



**UNIVERSIDAD ESTATAL
PENÍNSULA DE SANTA ELENA**

**FACULTAD DE CIENCIAS ADMINISTRATIVAS
ESCUELA DE INFORMÁTICA**

**“DESARROLLO DEL PORTAL WEB DE LA UNIVERSIDAD
ESTATAL PENÍNSULA DE SANTA ELENA UTILIZANDO
SOFTWARE LIBRE”**

TESIS DE GRADO

Previo a la obtención del Título de:

INGENIERO EN SISTEMAS

AUTORES: TORRES VIVANCO LUIS

CATUTO ROCA JIMMY

TUTOR: ING. WALTER OROZCO

LA LIBERTAD – ECUADOR

2010

**UNIVERSIDAD ESTATAL
PENÍNSULA DE SANTA ELENA**

FACULTAD DE CIENCIAS ADMINISTRATIVAS
ESCUELA DE INFORMÁTICA

**“DESARROLLO DEL PORTAL WEB DE LA UNIVERSIDAD
ESTATAL PENÍNSULA DE SANTA ELENA UTILIZANDO
SOFTWARE LIBRE”**

TESIS DE GRADO

Previo a la obtención del Título de:

INGENIERO EN SISTEMAS

AUTORES: TORRES VIVANCO LUIS

CATUTO ROCA JIMMY

TUTOR: ING. WALTER OROZCO

LA LIBERTAD – ECUADOR

2010

La Libertad, 8 de Junio del 2010.

APROBACIÓN DEL TUTOR

En mi calidad de Tutor del Trabajo de Investigación, “DESARROLLO DEL PORTAL WEB DE LA UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA UTILIZANDO SOFTWARE LIBRE” elaborado por los Srs. Jimmy Stanley Catuto Roca y Luis Vinicio Torres Vivanco, Egresados de la Escuela de Informática, Facultad de Ciencias Administrativas de la Universidad Estatal Península de Santa Elena, previo a la obtención del Título de Ingenieros en Sistemas, me permito declarar que luego de haber orientado, estudiado y revisado, la apruebo en todas sus partes.

Atentamente

.....
Ing. Walter Orozco I.
TUTOR

DEDICATORIA

A mis Hijos, quienes motivan mi existencia y mi lucha; a mi Madre, combinación casi divina de bondad, ternura y tenacidad; y a mi Padre por enseñarme que la vida sí da segundas oportunidades.

Luis.

DEDICATORIA

A mi querida Madre, quien ha sido mi principal fuente de motivación, y quien jugó un papel muy importante en mi vida en los momentos en que empezaba mis estudios superiores.

Jimmy

AGRADECIMIENTO

A todas las personas que de una u otra manera aportaron para lograr esta meta, a mis Padres, Hermanos, Profesores y Amigos, gracias!.

Luis.

AGRADECIMIENTO

A cada una de las personas que a través de los años han aportado en mi desarrollo personal, académico y profesional. A cada familiar, amigo, compañero de aula, compañero de trabajo, maestro ¡Muchísimas Gracias!

Jimmy.

TRIBUNAL DE GRADO

**EC. CARLOS CASTILLO GALLO
DECANO DE LA FACULTAD
C.C. ADMINISTRATIVAS**

**ING. FREDDY VILLAO S.
DIRECTOR DE LA ESCUELA
DE INFORMÁTICA**

**ING. WALTER OROZCO I.
TUTOR**

**ING. MARIUXI DE LA CRUZ
PROFESOR DEL ÁREA**

**AB. MILTON ZAMBRANO CORONADO, MSc.
SECRETARIO GENERAL - PROCURADOR**

**UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA
FACULTAD DE CIENCIAS ADMINISTRATIVAS
ESCUELA DE INFORMÁTICA**

**“DESARROLLO DEL PORTAL WEB DE LA UNIVERSIDAD ESTATAL
PENÍNSULA DE SANTA ELENA UTILIZANDO SOFTWARE LIBRE”**

AUTORES: TORRES VIVANCO LUIS VINICIO
CATUTO ROCA JIMMY STANLEY

TUTOR: OROZCO I. WALTER ARMANDO

RESUMEN

La constante evolución de las tecnologías aplicables al desarrollo de software hacen casi obligatoria la continua renovación de los sitios web, hoy en día las instituciones de cualquier tipo, ya sean comerciales, estatales o sociales, aprovechan las potencialidades que brinda el poseer un medio de interacción directo con los consumidores de los servicios que proveen, más aún en las entidades de carácter educativo, ya que en el nivel en que se encuentran permiten establecer canales de comunicación directa con sus estudiantes, docentes o empleados. En este Trabajo de Investigación hemos querido proporcionar a la Universidad Estatal Península de Santa Elena, una plataforma que integra varias herramientas, las cuales son de código abierto, para soportar sobre ella el nuevo portal web universitario, brindando la posibilidad de mantenerla con contenido dinámico, añadiendo funcionalidades como consultas del historial académico, acceso al contenido existente en la biblioteca, correo electrónico y dando apertura a que se sigan añadiendo a ella, las funcionalidades que demande la institución, dando esa sensación de cercanía con los directivos de la institución que es muy valorada por cada uno de los estudiantes y estableciendo las directivas de seguridad necesarias en este tipo de aplicaciones. El Capítulo I presenta la misión y visión de la Universidad Estatal Península de Santa Elena en la cual está enfocado el trabajo realizado, también proporciona una referencia teórica sobre los contenidos básicos que permiten profundizar los conocimientos sobre el tema tratado en este trabajo de investigación, y de esta manera poder conocer plenamente el significado de cada uno de los componentes de la plataforma. En la Metodología de Investigación, mostrada en el Capítulo II, se indican las técnicas de investigación utilizadas para la recolección y análisis de la información, lo que finalmente nos permitió establecer las necesidades que se debían satisfacer una vez que se implementara la plataforma. Finalmente, en el Capítulo III, se define el modelo conceptual de la plataforma, se establece la forma en que interactúan los componentes entre sí, se detallan las características que deben tener las herramientas para ser consideradas en el proyecto, y se determina el procedimiento seguido en la selección las más óptimas, para luego ser integradas utilizando los procedimientos detallados en los anexos.

ÍNDICE GENERAL

	Pág.
PORTADA	i
APROBACIÓN DEL TUTOR	ii
DEDICATORIAS	iii
AGRADECIMIENTOS	v
TRIBUNAL DE GRADO	vii
RESUMEN	viii
ÍNDICE GENERAL	ix
ÍNDICE DE CUADROS	xii
ÍNDICE DE GRÁFICOS	xiv
ÍNDICE DE ANEXOS	xvii
INTRODUCCIÓN	1
CAPÍTULO I	
1. MARCO TEÓRICO	3
1.1. Universidad Estatal Península de Santa Elena	3
1.1.1. Misión	3
1.1.2. Visión	3
1.1.3. Filosofía	3
1.1.4. Antecedentes	4
1.1.5. Objetivos	4
1.2. Los Sistemas Informáticos	5
1.2.1. Teoría General de Sistemas	5
1.2.2. Los Sistemas Informáticos	6
1.2.3. Las Aplicaciones Web	7
1.2.4. Los Frameworks	9

1.2.5.	Los Servicios Web	10
1.2.6.	Los Servicios de Directorio	15
1.2.7.	Las Conexiones Seguras	17
1.2.8.	Los Servidores de Aplicaciones	21
1.3.	El Software Libre	21
1.3.1.	Licencias de Software	25
1.3.2.	Tipos de Licencia de Software Libre	25
1.3.3.	El Open Source	28
1.4.	Tecnologías de la Información y la Comunicación	31
1.4.1.	Las Redes Interconectadas	31

CAPÍTULO II

2.	METODOLOGÍA DE LA INVESTIGACIÓN	34
2.1.	Diseño de la Investigación	34
2.2.	Modalidad de la Investigación	34
2.3.	Tipos de Investigación	35
2.4.	Población y Muestra	36
2.4.1.	Población	36
2.4.2.	Muestra	36
2.5.	Operacionalización de las variables	38
2.5.1.	Variables de Estudio	39
2.6.	Técnicas e Instrumentos	41
2.6.1.	Técnicas	41
2.6.2.	Instrumentos	42
2.7.	Análisis de Resultados	42
2.7.1.	Resultado de la Encuesta al Personal Técnico	42
2.7.2.	Resultado de la Encuesta a los Usuarios de la Página Web	49
2.8.	Análisis e Interpretación de Resultados	56

CAPÍTULO III

3.	DESARROLLO DEL PORTAL WEB DE LA UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA UTILIZANDO SOFTWARE LIBRE	57
3.1.	Análisis de la Problemática	57
3.1.1.	Principales componentes de un Portal Web	58
3.1.2.	Detalle de los requerimientos del Portal Web	59
3.2.	Diseño de la Plataforma	61
3.2.1.	Modelo Conceptual	62
3.2.2.	Plataforma de Ejecución	65
3.3.	Selección de las Herramientas	66
3.3.1.	Procedimiento de selección	67
3.3.2.	Definición de los requerimientos	68
3.3.3.	Descripción de las características analizadas	72
3.3.4.	Evaluación de las herramientas pre-seleccionadas	78
3.3.5.	Resumen de aplicaciones seleccionadas	83
3.4.	Software Desarrollado	83
3.4.1.	Aplicaciones desarrolladas	83
3.4.2.	Modificaciones a los componentes de la plataforma	84
3.5.	Implementación de la Plataforma	85
3.6.	Pruebas	85
	CONCLUSIONES	86
	RECOMENDACIONES	87
	BIBLIOGRAFÍA	88
	ANEXOS	91
	GLOSARIO DE TÉRMINOS	310
	ABREVIATURAS	313

ÍNDICE DE CUADROS

		Pág.
CUADRO # 1	Población.	36
CUADRO # 2	Muestra.	38
CUADRO # 3	Operacionalización de la variable independiente.	39
CUADRO # 4	Operacionalización de la variable dependiente.	40
CUADRO # 5	Pago de licencias de uso.	43
CUADRO # 6	Límite de equipos.	43
CUADRO # 7	Límite de usuarios.	44
CUADRO # 8	Restricciones de uso.	44
CUADRO # 9	Disponibilidad del código fuente.	45
CUADRO # 10	Autorización de copia.	45
CUADRO # 11	Facilidad de publicación de contenido.	46
CUADRO # 12	Facilidad de clasificación de contenido.	46
CUADRO # 13	Facilidad de rediseño de la página web.	47
CUADRO # 14	Facilidad de publicación de contenido.	47
CUADRO # 15	Posibilidad de administrar usuarios y grupos.	48
CUADRO # 16	Almacenamiento del historial de visitas.	48
CUADRO # 17	Nivel de seguridad en el acceso a información.	49
CUADRO # 18	Cifras sobre la clasificación de la muestra.	50
CUADRO # 19	Cifras sobre la utilización de la página web.	51
CUADRO # 20	Acceso a material bibliográfico desde la página.	52
CUADRO # 21	Cifras sobre la reserva en línea de materiales bibliográficos.	53
CUADRO # 22	Cifras sobre la consulta en línea del record académico.	54
CUADRO # 23	Cifras sobre la creación de cuentas de correo electrónico.	55
CUADRO # 24	Pre-selección de herramientas para gestión de contenidos.	69
CUADRO # 25	Pre-selección de Sistemas Integrados de Biblioteca.	69
CUADRO # 26	Pre-selección de Herramientas para Webmail.	70
CUADRO # 27	Pre-selección de Servidores de Autenticación.	70
CUADRO # 28	Pre-selección de Servidores de Directorio.	71

CUADRO # 29	Pre-selección de Contenedores de Servlets.	71
CUADRO # 30	Aplicaciones evaluadas por características.	79
CUADRO # 31	Evaluación de los Sistemas de Administración de Contenidos y los Sistemas Integrados de Biblioteca.	80
CUADRO # 32	Evaluación de WebMails y Servidores de Autenticación.	81
CUADRO # 33	Evaluación de Servidores de Directorio y Servidor de Aplicación.	82
CUADRO # 34	Aplicaciones seleccionadas para la plataforma.	83
CUADRO # 35	Listado de objetos implícitos.	135
CUADRO # 36	Listado de objetos implícitos(continuación).	136
CUADRO # 37	Propiedades hyphen, pre y maxlength.	141
CUADRO # 38	Valores de la propiedad position.	150
CUADRO # 39	Eventos generados por el mouse.	160
CUADRO # 40	Eventos generados por el teclado.	161
CUADRO # 41	Eventos de entrada.	162
CUADRO # 42	Eventos de lista y árboles.	163
CUADRO # 43	Otros eventos.	163
CUADRO # 44	Otros eventos (continuación).	164
CUADRO # 45	Definición del método ObtenerInfoGeneral.	169
CUADRO # 46	Definición del método ObtenerCarrerasTomadas.	169
CUADRO # 47	Definición del método ObtenerMateriasCursadas.	169
CUADRO # 48	Definición de la clase MateriasCursadas.	170
CUADRO # 49	Definición de la clase Estudiantes.	171
CUADRO # 50	Definición de la clase CarrerasTomadas.	171

ÍNDICE DE GRÁFICOS

	Pág.
GRÁFICO # 1 Aplicaciones heterogéneas con necesidades comunes	12
GRÁFICO # 2 Esquema de autenticación	19
GRÁFICO # 3 Clasificación de la muestra	50
GRÁFICO # 4 Utilización de la página web	51
GRÁFICO # 5 Acceso a material bibliográfico desde la página	52
GRÁFICO # 6 Reserva en línea de materiales bibliográficos.	53
GRÁFICO # 7 Consulta en línea del récord académico.	54
GRÁFICO # 8 Creación de cuentas de correo electrónico.	55
GRÁFICO # 9 Modelo Conceptual de la Plataforma.	61
GRÁFICO # 10 Diagrama de flujo del proceso de registro de un Estudiante.	63
GRÁFICO # 11 Diagrama de la Plataforma de Ejecución.	65
GRÁFICO # 12 Ubicación del Sistema Operativo en una Estructura Informática.	66
GRÁFICO # 13 Distribución de la página principal.	97
GRÁFICO # 14 Distribución para cuando se muestra contenido.	99
GRÁFICO # 15 Trabajando las imágenes en Inkscape.	100
GRÁFICO # 16 Trabajando con GIMP.	100
GRÁFICO # 17 Selección de la plantilla.	101
GRÁFICO # 18 Ubicación de los módulos en las secciones de la plantilla.	102
GRÁFICO # 19 Apariencia inicial del Portal Web Universitario.	104
GRÁFICO # 20 Logo de ZK.	105
GRÁFICO # 21 Utilidad de instalación YaST.	107
GRÁFICO # 22 Página de descarga de Eclipse.	108
GRÁFICO # 23 Directorio de almacenamiento de proyectos.	108
GRÁFICO # 24 Definición del espacio de trabajo de Eclipse.	109
GRÁFICO # 25 Activación de ZK Studio.	110
GRÁFICO # 26 Selección de la versión del framework.	110
GRÁFICO # 27 Agregando un servidor a Eclipse.	111

GRÁFICO # 28	Estructura de aplicación según especificaciones API Servlet.	113
GRÁFICO # 29	Seleccionando el tipo de proyecto a importar.	113
GRÁFICO # 30	Importando un proyecto de ZK.	114
GRÁFICO # 31	Cambiando la versión del framework utilizada en el proyecto.	115
GRÁFICO # 32	Ambiente de Desarrollo ZK con una página en ejecución.	115
GRÁFICO # 33	Generando el archivo empaquetado.	118
GRÁFICO # 34	Acceso a la Aplicación de Consulta de Calificaciones.	119
GRÁFICO # 35	Integrando la aplicación a la plataforma operativa.	120
GRÁFICO # 36	Asignando la aplicación al menú de Servicios en Línea.	120
GRÁFICO # 37	Accediendo a la aplicación desde el Portal.	121
GRÁFICO # 38	Aplicación Hola Mundo con ZK.	122
GRÁFICO # 39	Mostrando una alerta.	123
GRÁFICO # 40	Arquitectura de ZK.	131
GRÁFICO # 41	El Desktop de ZK.	133
GRÁFICO # 42	Propiedades caption y label de la ventana.	147
GRÁFICO # 43	Uso de Tabbox.	152
GRÁFICO # 44	Uso del Listbox.	154
GRÁFICO # 45	ListBox multi-columna con encabezados.	155
GRÁFICO # 46	ListBox con pie de columnas.	155
GRÁFICO # 47	ListBox como lista desplegable.	156
GRÁFICO # 48	Uso de iframe.	159
GRÁFICO # 49	Creando un servicio web a partir de la plantilla.	165
GRÁFICO # 50	Código del Servicio Web.	166
GRÁFICO # 51	Código de programación del Servicio Web.	167
GRÁFICO # 52	Fragmento del archivo de configuración donde se declara la cadena de conexión a la base de datos.	172
GRÁFICO # 53	Ejecutando el servicio en un navegador.	182
GRÁFICO # 54	Publicación del servicio web.	184
GRÁFICO # 55	Configuración del servicio en Internet Information Services.	184
GRÁFICO # 56	Propiedades del servicio web.	185

GRÁFICO # 57	Accediendo a un método web.	185
GRÁFICO # 58	Selección de versión de .NET Framework.	186
GRÁFICO # 59	Seleccionando el tipo de proyecto.	188
GRÁFICO # 60	Seleccionando parámetros iniciales.	188
GRÁFICO # 61	Estructura inicial del proyecto ZK.	189
GRÁFICO # 62	Asistente de creación del cliente del servicio web.	190
GRÁFICO # 63	Selección del URI del servicio web publicado.	190
GRÁFICO # 64	Clases generadas por Axis para el cliente.	191
GRÁFICO # 65	Creando un documento zul.	192
GRÁFICO # 66	Agregando la paleta de elementos ZUL.	193
GRÁFICO # 67	Visualización del código del documento y el editor visual.	194
GRÁFICO # 68	Ejecución de la aplicación en Eclipse.	203
GRÁFICO # 69	Logo de CAS Server.	204

ÍNDICE DE ANEXOS

	Pág.
ANEXO # 1 Encuesta al personal técnico encargado de la administración y actualización de la página web de la Universidad Estatal Península de Santa Elena	92
ANEXO # 2 Encuesta a los Usuarios de la Página Web	94
ANEXO # 3 Manual Técnico del Portal Web de la Universidad Estatal Península de Santa Elena	95
ANEXO # 4 Manual de desarrollo en ZK	105
ANEXO # 5 Manual de Desarrollo de Servicios Web	165
ANEXO # 6 Manual del Servidor de Autenticación	204
ANEXO # 7 Manual de Instalación de Joomla!	221
ANEXO # 8 Manual de Usuario de Joomla!	230
ANEXO # 9 Manual de instalación de Zimbra	251
ANEXO # 10 Manual de Usuario de Zimbra	263
ANEXO # 11 Manual de instalación de EspaBiblio	275
ANEXO # 12 Manual de Usuario de EspaBiblio	280
ANEXO # 13 Manual de Implementación e integración de CENTOS DS	295
ANEXO # 14 Guía de Importación de Material Bibliográfico con estándar UNIMARC	305

INTRODUCCIÓN

La Universidad Estatal Península de Santa Elena (UPSE), es una institución educativa con el firme compromiso de formar profesionales de excelencia que incentiven el desarrollo de la región.

La presencia de la UPSE en el Internet es a través del sitio web www.upse.edu.ec. La actualización de la información publicada en este sitio se realiza mediante la creación de páginas estáticas HTML, pues, por el momento no se cuenta con un Sistema de Gestión de Contenidos (CMS por sus siglas en inglés) que facilite esta tarea.

Actualmente, los estudiantes pueden acceder a su historial académico desde los puntos de consulta ubicados en el campus universitario, motivo por el cual, solo se puede consultar esta información en los horarios laborables de la Universidad.

Otro de los valiosos recursos con los que cuenta la Institución es la Biblioteca Universitaria, a la que muchos estudiantes no pueden dirigirse por múltiples motivos, razón por la cual, es necesario, brindar facilidades para interactuar con el material bibliográfico disponible.

Debido a los puntos expuestos, hemos creído conveniente proponer la redefinición del sitio web de la UPSE con la finalidad de incorporar en él utilidades para la actualización de información, consulta académica, gestión bibliográfica y correo electrónico.

Con las posibilidades que brindan las herramientas informáticas actuales, muchas de las cuales son de uso libre y de código abierto, es posible adaptar soluciones

genéricas existentes, a la realidad de una institución, complementándolas con el desarrollo de aplicaciones ajustadas a la medida de las necesidades propias del medio.

De esta forma, es factible establecer una plataforma que sea fácil de administrar, fácil de mantener y altamente escalable, de manera que se pueda incorporar componentes que brinden funcionalidades adicionales a los usuarios.

El presente Proyecto pretende poner a disposición de la Universidad Estatal Península de Santa Elena una alternativa viable para la implementación de un portal en Internet que además de cumplir con las funciones informativas, permita establecer un medio de la interacción con la institución.

El capítulo I, señala la misión, visión y filosofía de la Universidad Estatal Península de Santa Elena en el cual está centrado el trabajo, se presentan las motivaciones que han desencadenado el desarrollo del proyecto y se definen las tecnologías existentes en la actualidad y que forman parte de la plataforma.

El capítulo II, trata sobre la metodología del proyecto, la viabilidad de la definición de la plataforma así como las modalidades de la investigación y las observaciones de la encuesta con su tabulación, todo lo que permite justificar la aplicabilidad del proyecto.

En el capítulo III, se relatan las fases requeridas para la concepción de la plataforma que se implementó en el portal web de la Universidad Estatal Península de Santa Elena. Se incluyen también los criterios utilizados para seleccionar las herramientas así como una breve descripción de cada una de ellas.

CAPÍTULO I

1. MARCO TEÓRICO

1.1. UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA

1.1.1. MISIÓN

La Universidad Estatal Península de Santa Elena tendrá como misión primordial la investigación científica, la formación profesional y técnica, la creación, difusión y desarrollo de la Cultura Nacional.

1.1.2. VISIÓN

La Península de Santa Elena lidera el desarrollo empresarial, turístico, agrícola, industrial y biomarítimo del país, gracias a la acción científica y tecnológica que ejerce la Universidad Estatal Península de Santa Elena en toda la micro región. Sus egresados son la fuerza profesional que orienta el uso racional de los recursos, con justicia, honestidad y servicio a la comunidad.

1.1.3. FILOSOFÍA

La Universidad Estatal Península de Santa Elena se constituye con el convencimiento de que el alma joven de los pueblos jamás puede corromperse, puesto que siempre existirán ciudadanos dispuestos a luchar por la vigencia de los principios fundamentales de Justicia y Amor inspirados y con Fe en la Sabiduría y en el Poder de DIOS en la búsqueda incesante de bienestar para todos.

1.1.4. ANTECEDENTES

La Universidad Estatal Península de Santa Elena (UPSE), es un Centro de Educación Superior dedicado a la formación de Profesionales capacitados técnica y científicamente para que sean un aporte a la sociedad, como hombres y mujeres de bien y sobre todo al desarrollo de la Provincia y del país, gracias a su amplio campo de acción en lo que a Educación se refiere.

Actualmente, la Universidad Estatal Península de Santa Elena cuenta con un sitio web (www.upse.edu.ec) que es actualizado mediante la creación de páginas estáticas HTML, dado que no se ha implementado un Sistema de Gestión de Contenidos (CMS por sus siglas en inglés) que facilite esta tarea. Existen servicios a los cuales los estudiantes acceden acercándose físicamente a las oficinas destinadas para tal efecto, tales como consulta de notas o préstamo de material bibliográfico, razón por la cual, en muchas ocasiones resulta complicado para un estudiante hacer uso de estos valiosos servicios.

1.1.5. OBJETIVOS

La Universidad Estatal Península de Santa Elena tiene como objetivos:

- Desarrollar un modelo de gestión que contribuya al mejoramiento continuo de los servicios educativos, mediante una adecuada infraestructura, óptimos recursos didácticos y tecnológicos.
- Fortalecer el proceso de formación profesional de las diferentes áreas del saber, a través de la implementación del currículo por competencias y el cumplimiento de los estándares e indicadores de calidad.
- Contribuir al desarrollo local, nacional mediante la investigación científica, tecnológica y social; fortaleciendo y modernizando los medios de producción y difusión.
- Proporcionar asistencia técnica, social, médica, cultural y consultoría especializada al sector público y privado.

- Los demás objetivos que constan en el Art. 3 de la Ley Orgánica de Educación Superior en vigencia.

1.2. LOS SISTEMAS INFORMÁTICOS

1.2.1. TEORÍA GENERAL DE SISTEMAS

La Teoría General de los Sistemas (T.G.S.) surge como una metateoría, una teoría de teorías (en sentido figurado), que partiendo del muy abstracto concepto de sistema busca reglas de valor general, aplicables a cualquier ciencia y en cualquier nivel de la realidad.

Aunque la T.G.S. surgió en el campo de la Biología, pronto se vio su capacidad de inspirar desarrollos en disciplinas distintas y se aprecia su influencia en la aparición de otras nuevas. Así se ha ido constituyendo el amplio campo de la sistémica o de las ciencias de los sistemas, con especialidades como la cibernética, la teoría de la información, la teoría de juegos, entre otras..

El contexto en el que la T.G.S. se puso en marcha, es el de una ciencia dominada por las operaciones de reducción, características del método analítico. Básicamente, para poder manejar una herramienta tan global, primero se ha de partir de una idea de lo que se pretende demostrar, definir o poner a prueba. Teniendo claro el resultado (partiendo de la observación en cualquiera de sus vertientes), entonces se le aplica un concepto que, lo mejor que se puede asimilar resultando familiar y fácil de entender, es a los métodos matemáticos conocidos como mínimo común múltiplo y máximo común divisor. A semejanza de estos métodos, la T.G.S. trata de ir desengranando los factores que intervienen en el resultado final, a cada factor le otorga un valor conceptual que fundamenta la coherencia de lo observado, enumera todos los valores y trata de analizar todos por separado y, en el proceso de la elaboración de un postulado, trata de ver cuantos conceptos son comunes y no comunes con un mayor índice de repetición,

así como los que son comunes con un menor índice de repetición. Con los resultados en mano y un gran esfuerzo de abstracción, se les asignan a conjuntos, formando objetos. Con la lista completa de objetos y las propiedades declaradas de dichos objetos, se conjeturan las interacciones que existen entre ellos, mediante la generación de un modelo informático que pone a prueba si dichos objetos, virtualizados, muestran un resultado con unos márgenes de error aceptables. En el último paso, se procede mediante las pruebas de laboratorio, siendo las conjeturas, postulados, especulaciones, intuiciones y demás sospechas, puestas a prueba naciendo así la teoría.

1.2.2. LOS SISTEMAS INFORMÁTICOS

Un sistema informático como todo sistema, es el conjunto de partes interrelacionadas, hardware, software y de Talento Humano que puede estar especializado en el manejo de cualquiera de las anteriores. Un sistema informático típico emplea una computadora que usa dispositivos programables para capturar, almacenar y procesar datos. La computadora personal o PC, junto a la persona que lo maneja y los periféricos que los envuelven, resultan de por sí un ejemplo de un sistema informático.

Incluso la computadora más sencilla se clasifica como un sistema informático, porque al menos dos componentes (hardware y software) tienen que trabajar unidos. Pero el genuino significado de "sistema informático" viene mediante la interconexión. Muchos sistemas informáticos pueden interconectarse, esto es, unirse para conformar un sistema mayor. La interconexión de sistemas informáticos puede tornarse difícil debido a incompatibilidades. A veces estas dificultades ocurren a nivel de hardware, mientras que en otras ocasiones se dan entre programas informáticos que no son compatibles entre sí.

Los diseñadores de sistemas informáticos no necesariamente esperan que sus sistemas se puedan interconectar con otros sistemas, sino que estos trabajen de

forma independiente en un ambiente empresarial. Por otro lado, los técnicamente eruditos a menudo pueden configurar sistemas diferentes para que se puedan comunicar entre sí usando un conjunto de reglas y restricciones conocidas como protocolos. Los protocolos tratan precisamente de definir la comunicación dentro de y entre sistemas informáticos distintos pero conectados entre sí. Si dos sistemas informáticos usan el mismo protocolo, entonces podrán ser capaces de interconectarse y formar parte de un sistema mayor.

Las tecnologías de información son un conjunto de conocimientos aplicados con la finalidad de obtener herramientas, tanto de hardware como de software, necesarias para el desarrollo de sistemas informáticos y que también sirven como soporte para que estos sistemas puedan ser utilizados en el ámbito para el cual fueron realizados.

La evolución de estas tecnologías han hecho necesaria la definición de normas para regir el desarrollo de los sistemas informáticos, concibiéndose así la ingeniería de software como el conjunto de técnicas que permiten a los desarrolladores elaborar aplicaciones informáticas o software que cumplan estándares que garantizan la calidad del producto desarrollado.

1.2.3. LAS APLICACIONES WEB

Una aplicación web es una sistema que puede ser utilizado accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación que se codifica en un lenguaje soportado por los navegadores web. Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, ya que no requieren la instalación previa de algún otro programa para poder funcionar. Otro de los factores que le han dado popularidad es la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales. Existen aplicaciones como los webmails, wikis, weblogs que son ejemplos de aplicaciones web.

Es importante, mencionar que una página Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto facilita al usuario acceder a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones, como por ejemplo rellenar y enviar formularios o participar en juegos diversos.

Breve reseña histórica

En los primeros tiempos de la computación, las aplicaciones eran utilizadas bajo un esquema conocido como cliente-servidor, esto implicaba que cada aplicación tenía su propio programa cliente que servía como interfaz de usuario, el mismo que tenía que ser instalado por separado en cada ordenador personal de cada usuario.

El cliente realizaba peticiones a otro programa -el servidor- que le daba respuesta. Una mejora en el servidor, como parte de la aplicación, requería normalmente una mejora de los clientes instalados en cada ordenador personal, añadiendo un coste de soporte técnico y disminuyendo la productividad.

A diferencia de lo anterior, las aplicaciones web generan dinámicamente una serie de páginas en un formato estándar, como HTML o XHTML, las cuales son soportadas por los navegadores web comunes. Para este fin se utilizan lenguajes interpretados en el lado del cliente, tales como JavaScript, para añadir elementos dinámicos a la interfaz de usuario. Generalmente cada página web en particular se envía al cliente como un documento estático, pero la secuencia de páginas ofrece al usuario una experiencia interactiva. Durante la sesión, el navegador web interpreta y muestra en pantalla las páginas solicitadas, actuando como cliente para cualquier aplicación web.

1.2.4. LOS FRAMEWORKS

La palabra inglesa framework define, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular, que sirve como referencia para enfrentar y resolver nuevos problemas de índole similar.

En el desarrollo de software, un framework es una estructura conceptual y tecnológica de soporte definida, normalmente con artefactos de software concretos, mediante el cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros programas para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extienden o utilizan las aplicaciones del dominio.

Los frameworks son diseñados con la intención de facilitar el desarrollo de software, permitiendo a los diseñadores y programadores pasar más tiempo identificando requerimientos de software que tratando con los tediosos detalles de bajo nivel de proveer un sistema funcional. Por ejemplo, un equipo que usa Apache Struts para desarrollar un sitio web de un banco, puede enfocarse en cómo los retiros de ahorros van a funcionar en lugar de preocuparse de cómo se controla la navegación entre las páginas en una forma libre de errores.

Sin embargo, hay quejas comunes acerca de que el uso de frameworks añade código innecesario y que la preponderancia de frameworks competitivos y complementarios significa que el tiempo que se pasaba programando y diseñando ahora se gasta en aprender a usar frameworks.

1.2.5. LOS SERVICIOS WEB

Un servicio web (en inglés, Web Service) es un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.

Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos tales como XML, SOAP, WSDL, entre otros. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios Web. Para mejorar la interoperabilidad entre distintas implementaciones de servicios Web se ha creado el organismo WS-I, encargado de desarrollar diversos perfiles para definir de manera más exhaustiva estos estándares.

Los servicios han cambiado el concepto que se tenía sobre la Web, mostrándonos la forma de aprovecharla para construir software.

Antes de los servicios, la Web era principalmente un medio para llegar al usuario a un bajo costo y aunque aún sigue siendo así, los servicios le han dado mayor potencial cambiando la manera en que construimos nuestras aplicaciones.

En su núcleo, los servicios representan una interfaz (o un conjunto de métodos) que proporcionan un acceso a funcionalidades compartidas mediante el uso de formatos y protocolos comunes, lo cual permite mantener independencia entre equipos, lenguajes o sistemas operativos utilizados.

En la actualidad la Universidad Estatal Península de Santa Elena cuenta con un Sistema de Control Académico, el mismo que almacena la información en una base de datos SQL Server.

Debido a que el Portal Web Universitario se desarrolla enteramente en Software Libre, es necesario, crear una interface que nos permita portar la información en un formato universal que no dependa exclusivamente de la plataforma operativa, por tal razón, se seleccionó el formato XML a través de servicios web.

Para desarrollar este servicio web se utilizó la herramienta Visual Studio 2005, principalmente para mantener compatibilidad con la plataforma operativa en la cual se encuentra la base de datos del Sistema de Control Académico.

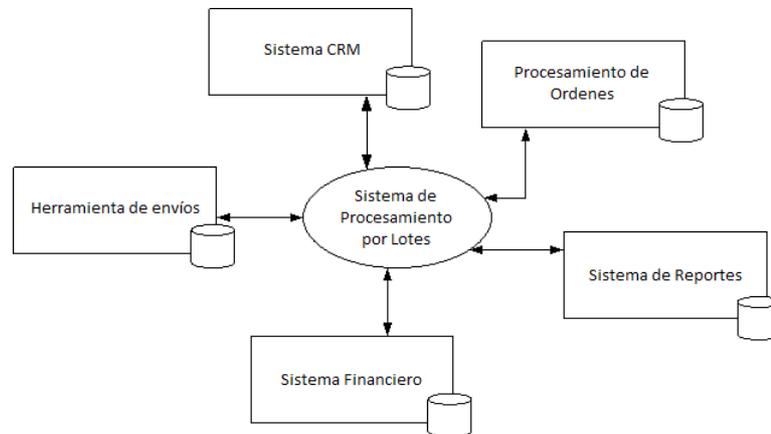
Una vez publicado el servicio, éste es consumido por una aplicación de consulta desarrollada enteramente para ambiente web (con la herramienta ZK), teniendo como base J2EE.

1.2.5.1. FUNDAMENTOS DE LOS SERVICIOS WEB

Un servicio define un contrato entre una llamada hecha desde un cliente y el servicio en sí mismo. Este contrato estipula algo como: “Si tú me envías datos en este formato, los procesaré y te devolveré los resultados en este otro formato”. El formato de estos datos y los parámetros de comunicación de estas llamadas están basados enteramente en estándares abiertos (XML y SOAP). Estándares similares permiten “descubrir” un servicio web y su contrato, y así trabajar con el mismo.

Estos estándares se aplican sin importar el tipo de tecnología utilizada, y por tanto, hacen a los servicios un recurso atractivo para el intercambio de datos entre ambientes heterogéneos. Para comprender los beneficios que tienen los servicios web, es de mucha ayuda colocarse en el contexto de los problemas para los cuales fueron diseñados.

GRÁFICO # 1 Aplicaciones heterogéneas con necesidades comunes



Fuente: Powers, Snell (2005) Microsoft Visual Studio 2005 Unleashed

Por ejemplo, muchas compañías grandes tienen múltiples aplicaciones que necesitan acceder y actualizar información similar. Entre estas aplicaciones, pueden haber sistemas CRM (Gestión de Relaciones con los Clientes), aplicaciones de procesamiento, herramientas de navegación, sistemas financieros, entre otros. Esto implicaría que se generaría duplicidad en el registro de información, provocando que en cierto momento exista información contradictoria o desactualizada en cualquiera de los sistemas.

Las compañías podrían optar por aplicar soluciones parche tales como ejecutar un proceso por lotes que sincronice la información de las diferentes aplicaciones o que consolide la información en una base central diariamente.

Estas situaciones son las que se intentan resolver mediante la definición de soluciones orientadas a servicios. Se debe tomar en cuenta que cada una de las aplicaciones pueden residir en servidores diferentes y con distintos sistemas operativos, incluso con diferentes tecnologías de base de datos.

Por tanto, un componente reutilizable que pueda ser conectado en cada aplicación no podría ser creado fácilmente, incluso si existiese la posibilidad, aún existiría la

necesidad de centralizar la información. Por ejemplo, la actualización del registro de un cliente en uno de los sistemas, de alguna manera, tendría que ser reflejado en los otros.

Lo que se requiere para resolver este tipo de problema es una interface compartida dentro de una vista centralizada del cliente. Esta interface debería ser capaz de trabajar sin importar las limitaciones de la aplicación, como protocolos, tipos de datos y procesos. Esta problemática ha existido desde siempre, pero no se había logrado definir una solución completa.

Con las tecnologías web, el protocolo HTTP es omnipresente. Los servidores podían entonces comunicarse unos con otros. Luego llegó el estándar XML para describir mensajes.

Con estas dos tecnologías (HTTP y XML), las aplicaciones que se ejecutan en distintas plataformas tenían una manera de comunicarse, pudiendo enviar y recibir mensajes estructurados sobre un protocolo común, naciendo así los servicios web.

1.2.5.2. LOS SERVICIOS WEB DE ASP.NET

A continuación se detallan algunos términos que permitirán comprender de mejor manera todo lo relacionado al desarrollo de servicios web.

- **Web Service:** Un web service representa un conjunto coherente de lógica de aplicación que realiza acciones y proporciona datos. Un servicio web agrupa esta lógica como métodos que pueden ser llamados a través de HTTP. No todos los servicios son servicios web, sólo aquellos que trabajan sobre Internet se consideran servicios web.
- **Web Service Method (o Web Method):** Un web service method o método de servicio web, representa un método puesto a disposición a través de un web service. Un método web puede tomar uno o más parámetros y devolver una respuesta.

- **XML (Lenguaje de marcado extensible):** XML es utilizado tanto para representar como para describir los datos de una manera independiente a la plataforma. XML puede ser utilizada para representar tanto elementos simples como complejos y sus relaciones. El estándar XML es el que hace posible la existencia de los servicios web.
- **WSDL (Lenguaje de descripción de servicios web):** WSDL es utilizado para describir el contenido de un servicio web, así como sus métodos. WSDL proporciona los datos del contrato que permiten al cliente trabajar con un determinado servicio.
- **SOAP (Protocolo de Acceso Simple a Objetos):** SOAP es un protocolo basado en XML que permite la comunicación entre un cliente y un servicio web. SOAP envuelve los mensajes XML en sobres para su transmisión en la Web. La mayoría de los mensajes SOAP se envían sobre el protocolo HTTP, pero es posible transmitirlos a través de otros protocolos como SMTP o FTP.
- **HTTP (Protocolo de transferencia de Hipertexto):** HTTP representa el protocolo de comunicación utilizado por los servicios web para transmitir mensajes formateados (o codificados) por SOAP. HTTP es también la manera estándar en que las páginas web realizan solicitudes de comunicación (GET y POST).
- **URI (Identificador Uniforme de Recursos):** Los URIs proporcionan un medio para la localización de ítems en la Web. En la mayoría de casos los URIs son URLs (Localizador Uniforme de Recursos) que apuntan a un servicio determinado.

En la actualidad, la mayoría de herramientas de desarrollo de aplicaciones que soportan la creación de servicios web simplifican, en gran manera, el proceso de definición de este tipo de aplicaciones, ya que generan automáticamente el conjunto de recursos necesarios para su publicación o para su consumo, Visual Studio 2005 no es la excepción, ya que este proceso es similar al de la creación de una clase. La definición de los métodos web es muy parecida a la definición de los

métodos de una clase. Visual Studio se encarga del trabajo de generar el contrato de esquema apropiado para los métodos y de la descripción del servicio web en términos de WSDL. Finalmente, es .NET Framework el que se encarga de empaquetar apropiadamente los datos en un mensaje SOAP y transmitirlos a través del protocolo HTTP.

1.2.6. LOS SERVICIOS DE DIRECTORIO

Un servicio de directorio (SD) es una aplicación o un conjunto de aplicaciones que almacena y organiza la información sobre los usuarios de una red de ordenadores, sobre los recursos de red, y permite a los administradores gestionar el acceso de usuarios a esos recursos. Además, los servicios de directorio actúan como una capa de abstracción entre los usuarios y los recursos compartidos.

Un servicio de directorio no debería confundirse con el repositorio de directorio, que es la base de datos la que contiene la información sobre los objetos de nombrado, gestionada por el servicio de directorio.

En el caso del modelo de servicio de directorio distribuido en X.500, se usa uno o más espacios de nombre (árbol de objetos) para formar el servicio de directorio.

El servicio de directorio proporciona la interfaz de acceso a los datos que se contienen en uno o más espacios de nombre de directorio. La interfaz del servicio de directorio es la encargada de gestionar la autenticación de los accesos al servicio de forma segura, actuando como autoridad central para el acceso a los recursos de sistema que manejan los datos del directorio.

Como base de datos, un servicio del directorio está altamente optimizado para lecturas y proporciona alternativas avanzadas de búsqueda en los diferentes atributos que se puedan asociar a los objetos de un directorio. Los datos que se

almacenan en el directorio son definidos por un esquema extensible y modificable. Los servicios de directorio utilizan un modelo distribuido para almacenar su información y esa información generalmente está replicada entre los servidores que forman el directorio.

Existen numerosas formas de implementación de servicios de directorio de diferentes compañías. Algunos de estos ejemplos son:

- **NIS protocolo Network Information Service**, nombrado originalmente como Páginas Amarillas, implementación de Sun Microsystems' en un servicio de directorio para redes de entorno UNIX. (Sun, a principios del 2000, se unió a iPlanet, alianza de Netscape y desarrolló la base de LDAP, servicio de directorio que formó parte de Sun ONE, la empresa que es ahora Sun Java Enterprise.
- **eDirectory**, desarrollado por Novell, es un servicio de directorio que soporta múltiples arquitecturas incluyendo Windows, NetWare, Linux, incluyendo algunas distribuciones de Unix. Se han utilizado durante tiempo para la administración de usuarios, gestión de configuraciones y gestión de software. eDirectory se ha desarrollado como componente central en una gama más amplia de productos para la gestión de identidad. Fue conocido previamente como servicios de directorio de Novell.
- **Servidor de directorio de Red Hat:** Red Hat lanzó un servicio del directorio, que adquirió de “Netscape Security Solutions de AOL”, el cual funcionaba como producto comercial bajo Red Hat Enterprise Linux denominado como servidor de directorio de Red Hat como parte del núcleo de Fedora.
- **CentOS Directory Server:** CentOS Directory Server está basado en Red Hat Directory Server, posee similares características, y está disponible para instalar vía yum, sin necesidad de tener un contrato de por medio. La base del software está licenciada bajo GNU/GPL 2, y se incluye una excepción para ser linkeado con software no libre, la cual proviene de RedHat.

- **Active Directory:** El servicio del directorio de Microsoft, es el directorio que se incluye en las versiones de los sistemas operativos Windows 2000 y Windows Server 2003.
- **Open Directory:** El servidor del Mac OS X de Apple ofrece un servicio del directorio llamado Open Directory que integra muchos protocolos estándares abiertos tales como LDAP y Kerberos así como soluciones propietarias de directorio como Active Directory y eDirectory.
- **Servidor de directorio de Apache:** Apache Software Foundation ofrece un servicio del directorio llamado ApacheDS.
- **Directorio de Internet de Oracle:** (OID) es el servicio del directorio de Oracle Corporation, que es compatible con la versión 3 de LDAP.
- **Directorio CA:** El directorio CA contiene un motor de caché previo que puede indexar todos los atributos que se usan en los filtros de búsqueda de LDAP, y poner en caché aquellos atributos devueltos en tales búsquedas. Teniendo bastante memoria, el directorio CA es el directorio más rápido.

Hay también gran cantidad de herramientas de código abierto para crear servicios de directorio, incluyendo OpenLDAP, el protocolo Kerberos, el software de Samba, que puede actuar como controlador de dominio con Kerberos y estar implementado con LDAP.

1.2.7. LAS CONEXIONES SEGURAS

Transporte seguro de datos

El Protocolo de Capa de Conexión Segura (Secure Sockets Layer o SSL) es un protocolo criptográfico que proporciona comunicaciones seguras por una red, comúnmente Internet. SSL proporciona autenticación y privacidad de la información entre extremos sobre Internet mediante el uso de criptografía.

Habitualmente, sólo el servidor es autenticado (es decir, se garantiza su identidad) mientras que el cliente se mantiene sin autenticar; la autenticación mutua requiere un despliegue de infraestructura de claves públicas (o PKI) para los clientes. Los protocolos permiten a las aplicaciones cliente-servidor comunicarse de una forma diseñada para prevenir escuchas, la falsificación de la identidad del remitente y alterar la integridad del mensaje.

SSL implica una serie de fases básicas:

- Negociar entre las partes el algoritmo que se usará en la comunicación.
- Intercambio de claves públicas y autenticación basada en certificados digitales.
- Cifrado del tráfico basado en cifrado simétrico.

Los Certificados Digitales

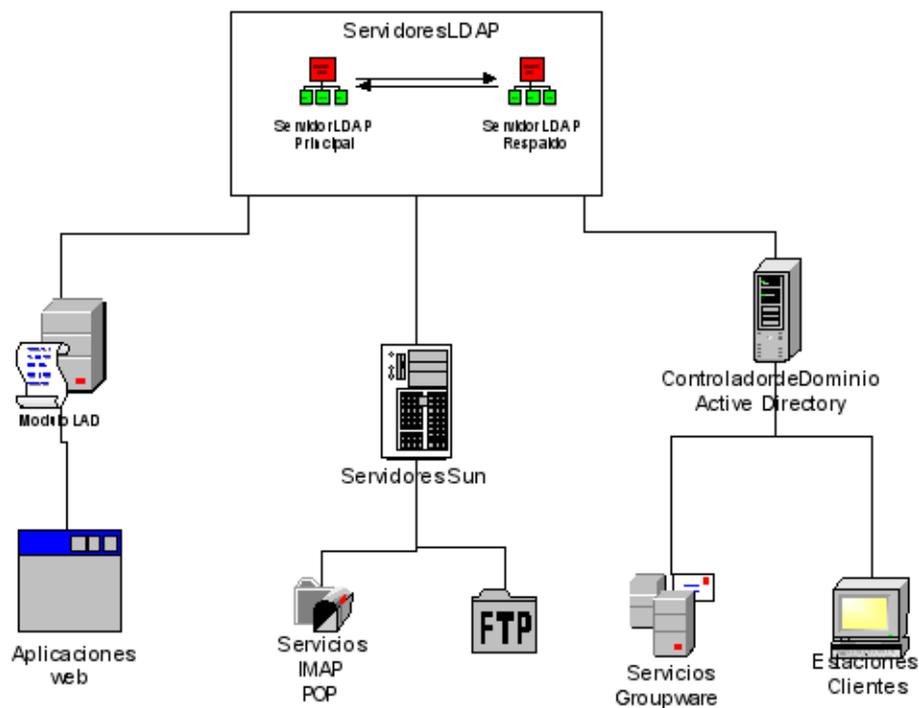
Un certificado digital es un documento digital mediante el cual un tercero confiable (autoridad de certificación) garantiza la vinculación entre la identidad de un sujeto o entidad y su clave pública. Si bien existen variados formatos para certificados digitales, los más empleados se rigen por el estándar UIT-T X.509. El certificado contiene usualmente el nombre de la entidad certificada, número de serie, fecha de expiración, una copia de la clave pública del titular del certificado (para verificación de la firma digital) y la firma digital de la autoridad emisora del certificado de forma que el receptor pueda verificar que ésta última realmente estableció la asociación.

Autenticación

Autenticación se refiere al proceso por medio del cual un usuario de una red adquiere el derecho a usar una identidad dentro de dicha red. Hay maneras de autenticar un usuario, como el uso de claves, Biométricos, smart cards,

certificados digitales. La ventaja es que la identidad del usuario en la red no necesariamente tiene que ser igual al nombre de la persona. Una misma persona puede tener muchas identidades virtuales y viceversa.

GRÁFICO # 2
Esquema de autenticación



Fuente: <http://es.wikipedia.org/wiki/LDAP>

Existen dos modelos de autenticación: uno descentralizado y otro centralizado. En el modelo descentralizado, cada servicio de la red maneja sus claves de forma independiente, por ejemplo, los usuarios de Oracle, los usuarios de un firewall, los administradores de un sitio Web; cada una de estas aplicaciones maneja por separado sus claves y las mismas no son compartidas.

En la autenticación centralizada los usuarios y sus claves se ubican en un repositorio central, las diferentes aplicaciones se configuran para identificar este lugar y hacer la autenticación contra el repositorio. Por lo general, este repositorio suele ser un servidor de directorio LDAP, pero también podrían estar almacenadas

en un archivo de texto plano o en una base de datos relacional entre otros métodos de almacenamiento de información.

Single Sign-On

Single sign-on (SSO) es un procedimiento de autenticación que habilita al usuario el acceso a varios sistemas con una sola instancia de identificación.

Hay cinco tipos principales de SSO, también conocidos como sistemas de autenticación reducida.

- **Enterprise single sign-on (E-SSO)**, también llamado legacy single sign-on, funciona para una autenticación primaria, interceptando los requerimientos de login presentados por las aplicaciones secundarias para completar los mismos con el usuario y contraseña. Los sistemas E-SSO permiten interactuar con sistemas que pueden deshabilitar la presentación de la pantalla de login.
- **Web single sign-on (Web-SSO)**, también llamado Web access management (Web-AM) trabaja sólo con aplicaciones y recursos accedidos vía web. Los accesos son interceptados con la ayuda de un servidor proxy o de un componente instalado en el servidor web destino. Los usuarios no autenticados que tratan de acceder son redirigidos a un servidor de autenticación y regresan solo después de haber logrado un acceso exitoso. Se utilizan cookies, para reconocer aquellos usuarios que acceden y su estado de autenticación.
- **Kerberos** es un método popular de externalizar la autenticación de los usuarios. Los usuarios se registran en el servidor Kerberos y reciben un "ticket", luego las aplicaciones-cliente lo presentan para obtener acceso.
- **Identidad federada** es una nueva manera de concebir este tema, también para aplicaciones Web. Utiliza protocolos basados en estándares para

habilitar que las aplicaciones puedan identificar los clientes sin necesidad de autenticación redundante.

- **OpenID** es un proceso de SSO distribuido y descentralizado donde la identidad se compila en una url que cualquier aplicación o servidor puede verificar.

1.2.8. LOS SERVIDORES DE APLICACIONES

En informática, se denomina servidor de aplicaciones a un servidor en una red de computadores que ejecuta ciertas aplicaciones. Usualmente se trata de un dispositivo de software que proporciona servicios de aplicación a las computadoras cliente. Un servidor de aplicaciones generalmente gestiona la mayor parte (o la totalidad) de las funciones de lógica de negocio y de acceso a los datos de la aplicación.

Los principales beneficios de la aplicación de la tecnología de servidores de aplicación son la centralización y la disminución de la complejidad en el desarrollo de aplicaciones. Si bien el término es aplicable a todas las plataformas de software, hoy en día el término servidor de aplicaciones se ha convertido en sinónimo de la plataforma Java EE (antes J2EE) de Sun Microsystems.

1.3. EL SOFTWARE LIBRE

Es la denominación del software que respeta la libertad de los usuarios sobre un producto adquirido y, por tanto, una vez obtenido puede ser usado, copiado, estudiado, cambiado y redistribuido libremente.

Las cuatro libertades de los usuarios del software, son:

1. La libertad de usar el programa, con cualquier propósito;
2. La libertad de estudiar cómo funciona el programa y modificarlo, adaptándolo a sus necesidades;

3. La libertad de distribuir copias del programa, con lo cual se puede ayudar a otros; y
4. La libertad de mejorar el programa y hacer públicas esas mejoras a los demás, de modo que toda la comunidad se beneficie.

La segunda y la última libertad mencionadas, requieren acceso al código fuente porque estudiar y modificar software sin su código fuente es muy poco viable.

El software libre suele estar disponible gratuitamente, o al precio de costo de la distribución a través de otros medios; sin embargo, no es obligatorio que sea así, por tanto no hay que asociar software libre a "software gratuito" (denominado usualmente freeware), ya que, conservando su carácter de libre, puede ser distribuido comercialmente ("software comercial"). Análogamente, el "software gratis" o "gratuito" incluye en ocasiones el código fuente; no obstante, este tipo de software no es libre en el mismo sentido que el software libre, a menos que sean garantizados los derechos de modificación y redistribución de dichas versiones modificadas del programa.

Tampoco debe confundirse software libre con "software de dominio público", ya que éste último es aquel que no requiere de licencia, pues sus derechos de explotación son para toda la humanidad, es decir, pertenece a todos por igual.

Cualquiera puede hacer uso de él, siempre con fines legales y consignando su autoría original. Este software sería aquel cuyo autor lo dona a la humanidad o cuyos derechos de autor han expirado, tras un plazo contado desde la muerte de éste, habitualmente 70 años. Si un autor condiciona su uso bajo una licencia, por muy débil que sea, ya no es del dominio público.

La mayoría del software libre se produce por equipos internacionales que cooperan a través de la libre asociación. Estos equipos, también conocidos como comunidades, están típicamente compuestos por individuos con una amplia

variedad de motivaciones, y pueden provenir tanto del sector privado, del sector voluntario o del sector público.

El término software no libre se emplea para referirse al software distribuido bajo una licencia de software más restrictiva que no garantiza las cuatro libertades. Las leyes de la propiedad intelectual reservan la mayoría de los derechos de modificación, duplicación y redistribución para el dueño del copyright; el software dispuesto bajo una licencia de software libre rescinde específicamente la mayoría de estos derechos reservados.

Breve reseña histórica

Entre los años 60 y 70 del Siglo XX, el software no era considerado un producto sino un añadido que los vendedores de las grandes computadoras de la época (las mainframes) aportaban a sus clientes para que éstos pudieran usarlos. En dicha cultura, era común que los programadores y desarrolladores de software compartieran libremente sus programas unos con otros.

Este comportamiento era particularmente habitual en algunos de los mayores grupos de usuarios de la época, como DECUS (grupo de usuarios de computadoras DEC). A finales de los 70, las compañías iniciaron el hábito de imponer restricciones a los usuarios, con el uso de acuerdos de licencia.

Allá por el 1971, cuando la informática todavía no había sufrido su gran boom, las personas que hacían uso de ella, en ámbitos universitarios y empresariales, creaban y compartían el software sin ningún tipo de restricciones. Con la llegada de los años 80 la situación empezó a cambiar. Las computadoras más modernas comenzaban a utilizar sistemas operativos privativos, forzando a los usuarios a aceptar condiciones restrictivas que impedían realizar modificaciones a dicho software.

En caso de que algún usuario o programador encontrase algún error en la aplicación, lo único que podía hacer era darlo a conocer a la empresa desarrolladora para que ésta lo solucionara. Aunque el programador estuviese capacitado para solucionar el problema y lo deseara hacer sin pedir nada a cambio, el contrato le impedía que mejorase el software.

El mismo Richard Stallman cuenta que por aquellos años, en el laboratorio donde trabajaba, habían recibido una impresora donada por una empresa externa. El dispositivo, que era utilizado en red por todos los trabajadores, parecía no funcionar a la perfección, dado que cada cierto tiempo el papel se atascaba. Como agravante, no se generaba ningún aviso que se enviase por red e informase a los usuarios de la situación.

La pérdida de tiempo era constante, ya que en ocasiones, los trabajadores enviaban por red sus trabajos a imprimir y al ir a buscarlos se encontraban la impresora atascada y una cola enorme de trabajos pendientes. Richard Stallman decidió arreglar el problema, e implementar el envío de un aviso por red cuando la impresora se bloqueara.

Para ello necesitaba tener acceso al código fuente de los controladores de la impresora. Pidió a la empresa propietaria de la impresora lo que necesitaba, comentando, sin pedir nada a cambio, qué era lo que pretendía realizar. La empresa se negó a entregarle el código fuente.

En ese preciso instante, Richard Stallman se vio en una encrucijada: debía elegir entre aceptar el nuevo software privativo firmando acuerdos de no revelación y acabar desarrollando más software privativo con licencias restrictivas, que a su vez deberían ser más adelante aceptadas por sus propios colegas.

Con este antecedente, en 1984, Richard Stallman comenzó a trabajar en el proyecto GNU, y un año más tarde fundó la Free Software Foundation (FSF).

Stallman introdujo la definición de free software y el concepto de "copyleft", que desarrolló para otorgar libertad a los usuarios y para restringir las posibilidades de apropiación del software.

1.3.1. LICENCIAS DE SOFTWARE

Una licencia de software es un contrato entre el licenciante (autor/titular de los derechos de explotación/distribuidor) y el licenciario del programa informático (usuario consumidor/usuario profesional o empresa), para utilizar el software cumpliendo una serie de términos y condiciones establecidas en sus cláusulas.

Las licencias de software pueden establecer entre otras cosas: la cesión de determinados derechos del propietario al usuario final sobre una o varias copias del programa informático, los límites en la responsabilidad por fallos, el plazo de cesión de los derechos, el ámbito geográfico de validez del contrato e incluso pueden establecer determinados compromisos del usuario final hacia el propietario, tales como la no cesión del programa a terceros o la no reinstalación del programa en equipos distintos al que se instaló originalmente.

1.3.2. TIPOS DE LICENCIA DE SOFTWARE LIBRE

Licencias GPL

Una de las más utilizadas es la Licencia Pública General de GNU (GNU GPL). El autor conserva los derechos de autor (copyright), y permite la redistribución y modificación bajo términos diseñados para asegurarse de que todas las versiones modificadas del software permanecen bajo los términos más restrictivos de la propia GNU GPL. Esto hace que sea imposible crear un producto con partes no licenciadas GPL: el conjunto tiene que ser GPL.

Es decir, la licencia GNU GPL posibilita la modificación y redistribución del software, pero únicamente bajo esa misma licencia. Y añade que si se reutiliza en un mismo programa código "A" autorizado bajo licencia GNU GPL y código "B" licenciado bajo otro tipo de licencia libre, el código final "C", independientemente de la cantidad y calidad de cada uno de los códigos "A" y "B", debe estar bajo la licencia GNU GPL.

En la práctica esto hace que las licencias de software libre se dividan en dos grandes grupos, aquellas que pueden ser mezcladas con código licenciado bajo GNU GPL (y que inevitablemente desaparecerán en el proceso, al ser el código resultante licenciado bajo GNU GPL) y las que no lo permiten al incluir mayores u otros requisitos que no contemplan ni admiten la GNU GPL y que por tanto no pueden ser enlazadas ni mezcladas con código gobernado por la licencia GNU GPL. En el sitio web oficial de GNU hay una lista de licencias que cumplen las condiciones impuestas por la GNU GPL y otras que no. Aproximadamente el 60% del software licenciado como software libre emplea una licencia GPL.

Licencias estilo BSD

Llamadas así porque se utilizan en gran cantidad de software distribuido junto a los sistemas operativos BSD. El autor, bajo tales licencias, mantiene la protección de copyright únicamente para la renuncia de garantía y para requerir la adecuada atribución de la autoría en trabajos derivados, pero permite la libre redistribución y modificación, incluso si dichos trabajos tienen propietario. Son muy permisivas, tanto que son fácilmente absorbidas al ser mezcladas con la licencia GNU GPL con quienes son compatibles. Puede argumentarse que esta licencia asegura “verdadero” software libre, en el sentido que el usuario tiene libertad ilimitada con respecto al software, y que puede decidir incluso redistribuirlo como no libre. Otras opiniones están orientadas a destacar que este tipo de licencia no contribuye al desarrollo de más software libre (normalmente utilizando la siguiente analogía: "una licencia BSD es

más libre que una GPL si y sólo si se opina también que un país que permita la esclavitud es más libre que otro que no la permite").

Licencias estilo MPL y derivados

Esta licencia es de Software Libre y tiene un gran valor porque fue el instrumento que empleó Netscape Communications Corp. para liberar su Netscape Communicator 4.0 y empezar ese proyecto tan importante para el mundo del Software Libre: Mozilla. Se utilizan en gran cantidad de productos de software libre de uso cotidiano en todo tipo de sistemas operativos. La MPL es Software Libre y promueve eficazmente la colaboración evitando el efecto "viral" de la GPL (si usas código licenciado GPL, tu desarrollo final tiene que estar licenciado GPL). Desde un punto de vista del desarrollador la GPL presenta un inconveniente en este punto, y lamentablemente mucha gente se cierra completamente ante el uso de dicho código. Estas licencias son denominadas de copyleft débil. La NPL (luego la MPL) fue la primera licencia nueva después de muchos años, que se encargaba de algunos puntos que no fueron tenidos en cuenta por las licencias BSD y GNU.

Copyleft

Esta forma de licencia puede ser usada para modificar el derecho de autor de obras o trabajos. Sus partidarios la proponen como alternativa a las restricciones que imponen las normas planteadas en los derechos de autor, a la hora de hacer, modificar y distribuir copias de una obra determinada. Hay que hacer constar que el titular de los derechos de autor (copyright) de un software bajo licencia copyleft puede también realizar una versión modificada bajo su copyright original, y venderla bajo cualquier licencia que desee, además de distribuir la versión original como software libre. Esta técnica ha sido usada como un modelo de negocio por una serie de empresas que realizan software libre (por ejemplo MySQL); esta práctica no

restringe ninguno de los derechos otorgados a los usuarios de la versión copyleft. También podría retirar todas las licencias de software libre anteriormente otorgadas, pero esto obligaría a una indemnización a los titulares de las licencias en uso. En España, toda obra derivada está tan protegida como una original, siempre que la obra derivada parta de una autorización contractual con el autor. En el caso genérico de que el autor retire las licencias "copyleft", no afectaría de ningún modo a los productos derivados anteriores a esa retirada, ya que no tiene efecto retroactivo. En términos legales, el autor no tiene derecho a retirar el permiso de una licencia en vigencia. Si así sucediera, el conflicto entre las partes se resolvería en un pleito convencional.

1.3.3. EL OPEN SOURCE

El término inglés Open Source (Código Abierto) es utilizado para definir al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones morales y/o filosóficas las cuales destacan en el llamado software libre.

La idea del código abierto se centra en la premisa de que al compartir el código, el programa resultante tiende a ser de calidad superior al software propietario, es una visión técnica. Por otro lado, el software libre tiene tendencias filosóficas e incluso morales: el software propietario, al no poder compartirse, es considerado "anti-ético" dado que prohibir compartir entre seres humanos va en contra del sentido común.

Al igual que el software libre, el código abierto u open source tiene una serie de requisitos necesarios para que un programa pueda considerarse dentro de este movimiento, éstos son:

- Libre redistribución: el software puede ser regalado o vendido libremente.

- Código fuente: el código fuente debe estar incluido u obtenerse libremente.
- Trabajos derivados: la redistribución de modificaciones debe estar permitida.
- Integridad del código fuente del autor: las licencias pueden requerir que las modificaciones sean redistribuidas sólo como parches.
- Sin discriminación de personas o grupos: nadie puede dejarse fuera.
- Sin discriminación de áreas de iniciativa: los usuarios comerciales no pueden ser excluidos.
- Distribución de la licencia: deben aplicarse los mismos derechos a todo el que reciba el programa
- La licencia no debe ser específica de un producto: el programa no puede licenciarse solo como parte de una distribución mayor.
- La licencia no debe restringir otro software: la licencia no puede obligar a que algún otro software que sea distribuido con el software abierto deba también ser de código abierto.
- La licencia debe ser tecnológicamente neutral: no debe requerirse la aceptación de la licencia por medio de un acceso por clic de ratón o de otra forma específica del medio de soporte del software.

