



Creación de una herramienta informática para la resolución de problemas de ingeniería química



María Isabel Vázquez Navarro

Profesora del Departament d'Enginyeria Química de l'Escola Tècnica Superior d'Enginyeria de la Universitat de València



Jesús V. Albert Blanco

Profesor del Departament d'Informàtica de l'Escola Tècnica Superior d'Enginyeria de la Universitat de València



Paula Marzal Doménech

Profesora del Departament d'Enginyeria Química de l'Escola Tècnica Superior d'Enginyeria de la Universitat de València



Sergio Velasco Ballestín

Becario del Departament d'Enginyeria Química de l'Escola Tècnica Superior d'Enginyeria de la Universitat de València

| Fecha presentación: 30/06/2008 | Aceptación: 10/09/2008 | Publicación: 22/12/2008

Resumen

El objetivo de este proyecto es crear una aplicación en web con una amplia colección de problemas de balances de materia. La herramienta potencia que el estudiante aprenda a resolver estos problemas de forma sistemática y le proporciona un mecanismo de resolución guiado, incidiendo en la metodología y los pasos a seguir. Los profesores introducen los problemas de forma interactiva y automatizada. La resolución por parte del estudiante sigue unos pasos guiados, pero dejando la máxima capacidad de decisión al mismo. Durante la resolución el estudiante es informado de los errores cometidos y se le proporcionan ayudas.

Palabras clave: *e-learning*, innovación educativa, balances de materia, herramienta informática, resolución guiada, autoevaluación

Resum

L'objectiu d'aquest projecte és el de crear una aplicació en web amb una àmplia col·lecció de problemes de balanços de matèria. L'eina potencia que l'estudiant aprenga a resoldre aquests problemes de forma sistemàtica i li proporciona un mecanisme de resolució guiat que incideix en la metodologia i els passos a seguir. Els professors introdueixen els problemes de forma interactiva i automatitzada. La resolució per l'estudiant segueix uns passos guiats però deixant la màxima capacitat de decisió a ell mateix. Durant la resolució, l'estudiant és informat dels errors comesos i se li proporcionen ajudes.

Paraules clau: *e-learning*, innovació educativa, balanços de matèria, eina informàtica, resolució guiada, autoavaluació

Abstract

The aim of this project is to create a web application containing a great collection of mass balance problems. This tool boosts students to learn to solve problems in a systematic manner and provides them a guided mechanism which emphasizes the methodology and the solving procedure. The problems are introduced by the instructor following some interactive and automated steps. Solving is done through some guided steps although the students are allowed to take decisions. Information about mistakes and helping are supplied on line.

Keywords: *e-learning*, educational innovation, mass balance, computing tool, guided solving, self-assessment



1. Introducción

La resolución de problemas en ingeniería química constituye el objetivo final y más importante de todas las asignaturas. La correcta, y casi la única forma de aprender a resolverlos es que el estudiante se enfrente a la realización individual de muchos problemas. Con este objetivo, en todas las asignaturas de ingeniería química se entrega a los estudiantes una colección de problemas adicional a la que se realiza en clase y se les da numerosa bibliografía que contiene problemas resueltos y propuestos, para que los resuelvan en casa. Sin embargo, la respuesta del estudiante no siempre es satisfactoria ya que los problemas son complejos, no sólo en la ejecución sino también en el planteamiento, y el estudiante suele estancarse en varios momentos y no puede o no sabe continuar si no recurre a la ayuda del profesor. Si el estudiante está trabajando en casa, esto supone paralizar la resolución del problema hasta que se le resuelva la duda y esta situación de paro-consulta-paro se puede repetir varias veces a lo largo de un problema. Por otra parte, cuando el estudiante lo intenta y no sabe continuar, no siempre recurre al profesor sino que, en ocasiones, simplemente abandona.

Los problemas de aplicación de los balances de materia (Reklaitis 1983; Felder 2000), cuya resolución se aborda con esta herramienta, son los primeros con los que se enfrenta un estudiante de la titulación de ingeniería química, pero también muchos otros estudiantes que cursan titulaciones en los que tienen una asignatura básica y general de esta especialidad. La principal característica de estos problemas es que son muy diferentes a todo lo que pueden haber visto y resuelto anteriormente, ya que tienen un enunciado bastante estándar y una resolución sistemática muy larga y laboriosa.

