y buscando también la aplicación o utilización de los conocimientos adquiridos durante el proceso investigativo. Se utilizaron técnicas de observación y experimentación, enfoque sistémico-estructural, encuestas, análisis y síntesis (Cortés, 2005).

Al iniciar el estudio se realizó una adaptación del instrumento MSLQ (*Motivated Strategies for Learning Questionnaire*), un cuestionario con formato tipo Likert (Martín, 2009), con 81 ítems que atienden a quince variables agrupadas en tres escalas, pero se adaptó a 22 ítems distribuidos en tres categorías y tres subcategorías. MSLQ permite conocer y evaluar las orientaciones motivacionales y el uso de diferentes estrategias de aprendizaje por parte de los estudiantes; permite valorar de forma integrada aspectos cognitivos, metacognitivos y motivacionales.

Se trabajó con estudiantes pertenecientes al programa Ingeniería de Sistemas de universidades con modalidad presencial y a distancia en la ciudad de Ibagué, durante los períodos 2009/B y 2010/A. El cuestionario fue aplicado en dos fases, antes y después del desarrollo de los Objetos Virtuales de Aprendizaje para Ingeniería del *Software*.

Con base en las etapas del modelo de diseño instruccional ADDIE, análisis, diseño, desarrollo, implementación, catalogación, publicación y evaluación de OA, siguiendo también el estándar IEEE LOM CO (MEN, 2006), y con la metodología Proceso Unificado de desarrollo de *software*, se planteó la propuesta para la producción de Objetos de Aprendizaje con un enfoque centrado en el docente.

La producción de estos OA requiere también un modelo pedagógico específico para los ambientes Virtuales de Aprendizaje, que reconozca

las transformaciones culturales asociadas a las tecnologías de la información y la comunicación. En este modelo, el docente o tutor cambia su rol de proveer contenidos, a acompañar dirigiendo los procesos, orientando procesos investigativos y procurando desarrollar habilidades meta cognitivas; es decir, docente y estudiante asumen nuevos retos. La tecnología puede mejorar, expandir y enriquecer la experiencia del aprendizaje (Tobón, 2007). Ésta induce a los alumnos a tener un rol activo en vez de un rol pasivo en el aprendizaje. En este modelo se incluyeron un modelo didáctico, un modelo de instrucción, un modelo educativo; son algunos de sus elementos característicos el elemento teórico o marco teórico, el elemento metodológico que es la descripción del contenido del modelo, en respuesta a los aspectos teóricos en que se sustenta y el elemento práctico que es la puesta en práctica del modelo.

El desarrollo de OA se realizó después de tener claro todo el diseño, lo que permitió enfocarse en producir cada uno de los componentes por separado y ensamblarlos en un solo producto digital. Los Objetos Virtuales desarrollados son reutilizables y se pueden integrar a una plataforma LMS como *Dokeos* o como *Moodle*. Se desarrolló una interfaz para estos objetos aplicando diferentes técnicas y estándares actuales de diseño gráfico.

Como nueva teoría del aprendizaje, se exploró el conectivismo, una reciente teoría de aprendizaje para la era digital. Su autor, *George Siemens* (2009), intenta redefinir el concepto de aprendizaje refiriéndose a los cambios que han surgido en la última parte del siglo pasado con la irrupción de la tecnología como herramienta de aprendizaje.

Para redefinirlo, él define el concepto de conectivismo como una actualización del concepto de aprendizaje a la época actual en que vivimos superando los conceptos de conductismo, cognitivismo o constructivismo y definiendo el aprendizaje como un suceso netamente social. *Siemens* es un poco radical cuando intenta explicar el concepto de conectivismo y sus implicaciones, pero lo que

deja claro es que, para reforzar su capacidad de aprendizaje, una persona que intente aprender algo necesita la definición de unas buenas redes personales de búsquedas de contenidos, la realización de actividades de generación de contenidos y la participación en grupos de interés y el desarrollo de habilidades de análisis crítico de la información. *George Siemens* tiene recopiladas sus ideas sobre el conectivismo en el libro *Knowing Knowledge*.

