

DESARROLLO DE *APPLETS* EDUCATIVOS PARA LA FORMACIÓN BÁSICA EN DISPOSITIVOS ELECTRÓNICOS

E. CASILARI, F. GONZÁLEZ-CAÑETE, J.M. VILLALBA, E. GÓMEZ GARCÍA
Departamento de Tecnología Electrónica, ETSI Málaga, Universidad de Málaga
Universidad de Málaga, 29071- Málaga, España

En este trabajo se resume un proyecto que, mediante sencillos applets Java accesibles por Internet, ofrece al alumno programas interactivos para comprender diversos aspectos, físicos y circuitales, de la fabricación, funcionamiento y modelado de los dispositivos electrónicos.

1. Introducción

En la asignatura troncal de Electrónica de Dispositivos (situada en el primer curso de Ingeniería de Telecomunicación de los planes de estudio vigentes de Universidad de Málaga), se tratan los aspectos más importantes del comportamiento físico de los semiconductores, los diodos, los transistores bipolares y los transistores unipolares JFET y MOSFET. También se estudian algunos circuitos simples con los que se pretende que los estudiantes aprendan a polarizar los dispositivos y a conocer su comportamiento en pequeña señal y en conmutación. Además, la asignatura permite introducir algunos aspectos de las tecnologías de fabricación de circuitos integrados así como comenzar a familiarizarse con los distintos materiales usados en dicho proceso.

Por tanto, esta asignatura requiere un gran esfuerzo por parte de los alumnos para asimilar la gran cantidad de nuevos contenidos, la mayoría de ellos abstractos, que se presentan. Añadidamente, el alumnado también tiene que enfrentarse a cientos de expresiones matemáticas, que incluyen multitud de variables y parámetros, a veces interdependientes, lo que dificulta claramente su comprensión.

En Electrónica (y en especial si se trata de los aspectos físicos), donde la mayoría de los parámetros dependen de bastantes variables, la transparencia, método muy empleado por los docentes universitarios, quizá sea un método expositivo que no da una idea global de todos los procesos físicos que están ocurriendo en el interior del dispositivo estudiado o en el circuito que está siendo analizado. Las transparencias pueden resultar más o menos adecuadas para la enseñanza de la Electrónica, pero no cabe duda de que a veces no cubren todas las necesidades de este tipo de materias. Para este fin deberían permitir incluir animaciones, un colorido amplio así como cierto grado de interacción con el estudiante.

Todo el proceso de aprendizaje se facilita en cierta medida por medio del uso de software de simulación, análisis o síntesis de los dispositivos electrónicos y de los circuitos de los que forman parte. Un problema añadido al usar estas herramientas software es el tiempo que hay que invertir en comprender el funcionamiento de la propia herramienta antes de empezar a sacar provecho de estos programas. De un modo general, los autores en [1] dividen en tres grandes grupos estas herramientas para facilitar el aprendizaje:

- Software de propósito general. Incluye las herramientas que no están diseñadas para aplicarlas en un contexto específico, sino que el estudiante emplea para materializar cálculos, ordenar y visualizar datos o redactar textos (hojas de cálculo, procesadores de texto y bases de datos,...).

- Software para la obtención de datos experimentales.

- Aplicaciones específicas o de enseñanza asistida por ordenador. Aquí se engloban los programas que permiten la adquisición de unos contenidos docentes mediante uso directo del ordenador.

Como ya se ha comentado, debido a la gran cantidad de expresiones matemáticas y parámetros que se han de emplear para la descripción de los fenómenos físicos que se producen en los dispositivos electrónicos, es deseable que el alumno disponga de una herramienta visual e interactiva con la que pueda representar, por medio de animaciones, gráficos, etc, algunos de los procesos físicos más

importantes involucrados en el comportamiento de los semiconductores y los dispositivos que con ellos se fabrican. De esta forma, los alumnos comprenderán mejor cómo varían las características físicas de los semiconductores al variar ciertos parámetros, y cómo están relacionados unos con otros. Disponer de una herramienta de este tipo puede ser una fuente muy útil para el aprendizaje.

