

**UNIVERSIDAD NACIONAL DEL ALTIPLANO
ESCUELA DE POST GRADO
MAESTRÍA EN INFORMÁTICA**



**“SISTEMA WWW DE EVALUACIÓN DEL DESEMPEÑO
DOCENTE EN LA UNA - PUNO”**

TESIS

PRESENTADA POR:

WILLIAM EUSEBIO ARCAYA COAQUIRA

PARA OPTAR EL GRADO ACADÉMICO DE:

MAGÍSTER SCIENTIAE EN INFORMÁTICA

PUNO - PERÚ

2010

UNIVERSIDAD NACIONAL DEL ALTIPLANO - PUNO

BIBLIOTECA CENTRAL

Fecha Ingreso: 28 SET. 2012

N° 00135

**UNIVERSIDAD NACIONAL DEL ALTIPLANO
ESCUELA DE POST – GRADO**

MAESTRIA EN INFORMATICA

**“SISTEMA WWW DE EVALUACION DEL DESEMPEÑO
DOCENTE EN LA UNA - PUNO”**

TESIS

Presentada a la Dirección de Investigación de la Maestría de Informática de la Escuela de Post Grado de la Universidad Nacional del Altiplano – Puno, para optar el Grado de MAGISTER SCIENTIAE EN INFORMÁTICA.

APROBADO POR:

PRESIDENTE DE JURADO :


.....

M. Sc. Edgar E. Carpio Vargas

PRIMER MIEMBRO :


.....

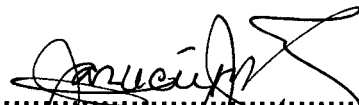
M. Sc. Samuel D. Perez Quispe

SEGUNDO MIEMBRO :


.....

M. Sc. Godofredo Quispe Mamani

ASESOR :


.....

Mag. César A. LLuen Vallejos

DEDICATORIA

A mi padre Eusebio.

AGRADECIMIENTOS

Mi infinita gratitud a Edgar Carpio, Samuel Pérez, Godofredo Quispe, C. Augusto Lluen; a mi esposa Martha, a mi hija Milagros, a mi colega Luis I. Chagua, que todos con su generosa colaboración hicieron posible que se concrete este anhelado trabajo. A Junior y Miguel mi gratitud y motivación para que ellos sigan superando lo andado.

INDICE

RESUMEN	VI
ABSTRACT	VIII
INTRODUCCION	1
CAPITULO 1	2
I. PLANTEAMIENTO DEL PROBLEMA, ANTECEDENTES, OBJETIVOS DE LA INVESTIGACION	2
1.1. Planteamiento y definición del problema	2
1.2. Antecedentes de la investigación	3
1.3. Objetivos de la investigación	6
CAPITULO 2	7
MARCO TEÓRICO CONCEPTUAL	7
2.1. Marco Teórico	7
2.2. Marco Conceptual	40
CAPITULO 3	49
MATERIALES Y METODOS	49
3.1. LUGAR DE ESTUDIO	49
3.2. DESCRIPCION DE LOS MATERIALES UTILIZADOS EN ESTA INVESTIGACIÓN	49
3.4. Población para la aplicación del sistema	56
3.5. Operacionalización de variables	57
CAPITULO 4	58
EXPOSICIÓN Y ANALISIS DE RESULTADOS	58
CONCLUSIONES	67
RECOMENDACIONES	68
BIBLIOGRAFIA	69
ANEXOS	71

RESUMEN

El presente trabajo de investigación, titulado “SISTEMA WWW DE EVALUACION DEL DESEMPEÑO DOCENTE EN LA UNA - PUNO”, aborda el problema de la Universidad Nacional de Altiplano concerniente a la evaluación del desempeño docente, que es un proceso complejo que involucra a estudiantes, órganos de gobierno de las facultades representados por los jefes de departamento, directores de estudios y otros, además de los mismos docentes ya sean contratados u ordinarios. En el primer capítulo se hace un planteamiento del problema considerando los diferentes aspectos que involucra la evaluación docente. En el segundo capítulo se hace una revisión de la tecnología existente en el área de la informática que pueda ayudar a resolver el problema planteado; se muestran las bondades de las aplicaciones web basadas en su despliegue en Internet, así como sus bondades tales como su posibilidad de ubicuidad, tiempo de respuesta, confiabilidad de resultados y flexibilidad en el manejo de contenidos. Se muestra la herramienta de desarrollo Joomla que es un sistema de gestión de contenidos que facilita la implementación de aplicaciones web que requiere un alto requerimiento de actualización de contenidos, gestión de usuarios, políticas de seguridad y acceso a datos en un sistema de gestión de bases de datos tal como es MySQL. En capítulo concerniente a la metodología de evidencia el uso de la metodología RAD que se refiere al desarrollo rápido de aplicaciones basado en el lenguaje de Modelamiento Unificado UML. En cuanto a los resultados se muestra la puesta en producción del sistema desarrollado, las evidencias del cumplimiento de requisitos se pueden verificar en la cantidad de estudiantes y

docentes registrados por el sistema, los cuales ha usado satisfactoriamente el sistema.

ABSTRACT

This work of research, entitled "System WWW of evaluation the performance teaching in the UNA – PUNO", concerning the evaluation of teaching performance, which is a complex process involving students, organs of Government powers represented by the heads of Department, National University directors studios and others, apart from teachers themselves whether they are hired or regular. The first chapter becomes a problem considering the different aspects that involves teaching assessment approach. The second chapter makes a review of existing in the area of computing that can help solve the problem; technology shows the virtues of its deployment on the Internet-based web applications, as well as their qualities such as their possibility of ubiquity, response time, reliability of results and flexibility in the management of content. Which is a system content management which facilitates deployment of web applications requiring a high requirement for updating content, management of users, access to data in a database management system and security policies as it is MySQL Joomla development tool is displayed. Chapter concerning the methodology of evidence using RAD methodology which refers to the rapid development of applications based on UML unified modeling language. On the results shown putting into production of the developed system, evidence of compliance with requirements can be verified in the number of students and teachers registered by the system, which has successfully used the system.

INTRODUCCION

La investigación se desarrolló en la Universidad Nacional del Altiplano, tomando en cuenta la directiva académica emitida por la Universidad mediante el Vice Rectorado Académico.

El trabajo de investigación emprendido implica la construcción de una **Aplicación Web**, que es un producto de software informático que funciona en Internet el cual ofrecerá al miembro de la comunidad universitaria la posibilidad de contar con un instrumento de evaluación altamente fiable, de fácil acceso a los usuarios y que garantice resultados consistentes con la realidad y potencialmente útiles en la toma de decisiones por parte de las autoridades en las diferentes instancias.

El valor teórico y práctico de emprender la presente investigación reside en la posibilidad de incorporar la tecnología informática, así como elementos de seguridad informática en la evaluación que habitualmente se administra con limitaciones de tiempo, limitaciones materiales en la administración del instrumento de evaluación, y un nivel mínimo de fiabilidad de resultados consistentes con la realidad.

Los resultados de la investigación serán relevantes para tomar decisiones concernientes al mejoramiento continuo del desempeño docente, así como en las decisiones concernientes a la relación contractual de algunos miembros con la Universidad.

CAPITULO 1

I. PLANTEAMIENTO DEL PROBLEMA, ANTECEDENTES, OBJETIVOS DE LA INVESTIGACION

1.1. Planteamiento y definición del problema

En toda organización la evaluación del desempeño de su personal permite el mejoramiento continuo y la supervivencia de la organización. La universidad como entidad académica está más ligada a esta función importante. A su vez ésta debe ser asumida y cumplida de la forma más fiable de tal forma que los resultados correspondan a la realidad y ayuden a mejorar a la organización.

En la Universidad Nacional del Altiplano, en el semestre académico 2008 II se implementó la evaluación del desempeño docente mediante instrumentos que fueron aplicados en aulas, los cuales tuvieron limitaciones en cuanto a que no fue aplicado a la totalidad de los interesados, la etapa final del semestre no fue la más conveniente para este tipo de evaluación, el procesamiento de resultados fue lento y mínimamente fiable.

Los resultados de tal evaluación, en la Facultad de Ingeniería Mecánica Eléctrica, Electrónica y Sistemas, fueron tomados en cuenta decisivamente para la permanencia del personal docente contratado.

Esto último configura un problema, por cuanto teniendo a la mano las tecnologías de información, se tomó decisiones sobre resultados pobres en fiabilidad y consistencia, entonces:

¿Cuál es la Tecnología Informática más apropiada para administrar este tipo de instrumento de evaluación? ¿Cómo ayudaría una aplicación Web en la fiabilidad y consistencia con la realidad de los resultados de la evaluación?

En este punto se puede observar que hay posibilidad de incorporar la tecnología WWW para una evaluación exhaustiva, así como la posibilidad de incorporar elementos de seguridad informática para que los resultados sean fiables y consistentes con la realidad.

1.2. Antecedentes de la investigación

Evaluación del desempeño y carrera profesional docente un estudio comparado entre 50 países de América y Europa

El objetivo general de este estudio es conocer la estructura y funcionamiento de los sistemas de carrera docente y de evaluación del desempeño docente en los diferentes países de

América y Europa con vistas a aportar algunas ideas que puedan ayudar a formular políticas sobre estos elementos en América Latina.

Estudio realizado por Unesco Santiago coordinado por F. Javier Murillo Torrecilla. La edición de este libro ha sido posible gracias a la contribución voluntaria de la Unidad de Cooperación Internacional del Ministerio de Educación y Ciencia de España a las actividades del Proyecto Regional de Educación para América Latina y el Caribe (PRELAC).

El trabajo se realizó con la participación de KIPUS, la Red Docente de América Latina y el Caribe, que respaldó el estudio a través de algunas de sus instituciones miembros.

La Evaluación del Desempeño Docente en las Universidades Públicas en México Mario Rueda Beltrán

Revista Iberoamericana de Evaluación Educativa 2008 - Volumen 1, Número 3 (e). Memorias del IV Coloquio Iberoamericano sobre la Evaluación de la Docencia México, DF. 29, 30 y 31 de octubre de 2008.

El texto presenta las características generales de la evaluación del desempeño docente en las universidades públicas de tres regiones de México. La información se obtuvo a través de un cuestionario, el análisis de documentos y la realización de reuniones académicas de intercambio entre los encargados de desarrollar la evaluación. Se reporta la presencia de estrategias para evaluar a los profesores en todas las universidades y el uso

de cuestionarios aplicados a estudiantes como el instrumento dominante en la obtención de la información. Las acciones de evaluación están asociadas a programas especiales vinculados con compensaciones salariales y procesos de acreditación de programas; se observó escasa presencia de acciones institucionales para emplear los resultados de la evaluación en programas de formación continua. Se destaca la presencia, aunque esporádica, de universidades en donde se emplean distintas fuentes para hacerse de información pertinente, se ensayan distintos instrumentos para valorar a profesores que enseñan en diversos escenarios y áreas de conocimiento y se prevén acciones de formación continua.

**“PORTAL WEB ACADÉMICO UTILIZANDO HERRAMIENTAS
CON LICENCIA GNU GPL PARA OPTIMIZAR LA GESTIÓN
ACADEMICA EN LA UNIVERSIDAD NACIONAL DEL
ALTIPLANO PUNO 2004”**

Presentado por. Bach. Juan Antonio Flores Moroco para optar el grado de Ingeniero de Sistemas de la Universidad Nacional del Altiplano. Actualmente se usa un sistema basado en éste en las UNAP.

1.3. Objetivos de la investigación

Objetivo General.

- Aplicar la Tecnología WWW en la evaluación del desempeño docente en la Universidad Nacional del Altiplano.

Objetivos específicos.

- Analizar las tecnología de información y los requerimientos que permitan la aplicación del instrumento de evaluación docente en la UNA Puno.
- Diseñar el sistema de evaluación aplicando la tecnología WWW y los elementos de seguridad Informática.
- Implementar y probar el sistema en una escuela profesional de la Universidad Nacional del Altiplano.

CAPITULO 2

MARCO TEÓRICO CONCEPTUAL

2.1. Marco Teórico

Desarrollo de Aplicaciones Web

En 1977, Apple Computers popularizó el fenómeno de la computación personal. Al principio era el sueño de todo aficionado. Las computadoras se hicieron lo suficientemente económicas para que la gente las pudiera adquirir para su uso personal o para negocios. En 1981, IBM, el vendedor de computadoras más grande del mundo, introdujo la PC de IBM. Literalmente de la noche a la mañana, la computación personal se posicionó en las empresas, en la industria y en las instituciones públicas.

Estas computadoras eran unidades “independientes” (la gente hacía su trabajo en su propia máquina y transportaba sus discos de un lado a otro para compartir información). Aunque las primeras computadoras personales no eran lo suficientemente poderosas para compartir el tiempo entre muchos usuarios, estas máquinas podían interconectarse entre sí mediante redes,

algunas veces mediante líneas telefónicas y otras mediante redes de área local (LANs) dentro de la organización o empresa. Esto derivó en el fenómeno denominado computación distribuida, en el que la computación de la organización en vez de llevarse a cabo dentro de un centro de cómputo, se distribuye a través de redes a los sitios en donde se realiza el trabajo de la organización.

Las computadoras personales actuales son tan poderosas como las máquinas de un millón de dólares de hace apenas una década. Las máquinas de escritorio más poderosas (denominadas estaciones de trabajo) proporcionan actualmente al usuario enormes capacidades. La información se comparte de manera muy sencilla a través de redes de computadoras, en donde algunas computadoras denominadas servidores de archivos ofrecen un lugar común de almacenamiento para programas y datos que pueden ser utilizados por computadoras cliente distribuidas a través de la red; que muy frecuentemente es Internet.

Mucha gente cree que el próximo campo importante en el que los microprocesadores tendrán un impacto profundo es en los dispositivos electrónicos inteligentes para uso doméstico. Al aceptar esto Sun Microsystems patrocinó, en 1991, un proyecto de investigación de la empresa denominado Green. El proyecto desembocó en el desarrollo de un lenguaje basado en C y C++, al cual, James Gosling llamó Oak, debido a un roble que tenía a la vista desde su ventana en las oficinas de Sun. Posteriormente se

descubrió que ya existía un lenguaje de programación con el mismo nombre. Cuando un grupo de gente de Sun visitó una cafetería local, sugirieron el nombre de Java (una variedad de café), y así se quedó.

Sin embargo el proyecto Green tuvo algunas dificultades. El mercado para los dispositivos electrónicos inteligentes de uso doméstico no se desarrollaban tan rápido como Sun había anticipado. Peor aún, un contrato importante por el que Sun había competido, se le otorgó a otra empresa. De manera que el proyecto corría peligro de ser cancelado. Pero para su buena fortuna, la popularidad de la World Wide Web explotó en 1993, y la gente de Sun se dio cuenta de inmediato del potencial de Java para crear contenido dinámico para páginas Web.