• Módulos y objetos virtuales de aprendizaje

Una de las principales falencias que se tienen en el desarrollo de competencias en ingeniería de *software* es que a los estudiantes de Ingeniería de Sistemas les resulta difícil valorar la ingeniería del *software* como una disciplina que se refiere a la producción de sistemas de *software* desarrollados por grupos y no sólo por individuos, la cual exige, para cualquier proyecto de *software*, una metodología de desarrollo y la utilización de distintas técnicas y herramientas para la producción de *software* de calidad. Por eso, los objetos a diseñar deben proveer contenidos y atractivas actividades de aprendizaje que posibiliten la interacción con la Ingeniería del *Software* y promuevan procesos de análisis, reflexión y aplicación del conocimiento, bajo el enfoque de aprendizaje auto dirigido que requiere disciplina, conectividad y autonomía por parte del estudiante.

Para el estudio de esta temática, se diseñaron contenidos sobre cinco módulos, que a su vez contienen teóricamente lo más importante de cada uno de los temas; con esto se pretende que el estudiante tome la iniciativa, investigue y profundice más, ya sea por medio de la *Web*, libros o videos, para que después refuerce sus conocimientos con las actividades de aprendizaje y autoevaluaciones. Al finalizar, los estudiantes estarán en capacidad de identificar conceptos fundamentales de la Ingeniería del *Software*, relacionados con: Gestión, planificación y estimación de proyectos de *software*, la notación UML, calidad, gestión y configuración del *software* e Ingeniería *Web*. Estos módulos fueron desarrollados con herramientas de diseño para *Web 2.0*, en un ambiente gráfico agradable

a la vista del usuario, implementado en una página *Web* e integrado a un LMS.

- **Interfaz de OVAISw**

Como se muestra a continuación, esta página *Web* fue diseñada con el fin de dar a conocer cada uno de los módulos desarrollados y que se podrán encontrar en la plataforma.

También se puede tener acceso a cada uno de ellos por medio de un registro por parte del estudiante y una aceptación por parte del docente, describe la metodología, actividades y enfoque de los módulos, como también una breve introducción por medio de audio, video de bienvenida y relacionados con el mundo de la Ingeniería de *Software*, para que el estudiante tenga un acercamiento con todos estos temas.

- **Módulo UML de OVAISw**

Se planteó que el diseño de cada módulo fuera estándar, por lo que cada uno de ellos contará con su respectiva teoría, imágenes, animaciones, videos, audio, enlaces a documentos y autoevaluaciones. Estas autoevaluaciones se encontrarán al final de cada módulo en varias presentaciones: falso/verdadero y selección múltiple única respuesta, crucigrama, sopa de letras, rompecabezas, relación de conceptos y el juego del ahorcado.

- **Módulo de autoevaluación OVAISw**

Los ejercicios están orientados a la revisión por parte del estudiante de su grado de comprensión de los contenidos disponibles y la competencia adquirida para reconocer los conceptos de la Ingeniería del *Software*.

Resultados y análisis

El primer análisis de información acerca del OVA se tomó sobre una muestra de 80 estudiantes,

de dos universidades de modalidad presencial y dos universidades de modalidad a distancia, permitiendo el desarrollo de un método abierto, aplicando técnicas de encuesta, diálogo, preguntas de opinión y de actitud, grupales e individuales, y observaciones en el salón de clases, en el caso de Unibagué y la Universidad del Tolima. En el desarrollo de las pruebas con los cursos académicos se registró una experiencia positiva para los alumnos, aunque fueron detectados algunos problemas que ellos mismos valoraron, relacionados con la interactividad, animaciones sin sonido, revisión de algunos contenidos y de preguntas con nivel de dificultad alto.

En los formularios de las encuestas a estudiantes las preguntas cerradas se clasificaron en varios niveles: Contenido pedagógico y organizativo, Comunicación (herramientas y utilidad), Comunicación (contenido de las intervenciones), Comunicación (participación en las actividades y valoración), presentación, diseño y facilidad de uso, y sobre el ámbito tecnológico. Complementariamente, un espacio de pregunta abierta les permitió consignar sus observaciones y sugerencias.

El 57% de las preguntas realizadas en la encuesta es de contenido pedagógico y organizativo, seguido de un 22% que tiene como tema el ámbito tecnológico, un 14% en presentación, diseño y facilidad de uso y, finalmente, un 7% en comunicación: herramientas y utilidad.

Con todas las preguntas ya clasificadas (14), en sus diferentes categorías, se realizaron las encuestas a los estudiantes del curso de Ingeniería de *Software* en cuatro universidades de la ciudad, dos públicas y dos privadas, obteniendo los siguientes resultados:

1. El 92.9% de los estudiantes encuestados respondió que sí se especifican los objetivos del curso.
2. El 88% opinó que la redacción e identificación de conceptos es bastante clara y comprensible.
3. El 97% manifestó que las animaciones y gráficas son bastante coherentes con los contenidos.

