

# Sistema de refuerzo del aprendizaje basado en cuestiones adaptadas al alumno

Enrique Lazcorreta<sup>1</sup>, Federico Botella<sup>1</sup>, Antonio Fernández-Caballero<sup>2</sup>,  
Pascual Gonzalez<sup>2</sup>

<sup>1</sup> Instituto Universitario “*Centro de Investigación Operativa*”,  
Universidad Miguel Hernández, Elche  
{[enrique](mailto:enrique@umh.es), [federico](mailto:federico@umh.es)}@umh.es

<sup>2</sup> Instituto de Investigación en Informática de Albacete,  
Universidad de Castilla-La Mancha, Albacete  
{[caballer](mailto:caballer@info-ab.uclm.es), [pgonzalez](mailto:pgonzalez@info-ab.uclm.es)}@info-ab.uclm.es

**Resumen.** Las actuales plataformas de e-learning y la tecnología actual permiten gestionar y procesar gran cantidad de datos en tiempo real. Esto da origen a la aparición de Sistemas Tutores Inteligentes que, modelando la figura del tutor humano, guían el proceso de aprendizaje de los alumnos de entornos virtuales.

Estos sistemas permiten adaptar los itinerarios formativos de la materia estudiada al ritmo de trabajo del alumno, partiendo de la experiencia del profesor y del propio sistema en su interacción con otros alumnos de la misma materia. Pero la adaptividad que presentan muchos de estos sistemas es una imposición al alumno, negándole en algunos casos estudiar usando su estilo personal, único e irrepetible.

El aprendizaje de lenguajes de programación informática requiere por parte del alumno de una mayor dedicación a la resolución de cuestiones prácticas, incluso muchos libros pretenden basar esos aprendizajes en la resolución de cuestiones. En un trabajo anterior [6] propusimos un sistema que permitía editar y actualizar on-line dichos libros en formato electrónico. En este artículo se expone la arquitectura del sistema y su interacción con el alumno.

**Palabras clave:** Adaptatividad, e-learning, sistemas tutores inteligentes.

## 1 Introducción

La tecnología actual permite que las plataformas e-learning gestionen eficientemente gran cantidad de material didáctico y de datos sobre los alumnos de un curso. Una buena distribución de dichos datos permite, en tiempo real, recuperar toda la trayectoria de cualquier alumno en el curso y aportarle el material más adecuado a su actual nivel de conocimiento y sus necesidades específicas.

Gran parte de los trabajos actuales de investigación se orientan hacia el desarrollo de los llamados Sistemas Tutores Inteligentes (STI), que monitorizan la interacción del alumno con el material y recursos del curso para descubrir su estilo de aprendizaje

y poder guiarle en sus estudios [1] o para obligarle a reforzar los conceptos mal adquiridos [2]. Estos sistemas evalúan al alumno para poder determinar su nivel de conocimiento sobre la materia y basan su eficiencia en la adaptación del sistema al ritmo de trabajo del alumno y a los itinerarios formativos diseñados por el profesor o descubiertos por el sistema.

Butz et al [3] flexibilizan el currículo del curso al nivel de que si un alumno necesita repasar un único tema (para un inminente examen, por ejemplo) el sistema le indique los mínimos conceptos previos que debería estudiar. Brusilovsky et al. [4] estructuran un curso básico de programación en 44 tópicos y preparan 2 ó 3 cuestionarios por tópico, consistentes en 5 o 6 cuestiones del tipo “¿Cuál es el valor final de una expresión?” o “¿Qué se imprimirá?”. Su propuesta consiste en el uso de anotaciones adaptativas para indicar al alumno la conveniencia de resolver cada cuestionario en función de sus conocimientos sobre el tópico tratado, sus conocimientos sobre los prerrequisitos del tópico y sus necesidades. Su trabajo es un complemento de una propuesta anterior [5] en que exponían la parametrización de las cuestiones de modo que el alumno pudiese reintentar resolverlas sin repetir exactamente el mismo código en la cuestión.

