

# Experiencia sobre el diseño y evaluación de una herramienta educativa para POO

Isidoro Hernán Losada

Universidad Rey Juan Carlos  
c/ Tulipán s/n  
28933 Móstoles, Madrid, España  
[isidoro.hernan@urjc.es](mailto:isidoro.hernan@urjc.es)

**Abstract.** Este artículo presenta una aplicación educativa para enseñar el concepto de herencia en Java. Al ser una herramienta pensada y diseñada para el aprendizaje, contiene un fundamento pedagógico sustentado por la taxonomía de Bloom. Las guías para usar esta jerarquía se pueden encontrar en [8]. Se muestra con un cierto grado de detalle el proceso de análisis y diseño, así como la evaluación llevada a cabo en un curso universitario de Programación Orientada a Objetos.

## 1 Introducción

En la mayoría de las asignaturas cursadas en cualquier carrera científico-técnica, los conceptos teóricos son fijados y afianzados mediante la realización de prácticas relacionadas con dichos temas. El software educativo se utiliza actualmente en muchas materias de distintas ciencias e ingenierías [1,2,9]. Nuestro campo de interés se centra en la enseñanza de la programación, en concreto en el paradigma de la orientación a objetos.

El alumno es el factor más importante del proceso de enseñanza-aprendizaje. Las herramientas que hemos diseñado lo tienen en cuenta, intentado abarcar todos los estilos de aprendizaje [7]. La ayuda al estudiante se produce mediante la exposición del concepto, su ejemplificación y la realización de tests o problemas agrupados en distintos grados de dificultad, clasificados mediante el nivel de Bloom que el alumno debe tener para resolverlos con éxito. Esta última característica, la resolución de problemas o tests, es una técnica que mejora la retención de los conceptos a largo tiempo [6] y hace que el alumno practique el aprendizaje activo lo que conlleva indudables beneficios [15] para el discente. Estos métodos de incentivación mediante resolución de problemas han sido evaluados con éxito en el campo de la programación [13].

Siguiendo con la línea marcada en [8] se ha desarrollado una herramienta que une ambos conceptos. Por un lado, se presenta al alumno la base o fundamento de la herencia en programación orientada a objetos, intentado abarcar todos los estilos de aprendizaje [7]. Como refuerzo teórico-práctico se permite al alumno interaccionar con la aplicación mediante unas demostraciones animadas. Por otro lado, la

herramienta genera problemas, en forma de test multirespuesta, para que el alumno practique sobre el concepto enseñado, que son corregidos de manera automática y provee realimentación si el usuario lo solicita. Estos test permiten medir el nivel de aprendizaje alcanzado medido según la taxonomía de Bloom.

La estructura del artículo es como sigue: la segunda sección describe brevemente la taxonomía de Bloom y los diferentes estilos de aprendizaje. La tercera sección describe el análisis y diseño de la herramienta desarrollada para la enseñanza de la programación. La cuarta sección muestra la evaluación. La quinta sección discute brevemente los resultados obtenidos del uso realizado por los alumnos de la asignatura Programación Orientada a Objetos perteneciente al plan de estudios de la Universidad Rey Juan Carlos del 2º curso de Ingeniería Informática. Finalmente, damos nuestras conclusiones y trabajo futuro.

## 2 Taxonomía de Bloom y estilos de aprendizaje de Felder

La educación debe estructurarse en torno a cuatro aprendizajes fundamentales que serán los pilares del conocimiento de cada individuo. El primero es aprender a conocer (dominio cognitivo), el segundo, aprender a hacer (dominio psicomotor), el tercero es aprender a vivir juntos (dominio relacional/social) y el cuarto es aprender a ser (dominio actitudinal).

Si nos centramos en la programación orientada a objetos, el factor más importante para nosotros es el cognitivo. El trabajo realizado por un grupo de educadores en este dominio se terminó en 1956 y se conoce comúnmente como taxonomía de Bloom [3]. La taxonomía establece una jerarquía de seis niveles con grado creciente de aprendizaje del alumno. Cada nivel presupone la capacitación del alumno en los niveles precedentes. Según ascendemos por la jerarquía nos encontramos un mayor grado de aprendizaje:

- **Nivel 1 o nivel de conocimiento.** El estudiante puede reconocer o recordar la información sin ser necesario cualquier clase de entendimiento o razonamiento sobre su contenido. Este nivel se divide en:
  - Conocimiento de lo específico (terminología y hechos).
  - Conocimiento de los modos y maneras de tratar lo específico.
  - Conocimiento de leyes universales y abstracciones de un campo concreto.
- **Nivel 2 o nivel de comprensión.** El estudiante puede entender y explicar el significado de la información recibida. A su vez se parte en:
  - Traducción.
  - Interpretación.
  - Extrapolación.
- **Nivel 3 o nivel de aplicación.** El estudiante puede seleccionar y utilizar datos y métodos para solucionar una tarea o un problema dado.
- **Nivel 4 o nivel de análisis.** El estudiante puede distinguir, clasificar y relacionar hipótesis y las evidencias de la información dada, así como descomponer un problema en sus partes. Se divide en:
  - Análisis de los elementos.

