

EGB-Virtual: el juego didáctico en el aprendizaje

Francisco J. Díaz
LINTI – Facultad de
Informática
La Plata, Buenos Aires,
Argentina.
jdiaz@unlp.edu.ar

Claudia A. Queiruga
LINTI – Facultad de
Informática
La Plata, Buenos Aires,
Argentina.
claudiaq@info.unlp.edu.ar

Laura A. Fava
LINTI – Facultad de
Informática
La Plata, Buenos Aires,
Argentina.
lfava@info.unlp.edu.ar

Palabras Claves

Aprender jugando, juego didáctico, EGB (Educación General Básica), interfaces visuales, técnicas de interacción, JAVA, *applets*, *servlets*, comunicación *applets* y *servlets*, JDBC (Java DataBase Connectivity).

Resumen

EGB-Virtual es un proyecto concebido desde la cátedra *Laboratorio de Software* de la Facultad de Informática de la UNLP y cuyo objetivo es realizar un aporte desde la Informática a la innovación pedagógica aplicada al nivel inicial y primer ciclo de EGB (Educación General Básica). El juego es un vínculo muy importante en el proceso de aprendizaje. Los niños aprenden mediante diferentes tipos de juegos. **EGB-Virtual** es una aplicación web centrada en *applets* JAVA 2 [Ref. 1] que implementa mediante animaciones, simulaciones, imágenes y otros recursos visuales, gestuales y de voz, **juegos didácticos**, para ser usados por niños en las escuelas de nivel EGB.

Como docentes universitarios, consideramos importante motivar a nuestros alumnos (avanzados en la carrera de Informática) en el desarrollo de proyectos, que involucren desafíos como la construcción de software en donde aspectos como la usabilidad, facilidad de uso y objetivos pedagógicos son centrales.

El presente artículo describe **EGB-Virtual**, tanto desde el punto de vista del contenido o juegos didácticos que provee, la forma de acceso, mecanismos de visualización e interacción implementados, así también como cuestiones arquitectónicas del software, que lo hacen altamente reusable y escalable.

EGB-Virtual: el juego didáctico en el aprendizaje

INTRODUCCION

A partir de los resultados obtenidos con el VJ-Lab [Ref. 2] hemos decidido continuar explorando las técnicas de aprendizaje basadas en juego, tomando como escenario la escuela EGB.

Laboratorio de Software es una materia del cuarto año de la currícula de Licenciatura en Informática¹, en la que se enseña a programar en el lenguaje JAVA focalizándose en el desarrollo de *applets* [Ref. 3] [Ref. 1]. A partir del año 2002, hemos diseñado juegos de computadora didácticos para niños del primer ciclo de EGB, altamente interactivos, icónicos, y que abarcan todas las áreas de aprendizaje de este ciclo.

Como trabajo final de la materia, los alumnos desarrollaron *applets* implementando los juegos diseñados por la cátedra. A partir de estas *applets*, se construyó el **EGB-Virtual** [Ref. 4], un sitio web diseñado especialmente para reforzar y estimular el aprendizaje lúdico.

Este artículo está organizado de la siguiente manera: en primer término, se analiza el rol del juego en el aprendizaje, en segundo término el **EGB-Virtual**, sus formas de acceso y las características de las *applets* que lo conforman, luego la arquitectura de la aplicación web y finalmente las conclusiones.

EL JUEGO EN EL APRENDIZAJE

Es sabido que existe una estrecha relación entre el juego y el aprendizaje. Diversos autores trabajan de una u otra manera en esta temática, encontrando diferentes niveles de relación y vinculación [Ref. 5] [Ref. 6] [Ref. 7] [Ref. 8]. Reichel [Ref. 9] por ejemplo, describe tres niveles de aprendizaje: la *curiosidad*, a partir de la cual se desarrollan los primeros pasos para aprender. El *deseo de investigar*, que lleva a seguir aprendiendo en forma espontánea, y finalmente está la *pulsión lúdica*, que impulsa a actuar más allá de la sola percepción de lo que lo rodea.

Al comenzar un juego, un niño raramente se propone lograr un determinado aprendizaje. Podrá tener como objetivo construir algo, ganar una partida, alcanzar alguna meta. Podría ocurrir que estas cosas no se logren y que sin embargo no sea causa de frustración ni interrupción del juego. El propio juego lo invita a seguir intentándolo.