Una característica común en estos trabajos es que desarrollan su investigación en cursos de e-learning sobre lenguajes de programación. El motivo suele ser la proximidad de la materia a la informática. En esta misma línea presentamos en [6] un sistema que facilita al profesor la labor de crear y gestionar gran cantidad de cuestiones sobre programación informática. En este artículo exponemos cómo el sistema refuerza el proceso de aprendizaje del alumno sugiriéndole las cuestiones que mejor se adaptan a sus necesidades particulares. Este enfoque no sustituye al refuerzo expuesto en [2], que se basa en una estrategia diseñada por el profesor y gestionada por los conocimientos del sistema sobre el curso y el alumno. Este enfoque complementa cualquier proceso de aprendizaje permitiendo que sea el alumno quien dirija su propio itinerario formativo en la resolución de cuestiones prácticas, como se propone en [3], y reduciendo la intervención del sistema en cuanto al uso de prerrequisitos o estrategias de aprendizaje diseñadas por el profesor o el propio sistema, como se propone en [4] pero liberando al profesor de la gestión y clasificación de las cuestiones, como se propone en [6].

La adaptación del refuerzo se logra mediante la consulta sobre un gran número de cuestiones basándose en tan solo tres parámetros indicados por el alumno: qué quiere repasar, con qué profundidad y cuánto está dispuesto a esforzarse en ese momento. Si el sistema logra adaptarse al alumno, éste no dudará en utilizarlo siempre que quiera practicar con la materia en estudio pues esta metodología puede mejorar su curva de aprendizaje, lo que constituye una de las bases de la adquisición del conocimiento.

El resto del trabajo se organiza como sigue. En la sección 2 se describen las cuestiones autoadaptativas. En la sección 3 se presenta la arquitectura del sistema, su interfaz con el alumno y el proceso que realiza para encontrar las cuestiones que más se adaptan a las necesidades de aprendizaje del alumno. En la sección 4 se presentan las conclusiones de este trabajo de investigación y las posibilidades que presenta como trabajo futuro.

## 2 Cuestiones Autoadaptativas

Las cuestiones autoadaptativas surgen de la idea de crear un libro electrónico del tipo “*Aprende a programar en lenguaje... con 1000 cuestiones*”. La adaptatividad de las cuestiones se debe a que ni el profesor se ha de preocupar de clasificarlas ni el alumno de decidir cuál es la que más le interesa en función de sus necesidades. El sistema se encargará de estas tareas:

- si el profesor quiere añadir una cuestión sólo ha de preocuparse de escribir un enunciado, el código fuente asociado y una solución que pueda ser comparada por el sistema con la respuesta dada por el alumno,
- si el alumno quiere practicar resolviendo cuestiones, el sistema se encargará de ofrecerle las que mejor se adapten a sus necesidades y conocimientos en el instante de la solicitud.

Los actores de este sistema serán los profesores y los alumnos. Inicialmente nos preocupa la figura del profesor, pues es quien tiene que crear la base de conocimiento del sistema: un diccionario de conceptos y una colección de cuestiones que se utilizarán cuando un alumno solicite su servicio.

Incluso respetando los estándares, la creación de grandes volúmenes de material didáctico es una tarea rechazada por muchos profesores debido a su experiencia en el campo del e-learning: la reutilización del material no está garantizada; su gestión y clasificación puede convertirse fácilmente en una tarea consumidora de tiempo. Si a esto añadimos las posibilidades de personalización que ofrecen los STI a los alumnos y la tarea de ajustar parámetros del sistema o resolver las incidencias que haya encontrado, la gestión de la información proporcionada por el STI al profesor duplica su trabajo y produce el incremento de abandonos (de alumnos y también de profesores) en el mundo del e-learning. Para facilitar la labor del profesor se hace necesario incorporar al STI la posibilidad de crear y clasificar cientos (miles) de cuestiones que sean gestionadas directamente por el sistema:

- Para crear el diccionario de conceptos se podría usar la información ya disponible sobre un curso si está estructurado mediante conceptos.
- La elaboración del diccionario de conceptos consistirá únicamente en la introducción de una palabra que identifique unívocamente al concepto (en lenguajes de programación podrían ser sus palabras reservadas, operadores o funciones de librerías estándar) y de una descripción del mismo. No será necesario definir prerequisites y se guardará en una base de datos o en un documento XML que podría ser reutilizado, por su sencillez, en cualquier otro sistema. No será necesario haber introducido todos los conceptos de un tema o unidad temática para poder comenzar el uso del sistema por los alumnos.
- El profesor no estará obligado a escribir cientos de cuestiones para empezar a usar el sistema. Cada vez que inserta una cuestión, el sistema la incorporará en su base de datos y, mediante un análisis automático de su contenido, la clasificará adecuadamente según la información que disponga del curso y de sus unidades/temas/conceptos y estará disponible para la siguiente solicitud de un alumno.