Sun anunció formalmente a Java en una exposición profesional que tuvo lugar en mayo de 1995. De inmediato Java generó interés dentro de la comunidad de negocios debido a la fenomenal explosión de la World Wide Web. En la actualidad, Java se utiliza para crear páginas Web con contenido dinámico e interactivo, para desarrollar aplicaciones a gran escala, para aumentar la funcionalidad de los servidores Web (las computadoras que proporcionan el contenido que vemos en los navegadores Web), para proporcionar aplicaciones para dispositivos domésticos (como teléfonos celulares, localizadores y asistentes digitales personales), y más.

La World Wide Web permite a los usuarios de computadoras, localizar y ver documentos basados en multimedia (es decir, documentos con texto, gráficos, animación, audio y/o video) de casi cualquier tema. Aunque Internet se desarrolló hace más de tres décadas, la introducción de World Wide Web es un suceso relativamente reciente. En 1990, Tim Berners-Lee, miembro de la CERN (European Organization for Nuclear Research) desarrolló la World Wide Web y los distintos protocolos de comunicación que forman su esqueleto.

Tanto Internet como la World Wide Web estarán en la lista de las creaciones más importantes de la humanidad. En el pasado la mayoría de las aplicaciones de cómputo se ejecutaban sobre computadoras "independientes", es decir computadoras que no estaban conectadas entre sí. Las aplicaciones actuales pueden ser escritas para comunicar a cientos de miles de computadoras alrededor del mundo. Internet combina las tecnologías de comunicación y computación. Facilitando muchas tareas. Hace que la información esté disponible de manera instantánea in conveniente a nivel global. Hace posible que los individuos, negocios y organizaciones puedan exponerse a nivel global. Está modificando la naturaleza de la forma como se llevan a cabo transacciones y negocios entre los elementos que interactúan. Las comunidades con intereses especiales pueden mantenerse en contacto entre sí y los investigadores pueden anunciar sus descubrimientos de manera instantánea a nivel mundial.

Antes de comenzar a analizar en detalle las estructuras y los modelos de desarrollo de los sitios Web, es conveniente enmarcarlos como una actividad dentro de la ingeniería de diseño de software, caracterizando los diferentes enfoques (modelos sistemáticos y modelos a detalles), identificando su problemática de desarrollo, específicamente las de aplicaciones Web. Seguidamente definiremos las estructuras de los sitios Web, a continuación se define el término usabilidad y se procede a construir los criterios de uso de estos sitios Web.

Diseño de software

En la actualidad el pensar en el desarrollo de un producto sin un diseño previo, equivale a condenarle al fracaso. Un buen diseño proporciona elementos idóneos para entender su funcionamiento, considera las propiedades perceptibles por los usuarios, y respeta los modelos mentales de uso, los cuales están definidos por las experiencias previas. Si el producto es una aplicación informatizada, es decir, si es un programa, resulta imprescindible la ingeniería de software.

El diseño es el proceso mental de concebir los detalles de un objeto o sistema, y si éste logra crear un sistema de fácil interpretación, cuya comprensión sea casi inmediata y satisfaga las necesidades reales del usuario, entonces se dice que es un buen diseño.

Una empresa de software con éxito es aquella que produce de una manera consistente software de calidad que satisface las

necesidades de sus usuarios. Una empresa que puede desarrollar este software de manera predecible y puntual, con uso eficiente y efectivo de recursos, tanto humanos como materiales, tiene un negocio sostenible. Para desarrollar software rápida, eficiente y efectivamente, con el mínimo de desechos de software y de trabajo repetido, hay que tener a la gente apropiada, las herramientas y el enfoque apropiado. Para hacer todo esto de forma consistente y predecible, con una estimación de los costes del sistema en cada etapa de su vida, hay que disponer de un proceso de desarrollo sólido que pueda adaptarse a las necesidades cambiantes del problema y de la tecnología.

Los proyectos de software que fracasan lo hacen por circunstancias propias, pero todos los proyectos con éxito se parecen en muchos aspectos. Hay muchos elementos que contribuyen a una empresa de software con éxito; uno en común es el uso del modelado.

A través del modelado, conseguimos cuatro objetivos:

- Los modelos nos ayudan a visualizar como es o queremos que sea un sistema.
- Los modelos nos permiten especificar la estructura o el comportamiento de un sistema.
- Los modelos nos proporcionan plantillas que nos guían en la construcción de un sistema.
- Los modelos documentan las decisiones que hemos adoptado.

Hay límites a la capacidad humana de comprender la complejidad. A través del modelado, reducimos el problema que se está estudiando, centrándonos en un aspecto cada vez. Éste es, esencialmente, el enfoque “divide y vencerás” que planteó Edsger Dijkstra. Además a través del modelado, se potencia la mente humana.

El uso del modelado tiene una abundante historia en todas las disciplinas de ingeniería. Esa experiencia sugiere cuatro principios básicos del modelado.

- a. La elección acerca de qué modelos crear tiene una profunda influencia sobre cómo se acomete un problema y como se da forma a una solución.
- b. Todo modelo puede ser expresado con diferentes niveles de precisión.
- c. Los mejores modelos están ligados a la realidad.
- d. Un único modelo o vista no es suficiente. Cualquier sistema no trivial se aborda mejor a través de un pequeño conjunto de modelos casi independientes con múltiples puntos de vista.

El ciclo de vida del software plantea un conjunto de actividades para el desarrollo de un programa. En síntesis, el ciclo de vida clásico o modelo cascada contiene las actividades de: levantamiento y análisis de la información, determinación de requerimientos, diseño, codificación, prueba, implantación y mantenimiento. El diseño de software es por tanto una actividad

en el ciclo de vida. Así, se define el diseño de software como un conjunto de actividades de modelado (abstracciones, es decir, separar los aspectos considerados claves del sistema de las características consideradas como no importantes), y de especificaciones de detalles del sistema, que indican como éste cumple los requerimientos establecidos en la fase de análisis.

En el proceso de diseño de software se distinguen las siguientes fases:

El diseño externo:

Cuyo objetivo es establecer la estructura general del sistema programado, definir las funciones del sistema y la interfaz usuario-sistema haciendo énfasis en la entrada y salida de los datos, y de los atributos de calidad.

El diseño interno:

Cuyo objetivo es modelar y especificar los detalles de las estructuras y de los componentes. Esto incluye el diseño de cada módulo, de cada algoritmo y de las estructuras de datos.

El diseño es una actividad compleja y estructurada, que persigue especificar en detalle cada componente, con la finalidad de lograr alcanzar los objetivos del sistema.

En el diseño de la interfaz usuario-sistema se definen el cuándo, el qué y el cómo se han de presentar los datos, haciendo énfasis en el cómo se consideran las necesidades, las

experiencias, el grado de formación y las capacidades de los usuarios. Por lo tanto, en el diseño de las interfaces se deben considerar las siguientes variables:

- Los conceptos familiares a los usuarios potenciales del sistema.
- Consistencia de formato y estilo.
- No debe confundir o sorprender al usuario.
- Permitir al usuario recuperarse de los errores.
- Incluir facilidades de orientación al usuario.

Estas consideraciones son válidas para el diseño de interfaces en general, es decir, tanto para las aplicaciones de software tradicional (interface gráfica de usuario –GUI- Graphic User Interface) como para las interfaces Web.

Como es bien sabido, no seguir un método para el desarrollo de un producto de software, generalmente se traduce en pérdidas de recursos, sobre todo si no se sigue una buena fase de diseño, ya que los errores y omisiones tendrán un efecto de bola de nieve para las siguientes fases.

Diseño de software web

La crisis del software ocurrida en los años 60, la cual unos años después dio origen a la ingeniería de Software, se caracteriza por:

Imprecisión de los planes de proyectos y por la imposibilidad de dar costos.

- Baja productividad del personal de desarrollo y de mantenimiento.
- Baja calidad del producto ofrecido.
- Insatisfacción del cliente y de los usuarios.
- Escasez del personal altamente capacitado.
- Complejidad creciente de los requerimientos de software y del mantenimiento del mismo.

La crisis del software de alguna manera se está repitiendo en el diseño de los sitios Web, bien por falta de personal capacitado para desarrollar este tipo de aplicaciones o por la complejidad y lo extenso de las mismas. Lo cierto es que las organizaciones públicas como la universidad invierten mucho recursos en el desarrollo de sus sitios Web, y por lo general cuando son interrogadas sobre su efectividad real, son pocas las que tienen esperanzas en el retorno de la inversión total hecha para su creación, esto se debe en buena medida a la repetición de los errores cometidos en los años 60. Los internautas, quienes son los verdaderos usuarios de los sitios Web, no es consultada sobre sus requerimientos. El ignorar el grado de amigabilidad, legibilidad y comprensión de un sitio Web se paga casi siempre con el desuso.

La falta de diseño lo hace poco navegable, poco útil y poco atractivo, características nada deseables si queremos que la audiencia visite nuestro sitio Web. Además, si lo que se pretende es que el sitio Web sea utilizado, las características antes mencionadas no se deben cumplir. El diseño de un sitio Web debe, por tanto, realizarse antes de proceder a su desarrollo al igual que cualquier producto de software.

Problemas del Diseño Web

Los problemas de diseño Web han sido analizados por diversos autores destacando los conflictos de diseño como la navegación, la estructura del sitio, la interfaz, y los medios. También, los problemas propios de diseño de software como las metodologías, los equipos interdisciplinarios, los conflictos de diseño comunicativos e instruccional, que deben ser estudiados desde diferentes puntos de vista para encontrar soluciones integrales.

Los conflictos de navegación y de estructuración del sitio Web: estos problemas están ligados al sentimiento de extravío, es decir, la sensación de sentirse perdido dentro de la Web, la cual ocurre cuando hay un desborde cognitivo por exceso de información o porque la estructura del sitio es pobre y no indica dónde se encuentran los recursos. Una pregunta clave en la navegación y estructuración del sitio Web y que siempre debemos tener presentes es ¿dónde está el usuario?. Así mismo, los

usuarios tienen problemas para localizar información específica ya que ésta puede no estar bien estructurada, lo que nos lleva a pensar en la segunda pregunta clave a responder en el diseño ¿Qué ofrece el sitio? ¿Qué hay y dónde está?.

Los conflictos de equipos interdisciplinarios: Son problemas propios de la ingeniería de software, ya que para el correcto desarrollo de aplicaciones Web es indispensable trabajar con equipos interdisciplinarios, conformados por expertos en los temas tratados, así como expertos en multimedia, en audiovisuales, programadores, diseñadores gráficos entre otros. Hoy existe un consenso sobre el carácter multidisciplinario del desarrollo de software orientado a la Web, tanto dentro como fuera de línea. Sin embargo, la realidad nos dice que la integración de equipos de profesionales de diversas áreas, aun es problemática y en muchos casos, en el desarrollo de software orientado al ámbito universitario ya sea de docentes, estudiantes o personal administrativo, la mayoría de las tareas aun son desempeñadas o lideradas por los ingenieros de software.

El problema de la metodología está relacionado con la construcción del software y su ciclo de vida. Resulta un error emprender un desarrollo si antes no existe una metodología (esta lección costó millones de dólares y a ello se debe la existencia de la ingeniería de software). Además, son pocas las metodologías de desarrollo de sistemas de información Web (SIW), y en muchos casos su difusión es escasa. El desarrollo de un SIW por

ser un área muy reciente en la ingeniería de software cuenta con un número de especialistas muy limitado para la demanda del mercado.

Los problemas de la interfaz Web radican en que la experiencia previa en las interfaces gráficas de usuario (GUI) no pueden ser extrapoladas directamente, ya que existen diferencias que deben ser consideradas en el diseño de las interfaces Web.

Comparación entre diseño de interfaces Web y diseño de interfaces gráficas de Usuario (GUI)

El diseño de interfaces Web comparte similitudes con el de las interfaces gráficas de usuario (GUI), ya que ambas interfaces son interactivas, ambos son diseños de software y en principio persiguen objetivos semejantes. Sin embargo, no son exactamente la misma actividad y los resultados de un tipo de interfaz no pueden extrapolarse directamente al otro. En este apartado, se identifica la problemática para entender la diferencia entre diseño Web y el software tradicional.

Los problemas a los que se enfrenta al diseño Web son los siguientes:

Diversidad de resolución en dispositivos de salida "monitor": En el diseño de GUI tradicional el diseñador conoce de antemano el dispositivo de display (pantalla, proyección), pudiendo controlar cada píxel de ella, es decir, el diseñador conoce para qué sistema se está diseñando (ancho y largo –en

pixeles- de la pantalla definido, número de colores, etc.). En el diseño Web esta presunción es imposible, no existe reglas, no es controlable el dispositivo con el cual interactúa el usuario, este podría estar viendo el sitio Web bien desde una computadora de escritorio, o una estación de trabajo, o bien desde su móvil, o su televisor y cada uno de estos dispositivos tiene una área de visibilidad diferente (resolución de pantalla).

Diversidad de dispositivos de conexión: Al igual que en el problema de los monitores o pantallas, el diseñador de interfaz Web se enfrenta con las disparidades de velocidades de conexión. En el diseño tradicional se puede “conocer” (usualmente estimar) el tiempo entre un requerimiento del usuario y la respuesta emitida por el sistema, ya que se conoce el sistema y las características de su arquitectura, pero por la variedad de la arquitectura Web es imposible conocer las velocidades de conexión, y por ende, los tiempos de respuestas, es decir, el usuario tendrá un tiempo de respuesta diferente dependiendo del dispositivo de conexión.

Control de la navegación: En la GUI tradicional el diseñador conoce exactamente el recorrido. La navegación de los usuarios define exactamente el orden de aparición de las pantallas, se puede habilitar o no las opciones de un menú dependiendo de la ubicación del usuario en el programa, en la Web este control se

pierde, es más, se desconoce de qué página proviene el usuario, ya que podría venir de un enlace interno del sitio Web o de un enlace externo, por ejemplo de un buscador, o directamente desde su opción de sitios favoritos.

Diferencia entre cliente y usuario: Cuando se diseña una GUI tradicional, se tiene la ventaja de conocer al cliente, se puede y de hecho se hacen estudios para probar las interfaces (prototipos). Cuando se diseña para una empresa un sitio Web, no se está diseñando para los clientes, es decir, para la empresa, ni para sus empleados, se está diseñando para los usuarios del sitio Web y esto hay que tenerlo muy presente.

Formar parte de un todo: El software tradicional es entendido como una unidad, es en esencia una aplicación cerrada, esto implica la utilización de premisas (válidas en dichos entornos) como lo es que el usuario ve e interactúa con el todo. Los usuarios Web no son los mismos de las GUI tradicionales, por lo tanto no se debe considerar que estos vayan a familiarizarse con el sitio Web y es ingenuo pensar que leerán las ayudas en línea como lo hace en una GUI tradicional. El usuario está familiarizado con la Web, es decir, obedece a metáforas, patrones de diseños ya establecidos y no debemos imponer un estilo propio que rompa las ideas de navegación del usuario, si esto ocurre es

muy probable que el usuario realice una sola visita al sitio Web y al verse frustrado decida abandonar la página.

