

LEARN-SQL: Una herramienta para el soporte al aprendizaje semipresencial en el ámbito de las bases de datos

LEARN-SQL: A blended learning tool for the database area

M. José Casany¹, Carme Martín¹, Alberto Abelló¹, Xavier Burgués¹,
Carme Quer¹, M. Elena Rodríguez², Toni Urpí¹

Universitat Politècnica de Catalunya¹, Universitat Oberta de Catalunya²
{*mjasany,martin,aabello,diafebus,cquer,urpi*}¹[@lsi.upc.edu](mailto:ls@lsi.upc.edu), *mrodriguezgo@uoc.edu*²

PALABRAS CLAVE

Aprendizaje semipresencial, Bases de datos, Corrección automática.

RESUMEN

Los cambios recientes en los planes de estudio de la UPC y la UOC tienen en cuenta el nuevo espacio europeo de educación superior (EEES). Una de las consecuencias directas de estos cambios es la necesidad de acotar y optimizar el tiempo a dedicar a las actividades de aprendizaje que requieren la participación activa del estudiante y que se realizan de forma continuada durante el semestre. En el ámbito de la docencia informática de asignaturas de bases de datos el problema es especialmente complejo debido a que los enunciados de las pruebas no acostumbran a tener una solución única. Nosotros hemos desarrollado una herramienta, llamada LEARN-SQL, cuyo objetivo es corregir automáticamente cualquier tipo de sentencia SQL (*consultas, actualizaciones, procedimientos almacenados, disparadores etc.*) y discernir si la respuesta aportada por el estudiante es o no correcta con independencia de la solución concreta que este proponga. Además esta herramienta ayuda a los profesores a diseñar las pruebas de evaluación, permitiendo también la opción de revisar cualitativamente las soluciones aportadas por los estudiantes. Por último, el sistema proporciona ayuda a los estudiantes para que aprendan de sus propios errores, proporcionando *feedback* de calidad.

KEYWORDS

Blended learning, Databases, Automatic correction.

ABSTRACT

The academic programs of the UPC and UOC are adapting to the European Credit Transfer System (ECTS). One of the changes introduced in the academic programs of the previous universities tries to optimize the time of the activities that require the active participation of the students. The definition of these activities is a very complex task specially when dealing with database teaching in ICT engineering degrees, because usually the questions do not have a unique solution. LEARN-SQL is the tool developed by our group that automatically evaluates the correctness of any SQL statement (queries, updates, stored procedures, triggers etc.) with independence of the student solution. Furthermore, LEARN-SQL helps teachers design their tests as well as allow

them review the solutions provided by the students. Finally, the system provides students with valuable feedback, so that they can learn from their mistakes.

1. INTRODUCCIÓN

En el aprendizaje semipresencial de las asignaturas de bases de datos existen varios sistemas (por ejemplo en varias universidades de Australia, Nueva Zelanda [2] [6] [8] y en la de Girona [9]) que permiten la corrección automática de ejercicios. No obstante, estos sistemas se limitan a corregir el tipo más básico de ejercicios posibles: las consultas en el lenguaje SQL. Así pues, nosotros hemos desarrollado una herramienta, llamada LEARN-SQL (*Learning Environment for Automatic Rating Notions of SQL*), cuyo objetivo es corregir automáticamente cualquier tipo de sentencia SQL (*consultas, actualizaciones, procedimientos almacenados, disparadores etc.*) y discernir si la respuesta aportada por el estudiante es o no correcta con independencia de la solución concreta que este proponga. La corrección automática basada en la comparación de caracteres entre la solución del estudiante y la del profesor no es eficiente. Además, la respuesta a los ejercicios de la asignatura acostumbra a no ser única. Para una misma pregunta, normalmente existen diferentes soluciones igualmente correctas desde un punto de vista semántico, pero totalmente diferentes desde un punto de vista sintáctico. Por ejemplo, en una base de datos con una tabla de empleados con los atributos: num_empl, nom_empl y num_dpt; y otra tabla departamentos con los atributos: num_dpt y nom_dpt, podemos realizar las siguientes consultas:

a) SELECT nom_dpt

FROM departamentos

WHERE num_dpt NOT IN (SELECT num_dpt FROM empleados);

b) SELECT nom_dpt

FROM departamentos

WHERE NOT EXISTS (SELECT *

FROM empleados

WHERE empleados.num_dpt=departamentos.num_dpt);

Ambas responderían de forma totalmente correcta a la pregunta “Obtén los nombres de los departamentos que no tienen ningún empleado”. Así pues, el problema está más en los tipos de corrección que en los tipos de respuesta. Por eso nosotros hemos necesitado implementar una estrategia basada en juegos de pruebas y conjuntos de resultados, que permite evaluar de forma objetiva si la solución proporcionada por el estudiante es correcta.

Además esta herramienta ayuda a los profesores a diseñar las pruebas de evaluación y corregirlas de forma automática, aportando también la opción de revisar cualitativamente las soluciones aportadas por los estudiantes. Para terminar, destacar que la corrección proporcionada por el sistema LEARN-SQL cumple los siguientes objetivos:

- Generar un almacén o repositorio de ejercicios o preguntas sobre una base de datos y sus posibles respuestas en forma de sentencias SQL. Las preguntas quedan guardadas en el almacén de forma permanente y pueden ser utilizadas posteriormente en diferentes pruebas de evaluación.

- Integrar las diferentes pruebas de evaluación en Moodle, que es el *Learning Management System* elegido tanto por la UPC como por la UOC para sus nuevos proyectos de campus virtual, que permita:
 1. Generar cuestionarios a partir de las preguntas previamente disponibles en el almacén.
 2. Corregir cualquier tipo de sentencias sobre bases de datos utilizando SQL. Esta corrección ha de ser:
 - Automática: la intervención del profesor no debe ser necesaria en el proceso de corrección.
 - Interactiva: el estudiante debe obtener *feedback* en cuestión de segundos acerca de la corrección de su solución.
 - Informativa: el estudiante debe obtener *feedback* sobre los errores que ha cometido en su solución y recibir una posible orientación para solucionarlos.
 - Escalable: el tiempo de respuesta, desde el punto de vista del estudiante, no debe verse condicionado por el número de usuarios del sistema.
 - Extensible: las nuevas necesidades que puedan aparecer deben ser fácilmente incorporables al módulo.
 3. Permitir la supervisión del profesorado del resultado de la corrección automática.
 4. Gestionar las calificaciones de los estudiantes.

2. ARQUITECTURA DE LEARN-SQL

La arquitectura de LEARN-SQL destaca por su flexibilidad en la implementación y distribución geográfica de los componentes.