Si bien en el nivel inicial, el juego ocupa un papel importante, en los primeros años de EGB, comienza a quedar relegado a espacios y tiempo de recreo. Con relación a esto, King [Ref.10] describe tres niveles de juego: el *instrumental o didáctico*, utilizados con el objetivo de enseñanza; el juego *recreativo*, que es el juego libre que ocurre en el recreo y el juego *ilícito* que ocurre a espaldas del docente. A pesar de que hay una amplia gama de bibliografía con propuestas lúdicas para la enseñanza, la implementación de las mismas está lejos de ser predominante y de ser *modus operandis* en este nivel [Ref. 11].

A medida que se avanza en los diferentes niveles del sistema educativo, el juego está más ausente, encontrando cada vez más restricciones para su ingreso en el sistema educativo. En el nivel Polimodal, se observan actividades lúdicas sólo en las horas de Educación Física, juegos que en líneas generales, son de alto nivel competitivo, quedando frecuentemente afuera el juego creativo.

En el nivel Terciario y Universitario, el juego ya no forma parte de la currícula. Sólo en algunas carreras pedagógicas, psicológicas y humanísticas en general aparece en alguna materia como parte

¹ Materia obligatoria de la Facultad de Informática de la UNLP

del programa, limitado en general, a su estudio teórico, con algún trabajo práctico de observación de juego.

Los psicólogos, psicopedagogos, fonoaudiólogos y docentes de diversas áreas, tendrán seguramente que implementar actividades lúdicas desempeñando su profesión, pero al no tener experiencias lúdicas en su trayectoria académica, no les resultará fácil la utilización del *juego en la práctica*.

En algunas instituciones educativas, el juego (didáctico, recreativo y libre) pasa a ser una expresión admitida sólo para determinados lugares y momentos (recreo) o para algunas actividades particulares (en el aula).

En otros casos se admite o se permite únicamente el *juego didáctico*, que en ocasiones es tan didáctico que de juego queda poco. Algunos docentes argumentan que el juego por jugar no es admisible en dicho ámbito, debiendo tener siempre un contenido pedagógico. Desde esta perspectiva, el recreo y otros espacios de expresión libre son desvalorizados.

El juego en la institución educativa, lejos de ser considerado una posibilidad de aprendizaje, en la práctica (aunque desde el discurso sea lo contrario) está casi ausente. **EGB-Virtual**, es una propuesta lúdica, de tipo didáctica, para la enseñanza de las diferentes áreas de nivel EGB. En el próximo punto se detallan las características funcionales y operativas del mismo.

EGB-VIRTUAL: APRENDER JUGANDO

EGB-Virtual es una aplicación web, que se accede a través de un *browser* de Internet y que ofrece a los alumnos y docentes de EGB, la posibilidad de interactuar con *juegos pedagógicos*.

La interfaz de usuario de **EGB-Virtual** es gráfica, icónica, usa un diálogo visual e infantil para presentar los contenidos, es de fácil acceso y utiliza técnicas de interacción acordes a un niño.

Modos de Operación

La aplicación provee dos modos de operación: **Autor** y **Alumno**.

El **modo Autor** se diseñó para que el docente pueda crear contenidos (nuevos *juegos didácticos*). Permite crear y/o configurar ejercicios y evaluaciones. La creación varía de acuerdo al tipo de juego, por ejemplo si se habla de una Sopa de Letras se podrían elegir entre juegos de palabras existentes o cargar un nuevo conjunto de familias de palabras; si se enseña sobre animales se podría seleccionar de una lista clasificada por hábitat, por forma de reproducción, de alimentación, etc. .

En el **modo Alumno**, el niño dispondrá de **demonstraciones** que ayudarán a entender como interactuar con el ejercicio haciendo uso de las herramientas de la interfaz de usuario y de **juegos**, en los que el niño podrá interactuar directamente.

Cada juego provee un test, a modo autoevaluativo, con dos modalidades de corrección: por demanda y automática. En ambos casos, se le muestra al niño los aciertos y errores en forma visual y/o mediante voz.

Estructura de EGB-Virtual

El **EGB-Virtual**, está accesible desde la url: <http://www.linti.unlp.edu.ar/vj-lab2/pagina/>. La Figura 1, muestra la página principal.