El primer paso es entender el enunciado y extraer todos los datos, que no siempre son evidentes. La experiencia demuestra que aquí el estudiante se encuentra con una gran dificultad, hasta el punto de no saber continuar o de que todo lo que se haga a continuación esté mal.

El siguiente paso, consiste en realizar un esquema (diagrama de flujo) del proceso donde se reflejen cada una de las etapas que se mencionan en el enunciado, y donde se sitúan los datos del problema. En la mayoría de los casos, este diagrama de flujo se les proporciona pero, aún así, se detectan errores a la hora de interpretarlo y de situar en él los datos del enunciado.

Por último, la solución del problema pasa por establecer una serie de balances de materia y de relaciones adicionales que llevan al planteamiento y resolución de un sistema de ecuaciones, generalmente no lineal.

La aplicación informática desarrollada permite al estudiante resolver los problemas siguiendo estos pasos y comprobar, en diferentes momentos de su resolución, si se avanza adecuadamente. Si es así, seguir y, si no, pedir ayuda para poder continuar. La complejidad de los problemas hace que el tipo de pistas y de ayudas que se contemplen sean muy variados, pero aun así se ha logrado sistematizarlas y automatizarlas.

El propósito principal de la aplicación es que el estudiante adquiera destreza, de un modo interactivo, sistemático y cómodo, en la resolución de problemas de este tipo

mediante la elaboración, de forma guiada y autocorrectiva, de muchos problemas de balances de materia de complejidad cada vez mayor.

2. Metodología y Resultados

Para desarrollar y poner en marcha la aplicación se cuenta con participantes y usuarios de dos perfiles:

- Perfil profesor: Este perfil se encarga de dar de alta los problemas en la aplicación, siguiendo unos pasos sistemáticos y que se han automatizado al máximo para minimizar el trabajo del profesor, facilitar su introducción y reducir al mínimo la posibilidad de errores. Los profesores también tienen acceso a estadísticas sobre la realización de los problemas por parte de los alumnos: número de fallos, ayudas solicitadas, problemas que más se han realizado, etc.
- Perfil alumno: Éste es el perfil principal de la aplicación. El alumno resuelve de forma guiada los problemas introducidos previamente por los profesores y tiene acceso a estadísticas sobre sus resultados y su evolución.

2.1 Arquitectura de la herramienta informática

La arquitectura lógica de la aplicación es la arquitectura web de varias capas, basada en la separación en capas de módulos de la aplicación con funcionalidades independientes (Hall 2001). Esta separación funcional y lógica se traduce en un software bien estructurado que facilita el desarrollo de la herramienta, la escalabilidad del sistema, la reutilización del código, los cambios posteriores, así como otros muchos factores que tienen gran importancia a la hora de construir sistemas software de calidad.

La aplicación que se ha desarrollado está estructurada en cuatro capas:

- La capa de almacenamiento permite guardar problemas e información de usuarios
- La capa de procesamiento es la que realiza la resolución de los problemas, la evaluación y la implementación de funciones
- a capa del controlador recibe las entradas del usuario y las transmite a la capa de procesamiento
- La capa de vista del sistema es la que visualiza la información en el navegador para el usuario.

La Figura 1 muestra la estructura en capas y la forma en que todas ellas se enlazan para dar como producto la aplicación distribuida final. A continuación se describe brevemente la tecnología utilizada en cada una de ellas.

En la capa de Almacenamiento, el sistema utiliza varias tecnologías complementarias. Como sistema gestor de base de datos se utiliza MySQL, ya que la licencia es gratuita para aplicaciones no comerciales y es un sistema de bases de datos muy potente y fiable, fácilmente integrable con Java, y que cumple con creces los requisitos de la aplicación. El método de conexión entre esta capa y la siguiente (capa de procesamiento) se realiza mediante JDBC ya que, aunque hay otras tecnologías más cómodas para el programador, se comprobó que utilizando JDBC se ganaba en eficiencia y rendimiento (puesto que permite hacer un diseño mucho más *ad-hoc* para cada aplicación) y no resultaba excesivamente compleja su utilización en este caso.