- Si el profesor detectara un concepto no clasificado al introducir una cuestión podrá incorporar el concepto y sus características a la base de conocimiento correspondiente.
- El profesor recibirá información del sistema pero sin distraerle de su tarea principal, de modo que las aportaciones introducidas por el profesor sean permanentes y no colisionen con anteriores aportaciones obligándole a rehacer su trabajo previo (o actual).
- El sistema podrá ser utilizado por distintos profesores con distintos criterios pedagógicos sin perder la información incorporada por cada uno de ellos.

Estas cuestiones pueden estudiarse con más detalle en [6], mientras que ahora expondremos por qué este sistema puede adaptarse también al alumno, y cómo lo hace. Cuando el alumno acceda al sistema, éste puede tener ya información sobre el aprendizaje del alumno, información obtenida del STI en que está alojado el sistema. Pero supongamos que esto no fuera así para explicar el comportamiento del sistema cuando atiende a cualquier alumno y en cualquier instante del curso, empezando por el instante en que el alumno ingresa por primera vez en el sistema.

Debemos hacer notar que las cuestiones utilizadas por este sistema son autocorrectivas, es decir, conocen su solución y por tanto el sistema puede saber si un alumno acertó o falló en la resolución de cualquiera de sus cuestiones. Preguntas como “¿Qué valor devuelve la siguiente función?” son válidas para un curso de programación, pero también lo son las cuestiones de alternativas múltiples, entre otras. No se contemplarán cuestiones que puedan ser evaluadas parcialmente. Aunque esto podría dotar al sistema de más versatilidad haría muy complejo su mantenimiento y reutilización, lo cual va en contra de los principios de diseño en que fundamentamos este sistema, basado en la sencillez del diseño y la facilidad de uso por los alumnos y, como no, por los profesores.

### 3 Arquitectura del sistema

El sistema de refuerzo del aprendizaje ha sido diseñado para el trabajo con grandes bases de datos que contengan miles de cuestiones, temas y unidades, por lo que desde un principio se buscará un equilibrio entre el volumen de datos a guardar en cada tabla y la velocidad de las consultas: si tenemos datos pre-calculados guardados permanentemente ganaremos en velocidad a la hora de buscarlos.

Los actores principales de este sistema serán los profesores, que introducirán el material docente de forma que les sea sumamente fácil su interacción con el sistema, y los alumnos, que recibirán un aprendizaje adaptado a sus intereses en cada sesión (avanzar, repasar) o nivel de conocimientos adquiridos. Los alumnos serán inscritos en los cursos que se hayan matriculado, y podrán acceder así a todas las asignaturas del curso en que también estén matriculados. Dichas asignaturas estarán formadas por conceptos y cada concepto podrá tener asociadas múltiples cuestiones que permitan el aprendizaje por parte de los alumnos. Este sistema pretende que el profesor y el alumno centren su trabajo en torno a las cuestiones, siendo éstas las que permitan al profesor la definición de itinerarios formativos, gracias a la clasificación automática

de las cuestiones por parte del sistema; y al alumno el refuerzo de su aprendizaje, gracias a la resolución de cuestiones adaptadas a su nivel de conocimiento adquirido en un tema y que el sistema le permitirá avanzar “todo lo rápido” que quiera el alumno para conseguir el grado de conocimiento deseado o grado de conocimiento mínimo definido por el profesor para cada asignatura.

En la figura 1 podemos ver un modelo lógico de la estructura de tablas definidas para la creación de la base de conocimiento del sistema de aprendizaje.