En LEARN-SQL optamos por almacenar las preguntas en un repositorio o almacén externo en una base de datos de cuestiones. Este repositorio no tiene que ser único. Cada usuario o grupo de usuarios puede definir y usar su propio repositorio de preguntas. El repositorio de preguntas está implementado en PostgreSQL 8.2 (un SGBD relacional *Open Source*). El profesor define las preguntas a través de la aplicación de gestión de cuestiones. Cada pregunta se puede reutilizar en diferentes cuestionarios o pruebas que define el profesor con ayuda del campus virtual *Moodle*. Una vez se ha definido el cuestionario, éste queda a disposición de los estudiantes. *Moodle* es el *software* de libre distribución escogido que nos permite definir los cuestionarios y que permite a los estudiantes resolver los cuestionarios. *Moodle* es también una comunidad *Open Source* de más de 150,000 miembros que publica un campus virtual (*Learning Management System*) en 75 idiomas en 160 países [10].

Para realizar la corrección de las preguntas se utilizan una serie de servicios *Web* (*Servicios Web Correctores*), que funcionan de forma independiente a *Moodle* y que prueban la solución del estudiante contra una segunda base de datos de pruebas. Los Servicios *Web* encargados de la corrección necesitan acceder a las preguntas disponibles para saber como corregirlas (consultar sus soluciones, los elementos a evaluar que se les

han definido, así como consultar y almacenar las respuestas a cada uno de esos elementos a evaluar).

En la figura 2.1 podemos ver gráficamente el funcionamiento de esta arquitectura.

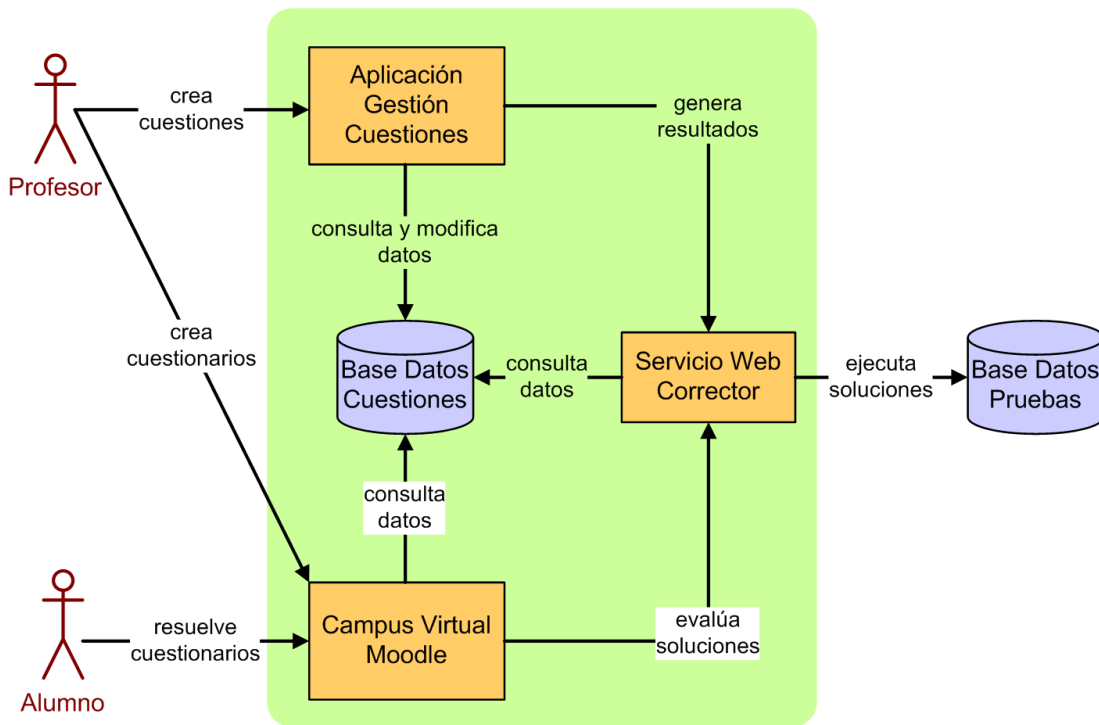


Figura 2.1: Arquitectura de LEARN-SQL

Esta arquitectura presenta unas propiedades muy importantes:

1. Asincronismo en la corrección: Tal como se ha mencionado, ni el usuario, ni *Moodle*, ni tan siquiera el Servicio *Web* quedan bloqueados durante la corrección. El estudiante solicita una corrección y puede seguir trabajando. *Moodle* envía una petición al Servicio *Web* y continúa atendiendo al resto de usuarios. El Servicio *Web* recibe una petición de corrección, pero eso no hace que no pueda recibir otras mientras atiende la primera.
2. Distribución de los componentes: Tanto los Servicios *Web* como los repositorios de preguntas se identifican mediante URLs. Así pues, pueden estar ejecutándose en cualquier máquina conectada a internet y mediante su URL *Moodle* será capaz de encontrarlos. Como efectos secundarios de esta propiedad, obtenemos:
 - Corrección en paralelo: En algunos casos, el algoritmo de corrección puede ser relativamente complejo y en otros casos pueden necesitarse ciertas herramientas con un tiempo de respuesta relativamente malo. Sea cual sea el motivo, en caso de necesitar una corrección interactiva, podemos utilizar la potencia de cálculo de tantas máquinas como sea necesario. El límite podría ser tener una máquina corrigiendo en exclusiva para cada estudiante.
 - Compartición de los repositorios de preguntas y Servicios *Web* entre diferentes instalaciones de *Moodle*: Proponer problemas para una cierta asignatura es un trabajo tedioso y especialmente delicado en el caso de definir métodos de autocorrección. Así pues, compartir todo el trabajo realizado entre profesores de diferentes lugares aparece casi como una

necesidad. En caso de que se considere apropiado, diferentes instancias de *Moodle* pueden compartir un mismo almacén de datos y los Servicios *Web* que éste tenga asociados.

3. DISEÑO DE JUEGOS DE PRUEBAS Y ALMACÉN DE PREGUNTAS

LEARN-SQL facilita al profesor el diseño de las pruebas que se utilizarán para corregir cada pregunta. Para cada pregunta se define una batería de juegos de pruebas que permitirá evaluar si la solución del estudiante es correcta. Cada juego de pruebas está diseñado para comprobar un solo error, de tal forma que se traten todos los posibles casos. La nota final se calcula en función del número de juegos de pruebas superados.

Para poder hacer esto, el almacén de preguntas guarda también la solución del profesor para cada pregunta y el resultado de la ejecución de esa solución para cada uno de los juegos de pruebas definidos. Las soluciones de los profesores únicamente pueden ser accedidas desde la aplicación de gestión de cuestiones.