Orden de ejecución de los eventos: Los eventos de una GUI siempre están controlados por la aplicación, en cambio en la Web cuando el usuario accede a una página del servidor Web para visualizarla, el browser baja la página al cliente, es decir, en la máquina del usuario, cediendo el control a éste, hasta que el usuario dispare el evento Clic en el botón principal, que es cuando se envía el resultado al servidor para su procesamiento. Es allí cuando las variables toman los valores ingresadas por el usuario, esto difiere del orden de ejecución de los eventos de una aplicación GUI tradicional puesto que en ella se tiene el control durante todas las fases.

Metodologías de diseño Web

En el desarrollo de sitios Web podemos identificar dos niveles de diseño, que están relacionados al nivel de abstracción, necesario para modelar las estructuras y los componentes. El primer nivel de diseño llamado enfoque sistemático pretende analizar la estructura. Se han creado metodologías que han sido en algunos casos adaptaciones de las ya existentes en el área de multimedia, probando un enfoque de reutilización de las metodologías (reutilización de componentes de software), y en otros casos, metodologías completamente nuevas que han nacido

de los entornos Web y que en buena medida han dado resultados satisfactorios. El segundo nivel de diseño llamado enfoque de detalle, caracterizado por el análisis en detalle de sus elementos y como pueden integrarse para construir la página Web.

El nivel de diseño llamado enfoque sistemático, pretende construir modelos de las estructuras que soportan la navegación del sitio Web. Los primeros modelos estaban ligados al paradigma relacional y en buena medida son los más divulgados, sin embargo tienen la limitación de que son metodologías para el desarrollo de software multimedia y que requieren de pequeñas adaptaciones para llegar a ser metodologías de diseño de sitios Web.

La mayoría de metodologías de diseños de sitios Web están ligadas al paradigma de orientación a objetos, el cual pretende capturar la estructura del sitio Web y el comportamiento de los elementos de dicha estructura. Estas metodologías están orientadas al desarrollo de sitios Web.

Los modelos más utilizados en el diseño de herramientas multimedia son:

Modelo de diseño de hipertexto HDM (Hypertext Design Model) es el primer modelo de multimedia que se publicó. Introduce la terminología de la multimedia y los conceptos de

entidad, tipos de entidades y estructura de acceso, su mayor logro es el permitir presentar la información de varias formas.

Modelo de referencia Dexter Es un modelo genérico para sistemas hipertexto, publicado en 1994, divide al sistema en tres capas o niveles: La capa de almacenamiento, donde se describe la estructura de un conjunto finito de componentes y de dos operaciones básicas denominadas "resolver" y "accesor" para recuperar los componentes; la capa de componentes, que permiten especificar la presentación, los atributos, las anclas, y los hijos de un componente. La capa de ejecución, que permite la instanciación de un componente al usuario. Sus detractores indican que fue destinado para hipertexto, no consideran el tiempo ni la necesidad de sincronizar los componentes.

Modelo Ámsterdam, Este modelo cubre las fallas del modelo Dexter, es decir, toma en consideración el tiempo y cumple con el criterio de sincronización indispensable en los sistemas multimedia, para ello considera dos relaciones temporales entre los datos, una de ellas llamada colección, que define una relación temporal entre componentes que estarán presentes conjuntamente, y la otra es la sincronización que define una relación temporal que se refiere al orden relativo de presentación.

Metodología RMM (Relationship Management Methodology) está basada en los conceptos implantados en el Modelo de diseño de hipertexto HDM, es decir, en las entidades y los tipos de entidades. Su objetivo es mejorar la navegación a través de un análisis de las entidades del sistema. En teoría, se obtiene una navegación más estructurada y logra que ésta sea más intuitiva para el usuario. Los conceptos de slices y m-slices, que consisten en la agrupación de datos de una entidad en diferentes pantallas, es una de las aportaciones más importantes de esta metodología. Fue la primera metodología completa que se publica para la creación de software multimedia. Su problema principal es que no permite realizar consultas a partir de dos entidades y como está muy atado al modelo entidad relación (modelo E-R) cuando se define una relación (M:N) se obliga a descomponerlas en dos relaciones (1:N) copiando el modelo E-R. Además, no considera las consultas a la base de datos para la creación de páginas Web dinámicas.

Modelo de Diseño Hipermedia Orientado a Objeto OOHDM (The Object-Oriented Hypermedia Design Model), es el sucesor del modelo de diseño de hipertexto HDM, se trata de una metodología que se fundamenta en la orientación a objeto. Propone las siguientes fases: Diseño conceptual o análisis de dominio que utiliza el método de análisis orientado a objeto para

obtener esquemas conceptuales de las clases y de las relaciones entre las mismas. Utiliza las “técnicas de modelado de objeto” llamada notación OMT para el diseño de navegación, donde se define la estructura de navegación por medio de modelos, es decir, a través de diferentes vistas del esquema conceptual; la fase de diseño de interfaz abstracta, se apoya en un modelo orientado a objeto para especificar la estructura y el comportamiento de la interfaz del sistema, este modelo se crea a través de tres tipos de diagramas: diagramas abstractos para cada clase, diagramas de configuración para reflejar los eventos externos y diagrama de estado para señalar el comportamiento dinámico; y por último, la fase de implementación, es decir, la construcción de los programas en programación orientada a objeto.

Modelo DRMM (Dynamic Relationship Management Model), está basado en el modelo RMM sólo que considera las aplicaciones dinámicas, permitiendo el diseño de pantallas con resultados de consultas realizadas por el usuario a una base de datos.

Los modelos más utilizados en el diseño de sitios Web son:

Método del diseño de sitios Web centrados en el usuario, (WSDM: A user-centered design method for Web sites), este método toma muchos de los conceptos previos de diseño de documentos hipermedia, y adiciona un concepto clave para los sitios Web, la audiencia,

definiéndola como la clase de los usuarios que determina los requerimientos del sitio Web.

Técnicas de Diseño de Hipermedia Estructurado, (Structured Hypermedia Design Technique (SHDT)) es una técnica de modelado y un ambiente de diseño basado en la computadora, útil para facilitar la creación de prototipos, considera las páginas estáticas y dinámicas.

Metodología de enfoque de ingeniería para el desarrollo de cursos basados en la Web (A software Engineering Approach to the Development of Web-based Courses) es una metodología basada en la ingeniería de software que incluye todas las fases para el desarrollo de cursos basados en la Web. Describe la estructura típica de una guía de estudio para la Web, se basa en la Orientación a Objeto y presenta cuatro fases: análisis de dominio de curso, para establecer el modelo conceptual de las clases que componen el sistema; la fase de definición de requerimientos, requerimientos de aprendizaje, de interacción, de la interfaz y de calidad; la fase de diseño de la guía de estudio, donde se diseña la estructura de la guía, de las unidades, las páginas Web, y por último, la fase de producción donde se realiza el desarrollo, se ensamblan las unidades y se verifican los requerimientos.

Diseño de un sitio Web – Kendall y Kendall

Cuando diseña un sitio Web, puede usar alguno de los principios del diseño de pantallas. Sin embargo se debe tomar en cuenta la palabra *sitio*. A los primeros documentos mostrados en Internet mediante el protocolo http se les llamó páginas de inicio, pero pronto quedó muy

claro que las compañías, universidades, entidades del sector público y las personas no iban a desplegar una sola página. El término *sitio Web* reemplazó a *página de inicio*, el cual indica que la serie de páginas se debería de organizar, coordinar, diseñar, desarrollar y mantener en un proceso ordenado.

Imprimir es un medio altamente controlado, y el analista tiene una idea muy buena de cómo se verá la salida. La GUI y las pantallas basadas en caracteres alfanuméricos (CHUI, CHaracter-based User Interface o interfaz de usuario basada en caracteres) también están altamente controlados. Sin embargo, Web es un entorno con poco control sobre las salidas.

Los diversos navegadores despliegan imágenes de forma diferente, y la resolución de pantalla tiene un gran impacto en el aspecto de un sitio Web. Las resoluciones estándar son 1024 x 768 ó 1600 x 1200 pixeles. El problema es más complicado por el uso de dispositivos portátiles que se usan para navegar por la Web. La complejidad aumenta cuando se comprende que cada persona podría ajustar su navegador para usar diferentes fuentes y podría desactivar el uso de JavaScript, cookies y otros elementos de programación en Web. Claramente, el analista debe tomar muchas decisiones al diseñar un sitio Web.

Además de los elementos de diseño generales, hay lineamientos específicos adecuados para el diseño de sitios Web de calidad profesional.

Los sitios Web buenos son bien planeados tomando en cuenta:

- Estructura
- Contenido
- Texto
- Gráficos
- Estilo de presentación
- Navegación
- Promoción

Historia de las aplicaciones web

Inicialmente la web era simplemente una colección de páginas estáticas, documentos, etc., que podían consultarse o descargarse.

El siguiente paso en su evolución fue la inclusión de un método para confeccionar páginas dinámicas que permitiesen que lo mostrado fuese dinámico (generado o calculado a partir de los datos de la petición). Dicho método fue conocido como CGI (common gateway interface) y definía un mecanismo mediante el cual podíamos pasar información entre el servidor HTTP y programas externos. Los CGI siguen siendo muy utilizados, puesto que la mayoría de los servidores web los soportan debido a su sencillez. Además, nos proporcionan total libertad a la hora de escoger el lenguaje de programación para desarrollarlos.

El esquema de funcionamiento de los CGI tenía un punto débil: cada vez que recibíamos una petición, el servidor web lanzaba un proceso que ejecutaba el programa CGI. Como, por otro lado, la mayoría de CGI estaban escritos en algún lenguaje interpretado (Perl, Python, etc.) o en

algún lenguaje que requería run-time environment (VisualBasic, Java, etc.), esto implicaba una gran carga para la máquina del servidor.

Además, si la web tenía muchos accesos al CGI, esto suponía problemas graves. Por ello se empiezan a desarrollar alternativas a los CGI para solucionar este grave problema de rendimiento. Las soluciones vienen principalmente por dos vías. Por un lado se diseñan sistemas de ejecución de módulos más integrados con el servidor, que evitan que éste tenga que instanciar y ejecutar multitud de programas. La otra vía consiste en dotar al servidor de un intérprete de algún lenguaje de programación (RXML, PHP, VBScript, etc.) que nos permita incluir las páginas en el código de manera que el servidor sea quien lo ejecute, reduciendo así el tiempo de respuesta.

A partir de este momento, se vive una explosión del número de arquitecturas y lenguajes de programación que nos permiten desarrollar aplicaciones web. Todas ellas siguen alguna de las dos vías ya mencionadas. De ellas, las más útiles y las que más se utilizan son aquellas que permiten mezclar los dos sistemas, es decir, un lenguaje de programación integrado que permita al servidor interpretar comandos que “incrustemos” en las páginas HTML y un sistema de ejecución de programas más enlazado con el servidor que no presente los problemas de rendimiento de los CGI.

Sun Microsystems con su sistema Java, que está integrada por dos componentes; un lenguaje que permite incrustar código interpretable en las páginas HTML y que el servidor traduce a programas ejecutables,

JSP (Java server pages) y un mecanismo de programación estrechamente ligado al servidor, con un rendimiento muy superior a los CGI convencionales, llamado Java Servlet.

Otra de las tecnologías que más éxito ha obtenido y una de las que más se utiliza en Internet es el lenguaje de programación interpretado por el servidor PHP. Se trata de un lenguaje que permite incrustar HTML en los programas, con una sintaxis que proviene de C y Perl.

Además, habida cuenta de su facilidad de aprendizaje, su sencillez y potencia, se está convirtiendo en una herramienta muy utilizada para algunos desarrollos.

Otros métodos de programación de aplicaciones web también tienen su mercado. Así sucede con mod_perl para Apache, RXML para Roxen, etc., pero muchos de ellos están vinculados a un servidor web concreto.

EVALUACION DOCENTE

En la ley universitaria 23733 se establece respecto a la evaluación docente lo siguiente:

Artículo 46.- La admisión a la carrera docente, en condición de profesor ordinario, se hace por concurso público de méritos y prueba de capacitación docente o por oposición, de acuerdo a las pautas que establezca al respecto el Estatuto de cada Universidad. La promoción, ratificación o separación de la docencia se realizan por **evaluación**

personal, con citación y audiencia del profesor. Participan en estos procesos la Facultad y el Departamento respectivo y corresponde a la primera formular la propuesta del caso al Consejo Universitario para su resolución.

Artículo 47.- Los Profesores Principales son nombrados por un periodo de siete años, los Asociados y Auxiliares por cinco y tres años, respectivamente. Al vencimiento de estos periodos son ratificados, promovidos o separados de la docencia por el Consejo Universitario previo el proceso de evaluación que determina el Estatuto.

Los Profesores Contratados lo son por el plazo máximo de tres años. Al término de este plazo tienen derecho de concursar, para los efectos de su admisión a la carrera docente, en condición de Profesores Ordinarios, de acuerdo al procedimiento establecido en el artículo anterior.

En caso de no efectuarse dicho concurso, el contrato puede ser renovado por una sola vez y por el mismo plazo máximo, previa **evaluación** del profesor.

Artículo 48.- Sin perjuicio de los demás requisitos que determine el Estatuto de cada Universidad, y previa **evaluación** personal, la promoción de los Profesores Ordinarios requiere:

a) Para ser nombrado Profesor Principal, haber desempeñado cinco años de labor docente con la categoría de Profesor Asociado, tener el grado de maestro o doctor y haber realizado trabajos de investigación de acuerdo con su especialidad. Por excepción podrán concursar también a

esta categoría, profesionales con reconocida labor de investigación científica y con más de diez (10) años de ejercicio profesional y,

b) Para ser nombrado Profesor Asociado, haber desempeñado tres años de docencia con la categoría de Profesor Auxiliar.

Toda promoción de una categoría a otra está sujeta a la existencia de vacante y se ejecuta en el ejercicio presupuestal siguiente.

Artículo 51.- Son deberes de los Profesores Universitarios:

a) El ejercicio de la cátedra con libertad de pensamiento y con respeto a la discrepancia;

b) Cumplir con el Estatuto de la Universidad y sus reglamentos y realizar cabalmente y bajo responsabilidad las actividades de su cargo;

c) Perfeccionar permanente sus conocimientos y capacidad docente y realizar labor intelectual creativa;

d) Observar conducta digna;

e) Presentar periódicamente informes sobre el desarrollo de su labor en caso de recibir remuneración especial por la investigación; y

f) Ejercer sus funciones en la Universidad con independencia de toda actividad política partidaria.

El Estatuto de cada Universidad establece un sistema de estricta evaluación de profesor, que incluye la calificación de su producción intelectual universitaria o extra universitaria.