A todo esto debe aunarse la facilidad de manejo para que los alumnos no pierdan tiempo en el conocimiento de la herramienta y puedan pasar directamente al contenido didáctico. Por tanto, es necesario desarrollar interfaces simples y de fácil manejo para el usuario. Además, se ha de permitir el mayor grado de configuración por parte del alumno para que pueda experimentar el mayor número de condiciones diferentes el experimento que se simula con la herramienta. Con esta característica también se puede hacer hincapié en la relación que guardan los parámetros y el estado o las características del dispositivo con el que estamos trabajando. También es deseable que la herramienta permita conocer al alumno su grado de conocimiento de la materia que está tratando. Esto puede conseguirse, por ejemplo, con un test de autoevaluación, con la posibilidad de ocultar datos de la pantalla para que el alumno pueda hacer el ejercicio de calcularlos por su cuenta y después confirmar los resultados con el programa, mediante la inclusión de algunas aplicaciones que formulen preguntas o problemas que tengan que ser resueltos, etc.

Una posible implementación de esta herramienta puede obtenerse mediante un conjunto de aplicaciones, realizadas con algún lenguaje de programación que se pueda embeber fácilmente en documentos HTML (*HyperText Markup Language*). Algunos ejemplos de estos lenguajes pueden ser Java, JavaScript y Flash. Además, con ellos estamos usando una tecnología muy conveniente para crear, ampliar fácilmente y difundir por la red Internet programas de simulación pequeños e interactivos con fines docentes.

Estas razones, junto con la difusión masiva y el conocimiento que tienen los alumnos de la *World Wide Web* (WWW), hacen de estas aplicaciones los elementos idóneos para construir esa herramienta visual e interactiva que facilite a los alumnos la comprensión y estudio de la física de los semiconductores así como los procesos necesarios para la fabricación de los dispositivos.

Existen en el mercado varios programas comerciales, preparados por diversas instituciones o empresas, que cubren ampliamente las necesidades que pueden surgir al hacer un curso *on-line*. Sin embargo, para el tema que nos ocupa, que es la enseñanza de conceptos de Electrónica a través de Internet, de momento no es necesario desarrollar todo un curso completo *on-line*, sino que sólo deseamos confeccionar un portal complementario a las clases tradicionales impartidas en el aula. Dicho de otro modo, nuestro curso a través de Internet no pretende ser un sustituto de las clases presenciales, sino que su objetivo es servir de complemento y ayuda para la comprensión de conceptos que, por su dificultad, son mucho más entendibles por medio de una simulación con animaciones, gráficos, etc. y mediante la interacción con el alumno. Quizá, cuando la colección de temas tratados en este portal sea suficientemente grande y se haya complementado con ciertas funcionalidades nuevas (exámenes *on-line*, tests, listas de correo, generación de problemas, etc.) podría comenzar a pensarse en la posibilidad de llevar, al menos parcialmente, los conocimientos de la asignatura a los alumnos sin la necesidad de asistir a las clases tradicionales.

Un curso interactivo y fuera de las aulas fomentará también la colaboración entre alumnos y podrá ahorrar bastante tiempo al tutor, ya que muchas de las dudas que surgen durante el proceso de aprendizaje podrán ser solucionadas por los propios alumnos con la ayuda de estas simulaciones. Éstos podrán simular una vez tras otra cualquier aspecto teórico que no comprendan, obtendrán los resultados en tiempo real y estarán en condiciones de disipar multitud de dudas. También podrán recurrir al tutor del curso con preguntas concretas, apoyadas con una simulación (con la configuración y los parámetros adecuados)

2. Objetivos del proyecto y tecnologías empleadas

El objetivo básico era desarrollar un conjunto de pequeñas aplicaciones accesibles vía Web e independientes entre sí que ilustrasen algunos aspectos del funcionamiento de los semiconductores, el diodo de unión PN, el transistor bipolar y el transistor unipolar MOS.

Los temas a tratar incluían temas de física de semiconductores, conmutación de los dispositivos, descripción de hojas de catálogo, fases de fabricación de la tecnología MOS y de un circuito integrado, etc.

La tecnología elegida para desarrollar estas aplicaciones fueron los *applets* de Java. Los *applets* son pequeños programas que aparecen embebidos en las páginas Web, al igual que gráficos o texto, pero con la capacidad de ejecutar acciones muy complejas, como animar imágenes, establecer conexiones de red (aunque con limitaciones) o presentar menús y cuadros de diálogo para luego emprender acciones. De acuerdo de nuevo con [1], los *applets* aportan una notable herramienta para desarrollar una visión constructivista del proceso enseñanza-aprendizaje, en el que, frente al clásico modelo conductista, el alumno construye su conocimiento a partir de la interacción con unos programas.