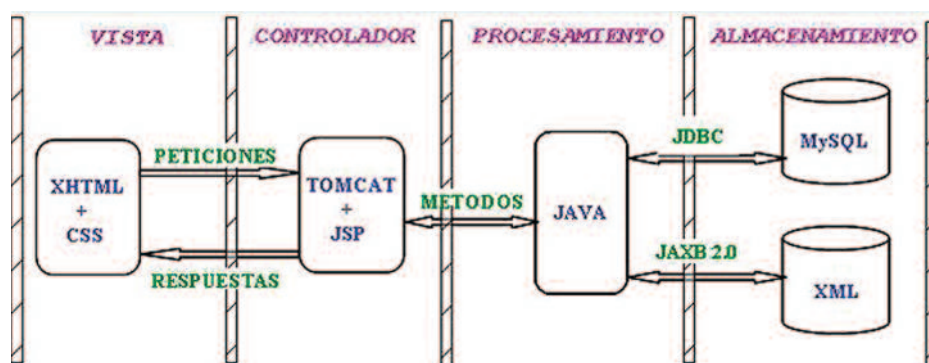


Figura 1. Arquitectura de la aplicación

Con esta estructura, se pueden almacenar todos los problemas que requiera la aplicación. Se ha previsto que, en principio, el número de problemas almacenados no sea superior a 100, aunque con este diseño no hay ningún inconveniente en cuanto a la escalabilidad de la aplicación y se podrán guardar muchos más sin perder rendimiento.

Por otra parte en la capa de almacenamiento, además de la base de datos MySQL se tiene un fichero en formato XML que almacena las ayudas que, en cada sección del problema se proporcionan al estudiante, en función del tipo de fallo que comete. Esta parte se ha realizado con JAXB 2.0 (Java Api For XML Binding).

La capa de Procesamiento del Sistema constituye el núcleo del mismo y está programada íntegramente en Java (Eckel, 2002). Como se ha comentado anteriormente, al crear este modelo se han tenido muy en cuenta las características de los problemas y los usuarios, y se ha generado un modelo de objetos acorde a todos estos requisitos. La parte principal y más compleja de este modelo es la que permite modelar y estructurar el problema en base a sus características y los pasos de resolución pero, además, se han creado otros objetos para leer y gestionar el XML de ayudas, leer archivos de parámetros y gestionar las estadísticas.

En toda aplicación con varias capas, debe existir una capa de Controlador del sistema, que se encarga de interactuar entre la vista y el modelo o capa de procesamiento (modelo-vista-controlador, MVC). El controlador es el que recibe entradas y solicitudes de la vista, se encarga de pasarlas al modelo para que las procese y devuelva una respuesta que es enviada otra vez a la vista.

La implementación del controlador se ha realizado con la tecnología Java Server Pages (JSP) y el servidor de aplicaciones Tomcat (Hall 2004). Se ha hecho uso también de la tecnología Ajax (Javascript Asincrono XML) ya que ésta da mucha velocidad a la web y hace que el uso de una aplicación web se asemeje mucho a la clásica interacción con cualquier aplicación de escritorio. De este modo se consigue que el sistema sea más usable y el usuario no tenga que esperar siempre las recargas de la página por parte del navegador.

La capa de la Vista del sistema es la de nivel superior y es, básicamente, lo que el usuario puede ver de la aplicación, con la cual interactúa y en la que introduce datos, peticiones, etc. y recibe información. En esta capa se ha aplicado XHTML para describir el contenido de la página web y CSS

para crear la apariencia de la aplicación. Se han intentado utilizar tipografías ligeras y colores suaves para no recargar excesivamente la pantalla y crear una aplicación amigable (Figura 2). Por otra parte, se ha realizado una vista bastante independiente del resto de las capas de la aplicación para que sea fácil cambiar la apariencia de la misma sin tener que modificar secciones en otras capas o módulos.

