

ENTORNO VIRTUAL DE AYUDA A LA DOCENCIA DE UN CURSO DE PROGRAMACIÓN BÁSICA

Imma Boada, Josep Soler, Ferran Prados y Jordi Poch

Departament Informàtica i Matemàtica Aplicada. Universitat de Girona
Escola Politècnica Superior. Edifici P4. 17071 Girona
{[@ima.udg.es](mailto:imma,soler,fprados,poch)}

RESUMEN.

Presentamos la ampliación realizada sobre la plataforma de e-learning ACME, desarrollada en el departamento de Informática y Matemática aplicada de la Universidad de Girona, con el fin de ser usada como soporte a la docencia de la programación básica. Describiremos las ventajas que nos proporciona dicha plataforma y como se ha aplicado en un curso piloto de introducción a la programación con resultados muy satisfactorios.

PALABRAS CLASE. Software educativo, e-learning, soporte a la docencia, auto-aprendizaje.

OBJETIVOS.

Nuestro objetivo es diseñar e implementar un entorno virtual de trabajo que pueda usarse como herramienta de soporte a la docencia de un curso de introducción a la programación, proporcionando:

- **Un entorno virtual de aprendizaje.**

El alumno podrá resolver problemas planteados por el profesor a través del entorno virtual, el cual de forma automática le facilitará on-line la corrección a las soluciones enviadas. Es decir, dispondrá de un entorno equivalente a un profesor virtual las 24 horas del día, con el único requisito de disponer de una conexión a Internet.

- **Un entorno de evaluación continua.**

El sistema asignará cuadernos personalizados de problemas a los alumnos. Dichos problemas deberán resolverse en unos periodos de tiempo determinados. Al llegar a la fecha límite y a partir de las soluciones enviadas por los alumnos, las cuales habrán sido corregidas de forma automática, el sistema, siguiendo las restricciones fijadas por el profesor, asignará una calificación. El sistema mantendrá todas las soluciones enviadas por el alumno, tanto las correctas como las erróneas, de esta forma el profesor podrá detectar los principales puntos débiles en el proceso de aprendizaje de la asignatura.

El sistema diseñado incorporará también las funcionalidades básicas de comunicación con el alumno (e-mail, foros, etc.) típicas de una plataforma de e-learning.

DESCRIPCIÓN DEL TRABAJO

Motivación

Actualmente son muchos los estudios universitarios en los que se imparte una asignatura que tiene entre uno de sus objetivos básicos la programación de ordenadores. Como ejemplos pueden considerarse las asignaturas de Fundamentos de Informática de los estudios de Ingeniería Industrial y de las Ingenierías Técnicas especialidades Mecánica, Electrónica, Química o la asignatura de Metodología y Tecnología de la Programación de las Ingenierías Técnicas de Informática de Gestión y Sistemas. En estas asignaturas se ha detectado que a pesar de que los alumnos están familiarizados con el uso del ordenador, aprender a programar no les resulta nada fácil. Normalmente asocian el trabajo con el ordenador a navegar por *Internet*, a participar en *chats* o a trabajar con paquetes ofimáticos. Son pocos los que asocian el ordenador con una herramienta programable pensada para resolver problemas.

Se han realizado distintos estudios con el fin de detectar cuales pueden ser las causas del elevado índice de fracaso asociado a las asignaturas de programación. La mayoría de ellos coinciden en que para poder programar deben combinarse una serie de conocimientos y habilidades que la mayoría de alumnos no han aplicado anteriormente (capacidad de abstracción, cierto nivel matemático, aprendizaje de un nuevo lenguaje, unas nuevas normas de escritura,...). Ante la carencia de estos conocimientos el alumno se siente desmotivado, se desanima y va perdiendo el interés por la asignatura. A todos estos factores debemos añadirle el hecho de que generalmente estas asignaturas de introducción a la programación suelen impartirse en los primeros cursos universitarios, los cuales se caracterizan por tener un elevado número de alumnos que dificulta la interacción profesor-alumno vital en este tipo de asignaturas.