Para ilustrar este apartado seguiremos un caso práctico. Supongamos las dos tablas *empleados* y *departamentos* presentadas en el apartado de introducción con los siguientes esquemas:

empleados(num_empl, nom_empl, sueldo, ciudad_empl, num_dpt, num_proj)

departamentos(num_dpt, nom_dpt, planta, edificio, ciudad_dpt)

Sobre estas tablas queremos definir una pregunta que obtenga el nombre y el sueldo de los empleados que trabajan en los departamentos con *num_dpt* 1 o 2. Para definir esta pregunta accederemos a la herramienta *Gestión de Cuestiones* que estará instalada en el ordenador del profesor. Una vez en la aplicación, en primer lugar, seleccionaremos la categoría, esquema y temática de la pregunta tal como puede verse en la figura 3.1. Podemos elegir entre los valores existentes o definir nuevos valores. Y a continuación, introduciremos el título y el enunciado de la pregunta. También podemos concretar una dificultad, un autor y un tipo de solución. Los estudiantes disponen de un fichero adjunto con las extensiones de los esquemas de las tablas. Para los datos del fichero adjunto, el profesor debe introducir el resultado esperado que debería obtener el estudiante al introducir una solución correcta para la pregunta definida.

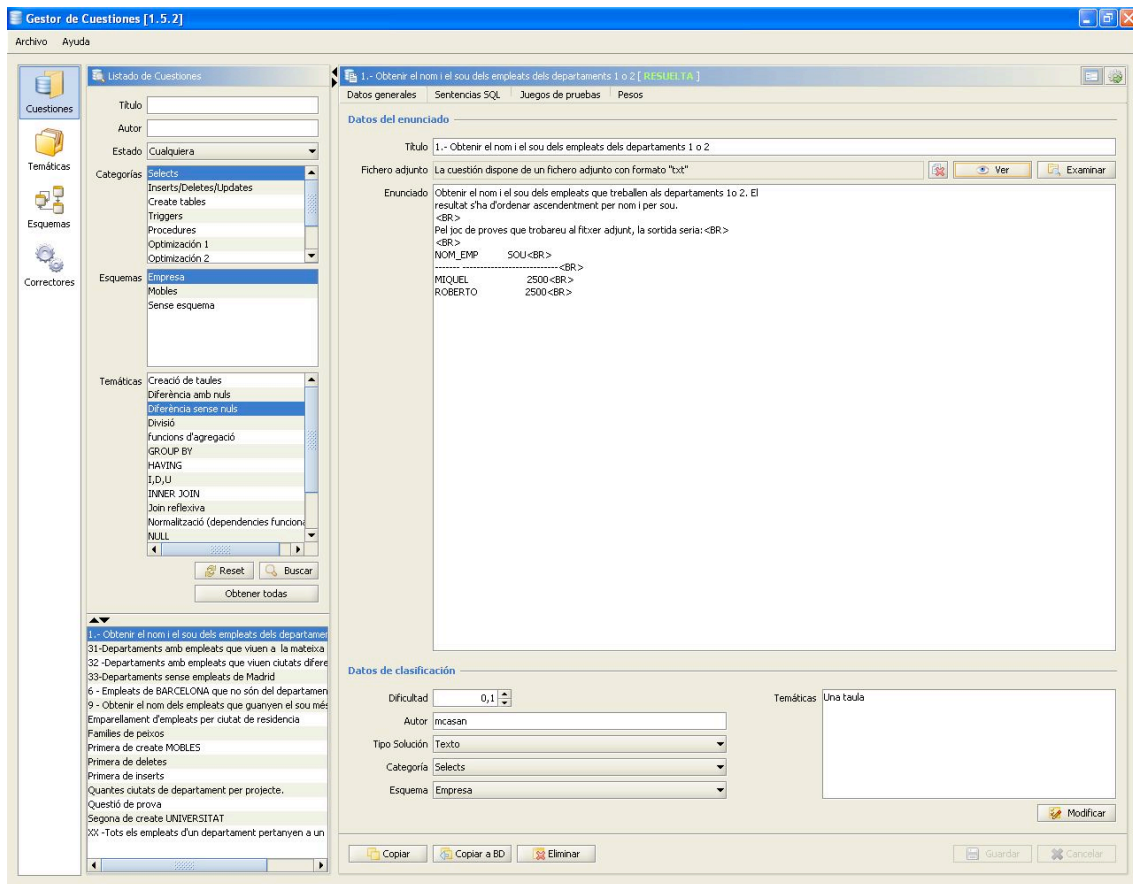


Figura 3.1: Alta de una cuestión en la herramienta de gestión de cuestiones. Enunciado de la cuestión.

Para continuar con el diseño de la pregunta pasamos a detallar las inicializaciones de las tablas que consistirán en su creación, la solución a la pregunta y la limpieza de las tablas que será ni más ni menos que su borrado. Todos estos pasos se muestran en la figura 3.2.