Software utilizado	Propósito
DJ Java Decompiler 3.7	Se ha usado para decompilar clases ya creadas y de las que no se disponía del código fuente.
JCreator Pro 2.5 Pro	Se ha usado para la programación de algunos <i>applets</i> . Debido a su falta de facilidad para el desarrollo gráfico de interfaz de usuario se desechó su uso para emplear JBuilder.
JBuilder X Enterprise	Se ha utilizado para la programación y depuración de la mayoría de los <i>applets</i> . Facilitó bastante el desarrollo de la interfaz gráfica.
Dreamweaver MX	Su utilizó para desarrollar los documentos HTML que van inmersos dentro de algunos <i>applets</i> (forman parte de él), las páginas web donde se incluyen los <i>applets</i> y la página principal del tutorial.
Circuit Maker 6	Se necesitó este software para dibujar los circuitos que están incluidos en algunos <i>applets</i> , ya que permitía exportar los esquemas de circuitos fácilmente.
Paint Shop Pro 7	Se empleó este software de edición gráfica para la transformación de imágenes que se insertaron en ciertos <i>applets</i> .
Paint 5.1	Se necesitó para algunas manipulaciones necesarias en las imágenes antes de ser insertadas en los <i>applets</i> .

Tabla 1. Software utilizado en el proyecto

Al estar desarrollados en Java, los *applets* se benefician de todas las ventajas de este lenguaje (portabilidad, capacidad de trabajo multihebra, independencia del sistema operativo empleado, robustez y seguridad, rapidez de desarrollo, existencia de amplias bibliotecas de clases, etc.). Al ejecutarse localmente en la máquina cliente que lo descarga a través de Internet, los *applets* liberan de su ejecución al servidor (lo que evita la exigencia de máquinas centrales de gran capacidad de proceso y reducen al mínimo el costo de la implementación de experiencias de este tipo). Además, una vez en el PC del alumno, éste podrá guardar la aplicación localmente y acceder a ella más tarde sin la necesidad de tener establecida una conexión a Internet, con lo que ganamos en disponibilidad. Las

ventajas de la ejecución local, junto con sus posibilidades gráficas, hacían más atractivos los *applets* que otras tecnologías Web que se ejecutan en el servidor, ya fuesen Java (como los Servlets o los JSP) o no (como PHP). En el caso concreto de este proyecto, añadidamente, la elección de esta tecnología posibilitó la reutilización de las clases Java que sobre dispositivos se han realizado en la Universidad Buffalo (EEUU), con motivo de un proyecto muy similar a este [2]. Por otro lado, frente a otras tecnologías de ejecución en el navegador, como Flash, de enorme potencial gráfico, los *applets* presentan la virtud de la gratuidad del uso de Java (si comparamos con el editor de Macromedia) así como una mayor facilidad a la hora de definir ciertos modelos de interactividad.

Para el funcionamiento de los *applets* basta con que el equipo del cliente posea una conexión a Internet y tenga instalada la máquina virtual Java, algo que normalmente ya se tiene si se trabaja con cierta asiduidad con cualquier navegador. Más allá de eso, el alumno no ha de instalar ningún software específico. Los *applets* se diseñaron con especial cuidado para que no fueran excesivamente grandes en tamaño, con el objeto de agilizar su consulta *on-line* con independencia del modo y velocidad de acceso del alumno a Internet.)

La tabla 1 resume las tecnologías software empleadas en el desarrollo de los *applets*.

3. *Applets* desarrollados

Hasta la fecha se han desarrollado los siguientes *applets*:

-Análisis de la evolución del nivel de Fermi: El *applet* ilustra la variación del nivel de Fermi, así como de la concentración intrínseca, de un material semiconductor según la temperatura, la posible presencia de luz, el tipo de material semiconductor (banda prohibida, número de estados equivalentes en banda de conducción y de valencia), el dopaje y el tipo de impurezas (que determinan la posición del nivel permitido dentro de la banda prohibida y, por tanto, el grado de ionización).

-Estudio del semiconductor homopolar en equilibrio con distribución no uniforme de impurezas: El *applet* ilustra la formación de carga, campo y potencial internos para distintos perfiles de dopaje (exponencial y lineal), así como los procesos de difusión y arrastre.

-Ilustración de la evolución de la conductividad de un material semiconductor: el programa representa la evolución con la temperatura de la conductividad de un material semiconductor en función del tipo de dopaje, de la concentración de dopante y de la posible presencia de un impulso luminoso en régimen estacionario. Queda igualmente modelada y representada la evolución de las movibilidades de huecos y electrones.