El reto que nos fijamos un grupo de profesores del Departamento de Informática y Matemática Aplicada de la Universidad de Girona fue el de desarrollar un sistema que facilitara y motivase el trabajo del alumno en las asignaturas de introducción a la programación y que además pudiera usarse para su evaluación. La ventaja que teníamos es que no partíamos de cero, ya que disponíamos de una plataforma de e-learning, denotada ACME, desarrollada en el mismo departamento como plataforma de Evaluación Continua y Mejora de la Enseñanza de las matemáticas. Dicha plataforma se estaba usando desde el 1998 con resultados muy satisfactorios. Se trataba por tanto de ampliar la plataforma con el fin de soportar las asignaturas de introducción a la programación.

Puesto que la base del trabajo que presentamos está en la plataforma ACME hemos considerado oportuno empezar el documento con una breve descripción de esta plataforma. A continuación presentaremos la ampliación que se realizó de la plataforma con el fin de soportar las asignaturas de programación. Para finalizar se presentarán los primeros resultados y impresiones obtenidos al aplicar el entorno diseñado en un curso piloto de introducción a la programación.

Trabajo previo. La Plataforma ACME.

En esta sección presentamos de forma sintetizada una descripción de los principales módulos que forman la plataforma ACME, sus funcionalidades y los diferentes tipos de usuarios que soporta. Para una descripción detallada véanse [1][2][3].

Descripción de la plataforma

En la figura 1 presentamos un diagrama de los módulos principales que componen la plataforma ACME y la conexión entre ellos. Estos módulos son:

1. Repositorio y Módulo de Introducción de Problemas.

Para cada asignatura, el sistema mantiene en un *repositorio* un conjunto de problemas con la forma de corregirlos y opcionalmente pautas de ayuda para guiar al alumno en el proceso de resolución. Además, asociando un conjunto de valores y parámetros variables a los problemas, el sistema es capaz de generar automáticamente múltiples enunciados.

El profesor a través del *módulo de introducción de problemas*, introduce los problemas en el repositorio siguiendo las especificaciones establecidas en función del tipo de los mismos. Por ejemplo, en las asignaturas de matemáticas además del problema se introducirá el código asociado para la verificación de la respuesta dada por el alumno.

2. Módulo Generador de Cuadernos de Problemas

A partir de un conjunto de parámetros (temática, categoría, nivel de dificultad,...) fijados por el profesor, este módulo selecciona de forma automática conjuntos de problemas del repositorio a partir de los cuales se generan automáticamente cuadernos de trabajo únicos y personalizados para cada alumno.

3. Módulo de Corrección

Este módulo corrige las soluciones enviadas por un alumno a un problema determinado, aplicando el método de corrección específico del mismo. Por ejemplo, en los ejercicios de matemáticas el código de verificación asociado al problema se ejecutará con la solución enviada por el alumno y se analizará si es o no correcta. Debe tenerse en cuenta que una misma expresión matemática puede escribirse de distintas formas todas ellas correctas. El alumno será informado en tiempo real del resultado de la corrección.

4. Módulo de Evaluación Continua

La base de datos del sistema mantiene toda la información relativa a un alumno (problemas asignados, soluciones enviadas,...). De esta forma el módulo de evaluación continua, a partir de la información del alumno y siguiendo los criterios fijados por el profesor, calcula la calificación que le corresponde.

5. Módulo de Auto-aprendizaje

El módulo de auto-aprendizaje facilita al alumno un conjunto de problemas con distintos niveles de ayuda. Dichos niveles se activan de forma automática en función del número de errores que cometa el alumno al intentar resolverlos. En el caso extremo y después de varias soluciones fallidas el sistema le proporciona la solución correcta.

6. Módulo de Consulta de Estadísticas

Este módulo permite al profesor consultar la información que mantiene la base de datos del sistema. Puede obtenerse información de los problemas asignados a los alumnos, de las soluciones enviadas a un determinado problema,...

