

**UNIVERSIDAD:** Universidad Nacional de La Plata / Facultad de Informática / Instituto de Investigación en Informática LIDI (III-LIDI).

**OTROS TEMAS:** Redes Académicas.

**TÍTULO DEL TRABAJO:** ESTRATEGIAS METODOLÓGICAS PARA FLEXIBILIZAR EL DISEÑO DE SISTEMAS DE E-LEARNING.

**AUTOR(ES):** Eduardo Ibáñez y Luciano Iglesias.

**E-MAIL DE LOS AUTORES:** eibanez@lidi.unlp.edu.ar, li@info.unlp.edu.ar

**PALABRAS CLAVES:** ingeniería web, e-learning, arquitectura MVC.

## **INTRODUCCIÓN**

El e-learning (electronic learning) refiere al aprendizaje a través de medios electrónicos, en particular se lo ha vinculado con el entorno de Internet [1][2]. Los entornos virtuales (en la Web) desarrollados con el fin de acompañar y mediar los procesos de enseñar y aprender han crecido a gran velocidad en los últimos años, debido a las posibilidades que ofrecen para la comunidad educativa. Desde el punto de vista informático son aplicaciones vinculadas con la Ingeniería Web [3].

El objetivo de este trabajo es describir la metodología con la que se viene desarrollando el entorno virtual de enseñanza y aprendizaje WebINFO (plataforma de e-learning), con el fin de resolver los problemas relacionados con el diseño de aplicaciones Web. En general se observan problemas en la construcción de estas aplicaciones, ya que se entremezclan componentes referidos a la lógica de la aplicación con aquellos referidos a la lógica de los aspectos visuales y de presentación.

Cabe aclarar que WebINFO está siendo desarrollado en el III-LIDI de la Facultad de Informática de la Universidad Nacional de La Plata, y se utiliza en el Programa de Educación a Distancia de esta universidad [4]. Es un sistema Web interactivo de carácter pedagógico que permite a los docentes el armado de sus cursos, personalizándolos de acuerdo a las características de los destinatarios y el contexto. Por otra parte, posibilita dar soporte a todo el proceso educativo a través de la mediación utilizando diferentes herramientas que provee el ambiente. Para el desarrollo de esta aplicación fue y es fundamental el trabajo de un equipo interdisciplinario que involucre el análisis de elementos pedagógicos, tecnológicos y de comunicación, y por ello resulta importante la estrategia de diseño que se planteará en este trabajo.

## **PROBLEMÁTICA**

Desde el origen de las aplicaciones gráficas se ha entremezclado aquello relativo a la lógica de la aplicación y la lógica correspondiente a aspectos visuales o de presentación. El

modelo MVC (Model-view-controller) surgió como una arquitectura de software que separa una aplicación en tres partes: el modelo de datos, la interfaz de usuario y la lógica de control, de forma tal que la modificación de una de las componentes tenga un mínimo impacto sobre las otras [5] [6].

En el contexto de las aplicaciones Web, el escenario típico es tener la lógica de la aplicación (escrita en lenguajes como ASP, PHP, ASP.NET, ColdFusion, etc.) y la lógica visual (por ejemplo en html) mezclados en el mismo módulo. Esta metodología de trabajo trae asociada una serie de problemas:

- \*) No permite tener para una misma aplicación más de una forma de visualización (al menos sin repetir el código ya realizado)

- \*) No permite que los diseñadores trabajen en forma directa con el código visual sin la intervención de los programadores (entonces esto perjudicaría el trabajo interdisciplinario).

- \*) No permite el trabajo simultaneo de los programadores y los diseñadores

- \*) Dificulta la interpretación y el mantenimiento de las aplicaciones, ya que suelen generarse códigos muy extensos, donde se mezclan diversos aspectos referidos a la construcción de estos sistemas

- \*) Puede ocurrir que el diseñador modifique accidentalmente el código de la aplicación introduciendo errores

Para llevar adelante una metodología de trabajo que reformule los problemas de diseño mencionados con anterioridad, se está utilizando un motor de plantillas, cuyo objetivo es separar el contenido de los aspectos visuales de la aplicación. De esta forma se logra tener las tres partes bien constituidas como se indica en la arquitectura MVC: la base de datos y el código para tratar con la información almacenada en ella, las plantillas referidas a los aspectos de interfaz gráfica y visualización, y el código para atender a los eventos y acciones de los usuarios.