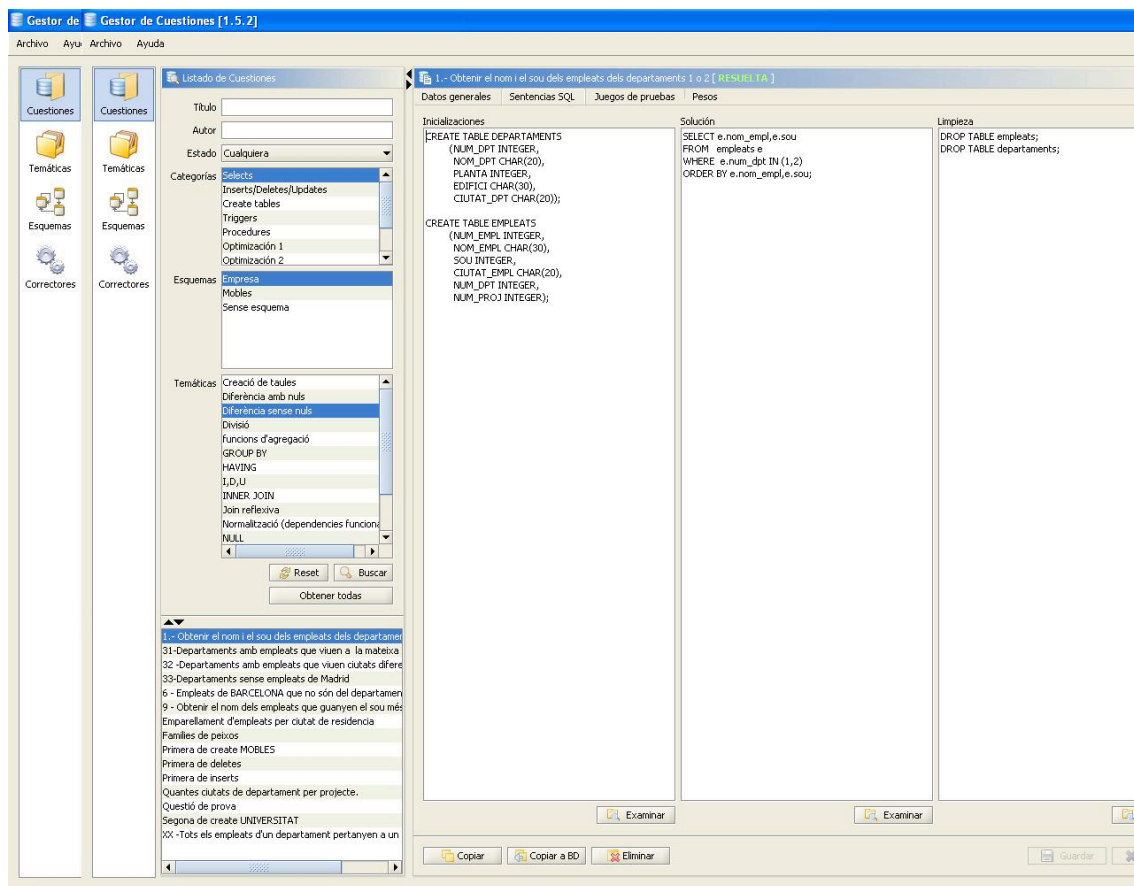


Figura 3.2: Alta de una cuestión en la herramienta de gestión de cuestiones. Inicializaciones de las tablas y solución de la cuestión.

En la figura 3.3, 3.4, 3.5 y 3.6 damos de alta los diferentes juegos de pruebas. Un juego de pruebas es una prueba unitaria, que con unos datos concretos trata de valorar si una solución a una pregunta funciona para estos datos. Estos juegos de prueba permiten establecer el grado de corrección de la respuesta del estudiante a la pregunta. Cada juego de pruebas debe comprobar un solo error posible, de tal forma que al final cuantos más juegos superados tenga el estudiante, más alta será la nota de la pregunta. Para cada juego de prueba hay que introducir los datos de entrada, un título, una descripción y el mensaje de error que verá el estudiante en caso que no superar el juego de pruebas. Este mensaje debe orientar al estudiante para que pueda corregir su solución. Finalmente una vez definidos todos los juegos de prueba de la pregunta, se asigna un peso a cada uno, como se puede ver en la figura 3.7. La nota de la pregunta se calcula sumando los pesos de los juegos superados, suponiendo que la pregunta solo se contesta una vez.

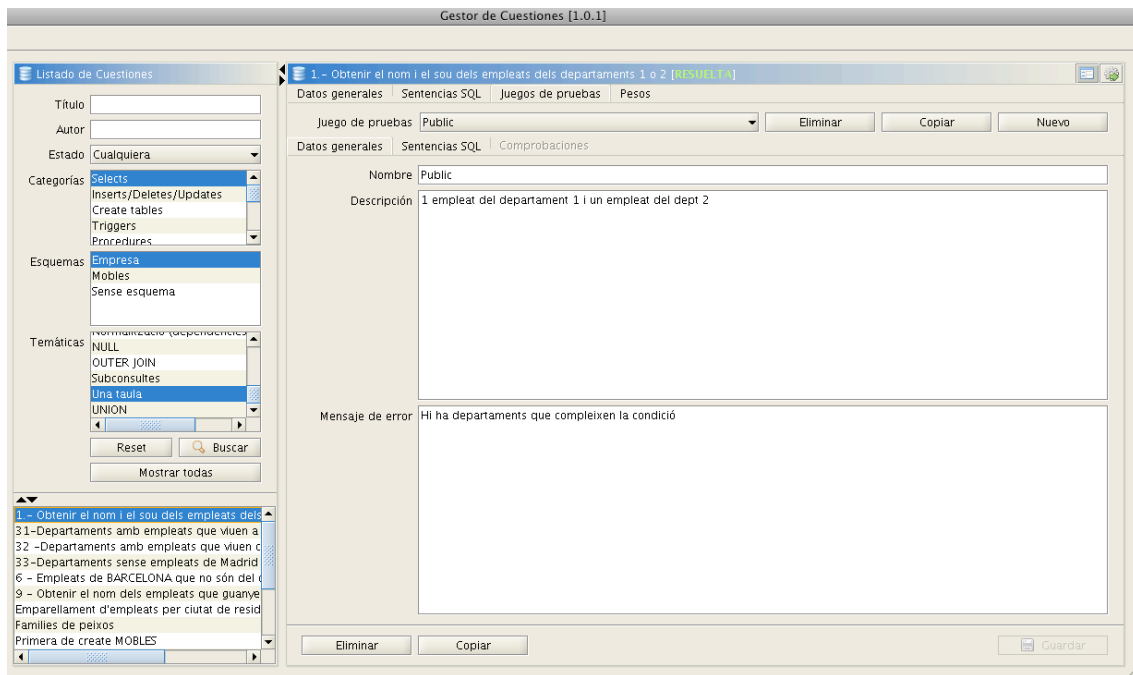


Figura 3.3: Alta de una cuestión en la herramienta de gestión de cuestiones. Definición de los juegos de pruebas.