A la izquierda de la página se encuentra una botonera que permite acceder a la descripción de la aplicación, a los juegos o actividades, al mapa del sitio y a otros sitios donde se proponen otros *juegos pedagógicos*.



Figura 1: Página Principal de EG-Virtual

A través del botón Juegos, se accede a todas las actividades, agrupadas por área y por ciclo lectivo.

Características de Implementación de los Juegos

EGB-Virtual es una aplicación web centrada en Java, especialmente enfocada en el desarrollo de *applets* en Java 2 [Ref. 12]. Las *applets* proveen la implementación de los juegos didácticos.

Las *applets* son componentes Java que se ejecutan del lado del cliente web y son adecuadas para escribir interfaces visuales. A continuación se describirán las características de implementación de las *applets* y de la calidad de la interfaz de usuario obtenida.

Las *applets* implementadas emplean gráficos, imágenes y animaciones para mostrar las entidades involucradas en los juegos, así también como las entradas y los resultados (o salidas). Todos los objetos intervinientes en el juego tienen asociada una representación visual, lo que atrae la atención del niño. La técnica de interacción más empleada es el *arrastre* (drag and drop), un gesto intuitivo hecho entre componentes *GUI* (Graphic User Interface) característico de las interfaces visuales.

La **Tecnología JAVA** usada es JFC [Ref. 13] que incorpora soporte para escribir interfaces de usuario visuales multiplataforma en Java.

Entre estas tecnologías, se encuentran: *AWT* (Abstract Windowing Toolkit), *Swing*, *Accesibility* y *Java 2D*. JFC provee un modelo de manejo de eventos basado en delegación, soporte para *clipboard* y un *framework* de interfaz de usuario. Las componentes *Swing* [Ref. 14] son componentes de

interfaz de usuario livianas que extienden las originales de AWT, agregando un conjunto de librerías de clases de interfaz de usuario gráficas completamente portables y que vienen incorporadas en la plataforma Java. Las componentes *Swing* mejoran el desarrollo de interfaces de usuario visuales, proveyendo de componentes de alta calidad, de plataforma neutral, altamente *customizables* y transparentes (de forma inorgánica). Adicionalmente, las componentes *Swing* son altamente *performantes*, debido al modelo de eventos que soporta y a las características de ser componentes livianas o *peerless*.

Las **Interfaces de Usuario** obtenidas, las hemos evaluado y corregido de acuerdo a los principios de Nielsen [Ref. 15]. Las *applets* cumplen cada uno de ellos de la siguiente forma:

1. **Diálogo natural y simple:** los diálogos no contienen información irrelevante. Toda la información aparece en orden natural y lógico. Se trató de poner la mínima cantidad de texto posible.
2. **Lenguaje en el que se expresa el usuario:** el diálogo fue expresado con palabras y frases acordes para niños. Se evitaron términos exclusivamente informáticos.
3. **Minimizar el uso de la memoria del usuario:** las instrucciones sobre el uso del sistema están siempre disponibles. Los niños no tienen que recordar desde un lugar determinado ningún tipo de información respecto al uso del sistema.
4. **Consistencia:** son consistentes en cuanto no se muestra la misma información de distintas maneras ni tampoco distinta información de la misma forma.
5. **Proveer feedback:** el sistema mantiene siempre informado al niño a cerca de la tarea que está desarrollando. Se usan imágenes, sonidos, textos para informarle al niño en cada momento sobre lo que está haciendo o hizo.
6. **Marcas para salir:** se proveen maneras claras de cancelar la tarea actual, retroceder al estado anterior o abandonar el juego.
7. **Proveer buenos mensajes de error:** proveen mensajes con lenguaje claro expresando el problema y soluciones alternativas.
8. **Prevenir, en lo posible de errores:** se trata de prevenir mediante un buen diseño e implementación la aparición de problemas.

Es importante destacar, que las *applets* utilizan una amplia gama de técnicas de interacción como **menús, manipulación directa y gestos, cajas de diálogo, animaciones e íconos** dependiendo de cada situación y necesidad.

A continuación, se describirán dos de las *applets* que implementan juegos didácticos: una de ellas utilizada para enseñar inglés y la otra para enseñar geografía Argentina.