En la Universidad Nacional del Altiplano el estatuto universitario, sobre la evaluación del docente universitario menciona:

En el CAPITULO III DE LA EVALUACIÓN, RATIFICACIÓN Y PROMOCIÓN DE DOCENTES ORDINARIOS:

Art.234.- La evaluación de los docentes será permanente, objetiva e integral y tiene como propósito fundamental, garantizar la superación continua y la calidad en el servicio académico.

Art.235.- Las áreas que comprende la evaluación docente son:

- Actividad académica.
- Investigación.
- Proyección y extensión universitaria.
- Capacitación en los eventos organizados por la Universidad.
- Capacitación permanente.
- Producción intelectual.
- Experiencia profesional en el área.
- Cómputo e idioma extranjero, y
- Otras funciones académicas y administrativas.

Art.236.- Los procedimientos, técnicas e instrumentos de evaluación del docente, en las áreas indicadas en el artículo precedente, serán establecidos por el reglamento correspondiente aprobado por el Consejo Universitario.

Art.237.- La evaluación personal y la audiencia son obligatorias, se harán en los marcos académicos e institucionales, bajo responsabilidad personal de quien tenga a su cargo dicho proceso. Esta evaluación se realizará en forma semestral, de conformidad con la Ley, el Estatuto y el reglamento respectivo.

La directiva académica de la Universidad Nacional del Altiplano, emitida para las labores durante el periodo académico 2009, menciona en el punto 2.2.1 sobre contrato de docentes, lo siguiente:

Para el contrato de docentes para el I semestre 2009 se implementará una evaluación los días 30 y 31 de julio del presente año en cada facultad, utilizando los instrumentos de Evaluación al Desempeño Docente.

Así mismo, en el punto 20. del Jefe de Departamento Académico inciso h) menciona: supervisa, monitorea y evalúa el desempeño académico de los docentes nombrados y contratados utilizando los instrumentos idóneos y fiables de Evaluación Formativa de acuerdo a los anexos que se toma en cuenta en esta investigación.

FUNDAMENTOS TEÓRICOS DE LOS SISTEMAS DE EVALUACIÓN DEL DESEMPEÑO DOCENTE

Cualquier propuesta de evaluación docente parte de una concepción de lo que es ser un buen docente. De esta forma, será diferente el modelo de evaluación si se considera que un buen docente es aquel que: se desarrolla bien en el aula, sus alumnos aprenden, sabe la materia y sabe enseñarla, o aquel que posee una serie de rasgos y

características positivas. Así, es clásica la propuesta de Scriven, quien en un análisis detallado de los conocimientos y competencias básicas que tiene un buen profesor, destaca los siguientes elementos para la evaluación de los docentes: conocimiento de la materia, competencias de instrucción, competencias de evaluación, profesionalidad y otros deberes con la escuela y la comunidad. Pero también es posible partir de un modelo teórico, como es el de la eficacia docente. Así, tendríamos que los contenidos o ámbitos de la evaluación serían: el conocimiento, las habilidades, la competencia, la eficacia, la productividad y la profesionalidad docentes.

Básicamente se han identificado seis teorías y/o modelos acerca de la caracterización del “buen docente” que pueden ser útiles como marco de referencia para analizar las propuestas encontradas:

- Modelo centrado sobre los rasgos o factores.
- Modelo centrado sobre las habilidades.
- Modelo centrado en las conductas manifiestas en el aula.
- Modelo centrado sobre el desarrollo de tareas.
- Modelo centrado en los resultados.
- Modelo basado en la profesionalización.

Una primera idea, extraída tras aplicar este marco de análisis, es que pocos son los sistemas de evaluación del desempeño que hagan explícitos los fundamentos teóricos de los que parten, lo cual supone, sin duda, una debilidad.

INSTRUMENTOS

Aunque cualquier procedimiento o instrumento de obtención de información con propósitos valorativos puede ser aplicado a la evaluación del desempeño docente, es posible afirmar que son ocho los procedimientos de obtención de información más utilizados para la evaluación externa del desempeño docente: observación en el aula, entrevistas o cuestionarios al docente, informe de la dirección del centro, cuestionarios dirigidos a los alumnos o a sus familias, test y pruebas estandarizadas, portafolio o carpeta del profesor, pruebas de rendimiento a los alumnos, así como diferentes procedimientos de autoevaluación del docente.

Observación en el aula

Un procedimiento muy habitual para obtener informaciones sobre el desempeño del profesor es observar cómo se desenvuelve en su contexto más habitual de trabajo: el aula. Esta observación puede ser formal (preparada de antemano), o informal (sin una preparación previa), puede utilizarse un protocolo de observación cerrada o recoger de forma abierta las percepciones del evaluador. La observación puede ser desarrollada por directivos, jefes de oficina, por colegas o por expertos, y suelen tratarse de dos o tres sesiones de observación de una hora.

Entrevistas a los profesores

Se considera un instrumento más las escalas de evaluación aplicadas tanto de manera escrita, en forma de cuestionarios, como oralmente a modo de entrevista. Pueden ser cuantitativas o cualitativas,

y suelen utilizarse para valorar estrategias de conducción del aula, habilidades instructivas, personalidad del docente, preparación académica y profesional, actitudes e intereses, etc.

Informe de la dirección o jefe inmediato

También es frecuente que se solicite a la dirección o al jefe inmediato del centro o del departamento, un informe donde se señalen tanto las capacidades del docente como el desempeño de la tarea que ha desarrollado.

Test y pruebas estandarizadas

En algunos países se utilizan pruebas estandarizadas o test para analizar las habilidades básicas, así como los conocimientos pedagógicos y académicos de los docentes.

Carpeta o portafolio del profesor

El portafolio, en su concepción original, es una recopilación de documentos variados sobre el desempeño del profesor, de tal forma que pueden incluirse documentos de planificación, notas escritas de los padres, apuntes o materiales de trabajo, resultados de los alumnos, etc., en síntesis, se constituye una colección de experiencias vividas por el docente en el desempeño de su labor profesional. En los últimos años se está extendiendo su uso como instrumento de evaluación externa pero, sobre todo, como herramienta para la auto reflexión por parte del profesorado.

Pruebas de rendimiento a los alumnos

Aunque mucho menos frecuente por las duras críticas que ha sufrido por parte de los estudiosos, otra información susceptible de ser utilizada para la evaluación externa del desempeño docente, es estudiar el impacto del trabajo sobre los estudiantes, es decir, considerar los resultados académicos de los alumnos.

Entrevistas y cuestionarios dirigidos a los alumnos o a sus familias

Otra estrategia utilizada es la de obtener información sobre el desempeño de los docentes mediante la aplicación de cuestionarios cerrados dirigidos a los alumnos o a sus familias.

Autoevaluación

Se pide a los propios profesores que hagan un informe que recoja su propia autoevaluación. Tomando en cuenta algunos descriptores y dominios como la preparación de la enseñanza, preparación de un ambiente para el aprendizaje, responsabilidades profesionales y otros.

OTROS ELEMENTOS

Es posible reconocer otros elementos diferenciadores en la evaluación del desempeño docente: quiénes son los responsables de la evaluación, si la misma es voluntaria u obligatoria, y su periodicidad.

Responsables de la evaluación

La situación más habitual es que los principales agentes responsables de la evaluación de los docentes sean las personas que

ocupan los puestos directivos, son las autoridades quienes se responsabilizan del proceso, puede darse el caso con profesores pares que apoyan el proceso de evaluación.

Voluntariedad de la evaluación

Dependiendo del tipo y los propósitos de la evaluación, ésta puede ser obligatoria o voluntaria.

Periodicidad

Puede ser anual, semestral o continua. Está ligado al criterio de la obligatoriedad y las leyes y reglamentos vigentes.

2.2. Marco Conceptual

Aplicación web.

Una aplicación web es la herramienta que contiene acciones de comunicación institucional, aporta servicios de valor agregado al público y contribuye a gestionar el conocimiento de la organización. Se trata de un producto en evolución constante, que contiene todas las necesidades en Internet determinadas por una organización. Una aplicación web puede incluir a varios sitios web dentro suyo y a múltiples productos hipermedia específicos.

A pesar de que no existe una definición académica precisa del término, en la consultoría en Internet actualmente se utiliza el concepto de aplicación web como etapa superadora de sitio web.

Es la tercera generación del desarrollo de contenidos en la Web, tomando a las páginas web de mediados de los '90 como primera

etapa evolutiva, siguiendo por una segunda generación de productos denominados sitios webs, hasta conformar esta tercera generación de aplicaciones, un tipo de producto de mayor sofisticación, superior a ambos antecesores.

Hojas de Estilo en Cascada (CCS)

Son herramientas que permiten administrar los estilos de presentación de un sitio Web, tanto al diseñador como al usuario final, su uso garantiza la consistencia, sin embargo, no todos los navegadores lo soportan. Antes de la introducción de las hojas de estilo, los creadores de páginas Web sólo tenían un control parcial sobre el aspecto final de sus páginas.

Las hojas de estilo permiten un mayor control sobre el aspecto de los documentos. Con ellas se pueden especificar muchos atributos, tales como: colores, márgenes, alineación de elementos, tipos y tamaños de letras y muchos más. Es posible utilizar bordes para hacer que ciertos elementos resalten del resto de un documento. También ayudan a especificar diferentes fuentes para párrafos, títulos, subtítulos, entre otros.

Además, puede emplearse hojas de estilo como patrones o páginas maestras, de forma tal, que múltiples páginas puedan tener el mismo aspecto.

Lenguaje Scripting

Son lenguajes de programación donde se hacen pequeñas rutinas para ser ejecutadas de lado del cliente y así dar más poder de dinamismo e interactividad a las páginas Web. Permiten empotrar

subprogramas en los documentos Web utilizando un lenguaje de programación (por ejemplo JavaScript) o permiten cargar y ejecutar applets basados en Java, independientemente de la plataforma utilizada para la comunicación en Internet. Durante su ejecución puede generar contenido dinámico, interactuar con el usuario, validar datos de formularios, o incluso, crear ventanas y ejecutar aplicaciones independientes de las páginas. Estos programas sobrepasan el modelo HTML original. Los applets son una desviación del modelo básico de la Web, sólo debemos recordar que los servidores realizaban casi todo el trabajo y en el lado del cliente sólo la visualización, los lenguajes scripting cambiaron esta filosofía, otorgándole al cliente la posibilidad de ejecutar programas. Los applets también permiten extender las características de los navegadores de los clientes sin obligar a los usuarios a cambiar de navegador. La ventaja de los applets es que a través de ellos se puede proporcionar interfaces de usuario más ricas a la Web. El inconveniente es que un programa del lado del cliente utiliza recursos de la computadora, es decir, aumenta la carga de trabajo en éste. Además el código también “pesa” (aumenta el tamaño de la página) lo que hace necesario un ancho de banda mayor, y si la aplicación requiere datos del servidor, también incrementa el peso en el sitio Web.

Evaluación

Es una unidad de información que contiene un grupo de preguntas que deben ser resueltas por el alumno o docente. Se refiere al instrumento que es desplegado por la aplicación Web.

Joomla!

Es un potente gestor de contenidos web (CMS o Content Management System) que permite crear sitios web elegantes, dinámicos e interactivos. Por su diseño, potencia, flexibilidad y por sus enormes posibilidades de ampliación se está convirtiendo en el sistema de publicación preferido por muchos desarrolladores y por millones de webmasters en todo el mundo para desarrollar su portal web.

Joomla! es una aplicación Open Source o de código abierto programada en lenguaje PHP bajo una licencia GPL y que utiliza una base de datos MySQL para almacenar el contenido y los parámetros de configuración del sitio. Además de libre, Joomla! es gratuito y no tendrás que pagar nada por usar este CMS para construir tu web.

Joomla! se basa en Mambo, otro CMS iniciado por la empresa australiana Miro, que tenía además la propiedad de la marca. El proyecto Joomla surge en 2005 cuando Miro decidió crear una Fundación que, a juicio de una buena parte de los desarrolladores de Mambo, se apartaba bastante de la filosofía del software libre. Estos desarrolladores se agruparon en torno al movimiento Open Source Matters que poco después daría lugar a Joomla!, una denominación

que procede de una palabra de la lengua swahili que significa "todos juntos".

PHP

PHP, cuyas siglas responden a un acrónimo recursivo (PHP: hypertext preprocessor), es un lenguaje sencillo, de sintaxis cómoda y similar a la de otros lenguajes como Perl, C y C++. Es rápido, interpretado, orientado a objetos y multiplataforma. Para PHP se encuentra disponible una multitud de librerías. PHP es un lenguaje ideal tanto para aprender a desarrollar aplicaciones web como para desarrollar aplicaciones web complejas. PHP añade a todo eso la ventaja de que el intérprete de PHP, los diversos módulos y gran cantidad de librerías desarrolladas para PHP son de código libre, con lo que el programador de PHP dispone de un impresionante arsenal de herramientas libres para desarrollar aplicaciones.

PHP suele ser utilizado conjuntamente con Perl, Apache, MySQL o PostgreSQL en sistemas Linux, formando una combinación barata (todos los componentes son de código libre), potente y versátil. Tal ha sido la expansión de esta combinación que incluso ha merecido conocerse con un nombre propio LAMP (formado por las iniciales de los diversos productos).

Apache, así como algunos otros servidores web, Roxen entre ellos, puede incorporar PHP como un módulo propio del servidor, lo cual permite que las aplicaciones escritas en PHP resulten mucho más rápidas que las aplicaciones CGI habituales.

MySQL

MySQL es un sistema para la administración de bases de datos relacionales (conocido con las siglas RDBMS en inglés o SGBDR en castellano) rápido y sólido. Las bases de datos permiten almacenar, buscar, ordenar y recuperar datos de forma eficiente. El servidor de MySQL controla el acceso a los datos para garantizar el uso simultáneo de varios usuarios, para proporcionar acceso a dichos datos y para asegurarse de que sólo obtienen acceso a ellos los usuarios con autorización. Por lo tanto, MySQL es un servidor multiusuario y de subprocesamiento múltiple. Utiliza SQL (del inglés Structured Query Language, Lenguaje de Consulta estructurado), el lenguaje estándar para la consulta de bases de datos utilizado en todo el mundo. MySQL lleva disponible desde 1996 pero su nacimiento se remonta a 1979. Ha obtenido el galardón de Choice Award del Linux Journal Readers en varias ocasiones. MySQL se distribuye bajo un sistema de licencias dual. Puede utilizarse bajo una licencia de código abierto (GPL), que es gratuita mientras cumpla las condiciones de la misma. Si desea distribuir una aplicación que no sea GPL y que incluya MySQL, se puede adquirir una licencia comercial.

Apache

Apache es un servidor web de código libre robusto cuya implementación se realiza de forma colaborativa, con prestaciones y funcionalidades equivalentes a las de los servidores comerciales. El proyecto está dirigido y controlado por un grupo de voluntarios de

todo el mundo que, usando Internet y la web para comunicarse, planifican y desarrollan el servidor y la documentación relacionada. Estos voluntarios se conocen como el Apache Group. Además del Apache Group, cientos de personas han contribuido al proyecto con código, ideas y documentación.

