

# Gestión Automática de entrega de Prácticas vía web

Juan Carlos Rodríguez del Pino  
Dept. de Informática y Sistemas  
Universidad de Las Palmas de Gran Canaria  
35017 Las Palmas de Gran Canaria  
[jcrodriguez@dis.ulpgc.es](mailto:jcrodriguez@dis.ulpgc.es)

## Resumen

Este artículo presenta un programa llamado GAP, Gestión Automática de entrega de Prácticas vía web, el cual es capaz de almacenar las prácticas de los alumnos y que a la vez permite a los profesores el acceso a éstas para su revisión y evaluación. GAP interactúa tanto con los alumnos como con los profesores. Para tener total acceso a las características del programa es suficiente con disponer de un acceso a internet y un navegador web. También es posible usar el programa GAP para detectar plagios en un gran conjunto de ficheros fuente.

## 1. Introducción

Desde el punto de vista del alumno, el programa GAP es un sistema encargado de recibir y almacenar sus prácticas, además de informarle de la valoración personal realizada por parte del profesor de su trabajo. Desde el punto de vista del profesor el sistema se encarga de mostrárselas y almacenar su evaluación.

Entre otras características el programa mantiene un registro por alumno de asistencia a sesiones de prácticas, calcula la nota final para cada alumno y dispone de un sistema anticopia de prácticas.

Este artículo tiene como contexto y ámbito la experiencia del uso del programa GAP como apoyo en las prácticas de las asignaturas de Metodología de la Programación I y II, Estructuras de Datos I y II y Lenguajes de Programación en las titulaciones de Ingeniería Informática e Ingenierías Técnicas en Informática de Gestión y Sistemas en la Universidad de Las Palmas de Gran Canaria.

## 1.1 La era preGAP

No están muy lejos los tiempos en que la documentación de las prácticas a realizar por parte de los alumnos se distribuía en papel. Como sistema de revisión de prácticas, al final del curso, se examinaban y calificaban los trabajos, debiendo el profesor recibirlos en papel o en disquete. La llegada del World Wide Web como medio ideal y universal para la transmisión de todo tipo de información, hizo que se produjera una mejora en el suministro de información sobre las prácticas, pasando del papel a las páginas Web.

## 1.2 Los problemas

El primer paso en las mejoras con la llegada de internet fue poder recibir las prácticas vía correo electrónico. En el primer cuatrimestre del curso 1999/2000 se cambia el medio de recepción de prácticas: se pone a disposición de los alumnos una cuenta de correo electrónico a la que debían enviar sus trabajos. Este sistema, aunque en principio parecía mejorar la situación ya que no se requerían las prácticas en papel, tenía otros inconvenientes: Los alumnos enviaban sus trabajos desde cuentas de correo que no eran las suyas, lo que dificultaba su identificación; no enviaban en el mensaje la información de referencia sobre la práctica entregada; consultaban frecuentemente sobre si su práctica se había recibido correctamente..., y a esto se unían, además, todos los problemas propios de este medio inseguro de comunicación.

## 1.2 La búsqueda de la solución

Estos problemas plantearon la necesidad de buscar un sistema mejor para la recepción de prácticas. Los requisitos a cumplir eran que fuese gratuito, sencillo y flexible. Aunque se han venido usando soluciones

más o menos adaptadas al problema, en general orientadas a entornos UNIX[7] y al uso de correo electrónico[2, 8], comportaban poca flexibilidad de uso y una interfaz poco amigable. También se disponía de herramientas de docencia vía web, como Webct [10], y experiencia sobre ella [4], que aunque muy interesantes, no encajaban en la solución buscada. Por ello se planteó desarrollar un programa accesible vía Web que se encargase de automatizar la gestión del almacenamiento de prácticas.

Tras un par de meses de trabajo, se sacó a la luz la primera versión, que aunque no tenía toda la funcionalidad que se describe en este trabajo, ya realizaba suficientemente la tarea principal para la que fue concebido. El programa se ejecuta en una máquina con S.O. Linux en colaboración con el servidor web Apache. Está escrito en C++ y actualmente tiene más de 10.000 líneas de código.

## 2. Características

El núcleo del programa es un almacén de prácticas de alumnos. Alrededor de este elemento se han añadido características que han tomado gran relevancia como son: gestión de usuarios, calificación de las prácticas por los profesores, el sistema de comparación de prácticas para detectar su plagio o el control de asistencia a sesiones de prácticas.