Breve reseña histórica

Su uso nació por primera vez en 1998 de la mano de algunos usuarios de la comunidad del software libre, tratando de usarlo como reemplazo al ambiguo nombre original en inglés del software libre (free software). Free en inglés significa dos cosas distintas dependiendo del contexto: gratuidad y libertad.

Lo cual implica, para el caso que nos ocupa, "software que podemos leer, modificar y redistribuir gratuitamente" (software gratuito) y, además, software libre, según la acepción española de libertad.

El término para algunos no resultó apropiado como reemplazo para el ya tradicional free software, pues eliminaba la idea de libertad, confundida usualmente con la simple gratuidad. No obstante, el término código abierto continúa siendo ambivalente, puesto que se usa en la actualidad por parte de programadores que no ofrecen software libre pero, en cambio, sí ofrecen el código fuente de los programas para su revisión o modificación previamente autorizada por parte de sus pares académicos. Dada la anterior ambivalencia, se prefiere el uso del término software libre para referirse a programas que se ofrecen con total libertad de modificación, uso y distribución bajo la regla implícita de no modificar dichas libertades hacia el futuro.

Desde el punto de vista de una "traducción estrictamente literal", el significado textual de "código abierto" es que "se puede examinar el código fuente", por lo que puede ser interpretado como un término más débil y flexible que el del software libre. Sin embargo, ambos movimientos reconocen el mismo conjunto de licencias y mantienen principios equivalentes. Sin embargo, hay que diferenciar los programas de código abierto, que dan a los usuarios la libertad de mejorarlos, de los programas que simplemente tienen el código fuente disponible, previa restricciones sobre su uso o modificación. Un término que pretende resolver posibles ambigüedades o confusiones que ambos términos generan es FOSS (free and open source software).

En la actualidad el código abierto se utiliza para definir un movimiento nuevo de software (la Iniciativa Open Source), diferente al movimiento del software libre, incompatible con este último desde el punto de vista filosófico, y completamente equivalente desde el punto de vista práctico, de hecho, ambos movimientos trabajan juntos en el desarrollo práctico de proyectos.

La idea bajo el concepto de código abierto es sencilla: cuando los programadores (en Internet) pueden leer, modificar y redistribuir el código fuente de un programa, éste evoluciona, se desarrolla y mejora. Los usuarios lo adaptan a sus necesidades,

corrigen sus errores a una velocidad impresionante, mayor a la aplicada en el desarrollo de software convencional o cerrado, dando como resultado la producción de un mejor software.

1.4. TECNOLOGÍAS DE LA INFORMACIÓN Y LA COMUNICACIÓN

Las tecnologías de la información y la comunicación agrupan los elementos y las técnicas utilizadas en el tratamiento y la transmisión de las informaciones, principalmente de informática, Internet y telecomunicaciones.

1.4.1. LAS REDES INTERCONECTADAS

Familia de protocolos de comunicación

La familia de protocolos de comunicación es un conjunto de protocolos de red en los que basan su funcionamiento las redes informáticas y que permiten la transmisión de datos. En ocasiones se le denomina conjunto de protocolos TCP/IP, en referencia a los dos protocolos más importantes que la componen: Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP), que fueron los dos primeros en definirse, y que son los más utilizados de la familia. Existen tantos protocolos en este conjunto que llegan a ser más de 100 diferentes, entre ellos se encuentra el popular HTTP (HyperText Transfer Protocol), que es el que se utiliza para acceder a las páginas web, además de otros como el ARP (Address Resolution Protocol) para la resolución de direcciones, el FTP (File Transfer Protocol) para transferencia de archivos, y el SMTP (Simple Mail Transfer Protocol) y el POP (Post Office Protocol) para correo electrónico, entre otros.

El TCP/IP es la base de Internet, y sirve para enlazar computadoras que utilizan diferentes sistemas operativos, incluyendo PC, minicomputadoras y computadoras centrales sobre redes de área local (LAN) y área extensa (WAN). Fue desarrollado y demostrado por primera vez en 1972 por el Departamento de Defensa de los

Estados Unidos, ejecutándolo en ARPANET, una red de área extensa de dicho departamento.

Internet

Internet es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas heterogéneas que la componen funcionen como una red lógica única, de alcance mundial. Sus orígenes se remontan a 1969, cuando se estableció la primera conexión de computadoras, conocida como ARPANET, entre tres universidades en California y una en Utah, Estados Unidos.

Uno de los servicios que más éxito ha tenido en Internet ha sido la World Wide Web (WWW, o "la Web"), hasta tal punto que es habitual la confusión entre ambos términos. La WWW es un conjunto de protocolos que permite, de forma sencilla, la consulta remota de archivos de hipertexto.

Existen muchos otros servicios y protocolos en Internet, aparte de la Web: el envío de correo electrónico (SMTP), la transmisión de archivos (FTP y P2P), las conversaciones en línea (IRC), la mensajería instantánea y presencia, la transmisión de contenido y comunicación multimedia, los boletines electrónicos (NNTP), el acceso remoto a otras máquinas (SSH y Telnet) o los juegos en línea.

Intranet

Una Intranet es una red privada de ordenadores que utiliza la tecnología que inicialmente fue desarrollada para Internet con el fin compartir de forma segura cualquier recurso, evitando que usuarios ajenos puedan ingresar desde ubicaciones externas a esta red privada.

Funciones de la Intranet

Las redes internas corporativas son potentes herramientas que otorgan a los empleados acceso a la información de la compañía con efectividad, consiguiendo que estos estén permanentemente informados con las últimas novedades y datos de la organización. También es habitual su uso en universidades y otros centros de formación, ya que facilita la consulta de diferentes tipos de información y el seguimiento de los contenidos académicos.

Tienen gran valor como repositorio documental, convirtiéndose en un factor determinante para conseguir el objetivo de una oficina sin papeles. Al añadir funcionalidades complementarias, tales como un buen buscador y una organización adecuada, se puede conseguir consultas rápidas y eficaces sobre un volumen importante de documentación. Los beneficios de una intranet pueden ser enormes.

Las Intranets también deberían cumplir unos requisitos de accesibilidad web permitiendo el acceso independientemente de las limitaciones físicas impuestas por la red corporativa. Gracias a esto, promueve nuevas formas de colaboración y acceso a los sistemas. Ya no es necesario reunir a todos en una sala para discutir un proyecto, equipos de personas alrededor del mundo pueden trabajar juntos sin tener que invertir en gastos de viaje. El resultado de esto es un aumento increíble en la eficiencia acompañada de una reducción de costos.

CAPÍTULO II

2. METODOLOGÍA DE LA INVESTIGACIÓN

2.1. DISEÑO DE LA INVESTIGACIÓN

La presente investigación se desarrolló considerando la investigación cualitativa, uno de los paradigmas recomendados en trabajos de grado por La Universidad Pedagógica Experimental Libertador, de Venezuela, conocido por ser: naturalista, participativo, etnográfico y humanista.

2.2. MODALIDAD DE LA INVESTIGACIÓN

La modalidad de investigación que se empleó en la implementación de la plataforma fue la de Proyecto Factible o de intervención, según Yépez E. (1995). “Un proyecto factible es el que permite la elaboración de una propuesta de un modelo operativo viable, o de una solución posible, cuyo propósito es satisfacer una necesidad o solucionar un problema.”

Para basar nuestro estudio se utilizó el Método Inductivo el cual partiendo de problemas particulares, se eleva a conocimientos generales, lo que nos indujo a la formulación de una hipótesis y así poder obtener los respectivos esquemas que ayudaron en el desarrollo de nuestro proyecto, ya que sin este tipo de parámetros hubiera sido casi imposible y muy complicado cumplir con los requerimientos necesarios para la culminación del proyecto.

Aplicando este método a la naturaleza de nuestro tema, se tomó cada uno de los componentes de la plataforma y por cada uno de ellos se empezó a investigar la existencia de herramientas disponibles bajo el esquema de software libre, se

estructuraron las características que hicieran posible la integración posterior, comparándolas, a fin de seleccionar la opción más conveniente, y finalmente, combinándolas y evaluando el desempeño de la plataforma obtenida.

2.3. TIPOS DE INVESTIGACIÓN

Los tipos de investigación utilizados para obtener la meta de nuestra tesis fueron:

Investigación Bibliográfica.- Este tipo de investigación se enfoca en el análisis teórico y conceptual reflejado en el manejo de documentos y libros que permitan conocer, comparar y deducir los diferentes criterios, conceptualizaciones, análisis, conclusiones, recomendaciones de los diversos autores e instituciones estudiadas.

Investigación de Campo.- Se efectúa en el lugar y tiempo en que ocurren los fenómenos objeto de estudio con el propósito de descubrir, explicar sus causas y efectos, entender su naturaleza e implicaciones, establecer los factores que lo motivan. En la modalidad de investigación de campo, se utilizó generalmente las encuestas, entre las más esenciales sin dejar de descartar aquellas técnicas que nos permitan afianzar las posibilidades de mejoramiento de la información para la propuesta.

Encuesta.- Una encuesta recoge datos más o menos limitados de un número relativamente extenso de sujetos. Su finalidad es conseguir información sobre las variables y no sobre los individuos. Intentan medir lo que existe sin preguntar por qué existe.

La importancia de este tipo de investigación radica en que permite hacer un diagnóstico de cómo está en realidad el problema. Fundamentalmente se refiere a la descriptiva, que tiene como finalidad; describir, registrar, analizar e interpretar la naturaleza del fenómeno en estudio.

2.4. POBLACIÓN Y MUESTRA

2.4.1. POBLACIÓN

Para determinar la población, se estableció quienes iban a beneficiarse al momento en que nuestra solución propuesta estuviera completamente implementada, por este motivo, se tomaron en cuenta a los estudiantes universitarios, personal docente y administrativo de la Institución, tal y como se detalla a continuación:

CUADRO # 1
Población.

	N	%
Estudiantes	7313	91.88%
Matriz	6308	
Extensión Manglaralto	215	
Extensión Playas	595	
Extensión Colonche	195	
Docentes	296	3.72%
Personal Administrativo	350	4.40%
TOTAL	7959	100.00%

Fuente: Dirección Escuela de Informática

2.4.2. MUESTRA

La Muestra es aquella parte representativa de la población, o considerada como el subconjunto del conjunto población.

Existen dos tipos de muestra, la muestra probabilística y la no probabilística.

La muestra probabilística considera aquellos elementos que tienen la misma posibilidad de ser elegidos, mientras que la no probabilística se distingue, no por la probabilidad, sino más bien, por las características de la investigación.

Por las características propias de la investigación, el tipo de muestra utilizada fue la probabilística, la misma que requiere de una fórmula que viabilice la investigación, la cual es:

$$n = \frac{Z^2 p q N}{NE^2 + Z^2 p q}$$

Donde:

- n = es el tamaño de la muestra
- Z = es el nivel de confianza
- p = es la variabilidad positiva
- q = es la variabilidad negativa
- N = es el tamaño de la población
- E = es la precisión o el error

El tamaño de la población es de 7959 personas.

$$n = \frac{1,96^2 * 0,5 * 0,5 * 7959}{(7959 * 0,05^2) + (1,96^2 * 0,5 * 0,5)}$$

$$n = \frac{7643,82}{20,86}$$

$$n = 366$$

Se considerará una confianza del 95%, un porcentaje de error del 5% y la máxima variabilidad por no existir antecedentes en la institución sobre la investigación y porque no se puede aplicar una prueba previa. El valor de Z se obtiene definiendo un nivel de confianza del 95%, es decir, buscando un valor de Z tal que $P(-Z < z < Z) = 0.95$. Utilizando las tablas o las funciones estadísticas resulta que $Z = 1.96$.

Hay que considerar que p y q son complementarios, es decir, que su suma es igual a la unidad: $p + q = 1$.

Además, cuando se habla de la máxima variabilidad, en el caso de no existir antecedentes sobre la investigación (no hay otras o no se pudo aplicar una prueba previa), entonces los valores de variabilidad es $p = q = 0.5$. Por tanto, el tamaño de la muestra es de 366 personas.

Aplicando el porcentaje que representa cada uno de los grupos de personas que conforman a la población, la muestra estaría compuesta por:

CUADRO # 2
Muestra.

Muestra	Fórmula aplicada	n
Estudiantes	$366 * 0.9188 = 336.28$	336
Docentes	$366 * 0.0372 = 13.61$	14
Personal Administrativo	$366 * 0.0440 = 16.10$	16
TOTAL		366

Fuente: Dirección Escuela de Informática

2.5. OPERACIONALIZACIÓN DE LAS VARIABLES

Las variables son aquellas propiedades que poseen ciertas características o particularidades y son susceptibles de medirse u observarse, las variables indican los aspectos relevantes del fenómeno en estudio y que está en relación directa con el planteamiento del problema. A partir de ello se selecciona las técnicas e instrumentos de información para la ejecución del problema en estudio.

La hipótesis de donde se derivan las variables es:

Con el uso exclusivo de Software Libre se puede optimizar el sitio web de la Universidad Estatal Península de Santa Elena para facilitar la publicación de contenido, el acceso a información Bibliográfica, a Información Académica y Correo Electrónico.

2.5.1. VARIABLES DE ESTUDIO

Como resultado del análisis de la hipótesis definida en el apartado anterior, hemos determinado la existencia de dos variables de estudio, identificadas como variable dependiente e independiente.

A continuación mostramos el desglose de la parte operacional de cada una de ellas:

VARIABLE INDEPENDIENTE: Herramientas de Software Libre.

PARTE OPERACIONAL: Se hizo un estudio de las herramientas de software libre disponibles y que cumplieran con cada una de las funcionalidades específicas que componen el portal web.

CUADRO # 3
Operacionalización de la variable independiente.

Variable: Herramientas de Software Libre.	
Significado	El Software libre es aquel que respeta las “ <i>cuatro libertades de los usuarios del software</i> ”: <ol style="list-style-type: none"> 1. La libertad de usar el programa, con cualquier propósito; 2. De estudiar el funcionamiento del programa, y adaptarlo a las necesidades; 3. De distribuir copias, con lo cual se puede ayudar a otros, y 4. De mejorar el programa y hacer públicas las mejoras, de modo que toda la comunidad se beneficie.
Categorías	Garantía de las libertades propias del Software Libre
Dimensiones	
Dimensión	Indicador
Libertad de Uso.	Requerimiento de pago por el uso.
	Limite en el número de instalaciones.
	Limite en el número de usuarios.
	Restricción de uso.
Libertad de estudio y adaptación.	Disponibilidad del código fuente.
Libertad de Distribución.	Restricción de Distribución del software.
Libertad de Mejoras y Redistribución.	Restricción de distribución de modificaciones.

Fuente: Autores

Nota: Todos los indicadores tienen como índices: Si y No.

VARIABLE DEPENDIENTE: Portal Web que facilita la publicación de contenido y el acceso a información Bibliográfica y Académica.

PARTE OPERACIONAL: Una vez establecida la plataforma se construye sobre la misma el nuevo portal web universitario, en el cual los estudiantes pueden tener acceso a un contenido más dinámico y estableciéndose así un canal de comunicación entre ellos y la institución.

CUADRO # 4
Operacionalización de la variable dependiente.

Variable: Portal Web que facilita la publicación de contenido, el acceso a información Bibliográfica y Académica.	
Significado	Sitio web con facilidades para publicación dinámica de contenido, acceso a información bibliográfica y académica.
Categorías	Cumplimiento de las facilidades requeridas.
Dimensiones	
Dimensión	Indicador
Publicación de contenido	Editor de contenido WYSIWYG en el sitio.
	Clasificación del contenido.
	Flexibilidad en el diseño.
	Descentralización de la edición del contenido.
Acceso a Información Bibliográfica	Herramienta de búsqueda de material bibliográfico.
	Administración de información bibliográfica.
	Gestión de reserva de materiales.
	Historial de acceso a los materiales.
Acceso a Información Académica	Consulta del record académico.
	Facilidad de acceso a la información.
Correo Electrónico	Existencia de un WebMail.
	Acceso por parte de toda la Comunidad Universitaria.
Control de Acceso	Administración de usuarios y grupos de usuarios.
	Historial de acceso.
	Mantenimiento de perfiles.
	Seguridad de acceso a la información.

Fuente: Autores

Una vez establecidas las variables de estudio se determinó un Cuestionario, el que fue aplicado a una muestra de Estudiantes, Docentes y Personal administrativo, los cuales son los principales beneficiados con la implementación de esta propuesta.

2.6. TÉCNICAS E INSTRUMENTOS

El éxito de la investigación depende de la calidad de la información que se adquiera, tanto de las fuentes primarias como de las secundarias así como del procesamiento y presentación de la información.

2.6.1. TÉCNICAS

La técnica en investigación es un instrumento que permite facilitar el procesamiento de la información en la investigación. Considerando los objetivos de la investigación, en el presente trabajo se utilizó técnicas de tipo documental y de campo.

Documental

Lectura Científica, mediante esta técnica permitió ampliar el conocimiento del fenómeno en estudio, y realizar las propias conclusiones que permitieron profundizar el desarrollo de la investigación. No sólo se tomaron en cuenta fuentes impresas, sino que gran cantidad de material fue obtenido en forma digital desde internet.

Análisis de contenido, Redacción y estilo. Aspectos importantes que permitieron, la confiabilidad y validez del trabajo de investigación considerando la diferente información que se obtuvo de los diferentes instrumentos utilizados.

Campo

Encuesta, instrumento guía, mediante el cual se estableció el diagnóstico, la factibilidad y la propuesta para el desarrollo del portal web de la Universidad Estatal Península de Santa Elena utilizando software libre.

2.6.2. INSTRUMENTO

Con la finalidad de dar respuestas a los objetivos planteados en la investigación, se diseñó un instrumento que permitió recoger información de las principales necesidades que se presentan en el mantenimiento de un portal web, así como los servicios que demandan los estudiantes universitarios, para ello se utilizó la técnica de la encuesta, y se diseñaron dos cuestionarios, los mismos que constan en el anexo, con preguntas cerradas y con aplicación de la escala de tipo Likert.

2.7. ANÁLISIS DE RESULTADOS

Como ya se indicó, para la recolección de datos se definieron dos cuestionarios, el primero de ellos dirigido al personal técnico encargado de la Administración del Portal Web de la Universidad Estatal Península de Santa Elena, en este caso la población estaba compuesta solamente por una persona y en consecuencia la muestra también, por tal motivo se detalla directamente el resultado obtenido.

2.7.1. RESULTADO DE LA ENCUESTA AL PERSONAL TÉCNICO

En este cuestionario se incluyeron preguntas que permitieran evaluar cada una de las dimensiones obtenidas al operacionalizar la variable independiente ya que éstas se relacionan directamente con el proceso de administración de la página.

También se incluyeron preguntas relacionadas con la dimensión Publicación de contenido perteneciente a la operacionalización de la variable dependiente.

CUADRO # 5
Pago de licencias de uso.

¿Alguna de las herramientas requirió un pago para su uso?			
Item	Respuestas	#	%
1	Si.	0	0.00%
	No.	1	100.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas utilizadas para dar mantenimiento a la página web requirieron de un pago antes de poder utilizarse, lo cual nos ayudará a medir la libertad de uso. La respuesta obtenida fue que no se necesitó el pago de una licencia de uso. Se concluye entonces que las herramientas utilizadas eran de uso libre o se utilizaban con un tipo de licencia especial que no requiere pago.

CUADRO # 6
Límite de equipos.

¿Por lo menos una de las herramientas limita el número de computadores en donde se la puede utilizar?			
Item	Respuestas	#	%
2	Si.	0	0.00%
	No.	1	100.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas utilizadas limitan su uso a una cantidad de equipos, lo cual también ayudará a medir la libertad de uso. La respuesta obtenida fue que no existen límites en la cantidad de equipos en los que se pueden utilizar las herramientas. Se concluye entonces, que las herramientas utilizadas eran de uso libre o se utilizaban con un tipo de licencia especial que no limita su utilización a una cantidad de equipos.

CUADRO # 7
Límite de usuarios.

¿Existe en alguna de las herramientas un límite máximo de usuarios?			
Item	Respuestas	#	%
3	Si.	0	0.00%
	No.	1	100.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas tienen un límite de personas que pueden utilizarlas a la vez, lo cual también ayudará a medir la libertad de uso. La respuesta obtenida fue que no existen límites en la cantidad de usuarios que pueden estar utilizando a la vez las herramientas. Se concluye entonces, que las herramientas utilizadas eran de uso libre o se utilizaban con un tipo de licencia especial que no limita su uso a una cantidad de personas al mismo tiempo.

CUADRO # 8
Restricciones de uso.

¿Ha apreciado alguna otra restricción en el uso de estas herramientas?			
Item	Respuestas	#	%
4	Si.	0	0.00%
	No.	1	100.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas utilizadas tienen alguna restricción en su uso, adicional a las tratadas en las preguntas anteriores. La respuesta obtenida fue que no se ha apreciado algún otra restricción en el uso de las herramientas. Se concluye entonces, que las herramientas utilizadas eran de uso libre o se utilizaban con un tipo de licencia especial que no impone ninguna restricción de uso.

CUADRO # 9
Disponibilidad del código fuente.

¿Está disponible el código fuente de todas las herramientas?			
Item	Respuestas	#	%
5	Si.	1	100.00%
	No.	0	0.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas utilizadas tienen el código fuente disponible para su visualización y estudio. La respuesta obtenida fue que el código fuente de las herramientas se encuentra disponible. Se concluye entonces, que las herramientas utilizadas eran de uso libre o se utilizaban con un tipo de licencia especial que permite la obtención de su código fuente.

CUADRO # 10
Autorización de copia.

¿En caso de ser necesario copiar las herramientas, se requiere alguna autorización previa?			
Item	Respuestas	#	%
6	Si.	0	0.00%
	No.	1	100.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas utilizadas requieren una autorización previa para poder distribuir las. La respuesta obtenida fue que las herramientas no requieren una autorización para su distribución. Se concluye entonces, que las herramientas utilizadas eran de uso libre o se utilizaban con un tipo de licencia especial que permite su libre distribución.

CUADRO # 11
Facilidad de publicación de contenido.

¿Cuenta con un editor integrado similar a un procesador de textos que facilite la publicación de contenido en el sitio web?			
Item	Respuestas	#	%
7	Si.	1	100.00%
	No.	0	0.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si en el conjunto de herramientas utilizadas, existe una herramienta que permita una fácil publicación de contenido. La respuesta obtenida fue que este tipo de herramienta existe. Se concluye entonces, que las herramientas utilizadas brindan una manera fácil de publicación de contenido.

CUADRO # 12
Facilidad de clasificación de contenido.

¿Existe la posibilidad de clasificar de alguna forma el contenido ingresado?			
Item	Respuestas	#	%
8	Si.	0	0.00%
	No.	1	100.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas utilizadas permiten una fácil clasificación del contenido una vez que este a sido publicado en el sitio web. La respuesta obtenida fue que con el conjunto de herramientas utilizado no se podía clasificar de manera sencilla el contenido publicado en el sitio web. Se concluye entonces, que las herramientas utilizadas no facilitan la clasificación del contenido publicado.

CUADRO # 13
Facilidad de rediseño de la página web.

¿Se puede modificar fácilmente el diseño y la distribución del sitio web?			
Item	Respuestas	#	%
9	Si.	0	0.00%
	No.	1	100.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas utilizadas facilitan el rediseño y redistribución de los componentes del sitio web en caso de ser requerido. La respuesta obtenida fue que con el conjunto de herramientas utilizado no se podía rediseñar de forma fácil y oportuna el sitio web. Se concluye entonces, que las herramientas utilizadas no facilitan el rediseño del sitio web.

CUADRO # 14
Facilidad de publicación de contenido.

¿Existe la posibilidad de que usuarios no especializados aporten directamente con contenido al sitio web?			
Item	Respuestas	#	%
10	Si.	0	0.00%
	No.	1	100.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas utilizadas habilitan a usuarios sin mucho o ningún conocimiento en el uso de herramientas de desarrollo web, a la publicación de contenido en el sitio web. La respuesta obtenida fue que con el conjunto de herramientas utilizado estas personas no pueden colaborar con la publicación de contenido. Se concluye entonces, que las herramientas utilizadas hacían imposible que este tipo de usuarios puedan publicar contenido.

CUADRO # 15
Posibilidad de administrar usuarios y grupos.

¿Permite administrar usuarios y grupos de usuarios?			
Item	Respuestas	#	%
11	Si.	0	0.00%
	No.	1	100.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas utilizadas facilitan la definición de usuarios y grupos con acceso a la página web. La respuesta obtenida fue que el conjunto de herramientas no permite administrar usuarios y grupos de usuarios. Se concluye entonces, que las herramientas utilizadas imposibilitan la tarea de administración de usuarios y grupos de usuarios.

CUADRO # 16
Almacenamiento del historial de visitas.

¿Existe algún historial para revisar el acceso de usuarios?			
Item	Respuestas	#	%
12	Si.	0	0.00%
	No.	1	100.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas utilizadas generan un historial de los accesos realizados al contenido publicado en la página web. La respuesta obtenida fue que el conjunto de herramientas no generan un historial de visitas. Se concluye entonces, que las herramientas utilizadas imposibilitan la revisión de un historial de visitas que permita obtener estadísticas de uso de la página web.

CUADRO # 17
Nivel de seguridad en el acceso a información.

¿Impide el acceso no autorizado a información reservada para usuarios específicos?			
Item	Respuestas	#	%
13	Si.	0	0.00%
	No.	1	100.00%
	Total	1	100.00%

Fuente: Encuesta

Elaborado: Autores

Esta pregunta permite conocer si las herramientas utilizadas impiden el acceso no autorizado a información de carácter reservado. La respuesta obtenida fue que el conjunto de herramientas no impiden que usuarios no autorizados accedan a información reservada. Se concluye entonces, que las herramientas utilizadas no garantizan protección a información reservada.

Este primer cuestionario fue orientado a conocer la existencia de restricciones de uso impuestas por el fabricante de las herramientas de administración de la página web, así como la complejidad de su operación.

Otro factor evaluado es la disponibilidad de herramientas para tareas complementarias como la gestión de usuarios, control de accesos, entre otras.

2.7.2. RESULTADO DE LA ENCUESTA A LOS USUARIOS DE LA PÁGINA WEB

El segundo cuestionario fue enfocado a los usuarios de la página web, que previamente fueron identificados al definir la población y contiene preguntas realizadas con el fin de evaluar cada una de las dimensiones resultantes de la operacionalización de la variable dependiente a excepción de la dimensión Publicación de contenido. La tabulación de la información obtenida se detalla a continuación.

CUADRO # 18
Cifras sobre la clasificación de la muestra.

¿Es usted estudiante, docente o parte del personal administrativo?			
Item	Respuestas	#	%
1	Estudiante.	331	91%
	Docente.	17	4%
	Personal Administrativo.	17	5%
	Total	365	100.00%

Fuente: Encuesta

Elaborado: Autores

GRÁFICO # 3
Clasificación de la muestra



Fuente: Encuesta

Elaborado: Autores

Esta pregunta busca verificar el cumplimiento de la proporción de estudiantes, docentes y personal administrativo, resultante al momento de definir la muestra. En los resultados de la encuesta se observó que la cantidad de estudiantes con respecto al número de docentes y personal administrativo es muy significativa. En vista de que la mayor cantidad de encuestados son estudiantes, éstos se convierten en los principales consumidores de los servicios que se ofertan en el portal web, lo cual nos permite tener una clara idea sobre las características de los componentes a seleccionar.

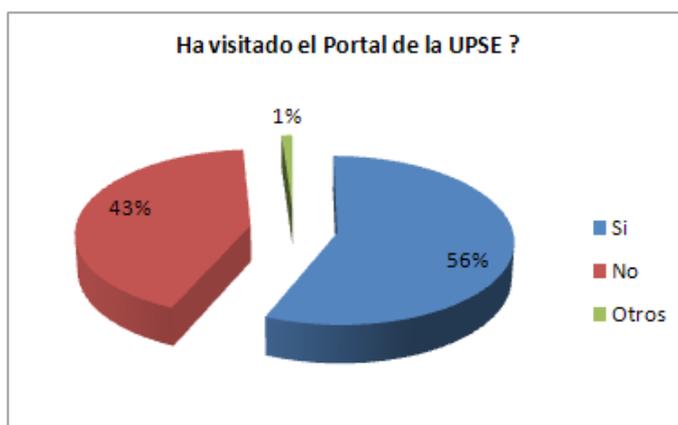
CUADRO # 19
Cifras sobre la utilización de la página web.

¿Ha visitado el portal de la UPSE?			
Item	Respuestas	#	%
2	Si.	206	56%
	No.	155	43%
	Otros.	4	1%
	Total	365	100%

Fuente: Encuesta

Elaborado: Autores

GRÁFICO # 4
Utilización de la página web



Fuente: Encuesta

Elaborado: Autores

En esta pregunta buscamos cuantificar la cantidad de personas que han accedido al menos una vez a la página web de la Universidad. Según los valores obtenidos, podemos apreciar que un poco más de la mitad de los estudiantes ha accedido al Sitio Web de la Universidad, seguido de una cantidad parecida que no lo han hecho. Los valores obtenidos delatan una situación preocupante debido a que cerca de la mitad de los potenciales usuarios no han accedido a la página web de la Universidad, pero a la vez nos crea una excelente oportunidad para promocionar este recurso tecnológico mediante la creación de servicios que incentiven a los estudiantes a hacer uso de esta nueva herramienta.

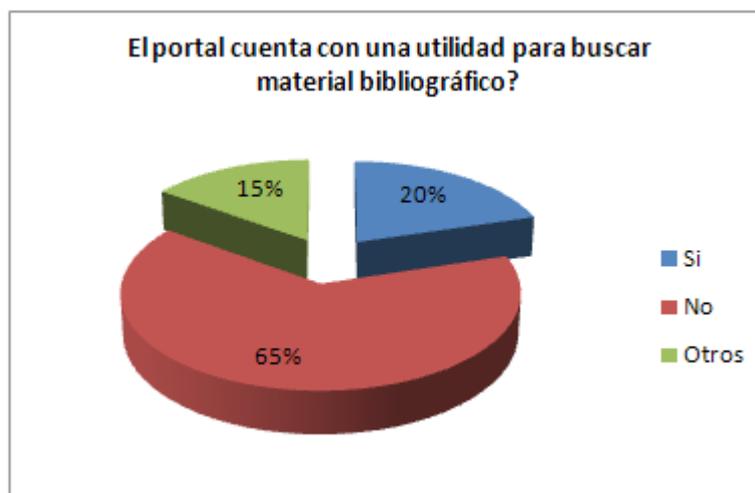
CUADRO # 20
Acceso a material bibliográfico desde la página.

¿El portal cuenta con una utilidad para buscar material bibliográfico?			
Item	Respuestas	#	%
3	Si.	74	20%
	No.	236	65%
	Otros.	55	15%
	Total	365	100%

Fuente: Encuesta

Elaborado: Autores

GRÁFICO # 5
Acceso a material bibliográfico desde la página



Fuente: Encuesta

Elaborado: Autores

Con esta interrogante buscamos determinar la cantidad de personas que conocen la realidad de la página con respecto al servicio de consultas de Material Bibliográfico. Un alto porcentaje (65%) indica que la página web de la Universidad no cuenta con una utilidad que permita realizar este tipo de búsquedas, seguido de un 20% que indica lo contrario. Se concluye entonces que: el portal no cuenta con una utilidad que permita catalogar y buscar libros, folletos, revistas y demás material para la consulta e investigación científica, con lo cual podemos visualizar claramente una oportunidad para la implementación de un nuevo servicio en el portal.

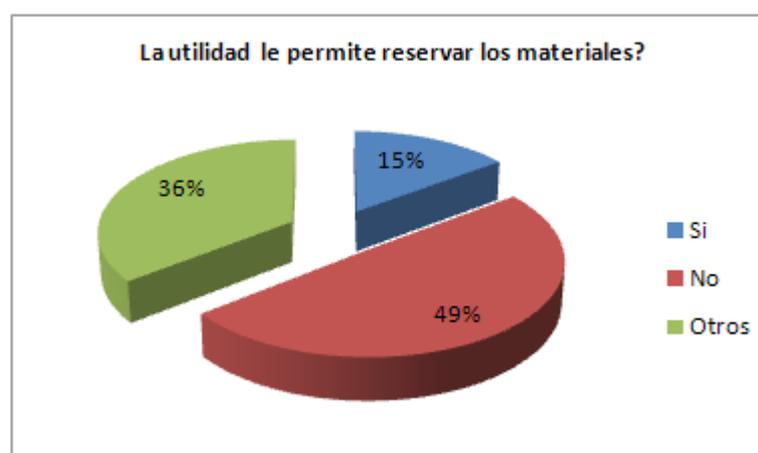
CUADRO # 21
Cifras sobre la reserva en línea de materiales bibliográficos.

¿La utilidad permite reservar los materiales?			
Item	Respuestas	#	%
4	Si.	54	15%
	No.	181	49%
	Otros.	130	36%
	Total	365	100%

Fuente: Encuesta

Elaborado: Autores

GRÁFICO # 6
Reserva en línea de materiales bibliográficos.



Fuente: Encuesta

Elaborado: Autores

En esta pregunta buscamos determinar el número de personas que conocen si la página web provee el servicio de reserva de material bibliográfico. Casi la mitad de potenciales usuarios indican que el portal no ofrece las facilidades para la reserva de libros, folletos, revistas y demás elementos disponibles en la Biblioteca Institucional. Con esto concluimos que en efecto no existe una utilidad que permita a los estudiantes reservar materiales bibliográficos utilizando el Portal Institucional, descubriendo con ello otro servicio adicional que se podría instalar en el Sitio Web.

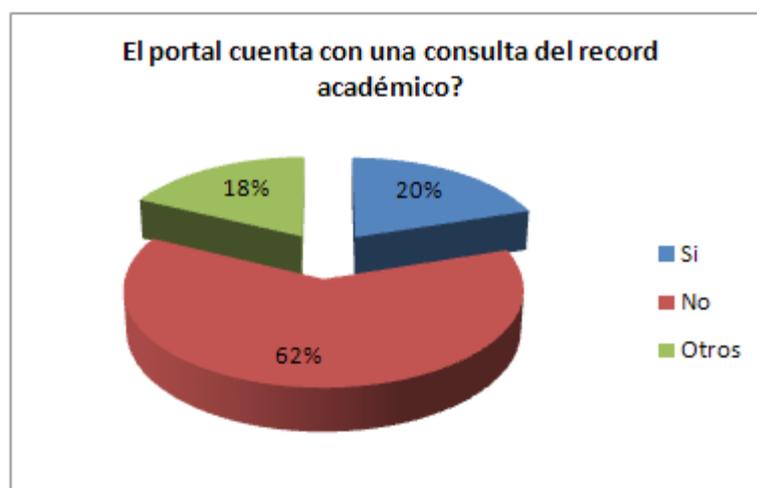
CUADRO # 22
Cifras sobre la consulta en línea del record académico.

¿El portal cuenta con una consulta del record académico?			
Item	Respuestas	#	%
5	Si.	73	20%
	No.	227	62%
	Otros.	65	18%
	Total	365	100%

Fuente: Encuesta

Elaborado: Autores

GRÁFICO # 7
Consulta en línea del récord académico.



Fuente: Encuesta

Elaborado: Autores

Esta interrogante nos permite cuantificar las personas que conocen si la página ofrece alguna utilidad que permita efectuar consultas al historial académico de los estudiantes. Como podemos apreciar, mas de la mitad de los usuarios potenciales manifestaron que el portal no cuenta con una herramienta para efectuar este tipo de consultas, dejando al descubierto una clara oportunidad para la creación de un nuevo servicio en el Portal Institucional. Es importante recalcar que la consulta del Record Académico es una utilidad de mucho interés para los Estudiantes.

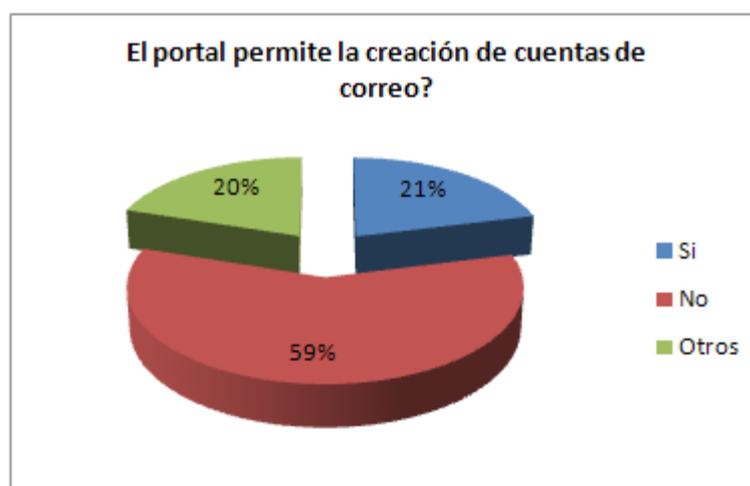
CUADRO # 23
Cifras sobre la creación de cuentas de correo electrónico.

¿El portal permite la creación de cuentas de correo?			
Item	Respuestas	#	%
6	Si.	77	21%
	No.	215	59%
	Otros.	73	20%
	Total	365	100%

Fuente: Encuesta

Elaborado: Autores

GRÁFICO # 8
Creación de cuentas de correo electrónico.



Fuente: Encuesta

Elaborado: Autores

En esta interrogante determinamos la cantidad de personas que conocen la realidad de la página con respecto al servicio de correo electrónico. Mas de la mitad de los encuestados indicó que el portal no permite la creación y uso de cuentas de correo, lo que nos permite confirmar que en efecto el portal carece de las facilidades necesarias para acceder a las cuentas de correo electrónico que se mantienen bajo el dominio upse.edu.ec. Esta conclusión nos lleva a visualizar una nueva oportunidad para la creación de un servicio de mucha utilidad en el Portal.

2.8. ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

Al interpretar las respuestas obtenidas en la primera encuesta, se puede notar que aparentemente no existen restricciones en cuanto al uso de las herramientas que sirven para administrar el sitio web, pero por otro lado, el nivel de complejidad que es requerido para manejarlas hace difícil que otras personas ajenas al departamento encargado de la Administración del Portal Web puedan colaborar directamente en la publicación de contenido.

De igual forma, se aprecia que estas herramientas son limitadas en características tales como la definición de usuarios o el control de acceso a la información publicada, siendo estos temas muy importantes a la hora de querer añadir otro tipo de funcionalidades a la página.

La segunda encuesta permitió comprobar que los usuarios están consientes de la falta de muchos de los servicios propuestos en la implementación de nuestro Proyecto.

La necesidad de poner a disposición de la Comunidad Estudiantil y de todas aquellas personas que de alguna u otra manera interactúan con la Institución Educativa, se da del hecho en que la actualidad todas las universidades ya prestan este tipo de servicios, y aunque en nuestro proyecto no se contempló el hecho de definir una aplicación de registro o matriculación en línea, la plataforma que se implementará dará todas las facilidades para que al ser desarrollada pueda integrarse sin mayor contratiempo.

CAPÍTULO III

3. DESARROLLO DEL PORTAL WEB DE LA UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA UTILIZANDO SOFTWARE LIBRE

3.1. ANÁLISIS DE LA PROBLEMÁTICA

La Universidad Estatal Península de Santa Elena, previo al inicio de este proyecto contaba con un sitio web orientado completamente a la difusión de información hacia la Comunidad.

La tecnología usada para la elaboración de la página, a pesar de que permite cumplir con los requerimientos básicos de un sitio informativo, nos limita al momento de integrar nuevos elementos interactivos que son característicos de los portales web modernos.

El Sitio Web de la Universidad Estatal Península de Santa Elena se encontraba desarrollado utilizando únicamente páginas estáticas, las cuales fueron elaboradas con el lenguaje HTML (HyperText Markup Language), el mismo que no permite crear efectos ni funcionalidades adicionales más allá de los enlaces entre páginas.

La evolución de las tecnologías asociadas con el Internet y particularmente las que posibilitan la elaboración de los sitios web, han hecho posible que se desarrollen sitios mucho más elaborados, que permiten acceder, de una manera fácil e integrada, a una serie de recursos y servicios adicionales con los cuales es posible interactuar, a este tipo de lugares en Internet se los denomina “Portales Web”.

3.1.1. PRINCIPALES COMPONENTES DE UN PORTAL WEB

La característica principal de un Portal Web se centra en la integración de varios servicios en una sola herramienta, permitiendo un acceso fácil y controlado a cada una de ellas. Es así que podemos identificar básicamente dos componentes:

- La herramienta básica que permite acceder a los recursos o servicios.
- Los recursos o servicios ofertados en el portal.

Existen una gran variedad de recursos y servicios que se pueden integrar en un portal web, los cuales van desde simples reportes del clima, hasta información altamente especializada como cotizaciones de acciones en alguna Bolsa de Valores.

La selección entre uno y otro depende básicamente de la orientación y necesidades del portal. Es así, que para el caso de la Universidad Estatal Península de Santa Elena hemos determinado la necesidad de integrar los siguientes recursos y servicios como componentes del portal.

- Consulta de documentación existente en la Biblioteca.
- Consulta de calificaciones por parte de los estudiantes.
- Acceso al Webmail Universitario.

Estos son solo algunos de los componentes que se integrarán en este trabajo, componentes que se implementarán o que ya existen en la Institución, como es el caso del Webmail. Sin embargo, la estructura del Portal permitirá la integración futura de otros componentes que se vayan creando como resultado natural de la evolución de nuestra institución.

Más adelante, en el Diseño de la Plataforma ahondaremos un poco mas en las herramientas que cumplirán cada una de estas funciones en nuestro portal.

3.1.2. DETALLE DE LOS REQUERIMIENTOS DEL PORTAL WEB

Lo antes expuesto permite definir un conjunto de requerimientos que deberá cumplir el nuevo Portal Web de la Universidad.

Flexibilidad al momento de organizar la información

El Portal Web debe estar preparado para organizar eficientemente los contenidos del sitio en secciones y categorías, facilitando la navegabilidad para los usuarios y creando una estructura sólida, ordenada y sencilla para los administradores. Desde el panel administración del Portal se puede crear, editar y borrar las secciones y categorías del sitio de la manera que mejor convenga a la institución.

Facilidad de publicación de contenidos

Posibilidad de crear un número ilimitado de páginas y editarlas desde un sencillo editor permitiendo formatear los textos con los estilos e imágenes deseados.

Escalabilidad e implementación de nuevas funcionalidades

Esta es una de las ventajas más significativas de un portal, ofrece la posibilidad de instalar, desinstalar y administrar componentes y módulos, que agregarán servicios de valor a los visitantes (galerías de imágenes, foros, noticias, encuestas, libros de visitas, etc.)

Administración de usuarios

Permitir almacenar datos de usuarios registrados y también la posibilidad de enviar e-mails masivos a todos los usuarios.

La administración de usuarios debe permitir clasificar a los usuarios de acuerdo a los niveles de privilegios que tendrán dentro de la gestión y administración del sitio.

Modificación del aspecto estético del sitio

Nos referimos con este punto a la posibilidad de cambiar todo el aspecto del sitio web de una manera sencilla, mediante la implementación de plantillas.

Elementos de navegación configurables

Los menús y demás elementos de navegación del sitio deben ser totalmente configurables desde la administración del sitio.

Administración de archivos

El Portal debe contar con herramientas que permitan una fácil administración de archivos a ser utilizados en el sitio, tales como imágenes, documentos, videos, etc.

Disposición de elementos modificables

La posición de los elementos que componen el sitio debe ser completamente configurable.

Estadísticas de visitas

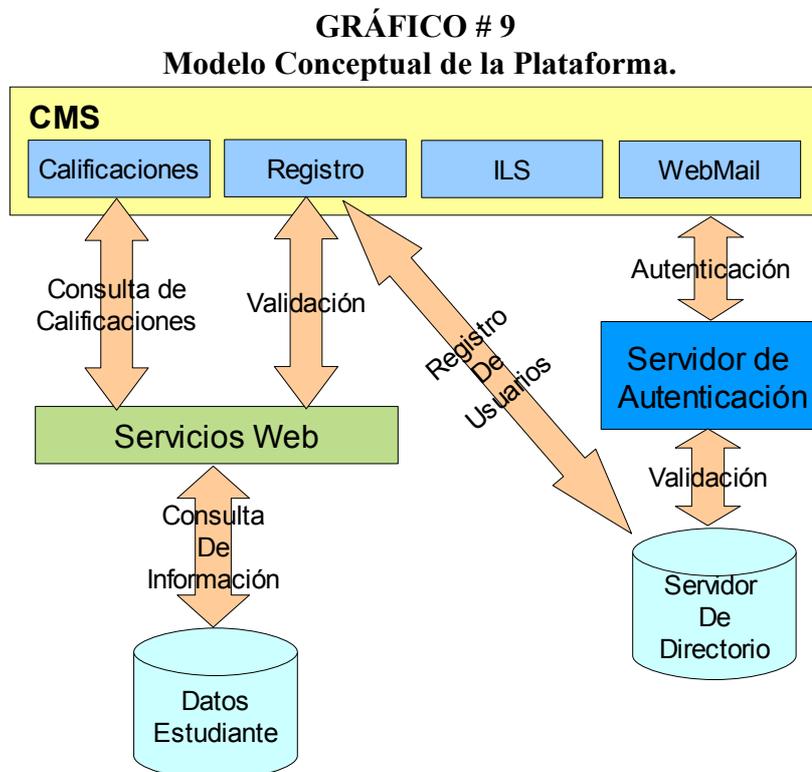
El portal debe contar con herramientas que permitan almacenar un registro detallado de las visitas realizadas para facilitar la administración del mismo.

Integración de recursos y servicios externos

Nos referimos con este punto a la facilidad con la que debe contar el portal para acceder a servicios y recursos externos de una manera integrada, dando al usuario la percepción de unidad entre todos ellos.

3.2. DISEÑO DE LA PLATAFORMA

Para ofrecer los servicios detallados en el análisis de la problemática, hemos definido la siguiente estructura:



Fuente: Universidad Estatal Península de Santa Elena.

Elaborado: Autores

3.2.1. MODELO CONCEPTUAL

En la estructura planteada podemos identificar los siguientes componentes:

Sistema de Gestión de Contenidos, CMS por sus siglas en inglés, es el componente que permitirá el acceso a los otros componentes del portal. Se trata de una aplicación web que provee facilidades para la publicación de contenido, administración de usuarios y acceso a aplicaciones externas, en caso de ser necesario.

En el gráfico se observa que el CMS interactúa con un Servidor de Autenticación, que valida el acceso de los usuarios.

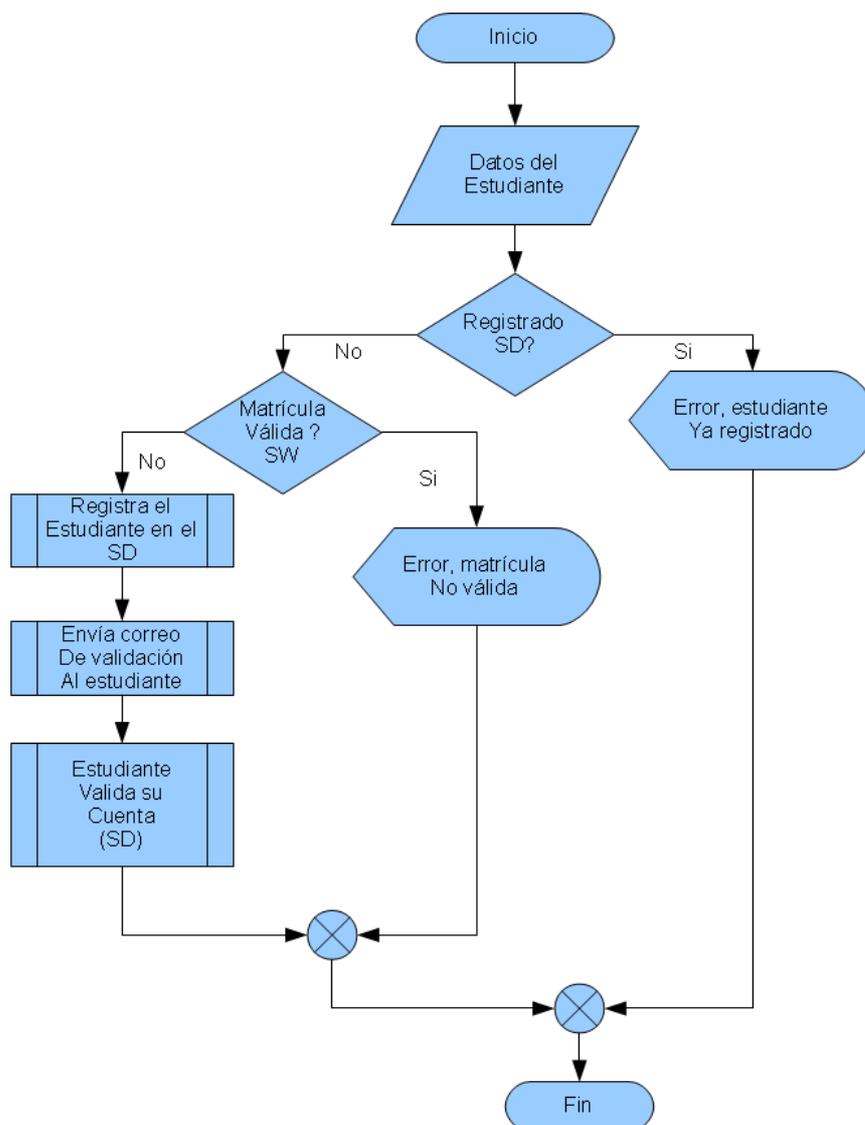
Aplicación de Consulta de Calificaciones, esta aplicación permite acceder a la base de datos de calificaciones de los estudiantes, sin embargo por cuestiones de seguridad, el acceso no se hace de manera directa sino mediante Servicios Web.

Servicios Web, (Webservices en inglés) tienen como principal función independizar la plataforma web de acceso público de la base de datos interna donde reposa la información académica de la institución, así, toda aplicación que desee información sobre un estudiante en particular deberá hacer uso únicamente de estos servicios y por ende solo accederá a la información definida en ellos.

Registro de Estudiantes, el registro de estudiantes permitirá capturar y validar la información de los estudiantes interesados en hacer uso de los servicios ofrecidos por el Portal Web.

En el siguiente gráfico se muestra el flujo a seguir por esta aplicación, observamos que el primer paso es capturar los datos del estudiante, posteriormente verificamos en el Servicio de Directorio (SD) si el estudiante ya se encuentra registrado, si no esta registrado aun, procede a validar la información del estudiante en los archivos de la Institución, mediante el uso de los Servicios Web (SW).

GRÁFICO # 10
Diagrama de flujo del proceso de registro de un Estudiante.



Fuente: Autores.

En el caso de que la información proporcionada por el usuario sea válida, procede al registro del estudiante en el Servicio de Directorio, luego envía un correo con el enlace necesario para la activación de la cuenta.

Una vez que el estudiante recibió en su correo el enlace, procede a activar la cuenta, luego de lo cual termina el registro y el estudiante puede hacer uso de su cuenta.

Sistema Integrado de Biblioteca, ILS por sus siglas en inglés, este módulo permitirá el acceso al catálogo de material bibliográfico con el que cuenta la Biblioteca de la Universidad.

WebMail, las personas que tengan una cuenta de correo activa en la institución, podrán acceder a ella mediante este módulo.

Servidor de autenticación, el servidor de autenticación se encargará de validar la información del usuario que ingresará al CMS.

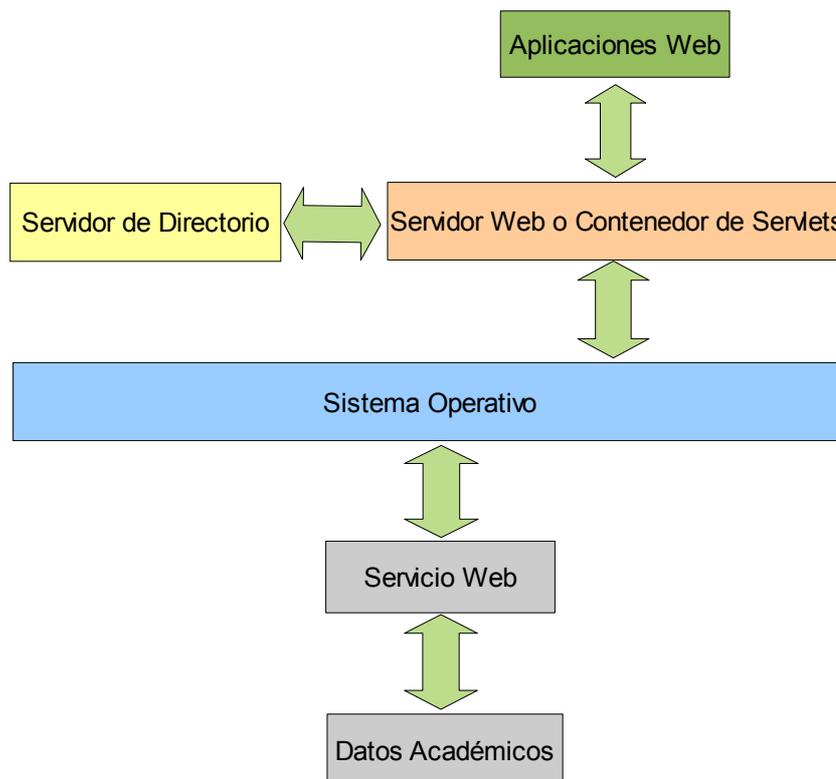
Las credenciales para el acceso son un Nombre de Usuario y su Contraseña, que se encriptará antes de almacenarse en el Servidor de Directorio.

Servidor de directorio, este componente se encargará de almacenar la información de los usuarios registrados en el portal.

3.2.2. PLATAFORMA DE EJECUCIÓN

Para la ejecución de todos estos componentes se necesita de ciertas herramientas básicas, las cuales se muestran en el siguiente diagrama de bloques.

GRÁFICO # 11
Diagrama de la Plataforma de Ejecución.



Fuente: Autores.

En este diagrama podemos identificar dos nuevos componentes que no figuraban en el modelo conceptual.

No fueron incluidos debido a que se asume su existencia, pues es imprescindible para la ejecución de las herramientas utilizadas en este trabajo. No obstante, pasamos a describirlas a continuación:

3.2.2.1. SERVIDOR WEB O CONTENEDOR DE SERVLETS

Estos componentes tienen como función proporcionar un entorno con todos los elementos necesarios para la ejecución de las aplicaciones web. Podemos contar entre estos elementos a las librerías, lenguajes de programación, extensiones a los lenguajes, compiladores, etc.

Sistema Operativo, permite la ejecución de todos los módulos detallados hasta el momento, pues su principal papel es el de servir de Interface entre el Hardware y el Usuario o la Aplicación que necesite hacer uso de él.

GRÁFICO # 12
Ubicación del Sistema Operativo en una Estructura Informática.



Fuente: <http://www.wikipedia.org>

3.3. SELECCIÓN DE LAS HERRAMIENTAS

En el apartado anterior definimos una serie de elementos que se utilizarán en el presente proyecto, sin embargo, de todas las alternativas existentes, aún no hemos seleccionado las más idóneas para su implementación final. En esta sección vamos a detallar las aplicaciones y herramientas consideradas y los criterios que se aplicaron para seleccionar las más idóneas.

3.3.1. PROCEDIMIENTO DE SELECCIÓN

Para seleccionar las herramientas utilizamos una adaptación del método denominado “*Punto de Función*”, al cual se hace referencia en el capítulo “Métricas Orientadas a la Función” del libro “Ingeniería del Software, un enfoque práctico. 5Ta edición.” escrito por Roger S. Pressman.

Este método es utilizado para determinar la funcionalidad del software, la adaptación realizada se detalla en el siguiente procedimiento:

1. Determinar los requerimientos generales para todos los componentes de la plataforma.
A estas características no se les asigna ningún peso debido a que su cumplimiento es obligatorio para cualquier componente seleccionado.
2. Determinar los requerimientos específicos de cada uno de los componentes.
3. Asociar un peso funcional a cada requerimiento de acuerdo a su relevancia para el proyecto.
4. Crear una matriz con los paquetes de software y las características requeridas.
5. Asignar un valor porcentual del cumplimiento de la herramienta con cada característica.
6. Calcular el valor funcional equivalente al porcentaje asignado.
7. Totalizar por cada herramienta los valores calculados en base al porcentaje de cumplimiento con cada característica. A este total lo denominaremos “*Valor Funcional*”
8. Seleccionar la herramienta que obtenga el mayor “*Valor Funcional*”.

El peso funcional en todos los casos puede tener un valor que va del 1 al 5, donde:

1 es Poco Importante y 5 significa Muy Importante.

3.3.2. DEFINICIÓN DE LOS REQUERIMIENTOS

REQUERIMIENTOS GENERALES

Todos los paquetes de software a considerar en este proyecto deben cumplir con las siguientes características básicas:

Ser clasificado como Software Libre, implica que la licencia bajo la cual se distribuye el software sea compatible con las licencias reconocidas por la Free Software Foundation (FSF). Entre estas licencias se encuentran: GPL, BSD, MPL y Copyleft. Se incluyen también en este grupo aquellos paquetes de software que se distribuyen con licencias aprobadas por la Open Source Initiative (OSI).

Ser compatible con Linux, hace referencia a la posibilidad de que el paquete de software se pueda instalar y ejecutar en el sistema operativo GNU/Linux.

REQUERIMIENTOS ESPECÍFICOS

Para seleccionar las herramientas de software que se implementarán, hemos realizado la pre-selección de 3 herramientas por cada categoría, a excepción de la herramienta para el Servidor Web, pues, en ese caso, la única opción a considerar es Apache Http Server (httpd.apache.org), debido a que hasta el momento no existe otra herramienta libre que se iguale con su versatilidad y eficiencia. Para los otros componentes a implementar, hemos realizado una exhaustiva investigación para la pre-selección, considerando parámetros como su antigüedad, disponibilidad de documentación, cantidad de implementaciones, disponibilidad de una comunidad de usuarios que soporte nuevos desarrollos, entre otras. Con las consideraciones antes mencionadas logramos obtener 3 alternativas por cada categoría, sobre las cuales hemos aplicado la metodología para la selección de la herramienta definitiva. En las siguientes páginas se muestra un resumen de las opciones consideradas y una breve descripción de cada una de ellas.

Sistema de Gestión de Contenidos (CMS)

CUADRO # 24
Pre-selección de herramientas para gestión de contenidos.

Herramienta	Descripción
Joomla! www.joomlaspanish.org	Entre sus principales características están la edición sencilla del contenido del sitio web. Es una aplicación de código abierto construida mayoritariamente en PHP bajo una licencia GPL. Puede trabajar en Internet o intranets y requiere de una base de datos MySQL, así como, preferiblemente, de un servidor HTTP Apache.
Tikiwiki tikiwiki.org	Es un CMS de índole colaborativa fácil de configurar y personalizar, sirve para crear portales, sitios comunitarios, intranets y aplicaciones web.
Drupal drupal.org	Es un CMS modular multipropósito y muy configurable que permite publicar artículos, imágenes, u otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs y administración de usuarios y permisos. El contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno Web.

Fuente: Autores

Sistema Integrado de Biblioteca (ILS)

CUADRO # 25
Pre-selección de Sistemas Integrados de Biblioteca.

Herramienta	Descripción
PhpMyBibli www.sigb.net	PMB (PhpMyBibli) es un ILS muy completo, fue iniciado por el director de la Librería Pública de Agneau, François Lemarchand en Octubre del 2002. En la actualidad el proyecto es mantenido por una compañía francesa denominada PMB Services.
OpenBiblio obiblio.sourceforge.net	Creado en el 2002 bajo licencia GNU/GPL. Está escrito en PHP y contiene las funcionalidades de Acceso al catálogo desde Internet, circulación de libros, catalogación y administrador de bibliotecarios.
Koha www.koha.org	Koha es un ILS único por ser el primero de código fuente abierto, liberado bajo la licencia GPL. Fue creado en 1999 por Katipo Communications para la Horowhenua Library Trust en Nueva Zelandia. La primera instalación se logró en enero del 2000.

Fuente: Autores

WebMail

CUADRO # 26
Pre-selección de Herramientas para Webmail.

Herramienta	Descripción
Zimbra www.zimbra.com	Esta Suite de Colaboración fue creada por Zimbra Inc. Posee tanto el componente de servidor como el cliente. Existen varias versiones disponibles: una soportada por la comunidad de Software Abierto, y otras con parte del código cerrado y soportada comercialmente con algunas mejoras.
Sogo www.scalablego.org	Fue creado en el 2004 y se basa en el proyecto OpenGroupware.org (OGGo). SOGo se especializa en la gestión del calendario, lista de direcciones y correo.
Zarafa www.zarafa.com	Zarafa es una herramienta de colaboración que permite la gestión de correos en el lado del servidor mediante su cliente web basado en Ajax denominado WebAccess. Todos los componentes del lado del servidor y el WebAccess están licenciados bajo AGPL. Algunas funciones avanzadas están disponibles solamente en las versiones comerciales.

Fuente: Autores

Servidor de autenticación

CUADRO # 27
Pre-selección de Servidores de Autenticación.

Herramienta	Descripción
CAS Server www.jasig.org	El Servicio de Autenticación Centralizada (CAS) es un protocolo que tiene como finalidad permitir que un usuario acceda a múltiples aplicaciones ingresando una sola vez sus credenciales (Usuario y Password). Fue creado originalmente por la Universidad de Yale para permitir autenticar usuarios de una manera segura.
OpenSSO opensso.dev.java.net	Es un sistema Single Sign-On (SSO), lo que significa que las aplicaciones que lo usen permitirán a sus usuarios proporcionar sus credenciales una sola vez y quedar identificados en el resto de aplicaciones sin volver a autenticarse.
OpenID openid.net	Es un standard de identificación digital descentralizado, con el que un usuario puede identificarse en una página web a través de una URL y puede ser verificado por cualquier servidor que soporte el protocolo. En los sitios que soporten OpenID, los usuarios no crean una nueva cuenta, solo necesitan disponer de un identificador creado en un servidor que verifique OpenID.

Fuente: Autores

Servidor de Directorio

CUADRO # 28
Pre-selección de Servidores de Directorio.

Herramienta	Descripción
OpenLDAP www.openldap.org	LDAP es un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red. OpenLDAP es una implementación libre y de código abierto de LDAP desarrollada por el proyecto OpenLDAP. Está liberada bajo su propia licencia OpenLDAP Public License.
Centos-DS www.redhat.com/directo ry_server	Red Hat Directory Server es un servidor basado en LDAP diseñado para centralizar las configuraciones de las aplicaciones, los perfiles de usuario, datos del grupo de trabajo, políticas e información para el control de acceso. CentOS-DS es la adaptación de RedHat-DS para CentOS.
OpenDS www.opens.org	OpenDS es un proyecto para crear un servicio de directorio fácil de implementar y usar. OpenDS está construido en su mayor parte con Java y se basa en los estándares LDAP y DSML.

Fuente: Autores

Contenedor de Servlets

CUADRO # 29
Pre-selección de Contenedores de Servlets.

Herramienta	Descripción
Tomcat tomcat.apache.org	Funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Implementa las especificaciones de los servlets y de JavaServer Pages (JSP) de Sun Microsystems. Funciona en cualquier sistema operativo que disponga de la máquina virtual Java.
Jetty jetty.codehaus.org/jetty	Jetty es un servidor HTTP y un contenedor de Servlets escrito en Java. Se publicó como un proyecto de software libre bajo la licencia Apache 2.0. Es utilizado por otros proyectos, como los servidores de aplicación JBoss y Geronimo; y por el plug-in Google Web Toolkit para eclipse.
Glassfish glassfish.dev.java.net	Servidor de aplicaciones desarrollado por Sun Microsystems que implementa las tecnologías de la plataforma Java EE y permite ejecutar aplicaciones que siguen esta especificación. Posee una versión gratuita y de código libre y una versión comercial denominada Sun GlassFish Enterprise Server.