El servidor HTTP Apache es un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTP 1.3, pero más tarde fue reescrito por completo.

En febrero de 1995, el servidor web más popular de Internet era un servidor de dominio público desarrollado en el NCSA (National Center for Supercomputing Applications en la Universidad de Illinois). No obstante, al dejar Rob McCool (el principal desarrollador del servidor) la NCSA en 1994, la evolución de dicho programa había quedado prácticamente reducida a la nada. El desarrollo pasó a manos de los responsables de sitios web, que progresivamente introdujeron mejoras en sus servidores. Un grupo de éstos, usando el correo electrónico como herramienta básica de coordinación, se puso de acuerdo en poner en común estas mejoras (en forma de “parches”, patches). Dos de estos desarrolladores, Brian Behlendorf y Cliff Skolnick, pusieron en marcha una lista de correo, un espacio de información compartida y un servidor en California donde los desarrolladores principales pudiesen trabajar.

A principios del año siguiente, ocho programadores formaron lo que sería el Grupo Apache.

Éstos, usando el servidor NCSA 1.3 como base de trabajo, añadieron todas las correcciones de errores publicadas y las mejoras más valiosas que encontraron y probaron el resultado en sus propios servidores. Posteriormente publicaron lo que sería la primera versión oficial del servidor

Apache (la 0.6.2, de Abril de 1995). Casualmente, en esas mismas fechas, NCSA reemprendió el desarrollo del servidor NCSA. En este momento el desarrollo de Apache siguió dos líneas paralelas. Por un lado, un grupo de los desarrolladores siguió trabajando sobre el Apache 0.6.2 para producir la serie 0.7, incorporando mejoras, etc. Un segundo grupo reescribió por completo el código, creando una nueva arquitectura modular. En julio de 1995 se migraron a esta nueva arquitectura las mejoras existentes para Apache 0.7, haciéndose público como Apache 0.8.

El día uno de diciembre de 1995, apareció Apache 1.0, que incluía documentación y muchas mejoras en forma de módulos incrustables. Poco después, Apache sobrepasó al servidor de NCSA como el más usado en Internet, posición que ha mantenido hasta nuestros días. En 1999 los miembros del Grupo Apache fundaron la Apache Software Foundation, que provee soporte legal y financiero al desarrollo del servidor Apache y los proyectos laterales que han surgido de éste.

HTML

siglas de HyperText Markup Language (Lenguaje de Marcado de Hipertexto), es el lenguaje para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes, audio, y otros elementos presentes en una página web. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo Javascript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

CAPITULO 3

MATERIALES Y METODOS

3.1. LUGAR DE ESTUDIO

Esta tesis de maestría se realizó en la Universidad Nacional del Altiplano, la implementación de la Aplicación Web, se desarrolló en el laboratorio del investigador, que a su vez es docente de la escuela profesional de Ingeniería de Sistemas de la UNA Puno.

3.2. DESCRIPCION DE LOS MATERIALES UTILIZADOS EN ESTA INVESTIGACIÓN

En la etapa de desarrollo se tuvo como principal plataforma de desarrollo

- Una LapTop HP Pavilion dv5
- Sistema operativo: Windows Vista Home Premium.
- RAM: 4 GB
- HD: 300 GB

En el software para el desarrollo se utilizó el paquete XAMPP, que es una integración de herramientas, donde "X" se refiere al sistema operativo que puede ser Linux, Windows, Mac u otro. "A" se refiere a Apache que es

un servidor del protocolo HTTP, "M" se refiere a MySQL que es el gestor de bases de datos, "P" se refiere a PHP, que es el lenguaje de desarrollo de aplicaciones Web dinámicas.

XAMPP para Windows

Versión para Windows 98, NT, 2000, XP y Vista. Esta versión contiene: Apache, MySQL, PHP + PEAR, Perl, mod_php, mod_perl, mod_ssl, OpenSSL, phpMyAdmin, Webalizer, Mercury Mail Transport System para Win32 y NetWare Systems v3.32, JpGraph, FileZilla FTP Server, mcrypt, eAccelerator, SQLite, y WEB-DAV + mod_auth_mysql.

English / Deutsch /
Francais /
Nederlands / Polski /
Italiano / Norwegian /
Español / 中文 /
日本語 / 한국어

XAMPP for Windows

XAMPP Lite
[PHP: 5.2.9]
Bienvenido
Estado
chequeo de seguridad
Documentación
Componentes
phpinfo()
Demos
Administración de CD
Biontmo
Instant Art
Flash Art
Agenda de telefonos
Herramientas
phpMyAdmin
Webalizer

Bienvenido a XAMPP para Windows Version 1.7.1 !

Felicitaciones:
XAMPP se instaló con éxito en su ordenador!

Ahora se puede empezar a trabajar. :) Primero por favor pulse encima de »Estado« en la parte izquierda. De esta manera tendrá una visión de que es lo que funciona ya. Algunas funciones estarán desactivadas. Es intencionado. Son funciones, que no funcionan en todas partes o eventualmente podrían ocasionar problemas.

Atención: XAMPP fue modificado a partir de la versión 1.4.x a una administración de paquete único. Existen los siguientes paquetes/Addons:

- XAMPP paquete básico
- XAMPP Perl addon
- XAMPP Tomcat addon
- XAMPP Cocoon addon
- XAMPP Python addon (developer version)

Figura 1. Inicio de XAMPP en el servidor.

Se puede encontrar la documentación así como los archivos binarios para la instalación en: <http://www.apachefriends.org/es/xampp.html>.

Joomla 1.5

Joomla es un sistema de gestión de contenidos CMS, de código abierto, basado en PHP y MySQL.

Se tiene los enlaces para los instalables y documentación:
<http://www.joomla.org/>, y en idioma español en:
<http://www.joomlaspanish.org/>.



Figura 2. Logo oficial de Joomla.

Net Beans 6.8

Es la interfaz de desarrollo integrado más versátil producido por Sun, soporta una serie de lenguajes de programación: java, c++, PHP, html, y otros. Se usó para la edición de código en el desarrollo de la aplicación.

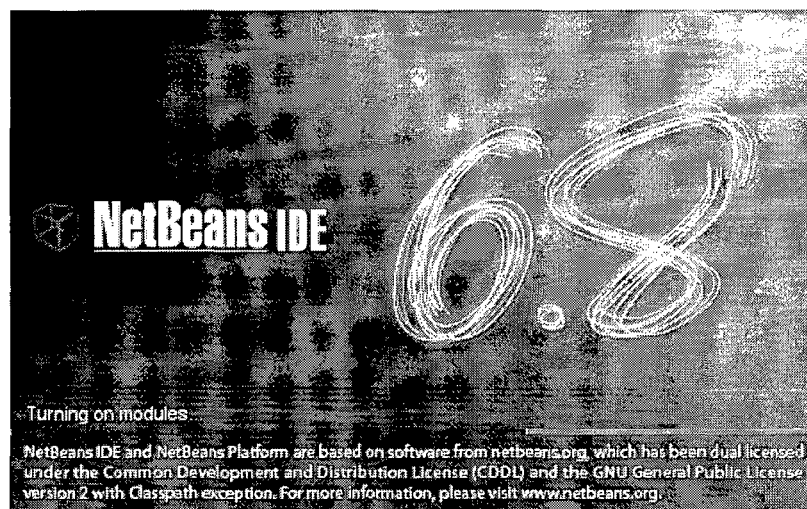


Figura 3 Inicio de Net Beans.

UML

Para el diseño de la aplicación se usó el Lenguaje de Modelamiento Unificado UML 2.0, que es una herramienta que permite modelar en base al enfoque orientado objetos. El UML también permite modelar la base de datos. Las especificaciones del UML se pueden encontrar en: <http://www.uml.org/>.

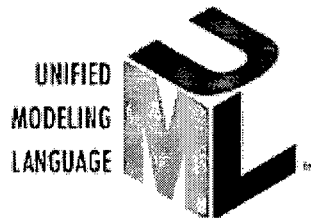


Figura 4. Logo oficial del UML

NaviCat 8

Es la herramienta de administración de bases de datos que se conecta con MySQL, facilita la ejecución de consultas así como las tareas de creación, recuperación, actualización y supresión de registros, así como la administración de usuarios.

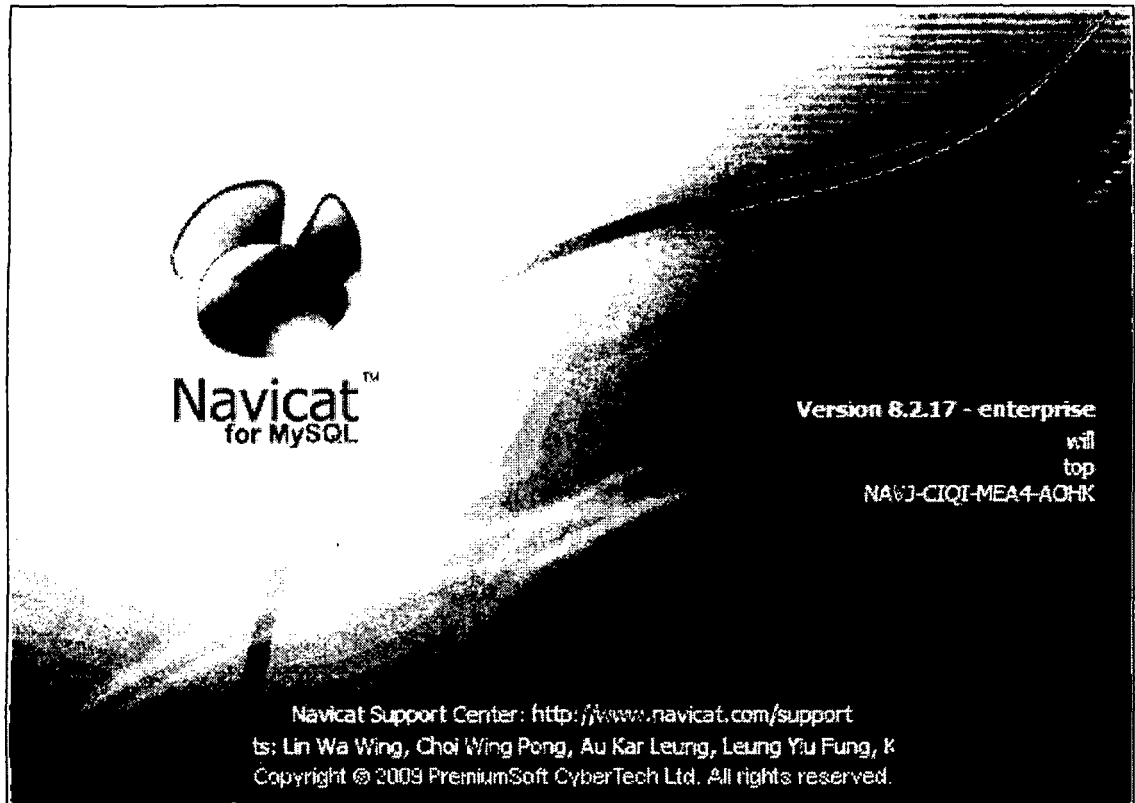


Figura 5 Inicio de Navicat

3.3. METODOLOGIA

El diseño de arriba abajo (top-down) es un proceso orientado en el sentido en que se enfoca en las acciones necesarias en lugar de en las estructuras de datos. En contraste, el diseño orientado a objetos (OOD) se enfoca en los elementos de datos que se requieren y en las operaciones que habrán de llevarse a cabo en dichos elementos. En el OOD, primero se identifica el objeto que participa en el problema y después la forma en que interactúa para formar la solución. Las características comunes de una colección de objetos similares define una clase, mientras que las interacciones se identifican como mensajes que un objeto envía a otro. Para que un objeto pueda recuperar un mensaje, su clase debe ofrecer un

operador (por lo general una función) para procesar dicho mensaje. El diseño orientado a objetos incorpora algunos elementos del diseño arriba abajo. También incorpora algunos de los elementos del diseño de abajo arriba (bottom-up), y se enfoca en el diseño de componentes individuales del sistema. De modo que el OOD es una combinación de ambas técnicas.

Abstracción procedural

La abstracción procedural es la filosofía que indica que el desarrollo del procedimiento debería de separar lo referente a lo que se debe alcanzar mediante un procedimiento (una función), de los detalles de cómo debe lograrse.

Abstracción de datos

La idea de la abstracción de datos nace para especificar los objetos de los datos para un problema y las operaciones que habrán de llevarse a cabo sobre dichos objetos sin preocuparse demasiado por cómo (los objetos de datos) serán representados y almacenados en la memoria. Podrá describir que información es almacenada en el objeto de datos sin tener que especificar cómo se organiza y representa la información. Ésta es la vista lógica del objeto de dato, de manera opuesta a la vista física, su representación interna actual en la memoria. Una vez que se haya entendido la vista lógica, puede usar el objeto de dato y sus operadores en los programas.

Metodología para el diseño de la Base de Datos

Modelo Relacional.- El modelo relacional se ha establecido actualmente como el principal modelo de datos para las aplicaciones de procesamiento de datos. Ha conseguido la posición principal debido a su simplicidad, que facilita el trabajo del programador en comparación con otros modelos anteriores como el de red y el jerárquico.

En el modelo relacional se utiliza un grupo de tablas para representar los datos y las relaciones entre ellos. Cada tabla está compuesta por varias columnas, y cada columna tiene un nombre único.

Una base de datos relacional consiste en un conjunto de tablas, a cada una de las cuales se le asigna un nombre exclusivo. Cada tabla tiene una estructura donde se representan los atributos de entidades en columnas (E-R). Cada fila de la tabla representa una relación entre un conjunto de valores. Dado que cada tabla es un conjunto de dichas relaciones, hay una fuerte correspondencia entre el concepto de tabla y el concepto matemático de relación, del que toma su nombre el modelo de datos relacional.

Normalización

La normalización es la transformación de las vistas del usuario complejas y del almacén de datos a un juego de estructuras de datos más pequeñas y estables. Además de ser más simples y estables, las estructuras de datos normalizadas son más fáciles de mantener que otras estructuras de datos.

En la primera forma normal se remueven los grupos repetitivos, en la segunda forma normal deben removerse las dependencias parciales y en la tercera forma normal se quitan las dependencias transitivas.

3.4. Población para la aplicación del sistema

El artefacto producido, en este caso la aplicación web, sistema o software, ha sido probado y aplicado a la población estudiantil de la facultad de educación de la Universidad Nacional del Altiplano. Los estudiantes matriculados en el semestre académico 2009-I desarrollado entre fines del 2009 e inicios del 2010. La aplicación web también involucra a los docentes que estuvieron laborando en dicho periodo en la misma facultad.