### 2.1 Usuarios

El programa dispone de un esquema de autenticación, mediante nombre de cuenta de usuario y clave, integrado con el servidor Web. Las cuentas de usuario son de distinto tipo, lo que define el comportamiento del programa hacia ese usuario. Los tipos de usuarios son: "invitado", "alumno", "profesor", "profesor corrector" y "profesor administrador". Los usuarios "invitados" representan personas con derecho a navegar por las páginas restringidas en el servidor. El tipo de usuario "alumno" puede entregar prácticas, ver aquellas entregadas previamente y su calificación. Los "profesores" pueden acceder a la información de los alumnos almacenada en el sistema, ver sus datos personales y sus prácticas. Los "profesores correctores", además, pueden calificar las prácticas. Los "profesores administradores" pueden realizar

todo lo anterior, más las operaciones necesarias para la gestión del sistema, como pueden ser: definir prácticas, criterios de calificación global, crear cuentas, etc.

### 2.2 Prácticas

Un administrador puede crear, modificar o borrar definiciones de prácticas. Una definición de una práctica establece un título, una página Web opcional donde se describe, unas fechas de entrega

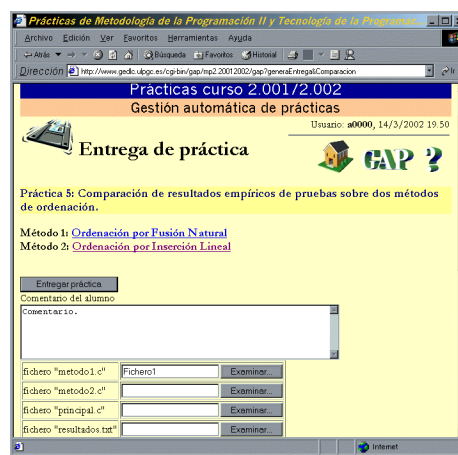


Figura 1

y los ficheros a entregar. Las fechas establecen el período de entrega de prácticas que se puede dividir en dos, opcionalmente. Las prácticas entregadas en el segundo período tendrían una penalización automática en su valoración, por retraso en la entrega. La configuración de los ficheros a entregar permite establecer qué ficheros y qué restricciones deben cumplir todos o cada uno de ellos. Las restricciones establecen el número mínimo de líneas, el número máximo y el número máximo de caracteres. Si la entrega de la práctica por parte del alumno no satisface los requisitos establecidos, el sistema rechaza la entrega. Otra característica añadida en las últimas versiones del programa es la posibilidad de que la práctica a realizar por cada alumno sea seleccionada aleatoriamente de entre un grupo prefijado. Estos tipos de prácticas tienen múltiples ventajas como se muestra en otros trabajos [1].

### 2.3 Revisión y visualización

La recepción, almacenamiento y visualización de prácticas es el punto fuerte del sistema. Para entregar una práctica el alumno accede a la opción del GAP correspondiente que muestra a éste un formulario a rellenar. En este formulario el alumno puede introducir los ficheros solicitados además de un comentario adicional para el profesor corrector. Ver figura 1.

Una vez entregada la práctica, los alumnos pueden comprobar la información almacenada en el sistema y verificar cómo éste se la presentará al profesor corrector, pudiendo alterarla si no le parece correcta y aun no ha sido evaluada. Si la práctica ha sido calificada por un profesor el sistema mostrará al alumno los datos de la revisión: fecha y hora de revisión, el profesor que la ha hecho, los comentarios de éste sobre la práctica y su valoración. El sistema muestra las prácticas en forma de página Web que contiene el comentario del alumno, una lista de ficheros entregados con enlaces para poder descargarlos, si se desea, y el contenido de los ficheros. Los tipos de ficheros que puede mostrar directamente son códigos fuente de programas, ficheros texto y gráficos. Los ficheros con código fuente en Java, C, C++, Ada 95, Scheme y Prolog se presentan con resalte sintáctico, mostrando en distinto color las palabras clave, los comentarios, las ristas literales, etc (ver figura 2) Si el usuario al que se muestra la práctica es un

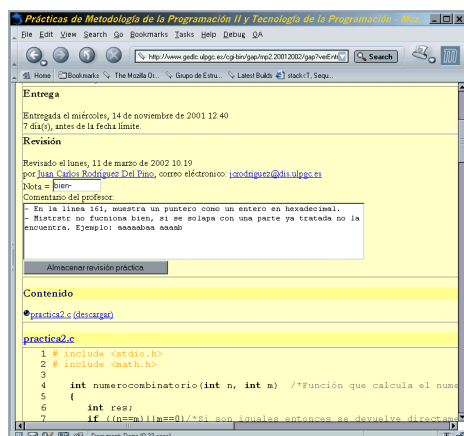


Figura 2

profesor corrector o administrador el sistema le ofrecerá la posibilidad de evaluarla.