- Análisis de las relaciones.
- Análisis de los principios de la organización.
- **Nivel 5 o nivel de síntesis.** El estudiante puede generalizar ideas y aplicarlas para solucionar un nuevo problema. Se puede crear una nueva solución, manejando diferentes aspectos de conocimiento y métodos, de tal manera que el resultado sea algo más que la suma de los componentes. Se subdivide en:
  - Producción de una nueva solución.
  - Realización de un plan o un conjunto de operaciones.
  - Derivación de un conjunto de relaciones abstractas.
- **Nivel 6 o nivel de evaluación.** El estudiante puede comparar, criticar y evaluar métodos o soluciones para solucionar un problema o para elegir el mejor. A su vez se divide en:
  - Evaluación en términos de evidencias internas.
  - Juicio en términos de criterios externos.

Krathwohl revisó la taxonomía para refinarla y dividirla en dos dimensiones [12]: La dimensión del conocimiento, basada en la materia que se pretende enseñar y la dimensión del proceso cognitivo, fundamentada en el verbo o acción que se quiere conseguir enseñar. Así en la primera dimensión quedan cuatro niveles principales y no jerarquizados, denominados: conocimiento de hechos, conocimiento de conceptos, conocimiento de procedimientos y conocimiento metacognitivo. La segunda dimensión, que mide el proceso cognitivo, se compone de seis niveles, en este caso jerarquizados, y son: recordar, comprender, aplicar, analizar, evaluar y crear. Como se puede observar es muy similar a la taxonomía original de Bloom en la que el primer nivel se pasa a denominar recordar, y los dos últimos niveles se cambian de orden, pasando a ser el quinto evaluar y el sexto y último crear.

En todo caso, es un marco comúnmente aceptado. Nosotros lo adoptamos para evaluar el conocimiento adquirido por el alumno.

Para el diseño de la herramienta, en cuanto a la presentación de contenidos, nos basamos en los estilos de aprendizaje de Felder [7]. Propone un modelo donde se clasifica a los alumnos en varias categorías, en principio excluyentes por parejas pero no por categorías:

- Estudiantes activos, que aprenden trabajando con otros y por experimentación vs. estudiantes reflexivos que aprenden razonando y trabajando solos.
- Estudiantes que aprenden razonando que se guían por lo concreto, lo práctico, se basan en hechos y métodos vs. aprendices intuitivos, que son conceptuales, innovadores y se basan en teorías y en intenciones.
- Aprendices visuales que prefieren los diagramas, los gráficos y los diagramas de flujo vs. aprendices verbales que prefieren explicaciones escritas o habladas.
- Aprendices secuenciales que aprenden de manera incremental, en pasos ordenados vs. aprendices globales que son holísticos y aprenden a grandes saltos.
- Aprendices inductivos que prefieren explicaciones que van de lo específico a lo general vs. deductivos que prefieren explicaciones que van de lo general a lo específico. Posteriormente, el autor decidió suprimir esta categoría.

#### 4 Isidoro Hernán Losada

Felder hace hincapié en que el estilo de aprendizaje define las preferencias pero no la capacidad o incapacidad para una determinada materia, disciplina o profesión. El propósito es identificar como se puede explicar o exponer para que al alumno le sea más fácil.

En [17] realizan un experimento en un curso de introducción a la programación con Java. Para conocer el estilo de aprendizaje preferido por cada alumno se realiza un test y luego se relaciona los estilos con las notas de clase y del examen. Observan que los aprendices reflexivos en el examen obtienen mejor nota que los activos. Ocurre lo mismo con los verbales (mejor nota) y los visuales. Luego hacen estudios por grupos de alumnos. Entre las conclusiones obtenidas se destaca que los aprendices activos, razonables y visuales están en desventaja con el resto.

### 3 Análisis, diseño e implementación de la herramienta

La aplicación se desarrollo partiendo de unos requisitos previos, que fueron redactados para conseguir una herramienta útil al alumno en su periodo de aprendizaje del concepto de herencia dentro de un curso de programación orientada a objetos. Estos requisitos siguieron las directrices dadas en [8]. Se describen a continuación con cierto grado de detalle cada una de las fases de desarrollo.

#### 3.1 Análisis

Se pretende realizar una herramienta con las siguientes características:

- La presentación de contenidos ha de contemplar la mayoría de estilos de aprendizaje [7] de los alumnos.
- Debe proporcionar un entorno intuitivo y fácil de usar, para que su desconocimiento no interfiera en la enseñanza.
- Proporcionar interactividad con el discente, para que el rol del alumno no sea meramente el de un observador.
- Ha de reforzar el aprendizaje mediante la realización de ejercicios prácticos que se generen y corrijan automáticamente.
- La herramienta debe proporcionar una realimentación al alumno ya que la auto evaluación es una buena técnica de aprendizaje [11].
- La aplicación tiene que llevar las estadísticas tanto de aciertos como de errores que el alumno comete en cada uno de los niveles.
- Ha de centrarse en un concepto concreto de la programación orientada a objetos. En este caso se ha elegido la herencia como materia de estudio.
- Debe cubrir los tres primeros niveles de la jerarquía de Bloom. Para ello debe proveer una parte teórica para los dos primeros niveles (conocimiento y comprensión) y una parte práctica para reforzar los tres niveles inferiores.