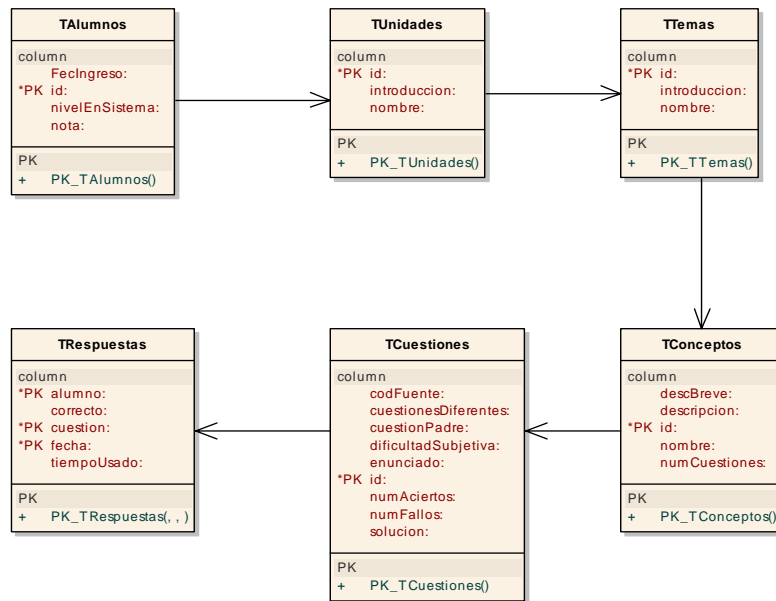


Fig. 1. Modelo lógico del sistema de refuerzo

### 3.1 Adaptación de cuestiones sugeridas

Cuando el alumno accede al sistema lo hace por iniciativa propia y sabiendo que será él quien decida qué es lo que quiere hacer en ese momento: normalmente, practicar en una materia determinada aprovechando el tiempo de que dispone. Su estado de ánimo le puede sugerir que está en un buen momento y puede resolver problemas difíciles, o que no está en su momento álgido y que sería mejor practicar con preguntas fáciles; lo importante es que es el alumno quien decide estos aspectos, y no el propio sistema sabiendo que se trata de un alumno de alto o bajo nivel. Un tutor humano es capaz de detectar el estado de ánimo de un alumno y aprovecharlo para obtener el máximo rendimiento del tiempo dedicado al estudio. Un STI no puede detectar ese parámetro

y suelen estar diseñados para que los buenos estudiantes trabajen al 100% aunque no estén animados, o que los malos estudiantes trabajen a un ritmo lento hasta que logren demostrar que son buenos estudiantes (si no se han decepcionado antes y han abandonado los estudios). Aunque el sistema conociera el verdadero nivel del alumno, sólo utilizaría esa información para indicar al alumno si es capaz de resolver unas cuestiones o no, o cuáles serían las “teóricamente” más adaptadas a él dados sus conocimientos sobre la materia (anotaciones adaptativas). En la figura 2 se muestra la interfaz del sistema diseñada en base al principio de facilidad de uso para el alumno.

Desarrollo de e-learning - DEMI - UMH : Inicio - Mozilla Firefox

Archivo Editar Ver Ir Marcadores Herramientas Ayuda

Fundamentos de informática

Inicio Foros Glosario

Jueves Mayo 11th, 2006

Quiero resolver cuestiones [Mostrar](#)

Quiero revisar el concepto

Como único objetivo

Practicando conceptos con baja puntuación

Quiero una revisión más general

Del tema

Hasta el tema

De los conceptos con peor puntuación

De los conceptos que más domino (quiero divertirme)

Y quiero que las cuestiones sean

Fáciles  No tan fáciles  Algo difíciles  Realmente difíciles  Decide tú

Quiero repetir cuestiones que ya haya intentado:

Sólo si no supe resolverlas  Incluso las que resolví bien  Sólo si las resolví bien

Fig. 2. Interface del sistema de refuerzo del aprendizaje

El alumno puede decidir qué tipo de cuestiones desea realizar, o bien consultar la información que dispone el sistema sobre su perfil para ayudarle en la decisión; pero será siempre el alumno quien determine el guión de su próxima sesión de prácticas.

