

SMART-TEST: una aplicación de las TIC al proceso de evaluación de estudiantes universitarios

SMART-TEST: an application of the TIC to the process of evaluation of university students

Gómez D., Abdías¹, Portela, Leandra² y Olivios, Oscar³
Fundación Universitaria de San Gil –Unisangil, Facultad de Ciencias Naturales e Ingeniería
Programa de Ingeniería de Sistemas
Yopal, Colombia

agomez@unisangil.edu.co
oscarjavier104@hotmail.com
leandra124@hotmail.com

Fecha de Recepción: julio 11 de 2013
Fecha de Aceptación: septiembre 04 de 2014

Resumen— Un aspecto importante en todo proceso de enseñanza-aprendizaje es el seguimiento que pueda hacerse al mismo. La evaluación periódica es una estrategia ampliamente utilizada en la educación formal para el diagnóstico, la formación y la promoción. Junto a esto, desarrollar habilidades y capacidades acordes a las necesidades sociales y tecnológicas es una demanda de las competencias profesionales dadas por el contexto actual [1]. En este sentido, como respuesta a la necesidad observada de integrar las Tecnologías de la Información y Comunicación (TIC) a la educación y al mismo tiempo facilitar tanto al docente como al estudiante la participación en actividades evaluativas, se ha desarrollado SMART-TEST bajo el paradigma Extreme Programming (XP), que hace parte de los modelos de desarrollo ágil [2]. Es una aplicación desarrollada para la web, que permite al docente crear diferentes tipos de pruebas, en cuanto número, preguntas y tiempos. En este artículo de reflexión se presenta SMART-TEST como resultado de un desarrollo tecnológico orientado hacia las TIC y la educación universitaria. Se hace una propuesta de integración de una herramienta web como entorno virtual al proceso de evaluación de estudiantes universitarios en Unisangil.

Palabras clave— Entorno web, evaluación, tecnologías de la información y la comunicación, internet, smart-test.

Abstract- An important aspect in every teaching - learning process, is related to the follow-up. The periodic evaluation is a widely used strategy in formal education, for the diagnosis, training and promotion. Besides, developing skills and abilities related to the social and technological needs is a demand for the professional skills provided by the current context. In that sense, as a response to the observed need to integrate the Information and Communication Technologies (ICT) to education, and at the same time to facilitate the participation in evaluative activities both for the teacher and for

the student, SMART-TEST has been developed under the paradigm Extreme Programming (XP), which is part of the agile development models. This is an application developed for the web, allowing the teacher to create different types of tests, the number, type of questions and times. This article of reflection presents SMART-TEST as a result of a technological development oriented to ICT and higher education. It is a proposal for integration of a web tool as a virtual environment to the process assessment of university students at UNISANGIL.

Keywords- Web environment, evaluation, information technologies and communication, Internet, Smart-test.

I. INTRODUCCIÓN

SMART-TEST es una aplicación en ambiente web que provee herramientas de elaboración y presentación de evaluaciones cortas y parciales. El desarrollo de este ambiente virtual busca agilizar las tareas de diseño, elaboración, generación de resultados, estadísticas, análisis de los consolidados y resúmenes obtenidos. Por otra parte, se busca cambiar el ambiente tradicional en el cual el estudiante es evaluado, mediante la incorporación de las TIC al proceso educativo [3].

En este sentido, las TIC, incorporadas al proceso de evaluación de estudiantes universitarios, pueden constituirse en un elemento innovador y diferenciador respecto a las prácticas tradicionales. No cabe duda de que contar con una herramienta tecnológica “en línea” como entorno virtual para la evaluación de los estudiantes universitarios es un paso hacia la integración de las TIC a la educación. Esta

¹ Ingeniero de Sistemas, Docente tiempo completo, programa Ingeniería de Sistemas, Unisangil

² Ingeniero de Sistemas.

³ Ingeniero de Sistemas.

integración contribuirá con la disminución de tiempos de respuesta en las tareas de realimentación del proceso evaluativo.