Los profesores pueden solicitar al sistema un listado de los alumnos junto con la información de si han entregado o no cada práctica y su valoración. Desde esta página es posible ver y revisar cualquier práctica de cualquier alumno.

### 2.4 Control de plagio

Al poco tiempo de poner el programa en marcha y comenzar a recibir entregas de prácticas, se descubrió que los alumnos con el nuevo sistema tenían más facilidad para copiarse. Para comprobar esta posibilidad se escribió código que verificase si varios alumnos habían entregado el mismo fichero. La sorpresa surgió al comprobar que el número de coincidencias era mayor de las esperadas, lo que motivó la adición de código que detectase el plagio de trabajos.

Los primeros sistemas de medida de similitud entre ficheros con código fuente se basaban en métricas puramente textuales que eran fácilmente vulnerables. Partiendo de un fichero original, un alumno puede, sin mucha dificultad, quitar, añadir o modificar comentarios, modificar ristas literales, cambiar nombres de variables, procedimientos o funciones, alterar el formato de línea añadiendo o eliminando espacios, o retornos de carro. También puede reordenar la ubicación de funciones o procedimientos o incluso grupos de instrucciones. Por ello se han desarrollado sistemas que han tenido en cuenta estos aspectos. Los más avanzados como YAP [11], Sim [3] y otros [6], usan distintas métricas que se basan en filtrar el código original, mediante un análisis sintáctico, para después compararlo usando generalmente una adaptación de técnicas conocidas en el ámbito textual. El filtrado del código tiene el fin de limitar al máximo el efecto en la comparación de las posibles alteraciones que se puedan haber introducido. El sistema de detección de copias empleado en GAP usa básicamente este método. Para cada lenguaje de programación se tiene un pequeño analizador sintáctico que permite eliminar las partes del código que pueden ser alterables fácilmente. Esta "firma" del programa es la que se compara con las obtenidas de los otros ficheros. GAP emplea como métrica una variante de DIT [9]. DIT es una distancia entre ristas que es invariante a la posición donde aparecen los

caracteres. Esto permite que GAP no se vea afectado por la reordenación del código, como ocurre en otros sistemas mencionados.

Actualmente el programa está preparado para comparar ficheros con código fuente en Java, C, C++, Ada 95, Scheme y Prolog. El programa ofrece a los profesores dos posibilidades de comprobar las similitudes entre los ficheros. Por un lado permite generar un informe sobre la similitud de los ficheros de una práctica comparándolos todos con todos. Por otro lado posibilita comparar todos los ficheros entregados por un alumno concreto, con los entregados por el resto de sus compañeros. Esta segunda alternativa muestra si un alumno se copia por sistema, o si tiene muchas similitudes siempre con el mismo compañero. El informe del sistema da una relación de ficheros cuya comparativa muestra un grado de similitud por encima de un umbral solicitado. El sistema no es infalible y se pueden dar casos de falsas similitudes, aunque la experiencia, incluso con gran número de prácticas, demuestra que estos falsos plagios son infrecuentes. En cualquier caso, es responsabilidad del profesor valorar el posible plagio encontrado ya que el sistema sólo da índices de similitud entre ficheros.

El empleo de este sistema ha dado resultados sorprendentes y todos los profesores que lo han empleado están muy satisfechos con él. En algunas asignaturas de primer curso y cuando los alumnos no sabían que se disponía de este elemento, los niveles de copias confirmadas alcanzaban el 15% de las prácticas entregadas. Seguramente aún escapan algunos plagios puesto que el sistema no puede detectar fraudes en el que el alumno que se copia conoce suficientemente el lenguaje y entiende la práctica para transformarla por encima del umbral establecido, o porque otro alumno le ha hecho la práctica. El primer caso no plantea problemas graves ya que el alumno demuestra tener suficientes conocimientos para superar la práctica. El segundo se puede neutralizar realizando exámenes prácticos.

### 2.5 Asistencia y accesos

GAP mantiene un registro personal de los accesos de cada alumno a las distintas partes del programa y de las páginas Web que se desee controlar. El profesor puede ver este registro de cada alumno que incluye desde qué máquina, a qué hora y qué día realizó el acceso.

Foto	Cur/Grp	Nombre	Accesos	EstructuraBasica	Funciones	NoHomogeneos	Ristraal	Ficheros	Com
	II-GO	2003	17 Paginas++	5 Bases	10 Bases++	11 Bases++			
	ITIC-2	2005	17 # Bases	17 Paginas++	5 Bases				
	ITIC-2	2005	16 # Bases	17 Paginas	10 Paginas++				

Figura 3

Antes de usar el programa GAP, uno de los medios empleados para que los alumnos no dejaran la realización de los ejercicios para el último día era valorar la asistencia a las sesiones de prácticas. El profesor en cada sesión de prácticas tenía que pasar lista para llevar una contabilidad de asistencias. Con el GAP el control de asistencia se realiza de la siguiente manera: el profesor establece cuándo pasar lista e informa a los alumnos de que seleccionen la opción de contabilizar asistencia a la sesión. Esta información permite saber en todo momento a qué sesiones ha asistido cada alumno y qué alumnos asistieron a cada sesión. Ver figura 3.