Descripción de los Juegos Didácticos: Inglés y Geografía Argentina

Para mayor claridad, las pantallas capturadas que figuran debajo, muestran solamente el área de ejecución del *applet* con la que interactuarán los niños.

Las Figuras 2 y 3, muestran el *applet* que enseña inglés. En este caso, se trata de lecciones acerca de vocabulario en idioma inglés de temas que el niño elige y de construcción de oraciones usando el verbo TO BE.

El *applet* de la Figura 2, está provisto de un conjunto de vocabularios, entre ellos animales, vegetales, frutas, etc. Adicionalmente, el docente puede incorporar dinámicamente un vocabulario, creando lecciones nuevas. La interfaz de usuario de este *applet* es icónica, todas las entidades del juego están representadas mediante íconos. El *applet* muestra las palabras de la lección sobre

animales mediante imágenes; el niño interactúa con el juego *arrastrando* la palabra hasta la imagen del animal que corresponde.

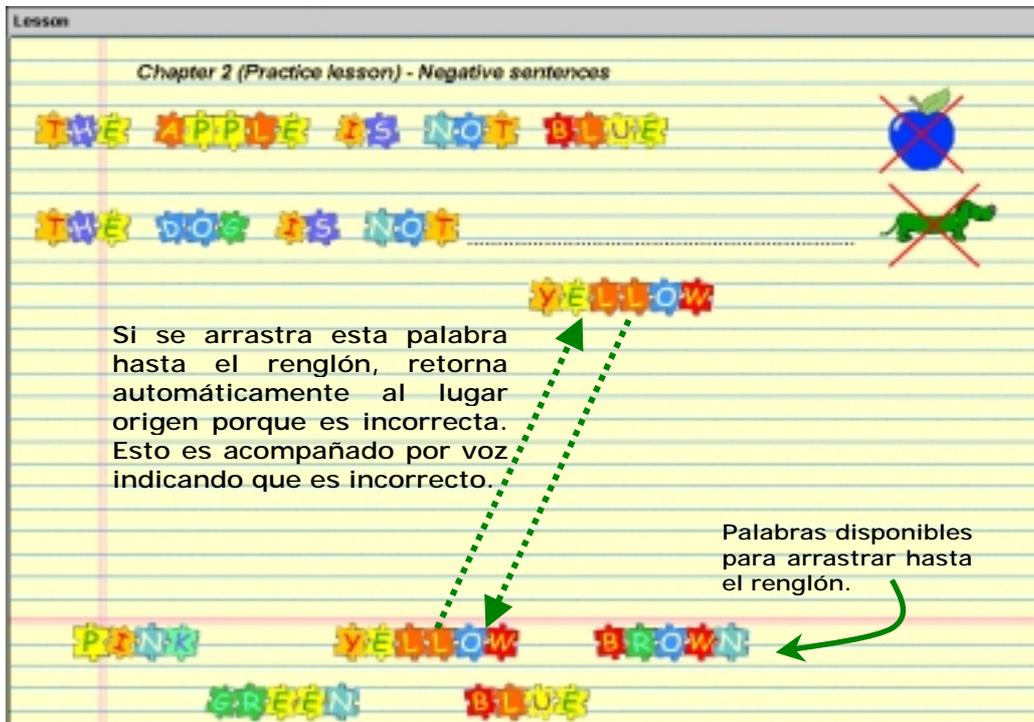


Figura 2 - Pantalla del Verbo TO BE

El *applet* de la Figura 3, muestra la interacción del niño, *arrastrando* palabras para construir oraciones correctas empleando diferentes conjugaciones del verbo TO BE..



Figura 3 – Pantalla de vocabulario de animales en inglés

La Figura 4, muestra el *applet* sobre geografía Argentina, que le permite a los niños aprender a reconocer la estructura geográfica de nuestro país, sus provincias, capitales, así como también su flora y fauna. Adicionalmente el *applet* posee evaluaciones sobre los contenidos enseñados y practicados.