-Evolución, en el tiempo, de la concentración de huecos y electrones de un semiconductor en presencia de un impulso luminoso: el programa permite definir el tiempo de vida, la tasa de ruptura de enlaces y los dopajes.

-Estudio de la unión PN en equilibrio y polarizada: *Applet* donde se representa la formación de bandas, así como la distribución de carga y el campo en la zona de carga espacial en función de la polarización.

-Estudio de La ley de Shockley: Representación de la evolución de la corriente en un diodo PN en función de la polarización, el dopaje, el tipo de semiconductor y la temperatura.

-Estudio del diodo varactor: Evolución de la capacidad equivalente de un diodo varactor para distintas polarizaciones, dopajes y tipos de unión (abrupta, hiperabrupta y gradual lineal).

-Análisis de un circuito con un diodo varactor: Evolución de la función de transferencia de un circuito sintonizador en función de la variación de la polarización del diodo.

-Conmutación del diodo (véase la Figura 1): el *applet* ilustra la conmutación de un diodo en un circuito donde la polarización pueda pasar (a voluntad del usuario) desde directa a disrupción pasando

por corte. Se representa la evolución en el tiempo de la corriente, la tensión en el diodo y la carga en las zonas neutras así como la anchura de la zona bipolar.

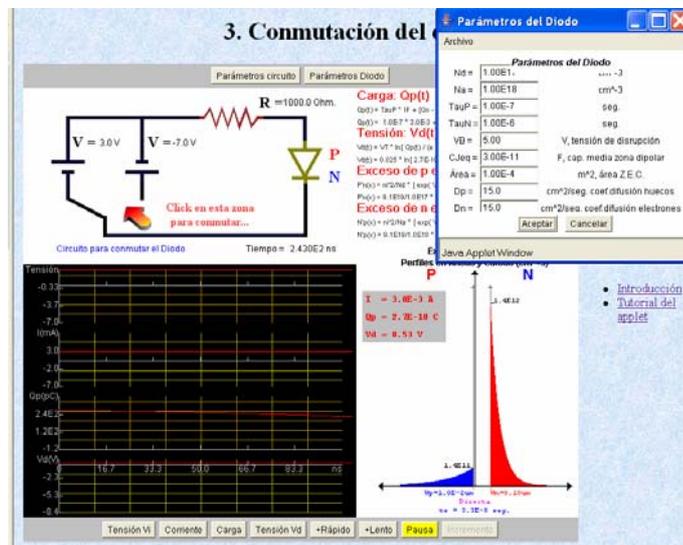


Figura 1. Ejemplo de *applet* sobre la conmutación del diodo

-Estudio de la distribución de corrientes y portadores en un BJT: este *applet* permite observar la distribución de corrientes y portadores, así como los parámetros del transistor, de un transistor bipolar polarizado en función de la tensión aplicada. Los parámetros físicos básicos del transistor (dopaje, naturaleza del semiconductor) son programables también.

-Análisis de la conmutación del transistor: Ejemplificación del proceso de conmutación de un transistor bipolar. Las gráficas del *applet* representan las variaciones de la carga en base, las corrientes de colector y base y la tensión de salida, que se producen en un bipolar cuando cambia la tensión con la que está polarizado. El programa permite definir los parámetros del circuito de polarización así como los del propio transistor.

-Generador de ejercicios de polarización (véase figura 2): eligiendo al azar un circuito a partir de una serie de circuitos tipo de polarización y dando valores a los componentes (resistencias, pilas de polarización, parámetros del componente activo) el programa genera un problema al alumno donde se le pregunta por el punto de polarización. Presionando un botón el alumno podrá conocer la solución.

-Estudio de la polarización del transistor: a partir de los parámetros de Ebers Moll y de las tensiones de polarización entre los terminales, el programa determina las corrientes y representa los perfiles de minoritarios en las tres zonas: colector, emisor y base. También se representan las anchuras de las dos zonas bipolares.

-Etapa amplificadora de ganancia en emisor común: a partir de un circuito en emisor común, el programa posibilitará definir los valores de las distintas resistencias y los parámetros típicos del transistor. Representará el circuito en pequeña señal, calculando sus parámetros, así como los híbridos. Asimismo se dibuja la recta de carga y el movimiento por ella del punto de trabajo conforme se amplifica la señal de entrada. Igualmente se ilustra la señal de salida junto con la de entrada (cuya amplitud podrá elegir el usuario), de modo que se pueda observar la amplificación del circuito y las limitaciones del rango dinámico.