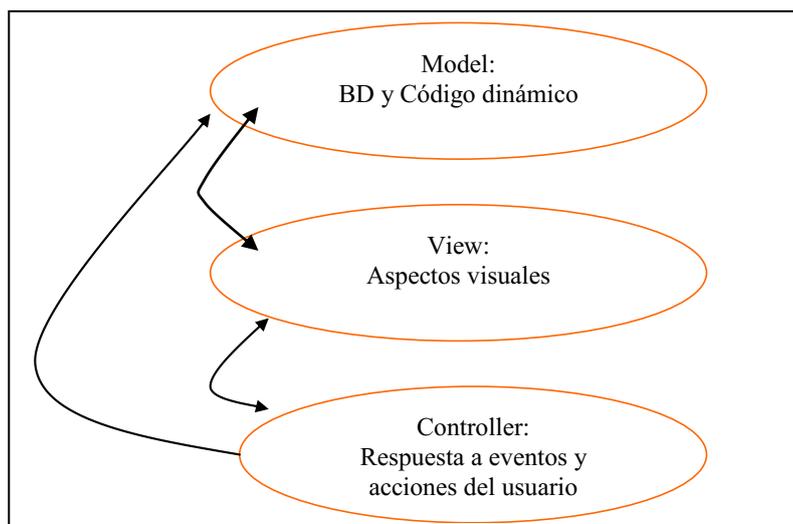


Figura 1. Aplicación Web con sus tres componentes

### **Componente de modelo de la aplicación**

El modelo encapsula el núcleo funcional de una aplicación, tiene relación con la lógica y el dominio de la información. La idea del MVC es lograr la independencia del modelo de los aspectos visuales o interfaz de la aplicación y del componente de control.

Para trabajar sobre esta componente en WebINFO, se utilizó un mecanismo de almacenamiento persistente (base de datos) y un método de comunicación con dicha base de datos a través de un “framework” de componentes php reusables. En particular se trabajó con Pear DB que es una capa de software que permite la abstracción del motor de base de datos utilizado [7]. De esta manera, se posibilita que WebINFO sea portable entre diferentes DBMS<sup>1</sup>, permitiendo que cada institución que haga uso de la aplicación puede decidir qué producto utilizar para almacenar los datos.

### **Componente de aspectos visuales de la aplicación**

La vista obtiene los datos del modelo y los presenta al usuario. La vista representa la salida de la aplicación. En general tienen acceso libre al modelo, pero no debería cambiar el estado del mismo.

En el caso de WebINFO, se ha decidido utilizar el motor de plantillas o templates para PHP llamado Smarty, que permite separar el contenido (datos) de la presentación (visualización) de una página Web [8].

Así se trabaja por un lado con plantillas que se componen de código estático (ejemplo html) y código dinámico, etiquetas de Smarty, que al momento de dar la respuesta al

<sup>1</sup> DBMS: Data Base Management System

requerimiento del cliente, son reemplazadas automáticamente por los datos retornados por el componente de modelo de la aplicación.

### **Componente referido al controlador de la aplicación**

Es el encargado de responder a eventos, usualmente acciones del usuario, y luego invoca a cambios en el modelo y probablemente en la visualización.

¿Cómo estamos trabajando al respecto en WebINFO? Al momento existe una librería de funciones javascript que pueden ser reutilizadas en cualquier punto de la aplicación. Se planifica en un futuro abordar el uso de XAjax que es una librería de clases php (de código abierto) que permite crear fácilmente poderosas aplicaciones Web basadas en Ajax. Las aplicaciones desarrolladas con XAjax pueden llamar a funciones del lado del servidor asincrónicamente, y actualizar el contenido sin recargar la página [9].

### **La propuesta metodológica de diseño y desarrollo de WebINFO**

La aplicación está siendo desarrollada por un grupo interdisciplinario conformado por programadores, analistas funcionales, un diseñador y una especialista en el tema pedagógico (dado que la aplicación requiere la mirada de un experto en estas temáticas). Además en el proceso se trabaja en experiencias con docentes y alumnos que participan aportando sus reflexiones acerca de la usabilidad de la aplicación.

La metodología de trabajo actual de acuerdo a los roles intervinientes se detalla a continuación y al modelo MVC planteado se describe en el siguiente párrafo.

Los analistas funcionales y la experta en educación son los encargados de estudiar los requerimientos de la aplicación, las necesidades a resolver, y determinar los circuitos funcionales requeridos para que los usuarios (docentes, alumnos y administradores) puedan encontrar criterios de calidad en la misma. Se trabaja también en el diseño de la base de datos. Esto está relacionado con el componente referido al Modelo de la aplicación.

Los programadores llevan adelante a partir de la especificación funcional el armado del código dinámico, y generan plantillas para que luego puedan ser transformadas por los diseñadores en elementos de comunicación visual.

Los diseñadores elaboran el proceso antes mencionado, generando un sistema visual coherente, y que permita presentar distintas imágenes del sistema de acuerdo al contexto en el que vaya a ser utilizado. Esto último está vinculado con el componente de aspectos visuales e interfaz que se mencionó anteriormente.

En los entornos virtuales de enseñanza y aprendizaje esto último, adquiere un especial matiz dado que se apunta a personalizar los espacios de interacción entre alumnos y docentes de acuerdo al contexto, y al perfil particular que poseen los alumnos. Por ejemplo, el uso de colores, tipografías, sistemas iconográficos, idioma, podrán cambiarse fácilmente de acuerdo a la institución y los destinatarios que utilicen la plataforma.

## **RESULTADOS OBTENIDOS**

Cómo resultado de este proceso se obtuvieron las siguientes mejoras respecto de la metodología de desarrollo:

\*) Se permitió a la aplicación la posibilidad de tener diferentes visualizaciones según el gusto y las necesidades de cada contexto en el que vaya a ser utilizado, con un costo mínimo en la programación. Podemos ver dos alternativas de visualización en la Figura 2 y Figura 3, que pueden ser seleccionadas por el usuario.