Smart-Test es una aplicación dinámica que permite integrar diferentes tipos de pregunta, ya sea selección múltiple con única respuesta, o con múltiple respuesta, falso o verdadero, ordenar respuestas, entre otras. Esto asegura que el docente tenga un amplio abanico de oportunidades para el diseño de las pruebas evaluativas; además permitirá establecer diversas configuraciones por parte de los educadores, como número de preguntas por prueba, tiempo de ejecución (que es bastante importante al momento de presentarlas), o puntaje asignado para cada una de las preguntas, etc.

II. MATERIALES Y MÉTODOS

Smart-test fue concebida bajo la metodología de Extreme Programming (XP), que hace parte de las metodologías ágiles para el desarrollo de software.

La programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad.

En la figura 1 se observan las fases de XP. El ciclo de esta metodología permite una constante adaptación de los requerimientos, para agilizar la corrección de errores y facilitar entregas rápidas de pequeñas versiones, para que el cliente verifique el cumplimiento de los requisitos y se realicen las modificaciones necesarias.

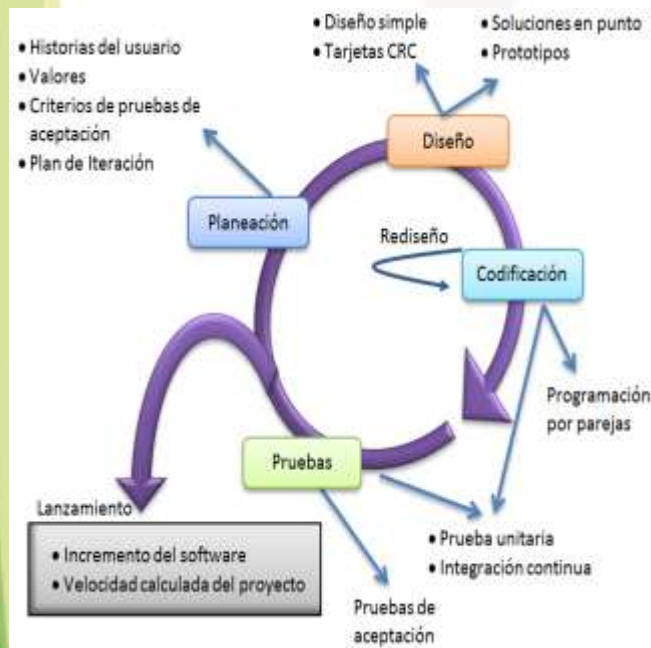


Fig. 1 Fases Extreme Programming.

Siendo Smart-test una aplicación en ambiente web, su funcionamiento está basado en los distintos protocolos y

tecnologías existentes para la comunicación a través de internet, como a continuación se describe.

A. HTTP. Hipertext Transfer Protocol (en español Protocolo de Transferencia de Hipertexto) es el protocolo utilizado en cada de las comunicaciones y transacciones sucedidas en Internet, orientado a transacciones entre un cliente y un servidor [4].

B. Servidor web. También conocido como servidor HTTP, es un programa informático que maneja las peticiones de los clientes y se encarga de servir las páginas web solicitadas.

C. Navegador web. Es una aplicación que actúa como cliente, solicitando contenido al servidor web e interpretándolo para su visualización [5].

Existen tecnologías que ayudan al enriquecimiento del contenido en internet, brindándole dinamismo y a su vez incrementando su uso en distintos sectores. Es el caso de los lenguajes de script del lado del cliente, los cuales se ejecutan en el navegador web y brindan interfaces agradables al momento de mostrar el contenido web, así como los lenguajes de programación del lado del servidor, los cuales se hospedan en él y tienen objetivo brindar dinamismo al contenido permitiendo interactuar y comunicarse con otros servicios disponibles (bases de datos, archivos).

SMART-TEST fue desarrollada aplicando estas tecnologías para su funcionamiento; todas las herramientas utilizadas son de código abierto o software libre, por tanto su uso no genera costo por licenciamiento.

El servidor web empleado fue Apache [6]; en la actualidad es el servidor más común para páginas web debido a su rendimiento y fácil implementación en cualquier plataforma. Su principal función es servir contenido en formato HTML (Hipertext Markup Language, en español Lenguaje de Marcado de Hipertexto), el cual es interpretado por el navegador web y mostrado en pantalla.

Con la intención de almacenar y gestionar la información ingresada al sistema, se empleó el motor de bases de datos Postgresql [7], que por sus características y licenciamiento se adapta perfectamente para este caso.