Por ejemplo, si quisiéramos realizar la misma aplicación en un entorno de escritorio en lugar de en un navegador web, simplemente tendríamos que adaptar ligeramente el controlador y cambiar la vista, pero las otras capas no se verían afectadas. De esta forma, con esta separación en módulos, hacemos que la aplicación sea más fácilmente modificable y gane en escalabilidad.

2.2 Características y desarrollo de la aplicación

Como se ha comentado anteriormente, los problemas cuya resolución se aborda con esta herramienta son los primeros con los que se enfrenta un estudiante de la titulación de ingeniería química, y otros estudiantes que cursan titulaciones en las que tienen una asignatura básica y general de ingeniería química.

La herramienta se ha desarrollado utilizando como base un problema de dificultad media, representativo del conjunto, y con todas las características y variables que pueden tener estos problemas (Figura 3). En todos los problemas tendremos un enunciado en forma de texto y una figura que representa el diagrama de flujo del proceso.



Figura 2. Vista de la aplicación

El yoduro de metilo puede producirse a partir de ácido yodhídrico con un exceso de metanol mediante la reacción:



En la figura se muestra el diagrama de flujo del proceso. Además de las composiciones (molares) que aparecen en ella, se dispone de los siguientes datos:

- Al reactor se alimentan 2 moles de CH_3OH por cada mol de HI.
- La conversión del HI en el reactor es del 50%.

Calcular, si el proceso funciona en estado estacionario:

- a) Los moles de CH_3I producidos
- b) Los moles de CH_3OH alimentados
- c) Los moles recirculados y la composición de la corriente de recirculación

por kmol/h de HI fresco (corriente A) alimentado al proceso.

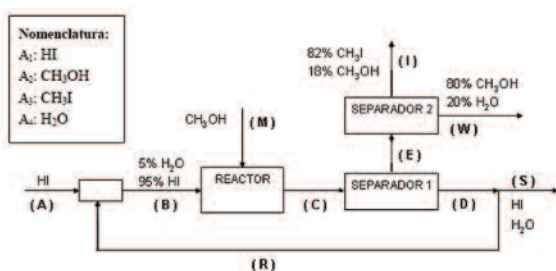


Figura 3. Problema Modelo utilizado para el desarrollo de la herramienta

La creación de problemas por parte del profesor se aborda como una sucesión de pantallas en las cuales se le va pidiendo que introduzca información relativa al problema para que la aplicación pueda almacenarlo. Las primeras pantallas que aparecen son la de introducción del título del problema, dificultad y enunciado del mismo. Cada problema se puede catalogar como fácil, de dificultad media, o difícil y, para introducir el enunciado, se pueden utilizar varios procedimientos, como escribir el texto o importar un documento word o pdf.

Para entender las siguientes etapas es importante conocer la nomenclatura habitual de ingeniería química. Un *componente* es cualquier especie química que participa en el

proceso. Hablamos de *corrientes* cuando nos referimos a las líneas que van de una etapa a otra del proceso y que en el diagrama de flujo están catalogadas con letras. Por otra parte, llamamos *volumen de control* a cualquier punto del proceso en el que cambia el caudal o la composición de las corrientes. Los siguientes datos que debe introducir el profesor son básicos y se refieren al número de componentes, corrientes, reacciones y volúmenes de control que intervienen en el problema. También debe introducir algunos detalles de las *reacciones*, como el valor de los coeficientes estequiométricos de cada componente, y algunos datos o características de cada *volumen de control*, como por ejemplo el tipo de volumen de control (reactor, mezclador, separador, divisor). Para que la introducción de todos estos datos resulte cómoda se ha recurrido a formatos muy intuitivos, en los que únicamente se introducen números o se escoge en desplegables (Figura 4).

A partir de esta información, la aplicación genera una matriz (Figura 5) en la que el profesor debe introducir los datos de *caudales y composiciones del problema*. Estos datos son muy importantes, pues son los que hay que interpretar y extraer de cada enunciado. Por defecto, todos los campos están vacíos y los que queden así tras la introducción de los valores conocidos se contabilizarán como incógnitas del problema que deben coincidir con el número de relaciones que se plantearán posteriormente.