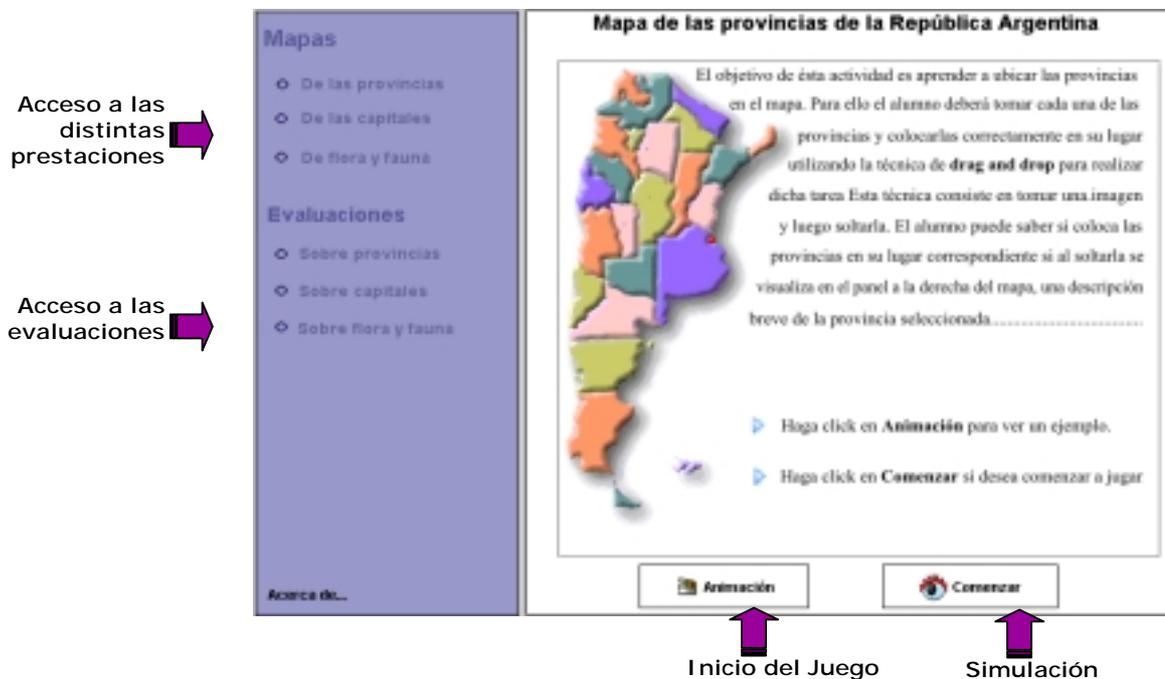


Figura 4: Pantalla principal del Applet sobre Geografía Argentina

Como todas las applets de **EGB-Virtual**, este *applet* cuenta con una simulación del juego, usada para indicarle visualmente al niño como mover las provincias hasta el lugar correspondiente.

Presionando en comenzar se accede el área de juego que se muestra en la Figura 5. La técnica de interacción usada para jugar es *drag & drop*. El niño *arrastra* la imagen de la provincia hasta la posición correspondiente dentro del mapa de la República Argentina. Cuando la provincia es situada en el lugar correcto, en el panel de la derecha aparece una imagen con información de la provincia y un ícono típico de la región. Este es el *feedback*, que el *applet* utiliza en este contexto, para indicarle al niño que ha ubicado correctamente la provincia.