Fuente: Autores

3.3.3. DESCRIPCIÓN DE LAS CARACTERÍSTICAS ANALIZADAS

La siguiente es una lista de las características analizadas en los paquetes de software y su descripción:

Alta flexibilidad y detalle en la definición de privilegios

Se refiere principalmente a la cantidad y variedad de opciones del sistema para las que se puede definir criterios de seguridad.

Antigüedad en el mercado

Este parámetro básicamente mide el tiempo de existencia de la aplicación, lo cual nos da una cierta idea de la madurez de la misma, aunque no se puede considerar como el único parámetro para medirla.

Basado en estándares

Indica si la construcción del paquete de software se basa en estándares de desarrollo generalmente aceptados por la industria.

Clasificación del contenido por Secciones y/o Categorías

Facilidades con las que cuenta la aplicación para organizar la información.

Compatibilidad con el Sistema de Administración de Contenidos

Dado que uno de los principales componentes de la plataforma es el Sistema de Administración de Contenidos, este parámetro mide cuan compatible es el paquete de software con el Sistema de Administración de Contenidos seleccionado.

Cabe mencionar que este parámetro y los paquetes de software que los contemplan deben ser evaluados luego de la selección del CMS.

Compatibilidad con estándares de Información Bibliográfica

Define la cantidad de estándares soportados, específicamente aquellos que se refieren a la representación de información bibliográfica.

Compatibilidad con las herramientas de desarrollo

Con este valor representamos la compatibilidad del paquete con las herramientas de desarrollo utilizadas para la implementación de la plataforma. Consideramos también la cantidad de utilidades disponibles para el desarrollo y que son propias del paquete de software.

Compatible con otras herramientas del Portal

Representamos con este parámetro el nivel de integración del paquete de software con el resto de componentes del portal web.

Control de privilegios de acceso

Mide la cantidad de facilidades ofrecidas por la aplicación para la definición de privilegios de acceso.

Diseño de interfaz de usuario

Nos referimos con esto al diseño y versatilidad de la interfaz gráfica de la aplicación.

Disponibilidad de documentación

Se refiere a la cantidad de documentación disponible y que se refiere a aspectos tanto técnicos como de usuario final.

Disponibilidad de herramientas de integración en otros lenguajes

Para el caso de paquetes de software que ofrecen componentes de integración con lenguajes de programación, este parámetro mide la cantidad de lenguajes con los que es factible esta integración.

Disponibilidad de plantillas que definan el aspecto gráfico del sitio

En los casos donde es factible la personalización del entorno del usuario, este parámetro mide la disponibilidad de plantillas que permitan esta personalización.

Edición y publicación del contenido en línea

Nos referimos con este valor a las facilidades ofrecidas por la aplicación para la edición de contenido en línea, y su posterior publicación.

Escalable

La escalabilidad de una aplicación se refiere a la posibilidad de actualizar la versión o implementar mejoras sin perder la información que se encuentre en ella.

Facilidad de Administración

La facilidad de administración evalúa la disponibilidad de una herramienta administrativa para la aplicación y la facilidad de uso de esa herramienta.

Facilidad de configuración del ambiente de pruebas

Para el caso de los paquetes de software que estén relacionados con el desarrollo de aplicaciones, este parámetro mide la posibilidad de implementar un ambiente de pruebas, así como la facilidad para actualizar estos ambientes.

Facilidad de instalación

Nos referimos con Facilidad de Instalación a la sencillez del proceso de instalación y configuración inicial.

Facilidad de modificación del código fuente

Evaluamos con este parámetro la documentación interna de los programas del componente de software, lo cual nos permite intuir la dificultad que representaría una modificación en ellos. Consideramos también la disponibilidad y claridad de la documentación técnica asociada.

Facilidad de uso

Con este valor representamos cuan intuitiva y sencilla es la interfaz para el usuario final, así como la disponibilidad y claridad de la documentación que se refiera al uso del programa.

Frecuencia de uso en aplicaciones similares o instituciones gubernamentales

Con este parámetro evidenciamos el uso de la aplicación en otras instituciones relacionadas al gobierno Ecuatoriano.

Gestión de la base de datos transparente para el usuario final y el administrador

Con este parámetro representamos el nivel de abstracción que ofrece la aplicación con respecto a su modelo de datos, especialmente en el proceso de instalación.

Gestión integrada de archivos gráficos y demás recursos del sitio

Este indicador se refiere a la cantidad y calidad de herramientas con las que cuenta la aplicación y que permiten la gestión de información multimedia.

Herramientas de búsqueda de información en el sitio

Indica si la aplicación cuenta con herramientas para buscar información en el contenido del sitio.

Herramientas de búsqueda de material bibliográfico

Con este valor nos referimos a las herramientas elaboradas para ubicar libros y demás publicaciones, dentro de un ILS. Integrable con el Sistema de Gestión de Contenidos. Con integrable hacemos referencia a la posibilidad de acceder a la aplicación desde el Sistema de Gestión de Contenido seleccionado y en especial a las facilidades para incrustar la aplicación dentro de las opciones y el contenido del CMS.

Integrado con el Sistema Operativo

Este valor permite medir el nivel de integración y compatibilidad de la aplicación con el Sistema Operativo. También consideramos aquí la facilidad de configuración de la aplicación para lograr esta integración.

Interacción con un Servidor de Autenticación Centralizada

Este indicador muestra la disponibilidad de herramientas que permitan lograr el intercambio de información entre la aplicación y el servidor de autenticación.

Interacción con un Servidor de Directorio LDAP

Este valor aplica para aquellas aplicaciones que necesitan interactuar con un Servidor de Directorio. A diferencia del resto de parámetros, en éste nos referimos directamente al servidor LDAP por ser considerado un estándar de los servidores de directorio. La plataforma desarrollada en este trabajo utiliza la versión libre de este servidor denominada OpenLDAP.

Modular

Se refiere a la capacidad de la aplicación para modificar su funcionalidad en base a la instalación de módulos adicionales a los que se instalan en su versión inicial. Se considera también la disponibilidad de herramientas para administrar estos módulos así como la facilidad del procedimiento de instalación.

Número ilimitado de usuarios

Este valor se refiere a la posibilidad de definir un número ilimitado de usuarios en aquellas aplicaciones que poseen o administran servicios de autenticación.

Personalizable

Con personalización nos referimos a la capacidad de adaptar la apariencia y funcionalidad de la aplicación a los requerimientos de nuestra plataforma.

Personalización de los atributos almacenados

Se relaciona específicamente a las herramientas de Servicio de Directorio y se relaciona con la posibilidad de modificar los atributos de los objetos almacenados en el directorio para adaptarlos a las necesidades de la plataforma.

Personalización del entorno gráfico

En este caso medimos las opciones de la aplicación que permitan modificar la apariencia de la interfaz gráfica del usuario.

Plataforma adaptable

Algunas de las aplicaciones instaladas en este trabajo poseen una plataforma de ejecución, este indicador mide la disponibilidad de las herramientas necesarias para adaptarla a las nuestras necesidades.

Similitud funcional con aplicaciones de escritorio

Una de las más notables deficiencias de las aplicaciones basadas en web es la diferencia que estas tienen cuando se las compara con aplicaciones de escritorio, la cual se evidencia aún más cuando consideramos la interfaz gráfica de usuario y su operación generalmente basada en ventanas. En la actualidad, tecnologías como AJAX permiten mejorar dramáticamente la experiencia del usuario al utilizar una aplicación Web. Este indicador mide el grado de similitud que una aplicación web tiene con respecto a una aplicación de escritorio poniendo especial énfasis en la Interfaz Gráfica de Usuario.

Soporte de Certificados Digitales

Cuando se transfiere información por Internet, una de las principales preocupaciones es la seguridad. Una de las estrategias más utilizadas es la encriptación de datos, para lo cual generalmente se utilizan los certificados digitales. Con el soporte nos referimos a la facultad de la aplicación para la instalación y uso de los Certificados Digitales en la transferencia de información.

Soporte de métodos de encriptación de información

Este parámetro se refiere a la posibilidad de encriptar la información antes de almacenarla en una base de datos, para protegerla de accesos no autorizados. No sólo se evalúa la capacidad de encriptación que la herramienta posee sino también cuales son los métodos de encriptación soportados.

Soporte para carga masiva de Materiales Bibliográficos

Dado que la instalación de una aplicación de gestión bibliográfica implica la carga del material existente y por lo general este material es abundante, se definió este parámetro para medir la disponibilidad de herramientas que faciliten este proceso.

Soporte para la interface de acceso del servidor de autenticación

Se refiere a la posibilidad de instalar e integrar el servidor de autenticación como parte de una aplicación, ya sea con la instalación de un módulo o mediante la utilización de una librería especializada.

3.3.4. EVALUACIÓN DE LAS HERRAMIENTAS PRE-SELECCIONADAS

En las siguientes páginas detallamos las características evaluadas para cada grupo de herramientas y los valores asignados a las preseleccionadas. En cada tabla se muestran las siguientes columnas:

- **Característica:** Contiene la característica a evaluar, su definición se encuentra en el apartado anterior.
- **Peso:** Refleja en una escala del 1 al 5 el nivel de importancia de cada característica para el proyecto. Donde 1 es poco importante y 5 es imprescindible.
- **% Cumplimiento:** Permite determinar el porcentaje de cumplimiento de cada aplicación con la característica evaluada. El porcentaje es asignado de manera empírica en base a la investigación y experiencia de los autores.
- **Puntaje:** Muestra el puntaje obtenido de de la relación porcentaje de cumplimiento con el peso del parámetro.

Al final de cada tabla en la columna “Puntaje” se muestra la sumatoria de los valores obtenidos por cada aplicación y se resalta en color rojo aquella que obtuvo la mejor calificación y que por ende es la seleccionada en esa categoría.

CUADRO # 30
Aplicaciones evaluadas por características.

Característica	CMS	ILS	Webmail	Autenticación	Servidor de Directorio	Servidor de Aplicaciones
Alta flexibilidad y detalle en la definición de privilegios.	√					
Antigüedad en el mercado.	√	√				
Basado en estándares.						√
Clasificación del contenido por secciones y/o categorías.	√					
Compatibilidad con el Sistema de Administración de Contenidos.			√			
Compatibilidad con Estándares de Información Bibliográfica.		√				
Compatibilidad con las Herramientas de Desarrollo.						√
Compatible con las otras herramientas del portal.				√		
Control de privilegios de acceso.		√				
Diseño de Interfaz de usuario.			√			
Disponibilidad de Documentación.	√					
Disponibilidad de herramientas de integración con otros lenguajes.				√		
Disponibilidad de plantillas que definan el aspecto gráfico del sitio.	√					
Edición y publicación de contenido en línea.	√					
Escalable.	√		√			
Facilidad de administración.					√	
Facilidad de configuración del ambiente de pruebas.						√
Facilidad de Instalación.	√	√	√	√	√	√
Facilidad de modificación del código fuente.	√	√				
Facilidad de uso.			√			
Frecuencia de uso en instituciones gubernamentales.	√		√			
Gestión de la base de datos transparente.	√	√				
Gestión integrada de archivos gráficos y demás recursos del sitio.	√					
Herramientas de búsqueda de información en el sitio.	√					
Herramientas de búsqueda de material bibliográfico.		√				
Integrable con el Sistema de Gestión de Contenido.						√
Integrado con el Sistema Operativo.					√	
Interacción con un servidor de autenticación centralizada.			√			
Interacción con un servidor de directorio LDAP.				√		
Modular.	√					
Número ilimitado de usuarios.	√					
Personalizable.				√		
Personalización de los atributos almacenados.					√	
Personalización del entorno gráfico.		√				
Plataforma Adaptable.			√			
Similitud funcional con aplicaciones de escritorio.			√			
Soporte de Certificados Digitales.				√		
Soporte de métodos de encriptación de información.					√	
Soporte para carga masiva de material bibliográfico.		√				
Soporte para la interface de acceso del Servidor de Autenticación.						√

Fuente: Autores

CUADRO # 31
Evaluación de los Sistemas de Administración de Contenidos y los Sistemas Integrados de Biblioteca.

SISTEMA DE ADMINISTRACIÓN DE CONTENIDO

Característica	Peso	% Cumplimiento			Puntaje		
		Joomla	TikiWiki	Drupal	Joomla	TikiWiki	Drupal
Facilidad de Instalación	4	100%	100%	80%	4.00	4.00	3.20
Gestión de la base de datos transparente para el usuario final y el administrador.	5	100%	100%	100%	5.00	5.00	5.00
Edición y publicación de contenido en línea.	5	100%	100%	100%	5.00	5.00	5.00
Clasificación del contenido por secciones y/o categorías.	3	100%	100%	100%	3.00	3.00	3.00
Modular.	4	100%	100%	100%	4.00	4.00	4.00
Escalable, disponibilidad de componentes.	4	100%	80%	90%	4.00	3.20	3.60
Número ilimitado de usuarios.	5	100%	100%	100%	5.00	5.00	5.00
Alta flexibilidad y detalle en la definición de privilegios.	5	80%	100%	90%	4.00	5.00	4.50
Disponibilidad de plantillas que definan el aspecto gráfico del sitio.	3	100%	70%	90%	3.00	2.10	2.70
Gestión integrada de archivos gráficos y demás recursos del sitio.	4	100%	100%	100%	4.00	4.00	4.00
Facilidad de modificación del código fuente.	4	100%	100%	100%	4.00	4.00	4.00
Herramientas de búsqueda de información en el sitio.	3	100%	100%	100%	3.00	3.00	3.00
Antigüedad en el mercado.	4	100%	80%	100%	4.00	3.20	4.00
Disponibilidad de Documentación	4	100%	80%	90%	4.00	3.20	3.60
Frecuencia de uso en aplicaciones similares o instituciones gubernamentales	4	100%	40%	60%	4.00	1.60	2.40
					60.00	55.30	57.00

SISTEMA INTEGRADO DE BIBLIOTECA

Característica	Peso	% Cumplimiento			Puntaje		
		PhpMyBibili	OpenBiblio	Koha	PhpMyBibili	OpenBiblio	Koha
Facilidad de Instalación	4	100%	100%	60%	4.00	4.00	2.40
Gestión de la base de datos transparente para el usuario final y el administrador.	5	100%	100%	100%	5.00	5.00	5.00
Control de privilegios de acceso	4	100%	100%	100%	4.00	4.00	4.00
Personalización del entorno gráfico	4	80%	90%	80%	3.20	3.60	3.20
Facilidad de modificación del código fuente.	4	100%	100%	80%	4.00	4.00	3.20
Herramientas de búsqueda de material bibliográfico.	5	100%	100%	100%	5.00	5.00	5.00
Antigüedad en el mercado.	4	90%	100%	90%	3.60	4.00	3.60
Soporte para carga masiva de material bibliográfico.	5	100%	100%	100%	5.00	5.00	5.00
Compatibilidad con Estándares de Información Bibliográfica	4	100%	100%	100%	4.00	4.00	4.00
					37.80	38.60	35.40

Fuente: Autores

CUADRO # 32
Evaluación de WebMails y Servidores de Autenticación.

WEBMAIL

Característica	Peso	% Cumplimiento			Puntaje		
		Zimbra	Sogo	Zarafa	Zimbra	Sogo	Zarafa
Facilidad de instalación	4	80%	100%	90%	3.20	4.00	3.60
Interacción con un servidor de autenticación centralizada.	5	100%	100%	100%	5.00	5.00	5.00
Compatibilidad con el Sistema de Administración de Contenidos.	4	100%	100%	100%	4.00	4.00	4.00
Facilidad de uso.	5	100%	80%	90%	5.00	4.00	4.50
Similitud funcional con aplicaciones de escritorio.	5	100%	90%	100%	5.00	4.50	5.00
Plataforma Adaptable.	5	100%	80%	90%	5.00	4.00	4.50
Escalable	4	100%	100%	100%	4.00	4.00	4.00
Frecuencia de uso en aplicaciones similares o instituciones gubernamentales	5	100%	50%	50%	5.00	2.50	2.50
Diseño de Interfaz de usuario	4	100%	70%	80%	4.00	2.80	3.20
					31.20	29.50	30.60

SERVIDOR DE AUTENTICACIÓN

Característica	Peso	% Cumplimiento			Puntaje		
		CAS	OpenSSO	OpenID	CAS	OpenSSO	OpenID
Facilidad de instalación	4	100%	80%	100%	4.00	3.20	4.00
Soporte de Certificados Digitales.	4	100%	100%	100%	4.00	4.00	4.00
Interacción con un servidor de directorio LDAP.	5	100%	100%	0%	5.00	5.00	0.00
Personalizable	3	100%	100%	0%	3.00	3.00	0.00
Compatible con las otras herramientas del portal	5	100%	80%	90%	5.00	4.00	4.50
Disponibilidad de herramientas de integración con otros lenguajes	4	100%	80%	100%	4.00	3.20	4.00
					25.00	22.40	16.50

Fuente: Autores

Como se aprecia en los cuadros anteriores las características aplicables difieren entre los grupos de paquetes de software, en algunos casos porque no son aplicables y en otros porque no son relevantes para nuestro análisis. Inclusive, el peso de la característica difiere entre ellos, esto se debe a que la importancia de cada característica también es diferente según el grupo de software al que se aplica. Todos estos detalles son fruto del procedimiento creado para la selección de herramientas y que ya definimos previamente.

CUADRO # 33
Evaluación de Servidores de Directorio y Servidor de Aplicación.

SERVIDOR DE DIRECTORIO

Característica	Peso	% Cumplimiento			Puntaje		
		OpenLDAP	CentosDS	OpenDS	OpenLDAP	CentosDS	OpenDS
Facilidad de instalación	4	100%	100%	100%	4,00	4,00	4,00
Soporte de métodos de encriptación de información.	5	100%	100%	100%	5,00	5,00	5,00
Facilidad de administración.	5	50%	100%	100%	2,50	5,00	5,00
Personalización de los atributos almacenados.	3	100%	100%	100%	3,00	3,00	3,00
Integrado con el Sistema Operativo	4	100%	100%	80%	4,00	4,00	3,20
					18,50	21,00	20,20

SERVIDOR DE APLICACIONES

Característica	Peso	% Cumplimiento			Puntaje		
		Tomcat	Jetty	Glassfish	Tomcat	Jetty	Glassfish
Facilidad de instalación	4	100%	100%	100%	4,00	4,00	4,00
Facilidad de configuración del ambiente de pruebas.	5	100%	100%	100%	5,00	5,00	5,00
Integrable con el Sistema de Gestión de Contenido.	5	100%	100%	100%	5,00	5,00	5,00
Soporte para la interface de acceso del Servidor de Autenticación.	5	100%	80%	80%	5,00	4,00	4,00
Basado en estándares	4	100%	100%	100%	4,00	4,00	4,00
Compatibilidad con las Herramientas de Desarrollo	4	100%	80%	90%	4,00	3,20	3,60
					27,00	25,20	25,60

Fuente: Autores

En los cuadros 31, 32 y 33 pudimos observar el desarrollo del procedimiento diseñado para la selección de las herramientas a utilizar en el presente proyecto. Como resultado hemos obtenido la lista de aplicaciones que mas se ajustan a nuestros requerimientos. Queda por lo tanto determinar las estrategias a seguir para integrarlas segun el modelo conceptual definido en el numeral 3.2 (Diseño de la Plataforma). Los procedimientos de instalación, configuración y adaptación de los paquetes seleccionados, así como los respectivos manuales de uso (donde aplican) se muestran en sendos anexos que se encuentran al final del presente documento y en algunos casos en documentos digitales contenidos en el CD que acompaña al presente trabajo.

3.3.5. RESUMEN DE APLICACIONES SELECCIONADAS

Las aplicaciones seleccionadas para integrar la plataforma son:

CUADRO # 34
Aplicaciones seleccionadas para la plataforma.

Categoría	Aplicación
Sistema de Gestión de Contenido	Joomla!
Sistema Integrado de Biblioteca	Openbiblio
Webmail	Zimbra
Servidor de Autenticación	CAS Server
Servidor de Directorio	CentOS Directory Server
Servidor de Aplicaciones	Tomcat

Fuente: Autores

3.4. SOFTWARE DESARROLLADO

3.4.1. APLICACIONES DESARROLLADAS

Como parte de este proyecto se desarrollaron un conjunto de aplicaciones, detalladas a continuación:

Consulta en línea de calificaciones

Esta aplicación permite que los estudiantes de la UPSE puedan consultar sus calificaciones en línea. El software fue desarrollado utilizando ZK Framework (<http://www.zkoss.org/>) como herramienta para la Interfaz Gráfica de Usuario y Apache Axis (<http://ws.apache.org/axis/>) como herramienta para la interacción con el Webservice de la UPSE que permite recuperar las calificaciones. La aplicación se ejecuta sobre Apache Tomcat (<http://tomcat.apache.org/>) o superior.

Registro en línea de estudiantes

Esta aplicación fue diseñada para que los estudiantes de la UPSE se registren en el portal universitario y así puedan acceder a las aplicaciones alojadas en él.

Este componente se desarrolló utilizando PHP 5 (<http://php.net/index.php>) y usa phpCAS (<http://www.ja-sig.org/wiki/display/CASC/phpCAS>) para la interacción con el servidor de autenticación.

3.4.2. MODIFICACIONES A LOS COMPONENTES DE LA PLATAFORMA

Además de las aplicaciones desarrolladas se realizaron las modificaciones necesarias en los componentes de la plataforma para integrarlas al Servidor de Autenticación Centralizada.

A continuación detallamos las modificaciones efectuadas:

Joomla!

Para la interacción del sistema de gestión de contenidos con el servidor de autenticación se procedió a instalar Joomla Authentication Manager (http://joomlancode.org/gf/project/auth_manager/).

OpenBiblio

Para que OpenBiblio interactúe con el servidor de autenticación se realizaron las modificaciones respectivas en el módulo de validación de acceso utilizando la librería phpCAS (<http://www.ja-sig.org/wiki/display/CASC/phpCAS>).

3.5. IMPLEMENTACIÓN DE LA PLATAFORMA

Para la implementación de la plataforma se siguió el siguiente orden de instalación de los componentes, los procesos de instalación se detallan en los anexos correspondientes:

- Instalación de CentOS
- Instalación de Apache
- Instalación de Tomcat
- Instalación de MySQL
- Instalación de Joomla
- Instalación de Openbiblio (EspaBiblio)
- Instalación de Zimbra
- Instalación de CentOS-DS
- Instalación de CAS-Server
- Instalación de la Aplicación de Registro de Estudiantes
- Instalación de la Aplicación de Consulta de Aplicaciones

3.6. PRUEBAS

Para las pruebas de la plataforma se la instaló en una máquina virtual de VirtualBox (<http://www.virtualbox.org/>) utilizando el procedimiento detallado en el apartado anterior. Esta máquina virtual posteriormente se utilizará para implementar el plan de contingencia del portal web.

CONCLUSIONES

- Una buena alternativa para solucionar las necesidades de las empresas o instituciones del medio, es la adaptación de herramientas disponibles bajo el esquema de software libre, muchas de las cuales han alcanzado un alto grado de madurez debido al continuo proceso de mejoramiento llevado a cabo por las comunidades de desarrolladores.
- Dentro de un proyecto de adaptación de software, el proceso de elección de las herramientas a utilizar es una de las etapas más importantes, ya que se debe evaluar minuciosamente las características más relevantes con la finalidad de garantizar que el producto final cumpla con las especificaciones del cliente.
- La plataforma implementada para la Universidad Estatal Península de Santa Elena, queda establecida como una base sólida sobre la cual se podrán establecer recursos que aprovechen todas sus bondades y que mantengan continuamente renovada la imagen del Portal Web Universitario.
- En el mercado se nota una creciente tendencia al uso de aplicaciones web como soluciones informáticas a las necesidades de gestión administrativa, financiera, académica e incluso en algunos casos de producción, por esta razón en el presente proyecto hemos puesto especial énfasis en aquellas herramientas necesarias para el desarrollo de aplicaciones web, las cuales, si se ajustan a los lineamientos especificados en este trabajo, se integrarán de manera sencilla con la plataforma instalada.
- Una de las ventajas del Software Libre es su constante evolución, lo cual hace que continuamente se estén generando mejoras o parches de seguridad de las herramientas instaladas, sin embargo, se debe tener especial cuidado a la hora de seleccionar el origen de estas actualizaciones, pues puede ocasionar potenciales agujeros de seguridad.

RECOMENDACIONES

- La herramienta de desarrollo de aplicaciones seleccionada en este trabajo tiene un gran potencial, por tanto, es recomendable que los profesionales dedicados al desarrollo de soluciones informáticas de la Universidad, profundicen lo más que puedan en su manejo, a fin de explotar todas sus características.
- En el apartado 3.3 (Selección de las Herramientas), detallamos una metodología basada en técnicas de Ingeniería de Software, para la selección de las herramientas a utilizar en nuestro proyecto. Recomendamos el uso de este procedimiento para la determinación de cualquier paquete adicional a implementar en la Institución, pues permite identificar claramente la mejor opción de entre las alternativas pre-seleccionadas.
- Como parte del presente trabajo hemos detallado una serie de guías que permiten el uso y administración de la plataforma instalada en el Portal Web de la Universidad Estatal Península de Santa Elena. Sugerimos utilizar estos documentos para facilitar la implementación de cualquier elemento adicional en el portal y así asegurar su integración con el resto de herramientas.
- Dada la versatilidad de las aplicaciones web, evidenciada especialmente en la facilidad de distribución y uso, recomendamos considerar esta plataforma como una alternativa para el desarrollo de futuras aplicaciones informáticas en la institución.
- Con el ánimo de mantener actualizados los componentes de software de la plataforma instalada, es necesario, verificar periódicamente la disponibilidad de nuevas versiones o parches de seguridad para estas aplicaciones, procurando seleccionar solo los repositorios auspiciados por las comunidades de desarrollo de cada una de ellas.

BIBLIOGRAFÍA

ARNOLD Ken, GOSLING James, HOLMES David, 2001, Programación, “El lenguaje de programación Java”, Prentice Hall.

AUMAILLE Benjamin, 2002, Programación, “J2EE Desarrollo de Aplicaciones Web”, Ediciones ENI.

CHAKRABARTI Anirban, 2007, Seguridad Informática, “Grid Computing Security”, Springer Berlin Heidelberg New York.

CHEN Henry & CHENG Robbie, 2007, Programación, “ZK Ajax without JavaScript Framework”, 1ra. Edición, firstPress.

DÍAZ Ma. Paloma, 2005, Programación, “Ingeniería de la Web y patrones de diseño”, Prentice Hall.

GAMMA Erich, HELM Richard, JOHNSON Ralph, VLISSIDES John, WATSON Thomas, 2002, Programación, “Patrones de diseño”, Prentice Hall.

KENDALL Kenneth & KENDALL Julie, 2005, Análisis de Sistemas, “Análisis y diseño de sistemas”, 3ra. Edición, Editorial Pearson.

KENNARD James, 2007, Programación, “Mastering Joomla! 1,5 Extension and Framework Development”, 1ra. Edición, Packt Publishing.

KRUG Steve, 2006, Programación, “No me hagas pensar”, Prentice Hall.

MARTÍN Gregorio, 2005, Programación, “Curso de XML”, Prentice Hall.

MÉNDEZ Carlos, 2006, Metodología, “Diseño y desarrollo del proceso de investigación con énfasis en ciencias empresariales”, editorial LIMUSA.

POWERS Lars & SNELL Mike, 2006, Programación, “Microsoft Visual Studio 2005 Unleashed”, Sams.

PRESSMAN Roger, 2002, Ingeniería de Software, “Ingeniería del software. Un enfoque práctico”, 5ta. Edición, Editorial Mc. Graw Hill.

RAMÍREZ Felipe, 2006, Programación, “Aprenda practicando Visual Basic 2005 usando Visual Studio 2005”, Prentice Hall.

SAMPIERI Roberto, FERNÁNDEZ Carlos, BAPTISTA Pilar, 1991, Metodología, “Metodología de la Investigación”, Mc Graw Hill.

SONMERVILLE Ian, 2005, Ingeniería de Software, “Ingeniería del software”, Séptima Edición, Editorial Pearson Addison Wesley.

TOUITOU David, RESNICK Marty, 2007, Programación, “Zimbra: Implement, Administer and Manage”, Packt Publishing.

TRAMULLAS Jesús, 2006, Programación, “Software Libre para servicios de información digital”, Prentice Hall.

WEISS Mark, 2000, Programación, “Estructuras de datos con Java”, Prentice Hall.

WELLING Luke, THOMSON Laura, 2008, Programación, “PHP and MySQL Web Development”, 4ta. Edición, Addison-Wesley Professional.

REFERENCIAS VIRTUALES

Universidad Sagrado Corazón de Puerto Rico, Programación, “Manual de Usuario de Zimbra”, [fecha de consulta: Enero del 2010]. Disponible en: http://www.sagrado.edu/noticias/zimbra/Manual_Usuario_Zimbra.pdf.

Potix Corporation, Programación, “ZK The Developer's Guide”, [fecha de consulta: Octubre del 2009]. Disponible en: <http://www.zkoss.org>.

Comunidad Joomla! Spanish, Programación, “Manual de Usuario Online de Joomla!”, [fecha de consulta: Octubre del 2009]. Disponible en: <http://www.joomlaspanish.org>.

Mora Verónica, Programación, “MANUAL ZIMBRA Colaboration Suite CENTOS 5,3”, [fecha de consulta: Diciembre del 2009]. Disponible en: <http://ver18.files.wordpress.com/2009/07/manual-zimbra.pdf>.

Web Oficial de Jasig, Wiki de Central Authentication Service CAS, [fecha de consulta: Enero del 2010]. Disponible en: <http://www.jasig.org/wiki/display/CAS/Home>.

Web Oficial de OpenBiblio, [fecha de consulta: Febrero del 2010]. Disponible en: <http://obiblio.sourceforge.net/index.php/My/Documentation>.

Web Oficial de EspaBiblio, [fecha de consulta: Febrero del 2010]. . Disponible en: <http://www.desem.cl/espabiblio/descargar.php>.

Wikipedia, [fecha de consulta: Octubre del 2009 hasta Febrero del 2010]. Disponible en: <http://es.wikipedia.org>.

ANEXOS

ANEXO # 1



UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA FACULTAD DE CIENCIAS ADMINISTRATIVAS ESCUELA DE INFORMÁTICA

ENCUESTA AL PERSONAL TÉCNICO ENCARGADO DE LA ADMINISTRACIÓN Y ACTUALIZACIÓN DE LA PÁGINA WEB DE LA UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA

Objetivo:

Conocer la existencia de restricciones impuestas por el fabricante de las herramientas de administración de la página web relacionadas a su utilización, así como la complejidad en su manejo. Conocer qué tan completas son estas herramientas en tareas que complementan el funcionamiento de la página web.

ENCUESTA

CON RESPECTO A LAS HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO Y MANTENIMIENTO DEL SITIO WEB:

1. ¿Alguna requirió algún pago para su uso?

SI () NO ()

2. ¿Por lo menos una limita el número de computadores en donde se la puede utilizar?

SI () NO ()

3. ¿Existe en alguna de ellas un límite máximo de usuarios?

SI () NO ()

4. ¿Ha apreciado alguna otra restricción en el uso de estas?

SI () NO ()

5. ¿Está disponible el código fuente de todas ellas?

SI () NO ()

En caso afirmativo y si usted realiza alguna modificación, ¿existe alguna restricción que le impida distribuir estos cambios?

SI () NO ()

6. ¿En caso de que sea necesario copiarlas, se requiere alguna autorización previa?

SI () NO ()

CON RESPECTO AL SITIO WEB:

7. ¿Cuenta con un editor integrado similar a un procesador de textos que facilite la publicación de contenido en el mismo?

SI () NO ()

8. ¿Existe la posibilidad de clasificar de alguna forma el contenido ingresado?

SI () NO ()

9. ¿Se puede modificar fácilmente el diseño y distribución del sitio web?

SI () NO ()

10. ¿Existe la posibilidad de que usuarios no especializados aporten directamente con contenido al sitio web?

SI () NO ()

11. ¿Permite administrar usuarios y grupos de usuarios?

SI () NO ()

12. ¿Existe algún historial que permita revisar el acceso de los usuarios a la página?

SI () NO ()

13. ¿Impide al acceso no autorizado a información reservada para usuarios específicos?

SI () NO ()

ANEXO # 2



UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA FACULTAD DE CIENCIAS ADMINISTRATIVAS ESCUELA DE INFORMÁTICA

ENCUESTA A LOS USUARIOS DE LA PÁGINA WEB

Objetivo:

Conocer la realidad de la página web de la Universidad Estatal Península de Santa Elena, en cuanto a las funcionalidades que brinda a la comunidad estudiantil, docente y público en general.

ENCUESTA

1. ¿Es usted estudiante, docente o parte del personal administrativo?

ESTUDIANTE ()

DOCENTE ()

PERSONAL ADMINISTRATIVO ()

2. ¿Ha visitado el portal de la UPSE?

SI () NO ()

3. ¿El portal cuenta con una utilidad para buscar material bibliográfico?

SI () NO ()

4. ¿La utilidad permite reservar los materiales?

SI () NO ()

5. ¿El portal cuenta con una consulta en línea del record académico?

SI () NO ()

6. ¿El portal permite la creación de cuentas de correo?

SI () NO ()

ANEXO # 3



MANUAL TÉCNICO DEL PORTAL WEB DE LA UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA

The screenshot displays the website interface for Universidad Estatal Península de Santa Elena (UPSE). At the top, there is a navigation bar with the university's logo on the left and social media links for Facebook, Twitter, and YouTube on the right. Below the navigation bar, the main content area is divided into several sections:

- El Rector:** A section featuring a portrait of Ing. Jimmy Candell Soto.
- Servicios en Línea:** A section with links for Sistema Académico, Galería de Imágenes, Descargas, and Comentarios.
- Usuarios Registrados:** A section with a login form for "Nombre de usuario" and a password field.
- Contador de visitas:** A section showing a visit counter with the number "000047" and a table of daily visits:

Hoy	1
Ayer	0
Esta Semana	1
- Facultades y Escuelas:** A grid of logos for various faculties and schools, including UNESIP, CONEA, SENPLADES, SENACYT, and IECE.
- Últimas noticias:** A section with two news items:
 - Oferta Académica:** Dated "Jueves, 07 Enero 2010 18:16", it mentions the university's offer of academic careers for 2010-2011.
 - Taller para consensuar Plan Operativo Anual:** Dated "Domingo, 22 Noviembre 2009 02:04", it reports on a workshop held to discuss the annual operational plan.
- Artículos más visitados:** A list of popular articles, including "Oferta Académica", "Nuevo Portal Universitario", "Pasantías Estudiantiles", "Reseña Histórica de la Creación de la Universidad", "UPSE imputa informe del CONEA", "Ingeniería Comercial", "Tercer Simposio sobre tortugas marinas", "Taller para consensuar Plan Operativo Anual", and "El Rector - Hoja de Vida".

INTRODUCCIÓN

El Portal Web desarrollado para la Universidad Estatal Península de Santa Elena está construido sobre una plataforma que integra una diversidad de herramientas de software libre, que le otorgan funcionalidades útiles para la interacción con los estudiantes.

En la presente sección se revisará todo lo relacionado con la interface del portal web (frontend) a fin de conocer las herramientas utilizadas para su construcción así como los complementos utilizados con el fin de enriquecer la experiencia del usuario al interactuar con el portal.

EL SISTEMA GESTOR DE CONTENIDOS

Una de las herramientas más importantes en la apariencia visual del portal, es el Sistema Gestor de Contenidos Joomla!.

Gracias a esta herramienta fue posible concebir un diseño personalizado que permita una fácil actualización, dinamismo y posibilidades de expansión a futuro.

LA PLANTILLA DEL PORTAL WEB

Joomla! es un sistema que genera interfaces web en base a la información registrada por el administrador del sitio. La distribución de esta información obedece a un esquema definido en un elemento denominado plantilla.

Una plantilla también define las características del sitio tales como colores, tipos de letra, espaciado entre los elementos, imágenes de fondo, y todo aquello que pueda ser personalizado mediante hojas de estilo css. Por lo general, una plantilla

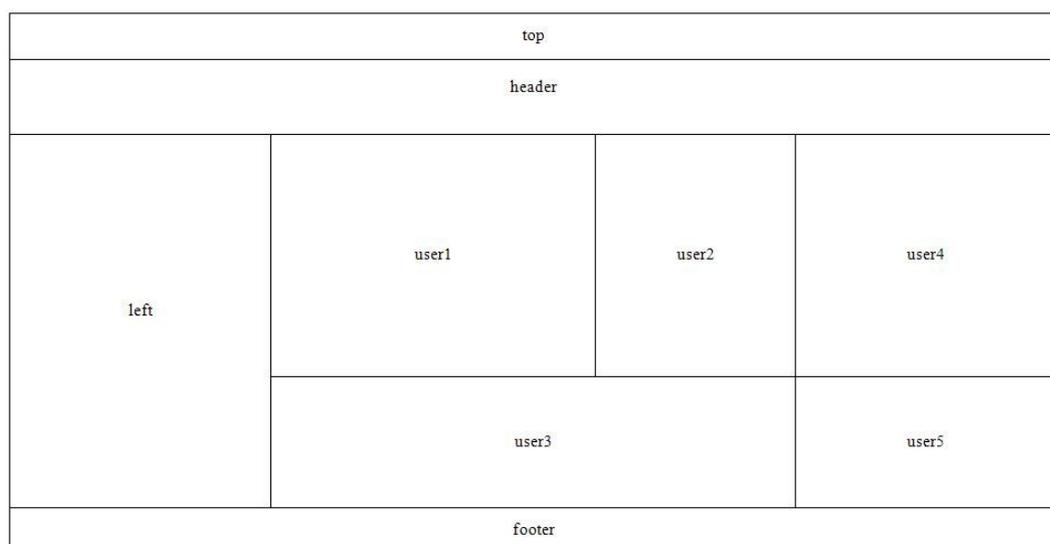
define dos distribuciones: una para la página principal y otra para mostrar el contenido de los artículos o cualquier otro recurso.

En internet existe una innumerable cantidad de plantillas disponibles, tanto gratuitas como comerciales. Pero si lo que se busca es cumplir necesidades específicas en cuanto al diseño del portal, la mejor opción es desarrollar una plantilla propia.

Para el portal universitario se creó una plantilla llamada Upse Azul, nombre asignado debido a que el color predominante es una variación del azul. Esta plantilla define algunas secciones que permiten al administrador del sitio combinar el contenido informativo con recursos que mejoren la funcionalidad global del portal.

Existe gran cantidad de recursos que se pueden acoplar a Joomla!, tales como contadores de visitas, encuestas, galerías de imágenes, gestores de comentarios, entre muchos otros.

GRÁFICO # 13
Distribución de la página principal.



Fuente: Autores

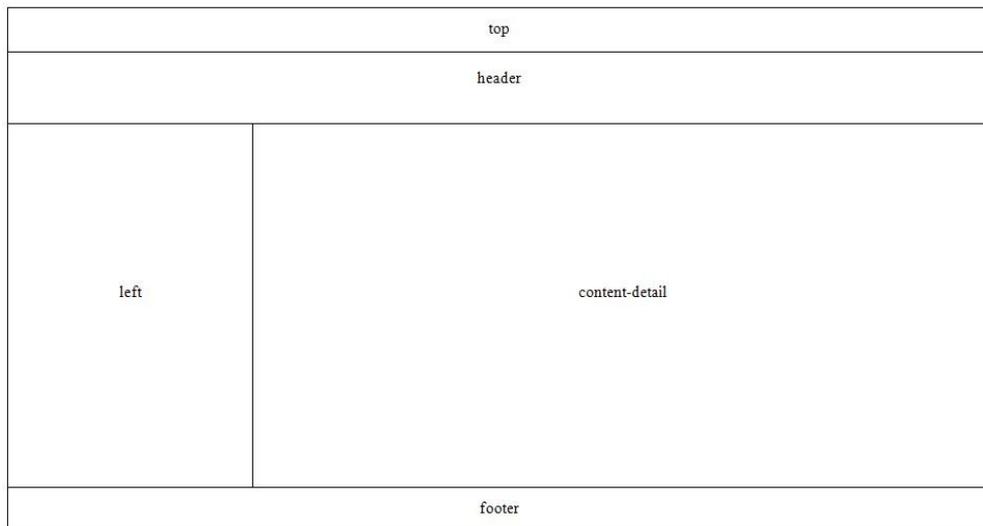
En el gráfico anterior apreciamos la distribución definida por la plantilla para la página principal, en ella se encuentran los siguientes componentes:

- **top:** Esta sección tiene unas dimensiones de 970 pixeles de ancho y 75 pixeles de alto y fue creada para insertar recursos que no necesitan gran cantidad de espacio. Por ejemplo: un administrador de enlaces a redes sociales o un cuadro de búsqueda de contenido.
- **header:** Con unas dimensiones de 970 pixeles de ancho y 55 pixeles de alto, esta sección fue definida para alojar el menú principal del sitio.
- **left:** En esta sección se pueden incluir recursos que estarán disponibles en todo el portal web, tales como el formulario de autenticación de usuarios, el contador de visitas o el menú de servicios en línea. Tiene unas dimensiones de 160 pixeles de ancho y el largo es variable de acuerdo a lo que ocupen los recursos ubicados en esta sección.
- **user1:** Esta sección tiene las dimensiones de 400 pixeles de ancho por 290 pixeles de alto y fue definida específicamente para mostrar las últimas noticias universitarias acompañadas de una fotografía.
- **user2:** Posee unas dimensiones de 210 pixeles de ancho por 290 pixeles de alto. Esta sección fue creada para mostrar los últimos videos publicados en el canal YouTube de la Universidad Estatal Península de Santa Elena.
- **user3:** Esta es la sección en donde se colocarán las últimas noticias de la institución acompañadas de un fragmente de las mismas. Tiene unas dimensiones de 610 pixeles de ancho y el largo tiene un mínimo de 480 pixeles pudiéndose expandir en caso de ser necesario.
- **user4:** Con unas dimensiones de 200 pixeles de ancho por 290 pixeles de alto, esta es la sección escogida para mostrar los enlaces de interés representados mediante un pequeño logo del sitio.
- **user5:** Esta sección está reservada a mostrar un listado de las noticias más leídas por los visitantes del portal. Tiene unas dimensiones de 190 pixeles de ancho por con una altura mínima de 480 pixeles, pudiendo expandirse en caso de ser requerido.

- **footer:** Reservado para incluir la información del pie de página, que por lo general incluye datos de la institución. Posee unas dimensiones de 970 pixeles de ancho por 30 pixeles de alto.

En la distribución definida para mostrar contenido se repiten las secciones top, header, left y footer, tal y como se observa a continuación:

GRÁFICO # 14
Distribución para cuando se muestra contenido.



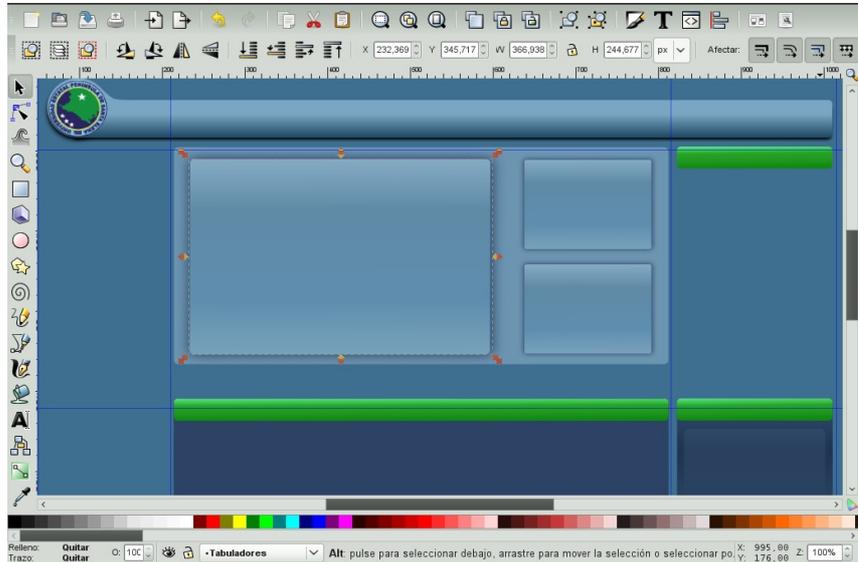
Fuente: Autores

- **content-detail:** Esta sección se reserva para el contenido ya sea del artículo o del recurso solicitado. Tiene unas dimensiones de 790 pixeles de ancho y la altura se ajusta al contenido mostrado.

DISEÑO DE LAS IMÁGENES DE LA PLANTILLA

Los recursos gráficos incluidos en la plantilla fueron realizados utilizando las herramientas Inkscape y GIMP. Inkscape permite la creación de gráficos vectoriales, por lo general, en formato SVG. GIMP, en cambio, permite el tratamiento de imágenes y de diseño gráfico.

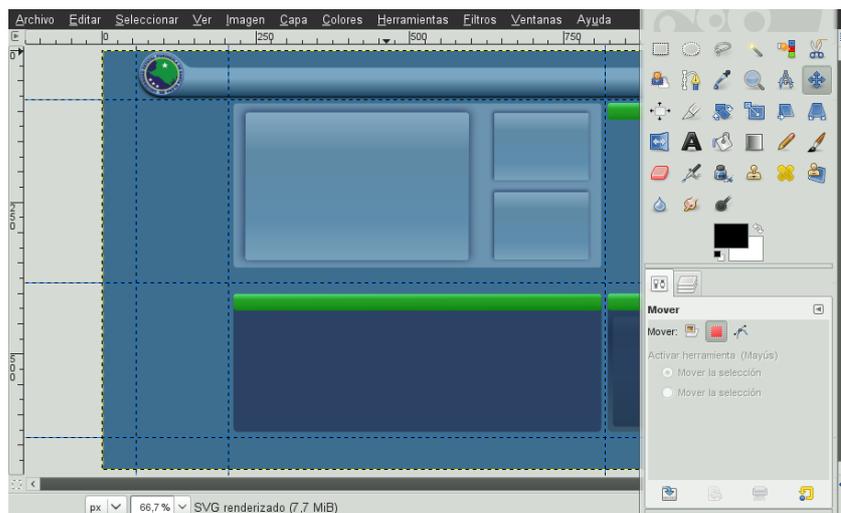
GRÁFICO # 15 Trabajando las imágenes en Inkscape.



Fuente: Autores

El procedimiento para la creación de imágenes consiste en definir los gráficos mediante Inkscape, y una vez listas, éstas se pasan a GIMP para fragmentarlas y así poder incluirlas en las secciones correspondientes.

GRÁFICO # 16 Trabajando con GIMP.



Fuente: Autores

APLICANDO LA PLANTILLA AL PORTAL WEB

Joomla! permite aplicar una nueva plantilla a un sitio web sin importar si este tiene o no información publicada. Todo el contenido se acopla según la distribución establecida en la plantilla. Antes de poder aplicar la plantilla, es necesario, que ésta sea instalada en el sitio web.

Para asignar la plantilla creada utilizamos la herramienta Gestor de Plantillas ubicada en el menú extensiones que se encuentra en la vista de administración del sitio web, luego seleccionamos la plantilla y damos click en el botón Predeterminado.

GRÁFICO # 17
Selección de la plantilla.



Joomla! es software libre liberado bajo la Licencia GNU/GPL.
Pack creado por Joomla! Spanish 2009 - Patrocinado por Web Empresa

Fuente: Autores

CONFIGURANDO LOS RECURSOS INCLUIDOS EN EL PORTAL

Con la plantilla aplicada, resta cargar todos los recursos que van a complementar el contenido del Portal. El proceso de búsqueda de los diferentes componentes, módulos y plugins, se centró en el directorio de extensiones Joomla!

(<http://extensions.joomla.org/>), de donde se seleccionó aquellos que tenían mejores características y que, por supuesto, no fueran de naturaleza comercial.

Cada vez que se encontraba un recurso de interés, éste era descargado, instalado e incluido en una de las secciones definidas por la plantilla. Con esto, se evaluaba la manera cómo el recurso se acoplaba al Portal, y en caso de presentar algún problema, éste se desechaba y se buscaba otro. En dependencia del recurso obtenido, se utilizaba su interfaz de administración (para los componentes), el gestor de módulos o el gestor de plugins a fin de darle los valores necesarios.

GRÁFICO # 18
Ubicación de los módulos en las secciones de la plantilla.

Joomla! es software libre liberado bajo la Licencia GNU/GPL.
Pack creado por Joomla! Spanish 2009 - Patrocinado por Web Empresa

Fuente: Autores

DISTRIBUCIÓN DE LOS RECURSOS EN EL PORTAL

La versión inicial del portal web universitario, contaba con la siguiente distribución de recursos:

Sección top

- **Módulo Stalker:** Este módulo permite administrar una lista de redes sociales insertando los íconos de aquellas que hayan sido configuradas en el sitio.

- **Módulo RokAjaxSearch:** Inserta un cuadro de texto que permite realizar búsquedas dentro de los artículos publicados. Trabaja con Ajax a fin de realizar búsquedas dinámicas.

Sección header

- **Módulo RokNavMenu:** Genera el menú principal del Portal Web.

Sección left

- **Módulo Simple Image Holder:** Muestra una imagen permitiendo colocar un título. Este módulo es utilizado para mostrar la foto del Rector y colocar un enlace a su hoja de vida.
- **Módulo YOOLogin:** Reemplaza al formulario de acceso que viene por defecto.
- **Módulo Vinaora Visitors Counter:** Muestra un resumen de las visitas registras por día, semana, mes y el total.

Sección user1

- **Módulo Sliderku:** Permite definir enlaces a noticias a modo de imágenes con un pie informativo donde se define el título de la noticia y un breve resumen.

Sección user2

- **Módulo The Joomla Shop YouTube:** Permite mostrar los enlaces a videos de un determinado canal de YouTube.

Sección user3

- **Módulo Thumbsup:** Muestra los últimos artículos publicados.

Sección user4

- **Módulo JI Icon Menú:** Permite definir una lista de enlaces con íconos.

Sección user5

- **Módulo Mostread:** Lista los artículos más visitados del Portal.

Sección footer

- **Módulo Custom Footer:** Permite definir un pie de página personalizado con datos de la institución.

PRESENTACIÓN DEL PORTAL WEB UNIVERSITARIO

GRÁFICO # 19
Apariencia inicial del Portal Web Universitario.

The screenshot shows the homepage of the Universidad Estatal Península de Santa Elena (UPSE) website. The layout is organized into several sections:

- Header:** Includes the university logo, navigation menu (Inicio, Nosotros, Estructura Académica, Estudiantes, Profesores, Administrativo, Ley de transparencia), social media links (Facebook, Twitter, YouTube), and a search bar.
- Left Sidebar:**
 - El Rector:** Profile of Ing. Jimmy Candell Soto.
 - Servicios en Línea:** Links to Sistema Académico, Galería de Imágenes, Descargas, and Comentarios.
 - Usuarios Registrados:** A login form with fields for "Nombre de usuario" and a password field.
 - Contador de visitas:** Shows a total of 000047 visits, with a breakdown: Hoy (1), Ayer (0), and Esta Semana (1).
- Main Content Area:**
 - Entró en funcionamiento:** A large banner image showing a traffic light system with the text "Sistema de Semaforización frente a la UPSE".
 - Facultades y Escuelas:** A grid of logos for various faculties and schools, including UPSE, CONEA, SENPLADES, BENADYT, and IECE.
 - Últimas noticias:** A section titled "Oferta Académica" with two news items:
 - Oferta Académica:** "La Universidad Estatal Península de Santa Elena se complace en poner a consideración de la ciudadanía la Oferta Académica para el período 2010 - 2011. La gama de carreras ofertadas por nuestra institución incluye: ORGANIZACIÓN Y DESARROLLO COMUNITARIO PSICOLOGÍA INFORMÁTICA ELECTRÓNICA Más..." (dated Jueves, 07 Enero 2010 18:16).
 - Taller para consensuar Plan Operativo Anual:** "Universidad Estatal Península de Santa Elena realiza taller para consensuar el Plan Operativo Anual. Autoridades, Funcionarios y responsables de las unidades académicas, participaron los días 13 y 14 del presente mes del taller de capacitación para la elaboración del Plan Operativo Anual de nuestro centro de Más..." (dated Domingo, 22 Noviembre 2009 02:04).
- Right Sidebar:**
 - Artículos más visitados:** A list of popular articles:
 - Oferta Académica
 - Nuevo Portal Universitario
 - Pasantías Estudiantiles
 - Reseña Histórica de la Creación de la Universidad
 - UPSE imputa informe del CONEA
 - Ingeniería Comercial
 - Tercer Simposio sobre tortugas marinas
 - Taller para consensuar Plan Operativo Anual
 - El Rector - Hoja de Vida

Fuente: <http://www.upse.edu.ec>

ANEXO # 4



MANUAL DE DESARROLLO EN ZK

INTRODUCCIÓN

ZK es un framework de aplicaciones web soportado completamente en Java, está desarrollado bajo el esquema de código abierto, y permite desarrollar aplicaciones con soporte AJAX generando ricas interfaces de usuario para aplicaciones web sin usar explícitamente JavaScript y con poca programación.

GRÁFICO # 20 Logo de ZK.



Fuente: <http://www.zkoss.org>

El núcleo de ZK es un mecanismo conducido por eventos basado en AJAX, sustentado sobre 70 componentes XUL y 80 componentes XHTML (extensiones al lenguaje HTML que permiten definir elementos más elaborados), y un lenguaje de marcación para diseñar interfaces de usuario.

Los programadores diseñan las páginas de su aplicación en componentes XUL/XHTML ricos en características, y los manipulan con eventos disparados por la actividad del usuario final. Es similar al modelo de programación encontrado en las aplicaciones basadas en GUI de escritorio. Además de la programación basada en componentes y orientación a eventos ZK soporta un lenguaje de marcación para la definición de una potente interfaz de usuario llamada ZUML.

- ZUML está diseñado para que desarrolladores no expertos diseñen interfaces de usuario de forma eficiente.
- ZUML permite mezclar diferentes lenguajes de marcación, tales como XUL de Mozilla y XHTML, todos en la misma página.
- ZUML permite a los desarrolladores embeber scripts en lenguaje Java (interpretado por BeanShell) y usar expresiones EL para manipular los componentes y acceder a los datos.

REQUISITOS DE HABILIDADES EN PROGRAMACIÓN

Imprescindible

- Conocimiento Básico de Java.
- Conocimiento Básico de HTML y XUL.

Recomendado

- Programación Orientada a Objetos.
- Programación con Servlet/JSP.
- Ajax.
- JavaScript.

CONFIGURACIÓN DEL AMBIENTE DE DESARROLLO

El siguiente procedimiento se realizó en un equipo con OpenSUSE 11.2, en el entorno de escritorio GNOME.

Software a instalar

- **Java 1.6.0 openjdk-devel:** Kit de desarrollo Java basado en OpenJDK 6, que es la versión open source del kit de desarrollo Java.
- **Eclipse Galileo:** Es un ambiente de desarrollo integrado con herramientas orientadas a desarrolladores de aplicaciones Java de escritorio o web.

- **ZK Studio 0.9.6:** Herramienta de desarrollo para ZK que puede ser añadido como un plug-in a Eclipse dotándole de herramientas que facilitan la creación de aplicaciones ZK.
- **ZK Framework 5:** Conjunto de librerías que soportan el framework ZK.
- **Apache Tomcat 6.0:** Contenedor de servlets para aplicaciones web Java, que servirá para ejecutar las aplicaciones antes de que se publiquen.

Pre-requisitos

- Computador o máquina virtual con OpenSUSE 11.2 instalado.
- Conexión a internet para la descarga de los paquetes de instalación.

Procedimiento de instalación

- **Java 1.6.0 openjdk devel:** La instalación se realiza mediante la herramienta Instalar/Desinstalar software del Centro de Control YaST, aquí se hace una búsqueda para verificar si el software se encuentra instalado, y si no se procede a la instalación del mismo.

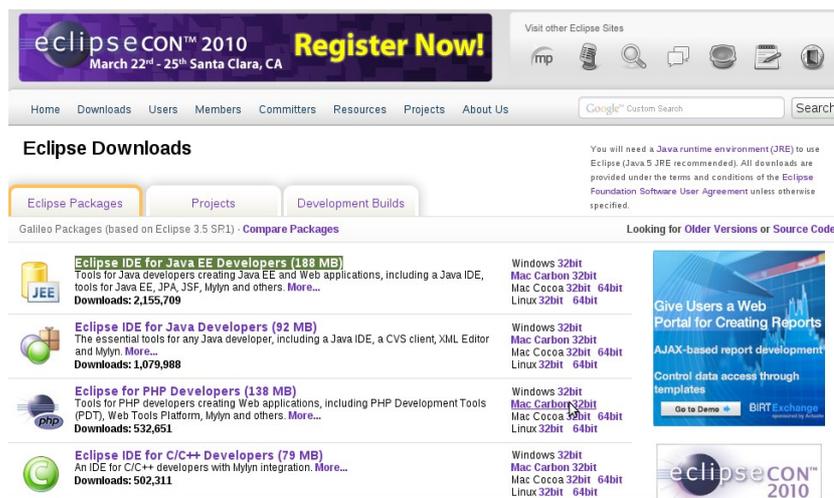
GRÁFICO # 21
Utilidad de instalación YaST.



Fuente: Autores

- **Eclipse Galileo:** Para utilizar la herramienta es necesario descargar desde el sitio web (<http://www.eclipse.org/downloads/>) el paquete del IDE para Desarrolladores de Java EE, este se descomprime en una carpeta y se utiliza sin necesidad de realizar algún procedimiento de instalación.

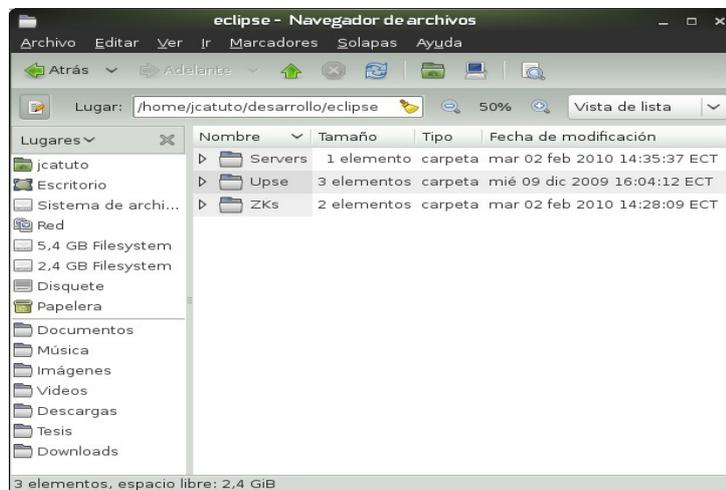
GRÁFICO # 22 Página de descarga de Eclipse.



Fuente: <http://www.eclipse.org/downloads/>

Para mantener un ambiente de desarrollo organizado, se recomienda definir una estructura de directorios en donde se almacenarán las aplicaciones organizadas en carpetas.

GRÁFICO # 23 Directorio de almacenamiento de proyectos.

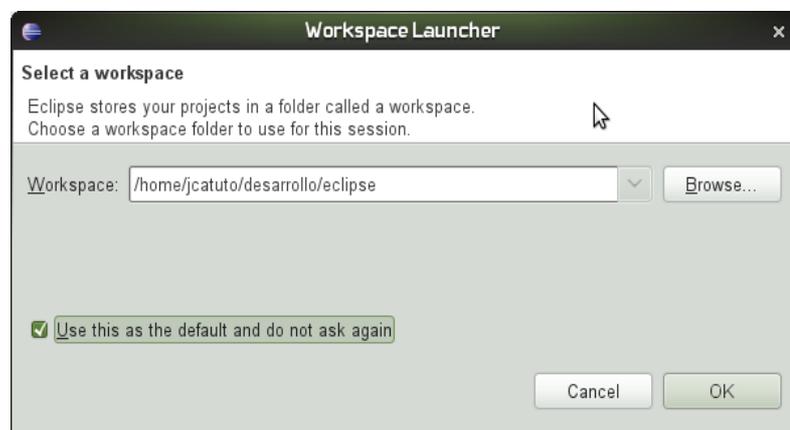


Fuente: Autores

Ejm: /<directorio_home>/desarrollo/eclipse/<proyecto>

Al momento de iniciar la herramienta se pide definir el espacio de trabajo (workspace) con el cual se trabajará, este espacio es un directorio en el cual la herramienta almacena los recursos necesarios para editar los proyectos. En este caso, seleccionaremos el previamente definido, /<directorio_home>/desarrollo/eclipse/. El directorio home es el directorio de inicio creado para cada usuario del sistema operativo.

GRÁFICO # 24 Definición del espacio de trabajo de Eclipse.



Fuente: Autores

- **ZK Studio 0.9.6:** Este componente se instala como un plug-in del IDE Eclipse: Es necesario descargar el paquete con las librerías necesarias desde el sitio <http://www.zkoss.org/download/zkstudio.dsp>. Una vez descargado el paquete, se descomprime y se copian los archivos .jar que se encuentran dentro del directorio plugins al directorio plugins de eclipse, así como los archivos que se encuentren en la carpeta features a la carpeta /features de eclipse.

La próxima vez que se inicie Eclipse se pedirá la activación de ZK Studio y luego se podrá utilizar sin ninguna limitación. Para la activación es necesario que el equipo tenga acceso a internet y que previamente se haya realizado un registro en el sitio del framework.

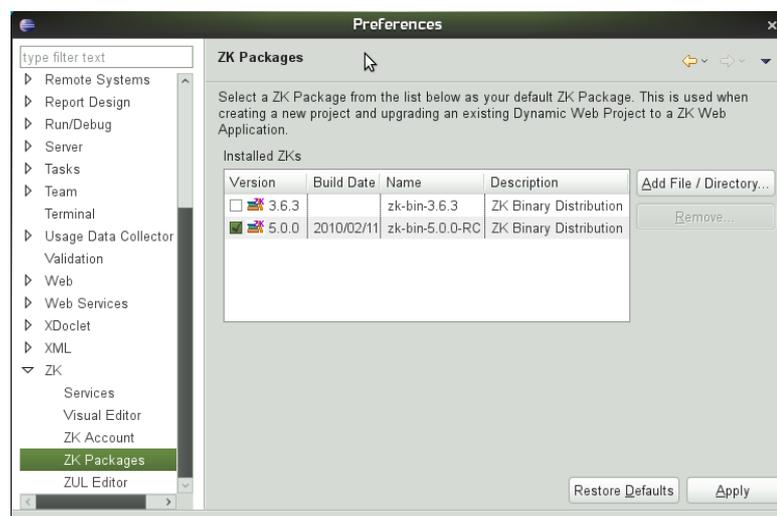
GRÁFICO # 25 Activación de ZK Studio.



Fuente: Autores

- **ZK Framework 5:** Para instalar el framework es necesario descargar desde el sitio <http://www.zkoss.org/download/zk.dsp> el paquete con las librerías necesarias y luego descomprimirlas. Para indicarle a Eclipse que versión del framework se desea utilizar, se selecciona en el menú Window la opción Preferences y en el diálogo se busca la opción ZK-> ZK Packages, luego se añade el directorio donde se descomprimió el paquete descargado, con esto se importan las librerías del framework, finalmente se selecciona la versión 5.

GRÁFICO # 26 Selección de la versión del framework.



Fuente: Autores

- **Apache Tomcat 6.0:** Para usarlo en el ambiente de desarrollo, se descarga la distribución binaria desde el sitio <http://tomcat.apache.org/download-60.cgi> y se descomprime, de preferencia, en una carpeta definida para el almacenamiento de los programas.