Cuando el alumno pulse sobre “*Sugiere me cuestiones*” (ver figura 2), el sistema revisará los parámetros seleccionados por el alumno, consultará la información disponible sobre dicho alumno y le mostrará una colección de cuestiones ordenada según las preferencias del alumno. El orden de las cuestiones sugeridas se regirá por lo interesante que sea cada cuestión en el instante en que el alumno haga su solicitud.

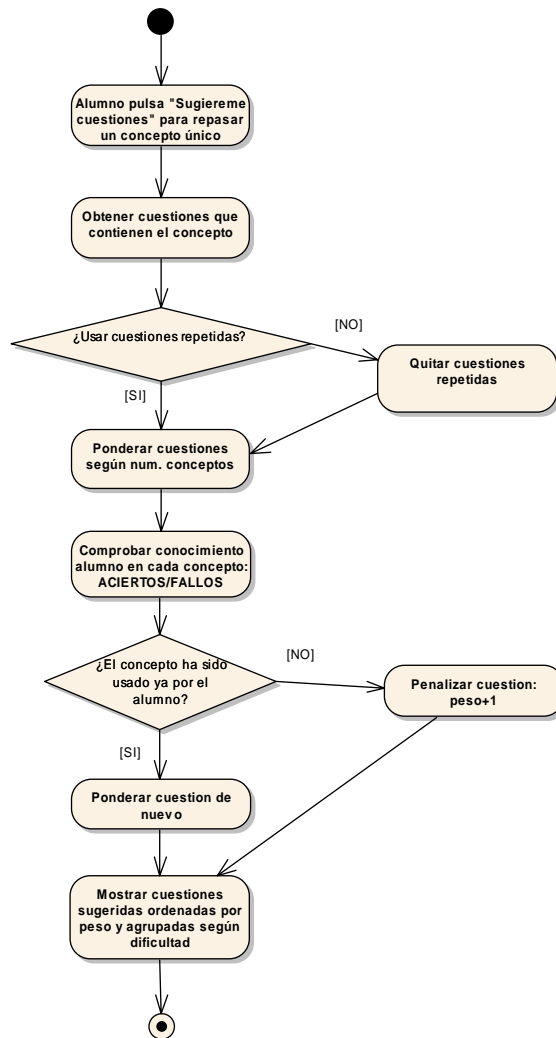


Fig. 3. Diagrama de actividad: caso estudio de un concepto como objetivo único.

Una cuestión será muy interesante si se adapta bien a lo que quiere hacer el alumno y además presenta la dificultad esperada por éste. Aunque el sistema localice cientos de cuestiones que podrían ser resueltas por el alumno, sólo le mostrará las diez cuestiones que presenten mayor interés en ese instante, permitiéndole navegar a través del resto de cuestiones en grupos de diez de interés decreciente.

Si el alumno quiere repasar un concepto “Como único objetivo” se consultará en la tabla TCuestiones para obtener todas las cuestiones que contienen el concepto objetivo; en función de las preferencias del alumno se eliminarán o no las cuestiones ya resueltas por el alumno, a continuación se ponderarán por el número de conceptos

que tiene cada cuestión (las cuestiones que sólo tengan un concepto serán las más interesantes pues se trata del concepto-objetivo; luego irá disminuyendo su interés según el número de conceptos) y, por último, se comprobará el número de aciertos/fallos en conceptos realizado por dicho alumno, de modo que las cuestiones quedarán ponderadas en base a esta información. Debido a que el alumno quiere repasar el concepto como único objetivo, esta última ponderación penalizará las cuestiones con conceptos desconocidos o mal comprendidos por el alumno, y premiará aquellas con conceptos comprendidos. Para lograrlo se restará al peso de la cuestión el resultado de la fórmula

$$\frac{ACIERTOS}{INTENTOS} * \text{signo}(2 * ACIERTOS - FALLOS)$$

Una vez ponderadas las cuestiones se agruparán en función de su dificultad y se presentarán al alumno las cuestiones ordenadas en función de su peso para que pueda elegir la cuestión que quiere resolver. El comportamiento del sistema ante esta situación se muestra en la figura 3.