PERCEPCIÓN DE LOS ESTUDIANTES DE INGENIERÍA DEL SOFTWARE



FIGURA 1. Módulos y Objetos Virtuales - OVAISw.

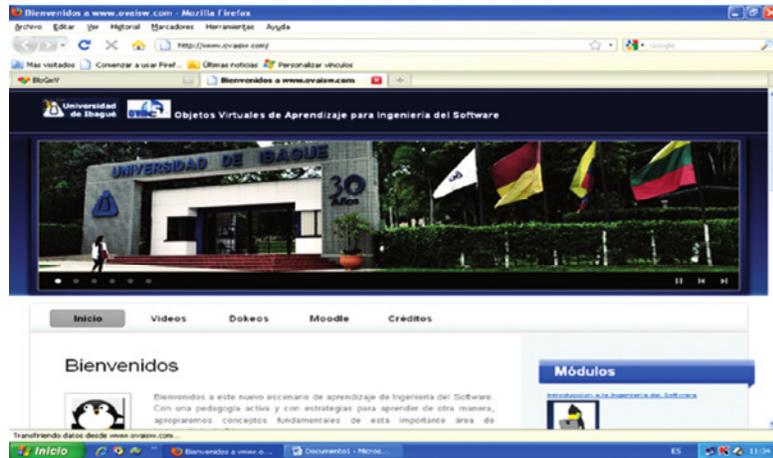


FIGURA 2. Interfaz principal - OVAISw.



FIGURA 3. Módulo UML - OVAISw

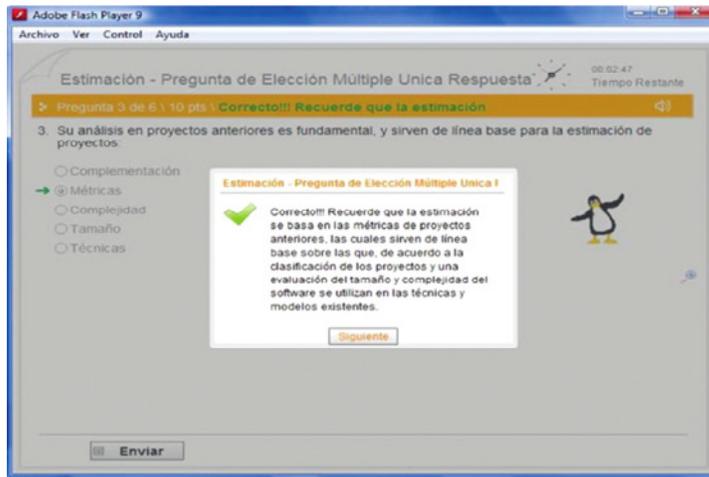


FIGURA 4. Interfaz autoevaluación módulo Estimación de Proyectos de software - OVAISw



FIGURA 5. Interfaz juego El Ahorcado - OVAISw

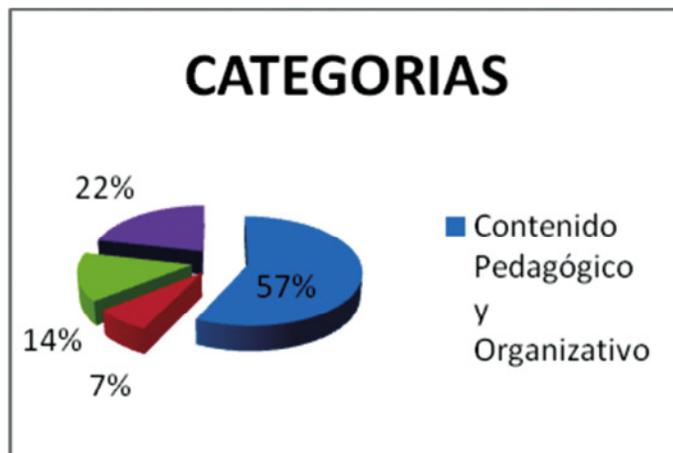


FIGURA 6. Categorías encuesta OVAISw

4. El 85.7% consideró que sí hay actividades o recursos que permiten diversas formas de acercamiento al conocimiento y su aplicación.
5. El 92.9% opinó que existen varias formas de autoevaluación.
6. El 100% manifestó que existe evaluación.
7. El 100% dijo que pueden ver los resultados de las autoevaluaciones.