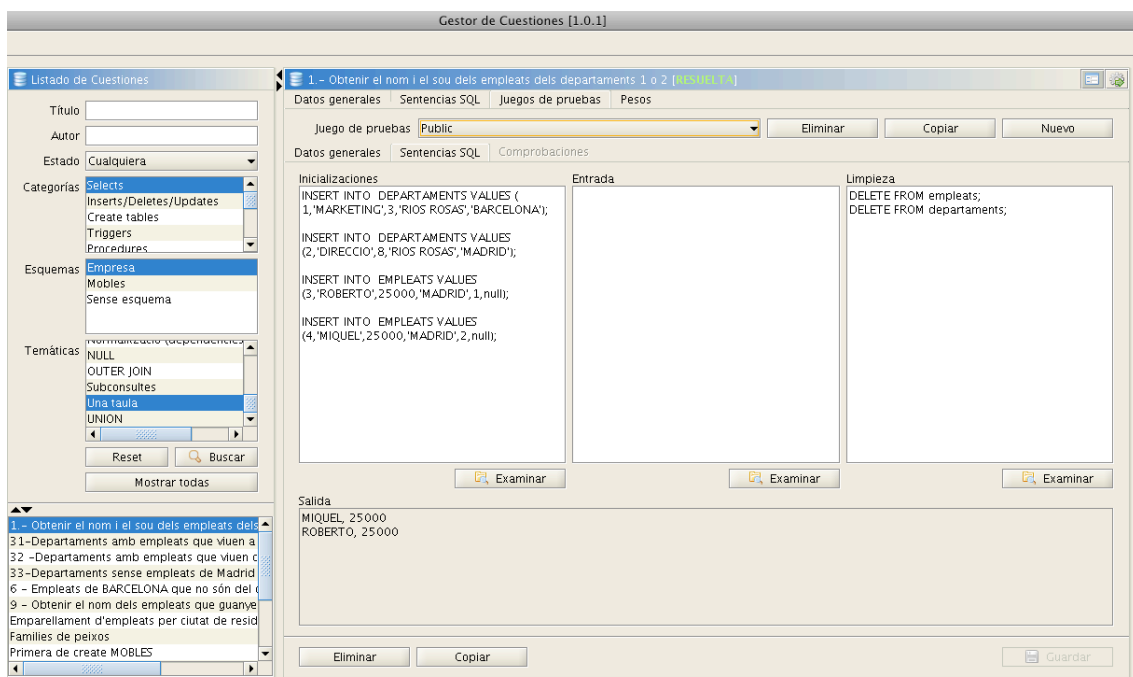


Figura 3.4: Alta de una cuestión en la herramienta de gestión de cuestiones. Definición de los juegos de pruebas.

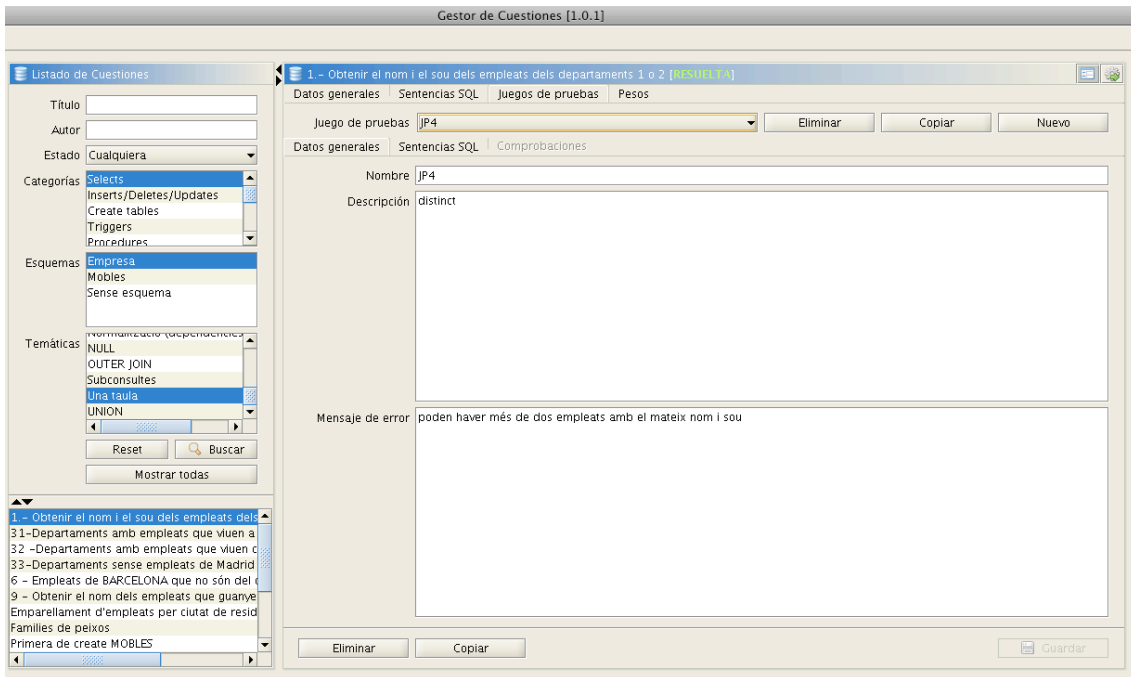


Figura 3.5: Alta de una cuestión en la herramienta de gestión de cuestiones. Definición de los juegos de pruebas.

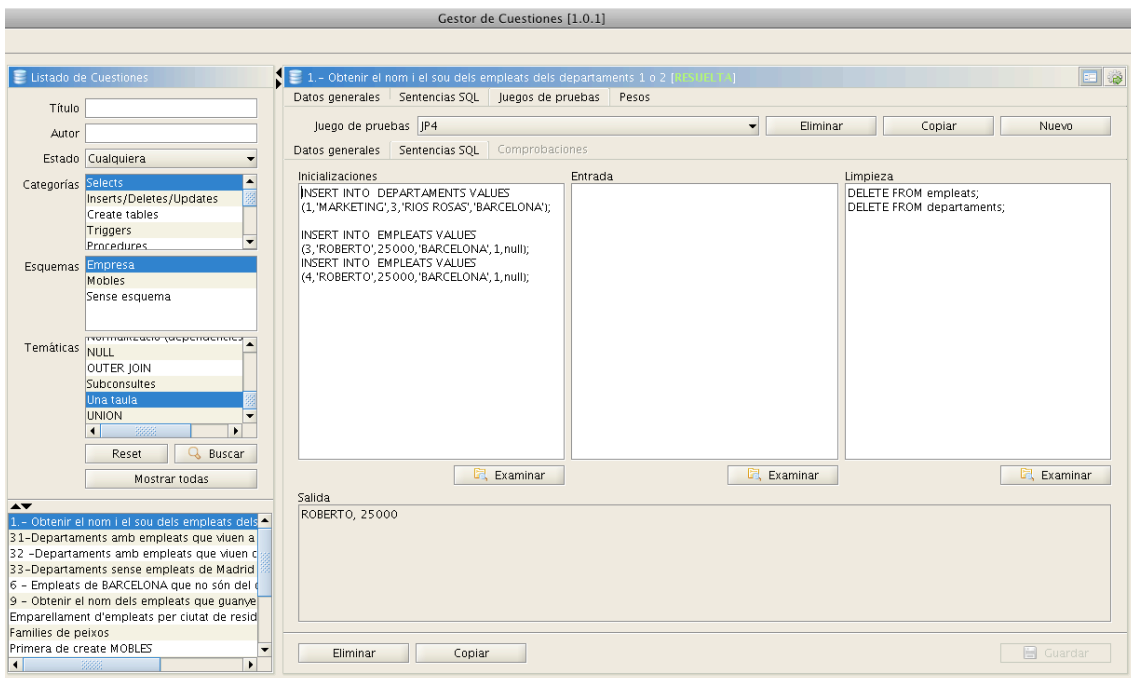


Figura 3.6: Alta de una cuestión en la herramienta de gestión de cuestiones. Definición de los juegos de pruebas.