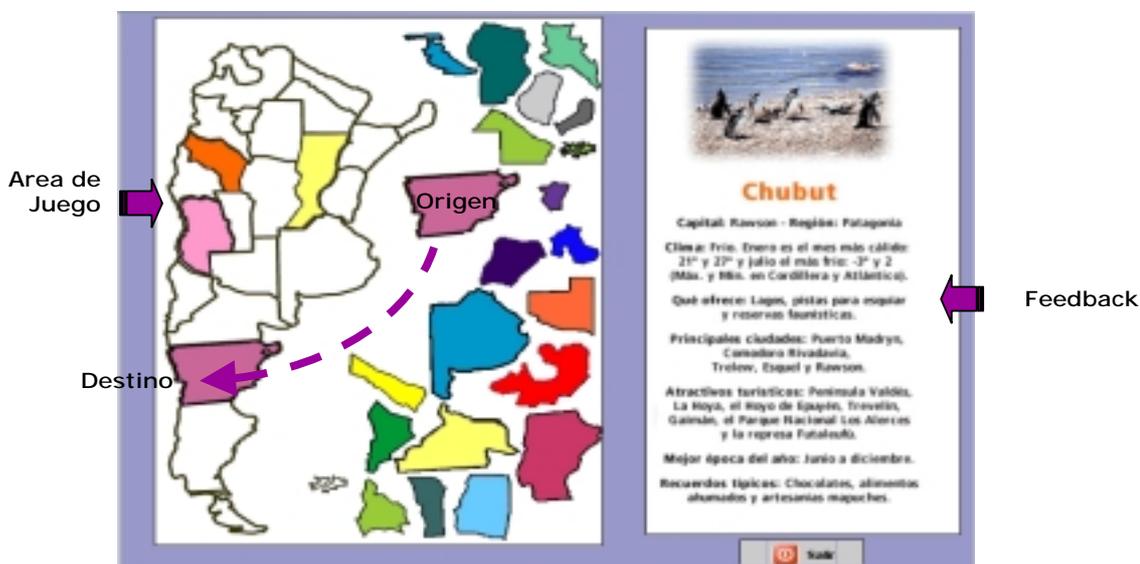


Figura 5: Área de Juego del applet sobre Geografía Argentina

De forma similar se puede jugar con las capitales y con la flora y fauna, arrastrando el nombre de las capitales o las regiones sobre el mapa.

Después de haber interactuado, el niño tiene la posibilidad de autoevaluarse a través de un test interactivo y fácil de usar. Como puede observarse en la Figura 6, las evaluaciones utilizan **cajas de diálogo** con opciones que permiten una única respuesta (*radio buttons*) y con imágenes de las provincias para que traten de reconocer sus formas.

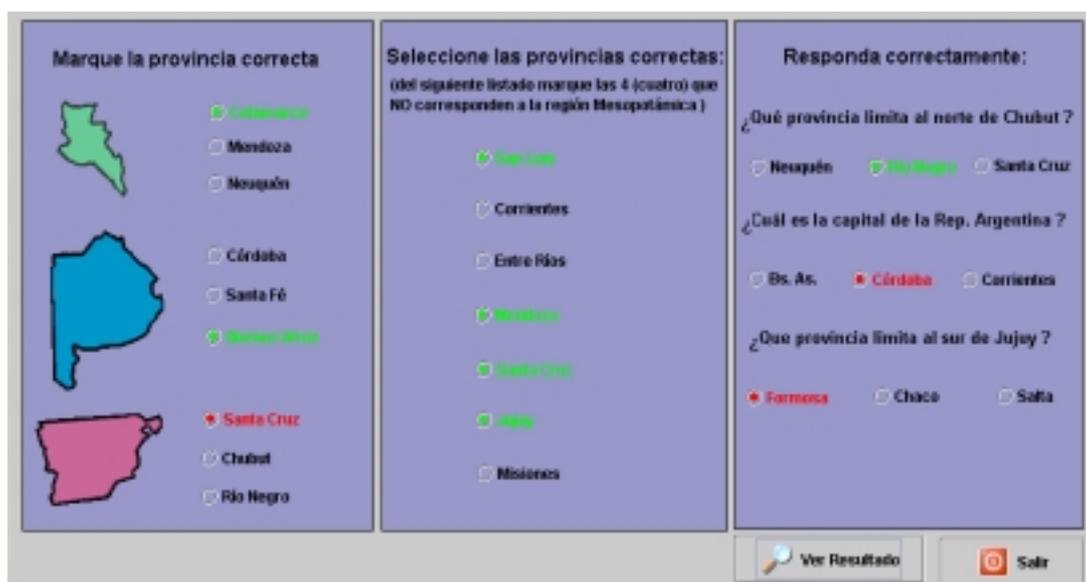
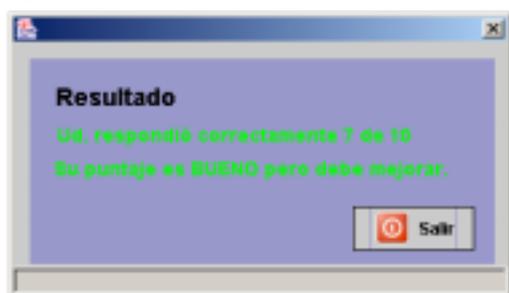


Figura 6: Área de Juego del applet sobre Geografía Argentina

Una vez finalizado el examen el niño, puede presionar el botón "Ver Resultado" para obtener el resultado de la evaluación. Como muestran las Figura 6 y 7, el **feedback** al alumno puede observarse de dos maneras: una pequeña ventana con el puntaje obtenido con una observación y un cambio de color sobre la misma ventana de diálogo del examen, marcando con color verde las respuestas correctas y con color rojo las incorrectas (Figura 6).