El siguiente paso es establecer las *ecuaciones* características del problema. El número de incógnitas de estos problemas y, por tanto, de balances a plantear, suele ser muy elevado, por lo que este paso podría resultar muy laborioso y tedioso para el profesor y, por otra parte, sería una causa de posibles errores. Por esta razón, al desarrollo de esta etapa de la herramienta se le ha dedicado un especial esfuerzo para que, a partir del número de componentes y del número y tipo de volúmenes de control del problema (datos que ya se han introducido previamente) la aplicación genere automáticamente todos los balances

Figura 4. Diferentes pantallas de introducción de datos básicos del problema



Figura 5. Introducción por el profesor de datos de caudal y composición

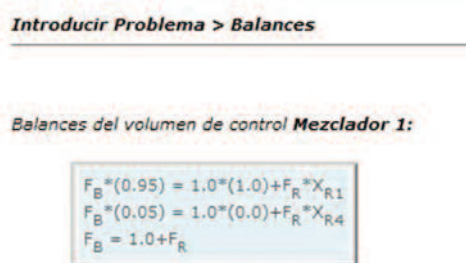


Figura 6. Balances de materia generados para el mezclador



Figura 7. Introducción por el profesor del dato de grado de conversión



Figura 8. Buscador de problemas

específicos de cada problema. En la Figura 6 se muestran los balances generados para uno de los volúmenes de control de este problema.

Para completar el planteamiento del conjunto de ecuaciones que van a resolver estos problemas, además de los balances de materia, hay que escribir unas *relaciones "ad-hoc"* que, si bien son propias de cada problema, también se han podido catalogar en varios tipos. Por ejemplo, es muy habitual tener como dato de estos problemas valores del grado de conversión de un reactivo, de la selectividad de un producto de reacción, una relación entre caudales de dos corrientes o entre la composición de dos componentes en la misma u otra corriente, entre otras opciones. Estos datos no se han podido sistematizar hasta el punto de que sea la propia herramienta quien genere la ecuación, pero sí de forma que al profesor se le pida una mínima información, como se muestra en la Figura 7.

Por último, el profesor introduce los valores correctos de todas las incógnitas del problema para que después puedan ser comparados con los que obtenga el estudiante.

La tarea del profesor ha finalizado y, a partir de ahora, el estudiante tiene disponible el problema para su resolución, que comienza por la elección del mismo. La primera pantalla que se le presenta (Figura 8) le permite decidir en tres niveles: puede seleccionar un problema concreto, seleccionar por dificultad o, simplemente, que se le muestren todos. Por otra parte, el estudiante dispone de un breve resumen de algunas características del problema: número de reacciones, número de corrientes y número de componentes que intervienen en el problema, que le permite obtener una idea aproximada de su grado de dificultad. Desde esta pantalla el estudiante puede retroceder, o continuar y comenzar su resolución. Es importante destacar que, en cada sección del apartado de resolución de problemas, el alumno tiene en todo momento accesible un botón desde el que se activa una pantalla emergente independiente para visualizar el enunciado, en la forma que se ha mostrado en la Figura 3, para poder consultar cualquier información del problema.

Tras estos pasos, y con esta información, comienza realmente el proceso de resolución en el cual se muestran pantallas semejantes al proceso de introducción del problema por el profesor, con la diferencia de que, en cada paso de la resolución, se va informando al alumno de los fallos que comete, y éste tiene la doble posibilidad de reflexionar y corregir por sí mismo el error, o solicitar ayudas que le faciliten la resolución, pero le penalizarán en la puntuación final.

Por ejemplo, la pantalla de introducción de datos y contabilización de variables puede tener en un momento dado de la resolución del estudiante el aspecto que se muestra en la Figura 9. Vemos que el estudiante ha introducido los datos y, tras validar, se han contrastado con los que previamente había introducido el profesor, comprobando que hay 19 fallos. La aplicación informa al estudiante de dos maneras diferentes: por una parte marca en rojo las casillas correspondientes a los valores incorrectos y, por otra, contabiliza e informa del número total de fallos.