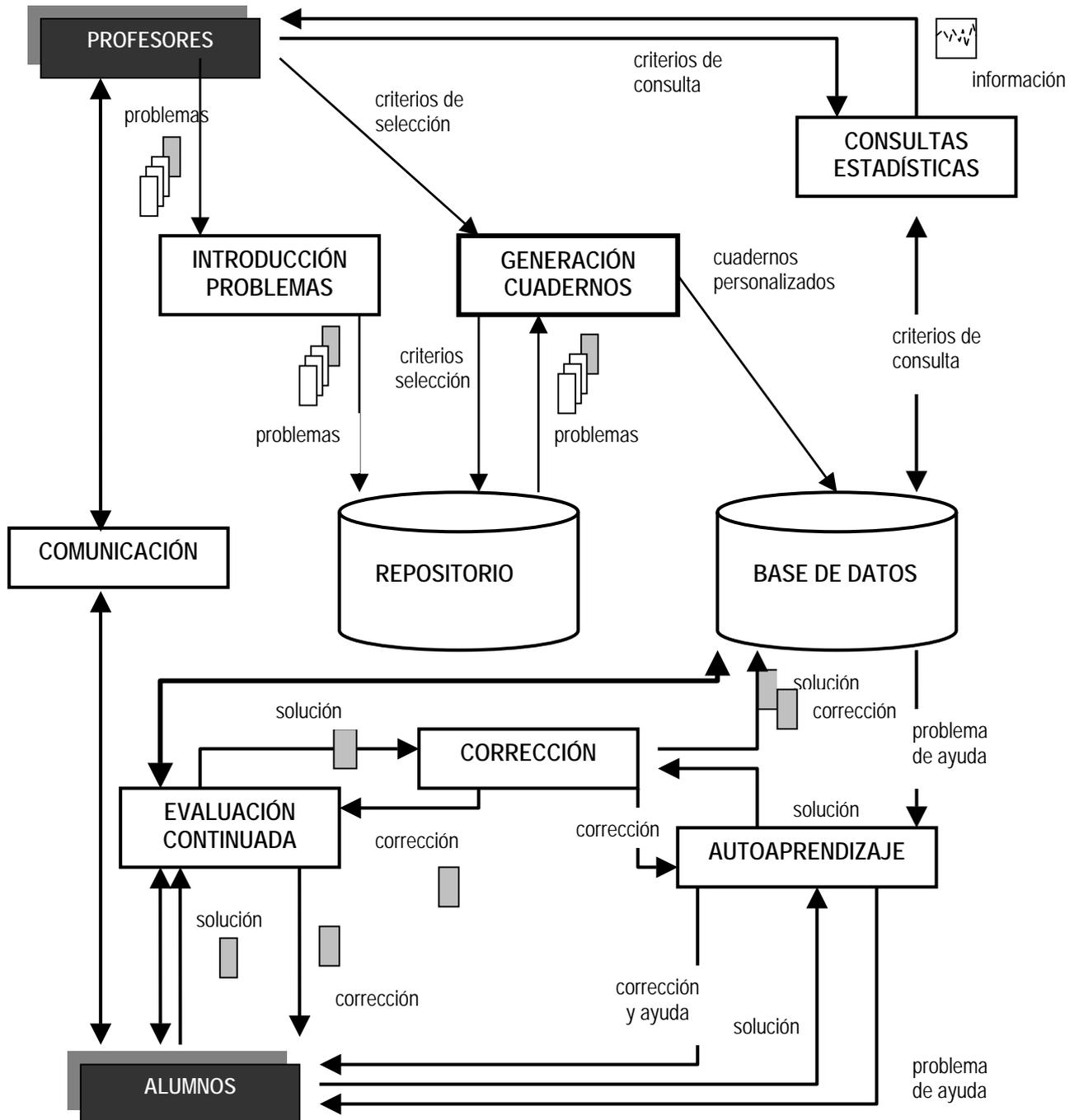


Figura 1. Estructura de la plataforma ACME.

7. *Módulo de Comunicación*

Este módulo proporciona un canal de comunicación entre el profesor y el alumno, permitiendo: el envío de e-mails entre profesor y alumnos, así como la participación en el foro asociado a la asignatura. También permite al profesor adjuntar notas con comentarios informativos a las soluciones enviadas por los alumnos.

Funcionalidades de ACME

Puede verse que la esencia de ACME radica en automatizar el proceso de corrección de problemas. Esta automatización nos permite utilizar el sistema con dos fines totalmente distintos:

- *Como módulo de autoaprendizaje.*
- *Como sistema de evaluación continuada.*

Por último, remarcar que el uso del sistema simplemente requiere de una conexión a Internet, un navegador y el código de entrada al sistema.