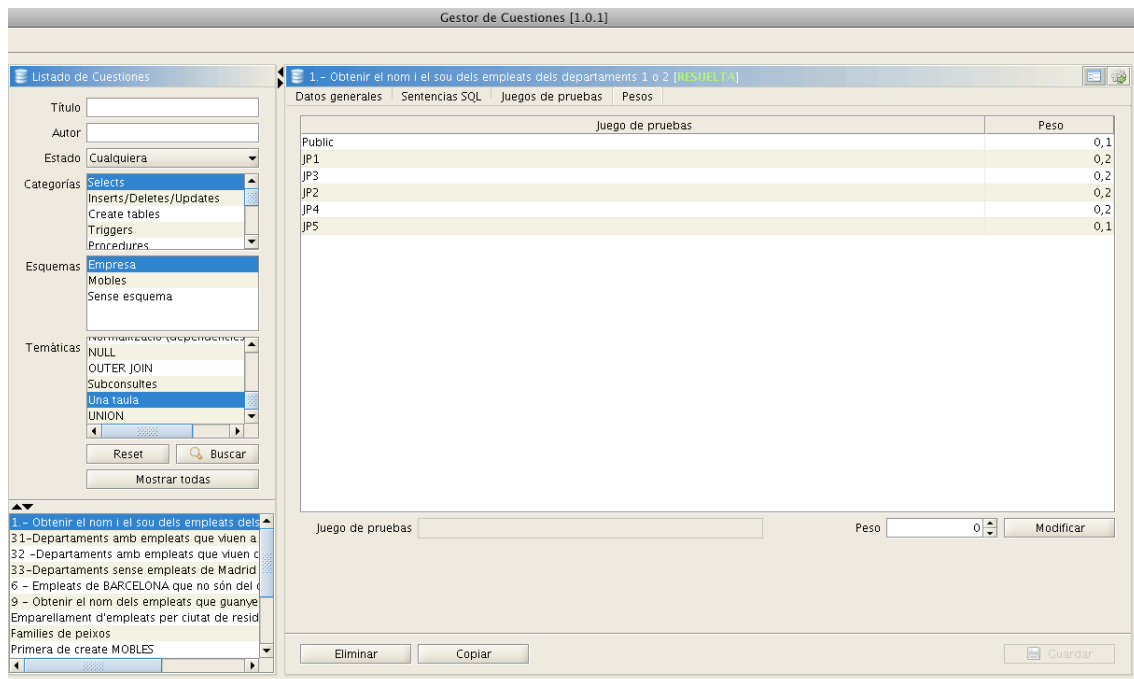


Figura 3.7: Alta de una cuestión en la herramienta de gestión de cuestiones. Asignación de pesos a cada juego de pruebas.

Una vez finalizado el diseño de la pregunta esta queda almacenada en un almacén de preguntas que puede ser accedido por otros profesores y posteriormente se utilizará para el módulo de corrección automática.

4. FUNCIONAMIENTO DE LA HERRAMIENTA DE CORRECCIÓN AUTOMÁTICA

Los cuestionarios o pruebas se definen a través del campus virtual *Moodle*. Aunque *Moodle* dispone de un módulo propio que permite definir cuestiones y utilizarlas posteriormente en diferentes cuestionarios, no hemos utilizado el módulo de cuestionarios de *Moodle*. El motivo principal es que la corrección de las preguntas de los estudiantes de *Moodle*, se realiza comprobando que el texto introducido por el estudiante coincide exactamente con la solución proporcionada por el profesor. La presencia de un espacio en blanco de más en la solución del estudiante, puede hacer que la respuesta sea considerada incorrecta, porque la corrección está basada en la comprobación carácter a carácter entre la solución del alumno y la del profesor.

Por ello, nos planteamos la posibilidad de desarrollar un nuevo módulo de cuestionarios para *Moodle*, que ampliara las posibilidades de corrección de módulo por defecto. LEARN-SQL utiliza este nuevo módulo de cuestionarios para definir los diferentes cuestionarios o pruebas. Las preguntas se obtienen del repositorio de cuestiones externo. Cada pregunta se puede reutilizar en diferentes cuestionarios o pruebas que define el profesor con ayuda del campus virtual *Moodle*. Una vez se ha definido el cuestionario, éste queda a disposición de los estudiantes.

A continuación, explicamos el funcionamiento del módulo de corrección automática de LEARN-SQL de forma gráfica utilizando capturas de pantallas directamente extraídas de la aplicación *Web*.

En primer lugar crearemos un cuestionario tal como se muestra en la figura 4.1. Para crear un cuestionario le daremos un nombre, una descripción y una fecha de apertura y cierre. Se define el número de intentos permitidos y la penalización que tiene cada reintento. También se indica el tipo de corrección que se hará:

- Binaria indica puntuación de 0 si no se evalúa como correcto alguno de los elementos a evaluar o 10 si todos los considerados son correctos; o real que indica que la puntuación es proporcional al número de elementos a evaluar evaluados correctamente.
- Si la nota será la mejor de todos los intentos o bien la última.
- Si se le mostrará su propia respuesta al estudiante, para facilitarle su edición en posteriores reintentos o no.

Además, se han definido dos mecanismos de seguridad (para forzar la resolución del cuestionario de forma presencial en la propia aula, en caso de que se considere necesario): la restricción de máquinas desde las cuales se puede ver el cuestionario y la necesidad de introducir una palabra de paso para verlo. En la figura 4.1 se observa igualmente que se puede abrir el cuestionario de forma selectiva para los diferentes grupos de la asignatura, pero la gran diferencia respecto a los cuestionarios ya existentes es que debemos introducir una URL, un nombre de usuario y una palabra de paso para que *Moodle* pueda acceder a nuestro repositorio de preguntas.

sumen 100) y seleccionar el o los Servicios *Web* que queremos que se encarguen de corregir cada una de las preguntas (cabe recordar que los Servicios *Web* capaces de corregir cada una de las preguntas deben haber sido registrados anteriormente en el almacén de preguntas). Una vez seleccionados los Servicios *Web*, su disponibilidad puede ser comprobada mediante el botón correspondiente.