GRÁFICO # 27 Agregando un servidor a Eclipse.



Fuente: Autores

En Eclipse se debe definir un servidor con este componente, para lo cual añadimos en la ficha Servers un nuevo registro con la configuración de un servidor Apache Tomcat v6.0, asignándole al directorio de instalación la ubicación donde se descomprimió el paquete. Si no se ubica la ficha Servers, puede ubicarla a través de Windows -> Preferences -> Servers.

En este momento ya tenemos el ambiente de desarrollo listo para poder crear aplicaciones ZK o dar mantenimiento a una aplicación existente. En el caso de la aplicación de consulta de calificaciones, se utilizó también la herramienta Apache Axis, la cual es una implementación open source de SOAP, y que sirve para proporcionar un entorno de ejecución de servicios Web a aplicaciones java. Las librerías necesarias para el uso de esta herramienta ya se encuentran en la aplicación dentro de la carpeta WEB-INF/lib, así que no es necesario realizar un procedimiento de instalación.

ESTRUCTURA DE UNA APLICACIÓN WEB BASADA EN ZK

Como ya se mencionó, ZK está basado totalmente en Java, por tanto la estructura del proyecto sigue las especificaciones de una aplicación de este tipo. Una aplicación web realizada con ZK consiste en un conjunto de páginas .zul, ficheros html, clases Java de apoyo y otro tipo de recursos (ficheros de imágenes, sonido, etc.). Este tipo de aplicaciones web puede existir de dos modos, dependiendo de la etapa en la que se encuentre el proyecto:

- **Etapa de desarrollo:** Se define una estructura de directorios basada en la especificación establecida para los Servlets, en donde deben ubicarse los componentes de la aplicación. En este esquema se realizan pruebas y modificaciones constantes en los componentes del proyecto.
- **Etapa de producción:** El contenido del directorio es empaquetado en un fichero de extensión war (Web Application Resource, a veces también se le suele llamar Web ARchive) que engloba a todo su contenido. Este empaquetamiento se produce en la etapa de producción, es decir, cuando la aplicación ha sido comprobada y depurada para su publicación.

Con estos modos de existencia se persigue que una misma aplicación web pueda ser desplegada en diferentes servidores web manteniendo su funcionalidad y sin ninguna modificación de código. Al momento en que se cree con Eclipse un nuevo proyecto ZK, automáticamente se genera la estructura de directorios requerida, siendo necesario empezar a crear y ubicar los componentes de la aplicación. Entre los directorios que juegan un papel de vital importancia en el proyecto, se encuentra el directorio WEB-INF, donde se almacena el archivo descriptor de la aplicación (web.xml) así como los archivos auxiliares a este. Todos los recursos ubicados aquí son de índole privada y no podrán ser accedidos directamente desde el cliente. El descriptor de la aplicación es un archivo en formato XML que detalla todos los recursos que el servlet va a necesitar y el comportamiento de los mismos.

GRÁFICO # 28

Estructura de aplicación según especificaciones API Servlet.



Fuente: Aumaille (2002) “J2EE Desarrollo de Aplicaciones Web”

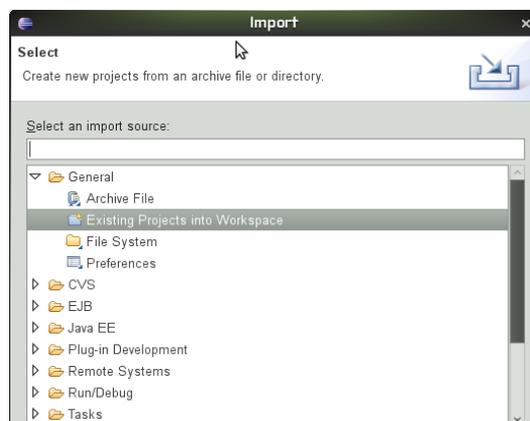
Dentro del directorio WEB-INF, también es generado un directorio con el nombre lib, que sirve para almacenar las librerías de clase en forma de ficheros jar que la aplicación web utiliza para referenciar a las clases provistas por terceros (ejm: drivers JDBC, librerías de etiquetas, etc.). Una vez que el proyecto está terminado, este es empaquetado, y a partir del momento en que se ubique el archivo war en el directorio del servidor definido para la publicación de aplicaciones, y sin hacer nada más, la aplicación web estará disponible para todo el mundo.

CONFIGURACIÓN DE LA APLICACIÓN DE CONSULTA DE CALIFICACIONES

Una vez establecido el ambiente de desarrollo, y después de ubicar la carpeta de la aplicación de consulta de calificaciones en el directorio definido para almacenar los proyectos, se debe importar el proyecto al ambiente de trabajo.

GRÁFICO # 29

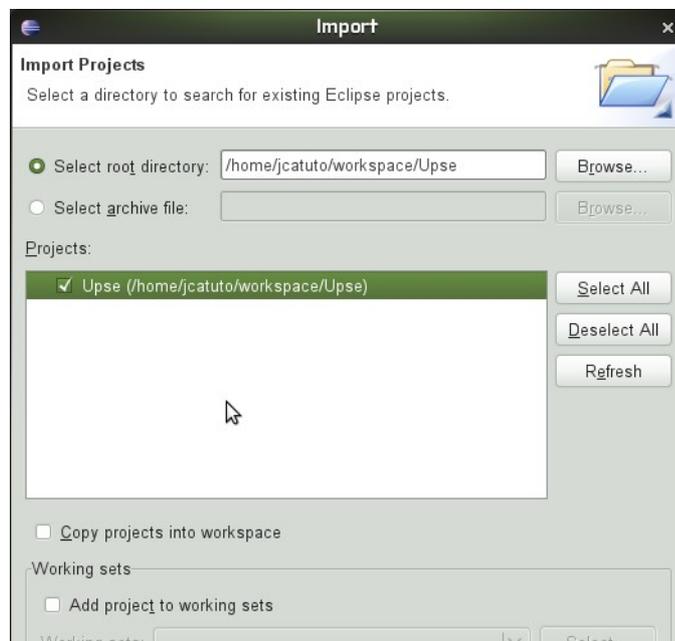
Seleccionando el tipo de proyecto a importar.



Fuente: Autores

Para el efecto abrimos Eclipse y seleccionamos la opción File -> Import, en el diálogo resultante aparece un árbol que permite seleccionar el tipo de proyecto a importar, en este caso importamos un proyecto que ya se encuentra ubicado en el workspace seleccionando la categoría General en la opción Existing Projects into Workspace, luego seleccionamos el directorio raíz de la aplicación, y si ese directorio contiene un proyecto válido, se permitirá seleccionarlo, luego de lo cual se finalizara el proceso de importación.

GRÁFICO # 30
Importando un proyecto de ZK.

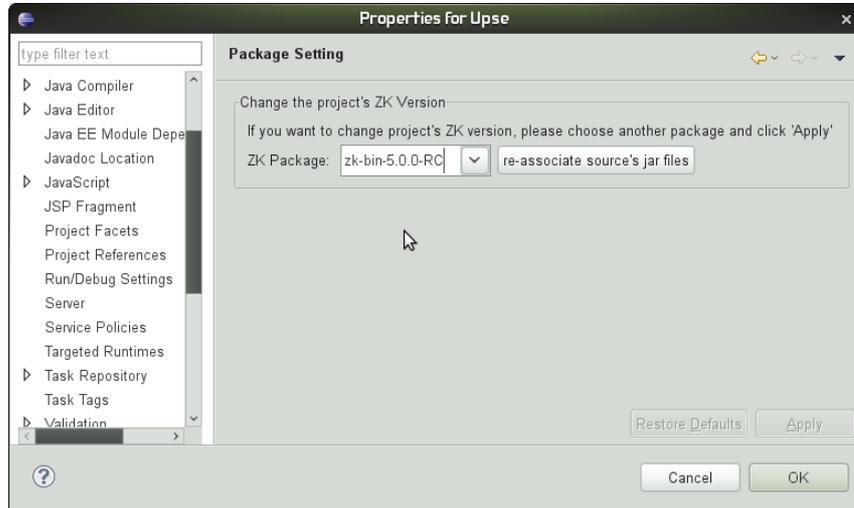


Fuente: Autores

Si el ambiente de trabajo fue correctamente montado, bastará con ejecutar el proyecto con el servidor Tomcat añadido al ambiente, dándole click derecho a uno de los documentos zul del proyecto y seleccionando la opción Run -> Server, para verificar la funcionalidad de la aplicación.

Si a futuro se desea realizar una actualización del framework, se deberá seguir el procedimiento establecido para su instalación, y adicionalmente debe seleccionarse dentro de las propiedades del proyecto la versión del framework con la que se va a trabajar.

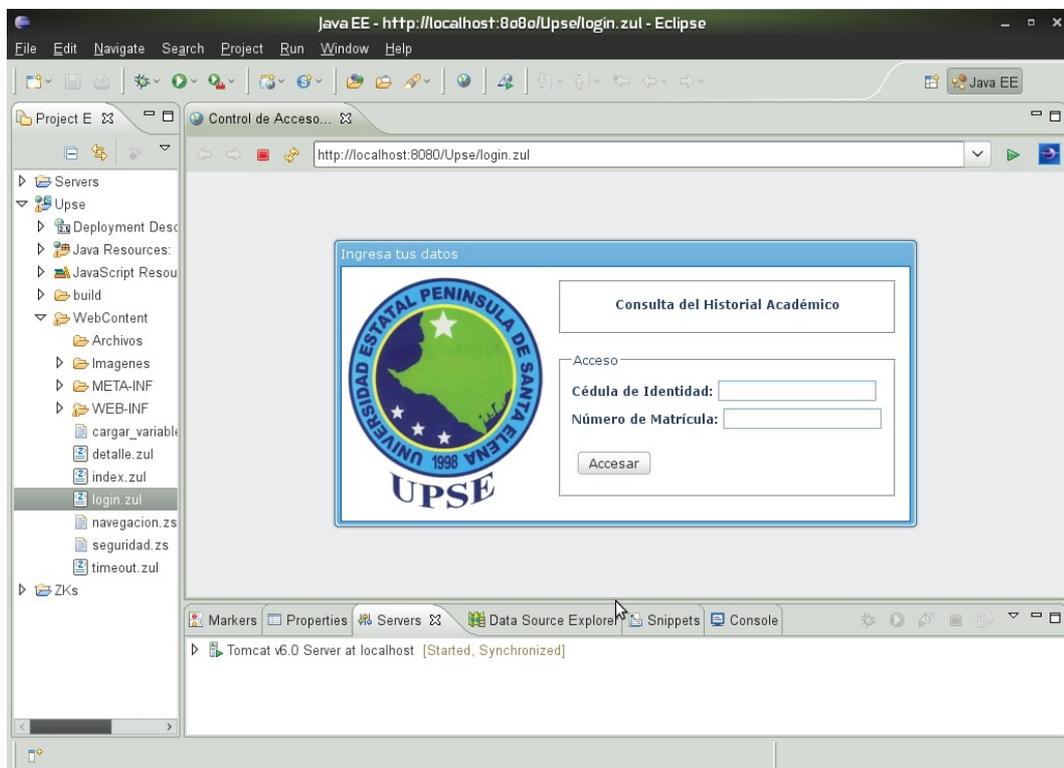
GRÁFICO # 31
Cambiando la versión del framework utilizada en el proyecto.



Fuente: Autores

CONFIGURACIÓN DEL CONTENEDOR DE SERVLETS

GRÁFICO # 32
Ambiente de Desarrollo ZK con una página en ejecución.



Fuente: Autores

Cuando se configuró el ambiente de desarrollo para ZK, se realizó la definición de un servidor para efectos de probar la funcionalidad de la aplicación a medida que esta iba desarrollándose, pero para cuando se necesite publicar nuestra aplicación en internet, se deberá realizar el procedimiento de instalación de Apache Tomcat como un servicio en el servidor de aplicaciones a fin de poder poner la aplicación a disposición de cualquiera que desee utilizarla.

El siguiente procedimiento indica los pasos necesarios para configurar el Contenedor de Servlets Apache Tomcat a fin de poder publicar la aplicación. El procedimiento se realizó en un equipo ejecutando CentOS 5.4 en modo texto.

Software a instalar

- **Java-1.6.0-openjdk.i386:** Kit de desarrollo Java basado en OpenJDK 6, que es la versión open source del kit de desarrollo Java.
- **Apache Tomcat 6.0:** Contenedor de servlets para aplicaciones web Java, que permitirá el acceso a esta aplicación web.

Pre-requisitos

- Computador o máquina virtual con CentOS 5.4 instalado.
- Conexión a internet para la descarga de los paquetes de instalación.

Procedimiento de instalación

Lo primero que se debe instalar es el entorno de ejecución java, el cual servirá de base para tomcat6. Para realizar esta instalación puede que exista la necesidad de agregar un nuevo repositorio de paquetes, esto se logra con los siguientes pasos:

Entramos a la ruta `/etc/yum.repos.d`

```
# cd /etc/yum.repos.d/
```

Descargamos el siguiente fichero.

```
# wget http://www.jpackage.org/jpackage50.repo
```

Actualizamos la lista de paquetes

```
# yum update
```

Con esto el repositorio está instalado y sólo resta añadir los paquetes de software de java con la siguiente instrucción:

```
# yum -y install java-1.6.0-openjdk java-1.6.0-openjdk-devel
```

Una vez que tenemos el soporte de java procedemos con la instalación de Tomcat, ejecutando el siguiente comando:

```
# yum -y install tomcat6 tomcat6-webapps tomcat6-admin-webapps
```

Iniciamos Tomcat6 de la siguiente manera:

```
# /etc/init.d/tomcat6 start
```

Finalmente verificamos que tomcat6 este iniciado accediendo a la siguiente url:

```
http://localhost:8080.
```

PUBLICACIÓN DE LA APLICACIÓN

De acuerdo a la norma establecida para las aplicaciones web basadas en Java EE, una vez que tenemos la aplicación lista para su uso, esta debe ser empaquetada en un solo fichero con extensión .war. Este procedimiento es realizado con el IDE Eclipse utilizando la opción Export dentro del menú File, en la ventana resultante se expande el nodo Web, y se selecciona la opción War File. Finalmente se selecciona la ubicación donde se guardará el archivo.

Con este archivo, resta publicarlo en el servidor web con los siguientes pasos:

GRÁFICO # 33 Generando el archivo empaquetado.



Fuente: Autores

Se detiene el servicio tomcat en el servidor:

```
#service tomcat6 stop
```

Si la aplicación ha sido previamente publicada se debe eliminar esa versión, quitando tanto el archivo empaquetado como el directorio que en su momento tomcat expandió para poner a disposición la aplicación a los usuarios de internet:

```
#rm -rf <Nombre de la aplicación>/
```

```
#rm -f <Nombre de la aplicación>.war
```

En donde:

Nombre de la aplicación, nombre del proyecto con el que se está trabajando.

Se coloca el nuevo archivo empaquetado en el directorio correspondiente:

```
#cp <Directorio de desarrollo>/<Nombre de la aplicación>.war  
/var/lib/tomcat6/webapps/
```

En donde:

Directorio de desarrollo, es la ubicación donde está el archivo empaquetado.

El comando previo sirve para una copia hecha en el mismo equipo. Si el archivo empaquetado esta en otro equipo se deberá utilizar el siguiente comando:

```
#scp -P <Número de puerto> <Directorio de desarrollo>/<Nombre de la aplicación>.war root@<Dirección del equipo>:/var/lib/tomcat6/webapps/
```

En donde:

Número de puerto, es el número de puerto utilizado para el acceso remoto al equipo, si se omite se utiliza el puerto por defecto.

Dirección del equipo, es la dirección IP o el nombre del equipo al cual deseamos acceder seguido del signo : y la ruta en donde se desea cargar el archivo.

El comando requiere la clave de administrador del equipo donde se hace la copia.

INTEGRACIÓN CON LA PLATAFORMA OPERATIVA

Una vez que se ha publicado la aplicación, esta es fácilmente accedida desde la url: <http://www.upse.edu.ec:8080/Upse/login.zul>.

GRÁFICO # 34 Acceso a la Aplicación de Consulta de Calificaciones.



The screenshot shows a web browser window titled "Control de Acceso...". Inside the browser, there is a login form titled "Ingresa tus datos". On the left side of the form is the logo of the Universidad Estatal Península de Santa Elena (UPSE), which features a globe and the text "UNIVERSIDAD ESTATAL PENINSULA DE SANTA ELENA 1998 UPSE". To the right of the logo, there is a box labeled "Consulta del Historial Académico". Below this box, there is a section labeled "Acceso" containing two input fields: "Cédula de Identidad:" and "Número de Matrícula:". Below these fields is a button labeled "Accesar".

Fuente: <http://www.upse.edu.ec:8080/Upse/login.zul>

Para integrarla con la plataforma del portal web, se incluye en el Sistema de Administración de Contenidos (Joomla!) un enlace que permita accederla.

GRÁFICO # 35
Integrando la aplicación a la plataforma operativa.



Fuente: <http://www.upse.edu.ec/administrator>

Para este fin accedemos al back-end del sitio, y procedemos a añadir una opción en el Menú de Servicios en Línea, especificamos que el tipo de contenido a desplegar es una URL embebida, y entre los atributos requeridos por el CMS establecemos la ruta de acceso a la aplicación.

GRÁFICO # 36
Asignando la aplicación al menú de Servicios en Línea.



Fuente: <http://www.upse.edu.ec/administrator>

GRÁFICO # 37 Accediendo a la aplicación desde el Portal.

The screenshot displays the UPSE VISION portal interface. On the left, a navigation menu lists various services such as 'Usuarios Registrados', 'Servicios en Línea', and 'Contador de visitas'. The main content area is titled 'Consulta de Calificaciones' and contains a login form for 'Consulta del Historial Académico'. The form includes input fields for 'Cédula de Identidad' and 'Número de Matricula', along with an 'Accesar' button. The UPSE logo is prominently displayed in the top right corner and within the login form's header.

Fuentes: <http://www.upse.edu.ec/index.php/calificaciones>

GUÍA RÁPIDA DE DESARROLLO CON ZK

La siguiente guía contiene una rápida descripción de los comandos más útiles de la herramienta de desarrollo de aplicaciones ZK, que servirán para el mantenimiento de la aplicación de Consulta de Calificaciones. Esta sección sólo describe una pequeña parte del potencial que la herramienta posee.

La primera aplicación con ZK: Hola Mundo!

Una vez listo el ambiente de desarrollo para ZK, la escritura de aplicaciones es bastante sencilla.

Basta con crear un archivo, por ejemplo hello.zul, en el directorio adecuado.

```
<window title="Hello" border="normal">
    Hello World!
</window>
```

Luego, probamos la aplicación ejecutándola en el servidor definido en el ambiente de desarrollo o escribiendo en un navegador la URL correcta, ejm: `http://localhost/myapp/hello.zul`, para cualquiera de los casos es necesario verificar que el servidor este iniciado.

GRÁFICO # 38 Aplicación Hola Mundo con ZK.



Fuente: Potix (2009) "ZK The Developer's Guide"

En una página de ZUML, un elemento XML describe qué componente crear. En este ejemplo, esta una ventana (`org.zkoss.zul.Window`). Los atributos XML se utilizan para asignar valores a las propiedades de los componentes de la ventana. En este ejemplo, se crea una ventana con un título, esto se hace estableciendo el título y las propiedades del borde con los valores "Hola" y "normal", respectivamente. El texto incluido en los elementos XML se interpreta también como un componente especial llamado label (`org.zkoss.zul.Label`). Así, el ejemplo anterior es equivalente a la siguiente.

```
<window title="Hello" border="normal">
  <label value="Hello World!"/>
</window>
```

Además, es equivalente a

```
<window title="Hello" border="normal">
  <label>Hello World!</label>
</window>
```

Interactividad

Vamos a poner algo de interactividad en ella.

```
<window title="Hello" border="normal">
  <button label="Say Hello" onClick="alert('Hello World!');"/>
</window>
```

Entonces, al hacer clic en el botón, aparecerá la siguiente manera:

GRÁFICO # 39 Mostrando una alerta.



Fuente: Potix (2009) “ZK The Developer's Guide”

El atributo `onClick` es utilizado para añadir un controlador de eventos en el componente. El valor puede ser cualquier código Java, `alert` es una función global para mostrar un mensaje en un cuadro de diálogo y es un acceso directo a uno de los métodos de la clase `org.zkoss.zul.Messagebox`.

```
<button label="Say Hello" onClick="Messagebox.show(&quot;Hello World!&quot;)" />
```

Notas:

- Los scripts incrustados en páginas ZUML pueden ser escritos en diferentes lenguajes, incluyendo pero no limitado a: Java, JavaScript, Ruby y Groovy. Por otra parte, estos scripts se ejecutan en el servidor
- ZK utiliza BeanShell para interpretar Java en tiempo de ejecución, por lo que se podría declarar funciones globales.
- Todas las clases en el paquete `java.lang`, `java.util`, `org.zkoss.zk.ui`, `org.zkoss.zk.ui.event` y `org.zkoss.zul` son importados antes de la evaluación de los códigos de script incrustados en páginas ZUML.

EL ELEMENTO `zscript`

El elemento `zscript` es un elemento especial que define el código de las secuencias de comandos que serán evaluadas cuando se procesa una página ZUML. El uso típico incluye la inicialización y la declaración de variables globales y métodos.

Este código se evalúa sólo cuando la página es cargada. Por ejemplo, en el siguiente código se muestra un mensaje diferente cada vez que se pulsa el botón.

```
<window title="Hello" border="normal">
  <button label="Say Hello" onClick="sayHello()"/>
  <zscript>
    int count = 0;
    void sayHello() { //declare a global function
      alert("Hello World! "+ ++count);
    }
  </zscript>
</window>
```

EL LENGUAJE DE SCRIPTING

De forma predeterminada, el lenguaje scripting es asumido por Java, pero puede seleccionarse diferentes lenguajes, especificando el atributo language.

```
<zscript language="javascript">
  alert('Say Hi in JavaScript');
  new Label("Hi, JavaScript!").setParent(win);
</zscript>
```

También es posible especificar el lenguaje scripting para un controlador de eventos, utilizando un prefijo, por ejemplo javascript: de la siguiente manera, (sin poner espacios en blanco antes o después del nombre del lenguaje a utilizar).

```
<button onClick="javascript: do_something_in_js();"/>
```

INCLUYENDO CÓDIGO EN ARCHIVOS SEPARADOS

Con el fin de mantener separado el código de los elementos visuales, se puede incluir el código de las secuencias de comandos en un archivo separado, por ejemplo sayHello.zs, y luego usar el atributo src para referenciarlo.

```
<window title="Hello" border="normal">
  <button label="Say Hello" onClick="sayHello()"/>
  <zscript src="sayHello.zs"/>
</window>
```

En donde se asume que el contenido sayHello.zs es similar a lo siguiente:

```
int count = 0;
void sayHello() { //declare a global function
    alert("Hello World! "+ ++count);
}
```

EL ELEMENTO attribute

Este elemento permite definir un atributo XML para el elemento en el cual se incluya. Con un uso apropiado, permite que la página sea más legible. El siguiente código es equivalente al script hello.zul descrito previamente.

```
<button label="Say Hello">
    <attribute name="onClick">alert("Hello World!");</attribute>
</button>
```

LAS EXPRESIONES “EL”

Tal como en JSP, es posible utilizar expresiones EL en cualquier parte de una página ZUML. Las expresiones EL utilizan la sintaxis `${expr}`.

```
<element attr1="${bean.property}".../>
${map[entry]}
<another-element>${3+counter} is ${empty map}</another-element>
```

```
<window if="${some > 10}">
```

Cuando una expresión EL es utilizada como el valor de un atributo, este puede retornar cualquier tipo de objeto siempre que sea aceptado por el componente. Por ejemplo, el siguiente código evaluará un objeto Boolean.

Los objetos implícitos estándares, como param y requestScope, y los objetos implícitos de ZK como self y page, son soportados con el fin de simplificar su uso.

```
<textbox value="${param.who} does ${param.what}"/>
```

EL ATRIBUTO id

Para acceder a un componente desde el código java y en las expresiones EL, se debe asignar un identificador con el atributo id.

El siguiente ejemplo muestra como se asigna un identificador a un elemento, el cual luego es manipulado cuando sucede un evento.

```
<window title="Vote" border="normal">
  Do you like ZK? <label id="label"/>
  <separator/>
  <button label="Yes" onClick="label.value = self.label"/>
  <button label="No" onClick="label.value = self.label"/>
</window>
```

Lo siguiente muestra como referenciar a un componente desde una expresión EL.

```
<textbox id="source" value="ABC"/>
<label value="${source.value}"/>
```

LOS ATRIBUTOS if Y unless

Los atributos if y unless son utilizados para controlar cuando se debe crear un componente.

En el siguiente ejemplo, ambas etiquetas son creadas sólo si en la llamada a la página se incluyó un parámetro llamado vote.

```
<label value="Vote 1" if="${param.vote}"/>
<label value="Vote 2" unless="${!param.vote}"/>
```

Si los dos atributos son especificados, el componente no será creado a menos que ambos sean evaluados a verdadero.

EL ATRIBUTO forEach

El atributo forEach es utilizado para controlar cuantos componentes serán creados. Si un conjunto de objetos es especificado para este atributo, ZK Loader creará un componente para cada ítem perteneciente al conjunto especificado. Por ejemplo, en la siguiente página, el elemento listitem será evaluado 3 veces y generará 3 items.

```
<zscript>contacts = new String[] {"Monday", "Tuesday", "Wednesday"};</zscript>
<listbox width="100px">
  <listitem label="{each}" forEach="{contacts}"/>
</listbox>
```

Cuando se evalúa un elemento con el atributo forEach, a la variable each se le asigna uno por uno los objetos del conjunto. El ejemplo anterior da el mismo resultado que el siguiente.

```
<listbox>
  <listitem label="Monday"/>
  <listitem label="Tuesday"/>
  <listitem label="Wednesday"/>
</listbox>
```

De forma alternativa, es posible asignar directamente los valores para el atributo forEach, separados por comas.

```
<listbox>
  <listitem label="{each}" forEach="Monday, Tuesday, Wednesday"/>
</listbox>
```

LOS ATRIBUTOS use Y apply

El código incrustado incorrectamente en una página puede ocasionar muchos dolores de cabeza cuando haya que darle mantenimiento. Hay algunas maneras de separar el código de los elementos visuales (vista). Lo primero sería determinar cuales son los eventos que se requiere manejar, luego invocar los métodos

necesarios. Por ejemplo, se puede invocar métodos para inicializar, procesar y cancelar mediante los eventos onCreate, onOk y onCancel.

```
<window onCreate="MyManager.init(main) "  
onOK="MyManager.process(main) " onCancel="MyManager.cancel(main) "/>
```

Adicionalmente, debe existir una clase llamada MyManager.

```
import org.zkoss.zul.Window;  
  
public class MyManager {  
    public static void init(Window main) { //does initialization  
    }  
    public static void save(Window main) { //saves the result  
    }  
    public static void cancel(Window main) { //cancel any changes  
    }  
}
```

Aunque en este ejemplo aún se requiere incrustar algo de código en la página ZUML. La ventaja de incrustar código incrustado en la interfaz de usuario es la facilidad de cambiar su comportamiento de manera dinámica (lo cual sirve en la fase de prototipo), pero esto hace que el rendimiento no sea muy bueno.

EL ATRIBUTO use

Si no se desea utilizar código java dentro de una página ZUML, es posible extender la implementación de un componente para el manejo de los eventos. Entonces, la clase creada es asignada a través del atributo use.

```
import org.zkoss.zul.Window;  
  
public class MyWindow extends Window {  
    public void onCreate() { //does initialization  
    }  
    public void onOK() { //save the result  
    }  
    public void onCancel() { //cancel any changes  
    }  
}
```

```
<window use="MyWindow">  
    ...  
</window>
```

EL ATRIBUTO `apply`

Si se prefiere el enfoque MVC (Modelo - Vista - Controlador), se puede implementar una clase para inicializar este elemento.

Esta clase deberá implementar la interface `org.zkoss.zk.ui.util.Composer`.

```
import org.zkoss.zk.ui.util.Composer;
import org.zkoss.zul.Window;
public class MyComposer implements Composer {
    public void doAfterCompose(Component comp) {
        ((Window)comp).setTitle("My Title"); //do whatever initialization you want
        //comp is Window since we will specify it to a window later
    }
}
```

Se asume que en la clase especificada existirán por lo menos tres controladores de evento: `MyCreate`, `MyOK` y `MyCancel`.

La clase luego será especificada mediante el atributo `apply`.

```
<window apply="MyComposer">
...
</window>
```

El elemento `window` aún será creado como una instancia de la clase `org.zkoss.zul.Window` y luego pasado al método `doAfterCompose` a través del parámetro `comp`, en este método puede inicializarse de la forma que mejor convenga.

Es posible utilizar múltiples compositores, asignándolos al atributo como una lista separada por comas.

```
<window apply="MyComposer, AnotherComposer">
    <textbox apply="{c:mycomposer()}" />
</window>
```

EL ATRIBUTO forward

Una ventana por lo general está compuesta por un conjunto de uno o varios elementos (botones, menús, cuadros de texto, etc.).

```
<window use="MyWindow">
  ...
  <button label="OK"/>
  <button label="Cancel"/>
</window>
```

Normalmente al hacer clic en un determinado botón, automáticamente se dispara el controlador del evento onClick asociado a dicho botón, en caso de que éste haya sido definido. Sin embargo, la mejor manera de manejar estas situaciones es controlar estos eventos de manera general, esto es, asociándolos a los eventos de la ventana. Para este fin, se utiliza el atributo forward.

```
<window use="MyWindow">
  ...
  <button label="OK" forward="onOK"/>
  <button label="Cancel" forward="onCancel"/>
</window>
```

En este código de ejemplo se interpreta que al ocurrir el evento onClick del botón OK, este se re-direccionará hacia el evento onOK de la ventana (definido a través de la implementación de la clase MyWindow).

De igual manera, al ocurrir el evento onCancel, este será re-direccionado al evento onCancel de la misma ventana. Finalmente definimos en la clase MyWindow cuales serán las acciones que se llevarán a cabo cuando estos eventos sean invocados.

```
public class MyWindow extends Window {
    public void onOK() {
        //called when the OK button is clicked (or the ENTER button is pressed)
    }
    public void onCancel() {
        //called when the Cancel button is clicked (or the ESC button is pressed)
    }
}
```

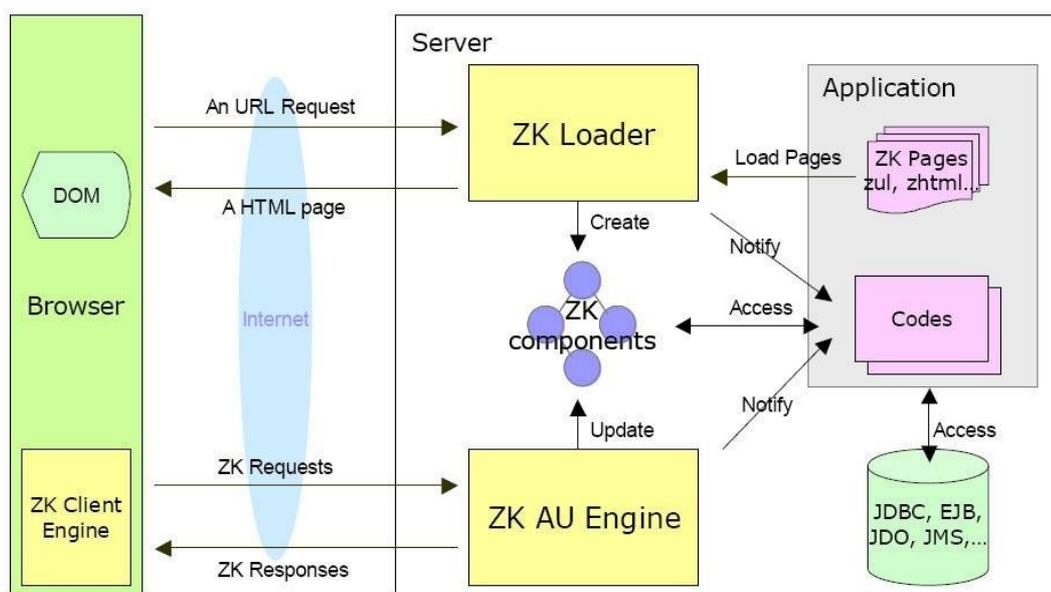
CONCEPTOS BÁSICOS

Esta sección describe los conceptos básicos de ZK. Se utiliza XUL para ilustrar las características de ZK, aunque es aplicable a otros lenguajes soportados por ZK.

DESCRIPCIÓN DE LA ARQUITECTURA

ZK incluye un mecanismo basado en AJAX para automatizar la interactividad, un conjunto de elementos enriquecidos basado en XUL para mejorar la usabilidad, y un lenguaje de marcación basado en XML para simplificar el desarrollo.

GRÁFICO # 40
Arquitectura de ZK.



Fuente: Potix (2009) "ZK The Developer's Guide"

El mecanismo basado en AJAX consta de tres partes: el ZK Loader, El ZK UA Engine y el ZK Client Engine. En base a la petición del usuario, ZK Loader carga una página ZK, la interpreta y devuelve el resultado en páginas HTML. La página de ZK está escrita en un lenguaje de marcación llamado ZUML, que igual que HTML, se utiliza para describir los componentes y su representación visual. El ZK AU Engine y el ZK Client Engine interactúan como un lanzador y un receptor.

Ellos entregan los eventos disparados en el navegador a la aplicación que se ejecuta en el servidor, y actualiza el árbol DOM en el navegador basado en cómo los componentes son manipulados por la aplicación. DOM es la interfaz que proporciona los objetos necesarios para representar documentos HTML y XML, permitiendo el acceso a estos objetos y su manipulación.

EL FLUJO DE EJECUCIÓN

1. Cuando un usuario escribe una dirección URL o hace clic en un hipervínculo en el navegador, se envía una solicitud al Servidor Web. Si se trata de una página perteneciente a una aplicación ZK, ZK Loader se invoca entonces para servir a esta solicitud.
2. ZK Loader carga la página especificada y la interpreta para crear los componentes definidos en la página.
3. Después de interpretar toda la página, ZK Loader convierte el resultado en una página HTML. La Página HTML se envía al navegador del cliente acompañado del ZK Client Engine
4. El ZK Client Engine se ubica en el navegador para detectar cualquier evento provocado por la actividad del usuario, tales como mover el ratón o cambiar un valor. Una vez detectada cualquier evento, se notifica al ZK UA Engine mediante el envío de un ZK request.
5. Ante los ZK requests recibidos desde el Client Engine, el AU Engine actualiza el contenido del componente correspondiente, si es necesario. Y entonces, el AU Engine notifica a la aplicación invocando los controladores de eventos, si los hubiere.
6. Si la aplicación decide cambiar el contenido de los componentes, agregar otros o mover componentes, el AU Engine envía el nuevo contenido de los componentes modificados al Client Engine utilizando ZK responses.
7. Estos ZK responses son realmente comandos que instruyen al Client Engine cómo actualizar el árbol DOM.

COMPONENTS, PAGES Y DESKTOPS

Components (Componentes)

Un componente es un objeto perteneciente a la interfaz del usuario, como una etiqueta o un botón. Este objeto, sólo o en conjunto con otros objetos, define la presentación y el comportamiento de una interfaz de usuario en particular. Mediante la manipulación de los componentes, los desarrolladores controlan la manera de representar visualmente una aplicación en el cliente. Un componente debe implementar la interfaz `org.zkoss.zk.ui.Component`

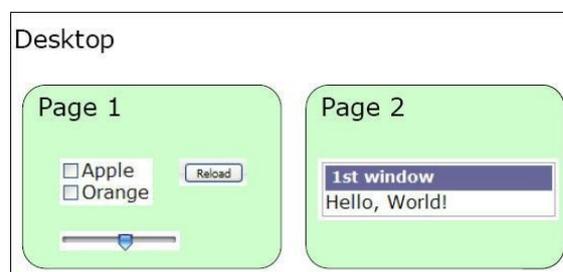
Pages (Páginas)

Una página es una colección de componentes. Limita los componentes de la página pertenecientes a la misma, de tal manera que se muestren en cierta sección del navegador. Una página es creada automáticamente cuando ZK Loader interpreta una página ZUML.

Desktops (Escritorios)

Una página ZUML podría incluir a otras páginas ZUML, directa o indirectamente. Estas páginas son creadas con el fin de servir a la misma solicitud URL, y en conjunto son conocidas como Desktop.

GRÁFICO # 41
El Desktop de ZK.



Fuente: Potix (2009) “ZK The Developer's Guide”

Como una aplicación ZK interactúa con el usuario, es posible que otras páginas puedan agregarse a un Desktop y de igual forma otras páginas podrían ser eliminadas del mismo. De manera similar, un componente puede añadirse o

quitarse de una página. Tanto pages y desktops se crean y eliminan de manera implícita. No hay API para crearlas o eliminarlas. Un page se crea cada vez que ZUML carga una página y es eliminado cuando ZK no encuentra más referencias al mismo. Un desktop es creado cuando la primera página ZUML es cargada y es eliminado si demasiados desktops han sido creados para una sesión específica.

Component: Una parte visual y un objeto Java

Además de ser un objeto de Java en el servidor, un componente tiene una parte visual en el navegador, si y sólo si pertenece a una página. Cuando un componente es anexado a una página, su parte visual es creada. Cuando un componente es descartado de una página, se elimina su parte visual.

Se puede incluir un componente en una página llamando al método `setPage` para hacer que se convierta en un componente raíz de determinada página o también llamando al método `setParent`, `insertBefore` o `appendChild` para hacer que se convierta en el hijo de otro componente. Del mismo modo, puede eliminarse un componente raíz de una página llamando al método `setPage` con el valor `null`. Un hijo es eliminado al eliminarse explícitamente de su padre o al eliminarse el padre de la página a la que pertenece.

Eventos

Un evento notifica a la aplicación de lo que sucede. Cada tipo del evento está representado por una clase distinta, `org.zkoss.zk.ui.event.MouseEvent` denota una actividad del ratón. Para responder a un evento, una aplicación tiene que registrar uno o varios controladores de eventos. Se puede registrar un controlador de eventos especificando el atributo `onXxx` en el lenguaje utilizado o también llamando al método `addEventListener` para el componente o la página que se desea controlar.

EL LENGUAJE DE MARCACIÓN DE LA INTERFAZ DE USUARIO

El lenguaje de marcación de la interfaz de usuario de ZK (ZUML por sus siglas en inglés) está basado en XML. Cada elemento XML describe el componente a crear. Un atributo XML describe un valor inicial que se asignará al componente creado. Una instrucción de procesamiento XML describe como procesar la página entera.

Los diferentes conjuntos de componentes son diferenciados por los XML namespaces (contextos en los cuales se referencian nombres o identificadores). Por ejemplo, el namespace de XUL es <http://www.zkoss.org/2005/zul>, y el de XHTML es <http://www.w3.org/1999/xhtml>.

OBJETOS IMPLÍCITOS

Para los scripts incrustados en una página ZUML, hay un conjunto de objetos implícitos que permiten a los desarrolladores acceder a los componentes de una manera más eficiente. Por ejemplo, `self` es una instancia de `org.zkoss.zk.ui.Component` para representar al componente que está siendo procesado. A continuación de detallan los principales objetos implícitos:

CUADRO # 35
Listado de objetos implícitos.

Nombre	Descripción
<code>self</code>	<code>org.zkoss.zk.ui.Component</code> El mismo componente.
<code>page</code>	<code>org.zkoss.zk.ui.Page</code> La página. Es lo mismo que <code>self.page</code> .
<code>desktop</code>	<code>org.zkoss.zk.ui.Desktop</code> El desktop. Es igual que <code>self.desktop</code> .
<code>execution</code>	<code>org.zkoss.zk.ui.Execution</code> Representa al elemento de ejecución actual.

Fuente: Potix (2009) “ZK The Developer's Guide”

CUADRO # 36
Listado de objetos implícitos(continuación).

Nombre	Descripción
session	org.zkoss.zk.ui.Session La sesión.
application	org.zkoss.zk.ui.WebApp La aplicación web.
applicationScope	java.util.Map Un mapa de atributos definidos en la aplicación, igual que el método getAttributes en la interface org.zkoss.zk.ui.WebApp.
forEachStatus	org.zkoss.zk.ui.util.ForEachStatus Estado de una iteración. ZK provee la información relativa a la iteración que se lleva a cabo cuando se evalúa un elemento.
componentScope	java.util.Map Un mapa de atributos definidos en el componente. Es lo mismo que el método getAttributes en la interface org.zkoss.zk.ui.Component.
pageScope	java.util.Map Un mapa de atributos definidos en la página. Es lo mismo que el método getAttributes en la interface org.zkoss.zk.ui.Page.
desktopScope	java.util.Map Un mapa de atributos definidos en el desktop. Es lo mismo que el método getAttributes en la interface org.zkoss.zk.ui.Desktop.
sessionScope	java.util.Map Un mapa de atributos definidos en la sesión. Es lo mismo que el método getAttributes en la interface org.zkoss.zk.ui.Session.
each	java.lang.Object El ítem actual de la colección que está siendo iterada, cuando ZK evalúa un elemento iterativo. Un elemento iterativo es un elemento con una asignación al atributo forEach.
requestScope	java.util.Map Un mapa de atributos definidos en la solicitud, igual que getAttributes en la interface org.zkoss.zk.ui.Execution.
event	org.zkoss.zk.ui.event.Event o derivado del mismo El evento actual. Disponible sólo para un controlador de eventos.

Fuente: Potix (2009) “ZK The Developer's Guide”

LOS ELEMENTOS DE ZK

Los elementos de ZK son usados para controlar las páginas ZUML a diferencia de la creación de componentes.

El elemento zk

Es un elemento especial utilizado para añadir otros componentes. A diferencia de un componente real (como hbox o div), este no forma parte del árbol de componentes que se genera al interpretar la página. Es decir, no representa a ningún componente.

```
<window>
  <zk>
    <textbox/>
    <textbox/>
  </zk>
</window>
```

Es equivalente a:

```
<window>
  <textbox/>
  <textbox/>
</window>
```

Múltiples elementos raíz en una página

Debido a la limitación de la sintaxis XML, sólo es posible especificar una raíz en el documento. Por tal razón, si es necesario definir más de una raíz en el documento, debe definirse el elemento zk para agrupar a esas raíces.

```
<?page title="Multiple Root"?>
<zk>
  <window title="First">
    ...
  </window>
  <window title="Second" if="{param.secondRequired}">
    ...
  </window>
</zk>
```

Iteración versátil sobre componentes

El elemento `zk`, soporta también el atributo `forEach`. De esta manera, es posible utilizarlo para generar componentes de diferente tipo en dependencia de las condiciones.

En el siguiente ejemplo, se asume que `mycols` es una colección de objetos que tienen algunos miembros, `isUseText()`, `isUseDate()` y `isUseCombo()`.

```
<window>
  <zk forEach="${mycols}">
    <textbox if="${each.useText}"/>
    <datebox if="${each.useDate}"/>
    <combobox if="${each.useCombo}"/>
  </zk>
</window>
```

El elemento `zscript`

```
<zscript [language="Java"]> Código del script </zscript>
<zscript src="uri" [language="Java"] />
```

Este elemento define una porción de código de programación, el mismo que será interpretado cuando la página sea evaluada. El lenguaje de este código es por defecto Java, aunque puede seleccionarse otros lenguajes con el atributo `language`. El elemento `zscript` tiene dos formatos. El primer formato es usado para incrustar código directamente en la página. El segundo formato sirve para referenciar un archivo externo que es el que contiene el código de programación, mediante el atributo `src` que debe tener la url del archivo a utilizar.

El elemento `attribute`

Define un atributo del elemento en el cual está incluido. Su contenido es el valor del atributo, mientras que el atributo `name` especifica el nombre del atributo. Es muy útil si el valor del atributo es sofisticado, o es un atributo condicional.

```
<window>
  <attribute name="title" if="{new}">Untitled</attribute>
  <attribute name="title" unless="{new}">{title}</attribute>
</window>
```

Adicionalmente, es posible especificar un fragmento de código XML como valor del atributo. Este fragmento es llamado contenido nativo.

```
<html>
  <attribute name="content">
    <ol>
      <li forEach="{values}">{each}</li>
    </ol>
  </attribute>
</html>
```

Donde ol y li son parte del contenido nativo. Ellos no son componentes de ZK y serán eventualmente convertidos en una instancia del objeto String para ser asignados al atributo especificado.

El elemento variables

Permite definir un conjunto de variables. Es equivalente al método setVariable de los componentes.

Este elemento es muy útil si se desea asignar variables sin utilizar código de programación.

```
<window>
  <variables rich="simple" simple="intuitive"/>
</window>
```

Es equivalente a:

```
<window>
  <zscript>
    self.setVariable("rich", "simple", false);
    self.setVariable("simple", "intuitive", false);
  </zscript>
</window>
```

Por supuesto, también pueden utilizarse expresiones EL para los valores.

```
<window>
  <window id="w" title="Test">
    <variables title="{w.title}"/>
      1: ${title}
    </window>
    2: ${title}
  </window>
```

ZUML CON EL CONJUNTO DE COMPONENTES XUL

Esta sección describe el conjunto de componentes XUL. Algunos componentes puede que no sean totalmente compatibles con el estándar XUL.

Para mejorar la comprensión del manual, este tipo de componentes pueden ser referidos como componentes ZUL.

COMPONENTES BÁSICOS

ETIQUETAS (LABEL)

El componente label (etiqueta) es la representación de un texto y no requiere una definición específica.

```
<window border="normal">
  Hello World
</window>
```

Si se desea especificar atributos para un label, será necesario definir la etiqueta <label>.

```
<window border="normal">
  <label style="color: red" value="Hello World"/>
</window>
```

LAS PROPIEDADES **pre**, **hyphen**, **maxlength** y **multiline**

Controlan la forma en que una etiqueta es mostrada. Por ejemplo, si el atributo **pre** se especifica a **true**, todos los espacios en blanco, los saltos de línea, así como los tabuladores son conservados.

CUADRO # 37
Propiedades **hyphen, **pre** y **maxlength**.**

hyphen	pre	maxlength	Descripción
false	false	positive	Trunca los caracteres que exceden la longitud máxima especificada con maxlength .
true	any	positive	Si la longitud de la línea es excedida, la línea es truncada con un guión y continuada en la siguiente línea.
false	true	any	La longitud máxima es ignorada.

Fuente: Potix (2009) “ZK The Developer's Guide”

La propiedad **multiline** es similar a la propiedad **pre**, excepto que sólo conserva los saltos de línea y los espacios en blanco al inicio de cada línea.

BOTONES (BUTTON)

Hay dos tipos de botones: **button** y **toolbarbutton**. Se comportan de manera similar pero su apariencia es diferente. El componente **button** usa la etiqueta html **BUTTON**, mientras que el componente **toolbarbutton** usa la etiqueta html **A**.

Es posible asignar una etiqueta y una imagen a un botón mediante las propiedades **label** e **image**. Si ambas propiedades son especificadas, se muestra primero la imagen y luego la etiqueta.

La propiedad `dir` permite cambiar el orden en que estos elementos se muestran y la propiedad `orient` permite especificar si se muestran horizontalmente o verticalmente.

```
<button label="Left" image="/img/folder.gif" width="125px"/>
<button label="Right" image="/img/folder.gif" dir="reverse" width="125px"/>
<button label="Above" image="/img/folder.gif" orient="vertical" width="125px"/>
<button label="Below" image="/img/folder.gif" orient="vertical" dir="reverse"
width="125px"/>
```

Además de identificar las imágenes por medio de una URL, se puede asignar una imagen generada dinámicamente mediante el uso del método `setImageContent`.

EL EVENTO `onClick` Y LA PROPIEDAD `href`

Hay dos formas de añadir comportamientos a estos componentes. La primera es especificando un procedimiento que se ejecute al ocurrir el evento `onClick`. La segunda es asignando una URL a través de la propiedad `href`. Si ambas propiedades son especificadas, la propiedad `href` es la que tiene mayor prioridad, por tanto, el evento definido en `onClick` no será tomado en cuenta.

```
<button onClick="do_something_in_Java()" />
<button href="/another_page.zul" />
```

EL MÉTODO `sendRedirect` DE LA INTERFAZ `org.zkoss.zk.ui.Execution`

Con el evento `onClick` es posible redireccionar a otra página utilizando el método `sendRedirect`. Podría decirse que los siguientes dos botones son equivalentes:

```
<button onClick="Executions.sendRedirect(&quot;another.zul&quot;)" />
<button href="another.zul" />
```

La diferencia entre `onClick` y `href` radica en que mientras el primero es enviado al servidor para la ejecución, el segundo se procesa completamente en el cliente.

CONTROLES DE ENTRADA

Un conjunto de controles de entrada de datos son soportados en el conjunto de componentes XUL, tales como: textbox, intbox, decimalbox, doublebox, datebox, combobox y bandbox. Éstos son utilizados para permitir al usuario el ingreso de diferentes tipos de datos. Los controles combobox y bandbox son controles de entrada especiales, comparten las propiedades comunes descritas en esta sección pero sus características particulares serán revisadas en una sección posterior.

LA PROPIEDAD type

Se puede establecer el valor de esta propiedad con el valor password para el componente textbox, con el fin de que no se muestre lo que el usuario está ingresando.

```
Nombre de usuario: <textbox/>
Clave: <textbox type="password"/>
```

LA PROPIEDAD format

Es posible controlar el formato de un componente de entrada. Por defecto esta propiedad tiene el valor null. Para el componente datebox, este valor especifica el formato yyyy/MM/dd, y para intbox y decimalbox, no asigna ningún formato.

```
<datebox format="MM/dd/yyyy"/>
<decimalbox format="#.##0.##"/>
```

De manera similar a otras propiedades, esta puede ser cambiada dinámicamente, tal como se muestra a continuación:

```
<datebox id="db"/>
<button label="set MM-dd-yyyy" onClick="db.setFormat(&quot;MM-dd-yyyy&quot;);"/>
```

Constraints

Es posible especificar qué valores son aceptados por un control utilizando la propiedad `constraint`. El valor de esta propiedad es una combinación de una expresión regular y de uno de los siguientes constraints: `no positive`, `no negative`, `no zero`, `no empty`, `no future`, `no past`, `no today`. Los primeros tres constraints sólo se aplican a los controles `intbox` y `decimalbox`. Los valores `no future`, `no past` y `no today` sólo son aplicables a controles de tipo `datebox`. El valor `no empty` puede aplicarse a cualquiera de los controles.

Los constraints con expresiones regulares son aplicables a controles que manejen datos de tipo cadena como `textbox`, `combobox` y `bandbox`. Se puede especificar más de un constraint separándolos con una coma. Ejm:

```
<intbox constraint="no negative,no zero"/>
```

Constraints para datebox

Además de los constraints aplicables ya indicados, un control de tipo `datebox` soporta la definición de un rango de fechas. Por ejemplo:

```
<datebox constraint="between 20071225 and 20071203"/>  
<datebox constraint="before 20071225"/>  
<datebox constraint="after 20071225"/>
```

Donde observamos lo siguiente:

1. El formato de la fecha dentro del constraint es `yyyyMMdd`. Es independiente de la localidad.
2. La fecha establecida en el constraint es inclusiva, esto quiere decir que en el constraint “before 20071225” se incluye el 25 de diciembre del 2007 y cualquier fecha anterior a esa.

3. El constraint es representado con una instancia de la clase `org.zkoss.zul.SimpleDateConstraint`, la cual devuelve validadas las fechas de inicio y fin mediante el uso de los métodos `getBeginDate` y `getEndDate`.

```
Ejm: ((SimpleDateConstraint) datebox.getConstraint()).getBeginDate();
```

Constraints personalizados

En caso de ser necesaria la definición de un constraint más sofisticado, es posible especificar un objeto que implemente la interface `org.zkoss.zul.Constraint`, como se observa a continuación:

```
<window title="Custom Constraint">
  <zscript><![CDATA[
Constraint ctt = new Constraint() {
  public void validate(Component comp, Object value) throws WrongValueException {
    if (value == null || ((Integer)value).intValue() < 100)
      throw new WrongValueException(comp, "At least 100 must be specified");
  }
}]]></zscript>
  <intbox constraint="{ctt}"/>
</window>
```

De igual manera, es posible implementar el constraint dentro de una clase Java, por decir `my.EmailValidator`, que sería utilizada de la siguiente forma:

```
<?taglib uri="http://www.zkoss.org/dsp/web/core" prefix="c"?>
<textbox constraint="{c:new('my.EmailValidator')}" />
```

EL EVENTO `onChange`

Un control de entrada de datos notifica a la aplicación a través de un evento `onChange` que el contenido del mismo ha sido cambiado por el usuario. Cuando el evento `onChange` es invocado ya el valor ha sido asignado al control, así que sería muy tarde si lo que se quiere es evitar el ingreso de un valor inválido, por tanto si lo que se desea es hacer una validación se recomienda el uso de constraints.

EL EVENTO onChanging

Un control de entrada también notifica a la aplicación con el evento onChanging cuando un usuario está modificando su contenido. En este caso, el valor cambiado aún no ha sido asignado al control, por tanto el valor anterior aún permanece en el control, y si se desea acceder al nuevo valor se debe acceder a la propiedad value del evento, tal como se indica a continuación.

```
<grid>
  <rows>
    <row>The onChanging textbox:
      <textbox onChanging="copy.value = event.value"/></row>
    <row>Instant copy:
      <textbox id="copy" readonly="true"/></row>
  </rows>
</grid>
```

En este caso, el evento ocurre muy temprano para poder validar el ingreso del contenido de un control de entrada, ya que el usuario puede que aún no haya completado el cambio.

Por tanto, es recomendable utilizar un constraint si se desea verificar un ingreso de datos.

EL COMPONENTE WINDOW

Una ventana (window) es, igual que la etiqueta DIV html, utilizada para agrupar componentes. Se diferencia de los otros componentes en que:

- Una ventana puede establecerse como un diálogo de tipo modal.

Una ventana puede tener un título, una leyenda y un borde. El título es especificado a través de la propiedad title. La leyenda es especificada declarando un componente hijo llamado caption.

GRÁFICO # 42

Propiedades caption y label de la ventana.



Fuente: Potix (2009) "ZK The Developer's Guide"

```
<window title="Demo" border="normal" width="350px">
  <caption>
    <toolbarbutton label="More"/>
    <toolbarbutton label="Help"/>
  </caption>
  <toolbar>
    <toolbarbutton label="Save"/>
    <toolbarbutton label="Cancel"/>
  </toolbar>
  What is your favorite framework?
  <radiogroup>
    <radio label="ZK"/>
    <radio label="JSF"/>
  </radiogroup>
</window>
```

LA PROPIEDAD closable

Al asignarle el valor true a la propiedad closable, un botón se muestra en la esquina superior derecha de la ventana, el mismo que al ser presionado permite cerrarla, generando un evento onClose, siendo posible sobre-escribir la acción realizada a través de la respectiva propiedad, event.stopPropagation() debe ser llamado para evitar que Window.onClose() sea llamado.

```
<window closable="true" title="Detach on Close" border="normal" width="200px"
onClose="self.visible = false; event.stopPropagation();">
  In this example, this window hides itself when the close button is clicked.
</window>
```

Los modos de una ventana

Una ventana puede estar en uno de los siguientes modos: overlapped, popup, modal, highlighted y embedded. Por defecto, está en el modo embedded. Puede cambiarse el modo de una ventana utilizando los métodos doOverlapped, doPopup, doModal, doHighlighted y doEmbedded.

```

<zk>
  <window id="win" title="Hi!" border="normal" width="200px">
    <caption>
      <toolbarbutton label="Close" onClick="win.setVisible(false)"/>
    </caption>
    <checkbox label="Hello, Wolrd!"/>
  </window>
  <button label="Overlap" onClick="win.doOverlapped();"/>
  <button label="Popup" onClick="win.doPopup();"/>
  <button label="Modal" onClick="win.doModal();"/>
  <button label="Embed" onClick="win.doEmbedded();"/>
  <button label="Highlighted" onClick="win.doHighlighted();"/>
</zk>

```

EMBEDDED

Una ventana en modo embedded se coloca alineada a los demás componentes. En este modo es imposible que cambie de posición.

OVERLAPPED

Una ventana en modo overlapped está sobrepuesta a otros componentes, así que los usuarios pueden arrastrarla y soltarla en cualquier lugar, y el desarrollador puede cambiar su ubicación con los métodos setLeft y setTop. Además del método doOverlapped, puede asignarse directamente a la propiedad mode de la ventana con valor “overlapped”.

POPUP

Una ventana en modo popup es similar a una ventana en modo overlapped, con la diferencia de que ésta se cierra cuando el usuario selecciona alguno de los componentes de la misma. Como el nombre lo sugiere, está diseñada para implementar ventanas emergentes.

MODAL

Una ventana modal es similar a una ventana overlapped, pero esta suspende la ejecución hasta que uno de los siguientes métodos sea ejecutado: endModal, doEmbedded, doOverlapped, doHighlighted y doPopup. Además de suspender la ejecución, evita que se pueda seleccionar cualquier otro componente que no pertenezca a la ventana modal. Una ventana modal es ubicada automáticamente en

el centro del navegador, por tanto no es posible controlar su ubicación mediante código. A diferencia de los demás modos, una ventana sólo puede ponerse en modo modal dentro de un controlador de evento, mediante los métodos `doModal()` o `serMode("modal")`.

```
<zk>
  <window id="wnd" title="My Modal" visible="false" width="300px">
    <button label="close" onClick="wnd.visible = false"/>
  </window>
  <button label="do it" onClick="wnd.doModal()" />
</zk>
```

Por tanto, el siguiente código es incorrecto debido a que se ejecuta en la etapa de creación de componentes.

```
//t1.zul
<window title="My Modal" width="300px" closable="true" mode="modal">
</window>
```

HIGHLIGHTED

Una ventana `highlighted` es similar a una ventana `overlapped`, excepto que el efecto visual es el mismo al de una ventana modal. En otras palabras, una ventana en este modo es posicionada en el centro del navegador, y los componentes que no pertenecen a la misma son deshabilitados. Sin embargo, la ejecución no es suspendida.

Tal como en las ventanas en modo `overlapped`, la ejecución continúa en la siguiente sentencia una vez que se ha cambiado el modo. En el siguiente ejemplo `f1()` es llamado sólo después de que `win1` es cerrada, mientras que `g()` es llamada inmediatamente después de que `win2` cambia a modo `highlighted`.

```
win1.doModal(); //the execution is suspended until win1 is closed
f1();

win2.doHighlighted(); //the execution won't be suspended
g1()
```

LA PROPIEDAD `position`

Además de las propiedades `top` y `left`, se puede controlar la ubicación de una ventana en el estado `overlapped`, `popup` o `modal` con el uso de la propiedad `position`. El siguiente código ubica la ventana en la esquina inferior derecha.

```
<window width="300px" mode="overlapped" position="right,bottom">  
...
```

El valor de la propiedad `position` puede ser una combinación de las siguientes constantes, separadas por una coma.

CUADRO # 38
Valores de la propiedad `position`.

Valor	Descripción
<code>center</code>	Ubica la ventana en el centro. Si <code>left</code> o <code>right</code> también fueron especificadas, ubica la ventana centrada verticalmente. Si <code>top</code> o <code>bottom</code> también fueron especificadas, ubica la ventana centrada horizontalmente. Las propiedades <code>left</code> y <code>top</code> son ignoradas.
<code>left</code>	Ubica la ventana en el borde izquierdo. La propiedad <code>left</code> es ignorada.
<code>right</code>	Ubica la ventana en el borde derecho. La propiedad <code>left</code> es ignorada.
<code>top</code>	Ubica la ventana en la parte superior. La propiedad <code>top</code> es ignorada.
<code>bottom</code>	Ubica la ventana en la parte inferior. La propiedad <code>top</code> es ignorada.

Fuente: Potix (2009) “ZK The Developer's Guide”

EL MODELO DE CAJAS (BOX MODEL)

Componentes: `vbox`, `hbox` y `box`.

El modelo de cajas de XUL es usado para dividir una porción del área de trabajo en una serie de cajas. Los componentes incluidos en las cajas se ubicarán así

mismos horizontal o verticalmente. Al combinar una serie de cajas y separadores, es posible controlar distribución de la presentación visual.

Una caja puede colocar a sus componentes hijos en una de dos orientaciones, horizontal o vertical. Una caja horizontal (hbox) alinea sus componentes horizontalmente y una caja vertical (vbox) los alinea de forma vertical.

Se podría decir que cada caja es una fila o una columna de una tabla HTML. Cada componente hbox es ubicado de forma horizontal de manera similar a las filas y cada vbox se añadiría como una columna.

```
<zk>
  <vbox>
    <button label="Button 1"/>
    <button label="Button 2"/>
  </vbox>
  <hbox>
    <button label="Button 3"/>
    <button label="Button 4"/>
  </hbox>
</zk>
```

Existe también un componente genérico orientado por defecto horizontalmente, aunque es posible utilizar la propiedad orient para controlar la orientación con los valores horizontal o vertical.

Las siguientes instrucciones son equivalentes:

```
<vbox>
<box orient="vertical">
```

Se puede añadir los componentes que se necesiten dentro de una caja, incluso otras cajas en caso de un vbox cada componente se ubica a la derecha del anterior.

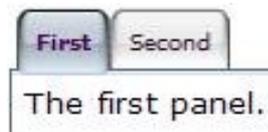
LOS TABULADORES (TAB BOXES)

Componentes: tabbox, tabs, tab, tabpanels y tabpanel.

Los componentes de esta categoría son de mucha utilidad cuando tenemos una interfaz que debe manejar una gran cantidad de componentes, ya que mediante su uso permite agrupar estos componentes en paneles y mostrar sólo los componentes del panel activo.

```
<tabbox>
  <tabs>
    <tab label="First"/>
    <tab label="Second"/>
  </tabs>
  <tabpanels>
    <tabpanel>The first panel.</tabpanel>
    <tabpanel>The second panel</tabpanel>
  </tabpanels>
</tabbox>
```

GRÁFICO # 43
Uso de Tabbox.



Fuente: Potix (2009) “ZK The Developer's Guide”

El ejemplo anterior muestra el efecto obtenido con el uso de estos componentes logrando una presentación más organizada.

A continuación se detalla cada uno de estos componentes:

- **tabbox:** Es el componente de más alto nivel, el cual contiene los elementos tabs y tabpanels.
- **tabs:** Es un contenedor de tabuladores o una colección de componentes de tipo tab.
- **tab:** Un tabulador específico. Al hacer clic en un tabulador un tabpanel es mostrado. Cada tabulador puede contener un elemento label y un image.

- **tabpanel:** El contenedor para los tabpanels o una colección de componentes de tipo tabpanel.
- **tabpanel:** Es el cuerpo de un tabpanel. En el se puede colocar un conjunto de componentes. Los elementos tabpanel se corresponden a cada elemento tab en el orden en que fueron definidos.

Al componente tab que se encuentre seleccionado se le asigna el valor de true en la propiedad selected. Sólo un tabulador puede estar seleccionado a la vez. Hay dos maneras de seleccionar el tabulador seleccionado mediante código Java.

```
tab1.setSelected(true);  
tabbox.setSelectedTab(tab1);
```

También puede asignarse directamente true a la propiedad del tabulador.

```
<tab label="My Tab" selected="true"/>
```

Si no se define qué tabulador va a quedar seleccionado, el primero es seleccionado automáticamente.

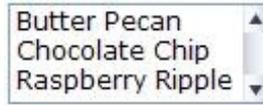
LIST BOXES

Componentes: listbox, listitem, listcell, listhead y listheader.