-Estudio de la formación de canal en una estructura JFET: Ilustración de la modificación que sufre el canal en un transistor JFET cuando se aplican tensiones en drenador. Se pueden alterar los parámetros físicos básicos del JFET

-Estudio de la capacidad MOS ideal: Representación de las variaciones y distribución espacial de carga, campo y potencial, así como de las bandas de energía, de una estructura MOS en función de la polarización. Los parámetros físicos básicos de la capacidad son programables.

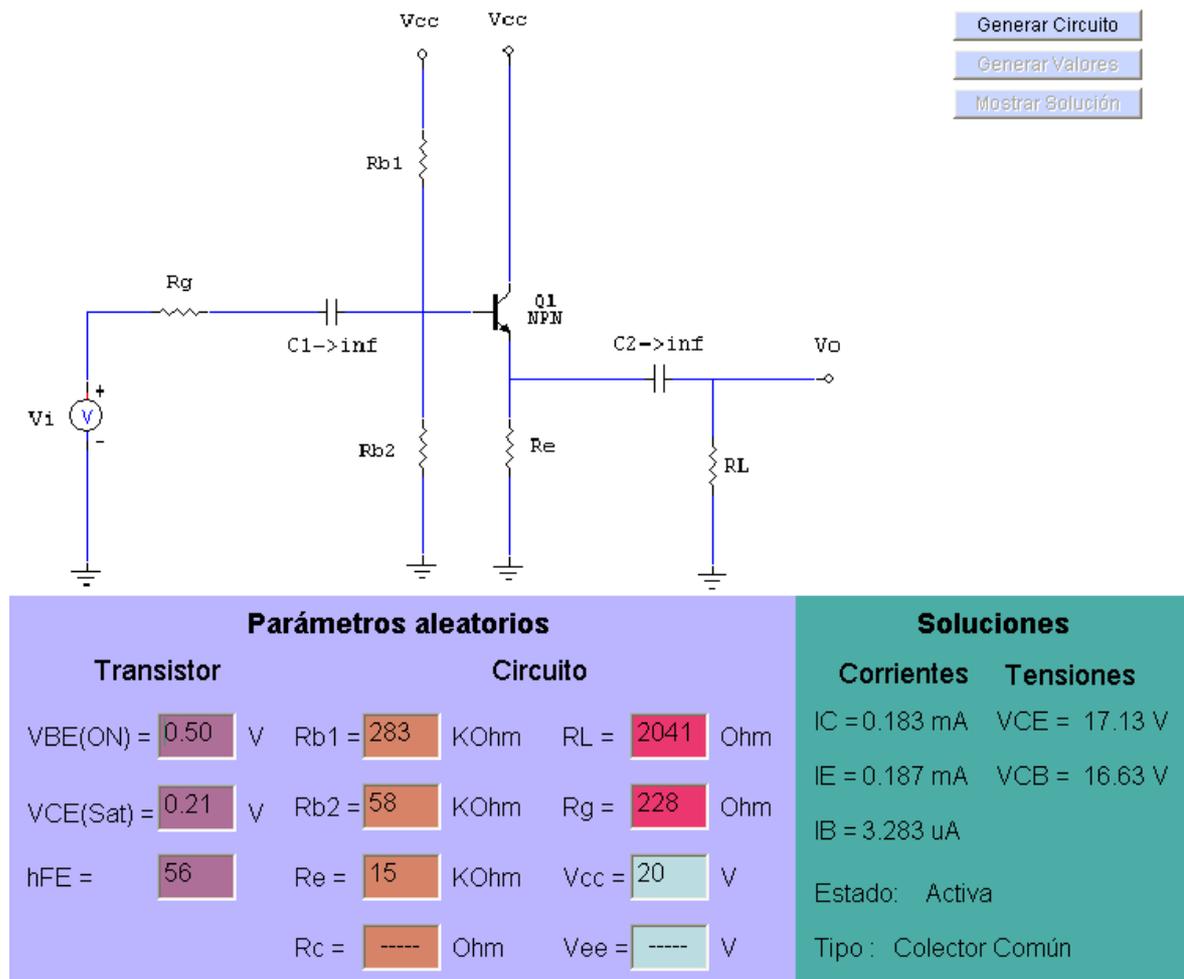


Figura 2. Ejemplo de *applet* sobre la generación de problemas sobre polarización

-Estudio de los pasos y capas en la fabricación de un transistor MOSFET: el *applet* ilustra los pasos en la fabricación de un transistor MOSFET de canal n. En él aparecen todos los pasos y las capas de los distintos materiales usadas en la fabricación del dispositivo.