Para gestionar la base de datos y realizar la lógica de negocios se utilizó PHP como lenguaje de programación del lado del servidor, el cual se encarga de recibir la información del usuario (docentes y estudiantes) y almacenarla por medio de transacciones SQL.

SMART-TEST se caracteriza por su alto contenido de lenguaje de script del lado del cliente, dado que toda su interfaz se asemeja a una aplicación de escritorio.

Todas las peticiones al servidor se realizan en modo asíncrono mediante la utilización de peticiones XMLHttpRequest generadas por Javascript, el cual es un lenguaje de programación del lado del cliente, lo cual le ofrece ventajas a la aplicación como la validación de datos de formularios, diseño dinámico de la interfaz de usuario; también pudo ser utilizado para tareas tan simples como mostrar cuadros de diálogo.

Debido a su gran capacidad de mejorar la interfaz de usuario y con la ayuda de librerías encargadas de facilitar la programación (jQuery, Mootools, entre otras), Javascript se convirtió en una tecnología imprescindible al momento de crear esta aplicación, por su característica de estar diseñada para entorno web. Incluso esta herramienta puede ser útil en el diseño de aplicaciones de escritorio, como es el caso de Windows 8 que permite correr aplicaciones creadas en html5, Javascript y CSS [8].

CSS, también conocidas como Hojas de Estilos en Cascada, son las encargadas de personalizar las interfaces mostradas por el navegador, añadir colores, sombras y movimientos, entre otros.

A continuación en la figura 2 se muestra cómo la aplicación interactúa con los distintos servicios y tecnologías mencionadas.

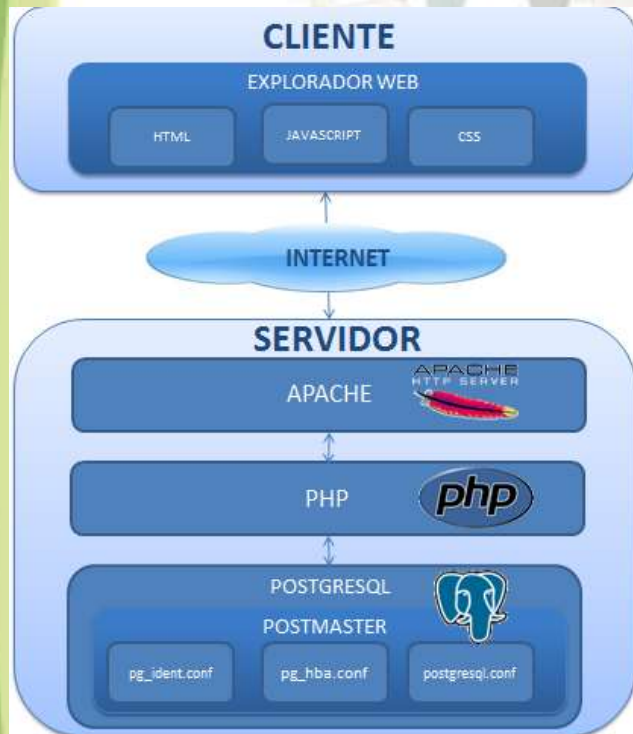


Fig. 2 Arquitectura del sistema.

III. FASES DE DESARROLLO

Como la metodología (XP) de desarrollo lo propone, SMART-TEST se desarrolló por fases [9].

A. Fase de Exploración

Comprendió la captura de requerimientos mediante el formato propuesto por la metodología denominado historias de usuario. Allí el cliente plasmó con sus propias palabras los requerimientos solicitados para la realización de sistema de información. La tabla 1 presenta la historia de usuario “Agregar lectura”.

TABLA 1. HISTORIA DE USUARIO

Historia de Usuario	
Número: 09	Usuario: Secretaria
Nombre historia: Agregar lectura	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Oscar Javier Olivos	
Descripción: Las lecturas son parte importante de las preguntas, por eso deberá existir una ventana dedicada a su creación. El aplicativo debe permitir subir al servidor en formato PDF el contenido de las lecturas.	
Observaciones:	

B. Planeación de las entregas e interacciones

Se estableció la prioridad de cada una de las 18 historias de usuario que se establecieron en la etapa preliminar; se estimó el esfuerzo necesario para realizar la actividad y se asignó un responsable para cada una de las 15 tareas de ingeniería, que se ejecutaron en cuatro iteraciones desarrolladas durante 8 meses. La tabla 2 muestra la historia de usuario “Planificación y diseño de la arquitectura del sistema” planificada y ejecutada en la primera iteración.