Temàtiques disponibles

GROUP BY ▼

Id	Títol	Enunciat	Afegeix
8	Departaments amb empleats que viuen a la mateixa ciutat	Obtenir el número i nom dels departaments que tenen 2 o més empleats que viuen a la mateixa ciutat. Pel joc de proves que trobareu al fitxer adjunt, la sortida seria: NUM_DPT NOM_DPT ----- 3 MARKETING	<input type="checkbox"/>
10	Quantes ciutats de departament per projecte.	Proposeu una sentència SQL per obtenir, per a tots els projectes, de quantes ciutats són els departaments dels empleats que hi treballen. Concretament volem, en aquest ordre, el número de projecte, el nom de projecte i el nombre de ciutats. Per al contingut corresponent al fitxer adjunt la sortida ha de ser: NUM_PROJ NOM_PROJ ----- 1 IBOTEL 1	<input type="checkbox"/>
15	Sumar sous de projectes de pressupost nul.	Per a cada projecte que té el valor nul a la columna "pressupost", obtenir la suma dels sous dels empleats assignats al projecte. Ordeneu el resultat per número de projecte. Per al joc de proves que trobareu al fitxer adjunt, la sortida seria: NUM_PROJ SUM ----- 1 40000	<input type="checkbox"/>
61	Quants empleats treballen a cada ciutat	Per a cada ciutat, quants empleats hi treballen. Ordeneu el resultat per ciutat. Per al joc de proves que trobareu al fitxer adjunt, la sortida seria: CIUTAT COUNT ----- Barcelona 1	<input checked="" type="checkbox"/>

Afegeix les qüestions seleccionades

Qüestions del qüestionari remot

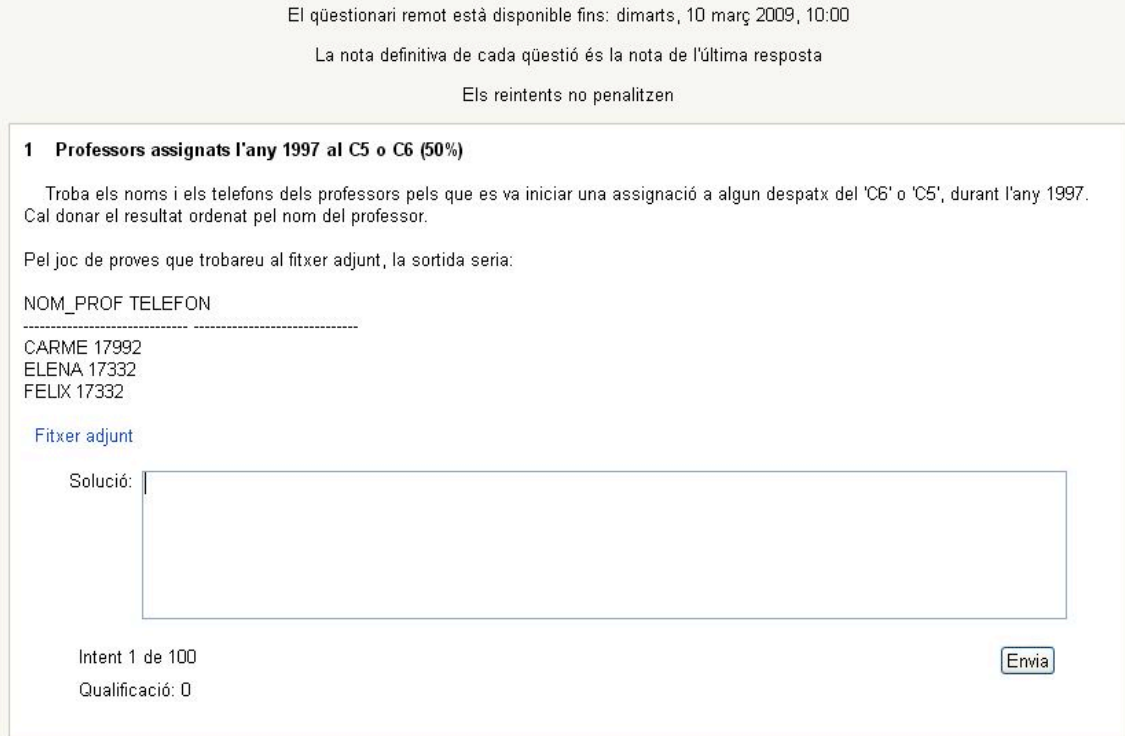
Id	Títol	Percentatge (sobre 100)	URLs	Suprimeix
55	Professors assignats l'any 1997 al C5 o C6	<input type="text" value="50"/>	http://fibdoweb.fib.upc.edu/dabd/services/corrector2?wsdl	<input type="checkbox"/>
56	Professors assignats a despatxos més grans de 15	<input type="text" value="20"/>	http://fibdoweb.fib.upc.edu/dabd/services/corrector2?wsdl	<input type="checkbox"/>
57	DNI major que dos identificadors de centre	<input type="text" value="20"/>	http://fibdoweb.fib.upc.edu/dabd/services/corrector2?wsdl	<input type="checkbox"/>
58	Despatxos grans	<input type="text" value="10"/>	http://fibdoweb.fib.upc.edu/dabd/services/corrector2?wsdl	<input type="checkbox"/>

Verifica els serveis web seleccionats

Figura 4.2: Selecció de las preguntas del cuestionario. Asignación de pesos y Servicios Web encargados de la corrección de cada pregunta del cuestionario.

Tal como se muestra en la figura 4.4, para cada una de las preguntas seleccionadas por el profesor, los estudiantes pueden ver además del enunciado y un enlace hacia el posible fichero adjunto de cada pregunta, un campo de texto donde introducir su

respuesta, el número del intento en que se encuentran actualmente, la puntuación y un botón que envía la respuesta del campo de texto al Servicio *Web* para su corrección. El hecho de haber implementado la corrección de forma asíncrona, permite que, en caso de que el Servicio *Web* tarde demasiado en llevarla a cabo, el estudiante vuelve a esta misma página (no encontrando en este caso el correspondiente botón de envío) y puede emplear el tiempo en responder otra pregunta o incluso enviarla a corregir en paralelo con la anterior.



El qüestionari remot està disponible fins: dimarts, 10 març 2009, 10:00

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

1 Professors assignats l'any 1997 al C5 o C6 (50%)

Troba els noms i els telèfons dels professors pels que es va iniciar una assignació a algun despatx del 'C6' o 'C5', durant l'any 1997. Cal donar el resultat ordenat pel nom del professor.

Pel joc de proves que trobareu al fitxer adjunt, la sortida seria:

NOM_PROF TELEFON

CARME 17992
ELENA 17332
FELIX 17332

[Fitxer adjunt](#)

Solució:

Intent 1 de 100 Envia

Qualificació: 0

Figura 4.3: Vista de una pregunta del cuestionario remoto en Moodle.