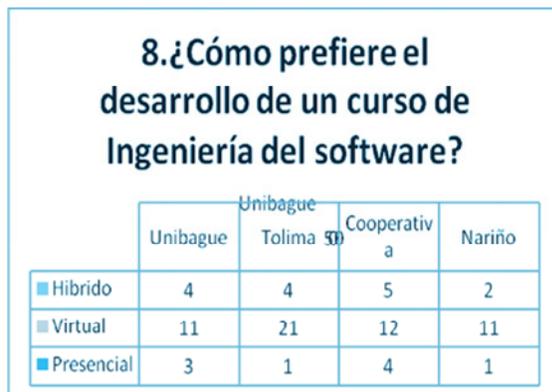


FIGURA 7. Encuesta pregunta-8

9. El 86% de los estudiantes encuestados respondió que el usuario tiene un papel bastante activo en el curso.
10. El 75% afirmó que el aspecto gráfico de OVAISw (colores, tipo de letra, identidad visual) es muy agradable.
11. El 81% respondió que la navegación dentro de OVAISw es buena.
12. Ante la pregunta de que si el sistema funciona correctamente, el 74% manifestó que bastante.
13. El 70% contestó que la visualización y la velocidad son bastante adecuadas.
14. El 72% manifestó que la actualización de cada uno de los contenidos en los módulos es continua.

Un resultado de gran impacto fue el correspondiente a la pregunta sobre preferencia de los estudiantes en el modo de enseñanza de un curso de Ingeniería de Software. Aunque los estudiantes consideran las TIC y los cursos virtuales como facilitadores,

dinámicos, favorables e importantes, la mayoría (70%) prefiere el desarrollo de un curso de Ingeniería de Software con modalidad virtual, 19% lo prefiere híbrida y 11% presencial.

Al culminar la tabulación y análisis de los resultados de la encuesta, se evidencia que el resultado ha sido positivo y satisfactorio, y que el producto es una alternativa para poder llegar a los estudiantes de una forma innovadora, visual e interactiva, utilizando herramientas de fácil manejo.

El impacto de la introducción de los OVA en términos académicos se evidenció con los resultados positivos del trabajo colaborativo, realizado a través del desarrollo de un proyecto de asignatura. Para el caso del grupo de estudiantes de Ingeniería del Software de la Universidad de Ibagué, se registró un 20% de mortalidad académica en el curso. No se consideró tan significativo confrontar este índice con el de las otras universidades, tomando en cuenta la diversidad en factores como los criterios de evaluación y porcentajes aplicados, manejo de la asignatura por créditos/intensidad horaria, infraestructura tecnológica, etc.

Conclusiones

La evaluación de la calidad de las actividades y contenidos publicados y la determinación del nivel de aceptación de los Objetos Virtuales de Aprendizaje se realizó mediante la aplicación de dos encuestas. Profesores y estudiantes de Ingeniería del Software emitieron juicios de valor sobre la calidad del OVA tanto en el contenido, la estructura, las actividades y el desempeño tecnológico, lo cual derivó en oportunidades de mejoramiento significativo. La mayoría de los estudiantes y docentes manifestó una positiva percepción ante la implementación y uso apropiado de Objetos Virtuales de Aprendizaje en su curso de Ingeniería del Software. En general, avalaron estos recursos tecnológicos como herramientas que los motivan y les permiten mejorar sus habilidades para aprender la asignatura.

Para el desarrollo de los Objetos Virtuales de Aprendizaje se exploraron las opciones tecnológicas y las recomendaciones, especificaciones y modelos que posibilitan la interoperabilidad de los OVA; estos modelos han sido definidos por varias organizaciones, como IEEE-LTSC, ADL, IMS *Global Consortium* y AICC. Entre ellas, se destaca ADL por la creación del modelo de referencia para objetos de contenido compartible SCORM. Estos requisitos definen las pautas para la creación de contenidos formativos de tal manera que sea posible su reusabilidad, accesibilidad, interoperabilidad y durabilidad. Las especificaciones de SCORM detallan cómo deben construirse y publicarse los contenidos y cómo se relacionan y comunican dichos contenidos con los LMS.

Se estudiaron también los sistemas de catalogación y clasificación existentes, en particular, la catalogación por el metadato colombiano LOM CO (bajo los estándares de metadatos IEEE LOM) y el Portal Colombia Aprende.

Mediante la utilización de diversas herramientas para *Web 2.0*, de diseño gráfico y de autor, se desarrollaron Objetos Virtuales de Aprendizaje e Informativos para cinco módulos del curso Ingeniería de *Software* y se probó su accesibilidad, adaptabilidad, durabilidad, reusabilidad e interoperabilidad integrándolos a los Sistemas de Gestión del Aprendizaje LMS *Moodle*, *Dokeos* y *efrontlearning*, con acceso desde la interfaz de *ovaisw.com*.