TABLA 2. TAREA DE INGENIERÍA

Tarea de Ingeniería	
Número: 01	Historia de usuario: No aplica
Nombre historia: Planificación y diseño de la Arquitectura del sistema	
Tipo de Tarea: Diseño	Puntos estimados: 4
Fecha de Inicio: Feb. 13 de 2013	Fecha de fin: Marzo 13 de 2013
Programador responsable: Oscar Javier Olivos –Anyi Leandra Portela	
Descripción: En reunión con el equipo de desarrollo se plantean las distintas soluciones posibles para dar cumplimiento a cada una de las historias de usuario. Creación de los distintos diagramas que definan la estructura y funcionalidad del programa. Se establecerán los distintos módulos en los que estará dividido el sistema. El modulo administrador tendrá control total del aplicativo, el módulo de docente será responsable de las pruebas y preguntas y el módulo estudiante contendrá las pruebas.	

C. Producción

Se asignaron roles y responsabilidades a cada uno de los integrantes del grupo de desarrollo de acuerdo a los roles

propuestos por la metodología. Se ejecutaron las tareas de ingeniería asignadas e hicieron pruebas de aceptación de las mismas. La tabla 3 ilustra un caso de prueba de aceptación realizado a la aplicación.

TABLA 3. PRUEBA DE ACEPTACIÓN

CASOS DE PRUEBAS DE ACEPTACIÓN	
Número: 01	Historia de usuario: 05
Nombre:	Inicio de sesión.
Descripción:	Esta prueba busca verificar el cumplimiento del requerimiento funcional que busca que los usuarios de sistema pueden ingresar de forma correcta a los distintos módulos del sistema.
Condiciones de ejecución:	El usuario debe estar registrado en el sistema. El usuario debe estar en la página inicial del aplicativo.
Entrada:	Se ingresa un nombre de usuario que exista en la base de datos. Se ingresa la combinación de caracteres que sirvan de contraseña. Pulsar el botón ingresar.
Resultado esperado:	El usuario debe ingresar a sesión.
Evaluación:	Prueba satisfactoria.

IV. RESULTADO

Smart-test fue pensado para brindar una interfaz amigable y cómoda al usuario final.

La presentación de los diferentes menús se hizo por pestañas que representan los módulos que integran el aplicativo. Cada pestaña contiene un área de trabajo que muestra su contenido y su barra de herramientas para interactuar con el módulo, como se indica en la figura 3.

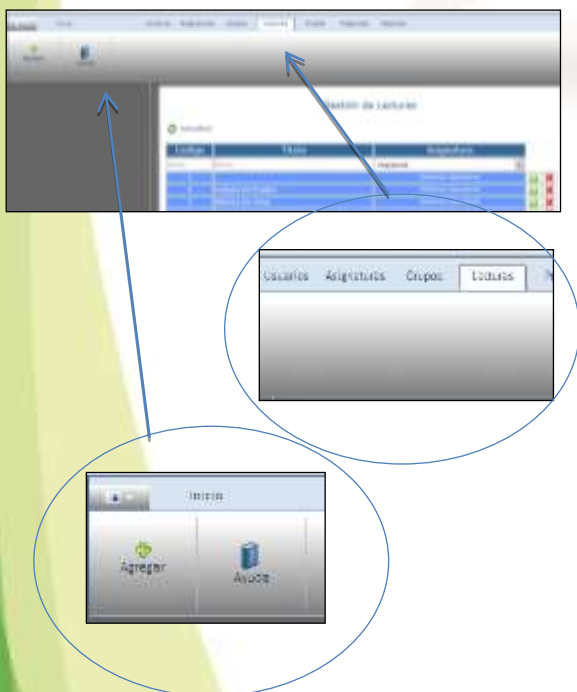


Fig. 3 Menú inicial SMART-TEST

El sistema de información cuenta con 7 módulos, los cuales representan las diferentes actividades que se deben realizar para llevar a cabo la creación y presentación de las pruebas. Cada módulo puede estar disponible dependiendo del rol que desempeñe en el sistema de información. Existen 3 roles para interactuar con el software, los cuales se describen a continuación, enumerando sus tareas.