Usuarios de ACME

El sistema ACME soporta tres tipos de usuarios:

- **Administrador del sistema.**

Encargado del mantenimiento general del sistema, control de seguridad,...

- **Profesores.**

Todos los profesores que tienen acceso al sistema tienen asignadas una o varias asignaturas y un grupo de alumnos por cada una de ellas. Pueden controlar el trabajo realizado por el alumno, enviarle comentarios mediante los distintos medios de comunicación de que dispone el sistema, detectar las necesidades de cada uno y desde el punto de vista de la asignatura detectar los temas que presentan mayores dificultades e insistir en ellos en las clases presenciales. Cada asignatura tendrá un profesor responsable que será el encargado del mantenimiento de la asignatura, introducir nuevos problemas en el repositorio, introducir la información de los alumnos en el sistema y generar los cuadernos de problemas personalizados.

Todos los profesores pueden acceder al foro o foros de las asignaturas a las que están asignados. También pueden enviar mensajes y adjuntar notas a los problemas de los alumnos de su grupo.

- **Alumnos.**

Los alumnos son los receptores del sistema. Pueden seleccionar una asignatura, visualizar su cuaderno de problemas e imprimir los problemas de un tema concreto. Tienen un profesor asignado al que pueden consultar vía correo electrónico o mediante los foros de la asignatura.

Cuando ha resuelto un problema pueden enviar la solución a corregir y el sistema les informa de forma inmediata del resultado de dicha corrección. Este resultado puede ser *correcto* si la solución es correcta, *error sintáctico* si la solución no puede ser interpretada por el sistema en el caso de problemas de matemáticas o se han producido errores de compilación en el caso de los programas informáticos e *incorrecto* en el caso que la solución sea incorrecta. El alumno puede enviar tantas soluciones como sea necesario hasta conseguir resolver el problema correctamente.

LA PLATAFORMA ACME APLICADA EN LAS ASIGNATURAS DE INTRODUCCIÓN A LA PROGRAMACIÓN.

La plataforma ACME era la herramienta ideal para resolver los problemas que se planteaban en las asignaturas de programación. Ofrecía por un lado un entorno de autoaprendizaje y por otro lado un sistema de evaluación continua, coincidiendo exactamente con los objetivos que nos habíamos marcado. Sin embargo ACME no estaba pensado para resolver problemas de programación, los cuales siguen una metodología de corrección y validación distinta de los problemas matemáticos con lo que debíamos ampliar y adaptar la plataforma. Para situarnos en el tipo de problemas que vamos a tratar empezaremos dando una breve descripción de la estructura y de los objetivos de un curso básico de introducción a la programación.

Las asignaturas de Introducción a la Programación.

Las asignaturas de introducción a la programación tal y como su nombre indica, tienen como objetivo presentar las nociones más básicas de la programación. Un programa podemos definirlo como un conjunto de instrucciones escritas en un lenguaje de programación que indican al computador el conjunto de acciones que debe ejecutar para obtener una solución correcta de un problema concreto en un tiempo finito.

Programar equivale a resolver problemas con el ordenador utilizando un lenguaje de programación. El proceso de aprendizaje requiere adquirir dos tipos de conocimientos, por un lado *estrategias de resolución de problemas* que permitan diseñar soluciones y por otro lado el *aprendizaje de un lenguaje de programación* que permita escribir las soluciones en un lenguaje que sea comprensible para el computador.

Para alcanzar ambos objetivos, los cursos de introducción a la programación suelen dividirse en dos tipos de sesiones:

- *Sesiones de teoría.* El objetivo es familiarizar al alumno con los conceptos más básicos de la programación y con las metodologías de escritura de programas. El alumno debe aprender las estrategias de resolución de problemas. Para facilitar el proceso de aprendizaje, los profesores optamos por sustituir el lenguaje de programación por una notación expresada en el idioma del alumno y denotada notación algorítmica o pseudo-código. La mayoría de los lenguajes de programación usan como idioma base el inglés, por tanto para los alumnos de habla no inglesa usar un lenguaje de programación como lenguaje básico para introducir conceptos añade trabas a todo el proceso de aprendizaje. Por el contrario usar una notación algorítmica elimina la dificultad del idioma y facilita

el proceso de aprendizaje, instrucciones como *if*, *while*, *repeat*... específicos de un lenguaje de programación se convertirán en *si*, *mientras*, *repetir*... El *pseudo-código* está desligado de las especificaciones de un lenguaje concreto y permite introducir de forma más llana los conceptos básicos de la programación. De esta forma, el primer paso para la resolución de un problema consistirá en determinar la secuencia de acciones escritas en *pseudo-código* que nos conduzcan a una solución correcta del problema. Esta secuencia de acciones se conoce con el nombre de *algoritmo*.

- *Sesiones de laboratorio*. En estas sesiones se introduce el lenguaje de programación. Se enseñan las especificaciones y el entorno de desarrollo de un lenguaje de programación concreto (C, Pascal, Java,...) que variará en función de los estudios. En estas sesiones el alumno deberá resolver los problemas propuestos por el profesor, pasando por las diferentes etapas que ello comporta (Ver figura 2):
 - i. Creación de un algoritmo, usando el *pseudo-código* y los conceptos adquiridos en las sesiones teóricas.
 - ii. Creación del programa a partir de la traducción del algoritmo y usando un lenguaje de programación.
 - iii. Corrección de errores
 - iv. Validación del programa.

El proceso de resolución finaliza cuando el programa resuelva el problema que se planteaba inicialmente. Tal y como se refleja en el esquema de la figura 2 el diseño del algoritmo es el punto clave de todo el proceso de resolución del problema. Los errores que se cometan en esta fase se arrastrarán hasta el final y no será hasta el momento de probar su correcta ejecución que nos daremos cuenta de que la solución propuesta no funciona. Debe tenerse en cuenta que esta primera etapa del proceso de resolución se realiza sobre papel, es decir de forma desligada del ordenador, por tanto no disponemos de ninguna técnica que nos permita validar de forma automática su validez.

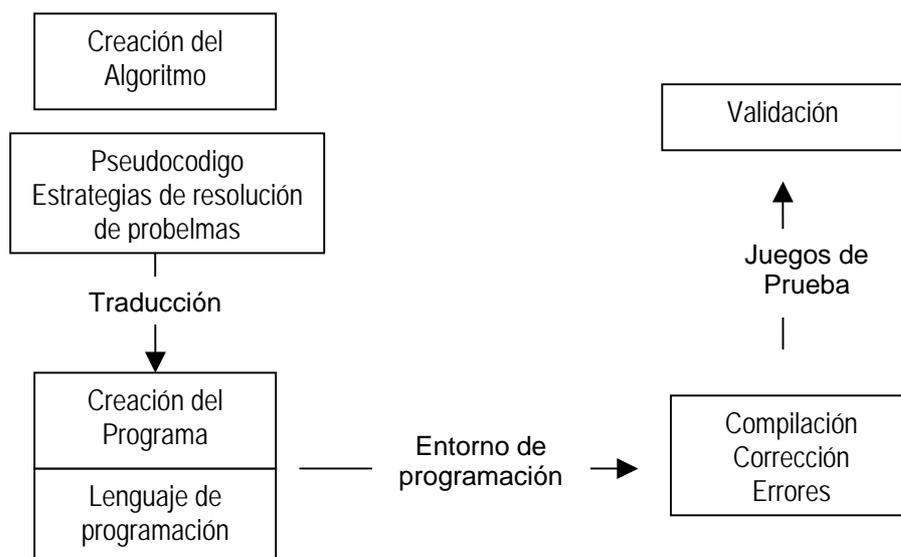


Figura 2. Fases del proceso de resolución de un problema

Es importante remarcar que el trabajo personal del alumno juega un papel fundamental, ya que a programar sólo se aprende programando. Es nuestro objetivo motivar al alumno a trabajar evitando que se desanime ante las primeras dificultades que le surjan. Debemos eliminar todas las trabas que se le presentan. Básicamente hemos detectado dos grandes dificultades:

- (i) Al alumno le cuesta aplicar las técnicas de resolución de problemas y escribir los algoritmos.
- (ii) En el caso de ser capaz de escribir el algoritmo le resulta difícil validarlo, es decir comprobar si la solución que obtiene a través del algoritmo es realmente la solución correcta del problema.