Si el alumno quiere repasar un concepto “Practicando conceptos con baja puntuación” estaremos ante la misma consulta anterior, pero ahora para calcular el interés de las cuestiones se utiliza la fórmula

$$\frac{FALLOS}{INTENTOS} * \text{signo}(2 * FALLOS - ACIERTOS)$$

pues se trata de premiar las cuestiones que contengan conceptos que el alumno no ha demostrado comprender, y penalizar el resto de cuestiones.

Si el alumno quiere hacer un repaso general “del tema...” estaremos en un caso similar al de repaso de un concepto, pero considerando el conjunto de conceptos que forman el tema. En primer lugar se obtendrá de la base de conocimientos el conjunto de conceptos que desea reforzar el alumno con la idea de que sean reforzados aquellos que más necesite. En el interés se premiarán las cuestiones que sólo contienen conceptos del tema, lo que se logra contando el número de conceptos contenidos en cada cuestión ajenos al conjunto de conceptos del tema. Estas cuestiones, a continuación, modificarán su interés en función del nivel que tiene el alumno sobre los conceptos que contienen y de si éstos son objetivo del estudio. Los conceptos del tema que aún no haya estudiado el alumno aumentarán drásticamente el interés de la cuestión para lograr que el alumno complete el tema en estudio; en cambio, los conceptos ya estudiados aportarán interés a la cuestión sólo si el alumno no los comprende bien, buscando el mismo objetivo. Por último, las cuestiones perderán interés en función del número de cuestiones obtenidas que se refieran al mismo conjunto de conceptos del tema (la primera no pierde interés y el resto van perdiéndolo conforme aumente su ordinal respecto a este subconjunto), así lograremos que se entremezclen los conceptos utilizados en las cuestiones y que el alumno no practique varias veces con el mismo concepto (para eso está la primera opción). La figura 4 muestra el diagrama de actividad del sistema para el refuerzo de un tema.

Las tres restantes opciones son meras modificaciones de las anteriores, que ayudan al alumno a prepararse para la realización de un examen parcial o final del curso, a reforzar conceptos específicos sin decidir él cuáles serán o a resolver cuestiones que,



a priori, sabrá contestar sin esfuerzo, con lo bueno que eso resulta para la motivación de cualquier aprendiz.