-Definición y fases en la fabricación de circuitos integrados. En este último bloque se han realizado dos programas. Uno muestra de forma genérica todos los pasos necesarios para la fabricación de un circuito integrado y el otro recrea todas las fases necesarias para la fabricación de un diodo PN de unión.

LA URL de la página de acceso a los *applets* (cuyo menú principal se ha representado en la figura 3) es: <http://pc21te.dte.uma.es/asignatura/applets/tutorial.html>. También se puede consultar una versión en: <http://campusvirtual.uma.es/edispo/>

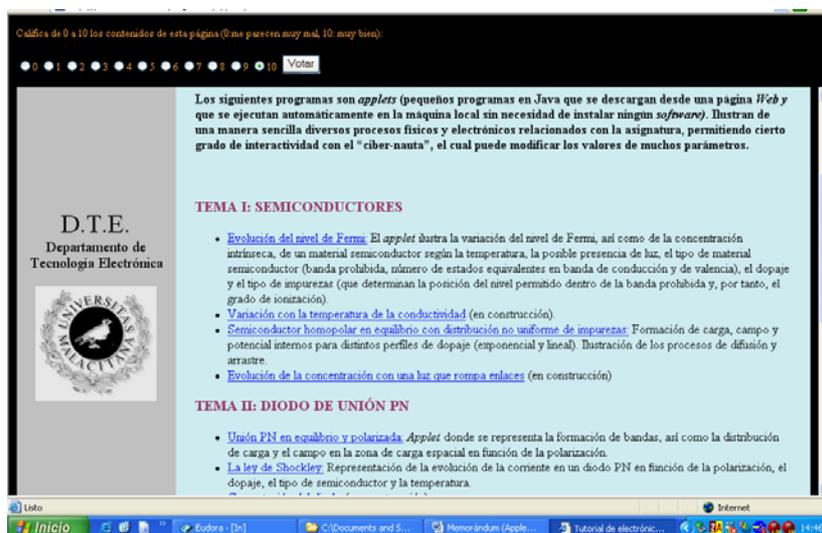


Figura 3. Página principal de acceso a los *applets*

4. Pruebas del software

Una vez que un *applet* es desarrollado y funciona correctamente ya que su interfaz gráfica es aparentemente la adecuada y la manipulación de los elementos configurables no presenta problemas en una primera observación, se realizan las llamadas pruebas de “caja blanca”, es decir, pruebas de código abierto donde se intenta comprobar el funcionamiento de todas y cada una de las líneas de código que forman el *applet* (de ordinario varios miles de líneas). Dentro de las pruebas de “caja blanca” se efectuaron las pruebas típicas de cubrimiento (ejecución de, al menos, una vez todas las instrucciones que forman cada *applet*), de condiciones (evaluación de todas las decisiones del código) y de bucles (ejecución de todos los bucles del *applet* cierto número de veces). Tras estas se efectuaron las también típicas pruebas de “caja negra”, las cuales se abstraen del código y tratan cada *applet* como un todo. En estas pruebas tan sólo se conoce la interfaz del *applet*. En ellas se trata de probar todos los elementos de la interfaz de usuario de todos los *applets* del proyecto, mediante tests de cubrimiento (probando todos los controles y funciones que el *applet* permite), pruebas de datos (probando con varias clases de datos para las posibles entradas a introducir en cualquier control del *applet*) y pruebas de valores límites (complementando las de datos anteriores mediante la sistemática inclusión como entradas de valores dentro y fuera de los rangos esperados).