Los ordenadores son capaces de compilar y ejecutar programas escritos en un lenguaje de programación, pero no los algoritmos escritos en *pseudo-código*. Por tanto, no podemos usar el ordenador para validar los algoritmos. Sin duda alguna estas dificultades podrían resolverse si dispusiéramos de más horas de teoría para poder presentar muchos más ejemplos a los alumnos y si fuera posible realizar una atención personalizada a los alumnos. Desgraciadamente disponemos de poco tiempo y el número de alumnos es demasiado elevado como para poder atenderlos de forma individualizada.

La solución que nosotros proponemos es usar la plataforma ACME para solventar todos estos problemas.

AMPLIACIÓN DE ACME

Las ampliaciones que debían realizarse sobre la plataforma afectaban principalmente al módulo de corrección del sistema (ver figura 1). Este debe ampliarse para soportar la corrección tanto de algoritmos como programas.

La corrección de un algoritmo/programa sigue dos etapas diferentes:

1) Compilación del algoritmo/programa

El alumno escribe la solución en un fichero de texto. Una vez escrita la solución se envía al sistema y este la traduce a un lenguaje comprensible para el ordenador. Este proceso de traducción es el que se conoce como compilación. En el caso de que el programa esté escrito en un lenguaje de programación, el módulo de corrección usará el compilador correspondiente. En el caso de que la solución esté escrita en lenguaje algorítmico se usará el intérprete de *pseudo-código* creado para el propio sistema. Este intérprete genera un programa ejecutable en JVM (Java Virtual Machine) comprensible para el ordenador.

El proceso de compilación se repetirá hasta que la solución propuesta por el alumno este libre de errores. Una vez enviada la solución se corrige y se comunican los errores al alumno que puede corregirlos y enviar una nueva solución. En el momento en que el código está libre de errores de compilación empieza la siguiente fase del módulo de corrección.

2) Validación del algoritmo/programa.

Un problema puede tener varias formas de resolverlo. Diremos que la solución enviada es correcta si a través de ella se obtiene la solución esperada. El profesor cuando introduce un problema en el repositorio del sistema introducirá también varios juegos de prueba con su correspondiente resultado, de forma que el módulo de corrección los pueda seleccionar de forma aleatoria y comprobar si a partir de la solución propuesta por el alumno se obtiene la solución esperada. En este caso la solución será correcta. En el caso de que no sea correcta, se mostrará al alumno la solución que se obtiene con su programa y la solución correcta que se esperaba. En el caso de incorrección el alumno puede enviar una nueva solución repitiéndose de nuevo todo el proceso.

Una vez realizada esta ampliación el sistema ya es capaz de interpretar y corregir soluciones escritas en diferentes lenguajes de programación.

APLICACIÓN DE ACME EN ASIGNATURAS DE PROGRAMACIÓN.

Una vez finalizada la ampliación de la plataforma ACME, además de en varias asignaturas de matemáticas y química, se ha utilizado durante este curso 2004 en un grupo piloto de introducción a la programación de 118 alumnos.

El sistema ha sido usado como complemento de las sesiones de teoría, utilizándolo como sistema de evaluación continua. A cada alumno se le ha asignado un cuaderno con nueve problemas que debían resolverse usando la notación en pseudo-código.

En las sesiones de laboratorio se ha usado como software base de las clases prácticas. Todos los problemas y los ejemplos presentados por el profesor se realizaban a través de la plataforma ACME. Durante la sesión se asignaban una serie de problemas a cada alumno, todos diferentes. Estos problemas debían resolverse usando la plataforma, de esta forma quedaba constancia del trabajo realizado por el alumno y podíamos ver los errores que cometía, el número de soluciones enviadas antes de llegar a la solución correcta, el tiempo que tarda en resolver el problema,...