Un listbox es utilizado para mostrar un número de ítems en una lista. El usuario puede seleccionar un ítem desde la lista. La siguiente es la forma más simple de definirla, un list box con una columna y de selección única.

```
<listbox>  
  <listitem label="Butter Pecan"/>  
  <listitem label="Chocolate Chip"/>  
  <listitem label="Raspberry Ripple"/>  
</listbox>
```

GRÁFICO # 44 Uso del Listbox.



Fuente: Potix (2009) "ZK The Developer's Guide"

Un listbox soporta dos modos de apariencia: default y select. Si se selecciona el modo select, una etiqueta SELECT de HTML también es generada.

```
<listbox mold="select">...</listbox>
```

Listboxes multi-columna

Un listbox también soporta la definición de múltiples columnas. Cuando un usuario selecciona un ítem, toda la fila es seleccionada. Para especificar una lista de múltiples columnas, es necesario, especificar los componentes listcell como columnas usadas en cada listitem (tal como una fila).

```
<listbox width="200px">
  <listitem>
    <listcell label="George"/>
    <listcell label="House Painter"/>
  </listitem>
  <listitem>
    <listcell label="Mary Ellen"/>
    <listcell label="Candle Maker"/>
  </listitem>
  <listitem>
    <listcell label="Roger"/>
    <listcell label="Swashbuckler"/>
  </listitem>
</listbox>
```

Encabezados de columna

Se puede especificar los encabezados de columna con el uso de listhead y listheader. También es posible asignar una imagen como encabezado a través de la propiedad image.

```

<listbox width="200px">
  <listhead>
    <listheader label="Name"/>
    <listheader label="Occupation"/>
  </listhead>
  ...
</listbox>

```

GRÁFICO # 45
ListBox multi-columna con encabezados.

Name	Occupation
George	House Painter
Mary Ellen	Candle Maker
Roger	Swashbuckler

Fuente: Potix (2009) “ZK The Developer's Guide”

Pie de columnas

Se puede especificar pies de columna con el uso de listfoot y listfooter. Nótese que el orden de listhead y listfoot no es relevante.

GRÁFICO # 46
ListBox con pie de columnas.

Population	%
A. Graduate	20%
B. College	23%
C. High School	40%
D. Others	17%
More or less	100%

Fuente: Potix (2009) “ZK The Developer's Guide”

Cada vez que una instancia de listhead es añadido a un listbox, este deberá ser el primer hijo, y una instancia de listfoot deberá ser el último.

```

<listbox width="200px">
  <listhead>
    <listheader label="Population"/>
    <listheader align="right" label="%" />
  </listhead>
  <listitem id="a" value="A">
    <listcell label="A. Graduate"/>
    <listcell label="20%" />
  </listitem>
  <listitem id="b" value="B">
    <listcell label="B. College"/>
    <listcell label="23%" />
  </listitem>

```

```

<listitem id="c" value="C">
  <listcell label="C. High School"/>
  <listcell label="40%"/>
</listitem>
<listitem id="d" value="D">
  <listcell label="D. Others"/>
  <listcell label="17%"/>
</listitem>
<listfoot>
  <listfooter label="More or less"/>
  <listfooter label="100%"/>
</listfoot>
</listbox>

```

Listas desplegables

Para crear una lista desplegable se modifica la apariencia del Listbox y se define la lista con una sola fila. En este caso no se puede definir múltiples columnas.

```

<listbox mold="select" rows="1">
  <listitem label="Car"/>
  <listitem label="Taxi"/>
  <listitem label="Bus" selected="true"/>
  <listitem label="Train"/>
</listbox>

```

GRÁFICO # 47
ListBox como lista desplegable.



Fuente: Potix (2009) “ZK The Developer's Guide”

TRABAJAR CON ETIQUETAS HTML

Hay algunas maneras de utilizar etiquetas HTML con componentes XUL en la misma página ZUML. La selección se realiza según los requerimientos del caso. La primera manera es utilizando el componente html para incrustar etiquetas HTML. Bajo este enfoque, las etiquetas HTML son simplemente el contenido del componente html. Estas etiquetas son enviadas directamente al cliente y no tienen ningún significado especial para ZK.

La segunda manera es utilizando el namespace XHTML (<http://www.w3.org/1999/xhtml>) para especificar un componente desde el conjunto de componentes XHTML. Igual que con los componentes ZUL (<http://www.zkoss.org/2005/zul>), ZK crea una instancia para cada elemento XML en la página ZUML.

La tercera forma es mediante el uso del namespace nativo (<http://www.zkoss.org/2005/zk/native>) para representar una etiqueta HTML que tendrá que ser enviada directamente al cliente en vez de crear un componente ZK por cada uno de ellos. De esta manera, se logra mayor eficiencia, pero en cambio no es modificable dinámicamente.

La última forma, pero no por eso la menos importante, se puede utilizar inclusiones (include) y frames inline (iframe) para incrustar otra clase de contenido en una página ZUL, en teoría es posible incluir cualquier clase de contenido (no está limitado a etiquetas HTML).

EL COMPONENTE html

```
<window title="Html Demo">
  <html><![CDATA[
    <h4>Hi, ${parent.title}</h4>
    <p>It is the content of the html component.</p>
  ]]></html>
</window>
```

La forma más sencilla es utilizando un componente XUL llamado html para incrustar cualquier etiqueta HTML que se desee enviar directamente al navegador. Para evitar que ZK interprete las etiquetas HTML usualmente se las encierra entre `<![CDATA[y]]>`. En otras palabras, estas etiquetas no pueden especificarse como componentes hijo. Más bien son almacenadas dentro de la propiedad content. Nótese que es posible la utilización de expresiones EL dentro de ella. Donde `<h4> ... </p>` formarán parte del contenido del elemento html.

También es posible utilizar el elemento `attribute` para especificar el fragmento XHTML en lugar de usar `CDATA`.

```
<html>
  <attribute name="content">
    <h4>Hi, ${parent.title}</h4>
    <p>It is the content of the html component.</p>
  </attribute>
</html>
```

El componente `html` genera una etiqueta `span` HTML para incluir este contenido. En otras palabras, el siguiente contenido es generado cuando la página es enviada al navegador.

```
<span id="z_4a_3">
  <h4>Hi, Html Demo</h4>
  <p>It is the content of the html component.</p>
</span>
```

El componente `html` no es diferente de los otros componentes XUL. Por ejemplo, se puede especificar un estilo CSS y cambiar su contenido dinámicamente.

```
<html id="h" style="border: 1px solid blue;background: yellow"><![CDATA[
<ul>
<li>Native browser content</li>
</ul>
]]></html>
<button label="change" onClick="l.setContent('&quot;Hi, Update&quot;')"/>
```

EL COMPONENTE `iframe`

El componente `iframe` utiliza la etiqueta HTML `iframe` para delegar una porción del área de la página a otra URL. Aunque el resultado obtenido es muy similar al logrado con el componente `include`, el concepto y significado es diferente. El contenido incluido en un componente `include` es un fragmento de la totalidad de la página HTML. Debido a que el contenido es parte de la página HTML, este es parte del desktop y es posible acceder a cualquiera de sus componentes. La inclusión es hecha en el servidor, y el navegador no sabe nada

sobre él. Significa que la URL especificada por src podría hacer referencia a un recurso interno. Por otra parte, el contenido de un componente iframe es cargado como una página independiente, y el formato del contenido podría ser diferente de HTML. Por ejemplo, se podría incrustar un archivo PDF, tan sólo con especificar la ruta del recurso.

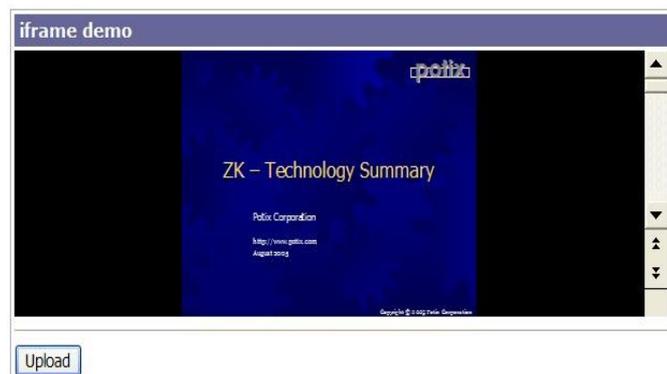
```
<iframe src="/my.pdf"/>
...other HTML content
```

La incrustación es realizada por el navegador, cuando este interpreta la página HTML. Esto también implica que la URL debe especificar un recurso que pueda ser accedido por el navegador. El siguiente ejemplo ilustra la posibilidad de incrustar cualquier contenido, siempre que el cliente soporte dicho formato.

```
<window title="iframe demo" border="normal">
  <iframe id="iframe" width="95%"/>
  <separator bar="true"/>
  <button label="Upload">
    <attribute name="onClick">{
      Object media = Fileupload.get();
      if (media != null)
        iframe.setContent(media);
    }</attribute>
  </button>
</window>
```

En la siguiente imagen se observa como luce la página luego de que se ha subido un archivo con una presentación de diapositivas.

GRÁFICO # 48 Uso de iframe.



Fuente: Potix (2009) "ZK The Developer's Guide"

EVENTOS

Los eventos son soportados de acuerdo al componente utilizado, los cuales son ejecutados después de que el contenido del componente ha sido actualizado.

Eventos del mouse

Son provocados por la interacción del usuario con el mouse.

CUADRO # 39
Eventos generados por el mouse.

Nombre	Componente/Descripción
onClick	button, caption, column, div, groupbox, image, imagemap, label, listcell, listfooter, listheader, menuitem, tab, tabpanel, toolbar, toolbarbutton, treecell, treecol, window Evento: org.zkoss.zk.ui.event.MouseEvent Indica que el usuario le ha dado un click a un componente.
onRightClick	button, caption, checkbox, column, div, groupbox, image, label, listcell, listfooter, listheader, listitem, radio, slider, tab, tabbox, tabpanel, toolbar, toolbarbutton, treecell, treecol, treeitem, window Evento: org.zkoss.zk.ui.event.MouseEvent Indica que el usuario dio un click derecho al componente.
onDoubleClick	button, caption, checkbox, column, div, groupbox, image, label, listcell, listfooter, listheader, listitem, tab, tabpanel, toolbar, treecell, treerow, window Evento: org.zkoss.zk.ui.event.MouseEvent Indica que el usuario le ha dado doble click a un componente.

Fuente: Potix (2009) "ZK The Developer's Guide"

Eventos del teclado

Este tipo de eventos son enviados al elemento window más cercano que tiene un controlador de eventos registrado para manejar el evento especificado.

Esta diseñado para implementar las funcionalidades submit, cancel y accesos directos a otras funcionalidades.

CUADRO # 40
Eventos generados por el teclado.

Nombre	Componente/Descripción
onOK	window, textbox, intbox, longbox, doublebox, decimalbox, datebox, timebox, combobox, bandbox Evento: org.zkoss.zk.ui.event.KeyEvent Indica que el usuario ha presionado la tecla ENTER.
onCancel	window, textbox, intbox, longbox, doublebox, decimalbox, datebox, timebox, combobox, bandbox Evento: org.zkoss.zk.ui.event.KeyEvent Indica que el usuario ha presionado la tecla ESC.
onCtrlKey	window Evento: org.zkoss.zk.ui.event.KeyEvent Indica que el usuario ha presionado una tecla especial, tal como PgUp, Home o una tecla combinada con la tecla Ctrl o Alt.

Fuente: Potix (2009) “ZK The Developer's Guide”

Eventos de entrada

CUADRO # 41
Eventos de entrada.

Nombre	Componente/Descripción
onChange	textbox, datebox, decimalbox, doublebox, intbox, combobox, bandbox Evento: org.zkoss.zk.ui.event.InputEvent Indica que el contenido de un componente fue modificado.
onChangeing	textbox, datebox, decimalbox, doublebox, intbox, combobox, bandbox Evento: org.zkoss.zk.ui.event.InputEvent Indica que el contenido de un componente está siendo modificado.
onSelection	textbox, datebox, decimalbox, doublebox, intbox, combobox, bandbox Evento: org.zkoss.zk.ui.event.SelectionEvent Indica que el usuario está seleccionando una parte del texto de un componente de entrada. Para obtener la posición de inicio y fin del texto seleccionado se usa los métodos getStart y getEnd.
onFocus	textbox, datebox, decimalbox, doublebox, intbox, combobox, bandbox, button, toolbarbutton, checkbox, radio Evento: org.zkoss.zk.ui.event.Event Indica que un componente ha obtenido el enfoque.
onBlur	textbox, datebox, decimalbox, doublebox, intbox, combobox, bandbox, button, toolbarbutton, checkbox, radio Evento: org.zkoss.zk.ui.event.Event Indica que un componente ha perdido el enfoque.

Fuente: Potix (2009) "ZK The Developer's Guide"

Eventos de lista y árboles

CUADRO # 42
Eventos de lista y árboles.

Nombre	Componente/Descripción
onSelect	listbox, tabbox, tab, tree, combobox Evento: org.zkoss.zk.ui.event.SelectEvent Indica que el usuario ha seleccionado uno o más componentes hijos.
onOpen	north, east, west, south, groupbox, treeitem, combobox, bandbox, menupopup, window Evento: org.zkoss.zk.ui.event.OpenEvent Indica que el usuario ha abierto o cerrado un componente. A diferencia de onClose, este evento es sólo una notificación. El cliente envía este evento después de abrir o cerrar el componente.

Fuente: Potix (2009) “ZK The Developer's Guide”

Otros eventos

CUADRO # 43
Otros eventos.

Nombre	Componentes	Descripción
onCreate	Todos	Evento: org.zkoss.zk.ui.event.CreateEvent Indica que un componente es creado cuando la página ZUML se muestra.
onClose	Window, tab, fileupload	Evento: org.zkoss.zk.ui.event.Event Indica que el usuario ha presionado el botón cerrar y el componente se cerró por sí mismo.

Fuente: Potix (2009) “ZK The Developer's Guide”

CUADRO # 44
Otros eventos (continuación).

Nombre	Componentes	Descripción
onMove	window	Evento: org.zkoss.zk.ui.event.MoveEvent Indica que el componente ha sido movido.
onSize	window	Evento: org.zkoss.zk.ui.event.SizeEvent Indica que el componente ha sido redimensionado.
onTimer	timer	Evento: org.zkoss.zk.ui.event.Event Indica que el timer especificado ha disparado un evento. Para conocer cual es el timer, se puede utilizar el método getTarget de la clase Event.
onColSize	columns, listhead, treecols	Evento: org.zkoss.zk.ui.event.ColSizeEvent Notifica al padre de un grupo objetos que el ancho de hijos fue cambiado por el usuario..
onPaging	grid, listbox, paging	Evento: org.zkoss.zk.ui.event.PagingEvent Notifica que una de las páginas de un componente multi-página fue seleccionada por el usuario.
onUpload	fileupload	Evento: org.zkoss.zk.ui.event.UploadEvent Notifica que el o los archivos han sido cargados y la aplicación puede procesarlos con los métodos getMedia o getMedias.
onDrop	Todos	Evento: org.zkoss.zk.ui.event.DropEvent Indica que otro componente fue arrastrado al componente que recibe el evento.
onCheck	checkbox, radio, radiogroup	Evento: org.zkoss.zk.ui.event.CheckEvent Indica que el estado de un componente ha sido cambiado por el usuario.

Fuente: Potix (2009) “ZK The Developer's Guide”

ANEXO # 5



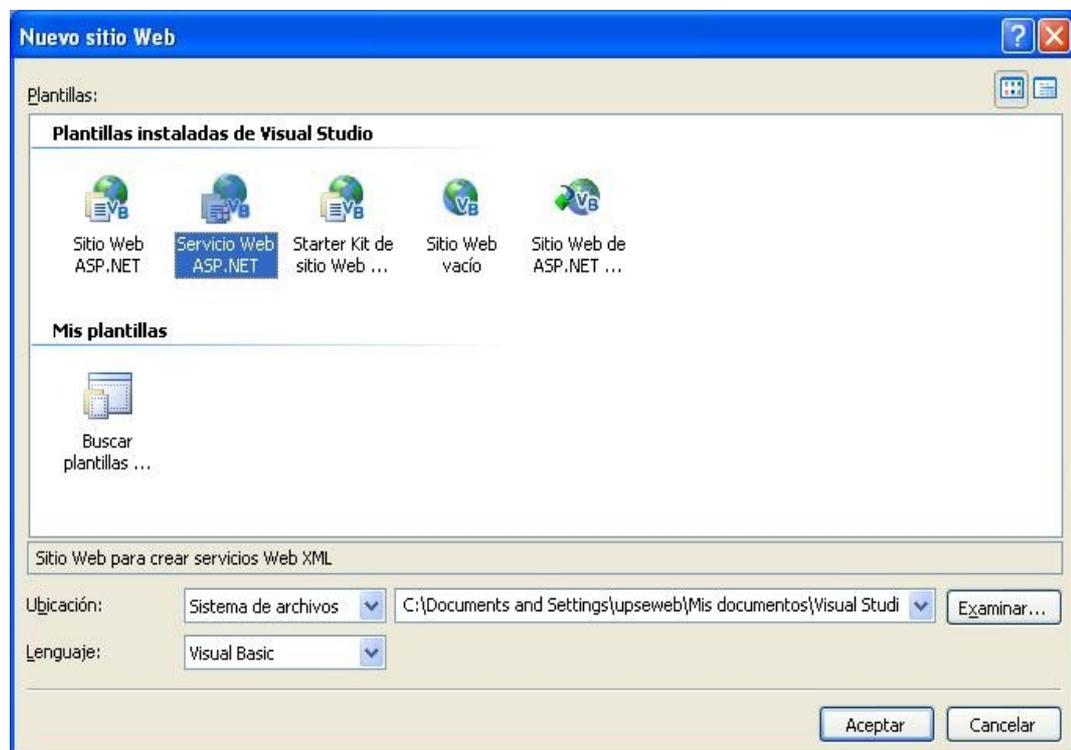
MANUAL DE DESARROLLO DE SERVICIOS WEB

UN PROYECTO DE SERVICIO WEB EN ASP.NET

Un servicio web puede ser añadido como cualquier otro proyecto web de ASP.NET. Visual Studio también provee una plantilla para este tipo de aplicaciones.

Para crear un nuevo proyecto a partir de esta plantilla se debe seleccionar desde el menú Archivo, opción Nuevo Sitio Web y en el diálogo resultante seleccionamos el ítem Servicio Web ASP.NET.

GRÁFICO # 49
Creando un servicio web a partir de la plantilla.



Fuente: Powers, Snell (2005) Microsoft Visual Studio 2005 Unleashed

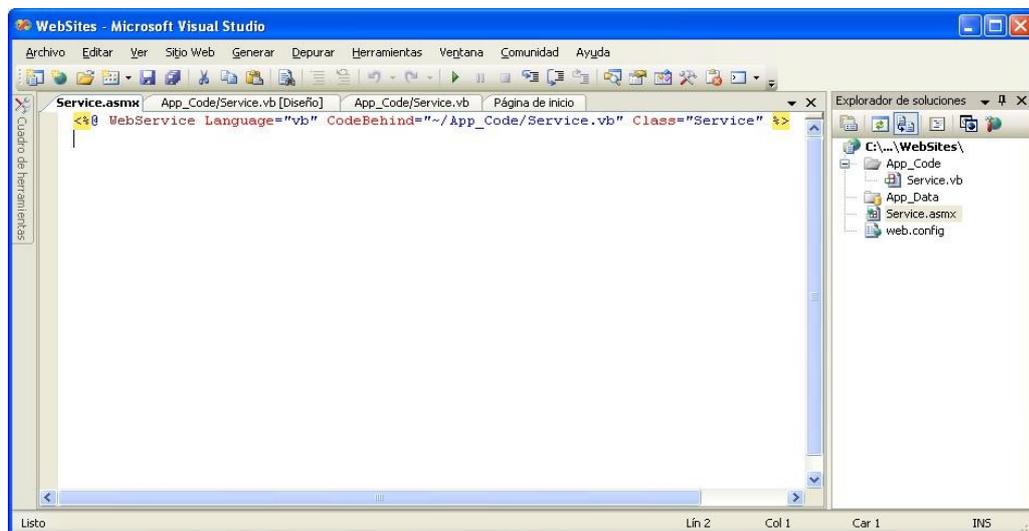
En este diálogo se establece la ubicación del proyecto, ya sea en el sistema de archivos o mediante los protocolos http o ftp. De igual manera, se define qué lenguaje será utilizado para escribir el código del servicio web.

LOS ARCHIVOS DE UN SERVICIO WEB

El nuevo proyecto contiene un archivo que representa al servicio (con extensión asmx), el cual es usado como la URI que sirve para hacer referencia al servicio. El archivo asmx contiene una única directiva, la cual indica mediante el atributo CodeBehind, que el archivo del servicio web es en realidad un puntero al código del servicio.

El archivo de código tiene la extensión vb (por haberse desarrollado con Visual Basic, pero podría variar en dependencia del lenguaje seleccionado para el desarrollo) y se encuentra ubicado dentro del directorio App_Code, en este lugar reside toda la lógica de la aplicación. Esta directiva también utiliza el atributo Class para indicar una clase que está contenida en el mismo archivo de código, esta clase es el servicio web. Los métodos de esta clase son los métodos web.

GRÁFICO # 50
Código del Servicio Web.



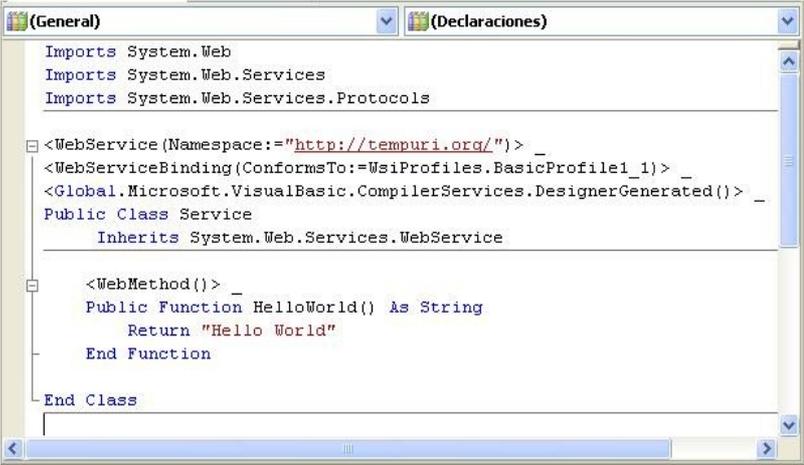
Fuente: Powers, Snell (2005) Microsoft Visual Studio 2005 Unleashed

Si observamos el contenido del archivo de código, notaremos que la estructura es similar a la de una aplicación web, en ella vemos las instrucciones de importación de espacios de nombre que facilitarán la referencia de los diferentes componentes utilizados en el proyecto.

Se distingue también la definición de la clase, la misma que se hereda desde la clase `System.Web.Services.WebService`, y que está precedida del atributo `<WebService>`, que permite a Visual Studio identificar la clase como un servicio web. La herencia desde la clase `System.Web.Services.WebService` es opcional, y bastaría sólo con definir el atributo `<WebService>` para identificar a la clase como un servicio web, pero esta herencia es la que otorga a la clase acceso completo a todas las características de ASP.NET.

Es importante notar que el constructor de la clase no se genera automáticamente al crear un nuevo proyecto, pero puede definirse manualmente como cualquier otro constructor de clase.

GRÁFICO # 51 Código de programación del Servicio Web.



```
Imports System.Web
Imports System.Web.Services
Imports System.Web.Services.Protocols

<WebService(Namespace:="http://tempuri.org/") > _
<WebServiceBinding(ConformsTo:=WsiProfiles.BasicProfile1_1) > _
<Global.Microsoft.VisualBasic.CompilerServices.DesignerGenerated() > _
Public Class Service
    Inherits System.Web.Services.WebService

    <WebMethod() > _
    Public Function HelloWorld() As String
        Return "Hello World"
    End Function
End Class
```

Fuente: Powers, Snell (2005) Microsoft Visual Studio 2005 Unleashed

Finalmente, observamos el método generado por defecto para el servicio, precedido del atributo `<WebMethod>`, el cual indica que se trata de un método web que puede ser invocado una vez que el servicio se encuentre publicado.

DESARROLLANDO UN SERVICIO WEB

Hasta este punto se ha observado como se crean los archivos del servicio web cuando utilizamos la plantilla disponible para este tipo de proyectos. En esta sección se revisará la forma de desarrollar un servicio web personalizado. Se tomará como ejemplo de desarrollo, al servicio web de consulta de historial académico.

DEFINICIÓN DEL SERVICIO WEB PARA CONSULTAS DEL HISTORIAL ACADÉMICO

El servicio web para consultas del Historial Académico servirá de enlace entre el servidor de datos del Sistema Académico Universitario (almacenado en SQL Server) y la aplicación de Consultas (desarrollado en ZK).

Con el fin de mantener compatibilidad con la plataforma en la que se mantienen los datos a consultar, el servicio web deberá estar hospedado en un equipo con un sistema operativo Windows y que ejecute Internet Information Services.

Como la versión de Visual Studio utilizada es 2005, deberá tener instalado mínimo .NET Framework 2.

Los atributos del servicio se detallan a continuación:

Nombre del Servicio Web:	Academico
Nombre del archivo:	Academico.asmx
Espacio de nombres:	upse.academico.XmlWebServices

MÉTODOS WEB

A fin de proveer la información requerida por la aplicación de consulta, el servicio web deberá dar acceso a un conjunto de métodos web, detallados a continuación:

CUADRO # 45
Definición del método ObtenerInfoGeneral.

Nombre:	ObtenerInfoGeneral		
Descripción:	Devuelve la información general sobre un estudiante.		
Parámetros de entrada:	Nombre	Descripción	Tipo de Datos
	p_matricula	Matrícula del Estudiante.	Cadena
Devuelve:	Una instancia de la clase Estudiantes.		

Fuente: Autores

CUADRO # 46
Definición del método ObtenerCarrerasTomadas.

Nombre:	ObtenerCarrerasTomadas		
Descripción:	Proporciona las carreras que el estudiante ha cursado en la universidad.		
Parámetros de entrada:	Nombre	Descripción	Tipo de Datos
	p_cedula	Cédula de Identidad del Estudiante.	Cadena
Devuelve:	Una o varias instancias de la clase CarrerasTomadas.		

Fuente: Autores

CUADRO # 47
Definición del método ObtenerMateriasCursadas.

Nombre:	ObtenerMateriasCursadas		
Descripción:	Proporciona el detalle de las materias cursadas por el estudiante en determinada carrera, así como las notas obtenidas en las mismas.		
Parámetros de entrada:	Nombre	Descripción	Tipo de Datos
	p_matricula	Matrícula del Estudiante.	Cadena
Devuelve:	Una o varias instancias de la clase MateriasCursadas.		

Fuente: Autores

Como se observa en la definición de los métodos web, es necesario, que se especifique un conjunto de clases que serán utilizadas para devolver la información debida. A continuación se observa las características de estas clases.

CUADRO # 48
Definición de la clase MateriasCursadas.

Nombre de la clase: MateriasCursadas		
Detalle de atributos		
Nombre	Descripción	Tipo de Datos
_matricula	Número de matrícula del estudiante.	Cadena
_materia	Nombre de la materia cursada.	Cadena
_primerp	Calificación obtenida en el primer parcial.	Double
_segundop	Calificación obtenida en el segundo parcial.	Double
_tercerp	Calificación obtenida en el tercer parcial.	Double
_cuartop	Calificación obtenida en el cuarto parcial.	Double
_recuperacion	Calificación obtenida por mejoramiento.	Double
_promedio	Promedio final obtenido en la materia.	Double
_nivel	Número de curso-paralelo en el que el estudiante recibió la materia.	Cadena
_semestre_ano	Número de año o semestre.	Cadena
_estado	Indica si la materia fue cursada de forma normal o por arrastre o equiparamiento.	Cadena
_obs	Estado de la materia (Aprobado o Reprobado).	Cadena
Detalle de métodos		
Sobrecarga del constructor para inicializar todos los atributos.		
Setters y Getters estándares para cada uno de los atributos definidos.		

Fuente: Autores

CUADRO # 49
Definición de la clase Estudiantes.

Nombre de la clase: Estudiantes		
Detalle de atributos		
Nombre	Descripción	Tipo de Datos
_cedula	Número de cédula de identidad del estudiante.	Cadena
_matricula	Número de matrícula del estudiante.	Cadena
_nombre	Apellidos y Nombres del estudiante.	Cadena
_promedio	Promedio global obtenido por el estudiante.	Double
Detalle de métodos		
Sobrecarga del constructor para inicializar todos los atributos.		
Setters y Getters estándares para cada uno de los atributos definidos.		

Fuente: Autores

CUADRO # 50
Definición de la clase CarrerasTomadas.

Nombre de la clase: CarrerasTomadas		
Detalle de atributos		
Nombre	Descripción	Tipo de Datos
_matricula	Número de matrícula del estudiante.	Cadena
_facultad	Nombre de la Facultad a la que pertenece la carrera.	Cadena
_escuela	Nombre de la Escuela a la que pertenece la carrera.	Cadena
_modalidad	Modalidad de Estudio.	Cadena
_tomadas	Cantidad de Materias Tomadas.	Entero
_aprobadas	Cantidad de Materias Aprobadas.	Entero
_promedio	Promedio obtenido en la carrera.	Double
Detalle de métodos		
Sobrecarga del constructor para inicializar todos los atributos.		
Setters y Getters estándares para cada uno de los atributos definidos.		

Fuente: Autores

CREANDO EL SERVICIO WEB

Una vez creado el proyecto a partir de la plantilla, procedemos a darle el nombre Academico al proyecto, igual que al archivo asmx.

Es necesario, establecer la cadena de conexión utilizada para acceder a la base de datos, una buena práctica es incluirla dentro del archivo de configuración del servicio (web.config), permitiendo, de esta manera, mantenerla como un recurso externo que puede ser modificado, sin necesidad, de volver a recompilar el proyecto.

GRÁFICO # 52 Fragmento del archivo de configuración donde se declara la cadena de conexión a la base de datos.



```
web.config* | App_Code/MateriasCursadas.vb | App_Code/CarrerasTomadas.vb | App_Code/Estudiantes.vb | App_Code/Service.vb | Página de inicio
<?xml version="1.0"?>
<configuration>
  <appSettings/>
  <connectionStrings>
    <add name="upseConnectionString"
          connectionString="Data Source=SERVER;Initial Catalog=upse;Integrated Security=True"
          providerName="System.Data.SqlClient"/>
  </connectionStrings>
</configuration>
```

Fuente: Autores

La cadena de conexión deberá contar con todos los datos necesarios para el acceso, como el nombre del servidor (o su dirección IP), el nombre de la base de datos, y el nombre de usuario y clave si es requerido.

Lo siguiente sería definir cada una de las clases requeridas para devolver la información obtenida por los métodos web. Cada clase deberá estar declarada en un archivo con extensión vb (variando en dependencia del lenguaje utilizado). A continuación observamos el código de la clase Estudiantes.

```
Imports Microsoft.VisualBasic

Public Class Estudiantes
  Inherits Object
  Private _cedula As String
  Private _matricula As String
  Private _nombre As String
  Private _promedio As Double

```

```

Public Sub New(ByVal cedula As String, ByVal matricula As String, _
    ByVal nombre As String, ByVal promedio As Double) _
    MyBase.New()
    _cedula = cedula
    _matricula = matricula
    _nombre = nombre
    _promedio = promedio
End Sub

Public Sub New()
    MyBase.New()
End Sub

Public Property Cedula() As String
    Get
        Return _cedula
    End Get
    Set(ByVal value As String)
        _cedula = value
    End Set
End Property

Public Property Matricula() As String
    Get
        Return _matricula
    End Get
    Set(ByVal value As String)
        _matricula = value
    End Set
End Property

Public Property Nombre() As String
    Get
        Return _nombre
    End Get
    Set(ByVal value As String)
        _nombre = value
    End Set
End Property

Public Property Promedio() As Double
    Get
        Return _promedio
    End Get
    Set(ByVal value As Double)
        _promedio = value
    End Set
End Property
End Class

```

Observando la estructura de la clase, se identifica a los atributos descritos en la definición de la clase, así como un constructor sin parámetros, un constructor con todos los atributos y los métodos set y get para cada uno de estos atributos.

Las clases CarrerasTomadas y MateriasCursadas siguen el mismo patrón observado. Una vez definidas todas las clases ya podemos centrarnos en el código de nuestro servicio web. Después de eliminar el método creado por defecto (HelloWorld), procedemos a hacer los cambios, de acuerdo, a los requerimientos

del servicio, para nuestro caso particular, necesitaremos importar los espacios de nombre System.Xml.Serialization y System.Data.SqlClient. El primero permite que los métodos de nuestro servicio puedan retornar un conjunto de objetos de determinada clase, por defecto un método sólo puede retornar un valor escalar, una instancia de un objeto o conjunto de datos sin un formato establecido.

El espacio de nombres System.Data.SqlClient permitirá acceder a los elementos utilizados para el acceso a la base de datos.

```
Imports System.Web
Imports System.Web.Services
Imports System.Web.Services.Protocols
Imports System.Xml.Serialization
Imports System.Data.SqlClient
```

EL ATRIBUTO DE CLASE WEBSERVICE

Es necesario, definir el espacio de nombres en el que se incluirá nuestro servicio web, lo cual le permitirá ser consumido desde un cliente. Esto se hace a través del atributo WebService.

```
<WebService (Namespace:="http://upse/academico/XmlWebServices/") > _
<WebServiceBinding (ConformsTo:=WsiProfiles.BasicProfile1_1) > _
<Global.Microsoft.VisualBasic.CompilerServices.DesignerGenerated() > _
```

Además de permitir definir el espacio de nombres, este atributo puede ser utilizado para proveer información adicional sobre el servicio web que estemos desarrollando. A continuación se detallan los elementos descriptivos que pueden ser utilizados en este atributo:

- **Description:** Usado para definir una descripción del servicio. Esta descripción será entregada a los consumidores del servicio.
- **Namespace:** Utilizado para declarar el espacio de nombres del servicio.
- **Name:** Permite establecer otro nombre para el servicio web, diferente al que Visual Basic le da en base al nombre del proyecto.

LA CLASE WEBSERVICE

La clase `WebService` representa la clase base para los servicios web de .NET. Esta no debe ser confundida con el atributo `WebService`. El uso de esta clase deriva en el uso de los objetos comunes de ASP.NET (`Session`, `Application`, `Context`, entre otros). Aunque su uso no es obligatorio, Visual Studio fuerza esta herencia cuando se crea un nuevo servicio web, pudiendo ser quitada si no existe la intención de utilizar objetos ASP dentro del servicio.

```
Public Class Academico
    Inherits System.Web.Services.WebService

    Dim gConnString As String

    Public Sub New()
        MyBase.New()

        gConnString = System.Configuration.ConfigurationManager._
            ConnectionStrings("upseConnectionString").ConnectionString

    End Sub
```

Declaramos una variable `gConnString` que luego es utilizada en el constructor de la clase para obtener la cadena de conexión almacenada en el archivo de configuración.

EL ATRIBUTO DE CLASE WEBMETHOD

El atributo `WebMethod` es utilizado para indicar que un método incluido en el servicio va a ser puesto a disposición una vez que el servicio web sea publicado. Esta declaración es obligatoria para todos los métodos que desee hacer disponibles a través del servicio web. Hay un número de parámetros que pueden ser definidos al crear un método web. Estos parámetros controlan la forma en que opera el método.

A continuación de detallan los parámetros soportados por el atributo `WebMethod`.

- **Description:** Permite asignar una descripción para el método web.

- **EnableSession:** Permite indicar si un método web requiere un inicio de sesión previo a su utilización.
- **CacheDuration:** Especifica el tiempo que el resultado devuelto por el método será almacenado en caché. El valor de este parámetro está dado en segundos.
- **MessageName:** Define un alias para el método. Esta propiedad es muy utilizada cuando se necesita sobrecargar métodos, lo cual es soportado por .NET pero no por los servicios web. De esta forma los métodos pueden sobrecargarse y luego aplicar un nombre único a través del uso de este parámetro.
- **BufferResponse:** Se utiliza para determinar si la totalidad de la respuesta de un método se almacenará en la memoria del servidor antes de enviarla al cliente. Por defecto, este parámetro tiene el valor true. Si el parámetro se define con false, la respuesta se almacenará en bloques de 16KB, y la respuesta se enviará al cliente en bloques de este tamaño.
- **transactionOption:** Permite definir que el servicio web deberá ser la raíz de una transacción.

EL MÉTODO WEB OBTENERINFOGENERAL

Este método devuelve la información del estudiante a través de una instancia de la clase Estudiantes. La fuente de donde se toma la información es el procedimiento almacenado Nt_Sp_Todas_Notas_Estudiantes, el cual requiere de tres parámetros. El primer parámetro representa el número de cédula o matrícula, en este caso las búsquedas siempre se realizan por número de matrícula.

El segundo parámetro puede recibir el valor AÑO o MATRICULA y establece la modalidad bajo la cual se realiza la consulta. Y el último parámetro puede recibir el valor de 0 si la búsqueda se hace por número de matrícula o 1 si la búsqueda se hace por cédula, obviamente siempre se le va a asignar el valor de 0.

Debido a que el estudiante puede tener un registro bajo la modalidad por semestre o por año, el método realiza primero la búsqueda bajo la modalidad por Año y si no encuentra información realiza una nueva búsqueda bajo la modalidad por semestre. Finalmente, es importante indicar que este procedimiento devuelve más de un conjunto de datos, siendo el primer conjunto el que contiene la información del estudiante. A continuación se observa el código de este método:

```

<WebMethod()> _
Public Function ObtenerInfoGeneral(ByVal p_matricula As String) As Estudiantes
    Dim vCon As New SqlConnection
    Dim vSqlCommand As New SqlCommand
    Dim vPar As New SqlParameter
    Dim vDatos As SqlDataReader
    Dim vRetorna As Estudiantes
    Dim vExiste As Boolean
    '
    vRetorna = Nothing
    vCon.ConnectionString = gConnString
    vCon.Open()
    vSqlCommand.Connection = vCon
    ' Busca datos del estudiante en la modalidad de estudios por año.
    vSqlCommand.CommandText = "exec Nt_Sp_Todas_Notas_Estudiantes " & _
        "@Identificador, 'AÑO', 0"
    vSqlCommand.CommandType = Data.CommandType.Text
    '
    vPar.Direction = Data.ParameterDirection.Input
    vPar.ParameterName = "Identificador"
    vPar.DbType = Data.DbType.String
    vPar.Value = p_matricula
    vSqlCommand.Parameters.Add(vPar)
    '
    vDatos = vSqlCommand.ExecuteReader
    vExiste = vDatos.HasRows
    If vDatos.HasRows Then
        ' Si la consulta devuelve datos
        vDatos.Read()
        ' El primer conjunto de registros devuelto por el procedimiento son
        ' los datos del estudiante.
        vRetorna = New Estudiantes(IIf(IsDBNull(vDatos("CEDULA")), " ", _
            vDatos("CEDULA")), IIf(IsDBNull(vDatos("MATRICULA")), " ", _
            vDatos("MATRICULA")), IIf(IsDBNull(vDatos("ESTUDIANTE")), _
            " ", vDatos("ESTUDIANTE")), 0)
    End If
    ' Si no encuentra datos por año intenta buscando por semestre.
    If Not vExiste Then
        vSqlCommand.CommandText = "exec Nt_Sp_Todas_Notas_Estudiantes " & _
            "@Identificador, 'SEMESTRE', 0"
        vSqlCommand.CommandType = Data.CommandType.Text
        vDatos = vSqlCommand.ExecuteReader
        If vDatos.HasRows Then
            ' Si la consulta devuelve datos
            vDatos.Read()
            ' El primer conjunto de registros devuelto por el procedimiento
            ' son los datos del estudiante.
            vRetorna = New Estudiantes(IIf(IsDBNull(vDatos("CEDULA")), " ", _
                vDatos("CEDULA")), IIf(IsDBNull(vDatos("MATRICULA")), _
                " ", vDatos("MATRICULA")), IIf(IsDBNull(vDatos("ESTUDIANTE")), _
                " ", vDatos("ESTUDIANTE")), 0)
        End If
        vDatos.Close()
    End If

```

```
vSqlCommand.Dispose()  
vCon.Close()  
vCon.Dispose()  
Return vRetorna  
End Function
```

EL MÉTODO WEB OBTENERCARRERASTOMADAS

Este método devuelve la o las carreras que un estudiante ha seguido en la Universidad. Esta información se retorna a través de un arreglo de instancias de la clase CarrerasTomadas. Para que el método sea capaz de devolver un arreglo de instancias de una clase, es necesario, incluir el atributo de clase XmlInclude indicando la clase que se aplica a cada uno de los elementos del arreglo devuelto.

El método obtiene los datos al realizar una llamada al procedimiento almacenado Nt_Sp_Estudiante_Registro_Carreras, al cual le envía un parámetro, cuyo valor es el número de cédula del estudiante.

Como un dato a resaltar, se debe tomar en cuenta que ninguno de los valores que se devuelvan por el método web deberá ser Nulo, debido a que se produciría un error por cuanto a nivel del servicio web no existe forma de tratar un valor de este tipo.

Debido a esto, en todos los procedimientos se hace una verificación de los datos obtenidos y en caso de ser un valor nulo, lo reemplaza por un espacio en blanco, si el tipo de dato es una cadena, y por el valor de -1 si se trata de un valor numérico.

```
<WebMethod(), XmlInclude(GetType(CarrerasTomadas))>  
Public Function ObtenerCarrerasTomadas(ByVal p_cedula As String) As ArrayList  
    Dim vCon As New SqlConnection  
    Dim vSqlCommand As New SqlCommand  
    Dim vPar As New SqlParameter  
    Dim vDatos As SqlDataReader  
    Dim vRetorna As ArrayList
```

```

'
vRetorna = New ArrayList
vCon.ConnectionString = gConnString
vCon.Open()
vSqlCommand.Connection = vCon
'
' Obtiene las carreras registradas.
vSqlCommand.CommandText = "exec Nt_Sp_Estudiante_Registro_Carreras " & _
                            " @Identificador"
vSqlCommand.CommandType = Data.CommandType.Text
'
vPar.Direction = Data.ParameterDirection.Input
vPar.ParameterName = "Identificador"
vPar.DbType = Data.DbType.String
vPar.Value = p_cedula
vSqlCommand.Parameters.Add(vPar)
'
vDatos = vSqlCommand.ExecuteReader
'
If vDatos.HasRows Then
' Si la consulta devuelve datos
While vDatos.Read()
' Obtiene el detalle de cada carrera cursada
Dim vCon1 As New SqlConnection
Dim vSqlCommand1 As New SqlCommand
Dim vPar1 As New SqlParameter
Dim vDatos1 As SqlDataReader
vCon1.ConnectionString = gConnString
vCon1.Open()
vSqlCommand1.Connection = vCon1
'
' Obtiene las carreras registradas por semestre.
vSqlCommand1.CommandText="exec Nt_Sp_Todas_Notas_Estudiantes " & _
                            " @Matricula, 'SEMESTRE', 0"
vSqlCommand1.CommandType = Data.CommandType.Text

vPar1.Direction = Data.ParameterDirection.Input
vPar1.ParameterName = "Matricula"
vPar1.DbType = Data.DbType.String
vPar1.Value = vDatos("MATRICULA")
vSqlCommand1.Parameters.Add(vPar1)

vDatos1 = vSqlCommand1.ExecuteReader
If vDatos1.HasRows Then
' Si la consulta devuelve datos
vDatos1.Read()
' Devuelve la informacion de la carrera
vRetorna.Add(New CarrerasTomadas (IIf(IsDBNull(vDatos1("MATRICULA")), _
    " ", vDatos1("MATRICULA")), IIf(IsDBNull(vDatos1("FACULTAD")), _
    " ", vDatos1("FACULTAD")), IIf(IsDBNull(vDatos1("ESCUELA")), _
    " ", vDatos1("ESCUELA")), IIf(IsDBNull(vDatos1("MODALIDAD")), _
    " ", vDatos1("MODALIDAD")), 0, 0, 0))
End If
vDatos1.Close()
' Obtiene las carreras registradas por año.
vSqlCommand1.CommandText = "exec Nt_Sp_Todas_Notas_Estudiantes " & _
                            " @Matricula, 'AÑO', 0"
vSqlCommand1.CommandType = Data.CommandType.Text
'
vDatos1 = vSqlCommand1.ExecuteReader
'
If vDatos1.HasRows Then
' Si la consulta devuelve datos
vDatos1.Read()
' Devuelve la informacion de la carrera
vRetorna.Add(New CarrerasTomadas (IIf(IsDBNull(vDatos1("MATRICULA")), _
    " ", vDatos1("MATRICULA")), IIf(IsDBNull(vDatos1("FACULTAD")), _
    " ", vDatos1("FACULTAD")), IIf(IsDBNull(vDatos1("ESCUELA")), _
    " ", vDatos1("ESCUELA")), IIf(IsDBNull(vDatos1("MODALIDAD")), _
    " ", vDatos1("MODALIDAD")), 0, 0, 0))

```

```

        End If
        vDatos1.Close()
        vSqlCommand1.Dispose()
        vCon1.Close()
        vCon1.Dispose()
    End While
End If

vDatos.Close()
vSqlCommand.Dispose()
vCon.Close()
vCon.Dispose()

Return vRetorna
End Function

```

EL MÉTODO WEB OBTENERMATERIASCURSADAS

Este método devuelve las materias que un estudiante recibió en una carrera, así como las notas parciales y promedios obtenidos en las mismas. Esta información se retorna a través de un arreglo de instancias de la clase MateriasCursadas. El método obtiene los datos al realizar una llamada al procedimiento almacenado Nt_Sp_Todas_Notas_Estudiantes, el cual fue revisado en el Método Web ObtenerInfoGeneral.

```

<WebMethod(), XmlInclude(GetType(MateriasCursadas))> _
Public Function ObtenerMateriasCursadas(ByVal p_matricula As String) As _
                                                                    ArrayList

    Dim vCon As New SqlConnection
    Dim vSqlCommand As New SqlCommand
    Dim vPar As New SqlParameter
    Dim vDatos As SqlDataReader
    Dim vRetorna As ArrayList
    '
    vRetorna = New ArrayList
    vCon.ConnectionString = gConnString
    vCon.Open()
    vSqlCommand.Connection = vCon
    '
    ' Busca datos del estudiante en la modalidad de estudios por semestre.
    vSqlCommand.CommandText = "exec Nt_Sp_Todas_Notas_Estudiantes " & _
        " @Identificador, 'SEMESTRE', 0"
    vSqlCommand.CommandType = Data.CommandType.Text
    vPar.Direction = Data.ParameterDirection.Input
    vPar.ParameterName = "Identificador"
    vPar.DbType = Data.DbType.String
    vPar.Value = p_matricula
    vSqlCommand.Parameters.Add(vPar)

    vDatos = vSqlCommand.ExecuteReader
    ' Obtiene el siguiente resultset en donde esta la informacion de las
    ' materias tomadas
    vDatos.NextResult()

```

```

If vDatos.HasRows Then
    ' Si la consulta devuelve datos de las materias cursadas
    While vDatos.Read()
        ' Evita el envio de valores nulos asignando cadenas con un
        ' espacio en blanco o el valor de -1 de acuerdo al tipo de datos
        vRetorna.Add(New MateriasCursadas(p_matricula, _
            IIf(IsDBNull(vDatos("MATERIA")), " ", vDatos("MATERIA")), _
            IIf(IsDBNull(vDatos("PAR1")), -1, vDatos("PAR1")), _
            IIf(IsDBNull(vDatos("PAR2")), -1, vDatos("PAR2")), _
            IIf(IsDBNull(vDatos("PAR3")), -1, vDatos("PAR3")), _
            IIf(IsDBNull(vDatos("PAR4")), -1, vDatos("PAR4")), _
            IIf(IsDBNull(vDatos("RECP")), -1, vDatos("RECP")), _
            IIf(IsDBNull(vDatos("PROM")), -1, vDatos("PROM")), _
            IIf(IsDBNull(vDatos("NIVEL")), " ", vDatos("NIVEL")), _
            IIf(IsDBNull(vDatos("SEMESTRE_AÑO")), " ", _
                vDatos("SEMESTRE_AÑO")), _
            IIf(IsDBNull(vDatos("ESTADO")), " ", vDatos("ESTADO")), _
            IIf(IsDBNull(vDatos("OBS")), " ", vDatos("OBS"))))
    End While
End If
vDatos.Close()

' Busca datos del estudiante en la modalidad de estudios por año.
vSqlCommand.CommandText = "exec Nt_Sp_Todas_Notas_Estudiantes " &
    " @Identificador, 'AÑO', 0"
vSqlCommand.CommandType = Data.CommandType.Text
vDatos = vSqlCommand.ExecuteReader
' Obtiene el siguiente resultset en donde esta la informacion de las
' materias tomadas
vDatos.NextResult()

If vDatos.HasRows Then
    ' Si la consulta devuelve datos de las materias cursadas
    While vDatos.Read()
        ' Evita el envio de valores nulos asignando cadenas con un
        ' espacio en blanco o el valor de -1 de acuerdo al tipo de datos
        vRetorna.Add(New MateriasCursadas(p_matricula, _
            IIf(IsDBNull(vDatos("MATERIA")), " ", vDatos("MATERIA")), _
            IIf(IsDBNull(vDatos("PAR1")), -1, vDatos("PAR1")), _
            IIf(IsDBNull(vDatos("PAR2")), -1, vDatos("PAR2")), _
            IIf(IsDBNull(vDatos("PAR3")), -1, vDatos("PAR3")), _
            IIf(IsDBNull(vDatos("PAR4")), -1, vDatos("PAR4")), _
            IIf(IsDBNull(vDatos("RECP")), -1, vDatos("RECP")), _
            IIf(IsDBNull(vDatos("PROM")), -1, vDatos("PROM")), _
            IIf(IsDBNull(vDatos("NIVEL")), " ", vDatos("NIVEL")), _
            IIf(IsDBNull(vDatos("SEMESTRE_AÑO")), " ", _
                vDatos("SEMESTRE_AÑO")), _
            IIf(IsDBNull(vDatos("ESTADO")), " ", vDatos("ESTADO")), _
            IIf(IsDBNull(vDatos("OBS")), " ", vDatos("OBS"))))
    End While
End If
vDatos.Close()
vSqlCommand.Dispose()
vCon.Close()
vCon.Dispose()

Return vRetorna
End Function
End Class#

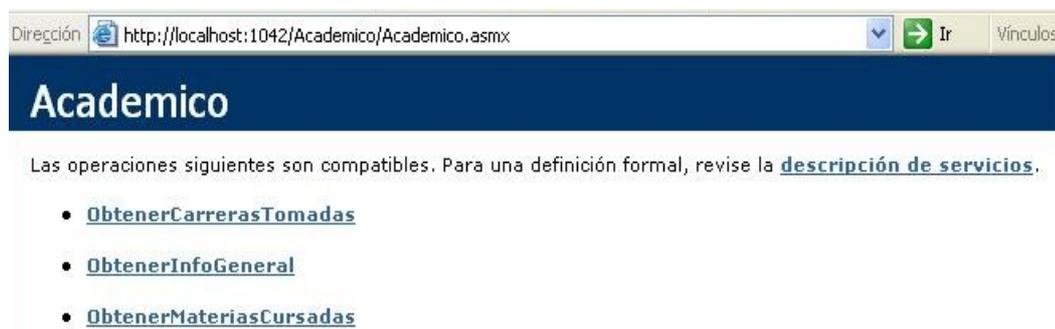
```

ACCEDIENDO E INVOCANDO EL SERVICIO WEB

Visual Studio y .NET Framework permiten visualizar un servicio web en un navegador. Esta capacidad es muy útil tanto para realizar pruebas como para descubrir la manera en que trabajan los servicios (la información que requieren y la que devuelven). Para acceder a un servicio web, este debe ser previamente compilado en Visual Studio y luego accedido a través del archivo asmx en el navegador.

A continuación se muestra la manera en que se accede al servicio desde el navegador:

GRÁFICO # 53
Ejecutando el servicio en un navegador.



Fuente: Autores

En el gráfico observamos que al acceder al servicio se muestra la lista de todos los métodos web incluidos en el mismo.

En caso de que se hubiera definido una descripción, esta se mostrará debajo del nombre del método. Para revisar la definición formal del servicio web tal y como se define en WSDL, puede seleccionarse el enlace Service Description.

Cuando esto se hace, el parámetro WSDL es pasado a la llamada del archivo asmx. Esto le indica a .NET que debe generar el descriptor WSDL del servicio.

VISUALIZANDO LOS MÉTODOS WEB

Al acceder a uno de los métodos web, .NET genera un formulario web en el cual se prueba dicho método web.

Este formulario es accedido desde el archivo asmx acompañado de un parámetro llamado op que indica el nombre método web.

En el formulario se permite asignar el valor que tendrá el o los parámetros utilizados por el método para realizar la consulta requerida.

Si se desea saber el tipo de datos requerido por un parámetro, basta con revisar el descriptor WSDL.

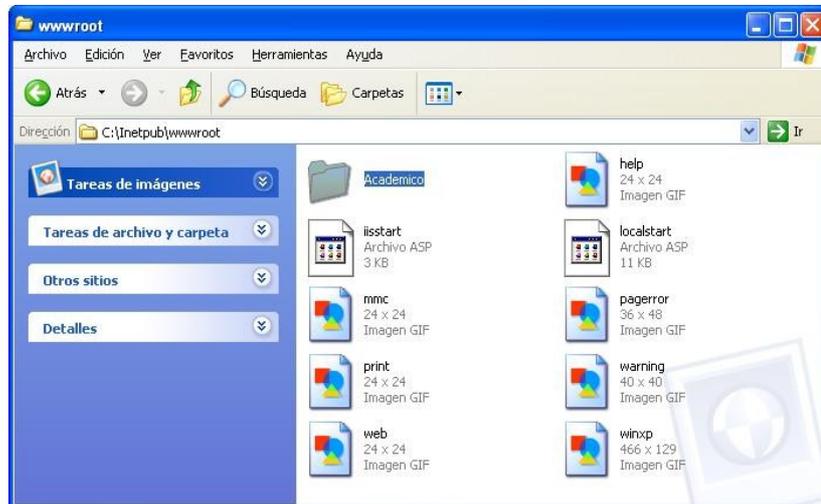
Una vez que establecemos el valor e invocamos el método, este nos da un resultado en formato XML, que contiene la información estructurada de acuerdo a lo que se esté devolviendo.

```
<?xml version="1.0" encoding="utf-8"?>
<Estudiantes xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
             xmlns:xsd="http://www.w3.org/2001/XMLSchema"
             xmlns="http://upse/academico/XmlWebServices/">
  <Cedula>0919294314</Cedula>
  <Matricula>2004120372 </Matricula>
  <Nombre>Jimmy Stanley Catuto Roca </Nombre>
  <Promedio>0</Promedio>
</Estudiantes>
```

PUBLICANDO EL SERVICIO WEB

Una vez que el proyecto está terminado, la publicación del servicio es muy sencilla. Previo a una compilación, se copia una carpeta ubicada dentro de la carpeta Precompiled Web, en la carpeta de nuestro proyecto. Esta carpeta debe ser ubicada en el directorio de publicación web de Internet Information Services.

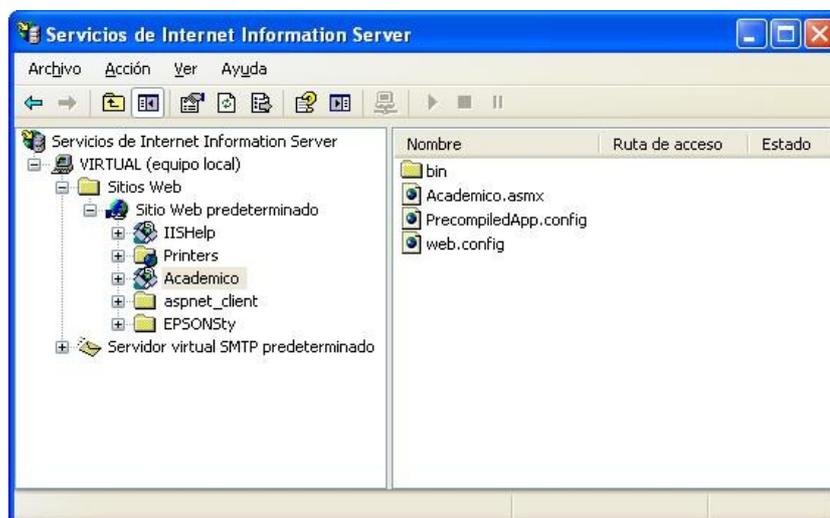
GRÁFICO # 54
Publicación del servicio web.



Fuente: Autores

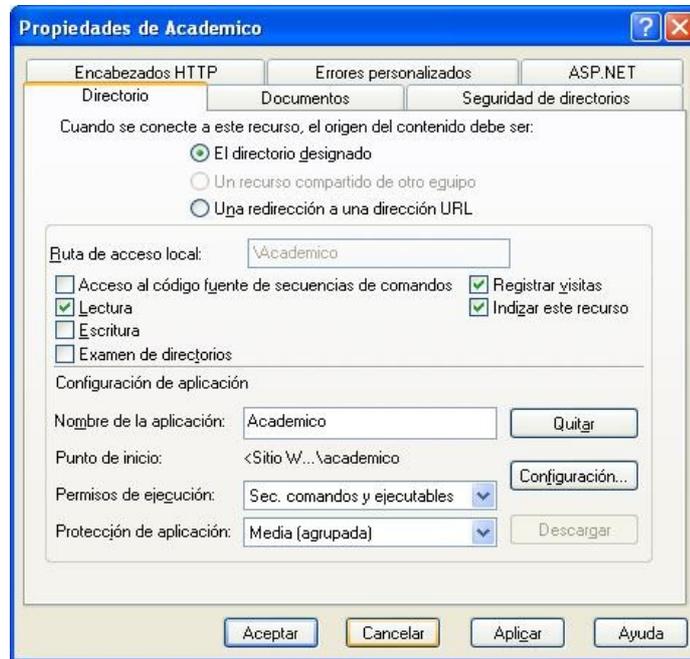
Adicional a esto, es necesario, establecer ciertas configuraciones para el servicio web desde el Administrador de Internet Information Services, ubicado en las Herramientas Administrativas de Windows. Lo primero que se debe hacer es establecer la carpeta que acabamos de copiar como una aplicación web. También se debe verificar que tenga acceso a ejecutar secuencias de comandos.

GRÁFICO # 55
Configuración del servicio en Internet Information Services.



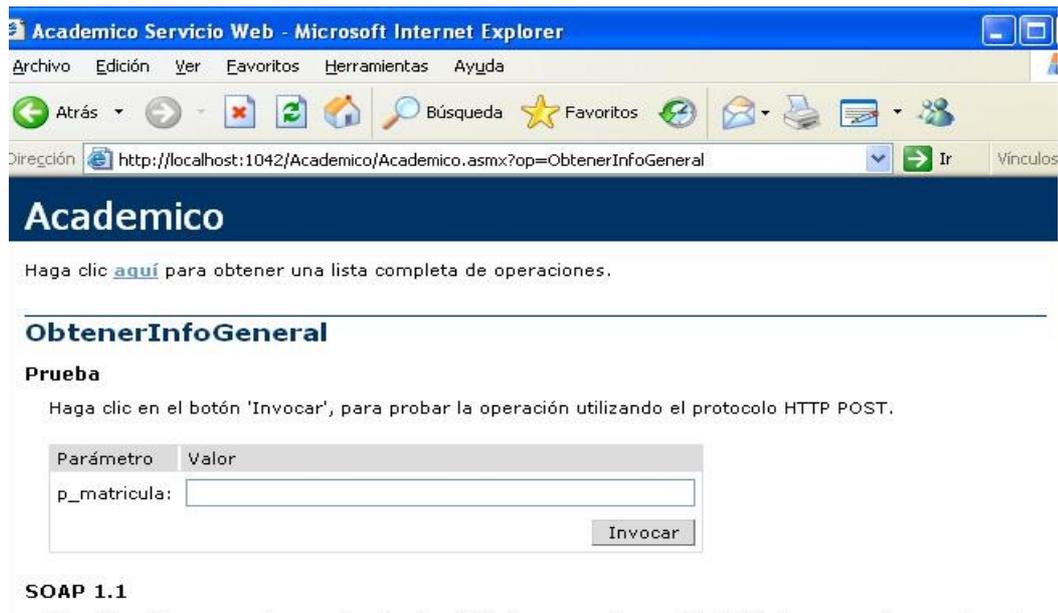
Fuente: Autores

GRÁFICO # 56
Propiedades del servicio web.



Fuente: Autores

GRÁFICO # 57
Accediendo a un método web.



Fuente: Autores

Adicionalmente se debe verificar en la ficha ASP.NET que se haya seleccionado la versión correcta de .NET framework (mínimo la versión 2), ya que de no ser así, el servicio no podría ser utilizado.

GRÁFICO # 58
Selección de versión de .NET Framework.



Fuente: Autores

CONSUMIENDO EL SERVICIO WEB

Una vez que el servicio está publicado y listo para ser consumido, puede ser accedido por cualquier aplicación sin importar la plataforma operativa en que se encuentre, ni el lenguaje en que haya sido desarrollado.

Para nuestra plataforma, la aplicación se encuentra desarrollada en ZK, y consume el servicio web gracias a la integración realizada con Apache Axis, que es una implementación SOAP OpenSource basada en Java.

Axis viene disponible en la versión de Eclipse orientada al desarrollo Java EE, lo cual facilita aún más su utilización en un proyecto ZK.

DESARROLLO DE LA APLICACIÓN PARA CONSULTA DEL HISTORIAL ACADÉMICO

Esta aplicación consta de una interface que solicita el ingreso de la cédula de identidad y el número de matrícula del estudiante, básicamente el hecho de solicitar ambos datos a la vez se realiza con el fin de verificar que es el propio estudiante el que está realizando la consulta en el portal.

Con estos datos la aplicación hace uso de los métodos definidos en el servicio web para verificar la existencia del estudiante y otorgar el acceso a la pantalla de consulta de datos.

La interfaz de consulta toma el número de cédula del estudiante, y vuelve a invocar al servicio web para obtener las carreras que hubiere cursado el estudiante en la Universidad, por cada carrera genera un tabulador en el cual se incluye el historial académico correspondiente a la carrera, permitiendo así realizar consultas de estudiantes que tengan más de un número de matrícula asignado.

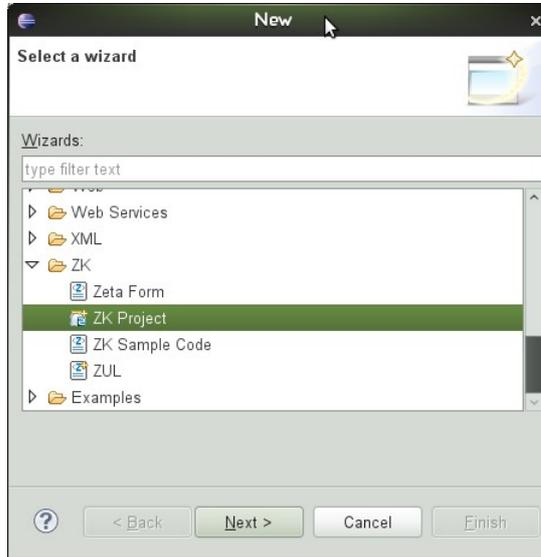
CREACIÓN DEL PROYECTO EN ZK

El siguiente procedimiento fue realizado en un equipo con la distribución OpenSUSE 11.2 instalada, y en el cual estaba configurado el ambiente de trabajo para ZK.

En el IDE Eclipse vamos a proceder a crear un nuevo proyecto a través del menú File, opción New, opción Other.