**CUADRO 1. ESTUDIANTES DE LA FACULTAD DE EDUCACIÓN
UNA PUNO**

Usuarios	Población
Estudiantes de la facultad de Educación Registrados en el sistema	1015
Docentes de la facultad Registrados	93

Fuente: OTI UNA Puno (2009-I)

3.5. Operacionalización de variables

CUADRO 2. OPERACIONALIZACION DE VARIABLES

VARIABLE	DIMENSION	INDICADOR	INDICE
DEPENDIENTE <ul style="list-style-type: none"> • Aplicación Web 	Software orientado a Internet.	Análisis Diseño Implementación Prueba	(¿Tiene?) Si – No Bueno Regular Deficiente
INDEPENDIENTE <ul style="list-style-type: none"> • Preguntas de la evaluación • Usuarios 	Desempeño docente Estudiantes evaluadores	Número de preguntas Número de estudiantes	N: Números Naturales
INTERVINIENTE <ul style="list-style-type: none"> • Servidor de Internet 	Dominio en Internet	Número de visitantes al <i>Web Site</i> ¹ .	N: Números Naturales

Fuente: Elaboración propia del autor.

¹ Página Web o portal.

CAPITULO 4

EXPOSICIÓN Y ANALISIS DE RESULTADOS

Para la implementación del sistema de evaluación docente se hizo el análisis respectivo utilizando los casos de uso. Los casos de uso se emplean para capturar el comportamiento deseado del sistema en desarrollo, sin tener que especificar cómo se implementa ese comportamiento. Los casos de uso proporcionan un medio para que los desarrolladores, los usuarios finales del sistema y los expertos del dominio lleguen a una comprensión común del sistema.

Los casos de uso son uno de los elementos que tiene el lenguaje UML, éste lenguaje nos permite comunicar de manera estandarizada el modelo de análisis del sistema que se está abstrayendo, de manera que los que visualicen el modelo puedan entender con precisión lo que significa.

Un elemento anexo al caso de uso es el actor que desencadena o inicia el caso de uso. El caso de uso es simbolizado con una elipse con el nombre en su interior y el actor con el gráfico de una persona que interactúa con el caso de uso.

4.1. Diagrama de Caso de Uso

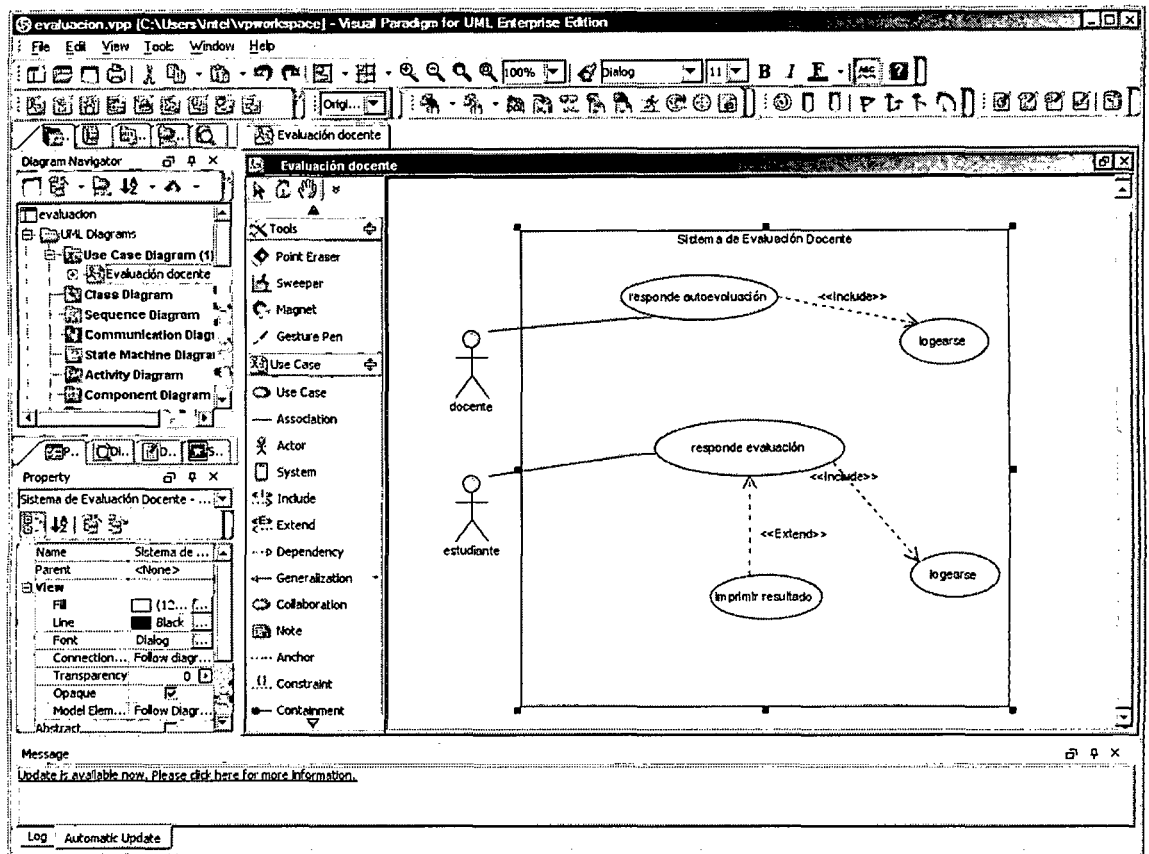


Figura 6. Entorno de “Visual Paradigm” .

Caso de Uso:

Responde Auto Evaluación .- el docente responde un cuestionario, este caso de uso hace uso de otro caso de uso con el estereotipo <<include>> que consiste en “loguearse” es decir ingresar sus credenciales en el sistema de acuerdo a las políticas de seguridad implementadas en el sistema.

El actor que activa el caso de uso es el docente que usa el sistema de evaluación docente. Deberá estar registrado o en su defecto podrá registrarse en el sistema.

Caso de Uso:

Responde Evaluación.- Este caso de uso se refiere al hecho de que el estudiante responde el cuestionario de respuestas múltiples que el sistema proporciona para que el estudiante evalúe al docente en determinada asignatura. El actor que desencadena el caso de uso en este caso es el estudiante.

El estudiante deberá estar previamente registrado o en su defecto deberá poder registrarse con las credenciales que lo identifican como estudiante válido, de acuerdo a las políticas adoptadas para la seguridad del sistema.

Este caso de uso también incluye a “loguearse” que significa ingresar al sistemas con el usuario y clave asignada por el sistema.

También se tiene el caso de uso “Imprimir resultado” que tiene el estereotipo “extend” que significa que opcionalmente se podrá imprimir resultado de la evaluación.

Cabe señalar que este caso de uso tiene como actor al estudiante. Por otro lado se tiene el actor “Administrador del Sistema” que en este caso el desarrollador que puede configurar el sistema de acuerdo a los requerimientos que se puedan presentar.

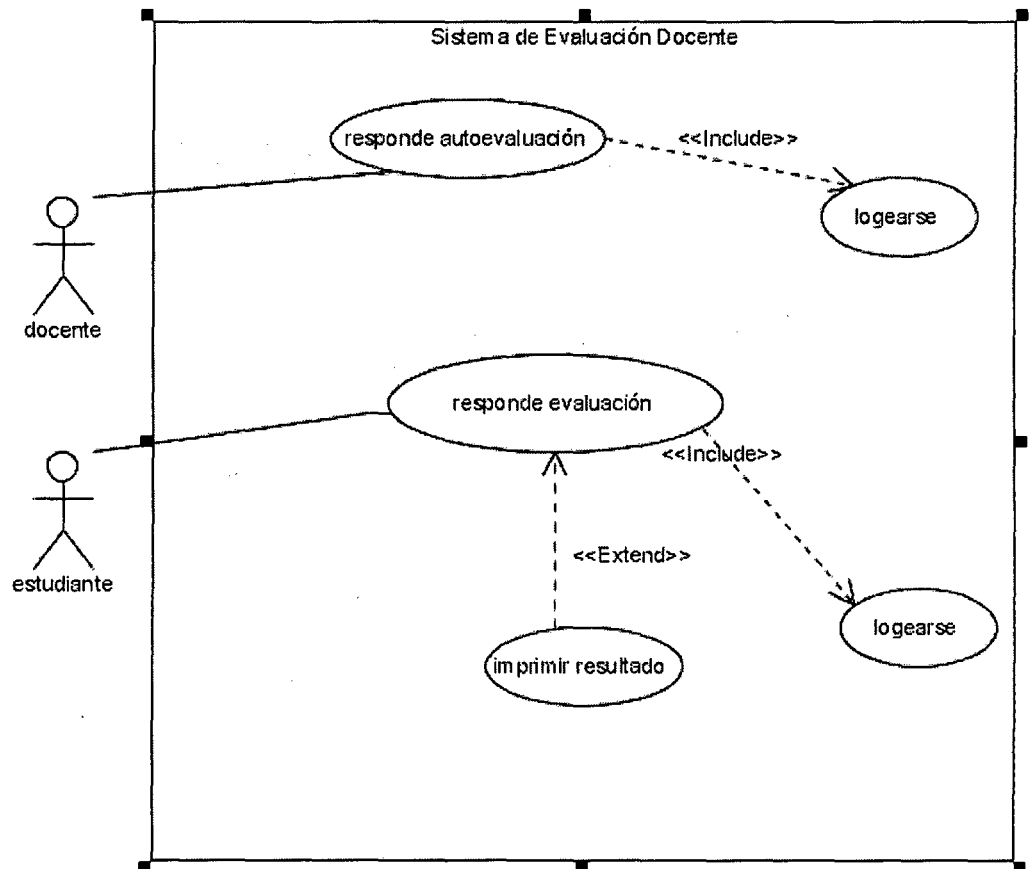


Figura 7. Principales Casos de Uso del Sistema.

4.2. Diagrama de Secuencia.

Un diagrama de secuencia es un diagrama de interacción que resalta la ordenación temporal de los mensajes. Un diagrama de secuencia presenta un conjunto de roles y los mensajes enviados y recibidos por las instancias que interpretan los roles. Los diagramas de secuencia se utilizan para describir la vista dinámica del sistema.

El principal diagrama de secuencia del sistema se produce cuando el estudiante pretende ingresar al sistema para evaluar al docente. Ingresa al sistema, el mismo que verifica que consta en la base

de datos. Si retorna que es encontrado, puede proceder a responder las preguntas que el sistema presenta para la respectiva evaluación del docente. El sistema presenta recursivamente las preguntas necesarias y previstas con antelación. Una vez concluido el ciclo iterativo de presentar preguntas se procede a dar por terminado.

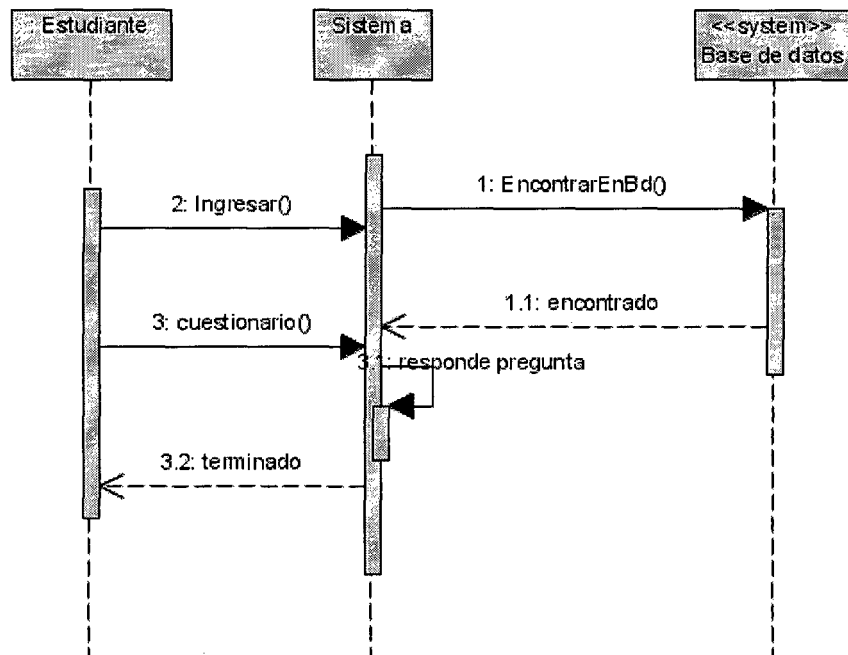


Figura 8. Diagrama de secuencia de evaluar por estudiante.

El diagrama de secuencia para la auto evaluación por parte del mismo docente es prácticamente similar, con el hecho de que es que inicia el proceso es el docente que pretende autoevaluarse. También se debe considerar que debe autenticarse en el sistema que internamente verificará sus credenciales para que pueda responder el cuestionario.

En cuanto al administrador del sistema debe seguir una secuencia similar para hacer el mantenimiento del sistema, ya sea para un

mantenimiento correctivo de algún requerimiento adicional o alguna mejora del sistema.

4.3.- Modelo de base de Datos

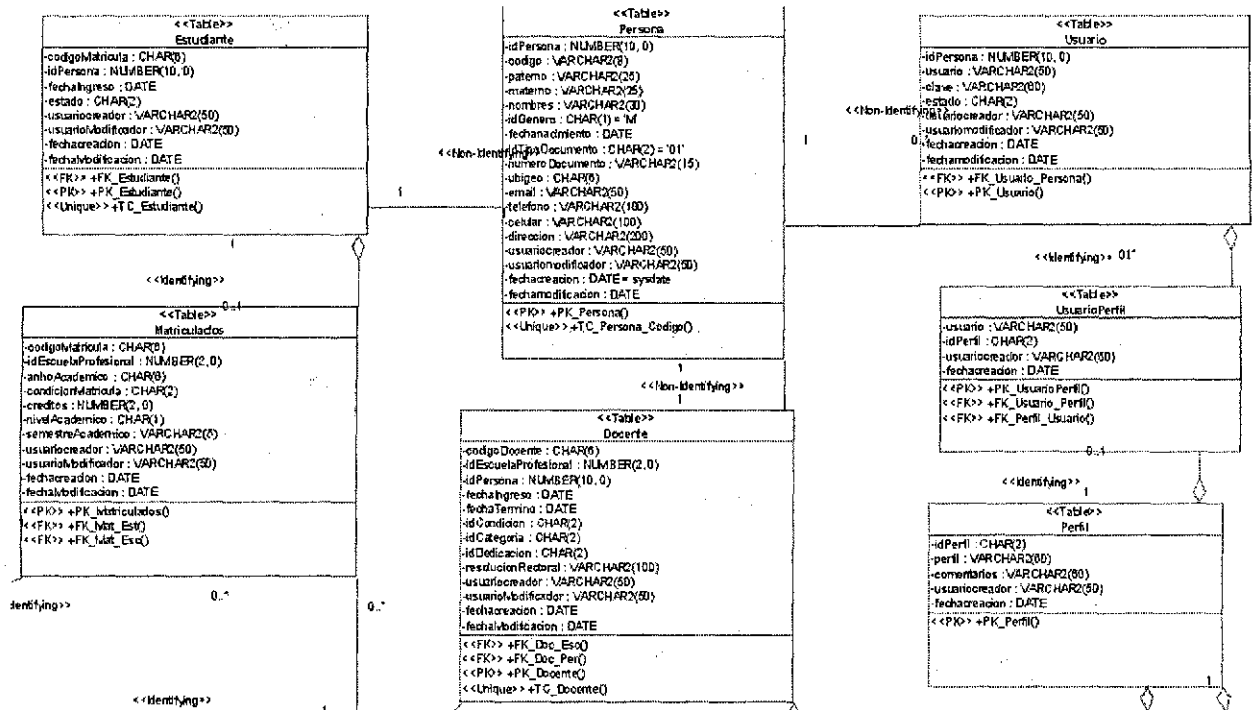


Figura 9 Modelo físico de la Base de datos

La Base de Datos del sistema considera entidades para Estudiante, Docente, que luego se concretan en el modelo físico en forma de tablas de la base de datos.