### 3.2 Diseño e implementación

Vistos las necesidades, se decidió dividir la herramienta en tres componentes complementarios: teórico, práctico y de demostración. Cada uno pretende cubrir uno o más niveles de la taxonomía de Bloom. La arquitectura de la aplicación es muy simple. Consiste en un árbol de directorios donde se almacenan los distintos componentes. Se eligió este formato debido a su simplicidad para instalarlo y para modificarlo o adaptarlo a otra materia objeto de estudio. La distribución puede hacerse a través de Internet como un archivo comprimido. Para instalarlo basta con descomprimirlo en el disco local.

El lenguaje de desarrollo elegido ha sido Java por sus propiedades: es un lenguaje no propietario, independiente de la plataforma y existen muchos entornos de programación gratis. Los componentes se desarrollan en forma de applets que pueden ejecutarse en cualquier navegador.

**Componente de teoría.** Esta componente se aloja en una carpeta denominada teoría, y consiste en un archivo html, con sus correspondientes imágenes y gráficos como se muestra en la Fig. 1. Este archivo contiene la explicación teórica del concepto en estudio. Debe ser escrito y diseñado siguiendo los siguientes criterios:

- Claridad en la exposición. Se debe tener en cuenta que el alumno se enfrentará al texto de manera individual, sin apoyo del profesor.
- Autocontenido. El concepto puede ser aprendido sin tener que consultar ningún otro medio. Advertimos que el objetivo de esta aplicación no es este, sino servir de apoyo a la enseñanza.
- Gráfico. Las ilustraciones son fundamentales para ayudar a comprender el concepto en estudio.

La teoría presentada tanto de manera textual como gráfica ayuda a los alumnos reflexivos, que estudian solos, y a los visuales, pues se les presenta la información también de manera gráfica. En este caso particular, la materia es la herencia en POO, concretada en el lenguaje Java (ver Fig. 1), pero podría ser cualquier otro concepto de cualquier materia. Este componente trata de cubrir los niveles de conocimiento y de comprensión de la taxonomía de Bloom.

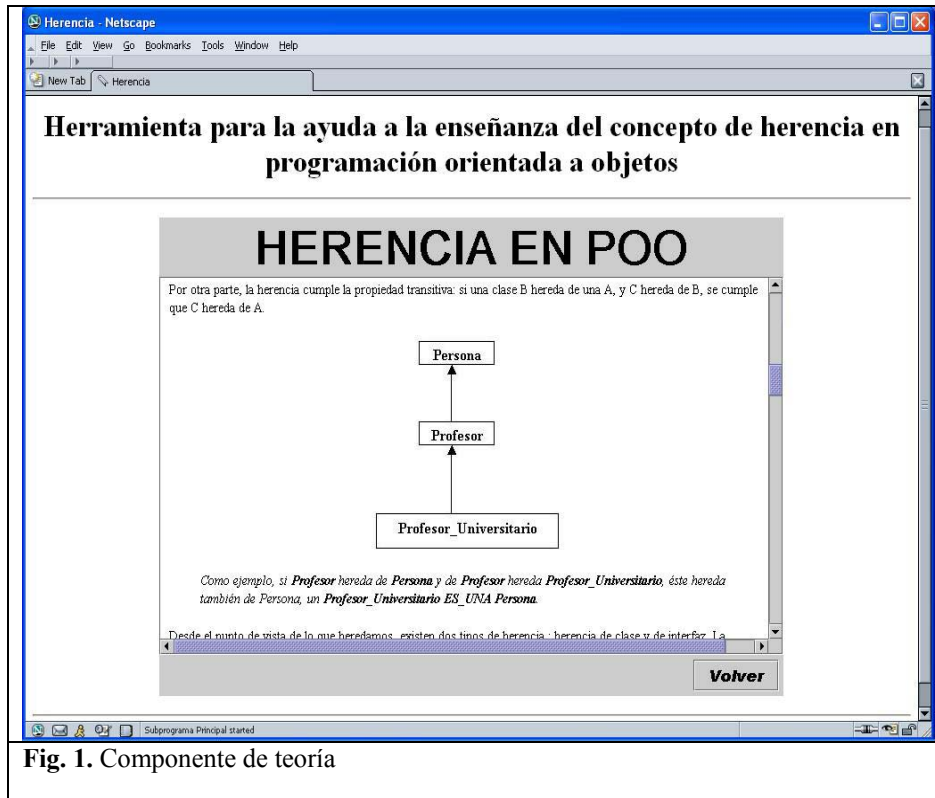


Fig. 1. Componente de teoría

**Componente de demostración.** Esta componente se aloja en una carpeta denominada demos, y cada demostración se guarda en una subcarpeta llamada demo<sub>i</sub>, donde i es el número de orden. Son animaciones realizadas en Flash y preparadas para poder visualizarse en navegadores con el plugin correspondiente. En ellas, el alumno puede interactuar con los ejemplos, avanzando o retrocediendo al ritmo que considere oportuno. Con las demostraciones se pretende ayudar a los alumnos activos, que necesitan experimentar para comprender el concepto a enseñar. La imposibilidad de interactuar con la aplicación suele ser un handicap para la enseñanza a distancia, ya que la mayoría de las plataformas de enseñanza lo único que permiten es navegar por los contenidos, pero no controlar ejemplos. La visualización tiene un gran potencial educativo, sobre todo para alumnos con un estilo de aprendizaje visual [17], pero no siempre se aprovecha [10]. La eficacia educativa de las visualizaciones aumenta con la implicación del alumno [16].