Esta ventana con el resultado varía de acuerdo al puntaje obtenido. Los mensajes (perfecto, regular, etc.) y sonidos (aplausos, sonidos alentadores, etc.) varían acordeamente.

Figura 7

ARQUITECTURA DEL EGB-Virtual

EGB-Virtual es una aplicación web basada en tecnologías JAVA, que responde a una arquitectura de software de tres capas, compuesta básicamente por *applets*, *servlets* y la base de datos. Las *applets* proveen la implementación de los juegos didácticos, obteniendo interfaces de usuarios de alta calidad, mientras que los *servlets* proveen mecanismos altamente eficientes para manejar requerimientos sobre un *servidor web*, fundamentalmente el acceso a la base de datos.

La clave de esta arquitectura es la comunicación entre *applets* del lado del cliente web (*web browser*) y *servlets* del lado del *servidor web*.

Debido a las limitaciones impuestas a las *applets* por el modelo de seguridad de los clientes web, obtener datos de entrada/salida de *applets* no es directo.

Las *applets* se descargan desde un *servidor web* y se ejecutan en el cliente web. Para prevenir la proliferación de *applets* maliciosas se han establecido las siguientes restricciones de seguridad:

- Las *applets* pueden comunicarse únicamente con el *servidor web* del que se descargaron.
- Las *applets* no pueden leer ni escribir en los discos locales del cliente web.
- Las *applets* no pueden ejecutar programas.
- Las *applets* no pueden examinar propiedades del ambiente local (cliente web).

Debido a estas restricciones se deben emplear estrategias especiales para intercambiar información con *applets*.

Los *servlets* son componentes web que se ejecutan en el entorno de un *servidor web*, manejan los requerimientos web y retornan datos o páginas HTML dinámicas al cliente web. Los *servlets* pueden acceder a la base de datos, efectuar cálculos, comunicarse con otras componentes web (por ejemplo con Java ServerPages) o empresariales (como Enterprise Java Beans) de manera muy eficiente.

A diferencia de las *applets*, los *servlets* no tienen restricciones de seguridad y se ejecutan en el *servidor web*.

La Figura 8, ilustra la arquitectura del **EGB-VIRTUAL**.

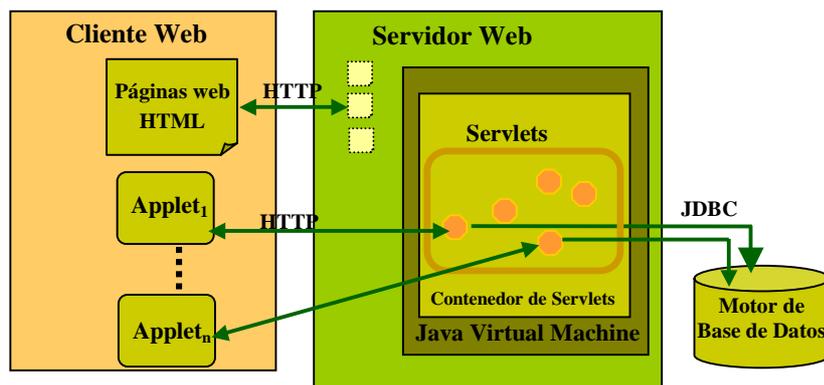


Figura 8. Arquitectura del EGB-VIRTUAL

La implementación de **EGB-Virtual** está basada en las siguientes componentes de software JAVA 2: *applets*, *servlets* 2.3 [Ref. 16] y *JDBC* 2.0 [Ref. 17].

JDBC es una API (Application Programming Interface) que provee conectividad con la capa de datos desde programas JAVA. JDBC permite acceder a una amplia variedad de bases de datos de una manera uniforme, independiente de la plataforma, ocultando las características propietarias de los diferentes motores de bases de datos.