El estudiante puede corregir el valor introducido en cada celda o recurrir al botón de ayuda que aparece en cada una

de las celdas incorrectas. En tal caso, dependiendo del tipo de variable y tipo de fallo, aparece un mensaje de ayuda diferente, como se muestra también en la Figura 9. Puesto que el tipo de fallos y, por lo tanto el tipo de ayudas a proporcionar, son similares para todos los problemas de balances de materia, se han logrado agrupar y sistematizar y forman parte de la base de datos de la aplicación, por lo que no tienen que ser introducidas por el profesor para cada uno de los problemas.

Cabe destacar que, si tras varios intentos y solicitud de ayudas, el estudiante no consigue acabar correctamente una sección, la aplicación le permite continuar. Para ello captura del apartado del profesor los resultados correctos de esa etapa y los utiliza para las siguientes. De esta forma, se garantiza que el estudiante pueda trabajar todos los pasos del problema, y que los errores de una etapa no influyan en la siguiente.

El siguiente paso en la resolución de estos problemas por el estudiante consiste en plantear los balances de materia en todos y cada uno de los volúmenes de control. Para ello el estudiante debe decidir, en primer lugar, cuántos posibles volúmenes de control -es decir, cuántos entornos donde puede aplicar estos balances- tiene el problema y, en cada uno de ellos, debe plantear los balances correctamente.

La aplicación permite al estudiante introducir los balances de forma bastante autónoma (Figura 10), sin dirigirle demasiado y, a la vez, realiza una comparación con los balances correctos, para detectar los posibles fallos cometidos por el estudiante. Por otra parte, el tipo de fallos y, por tanto de ayudas a suministrar, es bastante variado, ya que puede deberse a la introducción de un número inadecuado de balances o a una forma incorrecta de los mismos, pero se ha logrado realizar una buena agrupación y sistematización de los tipos de fallos y de las ayudas. Esta parte de la aplicación ha resultado ser bastante laboriosa de implementar, ya que resulta difícil compaginar la libertad de introducción (no dar demasiadas pistas al estudiante) con la capacidad de comparación (que se simplifica dando un formato rígido). Otra dificultad importante es el hecho de que en un sistema con S componentes se pueden plantear $S+1$ balances, pero sólo S son independientes, y la aplicación se ha programado para que reconozca como correcta cualquier combinación. Por otra parte, hay algunas excepciones a esta regla, como son el reactor, en el que es recomendable plantear un conjunto determinado de ecuaciones, o el divisor en el que, independientemente del número de componentes, sólo se puede plantear un balance.

El siguiente paso es la introducción de lo que hemos llamado relaciones *ad-hoc* o relaciones propias de cada problema. En este caso, se ha optado por plantear un formato similar a la introducción de los balances, que deje capacidad de decisión al estudiante, pero que facilite la comprobación. Si el número, tipo o forma de relaciones introducidas no es correcto, se le proporcionan al estudiante mensajes de error y ayudas que han sido automatizadas y catalogadas de acuerdo con el tipo de error.

El último paso de estos problemas es la resolución del sistema de ecuaciones resultante. Llegado este momento el estudiante puede guardar el problema, resolver por cual-

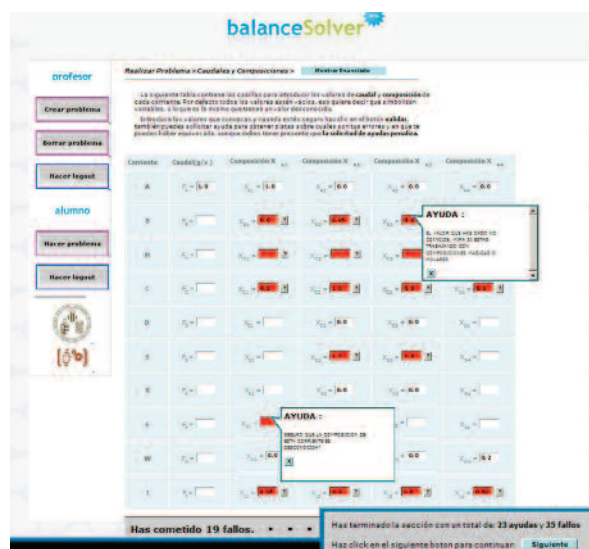


Figura 9. Introducción por el estudiante de datos, errores y ayudas



Figura 10. Introducción por el estudiante de los balances

quier procedimiento el sistema de ecuaciones (balances y otras relaciones) que ha planteado y, con los resultados obtenidos, volver a la aplicación y comparar sus resultados con los que haya introducido el profesor. En este sentido, uno de los siguientes objetivos de este proyecto es dotar a la aplicación del *output* que permita la resolución del sistema de ecuaciones mediante un paquete matemático (Matlab, Mathcad, etc.)