Estas demostraciones contienen ejemplos y contraejemplos de la implementación de la herencia en Java. La aplicación permite añadir más ficheros Flash, simplemente añadiendo subcarpetas con los archivos correspondientes. La herramienta reconoce automáticamente su existencia y muestra los respectivos botones para su ejecución.

La Fig. 2 muestra una de las demostraciones actuales de la aplicación. Se puede observar los botones de avance y retroceso, con lo que el usuario puede manejar y repetir tantas veces desee el ejemplo gráfico. Las clases se representan con rectángulos que encierran en la parte de arriba el nombre, a continuación los atributos y por último, los métodos. Las elipses representan los objetos. En la parte inferior van surgiendo las instrucciones y está resaltada en negrita la que se encuentra en ejecución. En la parte superior derecha, el objeto que recibe el método se marca con un trazo perimetral grueso y simultáneamente surge la explicación del mensaje que recibe el objeto en cuestión. Existen contraejemplos con instrucciones erróneas. El sistema explica el motivo del error y la tacha para resaltar el hecho.

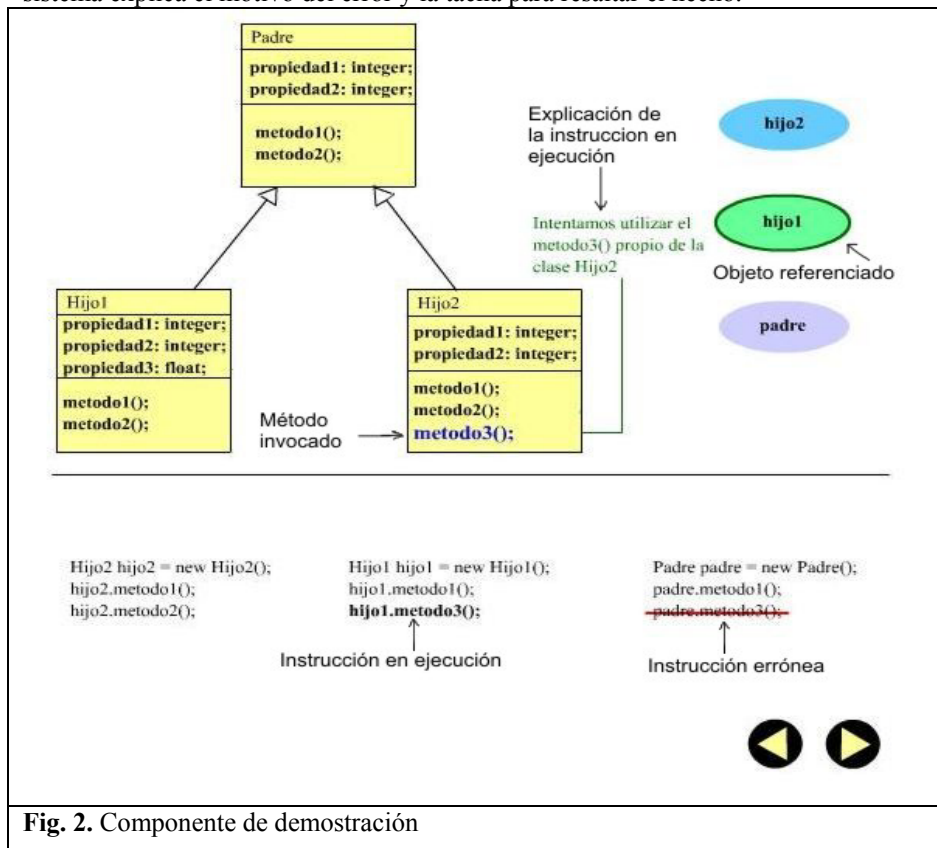


Fig. 2. Componente de demostración

**Componente de test.** Es la encargada de gestionar, visualizar y corregir todas las preguntas de las que consta la aplicación, además de llevar las estadísticas de aciertos y fallos del alumno por cada nivel. Existen tres baterías de preguntas, agrupadas en los tres niveles inferiores de conocimiento en concordancia con la taxonomía de Bloom [3]. Cada conjunto de preguntas se almacena en una carpeta. Cada pregunta posee su enunciado, las opciones y una explicación sobre las posibles respuestas para proveer al alumno, cuando este lo solicite, realimentación sobre la corrección del problema.