En el diálogo resultante desplegamos el nodo con nombre ZK, y seleccionamos el elemento ZK Project, hecho esto continuamos con el siguiente paso.

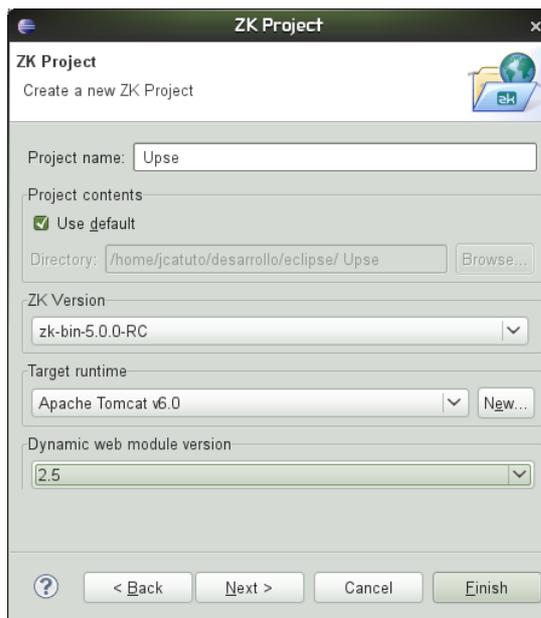
GRÁFICO # 59
Seleccionando el tipo de proyecto.



Fuente: Autores

Asignamos un nombre al proyecto, seleccionamos la versión del framework ZK, el servidor en el que se va a ejecutar el proyecto y la versión del módulo de web dinámico.

GRÁFICO # 60
Seleccionando parámetros iniciales.



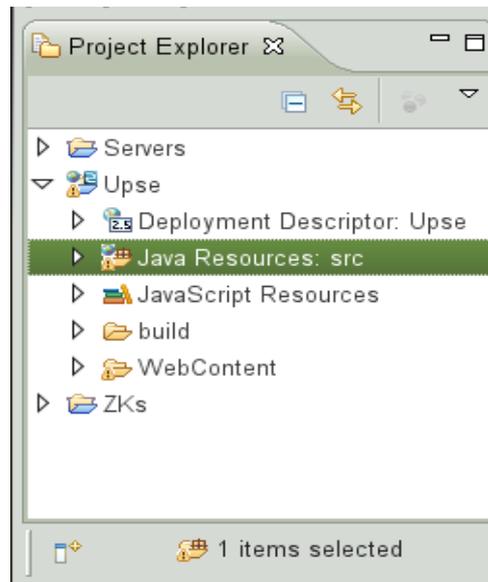
Fuente: Autores

Para los siguientes valores solicitados por el asistente, vamos a dejar los que se asignan por defecto.

Eclipse genera una estructura inicial para nuestro proyecto, en la cual vamos a proceder a crear mediante Axis las clases que permitirán consumir el servicio web.

En la estructura del proyecto ubicamos un nodo llamado Java Resources: src, que es donde ubicaremos al cliente del web service.

GRÁFICO # 61
Estructura inicial del proyecto ZK.



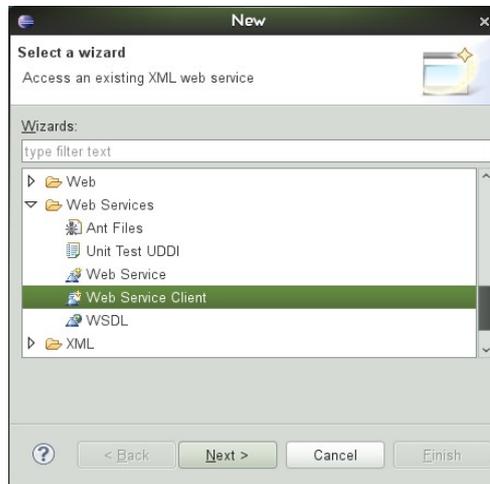
Fuente: Autores

DEFINICIÓN DEL CLIENTE DEL SERVICIO WEB

Procedemos a invocar al asistente de creación del cliente del servicio web, a través del menú contextual desplegado al dar click derecho sobre el nodo Java Resources: src, opción New, opción Other.

En el diálogo resultante desplegamos el nodo Web Services y luego seleccionamos la opción Web Service Client.

GRÁFICO # 62
Asistente de creación del cliente del servicio web.



Fuente: Autores

El siguiente paso permite ubicar el servicio web que se va a consumir, para esto debemos indicar el URI del servicio publicado incluyendo el parámetro WSDL.

GRÁFICO # 63
Selección del URI del servicio web publicado.



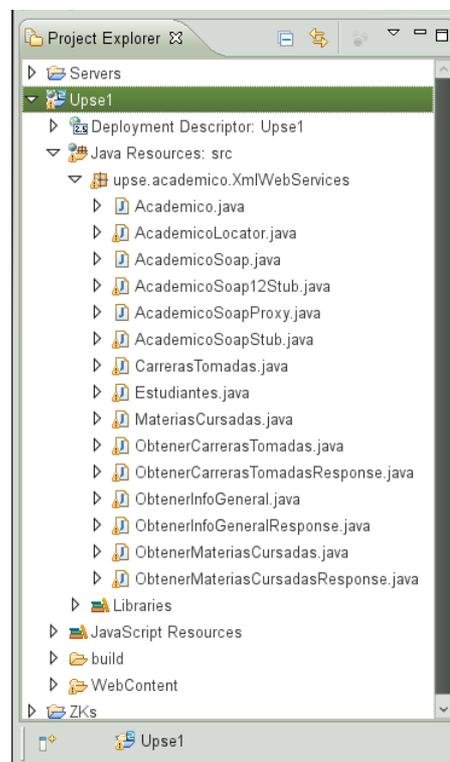
Fuente: Autores

Finalmente, confirmamos la ruta en donde se almacenarán todas las clases generadas por Axis. Al finalizar el asistente, automáticamente Axis se encargará de generar todos los recursos necesarios para acceder al servicio web. Al observar las clases generadas por Axis notamos que todas corresponden a los elementos contenidos en el descriptor del servicio web.

Una clase que genera especial interés es `AcademicoLocator.java`, ya que es aquí donde se guarda el URI del servicio web, por tanto, si en algún momento el servicio es movido a otro servidor, basta con reemplazar en esta clase todas las referencias al servidor antiguo por el nuevo servidor, y luego volver a generar la aplicación y publicarla.

En este momento nuestro proyecto cuenta con el soporte necesario para consumir el servicio web, restando solamente desarrollar las interfaces de la aplicación.

GRÁFICO # 64 **Clases generadas por Axis para el cliente.**



Fuente: Autores

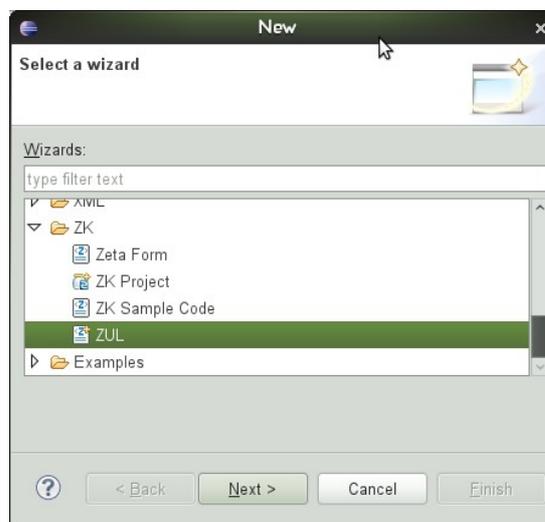
DESARROLLO DE LAS INTERFACES

La primera interface a desarrollar será la de acceso a la aplicación. Creamos un nuevo documento ubicándonos en el nodo WebContent de la estructura principal del proyecto, luego desde el menú contextual invocado en este punto, seleccionamos la opción New, opción Other.

En el diálogo resultante desplegamos el nodo ZK y seleccionamos la opción ZUL.

Una vez definido el nombre del archivo damos por finalizado el asistente, con lo cual el documento es añadido a la carpeta WebContent.

GRÁFICO # 65
Creando un documento zul.



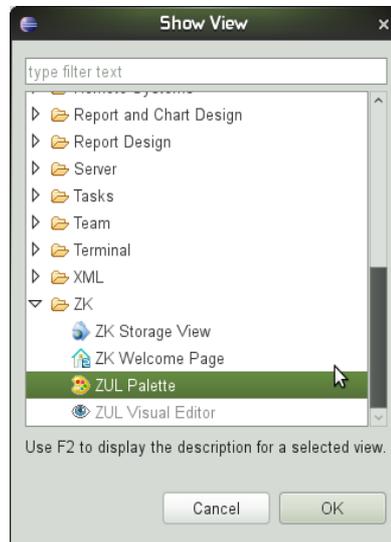
Fuente: Autores

Procedemos entonces a codificar el contenido del documento, ya sea a través del language de marcado ZUML, o con el uso de la paleta de elementos ZUL que permite ir añadiendo los componentes con ayuda del Editor Visual de ZUL.

Si alguna de estas herramientas no se muestra en el ambiente de desarrollo, basta con agregarlas a través del menú Window, opción Show View, opción Other.

GRÁFICO # 66

Agregando la paleta de elementos ZUL.



Fuente: Autores

En el diálogo resultante desplegamos el nodo ZK y seleccionamos una a la vez las herramientas que necesitamos que se muestren en el ambiente de desarrollo. A continuación observamos el código final del documento zul correspondiente a la interfaz de acceso a la consulta del Historial Académico.

```
<?page title="Control de Acceso..." contentType="text/html;charset=UTF-8"?>
<zk>
<!-- Importa funcionalidad de otros archivos con código -->
<zscript language="java" src="seguridad.zs"/>
<!-- Define la ventana principal -->
<window id="loginwin"
  title="Ingresa tus datos"
  border="normal"
  width="560px"
  position="center"
  mode="modal" use="upse.academico.controladores.EstudiantesController">
  <hbox>
    <image src="Imagenes/logo_upse.png" width="200px"/>
    <vbox>
    <separator/>
    <!-- Encabezado -->
    <groupbox style="text-align: center" width="300px">
      <separator/>
      <label value="Consulta del Historial Académico" style="color: #2a4262;
        font-weight: bold;" />
      <separator/>
    </groupbox>
    <separator/>
    <!-- Formulario con datos de acceso -->
    <groupbox>
      <caption style="color: #2a4262; font-weight: bold;" >Acceso</caption>
      <separator/>
      <label value="Cédula de Identidad: " style="color: #2a4262;
        font-weight: bold;" />
      <textbox id="txtCedula" width="150px" focus="true"
        onOK="txtMatricula.setFocus(true)"/>
    </groupbox>
  </hbox>
</window>
```

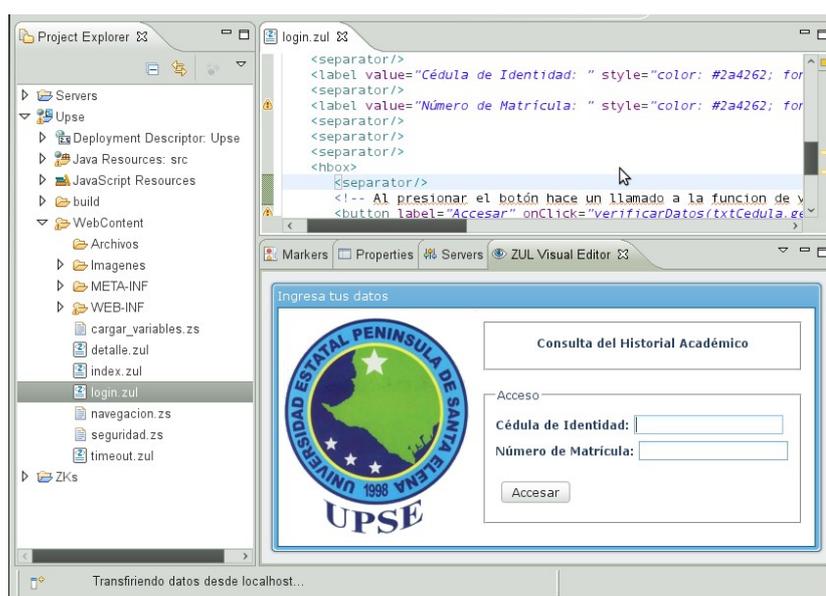
```

<separator/>
<label value="Número de Matrícula: " style="color: #2a4262;
font-weight: bold;" />
<textbox id="txtMatricula" width="150px"
onOK="verificarDatos(txtCedula.getValue(), txtMatricula.getValue())"/>
<separator/>
<separator/>
<separator/>
<hbox>
<separator/>
<!-- Al presionar el botón hace un llamado a la función de
verificación ubicado en el archivo de código seguridad.zs -->
<button label="Accesar"
onClick="verificarDatos(txtCedula.getValue(), txtMatricula.getValue())"/>
<separator/>
</hbox>
<separator/>
</groupbox>
</vbox>
</hbox>
</window>
</zk>

```

A breves rasgos podemos interpretar que este documento está diseñado con el modelo de cajas y se incluyen los controles necesarios para recoger la cédula de identidad y el número de matrícula del estudiante. También se observa que al aceptar el valor ingresado en el control del número de matrícula o al presionar el botón accesar, se invoca a la función verificarDatos, la cual no está definida dentro de este documento, sino en el archivo seguridad.zs, incluido al inicio.

GRÁFICO # 67
Visualización del código del documento y el editor visual.



Fuente: Autores

A continuación se observa el código de la función verificarDatos.

```
import upse.academico.XmlWebServices.Estudiantes;
...
void verificarDatos(String p_cedula, String p_matricula)
{
    Estudiantes datos = loginwin.obtenerDatos(p_cedula, p_matricula);
    if (datos!=null)
    {
        session.setAttribute("eid", datos.getCedula());
        session.setAttribute("enombre", datos.getNombre());
        session.setAttribute("ematricula", datos.getMatricula());
        execution.sendRedirect("index.zul", null);
    }
    else
    {
        alert("El intento de ingresar no fue exitoso, por favor revise sus datos y
vuelva a intentarlo.");
    }
}
```

Se observa como en esta función se hace uso del método obtenerDatos asociada al objeto loginwin, que si retrocedemos un poco, vemos que es el nombre de la ventana definida en el documento login.zul. Este método se hereda de la clase upse.academico.controladores.EstudiantesController que es instanciada por la ventana. A continuación el detalle de este método.

```
package upse.academico.controladores;
//Implementa el controlador para verificar los datos del estudiante.
import upse.academico.XmlWebServices.AcademicoSoapProxy;
import upse.academico.XmlWebServices.Estudiantes;

import org.zkoss.zk.ui.WrongValueException;
import org.zkoss.zul.*;

import java.rmi.RemoteException;
import java.util.*;

public class EstudiantesController extends Window {

    private static final long serialVersionUID = 1L;

    public EstudiantesController() {
    }

    // Obtiene los datos del estudiante haciendo una búsqueda por la cédula y
    // comparando el número de matricula para efectos de validación
    public Estudiantes obtenerDatos(String p_cedula, String p_matricula)
        throws RemoteException{
        AcademicoSoapProxy proxy = new
            upse.academico.XmlWebServices.AcademicoSoapProxy();
        Estudiantes estudiante = proxy.obtenerInfoGeneral(p_matricula);
        if (estudiante != null)
        {
            if (estudiante.getCedula().equals(p_cedula))
            {
                return(estudiante);
            }
        }
    }
}
```



```

<?page title="Historial Académico Universitario"?>
<zk>
<!-- Importa funcionalidad de otros archivos con código -->
<zscript language="java" src="cargar_variables.zs"/>
<zscript language="java" src="navegacion.zs"/>
<zscript language="java" src="seguridad.zs"/>
<!-- Define la ventana principal, solo si previamente se ha provisto de los
datos para la consulta -->
<window id="principal" width="100%" border="none"
        if="`${v_eid!=null and v_eid!=&quot;&quot;}`" >
    <!-- Barra principal -->
    <hbox width="100%">
        <panel border="normal" collapsible="true" open="true" Framable="true" >
            <panelchildren >
                <button label="Hacer otra consulta" onClick="cerrarSesion()"/>
            </panelchildren>
        </panel>
    </hbox>
<!-- Ventana donde se muestra la información general -->
<window id="info" border="none"
use="upse.academico.controladores.InfoGeneralController">
    <!-- Define un item para almacenar el dato de la consulta que es utilizado en
el controlador -->
    <textbox id="eid" value="`${v_ematricula}`" visible="false"/>
    <hbox>
        <vbox>
            </vbox>
    </hbox>
    <!-- Coloca los datos devueltos por la consulta -->
    <vbox align="center">
        <separator/>
        <label id="etiqueta" value="Información General"
            style="color: #2a4262; font-size:16px; font-weight: bold;" />
        <separator/>
        <hbox>
            <separator/>
            <listbox id="lb">
                <listhead sizable="true" visible="false">
                    <listheader label="Nombre" width="100px"/>
                    <listheader label="Descripción" width="250px"/>
                </listhead>
                <listitem forEach="`${info.infoGeneral}`" value="`${each}`">
                    <listcell label="`${each.nombre}`" style="font-weight : bold"/>
                    <listcell label="`${each.descripcion}`" style="color: #3e6f90;" />
                </listitem>
            </listbox>
            <separator/>
        </hbox>
    </vbox>
</vbox>
</vbox>
</window>
<!-- Ventana para mostrar el detalle de carreras -->
<window id="carreras" border="none"
use="upse.academico.controladores.CarrerasTomadasController" height="600px">
    <textbox id="eid" value="`${v_eid}`" visible="false"/>
    <hbox>
        <vbox>
            </vbox>
    </hbox>
    <vbox align="center" >
        <separator/>
        <label value="Carreras Cursadas"
            style="color: #2a4262; font-size:16px; font-weight: bold;"/>
        <separator/>
        <!-- Define el marco para los tabuladores -->
        <tabbox id="frame_app" >
            <!-- Genera las pestañas de acuerdo a los datos devueltos por la
consulta desde el controlador -->
            <tabs id="pestañas" >

```

```

        <tab forEach="{carreras.carrerasTomadas}" label="{each.escuela}"/>
    </tabs>
    <!-- Genera los paneles asociados a cada pestaña -->
    <tabpanel id="contenido" width="100%" >
        <tabpanel forEach="{carreras.carrerasTomadas}" >
            <!-- Muestra la información general de la carrera devuelta por la
consulta -->
                <listbox>
                    <listhead sizable="true">
                        <listheader label="Matricula"
                            style="color: #2a4262; font-weight: bold;" />
                        <listheader label="Facultad" width="200px"
                            style="color: #2a4262; font-weight: bold;" />
                        <listheader label="Escuela" width="200px"
                            style="color: #2a4262; font-weight: bold;" />
                        <listheader label="Modalidad"
                            style="color: #2a4262; font-weight: bold;" />
                    </listhead>
                    <listitem>
                        <listcell label="{each.matricula}"
                            style="color: #3e6f90;" />
                        <listcell label="{each.facultad}"
                            style="color: #3e6f90;" />
                        <listcell label="{each.escuela}"
                            style="color: #3e6f90;" />
                        <listcell label="{each.modalidad}"
                            style="color: #3e6f90;" />
                    </listitem>
                </listbox>
                <!-- Carga un enlace a una página que muestra el detalle de notas
enviándole el número de matrícula -->
                <iframe src="detalle.zul?matricula={each.matricula}"
                    style="width: 100%; height: 600px; padding: 0 0 0 0;
margin: 0 0 0 0;" />
            </tabpanel>
        </tabpanel>
    </tabbox>
</vbox>
</vbox>
</vbox>
</window>
</window>
<!-- En caso de que el usuario no haya iniciado sesión pide que se lo haga -->
<window id="redireccionar" title="Redireccionando a la Pantalla de Inicio de
Sesión" if="{v_eid==null or v_eid=="&quot;}"
apply="upse.academico.ventanas.MyComposer">
    <hbox>
        <vbox align="center" width="300px">
            <separator/>
            <label value="Usted no ha iniciado sesión en el Portal de Consultas del
Historial Académico."/>
            <separator/>
            <button label="Iniciar sesión"
                onClick="redireccionar(&quot;login.zul&quot;)" />
            <separator/>
        </vbox>
    </hbox>
</window>
</zk>

```

En la ventana info, existe un control textbox oculto que al cargar la página obtiene el número de matrícula, el cual fue almacenado en una variable de sesión luego de que se validaron los datos ingresados por el estudiante. En el controlador de la ventana se hace uso de este control para obtener los datos del método web

ObtenerInfoGeneral. En la ventana se hace uso de la propiedad forEach en el componente listitem accediendo al atributo infoGeneral, heredado desde la clase upse.academico.controladores.InfoGeneralController, que devuelve un conjunto de datos obtenidos desde el servicio web, y por cada registro genera un elemento listitem asignando cada columna a un elemento listcell. A continuación se detalla el código de la clase upse.academico.controladores.InfoGeneralController.

```

package upse.academico.controladores;

//Implementa el controlador para cargar la Información General.
import upse.academico.XmlWebServices.*;

import org.zkoss.zul.*;

import java.rmi.RemoteException;
import java.util.*;
import org.zkoss.zk.ui.*;

public class InfoGeneralController extends Window {

    private static final long serialVersionUID = 10;
    List _infogeneral = new ArrayList();

    public InfoGeneralController(){
    }

    public List getInfoGeneral() throws WrongValueException, RemoteException{
        // Obtiene el valor de un campo oculto de la ventana
        Textbox v_eid = (Textbox) getFellow("eid");
        // Obtiene los datos del web service y los almacena para su posterior uso
        AcademicoSoapProxy proxy = new
            upse.academico.XmlWebServices.AcademicoSoapProxy();
        Estudiantes estudiante =
            proxy.obtenerInfoGeneral(v_eid.getValue().toString());
        _infogeneral.add(new InfoGeneral("Matricula      :",
            estudiante.getMatricula().toString()));
        _infogeneral.add(new InfoGeneral("Identificación  :",
            estudiante.getCedula().toString()));
        _infogeneral.add(new InfoGeneral("Nombre          :",
            estudiante.getNombre().toString()));
        //
        return _infogeneral;
    }

    public void setInfoGeneral(List infogeneral) {
        _infogeneral = infogeneral;
    }
}

```

La ventana carreras también contiene un control textbox oculto que en cambio almacena la cédula de identidad, para con este valor realizar la consulta al método ObtenerCarrerasTomadas a través del controlador heredado. De manera similar a la ventana info, el contenedor de tabuladores es asociado al atributo carrerasTomadas, generando una pestaña y un panel por cada registro devuelto por

el método web. En esta ventana existe un elemento iframe en el cual se carga el documento detalle.zul llamado por cada carrera y al que se le envía como parámetro el número de matrícula para que muestre el historial académico de cada carrera.

A continuación se detalla el código de la clase `upse.academico.controladores.CarrerasTomadasController`.

```
package upse.academico.controladores;
//Implementa el controlador para cargar las Carreras Tomadas.
import upse.academico.XmlWebServices.AcademicoSoapProxy;
import upse.academico.XmlWebServices.CarrerasTomadas;
import upse.academico.XmlWebServices.MateriasCursadas;

import org.zkoss.zk.ui.WrongValueException;
import org.zkoss.zul.*;

import java.rmi.RemoteException;
import java.text.NumberFormat;
import java.util.*;

public class CarrerasTomadasController extends Window {

    private static final long serialVersionUID = 1L;

    List _carrerastomadas = new ArrayList();

    public CarrerasTomadasController() {
    }

    public List getCarrerasTomadas() throws WrongValueException, RemoteException
    {
        // Obtiene el valor de un campo oculto de la ventana
        Textbox v_eid = (Textbox) getFellow("eid");
        // Obtiene los datos del web service y los almacena para su posterior uso
        upse.academico.XmlWebServices.AcademicoSoapProxy proxy = new
            upse.academico.XmlWebServices.AcademicoSoapProxy();
        Object carreras[] =
            proxy.obtenerCarrerasTomadas(v_eid.getValue().toString());
        for (int i=0; i<=carreras.length - 1; i++)
        {
            _carrerastomadas.add((upse.academico.XmlWebServices.CarrerasTomadas)
                carreras[i]);
        }
        return _carrerastomadas;
    }

    public void setCarrerasTomadas(List carrerastomadas) {
        _carrerastomadas = carrerastomadas;
    }
}
```

En este controlador se observa como accedemos al método `obtenerCarrerasTomadas` utilizando el contenido del elemento `textbox` oculto. Una

vez obtenidos los datos estos son almacenados en el atributo `_carrerastomadas` que luego es utilizado para generar los tabuladores en la ventana.

La última interface que compone la aplicación de consulta se almacena en el documento `detalle.zul` y muestra el detalle de las materias tomadas en base al parámetro `matricula` enviado en la URL de la solicitud.

```
<?page title="new page title" contentType="text/html;charset=UTF-8"?>
<zk>
<!-- Importa funcionalidad de otros archivos con código -->
<zscript language="java" src="cargar_variables.zs"/>
<zscript language="java" src="navegacion.zs"/>
<zscript language="java" src="seguridad.zs"/>
<!-- Define la ventana principal, solo si se ha recibido el parámetro con el
número de matrícula -->
<window id="materias" border="none"
  use="upse.academico.controladores.MateriasCursadasController"
  if="{not empty param.matricula}" width="100%" height="100%"
  style="padding: 0; margin: 0">
  <!-- Almacena en un archivo oculto el dato para realizar la consulta -->
  <textbox id="eid" value="{param.matricula}" visible="false"/>
  <vbox align="center">
    <separator/>
    <label value="Historial Académico"
      style="color: #2a4262; font-size:16px; font-weight: bold;"/>
    <separator/>
    <hbox align="left">
      <separator/>
      <listbox id="lb">
        <!-- Define el encabezado de la lista donde se mostrará el resultado
de la consulta. -->
        <listhead sizable="true">
          <listheader label="Materia" width="150px"
            style="color: #2a4262; font-weight: bold;" />
          <listheader label="1° Parcial" width="50px"
            style="color: #2a4262; font-weight: bold;" />
          <listheader label="2° Parcial" width="50px"
            style="color: #2a4262; font-weight: bold;" />
          <listheader label="3° Parcial" width="50px"
            style="color: #2a4262; font-weight: bold;" />
          <listheader label="4° Parcial" width="50px"
            style="color: #2a4262; font-weight: bold;" />
          <listheader label="Mejoramiento" width="50px"
            style="color: #2a4262; font-weight: bold;" />
          <listheader label="Promedio" width="50px"
            style="color: #2a4262; font-weight: bold;" />
          <listheader label="Nivel" width="35px"
            style="color: #2a4262; font-weight: bold;" />
          <listheader label="Ciclo" width="50px"
            style="color: #2a4262; font-weight: bold;" />
          <listheader label="Tipo" width="50px"
            style="color: #2a4262; font-weight: bold;" />
          <listheader label="Estado" width="50px"
            style="color: #2a4262; font-weight: bold;" />
        </listhead>

        <!-- Genera un conjunto de items correspondientes a la cantidad de
registros devueltos por la consulta. -->
        <listitem forEach="{materias.materiasCursadas}" value="{each}">
          <listcell label="{each.materia}" style="color: #3e6f90;" />
        </listitem>
      </listbox>
    </hbox>
  </vbox>
</window>
```

```

        <!-- En caso de que lleguen valores en -1, estos son filtrados
        utilizando el operador condicional ? -->
        <listcell label="\${ (each.primerp==-1)?':each.primerp}'"
            style="text-align: right; color: #3e6f90;" />
        <listcell label="\${ (each.segundop==-1)?':each.segundop}'"
            style="text-align: right; color: #3e6f90;" />
        <listcell label="\${ (each.tercerp==-1)?':each.tercerp}'"
            style="text-align: right; color: #3e6f90;" />
        <listcell label="\${ (each.cuartop==-1)?':each.cuartop}'"
            style="text-align: right; color: #3e6f90;" />
        <listcell label="\${ (each.recuperacion==-1)?':each.recuperacion}'"
            style="text-align: right; color: #3e6f90;" />
        <listcell label="\${ (each.promedio==-1)?':each.promedio}'"
            style="text-align: right; color: #3e6f90;" />
        <listcell label="\${each.nivel}" style="color: #3e6f90;" />
        <listcell label="\${each.semestre_Ano}" style="color: #3e6f90;" />
        <listcell label="\${each.estado}" style="color: #3e6f90;" />
        <listcell label="\${each.obs}" style="color: #3e6f90;" />
    </listitem>
</listbox>
<separator/>
</hbox>
</vbox>
</window>

</zk>

```

Esta interface define una ventana llamada `materias` que se instancia desde la clase `upse.academico.controladores.MateriasCursadasController`. Esta ventana contiene, igual que las anteriores, un control `textbox` oculto que almacena el valor obtenido del parámetro `matricula` y que es utilizado para consultar los datos desde el método `web ObtenerMateriasCursadas`. La clase `MateriasCursadasController` tiene la misma funcionalidad detallada con los controladores revisados anteriormente, en la siguiente página se detalla el código fuente del mismo.

```

package upse.academico.controladores;
//Implementa el controlador para cargar la información de Historia Académica.
import upse.academico.XmlWebServices.*;
import org.zkoss.zk.ui.WrongValueException;
import org.zkoss.zul.*;
import java.rmi.RemoteException;
import java.text.NumberFormat;
import java.util.*;

public class MateriasCursadasController extends Window {
    private static final long serialVersionUID = 10;
    List _materiascursadas = new ArrayList();

    public MateriasCursadasController() {
    }

    public List getMateriasCursadas() throws WrongValueException, RemoteException
    {
        // Obtiene el valor de un campo oculto de la ventana
        Textbox v_eid = (Textbox) getFellow("eid");
        // Crea una instancia del objeto para utilizarlo de almacenamiento temporal.
        MateriasCursadas auxiliar;
    }
}

```

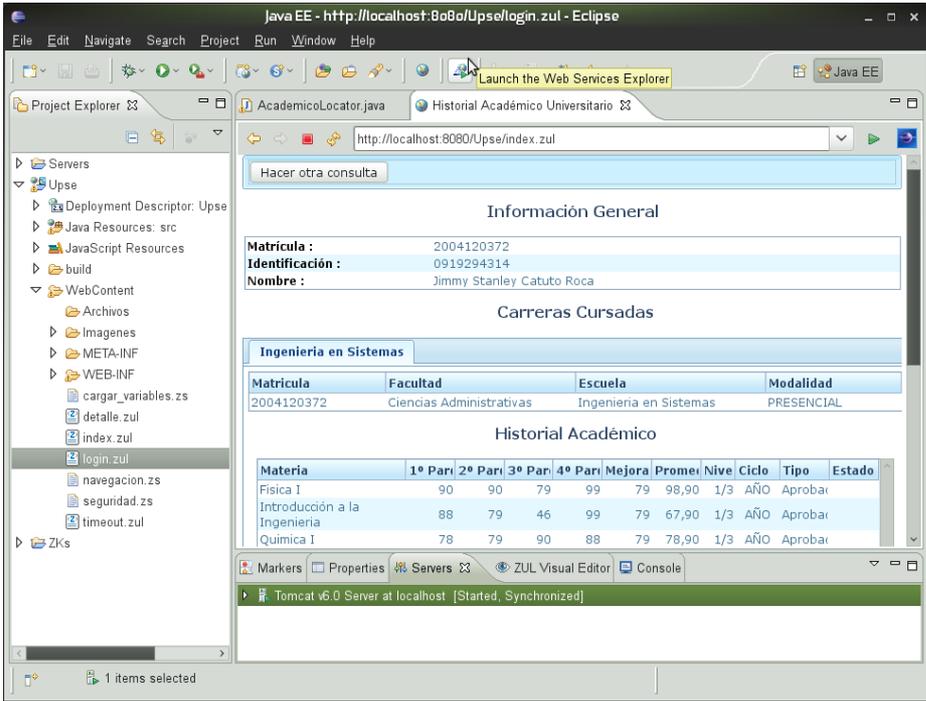
```

// Obtiene los datos del web service y los almacena para su posterior uso
AcademicoSoapProxy proxy = new
    upse.academico.XmlWebServices.AcademicoSoapProxy();
Object materias[] =
    proxy.obtenerMateriasCursadas(v_eid.getValue().toString());
for (int i=0; i<=(materias.length - 1); i++)
{
    auxiliar = (upse.academico.XmlWebServices.MateriasCursadas) materias[i];
    // Realiza la limpieza de ciertos valores que no se mostrarán.
    //
    NumberFormat nf = NumberFormat.getInstance();
    nf.setMaximumFractionDigits(2);
    auxiliar.setPromedio(Double.valueOf(nf.format(auxiliar.getPromedio()).r
eplace(",", "", ".")));
    // _materiascursadas.add((upse.academico.XmlWebServices.MateriasCursadas
) materias[i]);
    if (!(i==0 & auxiliar.getMateria().equals("PROMEDIO FINAL")))
    // Para no retornar la primera fila.
    _materiascursadas.add(auxiliar);
}
return _materiascursadas;
}
public void setMateriasCursadas(List materiascursadas) {
    _materiascursadas = materiascursadas;
}
}

```

Con los documentos zul listos, podemos realizar una ejecución de prueba para constatar los resultados.

GRÁFICO # 68
Ejecución de la aplicación en Eclipse.



Fuente: Autores

ANEXO # 6



MANUAL DEL SERVIDOR DE AUTENTICACIÓN

INTRODUCCIÓN

CAS es un sistema de autenticación creado originalmente por la Universidad Yale para proporcionar una forma de confianza para autenticar a un usuario en una aplicación. CAS se convirtió en un proyecto JASIG en diciembre de 2004.

GRÁFICO # 69 Logo de CAS Server.



Fuente: <http://www.jasig.org/cas>

CAS proporciona un servicio Sign-On uniempresarial:

- Un protocolo abierto y bien documentado.
- Un componente de servidor Java de código abierto.
- Una librería de clientes para Java, .Net, PHP, Apache, uPortal, y otros.
- Se integra con uPortal, BlueSocket, TikiWiki, Mule, Liferay, Moodle y otros.
- Documentación de la comunidad y soporte de implementación.
- Una amplia comunidad de adoptantes.

CONFIGURACIÓN

Hay algunas consideraciones fundamentales para la configuración de CAS:

1. La integración con un proveedor de autenticación.
2. La política de seguridad.
3. Autorización.

4. Consideraciones de disponibilidad.
5. Lanzamiento del atributo.

AUTENTICACIÓN

CAS se integra con los siguientes mecanismos de autenticación:

- Active Directory
- Generic
- JAAS
- JDBC
- LDAP
- Legacy
- RADIUS
- SPNEGO
- Trusted
- X.509 Certificates

POLÍTICA DE SEGURIDAD

CAS utiliza tickets para implementar los protocolos de autenticación soportados, por lo que se deduce que el comportamiento de la entrada determina la mayoría de los aspectos de la política de seguridad.

En su configuración por defecto (como la de CAS 3.3.3), CAS dispone lo siguiente:

- Tickets ticket-granting (TGT) que expira después de más de dos horas de inactividad.
- Tickets de servicio One-time-use (ST) que debe ser validado dentro de los 5 minutos.

El comportamiento por defecto puede ser modificado mediante parámetros de ajuste de la política de caducidad, o eligiendo entre políticas de caducidad alternativas. Para más información vea la sección de Política de Expiración de Ticket.

AUTORIZACIÓN

La autorización en CAS determina qué servicios pueden solicitar y validar tickets de servicio. Por defecto no hay restricciones y cualquier servicio en cualquier URL puede solicitar y validar un ticket de servicio. Esto puede ser deseable por un número de razones, de manera que CAS proporciona una herramienta de servicios de gestión administrativa para controlar qué servicios pueden utilizar el servidor CAS. Además, la herramienta de gestión de servicios permite un control detallado de los atributos que pueden ser liberados a los servicios.

DISPONIBILIDAD

CAS proporciona una alta disponibilidad mediante la colocación de datos de estado de cliente autenticado en tickets almacenados en un componente TicketRegistry que tiene un número de implementaciones que soportan distintos entornos:

- BerkleyDB
- Default
- JBOSS TreeCache
- JDBCTicketRegistry
- JpaTicketRegistry
- MemcacheTicketRegistry
- TicketRegistry Cleaner

Importante: El DefaultTicketRegistry, que está habilitado por defecto, no es adecuado para entornos de alta disponibilidad a menos que sea combinado con una herramienta como Terracota.

Para todos los registros de ticket de alta disponibilidad de arriba, cualquier número de instancias del servidor CAS pueden ser respaldadas por un registro distribuido y su carga balanceada en una configuración activo-activo. Además, la replicación del estado de sesión es innecesaria desde que los tickets almacenados en el registro contienen el estado autenticado de los usuarios, que permite a un nodo del servidor CAS fallar sin que los usuarios pierdan su estado de sesión SSO.

CONTROLADOR DE AUTENTICACIÓN LDAP

INCLUYENDO EL CONTROLADOR

En el archivo pom.xml para su aplicación web (por defecto es \$ {project.home}/cas-server-webapp/pom.xml) añada la siguiente dependencia:

```
<dependency>
  <groupId>${project.groupId}</groupId>
  <artifactId>cas-server-support-ldap</artifactId>
  <version>${project.version}</version>
</dependency>
```

A partir del 3.3.5, esta ya está incluida en el POM adecuado.

CLASES PRINCIPALES

Usted necesita decidir como desea que CAS autentique las credenciales. ¿Si debe intentar autenticarse al (enlazarse al) servidor LDAP usando las credenciales directamente como el usuario? ¿O debe primero buscar al usuario en algún sub-árbol y entonces intentar enlazarse como ese usuario?. Es más eficiente y más seguro para utilizar fastbind, pero esto no siempre es posible. Esto se explicará en detalle luego en este documento.

Ambos métodos requieren que usted configure un bean de contexto LDAP: esta es la configuración para acceder a su directorio, se recomienda configurar un nuevo bean en la lista y referenciarlo desde la configuración del AuthenticationHandler, como se explica en las instrucciones de esta página.

FastBindLdapAuthenticationHandler

Esta es la implementación de autenticación LDAP más directa. Simplemente intenta enlazarse al directorio con las credenciales proporcionadas por el usuario. Dado que el usuario no es consciente del árbol LDAP, usted tiene que configurar el controlador para probar y enlazar a cierto nodo preconfigurado. Esto significa que sólo puede utilizar este método si es capaz de configurar el nodo del usuario con antelación. Caso contrario usted necesitará utilizar el método search-and-bean (que se explica más adelante).

El FastBindLdapAuthenticationHandler soporta las siguientes propiedades:

- **filter** - la propiedad filter es el filtro LDAP que se utilizará para la búsqueda. Al construir el filtro, donde quiera que aparezca el nombre de usuario, colocar un “%u”.
- **ignorePartialResultException** – Esta característica informa al resorte LDAP para ignorar los PartialResultException que se puedan obtener cuando se conecta a un Active Directory.
- **ContextSource** – Esta es una referencia a un LdapContextSource (véase más abajo) que contiene los ajustes para conectarse al servidor LDAP.

BindLdapAuthenticationHandler

Este método primero hace una búsqueda general para el usuario antes de que este trate de enlazarse al directorio. Cuando encuentra el usuario intenta hacer un enlace con el DN que resultó de la búsqueda y las contraseñas desde las credenciales presentadas por el usuario.

Tenga en cuenta que para este método necesitará un usuario especial al que se le permita buscar en el directorio (o permitir el acceso anónimo para buscar en el directorio).

- **filter** - La propiedad filter es el filtro LDAP que se utilizará para la búsqueda. Al construir el filtro, donde quiera que aparezca el nombre de usuario, colocar un “%u”.
- **ignorePartialResultException** – Esta propiedad informa al LdapTemplate para ignorar los PartialResultExceptions que se pueden obtener cuando se conecta a un Active Directory.
- **contextSource** – Esta es una referencia al LdapContextSource (véase más abajo) que contiene los ajustes para conectarse al LDAP.
- **allowMultipleAccounts** – Permite que más de una cuenta sea devuelta.
- **maxNumberOfResults** – Este es el número máximo de resultados que permitimos.
- **scope** – Uno de los Scopes “SearchControl” predefinidos: SearchControls.OBJECT_SCOPE, SearchControls.ONELEVEL_SCOPE, o SearchControls.SUBTREE_SCOPE
- **searchBase** – La base de la búsqueda es el nodo en el directorio desde donde la búsqueda se realizará.
- **timeout** – Esta es la cantidad de tiempo que estamos dispuestos a esperar los resultados que la búsqueda retorna.

AuthenticatedLdapContextSource

Esta clase es una extensión de LdapContextSource del Spring LDAP. Se ha extendido para permitirle recuperar DirContexts específicos al mismo tiempo que se reutilizan ajustes del entorno existente.

Su método adicional es el modelo de la propiedad DataSource de Java.

A partir del CAS 3.3.2 esta clase ha sido removida porque el `LdapContextSource` del Spring LDAP 1.3 ahora es compatible con los métodos necesarios.

El `LdapContextSource` define algunas propiedades importantes, listados a continuación (nótese que algo de esto es copiado directamente desde los JavaDocs):

- **`anonymousReadOnly`** – Establecido como falso en un entorno autenticado debe ser creado para operaciones de sólo escritura.
- **`baseEnvironmentProperties`** – Si es necesaria una personalización de las propiedades de algún entorno. Éstas pueden ser configuradas usando este método.
- **`password`** – Configura la contraseña (credenciales) a utilizar para conseguir contextos autenticados.
- **`pooled`** – establece si la bandera de la puesta en común debe ser utilizada. Tenga en cuenta que incluso si se establece la puesta en común, el método adicional proporcionada por `AuthenticatedLdapContextSource` no utilizará la puesta en común.
- **`url o urls`** - La(s) url(s) para el servidor LDAP. Especificando una lista de servidores automáticamente configura la redundancia lo que es muy recomendable hacer. El puerto 636 es el típico puerto LDAP SSL, el 389 es el puerto LDAP estándar. Java SDK v1.4.2+ interpreta correctamente `ldap://` o `ldaps://` sin especificar el puerto. Si está usando un JDK antiguo no puede especificar el protocolo `ldaps://` por lo que necesitará definir explícitamente un puerto y establecer el protocolo de SSL en las propiedades del entorno.
- **`userName`** – el nombre de usuario (principal) para obtener contextos autenticados.

Por favor tenga en cuenta que la JVM necesita confiar en el certificado de su servidor SSL con LDAP habilitado, caso contrario CAS se negará a conectarse a

su servidor LDAP. Usted puede añadir el certificado del servidor LDAP en el almacén de confianza JVM (\$JAVA_HOME/jre/lib/security/cacerts) para resolver este problema.

CONFIGURACIÓN

Tenga en cuenta que toda la configuración debe llevarse a cabo en **cas-server-webapp/src/main/webapp/WEB-INF/deployerConfigContext.xml**

CONFIGURAR EL AUTHENTICATEDLDAPCONTEXTSOURCE

A partir de CAS 3.3.2, usted DEBE usar LdapContextSource del Spring LDAP (org.springframework.ldap.core.support.LdapContextSource) en lugar de AuthenticatedLdapContextSource. Dentro de <beans>...</beans>, coloque su bean de AuthenticatedLdapContextSource. Su configuración se verá como así:

```
<bean id="contextSource" class="org.springframework.ldap.core.support.LdapContextSource">
  <property name="pooled" value="true"/>
  <property name="urls">
    <list>
      <value>ldaps://ldap.rutgers.edu/</value>
      <value>ldaps://ldap2.rutgers.edu/</value>
    </list>
  </property>
  <property name="userDn" value="{bind_username_goes_here}"/>
  <property name="password" value="{bind_user_password_goes_here}"/>
  <property name="baseEnvironmentProperties">
    <map>
      <entry>
        <key>
          <value>java.naming.security.authentication</value>
        </key>
        <value>simple</value>
      </entry>
    </map>
  </property>
</bean>
```

Recuerde, el objeto Users de Active Directory es un **cn**, no un **ou**. Por lo que si usted está conectándose a **AD**, entonces su configuración se verá así:

```

<bean id="contextSource" class="org.springframework.ldap.core.support.LdapContextSource">
  <property name="urls">
    <list>
      <value>ldaps://yourdc.yourdomain.edu/</value>
    </list>
  </property>
  <property name="userDn" value="{cn=bind_user_goes_here,cn=Users,dc=yourdomain,dc=edu}"/>
  <property name="password" value="{bind_user_password_goes_here}"/>
  <property name="baseEnvironmentProperties">
    <map>
      <entry>
        <key>
          <value>java.naming.security.authentication</value>
        </key>
        <value>simple</value>
      </entry>
    </map>
  </property>
</bean>

```

Nota: usted debe establecer explícitamente la propiedad de entorno `java.naming.security.authentication` en “simple” para el usuario y password que se utilizarán. Si no hay la propiedad de entorno `java.naming.security.authentication` especificada, se pondrá por defecto en “none”.

Esto significa que el directorio siempre será accedido como “anonymous”. Esto es muy útil en el escenario de Fast Bind.

La especificación de una lista de servidores automáticamente configura la redundancia lo que es muy recomendable hacer.

El puerto 636 es el típico puerto SSL LDAP, el 389 es el puerto LDAP estándar. Java SDK v1.4.2+ interpreta correctamente `ldap://` o `ldaps://` sin especificar el puerto.

Si está usando una versión antigua de JDK no podrá especificar el protocolo `ldaps://` por lo que usted necesitará definir explícitamente un puerto y establecer el protocolo para SSL en las propiedades de entorno.

Por favor tenga en cuenta que la JVM necesita confiar en el certificado de su servidor SSL con LDAP habilitado, caso contrario CAS se negará a conectarse a

su servidor LDAP. Usted puede añadir el certificado del servidor LDAP en el almacén de confianza JVM (\$JAVA_HOME/jre/lib/security/cacerts) para resolver este problema.

ESTABLECER EL ENLACE LDAP Y LEER LOS TIEMPOS DE ESPERA

Con el fin de manejar adecuadamente las múltiples configuraciones de fallos de los servidores LDAP (es decir, múltiples entradas en la sección <urls>), usted puede establecer los tiempos de espera del entorno para la conexión e intentos de lectura ldap. Esto acelerará los fallos en caso de que su LDAP primario no esté disponible. Agregue lo siguiente al elemento baseEnvironmentProperties:

```
<property name="baseEnvironmentProperties">
  <!--
    Set the LDAP connect and read timeout(in ms) for the java ldap class
    See http://java.sun.com/products/jndi/tutorial/ldap/connect/create.html
  -->
  <map>
    <entry>
      <key>
        <value>com.sun.jndi.ldap.connect.timeout</value>
      </key>
      <value>2000</value>
    </entry>
    <entry>
      <key>
        <value>com.sun.jndi.ldap.read.timeout</value>
      </key>
      <value>2000</value>
    </entry>
  </map>
</property>
```

Nota: El ejemplo anterior tiene un tiempo de espera de 2 segundos para el enlace y las solicitudes de lectura. Esto puede no ser apropiado para su entorno.

DefaultTicketRegistry

El DefaultTicketRegistry está incluido automáticamente en el servidor CAS y está habilitado por defecto. No hay muchas opciones de configuración para el registro

en sí, ya que se basa en los valores por defecto del ConcurrentHashMap, que son suficientes para la mayoría de la gente.

Sin embargo, si se quiere modificar su registro para un rendimiento óptimo, el DefaultTicketOption ofrece algunas opciones. Hay un constructor adicional que toma los siguientes argumentos:

- **initialCapacity** – la capacidad inicial. La implementación realiza un apresto interno para acomodar muchos elementos.
- **loadFactor** – el umbral de factor de carga, usado para controlar el apresto. El apresto puede ser realizado cuando el numero medio de elementos por fin supera este umbral.
- **concurrencyLevel** – el número estimado de procesos de actualización concurrentes. La implementación realiza un apresto interno para intentar acomodar muchos procesos.

Puede ser configurado así:

```
<bean id="ticketRegistry" class="org.jasig.cas.ticket.registry.DefaultTicketRegistry">  
  <constructor-arg index="0" value="1000" />  
  <constructor-arg index="1" value="1" />  
  <constructor-arg index="2" value="16" />  
</bean>
```

REQUISITOS PARA LA INSTALACIÓN

1. Apache tomcat (instalado y funcionando).
2. Java JDK (instalado).

INSTRUCCIONES

1. Descargue el servidor de directorio de Apache desde <http://directory.apache.org/>.

2. Ejecute el programa de instalación con todas las opciones por defecto y pruebe que el servidor esté funcionando en localhost.
3. Descargue la instalación de CAS y encuentre el war file, por ejemplo: /cas-server-\$VERSION/modules/cas-server-webapp-\$VERSION.war.
4. Inicie el servidor Tomcat, por ejemplo: \$TOMCAT_HOME/bin/startup.bat.
5. Agregue el war file de CAS, cas-server-webapp-3.2.1.war, a la carpeta webapps, \$TOMCAT_HOME/webapps.
6. Confirme que exista el siguiente directorio, lo que indica una implementación exitosa, (\$TOMCAT_HOME/webapps/cas-server-webapp-\$VERSION).
7. Usted puede también examinar el archivo de registro de salida standar de Tomcat para errores, \$TOMCAT_HOME/logs/catalina.out.
8. Detenga el servidor tomcat, por ejemplo: \$TOMCAT_HOME/bin/shutdown.bat.
9. Agregue lo siguiente al fichero pom.xml en la carpeta META-INF, \$TOMCAT_HOME/webapps/cas-server-webapp-\$VERSION\META-INF/maven/org.jasig.cas/cas-server-webapp:

```

<dependency>
  <groupId>${project.groupId}</groupId>
  <artifactId>cas-server-support-ldap</artifactId>
  <version>${project.version}</version>
</dependency>

```

10. Edite \$TOMCAT_HOME/webapps/cas-server-webapp-\$VERSION/WEB-INF/deployerConfigContext.xml como sigue:
 - a) Añada el siguiente bean de autenticación LDAP
 - b) Quite el controlador de autenticación de demostración, org.jasig.cas.authentication.handler.support.SimpleTestUsernamePassword AuthenticationHandler, desde el authenticationHandlers propiedad del bean org.jasig.cas.authentication.AuthenticationManager Impl.

```

<bean id="contextSource"
  class="org.jasig.cas.adaptors.ldap.util.AuthenticatedLdapContextSource">
  <property name="pooled" value="true"/>
  <property name="urls">
    <list>
      <value>ldap://localhost:10389</value>
    </list>
  </property>
  <property name="userName" value="uid=admin,ou=system"/>
  <property name="password" value="secret"/>
  <property name="baseEnvironmentProperties">
    <map>
      <entry key="java.naming.security.authentication" value="simple" />
    </map>
  </property>
</bean>

```

c) Agregue el controlador de autenticación de enlace rápido LDAP:

```

<bean class="org.jasig.cas.adaptors.ldap.FastBindLdapAuthenticationHandler" >
  <property name="filter" value="uid=%u,ou=system" />
  <property name="contextSource" ref="contextSource" />
</bean>

```

11. Agregue el cas-server-support-ldap-\$VERSION.jar desde la instalación de CAS a \$TOMCAT_HOME/webapps/cas-server-webapp-\$VERSION/WEB-INF/lib.
12. Inicie tomcat y confirme que no hayan errores en el log \$TOMCAT_HOME/logs/catalina.out
13. Abra un navegador a la dirección http://localhost:8080/cas-server-webapp-\$VERSION/ y autentique con las siguientes credenciales, admin/secret.

PROTOCOLO CAS

CONVENIOS Y DEFINICIONES

Las palabras claves “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, y

"OPTIONAL" en este documento deben interpretarse como se describe en el RFC 2119

“**Cliente**” se refiere al usuario final y/o navegador web.

“**Servidor**” se refiere al servidor Central Authentication Service.

“**Servicio**” se refiere a la aplicación a que el cliente está intentando tener acceso.

“**Servicio de respaldo**” se refiere a la aplicación a la cual un servicio está intentando acceder en nombre de un cliente. Esto también puede ser denominado como “servicio de destino”.

<LF> es un avance de línea descubierto (valor ASCII 0x0a).

URIs de CAS

CAS es un protocolo basado en HTTP que requiere que cada uno de sus componentes sea accesible a través de URIs específicas. En esta sección se discutirá cada una de las URIs.

/login COMO SOLICITANTE DE CREDENCIAL

La URI /login opera de dos maneras: como un solicitante de credenciales, y como un receptor de credenciales. Responde a las credenciales actuando como un receptor de credenciales y de otra manera actúa como un solicitante de estas.

Si el cliente ya ha establecido un único inicio de sesión con el CAS, el navegador web presenta al CAS un cookie seguro que contiene una cadena identificando un TGT (ticket-granting ticket). Este cookie es llamado cookie ticket-granting. Si el cookie ticket-granting define un TGT válido, CAS emitirá un ticket de servicio siempre que todos los demás requisitos en esta especificación se cumplan.

Parámetros

Los siguientes parámetros de solicitud HTTP pueden ser pasados a /login mientras éste esté actuando como un solicitante de credencial. Todos ellos son case-sensitive (es decir, tienen la capacidad de distinguir entre mayúsculas y minúsculas), y todos ellos deben ser manipulados por /login.

service[OPCIONAL] – El identificador de la aplicación a la cual el cliente está tratando de acceder. En casi todas las clases, esta será la URL de la aplicación. Tenga en cuenta que como un parámetro de solicitud HTTP, este valor debe ser una URL codificada como se describe en la sección 2.2 del RFC 1738. Si un servicio no está especificado y una sesión sign-on ya existe, CAS DEBE mostrar un mensaje notificando al cliente que ya está conectado.

renew[OPCIONAL] – Si este parámetro está establecido, el single sign-on será dejado de lado. En este caso, CAS requerirá que el cliente presente credenciales a pesar de la existencia de una sesión SSO (single sign-on) con CAS. Este parámetro no es compatible con el parámetro “gateway”. La reorientación de los servicios a la URI /login y las visitas al formulario de inicio de sesión informando a la URI /login NO DEBEN establecer ambos parámetros de solicitud (“renew” y “gateway”). El comportamiento no está definido si ambos parámetros están establecidos. Se recomienda que las implementaciones de CAS ignoren al parámetro “gateway” si “renew” está establecido. Se recomienda que cuando el parámetro “renew” esté establecido, su valor sea “true”.

gateway[OPCIONAL] – Si se establece este parámetro, CAS no preguntará al cliente por las credenciales. Si el cliente tiene una sesión SSO con CAS, o si una SSO puede ser establecida a través de medios no interactivos (es decir, autenticación de confianza), CAS puede redirigir al cliente a la URL especificada por el parámetro “service”, anexando un ticket de servicio válido. (CAS también puede interponer una página de aviso informando al cliente que una autenticación CAS ha tenido lugar.) Si el cliente no tiene una SSO individual con CAS, y una

autenticación no interactiva no puede ser establecida, CAS DEBE redireccionar al cliente a la URL especificada por el parámetro “service” si el parámetro “ticket” adjunto a la URL. Si el parámetro “service” no está especificado y “gateway” está establecido, el comportamiento de CAS es indeterminado. Se recomienda que, en este caso, CAS solicite credenciales como si ninguno de los parámetros hubiese sido especificado. Este parámetro no es compatible con el parámetro “renew”. El comportamiento no está definido si ambos están establecidos. Se recomienda que cuando el parámetro “gateway” esté establecido, su valor sea “true”.

Ejemplos URL de /login

Ejemplo de inicio de sesión simple:

```
https://server/cas/login?service=http%3A%2F%2Fwww.service.com
```

No solicita usuario/contraseña:

```
https://server/cas/login?service=http%3A%2F%2Fwww.service.com&gateway=true
```

Siempre solicita usuario/contraseña:

```
https://server/cas/login?service=http%3A%2F%2Fwww.service.com&renew=true
```

Respuesta para la autenticación de usuario/contraseña

Cuando /login se comporta como un solicitante de credencial, la respuesta variará dependiendo del tipo de credenciales que éste está solicitando. En la mayoría de casos, CAS responderá mostrando una pantalla de inicio de sesión solicitando un usuario y su respectiva contraseña.

Esta página DEBE incluir un formulario con los parámetros “username”, “password”, y “lt”. El formulario puede también incluir el parámetro “warn”. Si “service” fue especificado para /login, “service” deberá también ser un parámetro en el formulario, conteniendo el valor originalmente pasado a /login. El formulario

debe ser enviado a través del método POST HTTP hacia /login, el cual entonces actuará como un receptor de credenciales.

Respuesta para la autenticación de confianza

La autenticación de confianza satisface la consideración de aspectos arbitrarios de la solicitud como una base para la autenticación. La experiencia de usuario apropiada para la autenticación de confianza será altamente específica para la implementación en consideración de la política local y de las logísticas de los mecanismos de autenticación particular implementados.

Cuando /login se comporta como un solicitante de credenciales para la autenticación de confianza, su comportamiento será determinado por el tipo de credenciales que va a recibir. Si las credenciales son válidas, CAS puede redirigir transparentemente al usuario hacia el servicio. Alternativamente, CAS puede mostrar en la pantalla una alerta de que las credenciales fueron presentadas y permitir que el cliente confirme que quiere usar esas credenciales. Se recomienda que las implementaciones de CAS permitan al implementador escoger el comportamiento preferido. Si las credenciales son inválidas o no existen, se recomienda que CAS muestre en pantalla al cliente la razón por la cual la autenticación falló, y posiblemente presentar al usuario medios alternativos de autenticación (como por ejemplo, la autenticación usuario/contraseña).

ANEXO # 7



MANUAL DE INSTALACIÓN DE JOOMLA!

INTRODUCCIÓN

Joomla! es un sistema de gestión de contenidos, que entre sus principales virtudes permite editar el contenido de un sitio web de manera sencilla. Es una aplicación de código abierto programada en su mayoría en PHP bajo una licencia GPL.



Joomla!™
GRÁFICO 70: Logo de Joomla!

Fuente: <http://www.joomla.org>

Este administrador de contenidos puede trabajar en Internet o intranets y requiere de una base de datos MySQL, así como, preferiblemente, de un servidor HTTP Apache. En Joomla! se incluyen características como: mejorar el rendimiento web, versiones imprimibles de páginas, flash con noticias, blogs, foros, polls (encuestas), calendarios, búsqueda en el sitio web e internacionalización del lenguaje.

REQUISITOS DE INSTALACIÓN

Para una instalación y puesta en marcha de un sitio web con Joomla!, es necesario contar con un servidor web totalmente operativo, el más recomendado es Apache, una base de datos MySQL y un lenguaje de programación web instalado (PHP), todos ellos juntos configurados correctamente. La versión mínima requerida para esos componentes es la siguiente:

Software	Versión Mínima	Web Site
PHP	4.3.x	http://php.net
MySQL	3.23.x o superior	http://dev.mysql.com/downloads/mysql/5.0.html
Apache	1.3 o superior	http://httpd.apache.org

CUADRO 51: Versiones mínimas de los requisitos de Joomla!

Fuente: Wallace(2007) Joomla! 1.5 Installation Manual

Hay que asegurarse de que la funcionalidad de los módulos MySQL, XML, y Zlib se encuentren activados dentro de la configuración del servidor web Apache. Estos están incluidos dentro del archivo php.ini. Se recomienda no utilizar PHP 4.4.2 o PHP 5.0.4 debido a que estas versiones tienen bugs que interfieren en el proceso de instalación de Joomla!.

INSTALACIÓN DE JOOMLA!

El siguiente procedimiento fue realizado en un equipo con OpenSUSE 11,2, el cual tenía instalados y configurados todos los requisitos de instalación.

DESCARGA DEL PAQUETE DE INSTALACIÓN

Se debe ubicar el paquete de instalación en el sitio de Joomla! en español (<http://www.joomlaspanish.org/>).

The screenshot shows the Joomla! Spanish website interface. At the top, there is a navigation bar with links for PORTAL, AYUDA, FOROS, and PACKS. Below the navigation bar, the main content area is divided into several sections. On the left, there is a 'Menú principal' (Main Menu) with links to Inicio, Noticias, Foros, Centro de Ayuda, Manual de Usuario Online, Centro de Extensiones, Asociación Joomla!Spanish, Enlaces web, and Contactar. The central section is titled 'Packs disponibles en Joomla! Spanish' and features a grid of buttons for Joomla! 1.5.15, Idiomas 1.5, Joomla! 1.0.x, and JS U-Server. Below these buttons, there is a detailed section for 'Joomla! Spanish 1.5.15 estable-Full' with a version number of [Wojmamni Ama Mamni] 05-11-2009. This section lists three download packs: 'Pack Joomla-1.5.15-spanish.zip', 'Pack Joomla-1.5.15-spanish.tar.gz', and 'Pack Joomla-1.5.15-spanish.tar.bz2'. To the right of this section, there is a 'Packs Actualizaciones 1.5.x a 1.5.15' section with a link to 'Actualizar a Joomla!1.5.15-Spanish'. On the far right, there are promotional banners for 'Joomla! Day' (Joomla!Day Spain BCN 2009), 'Demo1.5 Joomla! Spanish', and 'Packs Joomla! 1.5.x'. At the bottom of the screenshot, there is a 'HOSTING JOOMLASPANHISH' advertisement with a logo and text: 'Garantía de devolución 30 primeros días' and 'CPANEL+CENTOS+PACK JOOMLASPANHISH+BACKUP'.

GRÁFICO 71: Descarga del paquete de instalación

Fuente: <http://www.joomlaspanish.org>

Una vez descargado el paquete, se descomprime y se ubica el contenido en la carpeta raíz del servidor web o en una subcarpeta. Desde el navegador empezamos el proceso de instalación, accediendo a la carpeta installation, si estamos ejecutando localmente la instalación accedemos mediante <http://localhost/installation>.

SELECCIÓN DEL IDIOMA

El primer paso en la instalación es la selección del lenguaje que será utilizado durante todo el proceso de instalación. Igual que cualquier asistente de instalación de software, seleccionamos el botón siguiente para continuar con el proceso.



GRÁFICO 72: Selección del idioma

Fuente: Autores

COMPROBACIÓN PREVIA

En este punto el instalador realiza una verificación completa de todos los requisitos de instalación, resaltando alguno de estos requisitos en el caso de que no haya sido satisfecho. Si por alguna razón uno de los indicadores posee un valor No, deberá realizarse el correctivo necesario para poder proseguir con el proceso de instalación.

En caso de no realizar el correctivo, Joomla! no podría funcionar de una manera adecuada y podría requerirse de ajustes posteriores a la instalación que no se revisarán en este instructivo.

Pasos

- Idioma
- Comprobación previa**
- Licencia
- Base de datos
- Configuración de FTP
- Configuración
- Finalizar

Comprobación previa [Volver a comprobar | Anterior | Siguiente]

Comprobación previa para Joomla! 1.5.9 Production/Stable [Vatani] 9-January-2009

23:00 GMT.
Si alguno de estos elementos está marcado en rojo, debe corregirse. De no hacerlo, la instalación de Joomla! no funcionará correctamente.

Versión de PHP >= 4.3.10	Sí
- Soporte para compresión zlib	Sí
- Soporte para XML	Sí
- Soporte MySQL	Sí
El idioma predeterminado es multibyte (MB)	Sí
Variable de sobrecarga para funciones de cadenas multibyte desactivada	Sí
configuration.php Escriptible	Sí

Valores recomendados:

Éstos son los valores de PHP recomendados para asegurar una completa compatibilidad con Joomla! No obstante, Joomla! funcionará incluso si estos valores no coinciden con los recomendados.

Directiva	Recomendado	Actual
Modo seguro (safe_mode):	Desactivado	Desactivado
Mostrar errores (display_errors):	Desactivado	Activado
Subida de archivos (file_uploads):	Activado	Activado
Comillas mágicas en tiempo de ejecución (magic_quotes_runtime):	Desactivado	Desactivado
Registrar globales (register_globals):	Desactivado	Desactivado

GRÁFICO 73: Verificación de los requisitos de instalación

Fuente: Autores

El segundo bloque desplegado en la pantalla, muestra un conjunto de configuraciones propias de PHP. Aunque estas configuraciones no estén ajustadas con el valor recomendable, se podrá instalar y configurar un sitio en Joomla! de manera local.