El modelamiento de la base de datos parte de abstraer la realidad para comprender mejor el problema y plantear una solución. El modelo entidad relación parte de la realidad tomando en cuenta los objetos reales del dominio del problema. Seguidamente se echa mano del modelo relacional, que principalmente consiste en tablas que en sus columnas representa los atributos y las filas las entidades que en el caso propio del modelo relacional son relaciones, es decir las filas son relaciones entre los

conjuntos de los dominios modelados. Para mantener la integridad de los datos se consideran las restricciones de dominio en las columnas, las restricciones de entidad con la clave primaria en las filas. Entre tablas se implementa la integridad referencial que garantiza que una tabla padre debe garantizar la consistencia de las tablas derivadas o dependientes.

En el entorno de Joomla se usa el prefijo jos_ para las diferentes tablas en MySQL que es el gestor de base de datos predeterminado. En el sistema se tiene una vista de las tablas usando Navicat.

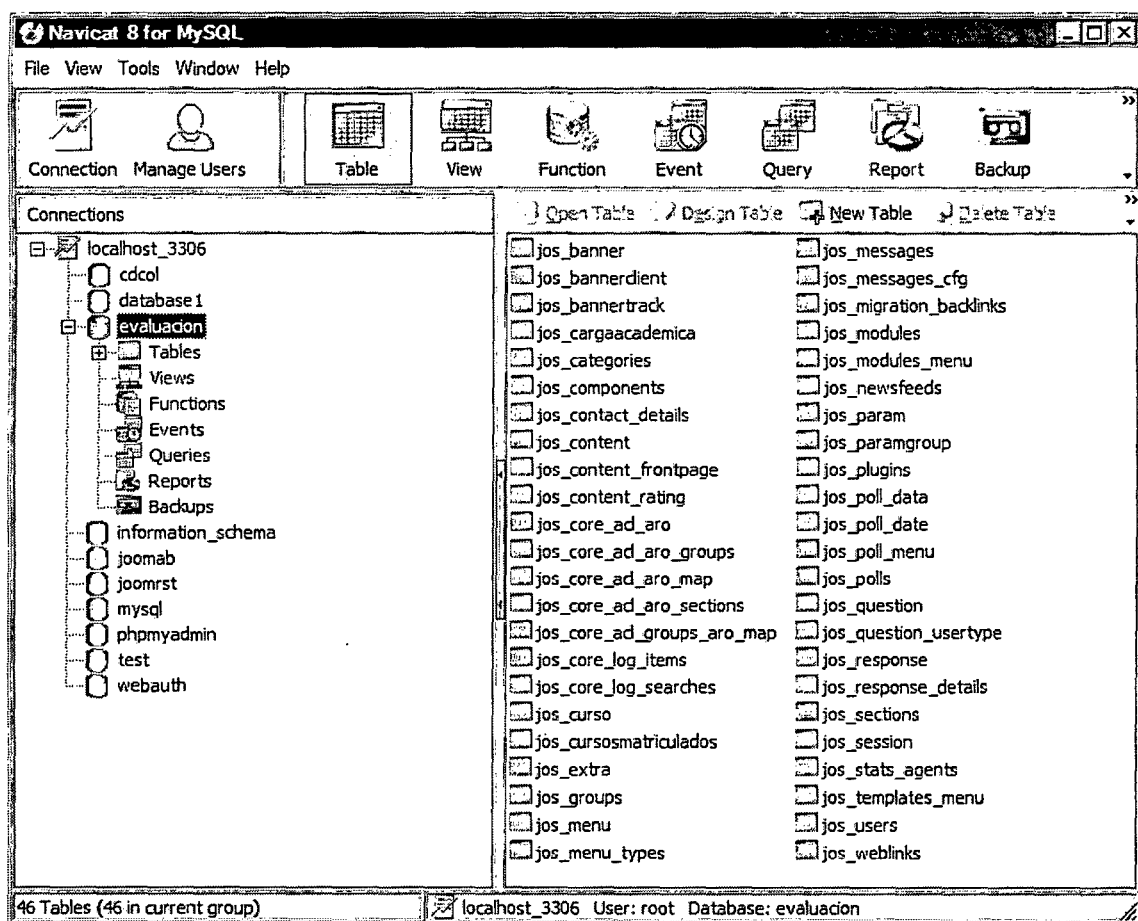


Figura 10 Una Vista de la Base de Datos en MySQL.

La Base de datos se denomina Evaluación y está alojado en MySQL que es un Sistema de Gestión de Bases de Datos, que garantiza la seguridad tomando en cuenta los usuarios y privilegios que son necesarios para acceder al sistema.

4.4. Interfaz de la Aplicación.

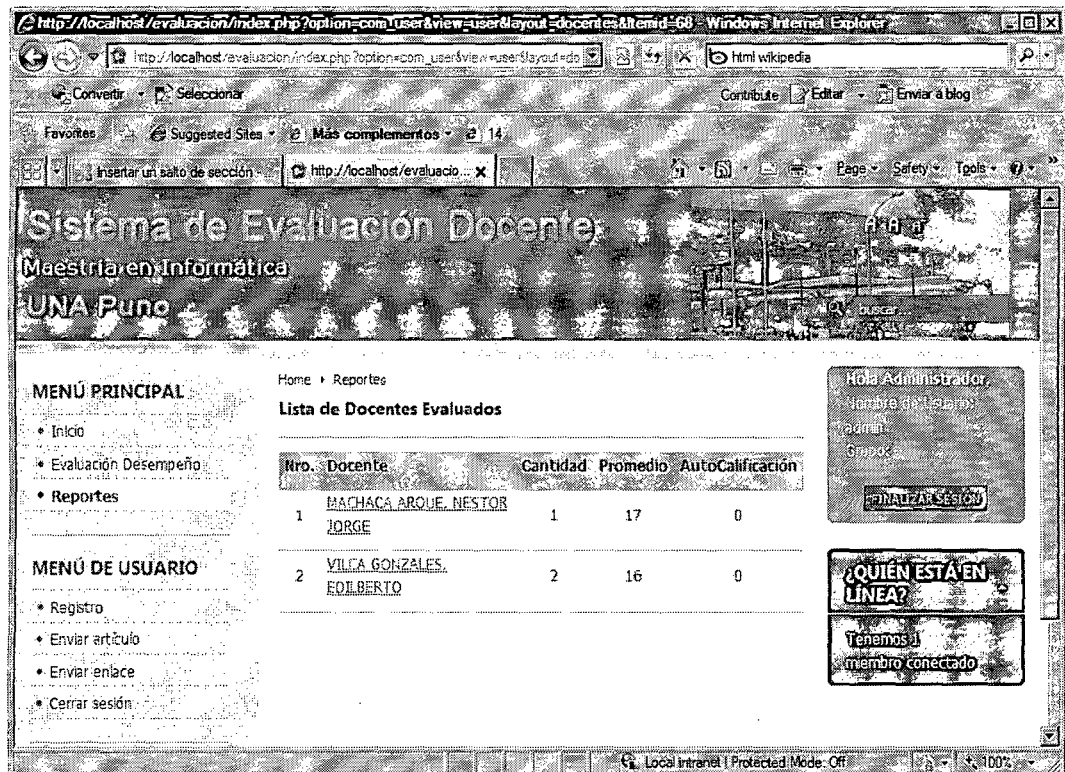


Figura 11 Interfaz de la aplicación

La aplicación tiene una interfaz para ingresar al sistema, que considera las políticas de seguridad mínimas, las cuales consideran el usuario o login, la contraseña o password. Para el administrador se muestra una interfaz con opciones de ejecutar reportes. El administrador también puede ingresar directamente al modo de administración de Joomla, para que pueda tener acceso a

las bases de datos y los diferentes elementos de la aplicación para editarlos y modificar el comportamiento del sistema.

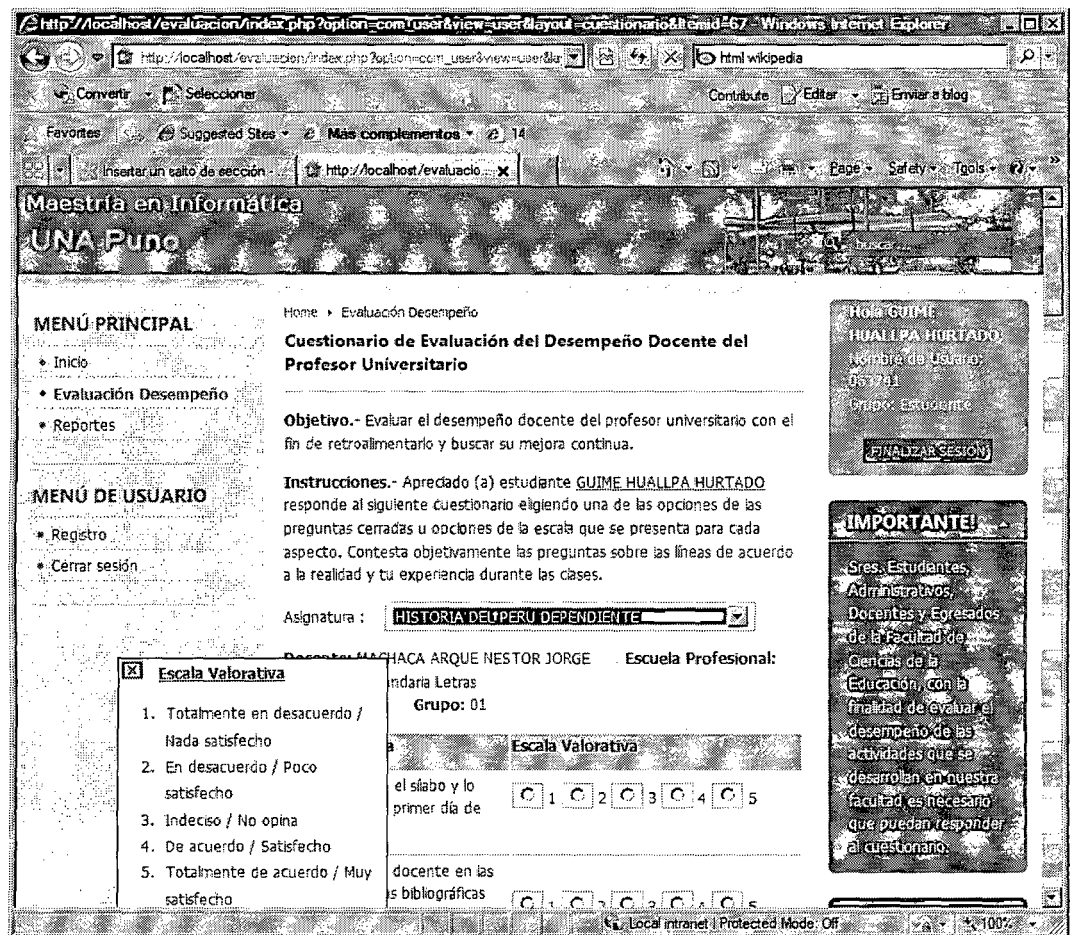


Figura 12 Interfaz para estudiante

El usuario estudiante tiene acceso al módulo que muestra la evaluación. Como ayuda se tiene un cuadro emergente que indica la valoración. Se presenta los ítems de preguntas que en forma de (radio button) selección única por pregunta debe ser seleccionada y al final almacenada. La interfaz es intuitiva y con ayuda de mensajes apropiados.

CONCLUSIONES

Primera.- La evaluación del desempeño docente en la UNA Puno es una actividad compleja que involucra muchos aspectos referidos al docente, instrumentos utilizados por el evaluador, criterios evaluados en el docente sujeto de evaluación, fiabilidad y oportunidad de los resultados.

Segunda.- El estado de arte de las tecnologías para el desarrollo de aplicaciones Web permite en la actualidad incorporar requisitos de seguridad, oportunidad, facilidad de acceso, navegabilidad, confiabilidad, así como cumplir con la función para la cual fue diseñada, la estructura arquitectónica implantada, y el desempeño esperado.

Tercera.- Los sistemas de gestión de contenidos CMS son una herramienta que reúne los elementos necesarios para desarrollar una aplicación web, es el caso de Joomla, que en combinación con un gestor de bases de datos como MySQL, el lenguaje de programación PHP, y java script permiten implementar la aplicación web para aplicar el instrumento de evaluación para el desempeño docente en la UNA Puno.

Cuarta.- El sistema de evaluación docente basado en aplicación Web, desarrollado, superó la prueba de contenido, la prueba de interfaz, la prueba de navegación, la prueba de componentes, la prueba de configuración, la prueba de seguridad y de desempeño; al registrar a más de 1000 estudiantes y 93 docentes registrados, los cuales probaron satisfactoriamente el sistema.

RECOMENDACIONES

Primera.- Se recomienda proponer el sistema de evaluación basado en aplicación Web, a la W3C (World Wide Web Consortium) para lograr su certificación WCAG 1.0 (Web Content Accesibility Guidelines); esta es una certificación internacional independiente elaborada en base a una metodología de evaluación e inspección que permite detectar problemas de accesibilidad combinado con herramientas de revisión automática y manual.

Segunda.- Se recomienda el uso del sistema de evaluación basado en aplicación Web, en las diferentes escuelas de la Universidad Nacional del Altiplano, de manera progresiva hasta lograr la generalización de su uso, con el aval de los órganos de gobierno de las facultades.

Tercera.- Los resultados del sistema de evaluación, pueden ser considerados en el proceso de renovación de contratos de docentes, con el arbitraje de los jefes de departamento, directores de estudios y la Oficina de Tecnología e Informática de la UNA.