La primera vez que se accede a esta parte, la aplicación pregunta al alumno el nivel donde desea comenzar. Una vez elegido, las preguntas se presentan de manera aleatoria y de manera que se asegura la no repetición en la misma sesión. Para incrementar el refuerzo de la autoevaluación [11] cuando el alumno logra acertar el 80% de las preguntas efectuadas sube de nivel. Esta característica le da a la aplicación un cierto grado de adaptación al alumno. El porcentaje para incrementar el nivel se empieza a calcular a partir de la quinta pregunta, para que no se de el caso de que respondiendo correctamente una única cuestión (100% de aciertos) se cambie de nivel. En la Fig. 3 se observa gráficamente esta componente. Cuando el usuario marca una opción y pulsa validar, la aplicación comprueba la corrección e informa al usuario del resultado. Si el alumno quiere pedir una explicación de la respuesta, pulsa el botón de explicación y se abre una ventana con el texto solicitado.

La manera de añadir preguntas a esta componente es similar a la forma anteriormente explicada. Los enunciados y las opciones están en un archivo `testi.html` donde *i* indica el número de orden. Dentro de cada nivel existe una subcarpeta explicación, que contiene archivos `explicacioni.txt` con las explicaciones correspondientes a la pregunta *i*. La aplicación de manera automática añade esa pregunta.





Fig. 3. Componente de test

## 4 Prueba y Evaluación de la Aplicación

A continuación presentamos con cierto grado de detalle el diseño del experimento para probar la herramienta desarrollada, así como los resultados obtenidos.

### 4.1 Formulación de hipótesis

Se plantearon dos cuestiones relacionadas: por una parte, determinar si el uso de la aplicación influye significativamente en el aprendizaje global de los alumnos y, por otra, ver si refuerza el aprendizaje en algún nivel concreto de la taxonomía de Bloom. El intervalo de confianza adoptado para rechazar o aceptar la hipótesis es del 95%.

### 4.2 Identificación de variables

Para realizar el estudio se tomaron dos variables: la aplicación constituye la variable independiente y la diferencia entre las notas obtenidas por cada grupo en la realización del pretest y del posttest la variable dependiente. Para la primera hipótesis se considera las diferencias de la nota global de cada test y para la segunda se tienen

en cuenta las notas de cada nivel. El objetivo de la investigación es determinar la influencia de la primera sobre la segunda.

El uso de la diferencia de notas entre los dos test como variable dependiente es una práctica habitual en este tipo de estudios, pues se eliminan factores externos al experimento, como distinta dificultad de las preguntas entre ambos test (que de hecho existe según muestran las medias en la Tabla 1.). Al restar las notas, cada alumno se mide a sí mismo.

### **4.3 Población y muestra**

La población objeto del estudio fueron los alumnos de la asignatura Programación Orientada a Objetos perteneciente al segundo curso de Ingeniería Informática de la Universidad Rey Juan Carlos en el curso 2004/2005.

Los 36 alumnos que asisten regularmente a las clases prácticas fueron divididos en dos grupos: uno experimental y otro de control. La división se realizó al azar por apellidos y teniendo en cuenta el sexo (está demostrado que hombres y mujeres se enfrentan de forma distinta a los ordenadores [5]), de manera que quedaron dos grupos de 18 miembros (13 alumnos y 5 alumnas) para evitar sesgos. Todos los alumnos están matriculados en la asignatura por primera vez y han recibido las mismas clases teóricas de la misma profesora. Con esta división se intentó que los grupos fueran lo más homogéneos posible. Los autores de este artículo no habían tenido contacto previo con los alumnos, para evitar generar test influenciados por los conocimientos previos, sobre todo para el nivel de aplicación, pues el resto de preguntas si es necesario saber lo aprendido con anterioridad [4].

### **4.4 Obtención de los datos**

Los tests fueron diseñados con 3 preguntas por nivel de Bloom, cuidadosamente seleccionadas para encajar en dichos niveles por los autores y con la ayuda de la profesora de la asignatura. El test, por lo tanto, está compuesto por 9 preguntas con 4 posibles respuestas. Para la puntuación, las respuestas acertadas contaban 1 punto y las falladas 0 (máximo 9 puntos), es decir, no se penalizó el uso del azar.

El cuestionario con las preguntas tipo test se realizó en clase práctica, correspondiente al tema de herencia, y se pasó a ambos grupos un test previo (pretest) al uso de la aplicación. Ambos grupos se encontraban en aulas distintas y no eran conscientes del experimento. Finalizado el pretest, los alumnos del grupo de control realizaron con el ordenador la práctica del tema. En contraposición, el grupo experimental instaló la aplicación y la usó durante aproximadamente media hora. Antes de finalizar la sesión práctica, se les entregó un cuestionario para evaluar cualitativamente la herramienta. Al día siguiente, a ambos grupos en clase teórica se les pasó un test (postest) con el mismo diseño y distintas preguntas.

#### 4.5 Resultados

Los test fueron respondidos por los dos grupos. Se pueden observar los resultados en la Tabla 1. Es de notar que se produjeron 3 bajas (muerte experimental) en ambos grupos, por lo que los datos de esos alumnos no se han tenido en cuenta en los resultados. Para el análisis de datos se usó la aplicación estadística R [18].