\*) Se permitió a los diseñadores, diseñar o rediseñar el sitio de forma totalmente autónoma

\*) Facilitó la realización de tareas de mantenimiento gracias a la separación en componentes independientes alcanzada en la aplicación

\*) Se independizaron las distintas partes del diseño (cabeceras, barra de navegación, área de datos, etc.) permitiendo así la posibilidad de cambiar en forma independiente cada parte del sitio con una única modificación

\*) Se logró extender la oferta del sistema a regiones con otros idiomas (internacionalización)

\*) Se introdujo la posibilidad de trabajar con diferentes motores de base de datos a través del uso de Pear DB.



Figura 2. Visualización Original del Ingreso a los Cursos

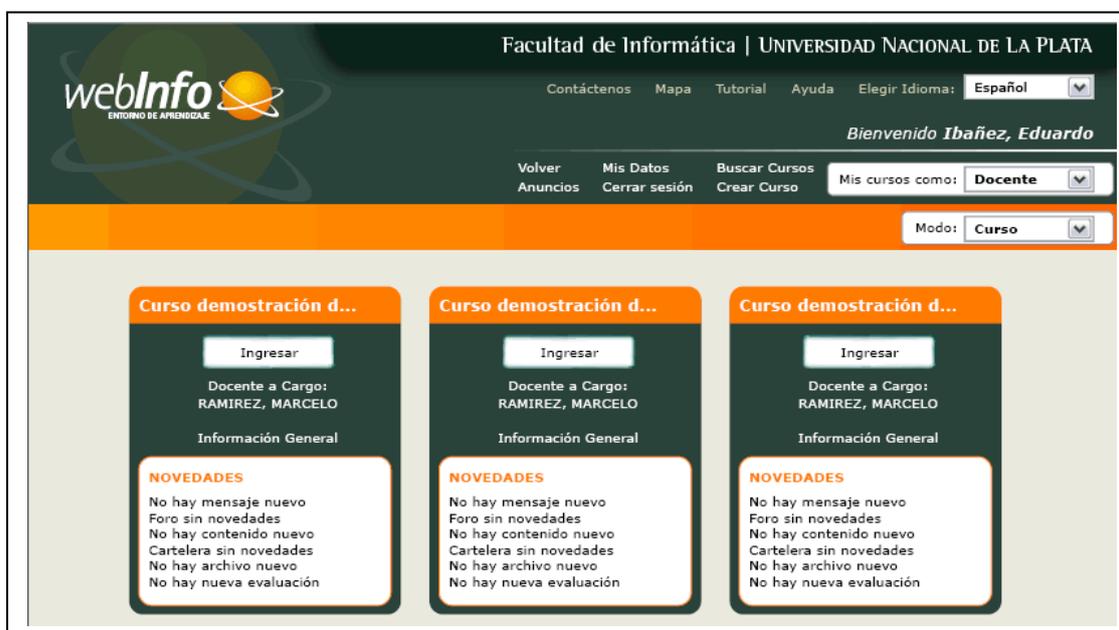


Figura 3. Visualización Alternativa del Ingreso a los Cursos

Cabe aclarar que la próxima versión de WebINFO incluirá todas estas mejoras, y se espera que los resultados obtenidos tengan un impacto positivo en la comunidad de docentes y alumnos que vienen utilizando WebINFO al momento.

## REFERENCIAS

- [1] Beer V. "The Web Learning Fieldbook: Using the World Wide Web to Build Workplace Learning Environments". San Francisco, California (USA): Jossey-Bass / Pfeiffer. Año 2000.
- [2] Chacón F. "Mind-Mapping for Web Instruction and Learning". Franciscan University of Steubenville. Año 2003.

- [3] A. Ginige - S. Murugesan, "Web Engineering: An Introduction" IEEE MultiMedia, vol. 8, no. 1, Jan.-Mar. 2001, pp 14-18, <http://computer.org/multimedia/mu2001/pdf/u1014.pdf>.
- [4] Sanz C., Gonzalez A., Zangara A., De Giusti A., "WebINFO, un entorno de aprendizaje web". EDUTEC 2004. Educar con Tecnologías: de lo excepcional a lo cotidiano. Barcelona, España. 17 a 19 de Noviembre, 2004. <http://edutec2004.lmi.ub.es/>
- [5] Yonglei Tao, "Component- vs. application-level MVC architecture". Frontiers in Education, 2002. FIE 2002. 32nd Annual. Volume 1, 6-9 Nov. 2002 Page(s):T2G-7 - T2G-10 vol.1 Digital Object Identifier 10.1109/FIE.2002.1157950
- [6] MVC Model - [http://www.phpwact.org/pattern/model\\_view\\_controller](http://www.phpwact.org/pattern/model_view_controller)
- [7] PEAR DB - <http://pear.php.net/package/DB>
- [8] Smarty - <http://smarty.php.net>
- [9] XAjax – <http://www.xajaxproject.org>
- [10] IEEE Digital Library
- [11] ACM Digital Library