La valoración que hacemos de ACME es muy positiva, puesto que no solamente nos facilita la tarea sino que ha motivado de forma considerable el interés del alumno por la programación. En estos momentos creemos que estamos explotando al máximo la potencia del sistema puesto que nos ofrece:

- ***Un entorno de auto-aprendizaje.*** El alumno se conecta al sistema ACME vía Internet. Una vez identificado y ya dentro del sistema puede seleccionar problemas, previamente introducidos por el profesor, y resolverlos. El alumno puede optar por obtener la solución del problema sin ayuda o con la ayuda del sistema. En este último caso el sistema indica al alumno las pautas a seguir para llegar a la solución. Esta solución puede escribirse en un lenguaje de programación o en pseudo-código. Cuando el alumno da la solución por buena la envía en un fichero de texto y el sistema la corrige siguiendo el proceso descrito anteriormente.

- ***Refuerza las clases de teoría.*** Uno de los principales problemas de la enseñanza de la programación es el poco tiempo que tenemos en las sesiones de teoría para poder presentar ejemplos. Seleccionando y proponiendo problemas similares a los ejemplos presentados en las sesiones de teoría el alumno tiene la posibilidad de resolverlos usando las facilidades del sistema, es decir las pautas de ayuda que llevan asociados. El trabajo realizado a través de estos problemas ayudará al alumno a reforzar los conceptos introducidos por el profesor.
- ***Proporciona un sistema de evaluación continua.*** El sistema ofrece la posibilidad de asignar a cada alumno cuadernos personalizados y únicos de problemas. Estos cuadernos se crean mediante un proceso que sortea automáticamente los problemas seleccionados por el profesor. El alumno enviará las soluciones antes de una fecha prefijada y el sistema guarda todas las soluciones enviadas. El sistema, aplicando los criterios definidos por el profesor, a partir de las soluciones enviadas y del número de intentos realizados hasta llegar a la solución correcta, determinará la calificación que le corresponde al alumno, calificación siempre modificable por el profesor.
- ***Facilita la atención personalizada.*** Al mantener el sistema las soluciones enviadas por el alumno, le permite al profesor detectar sus deficiencias y controlar la evolución que sigue en su proceso de aprendizaje. El sistema puede considerarse como un profesor virtual las 24 horas del día.
- ***Desvincula el trabajo del alumno de los laboratorios y de los problemas de instalación de software.*** Trabajar con la plataforma ACME únicamente requiere de una conexión a Internet. El alumno puede trabajar desde los laboratorios de la Universidad o desde su casa, sin necesidad de instalarse ningún compilador o entorno de programación.
- ***Fácil de usar.*** Las interficies de usuario del sistema son lo suficientemente simples como para poderlas usar sin necesidad de tener conocimientos de informática.

CONCLUSIONS

Hemos presentado la ampliación realizada sobre la plataforma ACME desarrollada en el Departamento de Informática y Matemática Aplicada de la Universidad de Girona. Dicha ampliación se usa como entorno de soporte a la docencia de las asignaturas de introducción a la programación. Se ha usado en un grupo experimental y los resultados obtenidos son muy satisfactorios, tanto desde el punto de vista de los alumnos como desde el punto de vista del profesor.

Como trabajo futuro nos planteamos la ampliación de la plataforma para adaptarla a diferentes estudios y a más materias.

Agradecimientos

El proyecto ACME se ha podido desarrollar gracias a la ayuda de la Universidad de Girona, Escuela Politécnica Superior y la ayuda del DURSI MQD2002.

Bibliografía

- [1] J. Soler, J. Poch, E. Barrabés, D. Juher, J. Ripoll. A tool for the continuous assessment and improvement of the student's skills on a mathematics course. Proceedings of the International Symposium. on Technologies of Information and Communication in education for Engineers and Industry. TICE 2002 pp. 105-110, Lyon, 2002
- [2] I. Boada, J. Soler, F. Prados, J. Poch. A teaching/learning support tool for introductory programming courses. IEEE Proceedings of the 5th International Conference on Information Technology Based Higher Education and Training. ITHET 2004 (aceptado).
- [3] J. Poch Garcia; E. Barrabés Vera; D. Juher Barrot; J. Ripoll Misse; J. Soler Villanueva; A. Calsina Ballesta. ACME 2.0 un sistema de evaluación continuada y ayuda a la resolución de problemas. Actas XI Congreso Universitario de Innovación Educativa en las Enseñanzas Técnicas. Vilanova y la Geltrú 23, 24 y 25 de Juliol de 2003.