**Tabla 1.** Resultados generales de los tests

GRUPO	Pretest		Postest	
	Control	Experimenta l	Control	Experimenta l
Media	5,00	6,28	4,60	5,27
Mediana	5	7	4	6
Varianza	3,14	2,35	2,69	2,64
D. típica	1,77	1,53	1,64	1,62
Rango	1..8	3..9	2..7	2..8

Para estudiar la normalidad de las muestras hemos utilizado las pruebas de Shaphiro-Wilk y la de Kolmogorov-Smirnov que dan un valor del estadístico P. Si el valor de P obtenido en cada una de las pruebas no supera 0.05 (debido al criterio del 95% anteriormente escogido), se concluye que la muestra no sigue una distribución normal. Que las muestras sean normales indica que se pueden usar los métodos paramétricos para su estudio. En todos los casos el valor de P supera el valor umbral, por lo que deducimos que son normales.

A continuación comprobamos si la división realizada en grupos aleatorios es correcta o, en cambio, presenta algún sesgo (por ejemplo, que los alumnos más brillantes se encuentren en el mismo grupo). Como las muestras tienen distribución normal, se hizo el test F de igualdad de varianzas para los pretest dando  $P=0,59$ . Como las muestras cumplen los dos requisitos (normalidad e igualdad de varianza) usamos la prueba T de Student para comprobar si ambos grupos provienen de la misma población (de la misma aula y con conocimientos parecidos). Este test se basa en probar que las medias de los pretest no difieren significativamente. Aplicada la prueba se obtuvieron los siguientes resultados:

**Tabla 2.** Prueba de T-Student para el Pretest

T	Grados de libertad	P
-1,9828	33	0,056

Al ser tan justo el valor de P en la Tabla 2. (recordamos que si es menor que 0,05 se rechaza la hipótesis y por tanto, se concluye que ambos grupos no son de la misma población), se realizó el cálculo del intervalo de confianza del 95% de la media (IC 95% de la media). Los resultados obtenidos se muestran en la Tabla 3. :

**Tabla 3.** Intervalo de confianza del Pretest

Pretest		
GRUPO	Control	Experimental
IC 95%	4,38-6,07	5,54-7,02

Como ambos intervalos se solapan parcialmente, junto con la prueba de T-Student, nos permiten no rechazar la hipótesis y afirmar que las diferencias entre ambos grupos son achacables al azar y por lo tanto pertenecen a la misma población.

A continuación analizamos el efecto de la aplicación en su aprendizaje. En la Tabla 4. se muestran los resultados generales de la variable dependiente, que es la diferencia entre ambos test (nota del pretest menos nota del postest).

**Tabla 4.** Diferencias entre el pretest y el postest

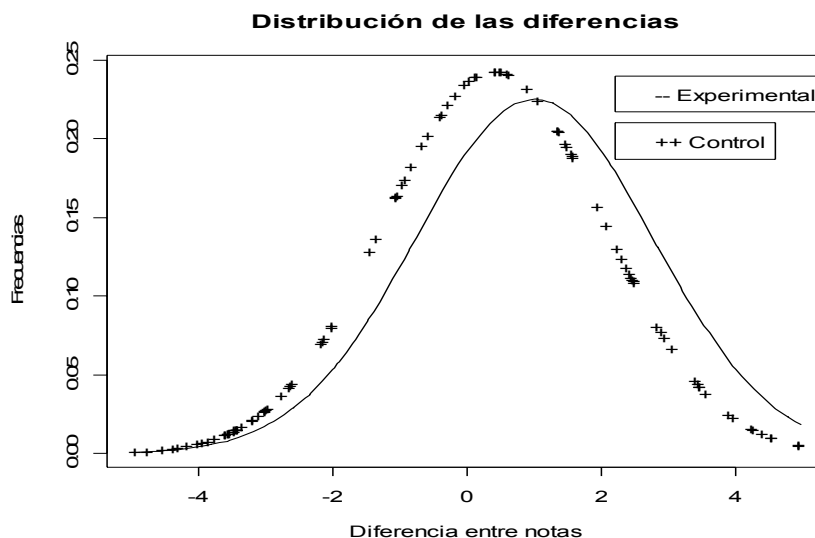
GRUPO	Control	Experimental
Media	0,4	1,0
Mediana	1,0	1,0
Varianza	2,69	3,14
D. típica	1,64	1,77
Rango	-2..3	-2..4

Antes de hacer la prueba de T-Student, hay que probar la normalidad de las diferencias y la igualdad de varianzas. Se aplican los mismos test explicados anteriormente. Los resultados de ambas pruebas nos indican que las diferencias son normales y poseen varianzas similares. La prueba de T-Student da los valores expresados en la Tabla 5.

**Tabla 5.** Prueba de T-Student para las diferencias

T	Grados de libertad	P
-0,9625	28	0,34

Para comparar visualmente los resultados entre ambos grupos se ha generado la Fig. 4 que muestra la distribución de ambos grupos.



**Fig. 4.** Distribución de las diferencias de notas entre ambos grupos

Con el fin de estudiar la influencia de la aplicación sobre los conocimientos adquiridos en los distintos niveles de Bloom, se han realizado las mismas pruebas anteriores, pero ahora separando las notas por grupos asociados a los niveles de la jerarquía. Los resultados se encuentran agrupados en la Tabla 6. Las pruebas realizadas sobre normalidad e igualdad de varianza nos permiten aplicar la T-Student (ver Tabla 7. ) para comparar los resultados.