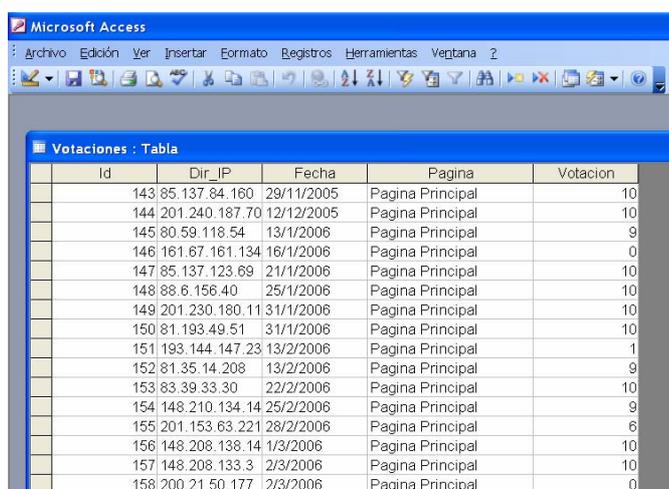
A estas pruebas genéricas se añadieron pruebas de entorno que comprobaron el funcionamiento de los programas cuando son accedidos desde distintos navegadores. Finalmente, se hicieron pruebas de prestaciones, concretamente midiendo repetidamente el tiempo de descarga de los *applets* cuando eran accedidos desde un equipo remoto conectado a Internet mediante una conexión típica ADSL a 1 Mbps. Salvo aquellos *applets* (dos en concreto), en los que había que descargar unas hojas de catálogo o, en su caso, un número considerable de fotografías JPEG, los tiempos de acceso para todos los *applets* resultaron ser prácticamente despreciables (menores a 2 s).

5. Evaluación de la experiencia y conclusiones

Tras publicitar entre los alumnos la existencia de la página creada, se adoptaron las siguientes estrategias para evaluar el uso del material desarrollado:

1. Uso de contadores de visitas. A tal efecto, se incluyó en el servidor Web propio un contador del número de visitas, y, al tiempo, se aprovechó el servicio que ofrece para tal fin el servidor del Campus Virtual.

2. Uso de mecanismo de votación. Para ello, se incorporó a la página principal del proyecto ubicada en el servidor propio un marco (*frame*) que permitía al alumno otorgar a la experiencia una calificación numérica entre 0 a 10. Para impedir en la medida de lo posible que un usuario votara varias veces (sesgando los resultados) el programa desarrollado no tenía en cuenta aquellas votaciones repetidas desde la misma dirección IP durante cierto número de horas. Los resultados de la votación (así como la fecha de la misma y la dirección IP desde la que se efectuaba) se guardaron en una base de datos SQL (accesible mediante mecanismos JDBC) en el servidor propio, ilustrada en la Figura 4.



Id	Dir_IP	Fecha	Pagina	Votacion
143	85.137.84.160	29/11/2005	Pagina Principal	10
144	201.240.187.70	12/12/2005	Pagina Principal	10
145	80.59.118.54	13/1/2006	Pagina Principal	9
146	161.67.161.134	16/1/2006	Pagina Principal	0
147	85.137.123.69	21/1/2006	Pagina Principal	10
148	88.6.156.40	25/1/2006	Pagina Principal	10
149	201.230.180.11	31/1/2006	Pagina Principal	10
150	81.193.49.51	31/1/2006	Pagina Principal	10
151	193.144.147.23	13/2/2006	Pagina Principal	1
152	81.35.14.208	13/2/2006	Pagina Principal	9
153	83.39.33.30	22/2/2006	Pagina Principal	10
154	148.210.134.14	25/2/2006	Pagina Principal	9
155	201.153.63.221	28/2/2006	Pagina Principal	6
156	148.208.138.14	1/3/2006	Pagina Principal	10
157	148.208.133.3	2/3/2006	Pagina Principal	10
158	200.21.50.177	2/3/2006	Pagina Principal	0

Figura 4. Base de datos de control de accesos a la página y de las notas recogidas sobre la aplicación desarrollada

3. Encuestas por escrito. En la encuesta se incluyen diversas preguntas sobre la opinión que los alumnos tienen acerca de las posibilidades docentes de Internet así como de la propia utilidad de la experiencia que aquí se comenta.

Como conclusiones se puede decir que, aunque el alumnado valora positivamente este tipo de experiencias, su aprovechamiento es marginal si, de algún modo, no se establecen dinámicas de uso desde la propias clases presenciales y, sobre todo, si no se incorpora de algún modo al proceso de evaluación.

Referencias

- [1] X. Bohigas, X. Jaén y M. Novell, “Applets en la Enseñanza de la Física”, *Enseñanza de las Ciencias*, 2003, 21 (3).
- [2] Applets educativos de la Universidad de Búfalo (EEUU), accesibles en: <http://jas.eng.buffalo.edu/>