En caso se necesitar cambiar alguno de estos valores, se requerirá cerrar el navegador, realizar los cambios e iniciar nuevamente la instalación.

Una vez que todo este correcto proseguimos dando clic en el botón Siguiente.

ACEPTACIÓN DE LA LICENCIA

En esta pantalla se muestra el detalle de la licencia que regula el uso de Joomla! Y sirve para conocimiento del usuario.



GRÁFICO 74: Aceptación de la Licencia

Fuente: Autores

CONFIGURACIÓN DE LA BASE DE DATOS

El siguiente paso solicita el ingreso de los parámetros de conexión a la base de datos que se utilizará por Joomla! al almacenar la información del sitio.

Es necesario que la base haya sido previamente definida.

Entre los valores solicitados se encuentran:

- **Tipo de base de datos.-** Por defecto será MySQL.
- **Nombre del servidor.-** Se puede colocar localhost, a menos que la base se encuentre físicamente ubicada en otro equipo diferente al servidor web.
- **Nombre del usuario.-** Nombre de un usuario creado en MySQL con el cual se accederá a la base de datos.
- **Contraseña.-** Clave del usuario.
- **Nombre de la base de datos.-** Nombre de la base de datos donde se almacenará la información del sitio.
- **Configuración avanzada.-** Utilizada en el caso que se requiera respaldar la información de instalaciones previas.

Pasos

- 1: Idioma
- 2: Comprobación previa
- 3: Licencia
- 4: Base de datos
- 5: Configuración de FTP
- 6: Configuración
- 7: Finalizar



Configuración de la base de datos

Parámetros de la conexión:

Configurar Joomla para que funcione en su servidor requiere de cuatro pasos sencillos...

Configuración básica

1. Seleccionar el tipo de base de datos que utilizará en la lista desplegable (generalmente **mysql**)
2. Escribir el nombre del servidor en el que se instalará Joomla.
3. Escribir el nombre de usuario de MySQL, la contraseña y el nombre de la base de datos que utilizará para Joomla!. Estos deben estar previamente configurados en su servidor
4. Configuración avanzada

Si la base de datos contiene tablas de instalaciones anteriores de Joomla!, deberá indicar cómo proceder. Eliminarlas o hacer una copia de seguridad.

Escribir un prefijo con el que comenzarán los nombres de todas las tablas de esta instalación de Joomla!.

Configuración básica

Tipo de base de datos: Normalmente será **MySQL**.

Nombre del servidor: Normalmente **localhost** o un nombre de host provisto por su proveedor.

Nombre de usuario: Puede ser algo como **root** o un nombre de usuario, para la base de datos, asignado por su proveedor.

Contraseña: Por razones de seguridad el uso de una contraseña para la cuenta de la base de datos MySQL es altamente recomendado. Esta es la contraseña para acceder a su base de datos. Es posible que esta información sea predeterminada por su proveedor.

Nombre de la base de datos: Algunos hosts permiten solo una base de datos por cuenta. Si debe instalar más de un sitio Joomla! con una misma base de datos, puede modificar la opción de prefijo de tabla en la sección **Parámetros avanzados** para cada sitio instalado.

GRÁFICO 75: Configuración de la base de datos

Fuente: Autores

CONFIGURACIÓN DE FTP

El siguiente paso permite definir la configuración opcional de la conexión a través del protocolo FTP.

Esto puede requerirse debido a posibles restricciones en los permisos en el sistema de archivos del servidor web y a las restricciones de modo seguro (Safe Mode) de PHP en algunos servidores.

Se debe ingresar el nombre de usuario FTP y la contraseña correspondiente, con acceso al directorio raíz de su sitio web Joomla!; ésta será la cuenta FTP que gestionará las operaciones en el sistema de archivos cuando Joomla! requiera un acceso FTP para completar una tarea.

Por razones de seguridad, es mejor utilizar una cuenta de usuario FTP diferente de la cuenta principal, con acceso únicamente a la instalación de Joomla!.

GRÁFICO 76: Configuración de FTP

Fuente: Autores

CONFIGURACIÓN PRINCIPAL

La siguiente sección permite establecer la configuración principal del sitio. Aquí se define el nombre del sitio web, y los datos de acceso para el usuario administrador del sitio.

Adicionalmente es posible instalar contenido de ejemplo para tener una visión de cómo podría lucir el sitio una vez que esté en funcionamiento con datos ya cargados, también se puede importar contenido de un sitio que va a ser migrado desde una versión anterior de Joomla!, para lo cual debería seguirse un proceso de migración que no está contemplado revisarlo.

GRÁFICO 77: Configuración principal

Fuente: Autores

FINALIZAR INSTALACIÓN

Al darle click al botón siguiente se presentará una notificación de que el proceso ha sido terminado con éxito. Al terminar la instalación se solicita que se elimine el directorio de instalación, por motivos de seguridad.

GRÁFICO 78: Finalizando la instalación

Fuente: Autores

Mientras no se elimine este directorio no se podrá acceder al sitio web, ni a la parte del contenido ni a la parte de administración.

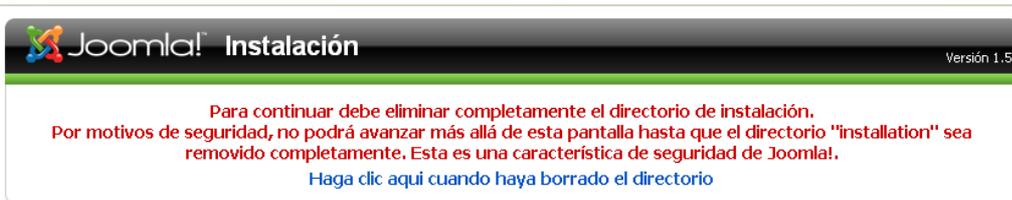


GRÁFICO 79: Bloqueo del acceso al sitio

Fuente: Autores

ANEXO # 8



MANUAL DE USUARIO DE JOOMLA!

INTRODUCCIÓN

Joomla! es un sistema de gestión de contenidos, y entre sus principales virtudes permite editar el contenido de un sitio web de manera sencilla.



GRÁFICO 80: Logo de Joomla!

Fuente: <http://www.joomla.org/>

Es una aplicación de código abierto programada mayoritariamente en PHP bajo una licencia GPL. Este administrador de contenidos puede trabajar en Internet o intranets y requiere de una base de datos MySQL, así como, preferiblemente, de un servidor HTTP Apache.

SECCIONES, CATEGORÍAS Y ARTÍCULOS

Una de las partes más importantes de un sitio web es el contenido. Los artículos pueden ordenarse por dos vías: secciones y categorías.

Las secciones pueden contener gran cantidad de categorías y las categorías pueden contener gran cantidad de artículos.

1. Para agregar una sección al el sitio web se debe acceder a la siguiente dirección: <http://www.upse.edu.ec/administrator> y acceda con el nombre de usuario “admin” y la contraseña respectiva.
2. Para empezar a gestionar secciones, haga clic sobre el gestor de secciones.

- Haga clic sobre “Nuevo” desde la barra de herramientas de Joomla!.
- Introduzca un título y descripción para la sección y luego pulse “Guardar” desde la barra de herramientas.



GRÁFICO 81: Pantalla principal de administración de Joomla.

Fuente: Autores

- Cree tantas secciones como necesite. Use las secciones para agrupar elementos similares o que estén dentro de una misma rama.
- Cuando haya acabado con las secciones, haga clic sobre el menú superior “Contenido” y seleccione el “Gestor de categorías”.

#	Título	Publicado	Ordenar por	Acceso	Sección	# Activo	# Papelera	ID
1	Ciencias Administrativas	✓	1	Público	Estructura Académica	0	8	31
2	Ingeniería Industrial	✓	2	Público	Estructura Académica	5	6	28
3	Ciencias del Mar	✓	3	Público	Estructura Académica	1	3	27
4	Ciencias Agrarias	✓	4	Público	Estructura Académica	1	5	32
5	Vicerrectorado Académico	✓	5	Público	Estructura Académica	8	0	37
6	Escuela de Postgrado y Educación Continua	✓	6	Público	Estructura Académica	2	0	38
7	Facultad de Ciencias Administrativas	✓	7	Público	Estructura Académica	6	0	39
8	Escuela de Informática	✓	8	Público	Estructura Académica	2	0	40
9	Escuela de Hotelería y Turismo	✓	9	Público	Estructura Académica	3	0	41
10	Servicios	✓	1	Público	Estudiantiles	3	0	45
11	Últimas noticias	✓	1	Público	Noticias	43	5	1
12	Destacado	✓	2	Público	Noticias	2	5	3
13	Nosotros	✓	1	Público	Sobre la Universidad	3	3	25
14	El Rector	✓	2	Público	Sobre la Universidad	1	15	29
15	La comunidad	✓	3	Público	Sobre la Universidad	0	2	30

GRÁFICO 82: Gestor de Categorías.

Fuente: Autores

7. Para crear categorías, siga los mismos pasos que con las secciones, asignando a cada una, alguna de las secciones creadas anteriormente. Use las categorías para agrupar cosas que formen parte de una misma jerarquía:

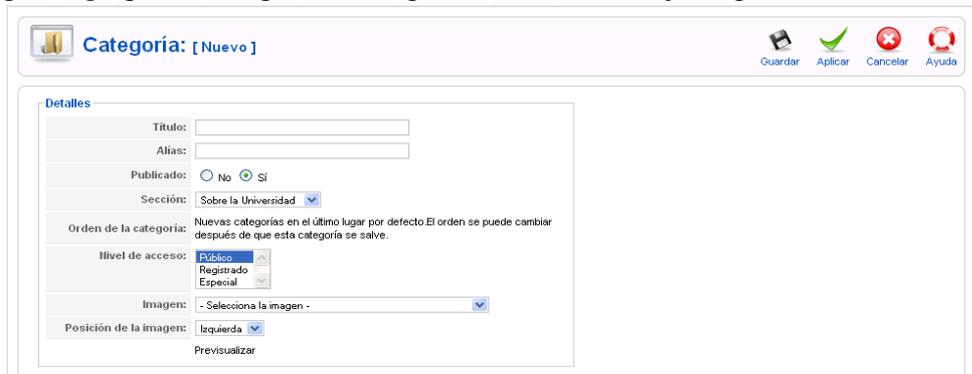


GRÁFICO 83: Creación de una nueva Categoría.

Fuente: Autores

8. Finalmente, necesitamos crear algunos artículos. Haga clic sobre “Contenido” y luego sobre “Gestor de artículos” y seleccione “Nuevo” desde la barra de herramientas.
9. Esto es la pantalla de artículos. Hay un montón de cosas distintas que puede hacer desde aquí:
- Agregue un título al artículo y póngalo dentro de una sección y categoría.
 - Tiene la opción de publicar o despublicar tu artículo, así como la de poder mostrar, o no, el artículo en la página principal.
 - El campo de texto grande es para el contenido del artículo.
 - Puede usar el botón “**Paginar**” para dividir el artículo en páginas.
 - Puede usar el “**Leer más**” para dividir el artículo de tal modo que primero se muestre un texto de introducción y luego, tras pulsar sobre el “Leer más”, se pueda acceder al texto completo.
 - Puede usar el botón “**imagen**” para insertar una imagen en el artículo, usando un simple formulario de subida que está combinado con el gestor multimedia de Joomla!.
10. Cuando acabe de escribir un artículo, haga clic sobre “**Guardar**” para guardarlo y regresar a la página de inicio, o, haga clic sobre “Aplicar”, para guardarlo sin salir de la edición. Si selecciona “Sí” para el parámetro “Página principal”, su artículo se mostrará en la página de inicio.

GESTIÓN DE USUARIOS

Para cada usuario que necesite poder editar el sitio, necesitamos crearles cuentas de acceso.



GRÁFICO 84: Acceso a la opción de gestión de usuarios.

Fuente: Autores

1. Primero, vaya a <http://www.upse.edu.ec/administrator> y acceda con el nombre de usuario “**admin**” y la contraseña respectiva. Si ya está en la administración, vaya al menú superior “**Sitio**” y seleccione “**Panel de control**”.
2. Haga clic sobre “**Gestor de usuarios**”.
3. Haga clic sobre “**Nuevo**” desde la barra de herramientas de Joomla!.
4. Introduzca los detalles para cada usuario que desee crear.

A screenshot of the Joomla! user creation form. The form is titled 'Usuario: [Nuevo]'. It is divided into two main sections: 'Detalles del usuario' and 'Parámetros'. The 'Detalles del usuario' section includes fields for 'Nombre', 'Nombre de usuario', 'Email', 'Nueva contraseña', and 'Verificar contraseña'. There is a dropdown menu for 'Grupo' with options like 'Público Front-end', 'Registrado', 'Autor', 'Editor', 'Publicador', 'Público Back-end', 'Gestor', 'Administrador', and 'Super Administrador'. Below these are radio buttons for 'Bloquear usuario' and 'Recibir e-mails del sistema'. The 'Parámetros' section includes dropdown menus for 'Idioma para la administración', 'Idioma del sitio(front-end)', 'Editor del usuario', 'Sitio de ayuda', and 'Zona horaria'. At the bottom, there is a section for 'Información del contacto' with a message: 'No hay detalles de contacto enlazados a este usuario. Accede a: Componentes -> Contactos -> Gestor de contactos, para ver los detalles.'

GRÁFICO 85: Formulario de creación de usuarios.

Fuente: Autores.

5. Pulse sobre “**Guardar**” cuando acabe de crear a cada usuario.
6. A continuación, podemos configurar una vía para que los visitantes del sitio puedan contactar con el personal de la institución. Desde el menú superior “Componentes” seleccione “Contactos” y luego “Categorías”.
7. Cree una nueva categoría llamada “UPSE” y pulse “Guardar”.
8. Haga clic sobre la pestaña “Contactos” y luego pulse “Nuevo”.
9. Cree los contactos que necesite.

Se debe asegurar que incluya la dirección de correo electrónico y posición dentro de la institución de cada contacto. Ubique a cada usuario dentro de la categoría “UPSE” que habíamos creado anteriormente y enlace a cada contacto con su correspondiente usuario que también anteriormente habíamos creado:

The screenshot shows a web interface for creating a new contact. At the top, there's a title 'Contacto: [Nuevo]' and navigation icons for 'Guardar', 'Aplicar', 'Cancelar', and 'Ayuda'. The main form is split into two columns. The left column, 'Detalles', contains: 'Nombre' (text input), 'Alias' (text input), 'Publicado' (radio buttons for 'No' and 'Sí'), 'Categoría' (dropdown menu), 'Enlazado a un Usuario' (dropdown menu), and 'Ordenar' (text input with a note: 'Los nuevos contactos por defecto se pondrán en el último lugar. El orden puede ser cambiado después de que este contacto se guarde.'). Below this is 'Nivel de acceso' with a dropdown menu showing 'Público', 'Registrado', and 'Especial'. The right column, 'Información', contains: 'Posición del contacto', 'Correo', 'Calle/Dirección', 'Pueblo/Ciudad', 'Estado/País', 'Código postal/ZIP', 'País', 'Teléfono', 'Número del teléfono móvil', 'Fax', 'URL Web', 'Información general', and 'Imagen del contacto'. To the right of these fields is a 'Parámetros' sidebar with a list of fields and radio buttons to toggle their visibility (e.g., 'Nombre' with 'Ocultar' and 'Mostrar' options).

GRÁFICO 86: Creación de un nuevo contacto.

Fuente: Autores.

GESTIÓN DE ENLACES WEB

Para crear enlaces a otros sitios web seguimos el siguiente procedimiento:

1. Primero, vamos a <http://www.upse.edu.ec/administrator> y accedemos a la administración con el nombre de usuario “admin” y la contraseña respectiva.

2. Desde el menú superior “**Componentes**”, seleccionamos “**Enlaces web**” y luego “**Categorías**”.
3. Pulsamos “**Nuevo**” para crear una nueva categoría, luego de llenar el formulario debemos pulsar el botón “**Guardar**”.
4. Hacemos clic sobre la pestaña “**Enlaces**”.
5. Hacemos clic sobre “**Nuevo**” para crear un nuevo enlace. Debemos pulsar el botón “**Guardar**” para guardar el enlace. Estos son ejemplos de algunos enlaces web:

<http://joomla.org>

<http://www.comunidadjoomla.org>

<http://www.google.com>

GRÁFICO 87: Creación de Enlaces Web.

Fuente; Autores.

GESTIÓN DE NOTICIAS EXTERNAS

Para crear una sección que muestre noticias externas desde otros sitios se sigue el siguiente procedimiento:

1. Primero, vamos a <http://www.upse.edu.ec/administrator> y accedemos a la administración con el nombre de usuario “admin” y la contraseña respectiva.
2. Desde el menú superior “**Componentes**”, seleccionamos “**Servidor de noticias externas**” y luego “**Categorías**”.

- Pulsamos el botón “**Nuevo**” y creamos una categoría para “**Noticias Externas**” y luego pulsamos el botón “**Guardar**”.
- Hacemos clic sobre la pestaña “**Servidores**”.
- Hacemos clic sobre “**Nuevo**” para agregar un servidor de noticias externas. Podemos buscar en internet varios servicios RSS con información de nuestro interés y luego, agregamos la URL del RSS en el campo “Enlace”. Debemos asegurarnos que el servidor se encuentre en la categoría respectiva y elegimos el número de artículos a mostrar para este servidor de noticias externas. Algunos ejemplos de servicios de noticias externos pueden ser:
http://www.joomla.org/index.php?option=com_jd-wp&Itemid=105&feed=rss2
<http://comunidadjoomla.org/index.php?format=feed&type=rss>
- Hacemos clic sobre “**Guardar**” cuando esté todo listo.

GRÁFICO 88: Creación de un nuevo servidor de noticias.

Fuentes: Autores.

GESTIÓN DE MENÚS

Un sitio web no será demasiado bueno mientras que no tenga un menú que ayude a los visitantes a navegar por las distintas páginas. Joomla! tiene un sistema integrado para gestionar menús. Este sistema está construido alrededor de tres tipos de información a mostrar en Joomla! — componentes, módulos y plugins—. Los componentes son la información principal en Joomla! y están conducidos por menús. Si se observa en la página principal del sitio de la UPSE, se puede ver solo artículos que hayan sido publicados para salir en la página principal. La

información de contacto, los enlaces web, las noticias externas y los artículos no publicados, no son visibles desde la página principal. Esto es porque cada página muestra un solo componente a la vez. A continuación mostramos como crear algunas opciones de menú:

MENÚ PRINCIPAL

1. Primero, vamos a <http://www.upse.edu.ec/administrator> y accedemos a la administración con el nombre de usuario “admin” y la contraseña respectiva.
2. Desde el menú superior “**Menús**” seleccionamos “**Menú principal**”
3. En este menú está por defecto el enlace “**Inicio**”(Home), el cual, proporciona acceso a los artículos que tengamos publicados en la “**Página principal**”. Podemos hacer clic sobre “**Inicio**” para editar muchos de sus parámetros, incluyendo el nombre de la página de inicio o número de artículos a mostrar.
4. Hacemos clic sobre “**Inicio**” para configurar la página principal.
5. En el área de los “**Parámetros – Básico**” hay un número de opciones diferentes que podemos configurar.

“**# Destacando**” es el número de artículos destacados a mostrar en la página de inicio. Estos artículos ocupan todo el ancho para el marco de noticias y se cortan con un “Leer más”.

“**# Introducciones**” es el número de introducciones de artículo a mostrar en la página principal con un enlace “**leer más**” para poder ver el artículo al completo. Estos artículos toman solo el ancho de la columna (a definir más abajo) y no el de la página entera.

“**Columnas**” es el número de columnas con las que se muestran las introducciones de los artículos (definido más arriba).

“**# Enlaces**” es el número de artículos a mostrar, solo como enlaces, al final de la página.

6. En el área de “**Parámetros – Avanzados**” debemos asegurarnos de que “**Mostrar un enlace a la noticia externa**” esté habilitado. Esto hará que la página principal sea accesible por un servidor RSS.

- En “**Parámetros – Sistema,**” introducimos un título para la página dentro del campo “**Título de la página**”. Esto se mostrará tanto en el encabezado del navegador como en el de esa página dentro del documento. Para este sitio web, introduce “**Universidad Estatal Península de Santa Elena - UPSE**” reemplazando el texto “**Bienvenido a la página de inicio**”.

The screenshot shows the Joomla! administration interface for configuring a menu item. The title is "Ítem del menú: [Editar]". The main configuration area is titled "Presentación de la página principal del Blog" and includes a description: "La página principal estandar del blog, muestra artículos que han sido configurados como artículos de la página principal, en formato Blog." Below this, the "Detalles del ítem del menú" section contains fields for ID (1), Title (Inicio), Alias (home), and Enlace (index.php?option=com_content&view=frontpa). The "Mostrar dentro:" dropdown is set to "Menú principal". The "Insertar ítem:" dropdown is open, showing a list of categories including "Superior", "Noticias", "Nosotros", "Reseña Histórica", "Ley de Creación", "Filosofía, Misión, Visión y Objetivos", "El Rector - Hoja de Vida", "Galería de Autoridades", "Ubicación", and "Estructura Académica". The "Publicado:" section has radio buttons for "No" and "Sí" (selected). The "Ordenar:" dropdown is set to "13 (Inicio)". The "Nivel de acceso:" dropdown is set to "Público". The "Al hacer click, abrir dentro:" dropdown is set to "Misma ventana con barra de navegación". On the right side, there are several parameter sections: "Parámetros - Básico" (with fields for #Destacando: 1, #Introducción: 4, Columnas: 2, #Enlaces: 4), "Parámetros - Avanzados", "Parámetros - Componente", "Parameters (Extended Query Strings)", and "Parámetros - Sistema". At the top right, there are icons for "Guardar", "Aplicar", "Cerrar", and "Ayuda".

GRÁFICO 89: Menú "Inicio".

Fuente: Autores.

- Hacemos clic en “**Guardar**” para finalizar la configuración de la página principal.

PRESENTACIÓN DE CATEGORÍAS

Podemos crear opciones de menú para diversas categorías mediante el siguiente procedimiento:

- Desde el menú superior “**Menús**”, seleccionamos “**Menú principal**”
- Hacemos clic en “**Nuevo**”, seleccionamos “**Artículos**” y luego “**Presentación, en formato blog, de los artículos de una categoría**”.



GRÁFICO 90: Selección de un nuevo ítem de menú para una categoría.

Fuente: Autores.

- Desde el área de “**Parámetros – Básico**” seleccionamos una categorías creadas previamente. Llamamos al elemento del menú de manera acorde a la categoría seleccionada. Ajustamos los parámetros de un modo similar al

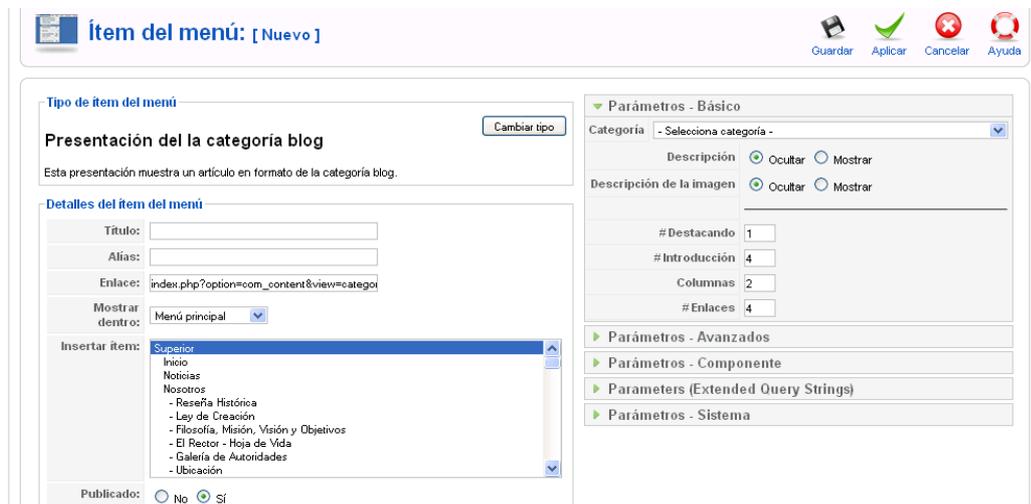


GRÁFICO 91: Creación de un nuevo ítem de menú para una categoría.

Fuente: Autores.

“**Inicio**” y debemos asegurarnos de incluir un enlace RSS a la noticia desde “**Parámetros – Avanzado**”. Damos un título a la página desde “**Parámetros – Sistema**”. También podríamos cambiar el “**Orden primario**” a “**Título – Alfabético**” en “**Parámetros-Avanzados**”:

- Hacemos clic en “**Guardar**”, con lo que la opción se añadirá al menú seleccionado.

PRESENTACIÓN DE CONTACTOS



GRÁFICO 92: Selección de un nuevo ítem de menú para los Contactos.

Fuente: Autores.

Para agregar una opción denominada “**Contáctenos**”:

1. Desde el menú superior “**Menús**”, seleccionamos “**Menú principal**”:
2. Seleccionamos “**Nuevo**” y luego hacemos clic sobre “**Contactos**” seguido de “**Presentación de los contactos de una categoría**”:
3. Llamamos a la página “**Contáctenos**” y seleccionamos la categoría deseada. No hay necesidad de servir noticias externas desde esta página, por lo tanto, deshabilitamos esa opción. Desde “**Parámetros – Sistema**” y debemos asegurarnos de llamar a la página “**Contáctenos**”:

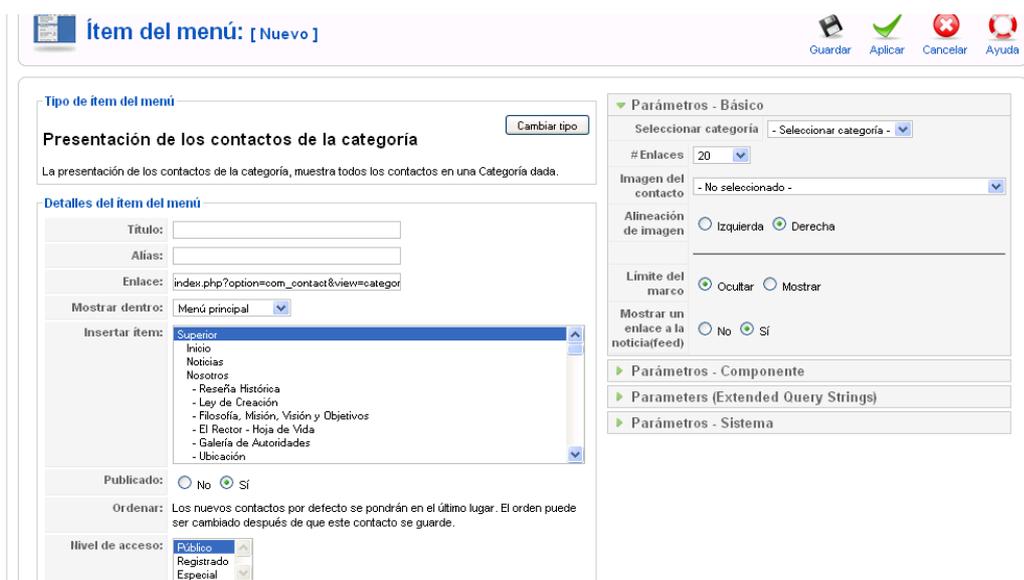


GRÁFICO 93: Creación de una opción de menú para los contactos.

Fuente: Autores.

4. Hacemos clic en “**Guardar**” para reflejar los cambios en el sitio.

PRESENTACIÓN DE ENLACES WEB

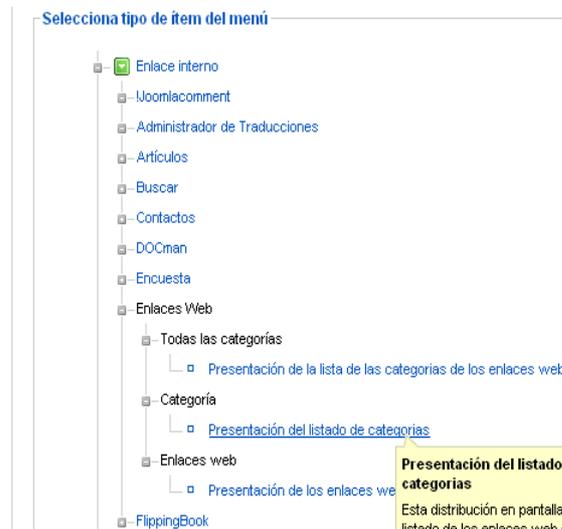


GRÁFICO 94: Selección de un nuevo ítem de menú para los Enlaces Web.

Fuente: Autores.

Nuestro próximo componente a acceder desde el sistema de menús es el de “**Enlaces web**”:

1. Desde el menú superior “**Menús**”, seleccionamos “**Menú principal**”
2. Hacemos clic sobre el botón “**Nuevo**”, seleccionamos “**Enlaces web**” y luego “**Presentación, en formato lista, de los enlaces de una categoría**”:
3. Llamamos a la página “**Enlaces web**” y seleccionamos la categoría deseada. No hay necesidad de servir noticias externas desde esta página, así que deshabilitamos la opción desde “**Parámetros – Sistema**” y nos aseguramos de que la página se llama “**Enlaces web**”:

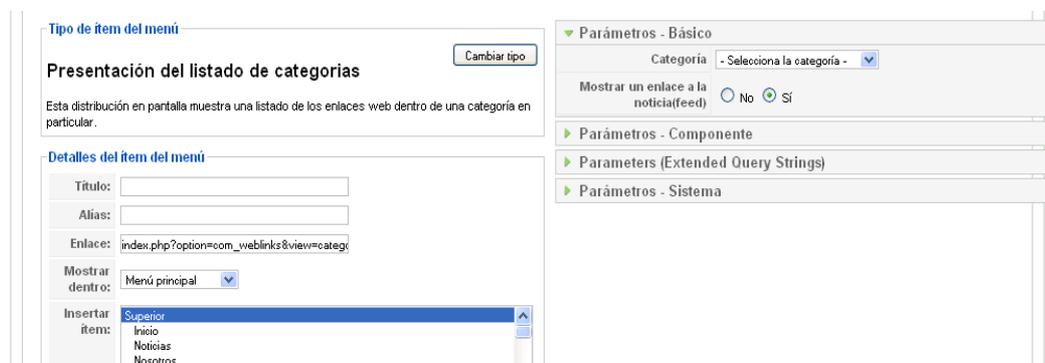


GRÁFICO 95: Creación de un nuevo ítem de menú para los Enlaces Web.

Fuente: Autores.

4. Hacemos clic en “**Guardar**” para almacenar los cambios.

PRESENTACIÓN DE NOTICIAS EXTERNAS

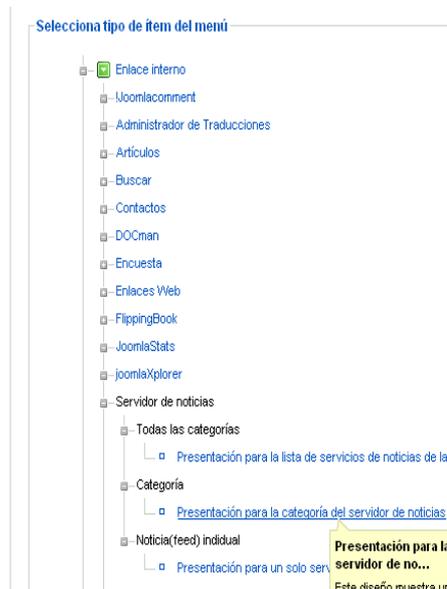


GRÁFICO 96: Selección de un nuevo ítem de menú para Noticias Externas.

Fuente: Autores.

Otro elemento del menú que podemos agregar es el de “**Noticias externas**”:

1. Desde el menú superior, seleccionamos “**Menú principal**”.
2. Hacemos clic sobre “**Nuevo**”, seleccionamos “**Servidor de noticias externas**” y a continuación “**Presentación de una categoría de servidores de noticias externas**”:
3. Llamamos a la página “**Noticias externas de jardinería**” y seleccionamos la categoría deseada desde “**Parámetros – Sistema,**” y nos aseguramos de que el título de la página es “**Noticias externas de jardinería**”:

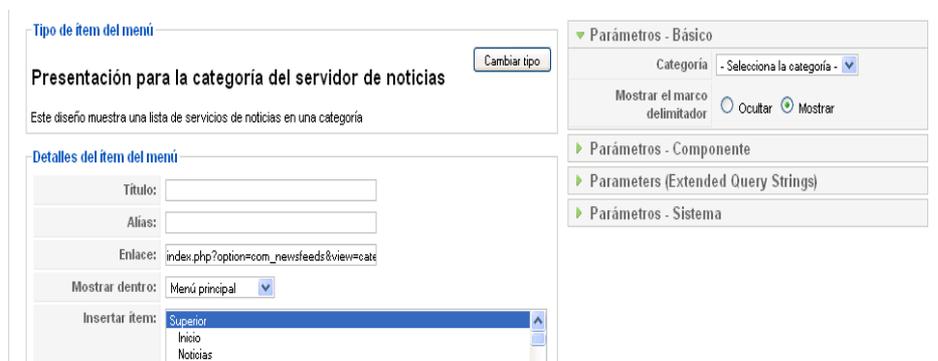


GRÁFICO 97: Creación de un nuevo ítem para las Noticias Externas.

Fuente: Autores.

4. Hacemos clic en “Guardar” para almacenar los cambios.

PRESENTACIÓN DEL BUSCADOR

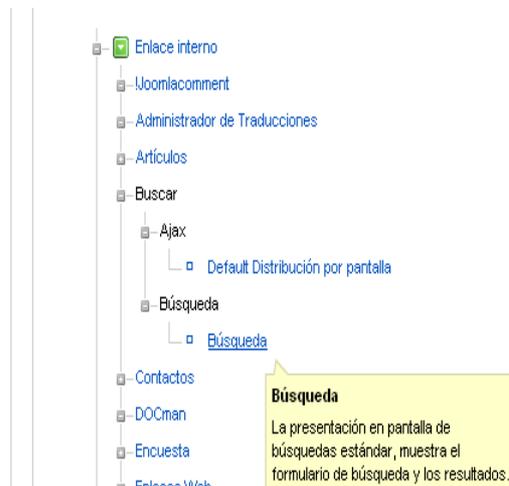


GRÁFICO 98: Selección de un nuevo ítem de menú para la opción **Buscar**.

Fuente: Autores.

Otro elemento del menú que podemos crear es de la página “**Buscar**”:

1. Desde el menú superior, seleccionamos “**Menú principal**”.
2. Hacemos clic sobre el botón “**Nuevo**”, selecciona “**Buscar**” y luego de nuevo “**Buscar**”:
3. Llamamos a la página “**Buscar**” desde los “**Parámetros – Sistema**”, y nos aseguramos de que el título es “**Buscar**”:

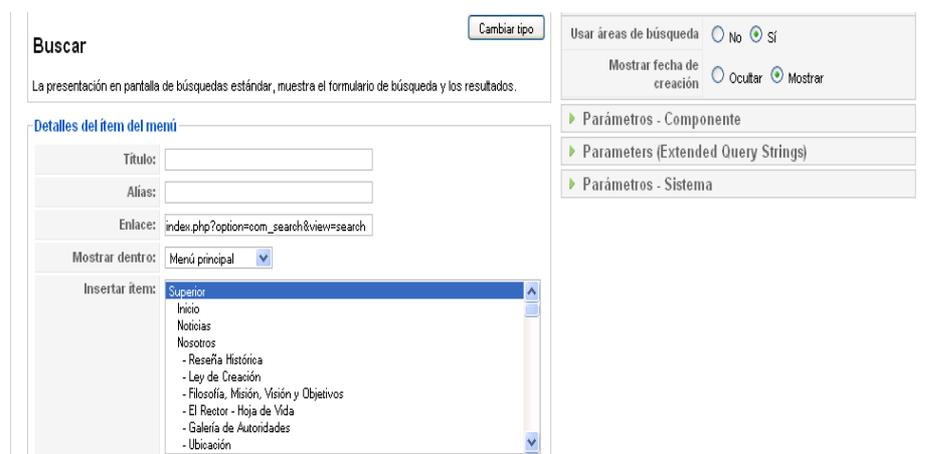


GRÁFICO 99: Creación de un nuevo ítem de menú para la opción **Buscar**.

Fuente: Autores

4. Hacemos clic en “**Guardar**” para almacenar los cambios.

CREACIÓN DE MÓDULOS

El próximo tipo de contenido Joomla! son los módulos. Los módulos son contenedores que aparecen predefinidos alrededor del componente principal. Los nombres de los contenedores pueden ser del tipo “**left**,” “**right**,” o “**footer**.”

El “**Menú principal**” mostrado en la página principal es un ejemplo de módulo. Como ejemplo vamos a agregar varios módulos muy útiles:

1. Primero, vamos a <http://www.upse.edu.ec/administrator> y accedemos a la administración con el nombre de usuario “admin” y la contraseña respectiva.
2. Desde el menú superior “**Extensiones**”, seleccionamos “**Gestor de módulos**”

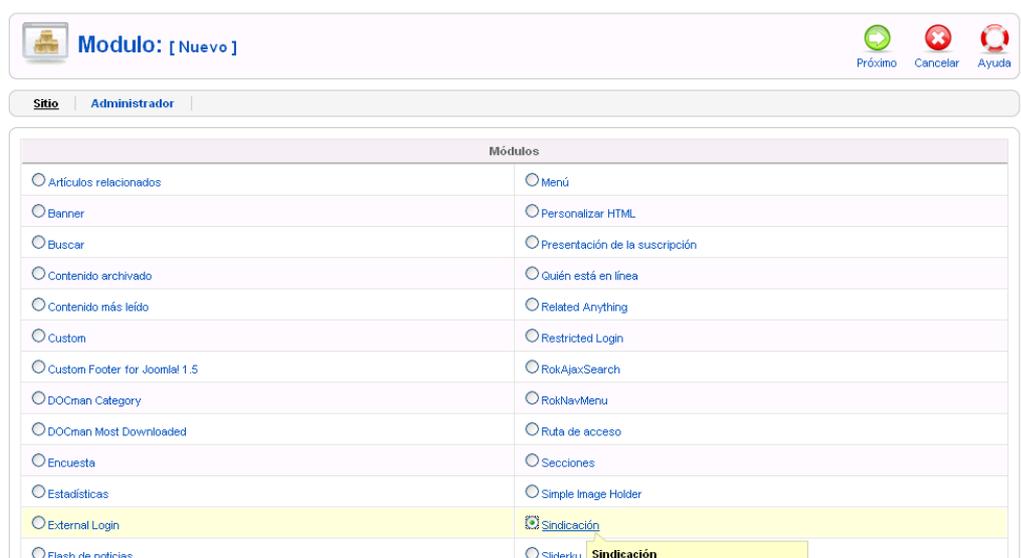


GRÁFICO 100: Creación de un nuevo módulo de sindicación (Paso 1).

Fuente: Autores.

MÓDULO DE SINDICACIÓN

Esta es la página de gestión de módulos. Desde aquí, podemos decidir que módulos se muestran en cada página, primeramente, vayamos a agregar un módulo que muestre noticias RSS:

3. Haz clic sobre el botón “**Nuevo**” y luego selecciona “**Sindicación**” y haz clic sobre “**Siguiente**”.

4. Escribimos un título para el módulo de sindicación. Seleccionamos una posición en la que mostrar tu módulo de sindicación (normalmente “**left**” o “**right**”). Para este módulo, la plantilla predeterminada tiene una sección especial para “**Sindicación**”. Desde la asignación de menús, elegimos de entre la lista “**Seleccionar elemento(s) del menú**” las páginas por donde el módulo de sindicación estará habilitado y se mostrará:

The screenshot shows the Joomla! administrator interface for configuring a module. At the top, there's a header with the module name 'Módulo: [Editar]' and action buttons: 'Guardar', 'Aplicar', 'Cancelar', and 'Ayuda'. The main content area is divided into two panels: 'Detalles' and 'Parámetros'.
The 'Detalles' panel includes:

- Tipo de módulo: mod_syndicate
- Título: (empty text box)
- Mostrar título: Radio buttons for 'No' and 'Sí' (selected).
- Habilitado: Radio buttons for 'No' and 'Sí' (selected).
- Posición: Dropdown menu with 'left' selected.
- Ordenar: Dropdown menu with '0: Ingreso Externo' selected.
- Nivel de acceso: Dropdown menu with 'Público' selected.
- ID: 0
- Descripción: Módulo inteligente de suscripción de noticias que crea una alimentación de noticias por suscripción para la página en la que es mostrado el módulo.

The 'Parámetros' panel includes:

- guardando en la caché: Dropdown menu with 'Nunca' selected.
- Texto: Text input field with 'Feed Entries'.
- formato: Dropdown menu with 'RSS 2.0' selected.
- Sufijo de la clase del módulo: (empty text box).

At the bottom, there's an 'Asignación de menú' section with radio buttons for 'Menús: Todo' (selected), 'Ninguno', and 'Selecciona item(s) del menú desde la lista'. Below it, a dropdown menu shows 'mainmenu' selected.

GRÁFICO 101: Creación de un nuevo módulo de sindicación (Paso 2)

Fuente: Autores.

5. Hacemos clic en “Guardar” para almacenar los cambios.

MÓDULO DE ACCESO

Para agregar una caja de control de acceso, por medio de la cual, los usuarios puedan acceder rápidamente al sitio seguimos el siguiente procedimiento.

1. Desde el menú superior “**Extensiones**”, seleccionamos “**Gestor de módulos**”.
2. Hacemos clic sobre el botón “**Nuevo**” y luego seleccionamos “**Iniciar sesión**” y hacemos clic en “**Siguiente**”:
3. Configuramos este módulo con el nombre “**Formulario de acceso**” y elegimos la posición “**left**”. Esto mostrará al módulo en la parte izquierda de la pantalla principal. Como queremos que este módulo aparezca en todas las páginas no necesitamos cambiar esa configuración.

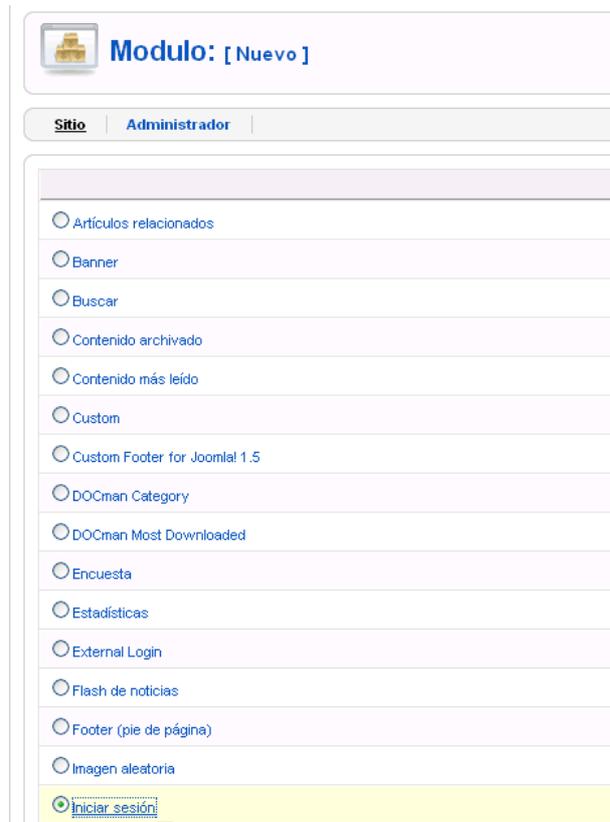


GRÁFICO 102: Creación de un módulo para Inicio de Sesión (Paso 1)

Fuente: Autores.

4. Hacemos clic en “Guardar” para almacenar los cambios.



GRÁFICO 103: Creación de un módulo para Inicio de Sesión (Paso 2)

Fuente: Autores.

MÓDULO DE NOTICIAS EN FLASH

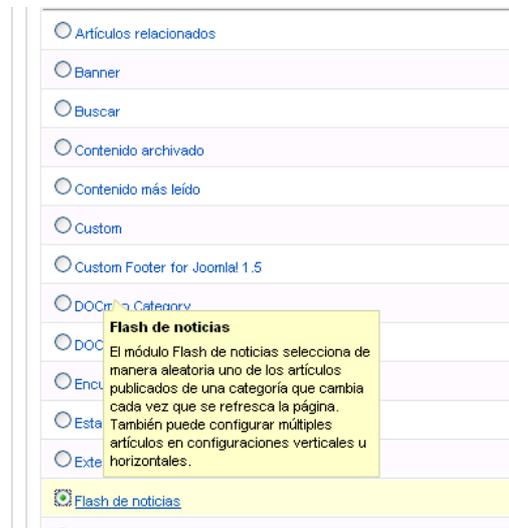


GRÁFICO 104: Creación de un módulo para Noticias Flash (Paso 1).

Fuente: Autores.

Otro módulo que se puede agregar es el de “Noticias en flash”. Este módulo muestra un artículo aleatorio desde una categoría específica.

1. Desde el menú superior “Extensiones”, seleccionamos “Gestor de módulos”.
2. Hacemos clic en “Nuevo” y luego seleccionamos “Noticias en flash” y pulsamos el botón “Siguiente”:
3. Llamamos al módulo “Noticias en flash” y elegimos la posición “Top” con la intención de mostrarlo después del logo. Elegimos la categoría de noticias que nos interese mostrar. Hacemos que el título sea enlazable para que el lector pueda ver el artículo al completo tras pulsar sobre él, y marcamos en “Sí” el “Título del artículo” para mostrar el título del artículo:



GRÁFICO 105: Creación de un módulo para Noticias Flash (Paso 2).

Fuente: Autores.

4. Hacemos clic en el botón “**Guardar**” para almacenar los cambios.

MÓDULO DE RUTAS

Podemos agregar el módulo “Breadcrumbs”, el cual ayuda a los usuarios a navegar por el sitio, mostrando la ruta de las zonas por donde han pasado para llegar adonde están.



GRÁFICO 106: Creación de un módulo para la Ruta de Acceso (Paso 1)

Fuente: Autores.

1. Desde el menú superior “**Extensiones**”, seleccionamos “**Gestor de módulos**”.
2. Hacemos clic sobre “**Nuevo**” y luego seleccionamos “**Ruta**” y pulsa sobre “**Siguiente**”.
3. Llamamos al módulo “**Ruta**”; pero, deshabilitamos “**Mostrar título**” desde la configuración. Asignamos la posición “**Breadcrumbs**” y un texto de entrada para el inicio llamado “**Inicio**”. Dejamos el resto de las opciones igual.



GRÁFICO 107: Creación de un módulo para la Ruta de Acceso (Paso 2)

Fuente: Autores.

4. Hacemos clic sobre el botón “**Guardar**” para almacenar los cambios.

GESTIÓN DE PLANTILLAS

Las plantillas nos permiten definir el aspecto visual del sitio Joomla!. Joomla! viene con tres plantillas, pero podemos encontrar muchas más buscando por Joomla.org.

1. Primero, vamos a <http://www.upse.edu.ec/administrator> y accedemos a la administración con el nombre de usuario “admin” y la contraseña respectiva.
2. Desde el menú superior “**Extensiones**”, seleccionamos el “**Gestor de plantillas**”.

#	Nombre de la plantilla	Predeterminado	Asignado	Versión	Fecha	Autor
1	beez			1.0.0	19 February 2007	Angie Radtke/Robert Deutz
2	JA_Purity			1.2.0	12/26/07	JoomlaArt.com
3	rhuk_milkyway			1.0.2	11/20/06	Andy Miller
4	Upse Azul	★		1.0	01/2010	Universidad Estatal Peninsula de Santa Elena
5	Upse Blanco			1.0	07/2009	Universidad Estatal Peninsula de Santa Elena

GRÁFICO 108: Gestor de Plantillas.

Fuente: Autores.

3. La plantilla seleccionada como predeterminada es la plantilla actual. Para cambiar una plantilla por otra, seleccionamos la plantilla a utilizar y hacemos clic sobre “**Predeterminada**”.

Hacemos clic sobre la opción “rhuk_milkyway” con la intención de editar algunos parámetros de la plantilla.

Plantilla: [Editar]

Detalles

Nombre: **Plantilla RHUK Milkyway**

Descripción: MilkyWay es una plantilla refrescante para Joomla! Su limpio diseño la hace muy ligera y rápida.

Asignación de menú

Menús: Ninguno Selecciona desde la lista

Selección de menú:

- mainmenu
- Inicio
- Noticias
- Nosotros
- Reseña Histórica
- Ley de Creación
- Filosofía, Misión, Visión y Objetivos
- El Rector - Hoja de Vida
- Ubicación
- Estructura Académica
- Facultad de Ciencias Administrativas

Parámetros

El parámetro del archivo `templates/rhuk_milkyway/params.ini` es

Variante del color:

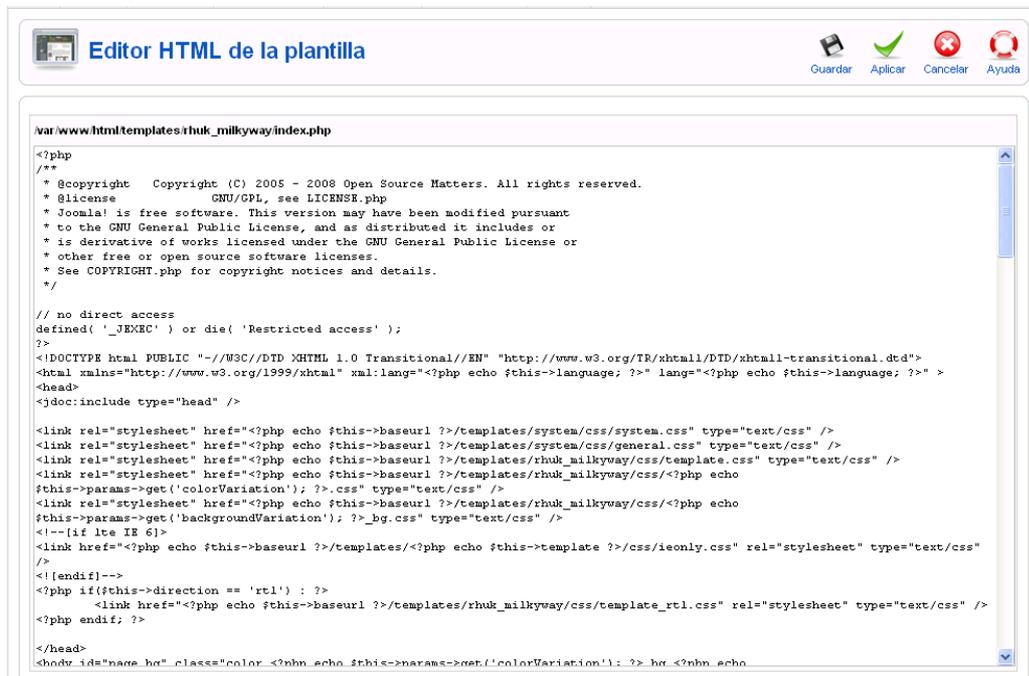
Variación del fondo:

Ancho de la plantilla:

GRÁFICO 109: Opciones de la plantilla "rhuk_milkyway".

Fuente: Autores.

4. A continuación, si necesitamos ajustar más la plantilla, sobre la página de edición, hacemos clic sobre “**Editar HTML**”:



Fuente: Autores.

5. Cada módulo tenía asignado un lugar dentro de un contenedor específico. Estos contenedores, están definidos en la plantilla. Por ejemplo, la posición “**top**” está definida por el código:

```
<jdoc:include type="modules" name="top" />.
```

ANEXO # 9



MANUAL DE INSTALACIÓN DE ZIMBRA

INTRODUCCIÓN

La Suite de Colaboración Zimbra es un programa colaborativo creado por Zimbra Inc., y que posteriormente fue adquirida por Yahoo! Inc. en Septiembre de 2007, haciendo un acuerdo de mantener sus estándares de Software Abierto. A finales del 2009 fue adquirido por VMware. Posee tanto el componente de servidor como su respectivo cliente.



GRÁFICO 111: Logo de Zimbra

Fuente: <http://www.zimbra.com>

Existen varias versiones de Zimbra disponibles: una versión soportada por la comunidad de Software Abierto (Open Source), y otras con parte de código cerrado y soportada comercialmente que contiene algunas mejoras.

Al igual que el programa de correo electrónico Exchange, de Microsoft, Zimbra permite a los empleados de oficina enviar, recibir, guardar y buscar los miles de mensajes procesados cada día.

El Servidor ZCS hace uso de proyectos Open Source existentes como ser: Postfix, MySQL, OpenLDAP. Expone una interfaz de programación de aplicaciones (API) SOAP para toda su funcionalidad y actúa tanto como un servidor IMAP y POP3.

El Cliente Web de ZCS es una interfaz de colaboración y administración completa creada empleando el Toolkit Zimbra Ajax. Este cliente soporta correos

electrónicos y calendarios a través de una impresionante interfaz web Ajax, que provee tips de sus objetos, ítems arrastrables, y menús que se expanden al hacer clic derecho. También incluye capacidades de búsqueda avanzada y permite relacionar fechas.

ZCS es compatible con clientes propietarios tales como Microsoft Outlook, Novell Evolution y Apple Mail. También provee soporte de sincronización nativo para muchos dispositivos móviles (Nokia E-Series, BlackBerry, Windows Mobile, etc).

ARQUITECTURA DEL PRODUCTO

Cuatro son los pilares de la arquitectura del producto:

- **Flexibilidad:** fácil personalización según las necesidades de la organización.
- **Libertad:** uso de cualquier navegador web y de aplicaciones de escritorio tradicionales.
- **Durabilidad:** servidor de correo electrónico y calendario extraordinariamente fiable y ampliable.
- **Bajo coste de mantenimiento:** gestión muy sencilla, tanto mediante una interfaz gráfica como desde la consola.

ZCS se divide en tres módulos principales:

- Cliente web.
- Servidor.
- Extras.

Los componentes que hay dentro de una distribución de Zimbra son:

- **Zimbra Core:** contiene las utilidades, librerías y archivos de configuración. Este paquete se instalará por defecto en todos los servidores.

- **Zimbra LDAP:** Contiene el sistema de autenticación bajo OpenLDAP para almacenar y gestionar el almacén de usuarios, integrando de serie el soporte para la replicación. Además, permite fácilmente su configuración para el uso de directorios LDAP externos, incluyendo Active Directory de Microsoft o Directory de Novell, entre otros.
- **Zimbra MTA:** (Mail Transfer Agent) contiene el servidor de Envío y recepción de correo en Postfix. También contiene el Spamassassin y ClamAV. El MTA puede integrarse con otras tecnologías, como Postgrey para greylisting o Spamhaus para DNSBL, Asimismo, soporta por defecto los protocolos principales de cifrado de canal, SSL y TLS
- **Zimbra Store:** contiene todos los buzones implementados con el Jetty, MySQL, Apache, etc.

Otro componente integrado en Zimbra es Lucene, un potente motor de indexación y búsquedas que permite a los usuarios y administradores buscar mensajes a través de múltiples carpetas de correo, tanto metadatos como contenidos en el cuerpo del mensaje.

Cada servidor de Zimbra tiene su propio almacén de datos, mensajes y de índices para los buzones de ese servidor. En cuanto llega un correo, el servidor de Zimbra crea un nuevo proceso para indexar el mensaje. También se crea un hilo para la conversión de adjuntos a formato HTML.

El almacén de datos es una base de datos MySQL. El almacén de datos relaciona el identificador del buzón con la cuenta de usuario a la que pertenece en el directorio LDAP.

El almacén de mensajes guarda todos los mensajes y sus adjuntos en formato MIME (del inglés, Multipart Internet Mail Extension).

CLIENTE WEB

Zimbra incorpora un cliente Web que hace un uso extensivo de AJAX y capaz de ejecutarse en la mayoría de los navegadores Web más habituales, tal como Firefox o Internet Explorer.

Debido al uso intensivo de Javascript, las diferencias en la experiencia del usuario son enormes dependiendo del navegador, siendo Opera y Safari los que brindan una mejor experiencia. En cualquier caso, el cliente Web de Zimbra también permite utilizar una versión sólo con HTML, el cual es muy recomendable para equipos antiguos pero que, de igual manera, ofrece un nivel de usabilidad muy buena.

SOFTWARE BASE DE ZIMBRA

Todo el conjunto de aplicaciones de terceros es:

- Apache + Tomcat (servidor Web).
- Clamav (antivirus).
- Spamassassin + DSpam (filtro de spam).
- Amavis-new (conector entre los antivirus y el servidor de correo).
- Jdk (Maquina Java).
- MySQL (servidor de base datos).
- OpenLdap (servidor de directorio).
- Postfix (servidor de correo SMTP).
- Cyrus (servidor de correo POP/IMAP).

VERSIONES DE ZIMBRA

1. **Open Source Edition:** Totalmente funcional pero con algunos recortes en el interfaz Web y en la herramienta de administración. Este producto, se

ofrece sin garantías y sin opción de adquirir soporte por parte del fabricante.

2. **Network Profesional Edition (ZCS):** La opción profesional, con funciones adicionales tanto en el interfaz Web, como en la herramienta de administración. Este tipo de versión, va acompañado por un soporte del fabricante y unas garantías del producto.

INSTALACIÓN DEL SERVIDOR ZIMBRA

El siguiente procedimiento fue realizado en un equipo con CentOS 5.4 ejecutándose en modo texto.

Pre-requisitos

- Computador o máquina virtual con CentOS 5.4 instalado.
- Conexión a internet para la descarga de los paquetes de instalación.

PROCEDIMIENTO DE INSTALACIÓN

Lo primero que se debe hacer es ubicar la última versión disponible para la distribución Red Hat 5, no existe un paquete disponible para CentOS 5.4, pero al ser una distribución derivada directamente de Red Hat puede utilizarse sin ningún problema.

Lo primero es ubicar la última versión disponible de la suite Zimbra, para esto buscamos en el sitio <http://www.zimbra.com/downloads/os-downloads.html>. En esta página se mostrarán los enlaces de descarga de los paquetes orientados hacia las distintas distribuciones soportadas por Zimbra, entonces copiamos la dirección de descarga del paquete para la plataforma Red Hat Enterprise Linux 5, teniendo en cuenta la plataforma de hardware sobre la que va a realizarse la instalación (32 o 64 bits). Finalmente descargamos desde el servidor CentOS el paquete de instalación con el comando wget:

```
# wget http://h.yimg.com/lo/downloads/6.0.5_GA/zcs-6.0.5_GA_2213.RHEL5.20100202220948.tgz
```

Una vez descargado el paquete de instalación es preciso realizar ciertos cambios de configuración a fin de garantizar un funcionamiento correcto de la suite. Es necesario corregir la configuración de identificación del dominio en el que se encuentra el servidor, esto se realiza a través del archivo hosts ubicado en el directorio etc. A continuación se compara el contenido previo a los cambios y luego de que estos se efectúan:

```
[root@demo ~]# cat /etc/hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1          demo localhost.localdomain localhost
::1               localhost6.localdomain6 localhost6
```

GRÁFICO 112: Configuración previa del archivo /etc/hosts

Fuente: Verónica Mora (2009) Manual Zimbra Collaboration Suite

Se debe agregar la dirección ip del equipo donde se va a instalar zimbra así como el nombre del dominio.

```
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1          localhost.localdomain localhost
#:::1             localhost6.localdomain6 localhost6
10.0.2.15         mail.midominio.com mail
```

GRÁFICO 113: Configuración modificada del archivo /etc/hosts

Fuente: Verónica Mora (2009) Manual Zimbra Collaboration Suite

También se debe verificar que los servicios sendmail y nfs no estén ejecutándose ya que crean conflictos con el uso del puerto 25.

```
# chkconfig sendmail
```

```
# chkconfig nfs
```

En caso de que estén activos hay que parar los servicios y desactivarlos

```
# service sendmail stop
```

```
# service nfs stop
```

```
# chkconfig sendmail off
```

```
# chkconfig nfs off
```

Una vez realizados estos pasos procedemos a iniciar la instalación, es necesario indicar que la instalación debe ser iniciada desde el directorio donde se descargó el paquete de instalación.

Primero se descomprime el paquete

```
# tar xvzf zcs-6.0.5_GA_2213.RHEL5.20100202220948.tgz
```

Una vez hecho esto, accedemos al directorio descomprimido y ejecutamos la instalación, es de notar que al ejecutar la instalación añadimos el parámetro `platform-override`, con esto le estamos indicando al instalador que omita la distribución del sistema operativo, logrando así instalar en CentOS el paquete destinado a Red Hat.

```
# ./install.sh -platform-override
```

En este momento inicia el proceso de instalación, verificando los requisitos de instalación, primeramente que se encuentren instaladas todas las librerías de soporte requeridas por la aplicación.

```
hostname: `Host' desconocido
hostname: `Host' desconocido

Operations logged to /tmp/install.log.3190
Checking for existing installation...
  zimbra-ldap...NOT FOUND
  zimbra-logger...NOT FOUND
  zimbra-mta...NOT FOUND
  zimbra-snmp...NOT FOUND
  zimbra-store...NOT FOUND
  zimbra-apache...NOT FOUND
  zimbra-spell...NOT FOUND
  zimbra-proxy...NOT FOUND
  zimbra-archiving...NOT FOUND
  zimbra-convertd...NOT FOUND
  zimbra-cluster...NOT FOUND
  zimbra-core...NOT FOUND

PLEASE READ THIS AGREEMENT CAREFULLY BEFORE USING THE SOFTWARE.
ZIMBRA, INC. ("ZIMBRA") WILL ONLY LICENSE THIS SOFTWARE TO YOU IF YOU
FIRST ACCEPT THE TERMS OF THIS AGREEMENT. BY DOWNLOADING OR INSTALLING
THE SOFTWARE, OR USING THE PRODUCT, YOU ARE CONSENTING TO BE BOUND BY
THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS
AGREEMENT, THEN DO NOT DOWNLOAD, INSTALL OR USE THE PRODUCT.

License Terms for the Zimbra Collaboration Suite:
http://www.zimbra.com/license/zimbra_public_eula_2.1.html
```

GRÁFICO 114: Verificación de requisitos de instalación de Zimbra

Fuente: Verónica Mora (2009) Manual Zimbra Collaboration Suite

Es muy probable que luego de la verificación, el instalador muestre que falten una o más librerías necesarias para el funcionamiento de Zimbra, más aún si el equipo contiene una instalación limpia del sistema operativo.

```
Press Return to continue

Checking for prerequisites...
FOUND: NPTL
FOUND: sudo-1.6.9p17-3
FOUND: libidn-0.6.5-1.1
FOUND: fetchmail-6.3.6-1.1
MISSING: gmp
MISSING: compat-libstdc++-296
MISSING: compat-libstdc++-33
MISSING: libtool-ltdl
FOUND: /usr/lib/libstdc++.so.6
Checking for suggested prerequisites...
FOUND: perl-5.8.8

###ERROR###

One or more prerequisite packages are missing.
Please install them before running this installer.

Installation cancelled.
```

GRÁFICO 115: Detección de librerías faltantes

Fuente: Verónica Mora (2009) Manual Zimbra Colaboration Suite

El instalador muestra la lista de requisitos indicando cuales son los que faltan (MISSING). Es muy común que entre las librerías que faltan se encuentren las siguientes:

- gmp
- compat-libstdc++-296
- compat-libstdc++-33
- libtool-ltdl

Si se necesita instalar alguna de las librerías listadas previamente, lo hacemos mediante el comando yum, tal y como se aprecia a continuación:

```
# yum install gmp
# yum install compat-libstdc++-296
# yum install compat-libstdc++-33
# yum install libtool-ltdl
```

Una vez realizada la instalación de las librerías, procedemos a iniciar nuevamente el proceso utilizando el mismo comando, con lo cual podemos observar que todos los requisitos para dar inicio a la instalación están completos, pudiendo ahora si seguir adelante.

```
Checking for prerequisites...
FOUND: NPTL
FOUND: sudo-1.6.9p17-3
FOUND: libidn-0.6.5-1.1
FOUND: fetchmail-6.3.6-1.1
FOUND: gmp-4.1.4-10
FOUND: compat-libstdc++-296-2.96-138
FOUND: compat-libstdc++-33-3.2.3-61
FOUND: libtool-ltdl-1.5.22-6.1
FOUND: /usr/lib/libstdc++.so.6
Checking for suggested prerequisites...
FOUND: perl-5.8.8
Prerequisite check complete.

Checking for installable packages

Found zimbra-core
Found zimbra-ldap
Found zimbra-logger
Found zimbra-mta
Found zimbra-snmp
Found zimbra-store
Found zimbra-apache
Found zimbra-spell
Found zimbra-proxy
```

GRÁFICO 116: Inicio de instalación con requisitos completos

Fuente: Verónica Mora (2009) Manual Zimbra Collaboration Suite

El siguiente paso es la selección de los componentes de Zimbra que se desean instalar. Una instalación típica sólo requiere la instalación de los componentes zimbra-ldap, zimbra-logger, zimbra-mta, zimbra-snmp, zimbra-store, zimbra-apache y zimbra-spell.

```
Select the packages to install

Install zimbra-ldap [Y] y
Install zimbra-logger [Y] y
Install zimbra-mta [Y] y
Install zimbra-snmp [Y] y
Install zimbra-store [Y] y
Install zimbra-apache [Y] y
Install zimbra-spell [Y] y
Install zimbra-proxy [N] n
```

GRÁFICO 117: Selección de paquetes

Fuente: Verónica Mora (2009) Manual Zimbra Collaboration Suite

Si se desea lograr una instalación distribuida, se puede instalar sólo ciertos componentes en el servidor e instalar el resto en uno o varios servidores adicionales.