**Tabla 6.** Diferencias agrupadas por niveles de Bloom

	Control			Experimental		
	N1	N2	N3	N1	N2	N3
Media	0,93	-0,80	0,27	0,47	-0,13	0,67
Mediana	1	-1	0	0	0	1
Varianza	1,35	0,59	0,77	1,56	0,85	0,51
D. típica	1,16	0,77	0,88	1,25	0,92	0,72
Rango	-2..3	-2..0	-1..2	-1..3	-1..2	0..2

**Tabla 7.** Prueba de t-student para las diferencias de notas por niveles de Bloom

	N1	N2	N3
T	1,07	-2,15	-1,36
Grados de libertad	28	27	27
P	0,29	0,04	0,19

Con el fin de evaluar el grado de satisfacción personal con la herramienta se realizó un cuestionario al grupo experimental. Se usó la escala de Likert, debiendo el alumno mostrar su opinión desde totalmente de acuerdo (5) hasta totalmente en desacuerdo (1). Los resultados se ven en la Tabla 8.

**Tabla 8.** Evaluación de la herramienta (grupo experimental)

PREGUNTA	OPINION	Media
Es fácil de instalar	Totalmente de acuerdo	5
El uso es intuitivo	Totalmente de acuerdo	4,7
La explicación teórica es completa	Bastante de acuerdo	3,8
Las demostraciones me ayudan a comprender la materia	Bastante de acuerdo	4,2
La realización de tests me parece acertada para motivarme	Bastante de acuerdo	4,4
He notado que la dificultad aumenta con los niveles	Totalmente de acuerdo	4,5
La explicación de las respuestas de los tests la encuentro útil	Bastante de acuerdo	3,9
La usaría en casa para practicar	Bastante de acuerdo	4,4
A grandes rasgos, la herramienta me parece útil	Totalmente de acuerdo	4,5

#### 4.6 Discusión de los resultados

Primero se intentó demostrar que la división realizada por grupos era correcta. Para ello, con las notas del pretest, se obtuvo que ambas muestras seguían distribuciones normales y presentaban varianzas homogéneas. Aplicada la prueba de T-Student (Tabla 2. ) y la prueba del IC 95% para contrastar las notas medias en el pretest, el resultado fue que no existían diferencias significativas entre ambos grupos. Así nos aseguramos de que las diferencias entre notas del pretest y del postest sólo pueden deberse al uso de la aplicación.

Para comprobar la primera hipótesis (la aplicación influye significativamente en el aprendizaje del concepto de herencia), se analizaron las diferencias entre las notas totales del pretest y del postest. Como se muestra en la Tabla 4. , la media del grupo experimental es mayor que la del grupo de control, lo que podría conducir erróneamente a pensar que el uso de la aplicación mejora considerablemente el aprendizaje. Sin embargo, el estudio estadístico muestra que no existen diferencias significativas entre ambos grupos (Tabla 5 y Fig. 4. ). Esto no quiere decir que el uso de la herramienta no sea eficaz: habría que experimentar con un grupo de alumnos mayor y en distintas universidades para demostrar o refutar definitivamente la hipótesis.

La segunda hipótesis es que la herramienta refuerza el aprendizaje en algún nivel de Bloom. Los datos obtenidos en la Tabla 6. y los estudios estadísticos de la Tabla 7. muestran que salvo para el nivel 2 (de comprensión), las diferencias no son significativas. La prueba de T-Student para el nivel 2 arroja un P-valor de 0.04, que es menor que el valor fijado a priori de 0.05. Esto implica que existe una diferencia significativa, no achacable al azar. Justamente nuestra aplicación apoya más el nivel

2, pues las tres componentes refuerzan ese nivel. La componente de teoría, sirve para los dos primeros niveles, la de demostración se centra en el nivel de comprensión y la de test cubre los tres niveles inferiores.

En cuanto a la opinión de los alumnos que usaron la aplicación, la impresión fue muy buena. No fue necesario explicar su funcionamiento. Como se muestra en los resultados, a los alumnos les pareció fácil de instalar, intuitiva y de utilidad. Queremos resaltar también la opinión de los alumnos en cuanto a la pregunta: "He notado que la dificultad aumenta con los niveles". Están totalmente de acuerdo en esa apreciación. Este tema es clave para conseguir alcanzar los objetivos marcados inicialmente, que consistían en cubrir los tres niveles inferiores de Bloom con la componente de tests. Esta apreciación nos permite ser optimistas en cuanto al hecho de que es posible alcanzar el nivel de aplicación con el uso de los problemas como método de aprendizaje. Como resumen final de la evaluación diremos que la herramienta obtuvo una muy buena valoración global.