### 3. Experiencia y resultados

Desde el punto de vista de los profesores los resultados han sido excelentes, todos coinciden en que una vez que han usado el GAP no se plantean volver a los antiguos métodos de entrega. El GAP ha hecho desaparecer muchos de los problemas que tenían los otros sistemas, siendo lo suficientemente flexible para no limitar el tipo de prácticas que pueden entregar los alumnos. Otro aspecto que ha cambiado es la forma en que se ven los plazos de entrega de prácticas. Muchos profesores sugieren que el plazo debe reducirse en lo posible, con el fin de que los alumnos trabajen regularmente y lleven las asignaturas al día. Las reuniones de coordinación se centran ahora en los criterios de corrección o en cómo plantear las prácticas de forma que su evaluación sea lo más simple y justa posible.

Para los alumnos el cambio ha supuesto unos plazos de entrega de prácticas más estrictos, pero

como mejora disponen de una más rápida y mejor realimentación. Reciben antes la calificación de sus prácticas, acompañadas de la descripción de sus errores.

Un cambio que ha afectado a ambos colectivos es la mejora de acceso a las prácticas. Los profesores pueden revisar prácticas desde su despacho, su casa o desde el extranjero. Los alumnos pueden realizar y entregar sus prácticas, ver sus calificaciones desde, el laboratorio, sus casas o desde cualquier lugar con acceso a internet.

#### 4. Perspectivas de futuro

Con respecto a la gestión de prácticas se puede trabajar hacia una automatización de su evaluación [5, 8]. También existen perspectivas más factibles ya que partiendo del esquema de autenticación disponible, se pueden añadir capacidades relacionadas con el resto de los elementos de la docencia como pueden ser: la gestión de eventos y noticias, suministro de documentación y software, generación y realización de encuestas y test, etc. Aunque muchas de estas características están resueltas mediante páginas Web, su integración en el GAP mejoraría su manejo.

#### Referencias

- [1] Bridgeman, S. SAIL: A System for Generating, Archiving, and Retrieving Specialized Assignments Using LATEX. *The Proceedings of the Thirty-first SIGCSE Technical Symposium on Computer Science Education* SIGCSE 2000. pp.300-304.
- [2] Dawson-Howe, K. Automatic Submission and Administration of Programming Assignments. *SIGCSE Bulletin*, Vol. 28 No. 2 June 1996. pp. 40-42.
- [3] Gitcheli, D.; Tran N. Sim: A Utility For Detecting Similarity in Computer Programs. *The Proceedings of the Thirtieth SIGCSE. SIGCSE'99*. New Orleans, Louisiana.
- [4] Goldberg, Murray W. WebCT and first year: student reaction to and use of web-base resource in first year Computer Science. *SIGCSE Bulletin*. Vol. 29, No. 3, (September 1997), pp. 127-129.
- [5] Jackson, D. A Semi-Automated Approach to Online Assessment. *Proceedings of the 5<sup>th</sup> Annual SIGCSE/SIGCUE Conference on Innovation and Technology in Computer Science Education*. July 2000. Helsinki, Finland
- [6] Leach R. Using Metrics to Evaluate Student Programs. *SIGCSE Bulletin*. Vol. 27 No. 2, June 1995. pp. 41-43,48
- [7] Macpherson, P. A.. A Technique for Student Program Submission on UNIX System. *SIGCSE Bulletin*. Vol. 29, No. 4, (December 1997), pp. 54-56.
- [8] Reek K.A. A software Infraestructure to Support Introductory Computer Science Course. *SIGCSE Bulletin*. Vol. 28 No. 1 March 1996. pp. 125-129. pp. 130-134.
- [9] Santana, O.; Dfaz, M. Esquemas y Estructura para la Búsqueda de las Palabras Más Similares a una dada. *Actas de la XIII Conferencia Latinoamericana de Informática*. Bogotá, Colombia. Noviembre, 1987. Vol. II. pp. 1169-1189.
- [10] WebCT. home page <http://www.webct.com>.
- [11] Wise M.J. YAP3: Improved Detection of Similarities in Computer Program and Other Texts. *SIGCSE Bulletin*. Vol. 28 No. 1 March 1996.