BIBLIOGRAFIA

1. BAZIAN Menachem, BOOTH Jim (1999) **"Visual Fox Pro 6"**. Edición Especial. Edit. QUE. Mexico.
2. BOOCH, Grady (1996) **"Análisis y Diseño Orientado A Objetos"**. Segunda Edición. Edit. Addison-Wesley Iberoamericana. Delaware E.U.A.
3. BOOCH Grady y otros.(2006) "El Lenguaje Unificado de Modelado". Segunda Edición Addison Wesley. España.
4. CEBALLOS Javier (1999) **"Visual C++ Programación Avanzada en Win 32"**. Edit RAMA. Madrid.
5. DEITEL Harvey y DEYTEL Paul. (2004). "Cómo Programar en C/C++ y Java" Cuarta edición. Prentice Hall. México.
6. DuBOIS Paul. (2009). "La Biblia de MySQL" Cuarta edición. Ediciones Anaya multimedia. España.
7. ESPOSITO Dino (2005) **"INTRODUCING MICROSOFT® ASP.NET 2.0"** Microsoft Press. USA.
8. HERNANDEZ SAMPIERI Roberto y otros. (2006) "Metodología de la Investigación" Mc Graw Hill Interamericana. Cuarta edición. México.
9. KENDALL Kenneth E. y KENDALL Julie E. (2005) "Análisis y Diseño de Sistemas" Sexta Edición. Prentice Hall. México.
10. KOFFMAN Elliot y WOLFGANG Paul. (2008). "Estructura de datos con C++. Objetos, Abstracciones y Diseño" Tercera Edición. Mc Graw Hill Interamericana. México.
11. KRUGLINSKI David, SHEPERD George (1999) **"Programación Avanzada con Visual C++"**. Quinta Edición. Edit. Microsoft Press. Madrid.
12. MATTHEW MacDonald (2006) **"Beginning Asp.Net 2.0 In C# 2005: From Novice To Professional"** Primera Edición. APress. USA.
13. McFEDRIES Paul (1997) **"Creando una Página Web con HTML Fácil"**. Segunda edición. Edit. QUE. Mexico.

14. MENDOZA, F. MENDOZA R, (1996) **“Estadística para Ciencias de la Educación”**, Edit. Titikaka. Puno.
15. NAGHI NAMAKFOROOSH, Mohammad (2002) **“Metodología de la Investigación”** Segunda Edición. Ed. Limusa. Mexico.
16. PALOMINO QUISPE, Platón (1997) **“Diseños y Técnicas de Investigación”** Segunda Edición. Edit. Titikaka. FCEDUC- UNA- Puno.
17. PRESSMAN S., Roger (2005) **“Ingeniería del Software. Un enfoque Práctico”**. Quinta Edición. Edit. McGraw-Hill. España.
18. HAWRYSZKIEWYCZ (1997) **“Análisis y Diseño de Bases de Datos”**, Ed. Megabyte Noriega Editores.
19. SHREVES Ric (2010) **“Joomla! Bible”** Wiley Publishing, Inc. USA.
20. SILER Brian, SPOTS Jeff (1999) **“Visual Basic 6”**. Edición Especial. Edit. QUE. Madrid.
21. SILVERSCHATZ Abraham y otros. (2006). **“Fundamentos de Bases de Datos”**. Quinta Edición. Mc Graw Hill Interamericana. España.
22. SOMMERVILLE, Ian (2002) **“Ingeniería de Software”**. Sexta Ed. Pearson Educación. México.
23. TANENBAUM Andrew (1996) **“Redes de Computadoras”**. Tercera Edición. Edit. Prentice Hall Hispanoamericana. Mexico.
24. ZARATIAN, Beck (1999) **“Microsoft Visual C++ 6.0 Manual del Programador”**, McGraw Hill, España.

ANEXOS

ANEXO A

VICE RECTORADO ACADÉMICO

INSTRUMENTO DE EVALUACIÓN FORMATIVA AL DESEMPEÑO DOCENTE

DOCENTE:	ESCUELA PROFESIONAL:	FECHA:
----------	----------------------	--------

A continuación, presentamos diversos criterios de evaluación sobre el desempeño docente, favor indicar su opinión para cada uno, marcando con X su elección en la escala siguiente:

ESTUDIANTE

RESPONSABILIDAD PERSONAL

Criterio de evaluación	Nunca lo hace	Algunas veces lo hace	Casi siempre lo hace	Siempre lo hace
P29 El docente está asistiendo puntualmente a sus clases.	0	1	2	3
F30 El docente viene a clases bien presentable.	0	1	2	3
F31 Práctica valores de autoestima, solidaridad y cordialidad.	0	1	2	3
F32 El docente viene a clases en estado ecuaníme.	0	1	2	3
F33 Ha entregado puntualmente el silabo de la asignatura al delegado del aula.	0	1	2	3

ENSEÑANZA-APRENDIZAJE

Criterio de evaluación	Nunca lo hace	Algunas veces lo hace	Casi siempre lo hace	Siempre lo hace
P34 Se hace entender el docente en el desarrollo de los contenidos de la asignatura.	0	1	2	3
P35 Se preocupa por saber si los alumnos han aprendido y refuerza los aprendizajes cuando es necesario.	0	1	2	3
P36 Está desarrollando el contenido de las asignaturas por lo menos en un 80% de los mismos.	0	1	2	3
P37 Tiene dominio, aceptable, de los contenidos del curso que imparte.	0	1	2	3
P38 Usa diversos medios y materiales para la enseñanza-aprendizaje.	0	1	2	3
P39 Está promoviendo el uso de textos, libros, revistas e internet constantemente.	0	1	2	3
P40 Investiga para el desarrollo de las asignaturas a su cargo.	0	1	2	3

EVALUACIÓN DEL APRENDIZAJE

Criterio de evaluación	Nunca lo hace	Algunas veces lo hace	Casi siempre lo hace	Siempre lo hace
P41 Los contenidos de las evaluaciones corresponden a los contenidos desarrollados en el curso.	0	1	2	3
P42 Es objetivo y justo en las calificaciones de los exámenes y tareas del aprendizaje.	0	1	2	3
P43 Utiliza el docente los resultados de las evaluaciones del aprendizaje, para hacer la reevaluación de sí mismo.	0	1	2	3
P44 Entrega los resultados de los exámenes a los alumnos en los plazos establecidos.	0	1	2	3

CLIMA EN EL AMBIENTE DE APRENDIZAJE (AULA/LABORATORIO/TALLER, ETC.)

Criterio de evaluación	Nunca lo hace	Algunas veces lo hace	Casi siempre lo hace	Siempre lo hace
P45 Propicia la participación al debate y la discusión.	0	1	2	3
P46 Respeto las opiniones diferentes a las suyas y atiende las dudas e inquietudes de los alumnos.	0	1	2	3
P47 Promueve y facilita el orden y la limpieza en el ambiente de aprendizaje.	0	1	2	3
P48 Tiene un trato respetuoso con los alumnos, sin agredir su dignidad y autoestima personal.	0	1	2	3

AREAS DE DOMINIO ACADEMICO PARA LA EVALUACIÓN DOCENTE

Este anexo se constituye en un factor organizador y orientador para que el docente pueda orientar su capacitación y perfeccionamiento profesional y ético para su desenvolvimiento académico en la Universidad y desempeño en aula.

CRITERIOS E INDICADORES	Escala de Calificación			
	1	2	3	4
<i>a) Dominio Académico Profesional</i>				
<ul style="list-style-type: none"> ▪ Especialista en un área académico y profesional. ▪ Capacitado y actualizado en su área de trabajo académico. ▪ Creador y aplicador del conocimiento científico, técnico y humanista. ▪ Relaciona el conocimiento con los problemas regionales y nacionales. 				
<i>b) Dominio Pedagógico</i>				
<ul style="list-style-type: none"> ▪ Muestra didáctica en el aula. ▪ Crítico, creativo e innovador. ▪ Comunicador, orientador y facilitador del aprendizaje ▪ Creador de condiciones y situaciones educativas para desarrollar el proceso de enseñanza – aprendizaje y solucionar problemas académicos. ▪ Trabaja con métodos y técnicas del aprendizaje grupal. ▪ Conocedor y aplicador de los conocimientos de informática, idiomas e investigación científica en el aprendizaje. 				
<i>c) Dominio de las relaciones humanas</i>				
<ul style="list-style-type: none"> ▪ Es ético: honesto, responsable, disciplinado, justo, cordial y leal. ▪ Practica los valores de autoestima, de la solidaridad, de la cordialidad, la responsabilidad en clase y fuera de ella. ▪ Realiza trabajo en equipo y crea oportunidades para planificar las actividades educativas y del PEA. ▪ Es amigo de sus alumnos, los considera como personas en formación que necesitan afecto, buen humor y ambiente positivo. 				
<i>d) Cumplimiento de normatividad</i>				
<ul style="list-style-type: none"> ▪ Conoce y aplica las disposiciones contenidas en la Constitución Política del Perú y la Ley Universitaria, que se relaciona con la labor de maestro. ▪ Conoce y aplica las disposiciones contenidas en el Estatuto UNA, el MOF, los reglamentos internos y la directiva académica anual. ▪ Conoce y aplica el Reglamento de Evaluaciones del Aprendizaje de la UNA. ▪ Soluciona problemas administrativos relacionados con la enseñanza – aprendizaje. 				

Leyenda:

(1) Deficiente (2) Regular (3) Bueno (4) Excelent

ANEXO B

Codigo ejemplo de una tabla

```
-- phpMyAdmin SQL Dump
-- version 3.1.3.1
-- http://www.phpmyadmin.net
--
-- Servidor: localhost
-- Tiempo de generación: 02-12-2010 a las 12:24:33
-- Versión del servidor: 5.1.33
-- Versión de PHP: 5.2.9

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";

--
-- Base de datos: `evaluacion`
--

--
-- Estructura de tabla para la tabla `jos_cargaacademica`
--

CREATE TABLE IF NOT EXISTS `jos_cargaacademica` (
  `codigoDocente` varchar(15) COLLATE latin1_spanish_ci NOT NULL,
  `idEscuelaProfesional` tinyint(2) unsigned NOT NULL,
  `codPlan` char(2) COLLATE latin1_spanish_ci NOT NULL,
  `idCurso` varchar(6) COLLATE latin1_spanish_ci NOT NULL,
  `anhoAcademico` char(6) COLLATE latin1_spanish_ci NOT NULL,
  `idGrupo` char(2) COLLATE latin1_spanish_ci NOT NULL
  DEFAULT '01',
  `turno` char(1) COLLATE latin1_spanish_ci DEFAULT 'M',
  `aula` varchar(60) COLLATE latin1_spanish_ci DEFAULT NULL,
  `usuarioCreador` varchar(15) COLLATE latin1_spanish_ci NOT
  NULL,
  `fechaCreacion` datetime DEFAULT '0000-00-00 00:00:00',
  `usuarioModificador` varchar(15) COLLATE latin1_spanish_ci
  DEFAULT '',
  `fechaModificacion` datetime DEFAULT '0000-00-00 00:00:00',
  PRIMARY KEY
  (`codigoDocente`,`idEscuelaProfesional`,`codPlan`,`idCurso`,`
  anhoAcademico`,`idGrupo`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
  COLLATE=latin1_spanish_ci;

--
-- Volcar la base de datos para la tabla `jos_cargaacademica`
--

INSERT INTO `jos_cargaacademica` (`codigoDocente`,
  `idEscuelaProfesional`, `codPlan`, `idCurso`,
  `anhoAcademico`, `idGrupo`, `turno`, `aula`,
  `usuarioCreador`, `fechaCreacion`, `usuarioModificador`,
  `fechaModificacion`) VALUES
  ('2004642', 16, '03', '043', '200901', '01', 'M', '102
  Pabellón de Ingeniería de Sistemas', 'wearcaya', '0000-00-00
  00:00:00', '', '0000-00-00 00:00:00'),
```

('86127', 16, '03', '045', '200901', '01', 'M', '201-B
Pabellón de Ingeniería de Sistemas', 'wearcaya', '0000-00-00
00:00:00', '', '0000-00-00 00:00:00'),
('87087', 17, '03', '045', '200901', '01', 'M', '201-A
Pabellón de Ingeniería de Sistemas', 'wearcaya', '0000-00-00
00:00:00', '', '0000-00-00 00:00:00'),
('86127', 17, '03', '046', '200901', '01', 'T', '16 Pabellón
Educación', 'wearcaya', '0000-00-00 00:00:00', '', '0000-00-
00 00:00:00'),
('2005746', 17, '03', '048', '200901', '01', 'T', '101
Pabellón de Ingeniería de Sistemas', 'wearcaya', '0000-00-00
00:00:00', '', '0000-00-00 00:00:00'),
('2061015', 16, '03', '048', '200901', '01', 'M', '104
Pabellón de Ingeniería de Sistemas', 'wearcaya', '0000-00-00
00:00:00', '', '0000-00-00 00:00:00'),
('2005754', 16, '03', '049', '200901', '01', 'T', '15 Pabellón
de Ciencias de la Educación', 'wearcaya', '0000-00-00
00:00:00', '', '0000-00-00 00:00:00'),
('29973', 17, '03', '049', '200901', '01', 'M', '09 Pabellón
de Colegio de Aplicación', 'wearcaya', '0000-00-00 00:00:00',
'', '0000-00-00 00:00:00'),
('2004642', 16, '03', '050', '200901', '01', 'T', '104
Pabellón de Ingeniería de Sistemas', 'wearcaya', '0000-00-00
00:00:00', '', '0000-00-00 00:00:00'),
('87087', 17, '03', '051', '200901', '01', 'T', '101 Pabellón
de Ingeniería de Sistemas', 'wearcaya', '0000-00-00
00:00:00', '', '0000-00-00 00:00:00'),
('86127', 16, '03', '052', '200901', '01', 'M', '302 Pabellón
de Ingeniería de Sistemas', 'wearcaya', '0000-00-00
00:00:00', '', '0000-00-00 00:00:00'),
('86127', 17, '03', '052', '200901', '01', '', '',
'wearcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2005749', 16, '03', '054', '200901', '01', '', '',
'wearcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('970412', 17, '03', '054', '200901', '01', '', '',
'wearcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2004642', 17, '03', '055', '200901', '01', '', '',
'wearcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('900118', 16, '03', '056', '200901', '01', '', '',
'wearcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('950541', 17, '03', '058', '200901', '01', '', '',
'wearcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2007916', 16, '03', '059', '200901', '01', '', '',
'wearcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2007917', 17, '03', '059', '200901', '01', '', '',
'wearcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2007916', 16, '03', '060', '200901', '01', '', '',
'wearcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('86085', 17, '03', '060', '200901', '01', '', '',
'wearcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),

('2005504', 17, '03', '061', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2005119', 17, '03', '062', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2007917', 16, '03', '062', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2005504', 17, '03', '063', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2005751', 16, '03', '063', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2061016', 16, '03', '064', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('89103', 16, '03', '065', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2007917', 17, '03', '066', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('86085', 17, '03', '067', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('89103', 16, '03', '067', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2007917', 16, '03', '068', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2071005', 17, '03', '068', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('970412', 17, '03', '069', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('200377', 17, '03', '070', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2007917', 16, '03', '070', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2005751', 16, '03', '071', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('920654', 17, '03', '071', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2061016', 16, '03', '072', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('920654', 17, '03', '072', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),
('2007916', 17, '03', '075', '200901', '01', '', '',
'warcaya', '0000-00-00 00:00:00', '', '0000-00-00
00:00:00'),