## 5 Conclusiones y Trabajos Futuros

Hemos presentado una aplicación de ayuda al alumno para aprender el concepto de herencia. Se ha descrito brevemente el marco general de nuestra propuesta y la propia aplicación. Ésta incluye módulos para los niveles de conocimiento y comprensión y otro módulo adaptativo para evaluar los 3 niveles inferiores. También hemos mostrado los resultados de una evaluación de la aplicación, donde se ha concluido que refuerza el aprendizaje en el nivel de comprensión, siempre teniendo en cuenta que el experimento se realizó sobre un número pequeño de alumnos (18 inicialmente en cada grupo). Los discentes han valorado positivamente el uso de esta herramienta como método de ayuda al aprendizaje.

Nuestro objetivo inmediato es continuar nuestro trabajo para alcanzar niveles superiores de la taxonomía de Bloom. Se encuentra en desarrollo otra herramienta para la enseñanza de la POO, que pretende alcanzar el nivel de aplicación basándose en la resolución de problemas. Por otro lado, como línea paralela, hemos desarrollado una herramienta a nivel de diseño denominada TextOO [14] que se encuentra en fase de evaluación y que se fundamenta en el enunciado del problema como base para el diseño de una aplicación. También queremos aumentar las facilidades de interacción del componente de demostración. Estos objetivos exigen una infraestructura distinta, que estamos desarrollando.

## Agradecimientos

Este trabajo se ha financiado con el proyecto TIN2004-07568 del Ministerio de Educación y Ciencia. Queremos agradecer la colaboración de Leandro Martín Molina y de Soto Montalvo para realizar la evaluación.

## Referencias

1. Apple, D. K., Nygren, K. P., Williams, M. W. & Litynski, D. M.,: Distinguishing and elevating levels of learning in engineering and technology instruction”, Proc. of the 32<sup>nd</sup> ASEE/IEEE Frontiers in Education Conference, IEEE Computer Society Press, 2002, pp. T4B 7-11.
2. Bell, J.T., Fogler S.: The investigation and application of virtual reality as an educational tool. Proc. Of the American Society for Engineering Education 1995. Session number 2513.
3. Bloom, B., Furst, E., Hill, W. & Krathwohl, D. R.: Taxonomy of Educational Objectives: Handbook I, The Cognitive Domain, Addison-Wesley, 1956.
4. Buckley J.; Exton C.,: A framework for assesing programmers’ knowledge of software systems. En Proceedings of the 11<sup>th</sup> IEEE International Workshop on Program Comprehension (IWPC’03), 2003.
5. Carter, J. & Jenkins, T., :Gender and programming: What’s going on?, Proc. 4<sup>th</sup> Annual Conf. Innovation and Technology in Computer Science Education (ITiCSE’99), ACM Press, 1999, pp. 1-4.
6. Farnsworth, C. C., Using computer simulations in problem-based learning. In proceedings of Thirty Fifth ADCIS conference, Omni Press, Nashville, TN, (1994), 137-140.
7. Felder, R.M.: Matters of style. ASEE Prism, 6(4), 18-23. December 1996.
8. Hernán Losada, I.; Lázaro Carrascosa C.A., Velázquez Iturbide, J.A.:On the use of Bloom’s taxonomy as a basis to design educational software on programming Proceedings of the World Conference on Engineering and Technology Education WCETE, 2004, pp 351-355.
9. Hernán-Losada, I., Lázaro-Carrascosa, C., and Velázquez-Iturbide, J.Á. Una aplicación educativa basada en la jerarquía de Bloom para el aprendizaje de la herencia de POO. In Proc. of the VII Simposio Internacional de Informática Educativa (SIIE 2005), 2005.pp 107-112
10. Hundhausen, C.D., Douglas, S.A., & Stasko, J.T.,:A meta-study of algorithm visualization effectiveness, Journal of Visual Languages and Computing, Vol. 13, No. 3, 2002, pp. 259-290.
11. Gayo, D.; Fernández, H.; Torre, F., La autoevaluación como método de aprendizaje, VIII Jornadas de Enseñanza Universitaria de la Informática (JENUI), 2002, pp. 359-366
12. Krathwohl D.R.: A revision of Bloom’s taxonomy: An overview. Theory into practice, Volume 41, Number 4, Autumn 2002. College of education. The Ohio State University.
13. Kumar, A.N.: Learning programming by solving problems. In: Cassel, L., Reis, R.A. (eds.): Informatics Curricula and Teaching Methods, Kluwer Academic (2003) 29–39
14. Lázaro, C.; Hernán, I.; Velázquez, J.A.; Gortazar, F.; Gallego, M.: Textoo: Learning Object Oriented Modeling using the Enunciate, TUCS Publication Series, General Publications 2006, Vol 41.
15. McConnell, J.J. Active Learning and its use in Computer Science, Conference Proceedings of ITiCSE 1996, New York: ACM Press, 1996. pp. 52-54
16. Naps, T., Roessling, G. *et al.*,: Exploring the role of visualization and engagement in computer science education, SIGCSE Bulletin, Vol. 35, No. 2, June 2003, pp. 131-152.
17. Thomas, L.; Ratcliffe, M.; Woodbury, J.; Darman, E. .: Learning Styles and Performance in the Introductory”, Proceedings of the SIGCSE’02, 2002 pp. 33-37
18. Venables, W.N., Smith, D. M.& the R Development Core Team, An Introduction to R, <http://www.r-project.org/> disponible 26-05-2005.