Conclusiones

Dentro del marco de la convergencia europea las formas de enseñanza deben cambiar desde una docencia basada en las clases impartidas por el profesor, en las que el estudiante es un objeto pasivo que únicamente recibe información, hacia otra docencia basada en el trabajo del estudiante dirigido a conseguir las competencias requeridas de forma autónoma, que le permita seguir formándose a lo largo de la vida, a medida que su iniciativa personal y/o sus compromisos laborales se lo exijan.

En este sentido, la aplicación informática que se ha desarrollado es bastante amigable para los diferentes usuarios y contribuirá a facilitar el trabajo autónomo del estudiante y, lo que es muy importante, permitirá que cada estudiante

lo ajuste a su nivel, ya que en cada momento puede escoger la dificultad del problema que realiza y el número de ayudas que solicita.

La introducción de los problemas requiere un esfuerzo por parte del profesor, pero se ha logrado minimizar los pasos y la información que éste debe introducir, y que la inserción de la misma se haga de forma intuitiva, cómoda y minimizando la posibilidad de cometer errores.

Por otra parte, para los estudiantes, ésta es también una forma interactiva y relativamente entretenida de resolver problemas, que les permitirá realizar muchos y, por tanto, adquirir habilidad y aprender a realizarlos de forma sistemática. Las pantallas en las que se ha estructurado la herramienta, la forma en que se solicita la información y las ayudas que se suministran, se han pensado para que el estudiante vaya adquiriendo conciencia de los pasos en los que falla, y por qué comete esos fallos y, en base a todo ello, vaya mejorando en el planteamiento y resolución de estos problemas.

Se puede concluir, por tanto, que la aplicación cumple el objetivo docente para el que se ha diseñado, en relación a los dos perfiles de usuario. Por otra parte, la creación y uso de esta herramienta no elimina ni reduce las clases presenciales de problemas sino que, por el contrario, la aplicación se ha desarrollado para trabajar de forma paralela y coordinada, ya que sigue los mismos pasos y metodología que el profesor explicará y enseñará a utilizar en clase.

Bibliografía

- Eckel, Bruce (2002). *Thinking in Java*. New Jersey: Prentice-Hall
- Felder, Richard M.; Rousseau, Ronald W. (2000). *Elementary principles of chemical processes*. New York: J. Wiley and Sons
- Hall, Marty; Brown, Larry (2001). *Core Web Programming*. New Jersey: Prentice-Hall PTR/Sun Microsystem Press.
- Hall, Marty; Brown, Larry (2004). *Core Servlets and JavaServer Pages (JSP)*. New Jersey: Prentice-Hall PTR/Sun Microsystem Press.
- Reklaitis, G.V. (1983). *Introduction to material and energy balances*. New York: J. Wiley and Sons.

Tutoriales

Se han realizado dos *demos* que muestran diferentes aspectos de la herramienta desarrollada:

La 1ª demo (**demoA.swf**) muestra el proceso de creación de problemas de balances por parte del profesor (ver en: <http://mmedia.uv.es/display?c=programatic@post.uv.es&name=demo1.swf>).

En la 2ª demo, (**demoB.swf**), se muestra el proceso de resolución del problema por parte del alumno (ver en: <http://mmedia.uv.es/display?c=programatic@post.uv.es&name=demob.swf>).

Estos dos archivos se pueden reproducir con Internet Explorer o con el reproductor de Flash.

| Cita recomendada de este artículo

Vázquez Navarro, María Isabel et al. (2008). Creación de una herramienta informática para la resolución de problemas de ingeniería química. @tic. revista d'innovació educativa. (nº 1) <http://ojs.uv.es/index.php/attic/article/view/41/57> Fecha de consulta, dd/mm/aa