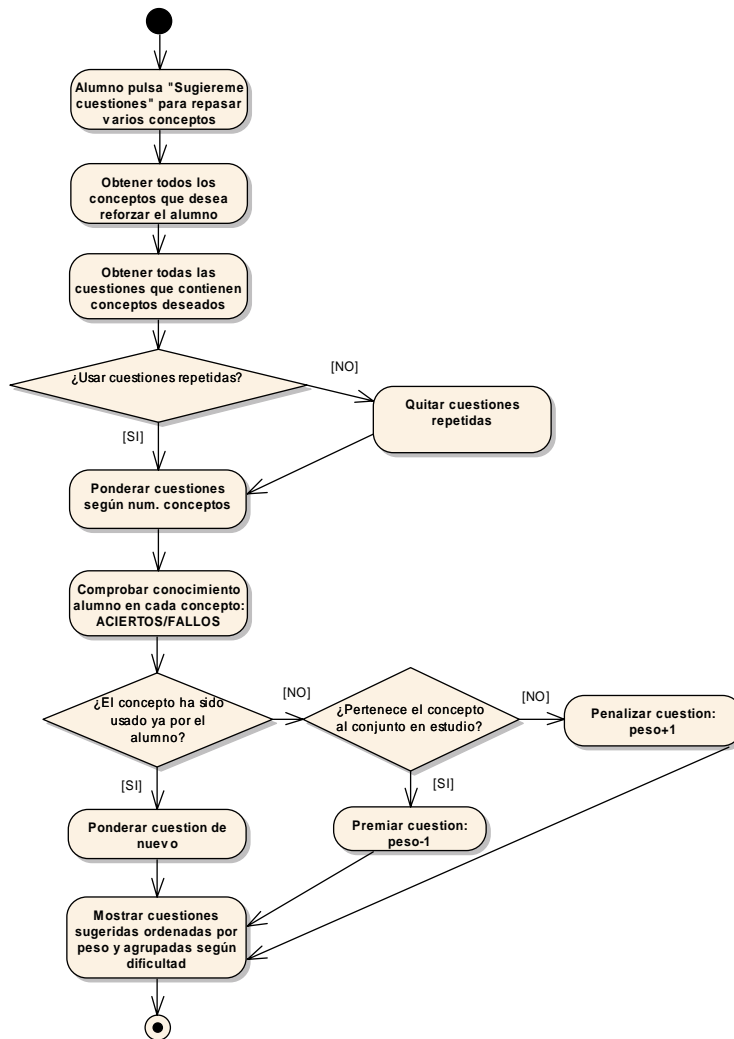


Fig. 4. Diagrama de actividad: caso estudio de un conjunto de conceptos.

Si se obtienen menos de diez cuestiones en cualquier consulta planteada por un alumno, se enviará automáticamente un mensaje al profesor con las características del alumno, la solicitud realizada y la consulta obtenida, de modo que el profesor conozca si el sistema de refuerzo presenta deficiencias en su contenido.

## 4. Conclusiones y trabajo futuro

En este trabajo se ha presentado un sistema de refuerzo del aprendizaje basado en cuestiones adaptadas al alumno y se ha descrito la arquitectura del sistema teniendo en cuenta los principios de facilidad de uso e implementación. El alumno que quiere reforzar sus conocimientos sobre una materia sólo ha de indicarle al sistema qué quiere estudiar (un concepto determinado, un tema, todo el curso, ...), qué nivel de profundidad desea (en cuanto a la incorporación en las cuestiones de otros conceptos distintos a los solicitados) y qué esfuerzo está dispuesto a realizar (dificultad de las cuestiones y posibilidad de responder sólo a cuestiones que a priori serán fáciles para el alumno). La sencillez del sistema garantiza su uso por parte de los alumnos interesados en practicar la materia; su potencia para trabajar con un gran número de cuestiones posibilita una adaptación a las necesidades del alumno en cualquier momento de sus estudios.

Una vez comprobada la eficiencia del sistema y ajustados sus parámetros, se tratará de extenderlo a materias de estudio que no sean tan estructuradas como el aprendizaje de un lenguaje de programación informática y trataremos de incorporar el concepto de anotaciones adaptativas utilizado en [4].

## Agradecimientos

Este trabajo está financiado, en parte, por la ayuda de la Universidad Miguel Hernández de Elche – Bancaja R.R.1256/04 y por el proyecto de la Junta de Comunidades de Castilla-La Mancha PAI06-0093.

## Referencias

- [1] C.I. Peña, J.L. Marzo, J.L. de la Rosa, "Intelligent agents in a teaching and learning environment on the Web", Proc. of the 2nd IEEE International Conference on Advanced Learning Technologies (ICALT), 2002, pp. 21-27.
- [2] A. Fernández-Caballero, J.M. Gascuña, F. Botella, and E. Lazcorreta, "Distance Learning by Intelligent Tutoring System", Procs. of the 7th International Conference on Enterprise Information Systems (ICEIS), 2005, vol. 5, pp. 75-82.
- [3] C.J. Butz, S. Hua, R.B. Maguire, "Bits: a Bayesian Intelligent Tutoring System for Computer Programming", Procs. of the Western Canadian Conference on Computing Education (WCCCE), 2004.
- [4] P. Brusilovsky, S. Sosnovsky, and O. Shcherbinina, "QuizGuide: Increasing the Educational Value of Individualized Self-Assessment Quizzes with Adaptive Navigation Support", Procs. of World Conference on E-Learning (E-Learn), 2004, pp 1806-1813.
- [5] S. Sosnovsky, O. Shcherbinina, and P. Brusilovsky, "Web-based parameterized questions as a tool for learning". Procs. of World Conference on E-Learning (E-Learn), 2003.
- [6] E. Lazcorreta, F. Botella, A. Fernández-Caballero, J. M. Gascuña. "Auto-Adaptive Questions in E-learning System". 6th IEEE International Conference on Advanced Learning Technologies (ICALT), 2006.