Para poner en producción **EGB-Virtual** hemos utilizado dos softwares *open sources*: Apache y Tomcat [Ref. 18] [Ref.19] [Ref.20]. Apache es uno de los *servidores web* más ampliamente usado por su funcionalidad, confiabilidad y eficiencia. Tomcat es un contenedor de *servlets* y también funciona como un *servidor web*. Apache y Tomcat pueden ponerse en funcionamiento integradamente, permitiendo ejecutar aplicaciones que contienen componentes web escritas en JAVA. **EGB-Virtual** se ejecuta sobre Tomcat 4.0.

CONCLUSIONES

El presente trabajo confirma las ventajas de la enseñanza de JAVA, tanto en lo que hace a poder focalizar la construcción de objetos de software concretos con un usuario genérico en mente, como por el de permitir establecer un conjunto de objetivos de desarrollo y validarlos explícitamente en un entorno concreto.

La experiencia con los alumnos de la carrera de Informática ha sido muy positiva, los estudiantes se han sentido muy motivados en el desarrollo de los *juegos* al conocer que el software que estaban construyendo sería usando en una institución educativa.

Actualmente **EGB-Virtual** está en etapa de testeo con alumnos de nivel inicial y primer nivel, de la escuela Anexa “Joaquín V. González”(colegio de la Universidad Nacional de La Plata), con el objetivo de que sea evaluado tanto desde el punto de vista de funcional, de *HCI* (Human Computer Interaction) como su impacto en el aprendizaje del niño.

Por otro lado, nuestros los alumnos de la tuvieron la posibilidad de realizar una experiencia práctica con tecnologías JAVA (*applets* comunicados con *servlets* para acceder a los datos) en un tipo de aplicación no convencional, en donde los aspectos de interacción y simplicidad de uso son centrales.

AGRADECIMIENTOS

Nuestro agradecimiento a todos los alumnos de *Laboratorio de Software* que desarrollaron las *applets* y en especial al Lic. José Ferreyra, pasante del LINTI quién ha colaborado con la integración de herramientas y en el diseño y desarrollo del sitio **EGB-Virtual**.

REFERENCIAS

[Ref. 1] <http://java.sun.com/docs/books/tutorial/applet/index.html>

[Ref. 2] Javier Díaz, Claudia Queiruga, Laura Fava, Claudia Villar. *A Virtual Java-simulation Lab for Computer Science Students*, Webnet 2000, Texas, EEUU.

[Ref. 3] <http://java.sun.com/applets/>

[Ref. 4] “*EGB-Virtual*”, <http://www.linti.unlp.edu.ar/vj-lab2/pagina>

[Ref. 5] Flitner, Andreas. *Spielen-Lernen*, 2002.

[Ref. 6] Goñi Radrizzani, Gonzalez, A. *El niño y el juego*, Buenos Aires, Nueva Visión, 1987

[Ref. 7] Scheuerl, H. *Untersuchungen über sein Wesen, seine pädagogischen Möglichkeiten und Grenzen*, Weinheim, 1990.

[Ref. 8] Walter G. *Spel und Spielpraxis in der Grundschule*, Donaworth, Auer Verlang, 1993.

[Ref. 9] Reichel, G. *Lebending statt brav*, Munster, Ökotopia,1995.

[Ref. 10] King, N. *Play and early childhood development*, New York, Longman, 1987.

[Ref. 11] María Regina Ofele. *Juego, Aprendizaje e Intituciones Educativas*, 2000.

[Ref. 12] <http://java.sun.com/j2se/> y <http://java.sun.com/j2ee/>

[Ref. 13] JFC (Java Foundation Classes): <http://java.sun.com/products/jfc/>

[Ref. 14] Swing: <http://java.sun.com/developer/Books/swing/>

[Ref. 15] Sitio web de Jacob Nielsen. <http://www.useit.com>, 2004.

[Ref. 16] Ayers D., Bergsten H.. *Professional Java ServerProgramming*, WROX, 1999.

[Ref. 17] *JDBC Database Access*, <http://java.sun.com/docs/books/tutorial/jdbc/>

[Ref. 18] *A Business Case Study of Open Source Software*

http://www.mitre.org/support/papers/tech_papers_01/kenwood_software/kenwood_software.pdf

[Ref. 19] David A. Wheeler. *Why Open Source Software / Free Software (OSS/FS)? Look at the Numbers!*, http://www.dwheeler.com/oss_fs_why.html

[Ref. 20] *Apache avoids most security woes*,

<http://www.zdnet.com/eweek/stories/general/0,11011,2792860,00.html>