A. Estudiante. Creado para permitir al estudiante ingresar y presentar o consultar las pruebas asignadas. Para poder ingresar, debe haber sido creado previamente en el sistema.

B. Docente. Con permisos para crear preguntas, lecturas, pruebas y revisar los reportes creados, además de modificar o eliminar los registros establecidos por él mismo.

C. Administrador. Además de tener acceso a los módulos de los roles anteriores, puede crear y modificar usuarios, asignaturas y grupos. Es el encargado de permitir o no el acceso al sistema por parte de los estudiantes y docentes.

Cada uno de los módulos que comprenden el aplicativo cuenta con su respectivo manual de usuario, en donde se encuentran las instrucciones para su correcto uso.

V. PUESTA EN MARCHA

Para poner en funcionamiento el software se debe instalar el motor de bases de datos Postgresql, instalar el servidor web y crear un nuevo sitio con el contenido de la aplicación software.

Para su uso, el administrador deberá crear los diferentes usuarios que utilizarán el software, asignado un rol dependiendo de las actividades que va a realizar.

El acceso al aplicativo se realizará por medio de un navegador web digitando la dirección URL asignada al sitio anteriormente creado.

IV. CONCLUSIONES

El desarrollo de Smart-test permitió crear una aplicación para la elaboración y presentación de pruebas, donde un administrador o docente crea usuarios, asignaturas, pruebas y conforma los grupos; y los estudiantes pueden presentarlas.

El éxito de la incorporación de las TIC en la educación requiere un decidido esfuerzo institucional; junto a esto, es necesario contar con la participación de los docentes en cuanto a adaptación y actualización en el uso de Tecnologías de la Información en el proceso de enseñanza.

REFERENCIAS

- [1] E. Vázquez Pizaña, A. R. Rojo Quiñonez. *El impacto de la evaluación en el proceso de enseñanza-aprendizaje. Bol Clin Hosp Infant Edo Son 2010*; 27(1): 63-66.
- [2] J. Andrew. Johnson, Chris S. *Annual International Conference on Computer Science Education: Innovation & Technology*. Dec2010, p13-18. 6p.
- [3] M. C López de Madrid. *Uso de las TIC en la educación superior de México*. Centro Universitario del Sur de la Universidad de Guadalajara, Prolongación Colon s/n, Ciudad Guzmán, municipio Zapotlan el Grande, Jalisco, México. Fecha de recepción: 31-ago.-07. Fecha de aceptación; 12-sept.-07
- [4] Bhole, Yogesh; Popescu, Adrian. *Journal of Network & Systems Management*. Dec. 2005, Vol. 13 Issue 4, p357-371. 15p. DOI: 10.1007/s10922-005-9000-y.
- [5] Pauw, Wim; Lei, Michelle; Pring, Edward; Villard, Lionel; Arnold, Matthew; Morar, John F. *IBM Systems Journal*. 2005, Vol. 44 Issue 4, p821-845. 25p.
- [6] J. Márquez Díaz, Leonardo Sampedro, Félix Vargas. *Instalación y configuración de Apache, un servidor web gratis. Ingeniería & Desarrollo*. Universidad del Norte. 12: 10-23, 2002.
- [7] K. G. López; García, Leosvani Núñez; Batista, Yenlidé González. *Tono: Revista Técnica de la Empresa de Telecomunicaciones de Cuba, S.A.* 2011, Vol. 8 Issue 1, p30-33. 4p. 3 Graphs. Language: Spanish.
- [8] K. Holger M. *IEEE Software*. It's About Time to Take JavaScript (More) Seriously May/Jun2010, Vol. 27 Issue 3, p60-62. 3p.
- [9] A. Johnston; Ch. S., Johnson. *Annual International Conference on Computer Science Education: Innovation & Technology*. Dec. 2010, p13-18. 6p. DOI: 10.5176/978-981-08-7466-7_itcse-05.