Definitivamente, es necesario realizar un trabajo enfocado al desarrollo y a la difusión del tema de Bancos de Objetos de Aprendizaje e Informativos en la Universidad de Ibagué. Fomentar la creación, difusión y catalogación de Objetos, apunta a que la comunidad educativa cuente con una amplia oferta de contenidos digitales elaborados y gestionados de manera participativa.

También se requiere guiar más a los estudiantes de Ingeniería de Sistemas hacia el trabajo independiente mediante el desarrollo de actividades

propias del aprendizaje autónomo. De este modo asumirán con mayor responsabilidad el aprendizaje de la Ingeniería de *Software* y de manera integral, su propia formación académica.

Todos estos proyectos de nuevas tecnologías para la educación son evolutivos y ofrecen un modo de facilitar el aprendizaje, pero es necesario aclarar que no existe una fórmula o una plantilla para diseñar el escenario perfecto. Cada institución es diferente, cada curso es único y los Ambientes Virtuales de Aprendizaje deben ser coherentes con las condiciones culturales, académicas, económicas y del contexto social, regional y nacional. De todos modos, es indiscutible que se debe seguir investigando en LMS, en competencias y en didáctica de la Ingeniería del *Software*.

Referencias

- ACM. (1989). Task Force on the Core of Computer Science, "Computing as a Discipline", *Communications of the ACM*. Consultado en: <http://www.acm.org/>
- Cortés, M. (2005). Generalidades sobre Metodología de la Investigación. Universidad Autónoma del Carmen. México: UAC.
- IEEE (1990). *IEEE Standard Glossary of Software Engineering Terminology*, IEEE Computer Society. Recuperado de: <http://ieeexplore.ieee.org/>
- McLoughlin, C. (2007). Social software and participatory learning: Pedagogical choices with technology affordances in the Web 2.0 era. Consultado en: <http://www.ascilite.org.au/conferences/singapore07/procs/mcloughlin.pdf>
- Martín, M. (2009). Cuestionario de estrategias y motivación para el aprendizaje. Consultado en: http://www.itesm.mx/va/dide2/enc_innov/3er08/memorias/pdfs/dra_marisa_martin_edgar_garcia_alicia_leal.pdf.
- Martínez, J. (2010). Red Escolar del Instituto Latinoamericano de la Comunicación Educativa (ILCE). Software social. Extraído de: <http://eae.ilce.edu.mx/software-social.htm>
- MEN. (2006). Ministerio de Educación Nacional Colombiano. *Objetos Virtuales de Aprendizaje e Informativos*. Recuperado de Portal Colombia Aprende: <http://>

- www.colombiaaprende.edu.co/html/directivos/1598/article-172369.html
- Peñalbo, F. (2004). Un enfoque de informática de gestión para los estudios de ingeniería informática en el marco de Bolonia. *Libro de actas JENUJ*. Ed. Thomson.
- Pressman, R. (2006) *Ingeniería del Software. Un enfoque práctico*. Sexta edición México: McGraw-Hill.
- Sicilia M.A. (2005) "Reusabilidad y reutilización de objetos didácticos: mitos, realidades y posibilidades". RED: Revista de Educación a Distancia, <http://www.um.es/ead/red/M2/>
- Siemens, G. (2008). Teoría del conectivismo. Consultado en: <http://www.elearnspace.org/>
- Sommerville, I. (2005) *Ingeniería del Software*. Séptima edición. USA: Pearson.
- Sommerville, I. (2008). Conferencia mundial sobre Componentes Software, Madrid, marzo 3 de 2008. <http://www.plataformasinc.es/index.php/esl/Entrevistas/El-reto-informatico-es-conseguir-un-software-adecuado-para-los-sistemas-modernos>
- Tobón, M. (2007). *Diseño Instruccional en un entorno de aprendizaje abierto*. Univirtual, Universidad Tecnológica de Pereira. Pereira: Postergraph.
- Villalobos, J. (2008). El actual ingeniero de software. Documento consultado en: http://eisc.univalle.edu.co/cursos/web/material/750092M/1/dllo2-Material_de_Trabajo.pdf
- Wiley, D. (2002). "Learning Objects". En Kovalchick & Dawson (Eds.). *Educational Technology. An Encyclopedia*. USA: Santa Bárbara.