Finalmente, existe una página en la que el profesor puede visualizar los resultados de los estudiantes en el cuestionario. Para cada uno de ellos vemos los intentos realizados y la puntuación obtenida en cada una de las preguntas, así como la media ponderada de todas ellas. Una vez cerrado el cuestionario, esta misma página permite también introducir modificaciones en las puntuaciones de cada pregunta.

5. EXPERIENCIA Y RESULTADOS OBTENIDOS

LEARN-SQL se está usando en el presente curso en las clases de laboratorio de la asignatura DABD (Diseño y Administración de Bases de Datos) de la titulación de ingeniería en informática de la Facultat d'Informàtica de Barcelona de la UPC.

LEARN-SQL se utiliza con dos objetivos: por una parte, como herramienta de evaluación en las sesiones de laboratorio y, por otra parte, como recurso de aprendizaje y preparación de las sesiones de problemas.

En las sesiones de laboratorio en que se ha utilizado la herramienta, los alumnos obtienen *feedback* del sistema, que les permite saber el grado de corrección de su respuesta en función del número de juegos de prueba superados. El sistema proporciona

una calificación provisional que normalmente es matizada posteriormente por el profesor, teniendo en cuenta criterios de eficiencia, claridad etc.

Adicionalmente, se ha utilizado LEARN-SQL preparando cuestionarios de ejercicios para que los estudiantes los resolvieran fuera de clase, desde algún ordenador con conexión a Internet. Se conciben estos ejercicios como una forma de trabajo semipresencial más eficaz que la publicación de listas de ejercicios con o sin solución: el estudiante no puede ver la solución, con lo que se evita que dé el ejercicio por resuelto sin haber meditado lo suficiente y mire la solución demasiado temprano y también se evita que lo dé por resuelto teniéndolo mal o que quede con la duda sobre la corrección de su respuesta.

Respecto a los resultados obtenidos, desde el punto de vista docente su utilización ha sido plenamente satisfactoria tanto para los profesores como para los alumnos. En efecto, las soluciones entregadas por los alumnos en las sesiones de laboratorio son mejores, lo que se ha reflejado en un aumento de las calificaciones obtenidas. Ello es debido a que los estudiantes intentan mejorar su solución hasta que supere todos los juegos de pruebas a los que es sometida. Como el sistema proporciona información sobre las posibles razones por las que no se supera un juego de pruebas, los estudiantes disponen de pistas de los errores cometidos y pueden corregir los fallos de sus respuestas. El uso de la herramienta fuera del aula no ha tenido mucho éxito, pero algunos alumnos han aprovechado este recurso y ello les ha impulsado a dirigirse al profesor para solucionar dudas.

Debido a los buenos resultados obtenidos en la asignatura de DABD, LEARN-SQL se utilizará en la asignatura de BD (Bases de Datos), que es una asignatura obligatoria tanto en la ingeniería en informática como en las ingenierías técnicas, a partir del próximo curso académico, en las sesiones de laboratorio.

6. CONCLUSIONES Y TRABAJO FUTURO

La elaboración de la herramienta LEARN-SQL nos ha permitido obtener resultados en dos ámbitos. Por un lado hemos adaptado al EEES el funcionamiento de las clases pasando de clases de problemas magistrales a clases donde el estudiante es el actor principal. Por otro lado, hemos desarrollado una herramienta *software* para dar soporte a un aprendizaje más participativo y menos presencial. LEARN-SQL ha sido testado durante dos semestres en una asignatura de bases de datos de la FIB de la UPC y una primera versión de la herramienta ha sido presentada en [1]. Desde la perspectiva de los estudiantes, [3] y [4] describen herramientas educativas que ayudan a los estudiantes en la comprensión de diversos temas de bases de datos. En esta línea LEARN-SQL ha resultado muy útil porque puede ser utilizada en cualquier momento y en cualquier lugar, proporcionando *feedback* automático y una calificación provisional. Desde el punto de vista del profesor LEARN-SQL ayuda a planificar las clases de problemas de forma clara y sistemática y sobretodo ayuda considerablemente en el proceso de corrección y calificación en una línea similar a [7].

Como trabajo futuro nos gustaría mejorar la herramienta utilizando Query Containment siguiendo las ideas presentadas en [5]. También planeamos añadir técnicas de aprendizaje colaborativo y la posibilidad de personalizar el proceso de aprendizaje a las necesidades de cada estudiante.

AGRADECIMIENTOS

Este trabajo se ha realizado gracias a un proyecto de mejora de la calidad docente de la UPC, al proyecto (2007MQD00202) concedido por la Generalitat de Catalunya y al proyecto de investigación PERSONAL (TIN2006-15107-C02) concedido por el Ministerio de Educación y Ciencia.

REFERENCIAS

- [1] Abelló, A., Urpí, T., Rodríguez, M.E., Estévez, M. “Extensión de Moodle para facilitar la corrección automática de cuestionarios y su aplicación en el ámbito de bases de datos”, disponible en: <http://sntrv-moodle.unex.es/moodlemoot07/>.
- [2] Dekeyser, S., de Raadt M., Lee, T.Y., “Computer Assisted Assessment of SQL query Skills”, *Proc. of 8th ADC, 2007*.
- [3] Dietrich S., Eckerd E., Piscator K., “WinRDBI: A Windows-based Relational Database Educational Tool”, *Proc. of the twenty-eighth SIGCSE technical symposium on Computer science education*, pp. 126-130, 1997.
- [4] Douce C., Livingstone D., Orwell J., “Automatic Test Base Assessment of Programming: a review”, *ACM Journal of Educational Resource in Computing*, 5(3), 2005.
- [5] Kenny C., Pahl C., “Automated tutoring for a database skills training environment”, *Proc. of SIGCSE’05*, pp. 59–62, 2005.
- [6] Mitrovic A., “Learning SQL with a computerized Tutor”, *Proc. of SIGCSE’98*, pp. 307–311, 1998.
- [7] Prior J., Lister R., “The backwash effect on SQL skills grading”, *Proc. Of ITiCSE’04*, pp. 32–36, 2004.
- [8] Sadiq S., Orłowska M., Sadiq W., Lin J., “SQLator—an online SQL learning workbench”, *Proc. of ITiCSE’04*, pp. 223–227, 2004.
- [9] Soler J., Prados F., Boada I., Poch J., “Utilización de una plataforma de e-learning en la docencia de Bases de Datos”, *Proc. of Jenui*, pp. 581-588, 2006.
- [10] Cole J. “Using Moodle” O’Reilly Press, 2005.