Por ejemplo, se podría tener el servidor LDAP en un equipo, el servidor STORE en otro y finalmente el servidor MTA en un último equipo. Continuando con nuestro proceso de instalación, el siguiente paso pedirá que se confirme si se desea continuar la instalación, ya que se detecta que se está instalando el paquete en una plataforma diferente.

```
You appear to be installing packages on a platform different
than the platform for which they were built.

This platform is CentO55
Packages found: RHEL5
This may or may not work.

Using packages for a platform in which they were not designed for
may result in an installation that is NOT usable. Your support
options may be limited if you choose to continue.

Install anyway? [N] █
```

GRÁFICO 118: Confirmación de instalación en otra plataforma

Fuente: Verónica Mora (2009) Manual Zimbra Collaboration Suite

A partir de este momento se empiezan a instalar todos los paquetes que previamente fueron seleccionados. En el caso de existir una instalación previa de Zimbra, el instalador se encargará de eliminarla para volver a instalar nuevamente esos componentes.

```
Installing packages

zimbra-core.....zimbra-core-5.0.18_GA_3011.RHEL5-20090707164432.i386.rpm...done
zimbra-ldap.....zimbra-ldap-5.0.18_GA_3011.RHEL5-20090707164432.i386.rpm...done
zimbra-logger.....zimbra-logger-5.0.18_GA_3011.RHEL5-20090707164432.i386.rpm...done
zimbra-mta.....zimbra-mta-5.0.18_GA_3011.RHEL5-20090707164432.i386.rpm...done
zimbra-snmp.....zimbra-snmp-5.0.18_GA_3011.RHEL5-20090707164432.i386.rpm...done
zimbra-store.....zimbra-store-5.0.18_GA_3011.RHEL5-20090707164432.i386.rpm...done
zimbra-apache.....zimbra-apache-5.0.18_GA_3011.RHEL5-20090707164432.i386.rpm...done
zimbra-spell.....zimbra-spell-5.0.18_GA_3011.RHEL5-20090707164432.i386.rpm...done

Setting defaults from saved config in /opt/zimbra/.saveconfig/config.save
HOSTNAME=localhost
LDAPHOST=
LDAPPOR=
SNMPTRAPHOST=
SMTPSOURCE=
SMTPDEST=
SNMPNOTIFY=0
SMTPNOTIFY=0
LDAPROOTPW=zimbra
```

GRÁFICO 119: Instalación de los paquetes de Zimbra

Fuente: Verónica Mora (2009) Manual Zimbra Collaboration Suite

Es posible que durante la instalación aparezca un error debido a que el DNS (interno o externo) no pueda resolver el nombre del servidor o el nombre del dominio, por lo tanto es necesario verificar esta información y el instalador nos permite corregirla en caso de ser necesario. El nombre del servidor debería quedar como servidor.dominio.com y el dominio como dominio.com.

Una vez terminada la instalación, se presenta un resumen con los parámetros de configuración, permitiendo cambiarlos si fuera necesario.

Este es el punto para definir la contraseña del administrador, zona horaria del servidor, o hasta cambiar los puertos que vienen asignados en la configuración por defecto, con el fin de evitar conflictos con otros servidores instalados en el mismo equipo (tomcat, apache, etc.).

```
1) Common Configuration:
2) zimbra-ldap: Enabled
3) zimbra-store: Enabled
   +Create Admin User: yes
   +Admin user to create: admin@magicax.org
***** +Admin Password UNSET
   +Enable automated spam training: yes
   +Spam training user: spam.jfa6fht1lh@magicax.org
   +Non-spam(Ham) training user: ham.gdvcqqz13@magicax.org
   +Global Documents Account: wiki@magicax.org
   +SMTP host: magicax.org
   +Web server HTTP port: 80
   +Web server HTTPS port: 443
   +Web server mode: http
   +IMAP server port: 143
   +IMAP server SSL port: 993
   +POP server port: 110
   +POP server SSL port: 995
   +Use spell check server: yes
   +Spell server URL: http://magicax.org:7780/aspell.php
   +Configure store for use with reverse mail proxy: FALSE
   +Configure store for use with reverse web proxy: FALSE

4) zimbra-mta: Enabled
5) zimbra-snmp: Enabled
6) zimbra-logger: Enabled
7) zimbra-spell: Enabled
8) Default Class of Service Configuration:
r) Start servers after configuration yes
```

GRÁFICO 120: Resumen de la configuración final

Fuente: Verónica Mora (2009) Manual Zimbra Collaboration Suite

Una vez que los servicios de Zimbra se encuentren ejecutándose, es posible ingresar a la Consola de Administración a través de la dirección IP del equipo si el acceso es local o a través del nombre de dominio si se accede desde internet, mediante el puerto asignado a este servicio (7071 por defecto).



GRÁFICO 121: Acceso a la Consola de Administración

Fuente: Verónica Mora (2009) Manual Zimbra Colaboration Suite

ANEXO # 10



MANUAL DE USUARIO DE ZIMBRA

INTRODUCCIÓN



GRÁFICO 122: Logo de Zimbra.

Fuente: <http://www.zimbra.com>

La Suite de Colaboración Zimbra (Zimbra Collaboration Suite o ZCS) es aplicación web de tipo “colaborativa” o Groupware. Existen varias versiones de Zimbra disponibles: una versión soportada por la comunidad de Software Abierto (Open Source), y otras que contienen algunas mejoras pero en las que parte de su código es cerrado, por lo que es soportada comercialmente.

ACCESO A ZIMBRA

Para acceder a la suite de colaboración damos click en la opción Correo del portal universitario.



GRÁFICO 123: Opción para acceder a Zimbra desde el Portal Web

Fuente: Autores

Con esto aparecerá la pantalla de inicio de control de acceso de Zimbra. En esta pantalla ingresamos los siguientes parámetros:

- **Usuario:** Nombre de la cuenta de correo usuario sin el dominio, Ejemplo: Si la cuenta es usuarioejemplo@upse.edu.ec, en este recuadro solo va **usuarioejemplo**.
- **Password:** Introduzca la clave de acceso a su cuenta de correo.
- Escoja el tipo de cliente de web que quiere usar (Avanzado (Ajax), Normal (HTML) o Mobile. (se recomienda usar el avanzado).
- Oprima el botón Log In.

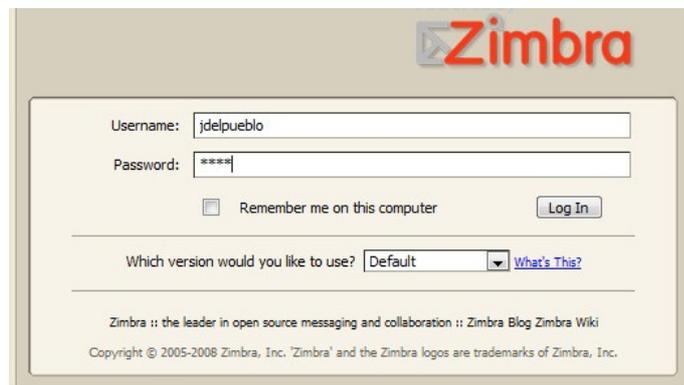


GRÁFICO 124: Pantalla de Acceso a Zimbra

Fuente: Autores

EXPLORAR LAS DIFERENTES APLICACIONES

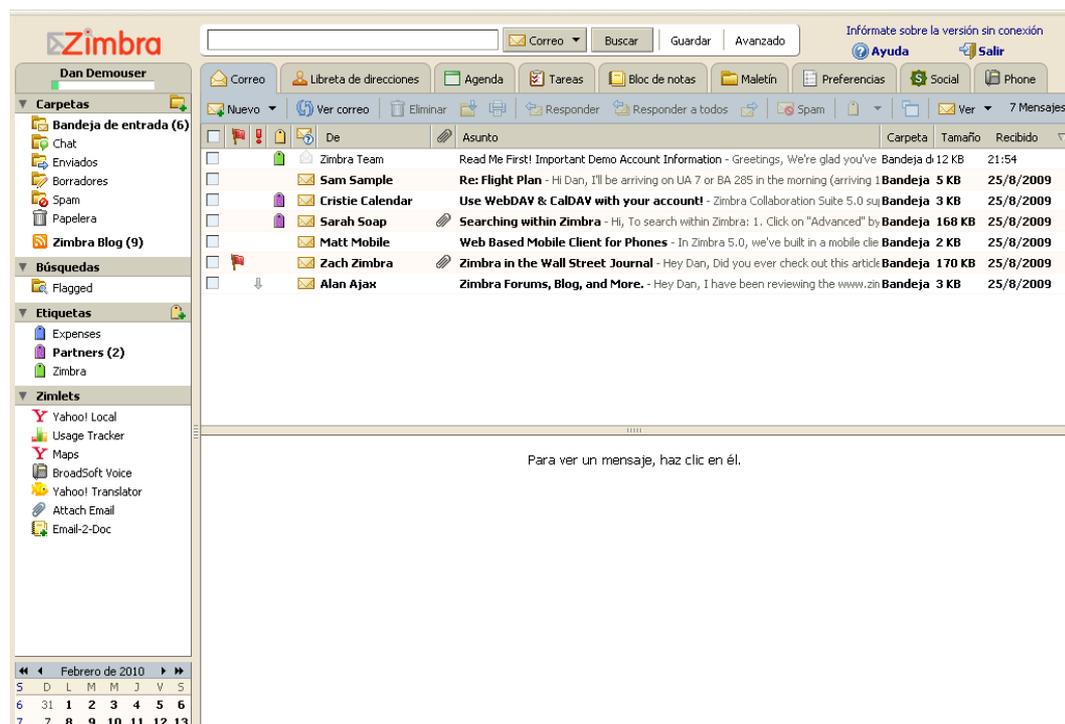


GRÁFICO 125: Pantalla principal de Zimbra

Fuente: Autores

Debajo de los campos para buscar información en Zimbra y buscar información en Internet, aparecen los tabs con las diferentes aplicaciones que componen Zimbra: **Correo, Libreta de Direcciones, Agenda, Tareas, Block de Notas, Maletín, y Preferencias.**

A la izquierda usted verá su nombre y se mostrará cuanto espacio está utilizando de su cuota. Debajo del nombre se muestran las carpetas y subcarpetas donde usted puede almacenar y clasificar los correos. Al final viene un mini calendario, en el que usted puede tocar una fecha para ver la agenda de ese día. Cuando se escoja una aplicación, aparecerá una Barra de Herramientas con opciones específicas para la aplicación seleccionada. Note también que hay opciones de Ayuda y Salir en la esquina superior derecha de la aplicación.

CONFIGURANDO LAS PREFERENCIAS

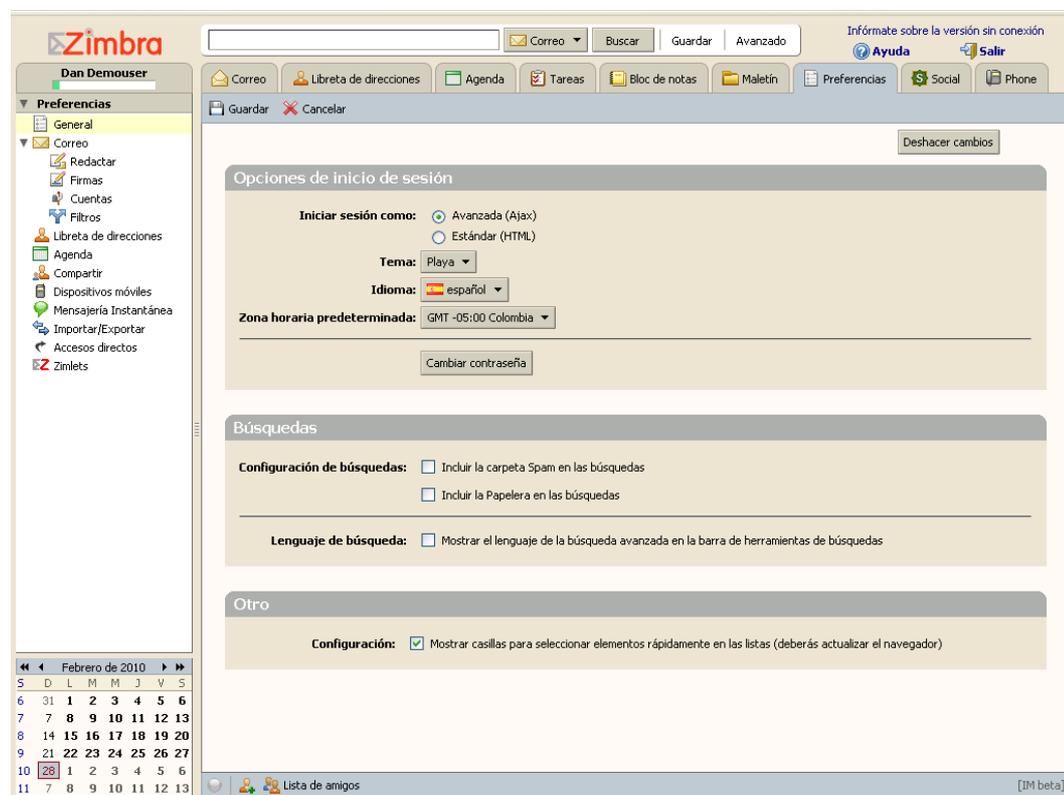


GRÁFICO 126: Configuración de las Preferencias

Fuente: Autores

Usted puede cambiar la forma de como se ve y comporta Zimbra ajustando las opciones mostradas en el tabulador Preferencias. Note las opciones que se despliegan en la parte izquierda de la pantalla: General, Correo, Redactar, Firmas, Cuentas, Libreta de Direcciones, Agenda, Compartir, Dispositivos Móviles, Mensajería Instantánea, Importar Exportar, Accesos Directos, Zimlets. Asegúrese de pulsar el botón Grabar, que se encuentra a la izquierda después de hacer cambios y antes de pasar a otra aplicación. Algunos detalles importantes de las opciones de configuración:

CORREO

Usted puede configurar cada cuanto tiempo se actualizará el buzón de entrada con correo nuevo. Usted puede cargar el correo nuevo al buzón de entrada al tocar el botón **Ver Correo** cuando esté en la aplicación de Correo.

REDACTAR

Si usted quiere redactar mensajes con diferentes tipos de letra, colores e imágenes, escoja redactar los mensajes en HTML.

FIRMAS

En esta opción usted puede crear múltiples firmas para los correos electrónicos, cada una con un nombre único.

CUENTAS

Defina aquí la firma por defecto a utilizar en todos los correos. Cuando usted esté creando un nuevo correo, podrá escoger otra firma o no firmar el correo.

AGENDA

Al dejar marcada la opción “Siempre mostrar la agenda en miniatura”, quedará activada la “Miniagenda” a la izquierda y le permitirá ver rápidamente citas y reuniones al sobreponer el cursor en una fecha específica.

Tómese su tiempo para explorar las diferentes opciones mostradas en Preferences.

USANDO LAS BÚSQUEDAS RÁPIDAS

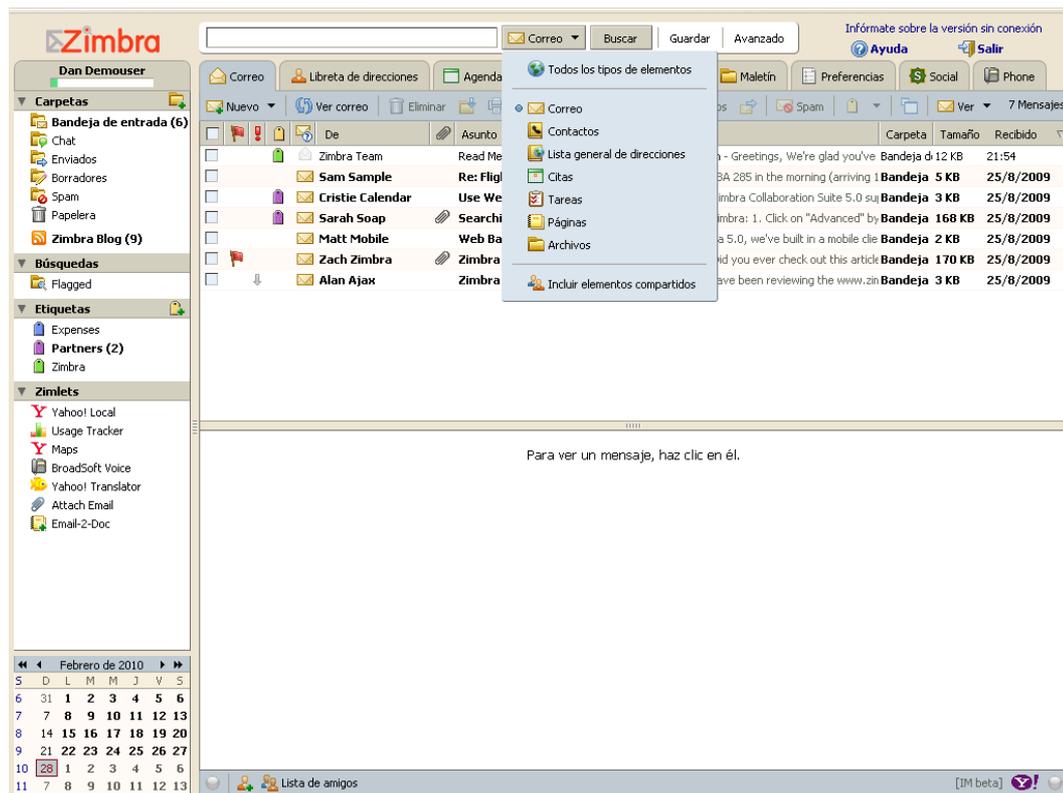


GRÁFICO 127: Uso de la opción de búsqueda.

Fuente: Autores.

La función para buscar información almacenada en Zimbra, esta a la derecha del logo de Zimbra.

Asegúrese que ha seleccionado el tipo de ítem que quiere buscar. Si no selecciona el tipo de ítem, la búsqueda se hará de acuerdo a la aplicación que tenga activa en el momento de la búsqueda.

Luego, escriba el nombre o la palabra que quiere buscar, presione el botón “Buscar”.

USANDO EL CORREO ELECTRÓNICO

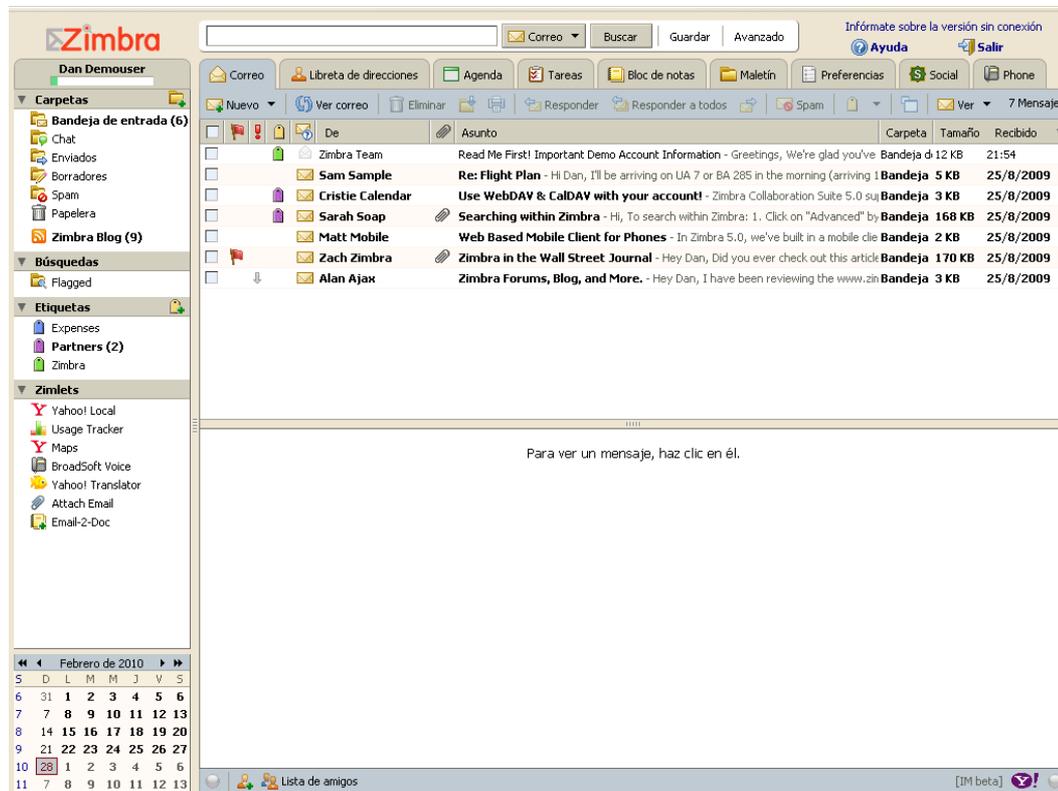


GRÁFICO 128: Ventana de Gestión de Correos

Fuente: Autores

Ordene sus correos tocando cualquiera de los campos de la ventana para ver el correo. Por ejemplo, para ordenar los correos por fecha toque el campo marcado como **Recibido**.

LEER CORREOS

El número de mensajes desplegados en su buzón de entrada es controlado en la opción de **Preferencias**.

Si tiene más mensajes que los mostrados utilice las flechas en la parte derecha de la barra de herramientas para navegar al siguiente grupo de mensajes.

Usted puede leer los mensajes en la ventana de lectura de mensajes, si usted activa esa ventana en el menú de **Ver** o dentro de la opción de **Preferencias**.

Haciendo doble click en un mensaje, podrá ver el mensaje en una ventana más grande o en una ventana separada, de acuerdo a las opciones seleccionadas en la opción de **Preferencias**.

Las opciones de **Responder**, **Responder a Todos** y **Reencviar** en la barra de herramientas realizan las respectivas funciones.

La opción de “**Spam**”, permite clasificar el mensaje como **Spam**.

IMPRIMIR CORREOS

Haga click en el icono de la impresora en la barra de herramientas o toque el botón derecho sobre el mensaje que quiere imprimir y escoja la opción **Imprimir**. El mensaje se abrirá en una gran ventana junto a una caja de diálogo de impresión.

Después de imprimir el mensaje presione la “**X**”, en la esquina superior derecha de la ventana para cerrarla.

CREAR UN NUEVO MENSAJE

Para crear un nuevo mensaje, siga este procedimiento:

- Presione el botón **Nuevo** en la barra de herramientas. Se mostrará la página para crear un nuevo correo electrónico.
- Complete los campos de dirección (Para:, CC:), Asunto y texto del mensaje conforme a sus necesidades.
- Si quiere habilitar el campo para copia oculta (CCO:), toque el botón **Mostrar CCO**.

- Para revisar la ortografía, presione el botón **Comprobar Ortografía**.
- Para adjuntar un archivo, presione el botón Añadir archivo adjunto.
- Presione el botón **Enviar** para mandar el mensaje.

Si por alguna razón tiene que interrumpir la creación de un mensaje, presione la opción **Guardar Borrador**. El mensaje se guardará en la carpeta Borradores. Para continuar trabajando en este mensaje, abra la carpeta y de doble click en el mensaje que quiere editar o enviar.

Si quiere agregar su firma de forma manual, presione la opción **Firma**. Esta opción solo se activa una vez que usted haya definido una firma en la opción de Preferencias. Usted puede configurar que la firma se agregue automáticamente a los mensajes usando la opción Preferencias - Cuentas. Escoja la firma que quiere asociar a la cuenta. Si quiere agregar las firmas manualmente escoja la opción **No añadir firma**.

Se puede cambiar la firma predeterminada para la cuenta, presionando la opción **Firma** a la hora de crear un nuevo mensaje

ADJUNTAR ARCHIVOS A UN MENSAJE

Los mensajes de correo electrónico pueden contener archivos adjuntos. Usted puede adjuntar documentos, hojas de trabajo, fotos y otros tipos de documentos.

Para adjuntar un archivo a un mensaje presione el botón **Añadir archivo adjunto**., luego presione el botón “Examinar” en la opción para el primer archivo, seleccione el archivo que usted quiere adjuntar y presione el botón Grabar. Repita los pasos para agregar más archivos. Presione el botón **Adjuntar** para incluir los archivos en el mensaje.

FILTROS DE CORREO

Los filtros para clasificar correos se pueden crear, modificar o eliminar en la opción Preferencias – Filtros.

Para crear un filtro, escoja la opción **Nuevo Filtro**. Se abrirá una nueva ventana. Escoja un nombre para el filtro, escoja si el filtro se basará en una o todas las condiciones que usted defina para aplicar este filtro. Defina las diferentes condiciones a las que se le aplica el filtro. Para agregar una nueva condición presione el botón “+”. Para eliminar una condición presione el botón “-”. Presione el botón **Aceptar** para guardar los cambios.

ORGANIZARSE USANDO CARPETAS

Para crear, renombrar, mover o borrar carpetas, presione el botón derecho del ratón sobre la carpeta que quiere modificar en la ventana que muestra las carpetas. Para crear una carpeta debajo de la carpeta principal, presione el botón derecho sobre el título Carpetas. Note que las carpetas pueden tener subcarpetas.

USANDO LA LIBRETA DE DIRECCIONES

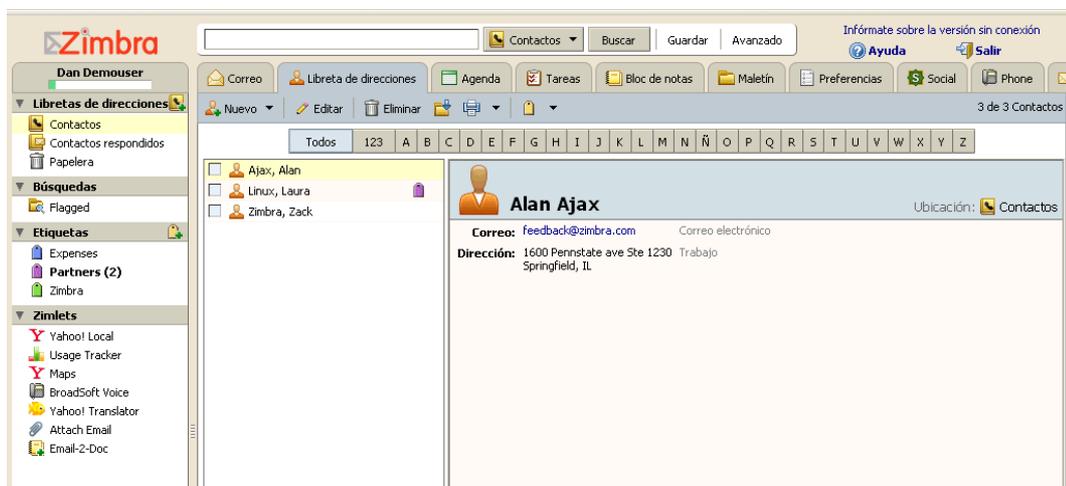


GRÁFICO 129: Opción de Libreta de Direcciones

Fuente: Autores

Zimbra le permite agregar contactos personales en la opción de Libreta de Direcciones. Para agregar un nuevo contacto personal haga lo siguiente:

- Presione la flechita hacia abajo que se encuentra en el botón Nuevo.
- Seleccione la opción **Contacto**.
- Incluya toda la información relevante de su contacto personal.
- Escoja como quiere almacenar el nuevo contacto en la opción Grabar Como. La opción normal es almacenarlo por Apellido, Nombre.
- Escoja uno de sus libros de direcciones personales para almacenar el contacto presionando la opción **Ubicación**.
- Presione el botón de **Guardar**.

USANDO EL CALENDARIO

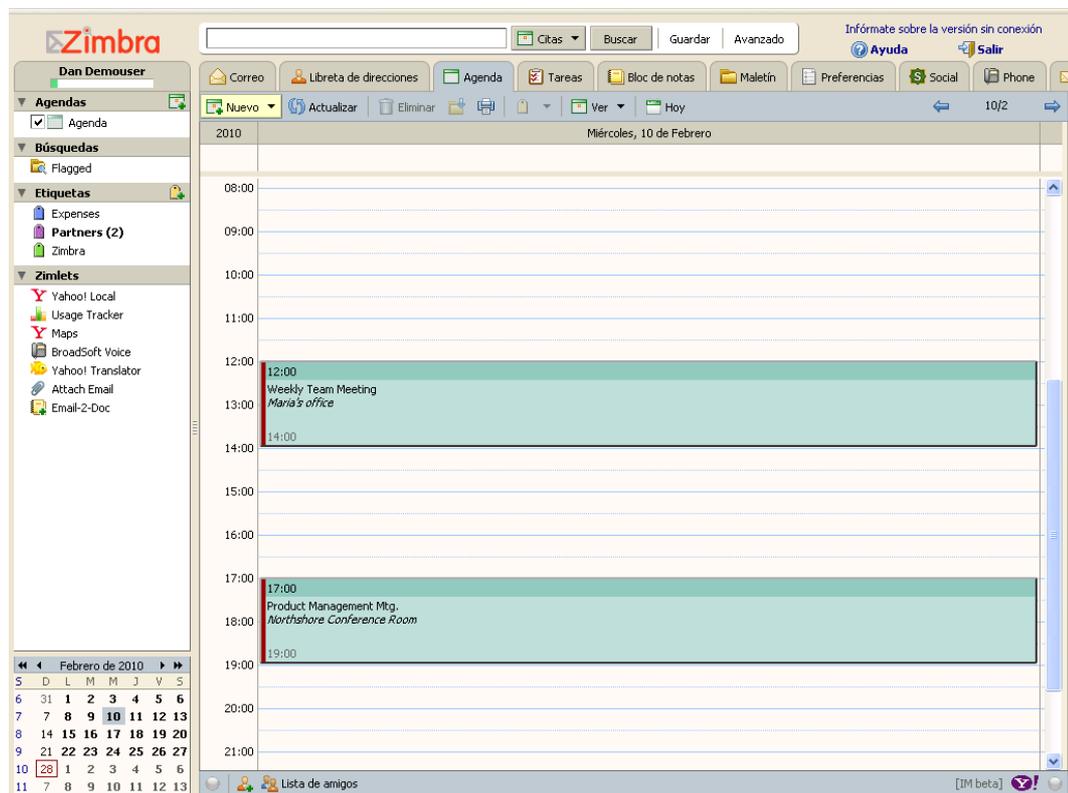


GRÁFICO 130: Opción de Calendario.

Fuente: Autores.

Para añadir una cita siga los siguientes pasos:

- En la barra de herramientas presione la flecha hacia abajo que se encuentra a la derecha del botón **Nuevo**. Seleccione la opción **Cita**.
- Llene el campo **Asunto**. Este campo es obligatorio. La información ingresada en este campo, será la información que se desplegará luego en el calendario.
- Llene el campo de Ubicación.
- Configure la fecha y la hora de la cita. Ingrese la fecha y hora de inicio (Empezar) o presione la flecha para desplegar un calendario para escoger una fecha. Seleccione la fecha y hora en que finaliza (Final) el evento.
- Use el área en blanco para incluir información adicional. Para adjuntar archivos presione el botón **Añadir archivo adjunto** en el menú superior.
- Presione el botón **Grabar** para almacenar la cita.

USANDO LA OPCIÓN DE TAREAS

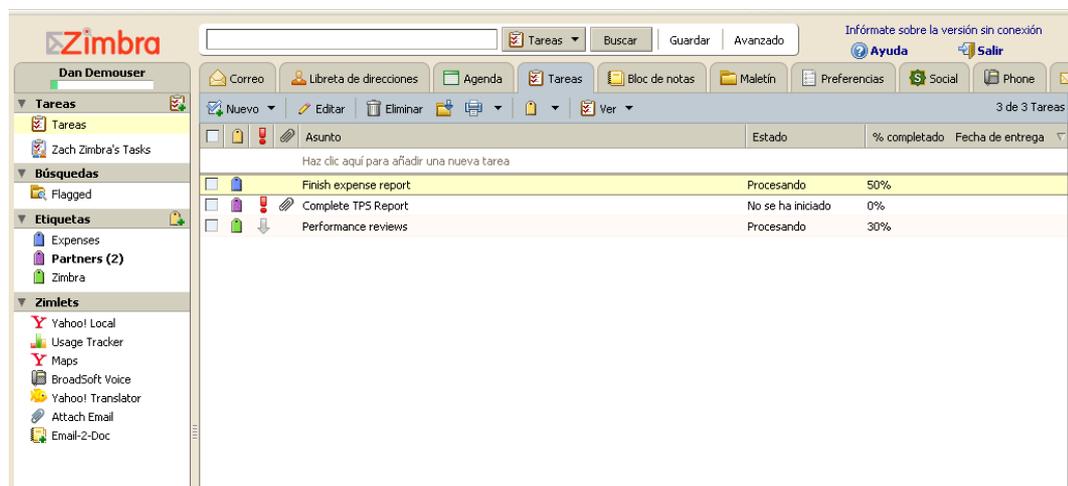


GRÁFICO 131: Opción de Tareas.

Fuente: Autores.

Para añadir una nueva tarea siga el siguiente procedimiento:

- En la barra de herramientas presione la flecha hacia abajo que se encuentra a la derecha del botón **Nuevo**. Seleccione la opción **Tarea**.
- Aparecerá una pantalla donde usted puede anotar los datos de la tarea:
 - Asunto (Es el único campo obligatorio).
 - Ubicación (Lugar donde se deberá de realizar la tarea)

- Prioridad
- Hay varios campos para llevar el control de avance de la tarea.
 - Fecha de inicio
 - Fecha de entrega
 - Espacio en blanco para describir la tarea.
- Presione el botón **Grabar**.

ANEXO # 11



MANUAL DE INSTALACIÓN DE ESPABIBLIO

INTRODUCCIÓN

Una de las herramientas seleccionadas para este trabajo de tesis fue OpenBiblio, para la categoría del Sistema Integrado de Administración de Biblioteca, por lo que hemos procedido a instalar la versión en español de OpenBiblio denominada



GRÁFICO 132: Logo de Espabiblio

Fuente: <http://www.desem.cl/espabiblio/>

EspaBiblio.

Para Instalar Espabiblio solo necesitamos tener instalado un servidor Apache con soporte para PHP y el motor de Bases de datos MySQL server, teniendo todo esto instalado correctamente solo hay que pegar la carpeta **openbiblio** dentro de la carpeta de documentos de apache, crear una base de datos llamada openbiblio con un usuario y contraseña el que debe ser definido en el archivo **database_constants.php** en la raíz de la carpeta **openbiblio**.

Se dará cuenta que el fichero **database_constants.php** sugiere un usuario ebiblio y Contraseña epassword.

Luego debemos ingresar al navegador con la siguiente dirección <http://localhost/openbiblio/install/> y comenzara la instalación. Una vez concluida la instalación elimine la carpeta install por seguridad.

CREACIÓN DE LA BASE DE DATOS OPENBIBLIO

Para crear la base de datos seguimos el siguiente procedimiento:

- Acceder a MySQL

```
# mysql -u root
```

- Crear la Base de Datos

```
mysql> create database openbiblio;
```

- Crear el Usuario openbiblio con el password openbiblio

```
mysql> create user openbiblio@localhost identified by  
'openbiblio';
```

- Asignar todos los privilegios al usuario

```
mysql> grant all privileges on openbiblio.* to  
openbiblio@localhost;
```

- Salimos de la herramienta de administración de MySQL.

```
mysql> quit
```

MODIFICACIÓN DEL ARCHIVO DATABASE_CONSTANTS.PHP

Modificamos el archivo denominado **database_constants.php** que está en el directorio de openbiblio, de manera que tenga el siguiente contenido:

```
define("OBIB_HOST",      "localhost"); //host ej. localhost  
define("OBIB_DATABASE", "openbiblio"); // base de datos  
define("OBIB_USERNAME", "openbiblio"); // usuario  
define("OBIB_PWD",      "openbiblio"); //contraseña
```

INSTALANDO ESPABIBLIO

- Descomprimir el archivo espabiblio descargado desde la Web de espabiblio de manera que solo quede una carpeta llamada espabiblio y dentro de ella los ficheros de la aplicación, luego renombramos la carpeta a **openbiblio** y copiaremos toda esta carpeta en el directorio **/var/www/html** del servidor.
- Acceder al navegador de internet (recomendamos mozilla Firefox) , una vez iniciado el navegador escriba en la barra de direcciones la siguiente ruta **http://www.upse.edu.ec/openbiblio/install/** y aparecerá la siguiente pantalla.



GRÁFICO 133: Pantalla de Instalación de EspaBiblio

Fuente: <http://www.desem.cl/espabiblio/>

Si nos aparece esta pantalla con el mensaje “puede comenzar... pues se ha realizado correctamente la conexión a la base de datos”.

Presione sobre el botón instalar. Si aparecen errores revise que los antecedentes ingresados en el fichero database_constans.php sean los correctos (nombre de base de datos, usuario y contraseña).

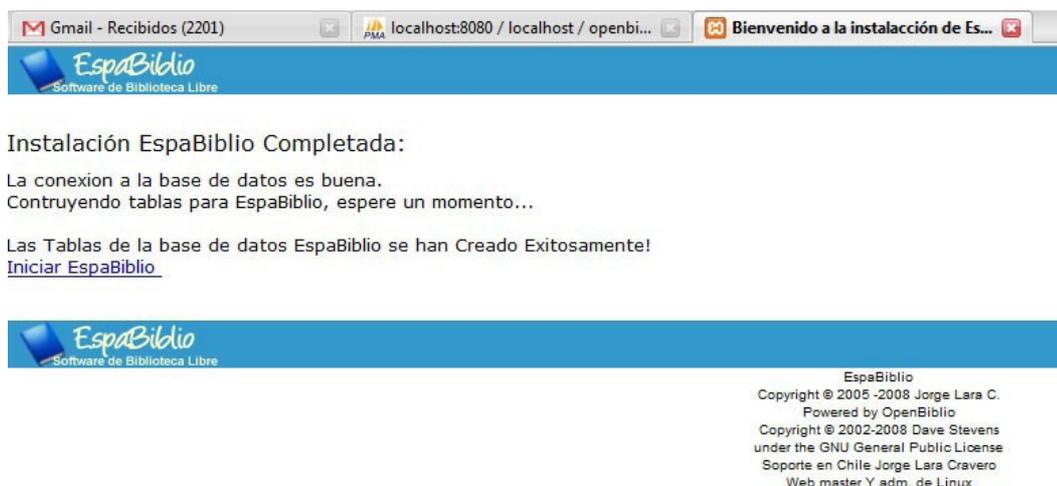


GRÁFICO 134: Confirmación de Instalación Correcta.

Fuente: <http://www.desem.cl/espabiblio/>

Si no se producen errores en la instalación aparecerá el mensaje “Instalación EspaBiblio Completada” “Las Tablas de la base de datos EspaBiblio de han creado Exitosamente”, realizamos clic sobre iniciar EspaBiblio para ver nuestra instalación.

Presionamos entrar o cualquier pestaña de las superiores y nos aparecerá el formulario de identificación. La clave es **admin** y el usuario también es **admin**.

Nota: Eliminar el directorio Install del directorio espabiblio y además cambie la clave por defecto para ingresar al sistema, para conservar la seguridad en el sistema.

Por último al final de la página aparece un enlace que dice OPAC (Catálogo de acceso al Público), este es el módulo que deben tener acceso los usuarios de la biblioteca para ver los materiales o colecciones de ella. (Esta dirección debe entregársela a sus usuarios).

EspaBiblio Chile
Software de Bibliotecas Chile

EspaBiblio Biblioteca Ver. 2.0

Inicio Préstamo Catalogación Administración Informes

Entrar

» Inicio
Licencia-Esp
Ayuda

Bienvenido a EspaBiblio
Utiliza las pestañas superiores para acceder a las diferentes páginas de administración de la biblioteca.

Página (pestaña)	Descripción
Circulación (prestamos) 	Utiliza esta página para administrar los datos de los socios. <ul style="list-style-type: none"> • Administración de socios (añadir nuevos, buscar, editar, borrar) • Préstamos, reservas, cuentas e historial de los socios de la biblioteca • Registros de bibliografía y del carrito de reposición en estantería
Catalogación 	Administración de datos bibliográficos. <ul style="list-style-type: none"> • Administración bibliográfica (nuevo, buscar, editar, borrar)
Administración 	Administración de bibliotecarios y de datos administrativos. <ul style="list-style-type: none"> • Administración de bibliotecarios (añadir nuevos, editar, cambiar contraseñas) • Configuración general de la biblioteca • Lista de colecciones • Lista de tipos de material • Editor de temas de diseño

GRÁFICO 135: Pantalla Principal de EspaBiblio

Fuente: <http://www.desem.cl/espabiblio/>

ANEXO # 12



MANUAL DE USUARIO DE ESPABIBLIO

INTRODUCCIÓN

Espabiblio es un sistema para gestión de Bibliotecas desarrollado en PHP y MySQL, se basa en OpenBiblio, desarrollado por Dave Stevens de EE.UU.

El nombre EspaBiblio quiere decir "Biblioteca En Español" y el desarrollo o la adaptación desde el proyecto original la lleva a cabo Jorge Lara Cravero, Administrador de Linux y Web Master del sitio oficial <http://www.desem.cl/espabiblio/>.

EspaBiblio incluye las funcionalidades de Circulación, Catalogación, Administración de staff, permisos de funcionarios, OPAC, Informes, Estadísticas, Impresión de Cartas, etiquetas, soporta la Importación en Formato estándar Bibliotecológico MARC o USMARC y clasificación de estatus, como disponible, prestado, en reparación agrega soporte para reservas, etc..

Entre las nuevas funcionalidades de EspaBiblio en su versión 2.0 se agrega la funcionalidad de crear las etiquetas con códigos de barras automáticamente.

Una vez catalogados los items solo es necesario ir a la pestaña reportes y generar las etiquetas (incluidos los codigos de barra). Otra funcionalidad importante es la habilitación de recuperar la información de la Library of Congress (Biblioteca del Congreso) utilizando SRU (Search Retrieval URL), que devuelve un formulario XML y el resultado se exporta automáticamente a EspaBiblio. Como sus Bases de Datos están realizadas en MySQL soporta también exportación a formatos de Microsoft Excel, Word, SQL, LaTeX, CSV y XML.

La administración de la Biblioteca ofrece una interfaz intuitiva con un diseño de pestañas y barra lateral. EspaBiblio puede ser instalado en Windows 98se, 2000, XP, 2003, Linux, Macintosh OS X, UNIX, en general cualquier sistema Operativo con soporte de Apache, PHP y MySQL.

ACCESO

Para acceder a la biblioteca, ingrese a la siguiente dirección en el navegador de Internet: <http://www.upse.edu.ec/openbiblio/home/index.php>. Para poder acceder a los servicios que otorga la biblioteca deberá ingresar con su cuenta de usuario haciendo clic en el botón Entrar ubicado en la parte izquierda de la ventana.

Una vez adentro estarán disponibles en forma de pestañas las opciones siguientes:

- Circulación (prestamos).
- Catalogación.
- Administración.
- Informes.

The screenshot shows the 'Biblioteca UPSE' web interface. At the top right, it displays the date 'Fecha: 22.02.2010', the hours 'Horario: Lunes a Viernes de 08H00 a 21H00', and the phone number 'Teléfono: (04) 2780-018'. Below the header, there is a navigation menu with 'Inicio', 'Préstamo', 'Catalogación', 'Administración', and 'Informes'. The main content area is titled 'Bienvenido a EspaBiblio' and includes a table of options:

Página (pestaña)	Descripción
Circulación (prestamos)	Utiliza esta página para administrar los datos de los socios. <ul style="list-style-type: none"> Administración de socios (añadir nuevos, buscar, editar, borrar) Préstamos, reservas, cuentas e historial de los socios de la biblioteca Registros de bibliografía y del carrito de reposición en estantería
Catalogación	Administración de datos bibliográficos. <ul style="list-style-type: none"> Administración bibliográfica (nuevo, buscar, editar, borrar) Permite al usuario recuperar la información de la Biblioteca del Congreso utilizando SRU (Search Retrieval URL), que devuelve un formulario XML.
Administración	Administración de bibliotecarios y de datos administrativos. <ul style="list-style-type: none"> Administración de bibliotecarios (añadir nuevos, editar, cambiar contraseñas, borrar) Configuración general de la biblioteca Lista de colecciones Lista de tipos de material Editor de temas de diseño
Informes	En esta página puedes ejecutar informes a partir de los datos de la biblioteca. <ul style="list-style-type: none"> Informes. Etiquetas.

At the bottom of the page, there is a link for 'Catálogo Online Público (OPAC)'.

GRÁFICO 136: Opciones de EspaBiblio.

Fuente: Autores.

PRÉSTAMO

BUSCAR

Esta opción permite obtener información sobre los usuarios registrados en la biblioteca mediante dos tipos de búsqueda:

- **Buscar socio por número de tarjeta:** Busca información del usuario cuyo número de tarjeta se ha ingresado como clave de búsqueda.
- **Buscar por apellido:** Realiza una búsqueda tomando como clave el apellido del usuario.

NUEVO SOCIO

(Opción habilitada exclusivamente para administradores)

Esta opción añade un nuevo usuario dentro de la base de datos de la biblioteca, para lo cual se deberán ingresar los siguientes datos:

- **Clasificación:** Se debe elegir el cargo del nuevo usuario dentro de la universidad, se puede elegir entre “Estudiante”, “Personal Administrativo”, “Profesores”.
- **Número de carnet:** Aquí se debe escribir el número de carnet del usuario.
- **Apellido.**
- **Nombre.**
- **Email.**
- **Dirección postal.**
- **Teléfono.**
- **Teléfono del lugar de trabajo del usuario.**
- **Grado escolar.**
- **Grado tutor.**

Luego de haber llenado todos los campos con los respectivos datos del usuario se lo podrá añadir oprimiendo el botón “Enviar”.

Biblioteca UPSE

Fecha: 22.02.2010
Horario: Lunes a Viernes de 08H00 a 21H00
Teléfono: (04) 2780-018

Inicio Préstamo Catalogación Administración Informes

Logout

Buscar socio
» Nuevo socio
» Devolución
» Ayuda

Añadir nuevo Socio:

Clasificación: Estudiantes
Número de carnet: 0123456789
Apellido: Pérez
Nombre: Juan
Email: juan@perez.com
Dirección postal:
Teléfono: 0123456
Teléfono trabajo: 7890123
Grado escolar:
Grado tutor:

Enviar Cancelar

Catálogo Online Público (OPAC)

EspaBiblio

EspaBiblio versión 2.0 basado en OpenBiblio 0.6.0 Versión de base de datos 0.6.0
Copyright © 2005-2008 Jorge Lara Cravero
 bajo GNU General Public License

GRÁFICO 137: Opción Nuevo Socio.

Fuente: Autores.

DEVOLUCIÓN

Esta opción controla el sistema de devolucion de libros prestados por la biblioteca. Se debe ingresar el código de barras del elemento para añadirlo al carrito de reposición en las estanterías de la biblioteca.

Biblioteca UPSE

Fecha: 22.02.2010
Horario: Lunes a Viernes de 08H00 a 21H00
Teléfono: (04) 2780-018

Inicio Préstamo Catalogación Administración Informes

Logout

Buscar socio
» Nuevo socio
» Devolución
» Ayuda

Devolución:

Código de barras: 0123456789 [Buscar](#) [Añadir al carrito de reposición en las estanterías](#)

[Devolver el material seleccionado](#) | [Devolver todo](#)

Lista actual del carrito de reposición en las estanterías:

Fecha de escaneado	Código de barras	Título	Autor
En la actualidad no hay items en meson para reponerlos en las estanterías.			

[Devolver el material seleccionado](#) | [Devolver todo](#)

Catálogo Online Público (OPAC)

EspaBiblio

EspaBiblio versión 2.0 basado en OpenBiblio 0.6.0 Versión de base de datos 0.6.0
Copyright © 2005-2008 Jorge Lara Cravero
 bajo GNU General Public License

GRÁFICO 138: Opción Devolución.

Fuente: Autores

CATALOGACIÓN

Contiene opciones de control sobre los elementos del catálogo de la biblioteca, estas son:

- **Buscar material:** Realiza búsquedas de bibliografías mediante código de barras, título, autor o fragmentos del resumen.

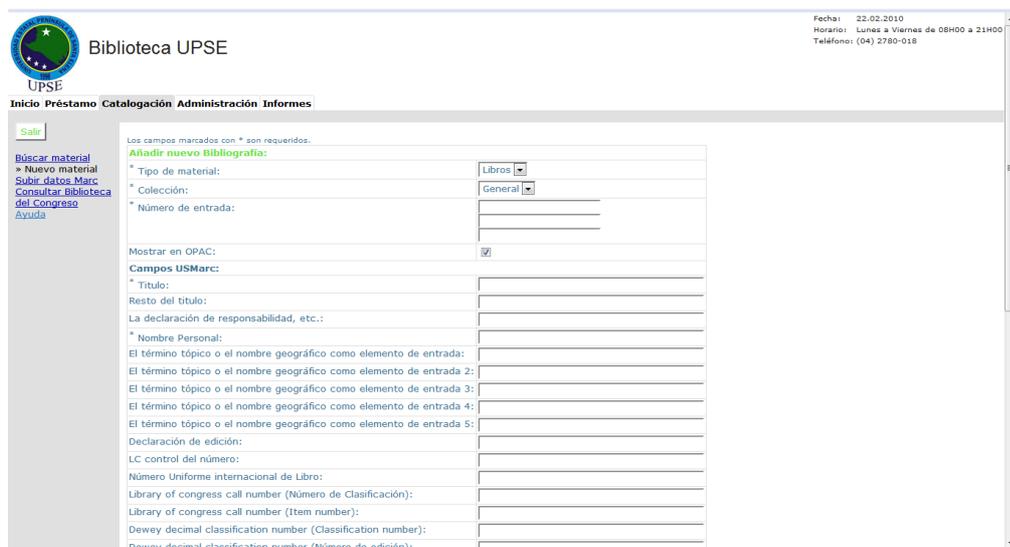


The screenshot shows the 'Biblioteca UPSE' web interface. At the top, there is a navigation menu with 'Inicio', 'Préstamo', 'Catalogación', 'Administración', and 'Informes'. The 'Catalogación' section is active, displaying two search options: 'Buscar bibliografía por código de barras' and 'Buscar bibliografía por frase de búsqueda'. The 'Buscar' button is visible for both. The interface also includes a sidebar with links like 'Salir', 'Buscar', 'Nuevo material', 'Subir datos Marc', 'Consultar Biblioteca del Congreso', and 'Ayuda'. The footer contains the 'EspaBiblio' logo and version information.

GRÁFICO 139: Opción para buscar material.

Fuente: Autores.

- **Nuevo material:** Aquí se podrá añadir un nuevo elemento al catálogo en línea llenando los respectivos campos.



The screenshot shows the 'Biblioteca UPSE' web interface with the 'Añadir nuevo Bibliografía' form. The form is titled 'Los campos marcados con * son requeridos.' and includes fields for 'Tipo de material' (Libros), 'Colección' (General), 'Número de entrada', 'Mostrar en OPAC' (checked), and 'Campos USMarc'. The 'Campos USMarc' section includes fields for 'Título', 'Resto del título', 'La declaración de responsabilidad, etc.', 'Nombre Personal', and five 'El término tópico o el nombre geográfico como elemento de entrada' fields. The form also includes fields for 'Declaración de edición', 'LC control del número', 'Número Uniforme internacional de Libro', 'Library of congress call number (Número de Clasificación)', 'Library of congress call number (Item number)', 'Dewey decimal classification number (Classification number)', and 'Número decimal classification number (Número de edición)'. The interface also includes a sidebar with links like 'Salir', 'Buscar material', 'Nuevo material', 'Subir datos Marc', 'Consultar Biblioteca del Congreso', and 'Ayuda'. The footer contains the 'EspaBiblio' logo and version information.

GRÁFICO 140: Ingreso de un nuevo material.

Fuente: Autores.

- **Subir datos Marc:** Mediante esta opción se tiene la posibilidad de subir archivos Marc (MACHine-Readable Cataloging) que es un formato de datos bibliográficos.

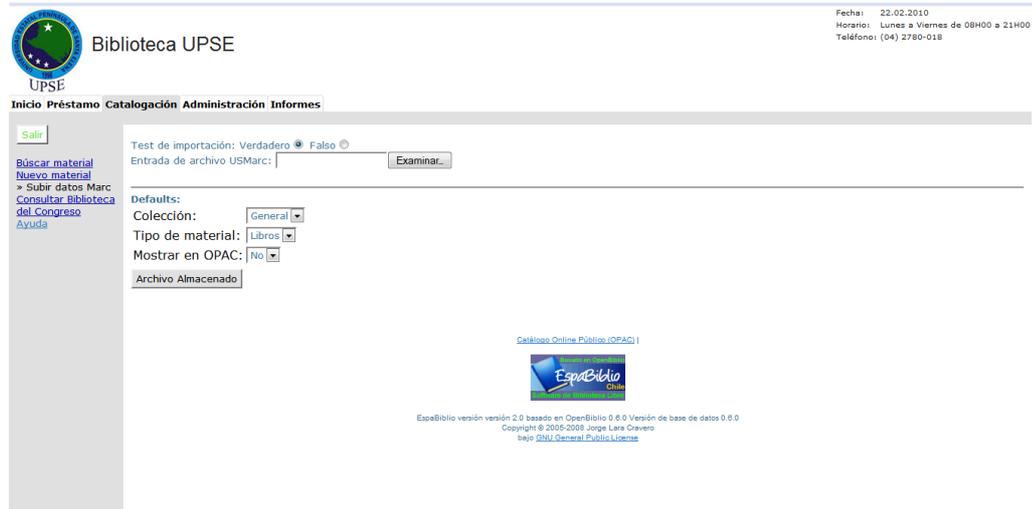


GRÁFICO 141: Subir información en formato MARC.

Fuente: Autores.

- **Consultar en Biblioteca del Congreso:** Permite al usuario recuperar la información de la Biblioteca del Congreso utilizando SRU (Search Retrieval URL), que devuelve un formulario XML. Se pueden realizar las consultas ingresando como clave de búsqueda el título, autor o ISBN (International Standard Book Number) de la bibliografía.

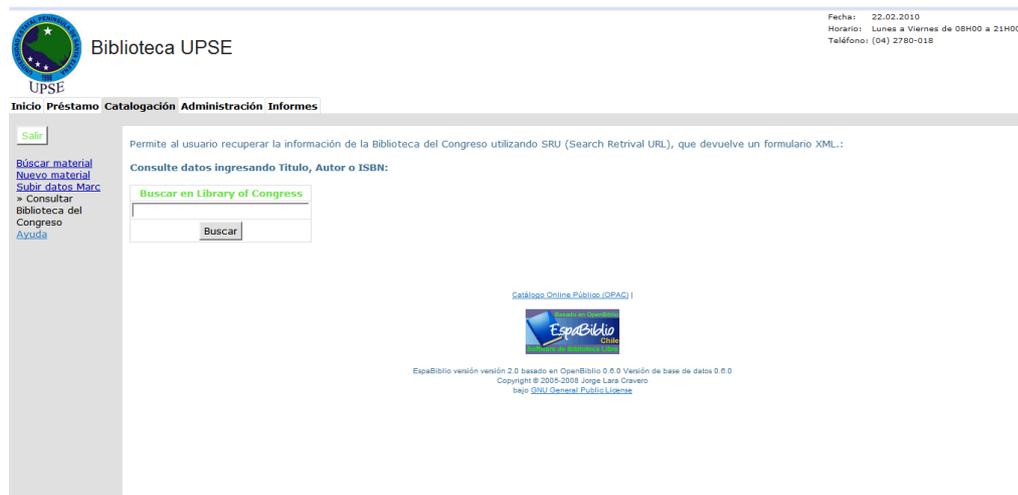


GRÁFICO 142: Consulta en la Biblioteca del Congreso.

Fuente: Autores.

ADMINISTRACIÓN

Contiene opciones para administrar los bibliotecarios y los datos administrativos.

- **Administrar bibliotecarios:** Permite añadir, modificar parámetros o eliminar bibliotecarios.

Función	Apellido:	Nombre:	Nombre de usuario:	Atributos:	Suspendido:
editar pwd borrar	Root	Administrator	admin	Circ: actualizar miembro Catalogar Admin Informes	No
editar pwd borrar	Torres	Luis	Itorres	Sí Sí Sí Sí	No

GRÁFICO 143: Administración de Bibliotecarios.

Fuente: Autores.

- **Configurar biblioteca:** Aquí se pueden editar las características de la biblioteca.

Editar los entornos de la biblioteca:

nombre de la biblioteca: Biblioteca UPSE

imagen URL de la biblioteca: ./images/logo_upse_small.jpg

Sólo mostrar la imagen en el encabezamiento:

horarios de la biblioteca: Lunes a Viernes de 08H00 a 21H00

teléfono de la biblioteca: (04) 2780-018

URL de la biblioteca: /openbiblio

OPAC URL: ./opac/index.php

Duración de la sesión: 20 minutos

resultados de la búsqueda: 10 artículos por página

Limpiar el historial de la bibliografía: 6 meses

Bloquear los préstamos cuando haya pendiente una multa:

Max. hold length: 14 days

Idioma: Castellano

HTML Charset: iso-8859-1

Atributos de etiquetas de lenguaje HTML:

Actualizar

GRÁFICO 144: Configuración de la Biblioteca.

Fuente: Autores.

- **Tipos de material:** Presenta opciones de edición para tipos de elementos dentro de la biblioteca.

La función de borrado sólo está disponible en colecciones que tienen una cuenta bibliográfica de cero. Si se desea borrar una colección con una cuenta bibliográfica mayor de cero primero se deberá cambiar el tipo de material en esas bibliografías a otro tipo de material.



GRÁFICO 147: Mantenimiento de los Tipos de Material.

Fuente: Autores.

- **Colecciones:** Posibilita el manejo de colecciones dentro de la biblioteca permitiendo modificar parámetros de cada una de ellas así como eliminarlas por completo o añadir nuevas colecciones.

La función de borrado sólo está disponible en colecciones que tengan una cuenta bibliográfica de cero. Si se desea borrar una colección con una cuenta bibliográfica mayor de cero primero se tendrá que cambiar el tipo de material de esas bibliografías a otro tipo de material.



GRÁFICO 148: Mantenimiento de Colecciones.

Fuente: Autores.

- **Materiales y privilegios:** Otorga la posibilidad de editar las preferencias de cada clase de miembro sobre los distintos tipos de materiales existentes en la biblioteca.



GRÁFICO 149: Mantenimiento de Materiales y Privilegios.

Fuente: Autores.

- **Temas de diseño:** Contiene opciones de modificación del diseño de la biblioteca virtual con temas, los cuales pueden ser añadidos modificados o eliminados(a excepción del que se encuentra en uso).

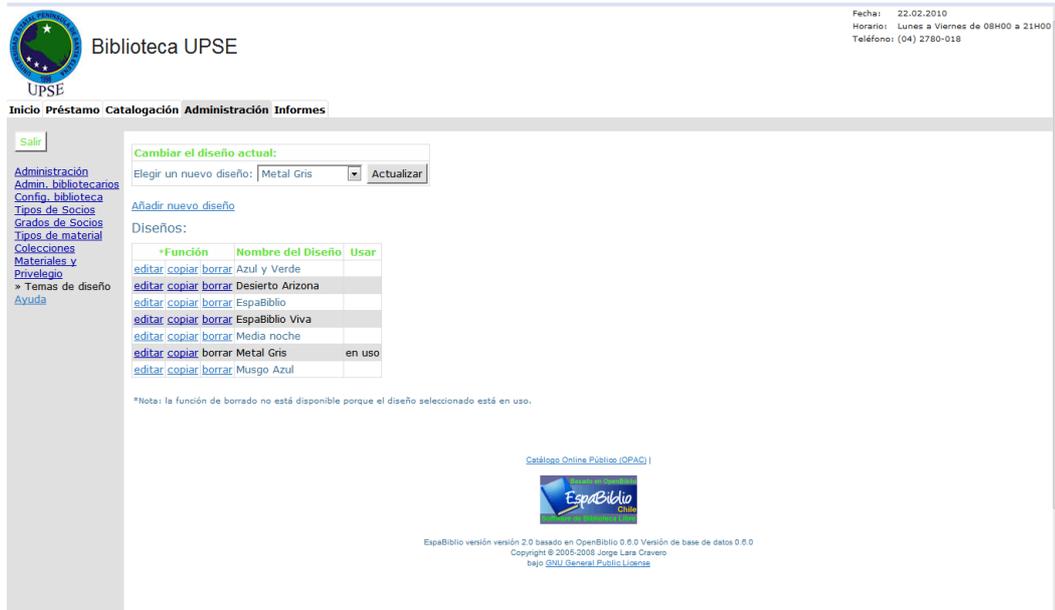


GRÁFICO 150: Temas de Diseño.

Fuente: Autores.

INFORMES



GRÁFICO 151: Opción de Informes.

Fuente: Autores.

Esta pestaña nos otorga la capacidad de realizar distintos tipos de informes útiles para apreciar el funcionamiento de la biblioteca.

Se presentan a disposición del usuario dos clases de informes:

– Informes de Catalogación:

- **Búsqueda de copias:** Redacta un informe en el que se detalla todo lo referente a los elementos que cumplen determinadas características especificadas por el usuario, dando también opciones de presentación como orden y formato de salida.}



GRÁFICO 152: Opción de Búsqueda de Copias.

Fuente: Autores.

– Informes de préstamos:

- **Bibliografías más populares:** Elabora un informe que contiene las bibliografías más populares en el orden y formato que desee el usuario.



GRÁFICO 153: Lista de bibliografía más popular

Fuente: Autores.

- **Lista de artículos pendientes de devolución por socios:** Permite obtener un listado de los elementos que pendientes de entregarse a la biblioteca.



GRÁFICO 154: Lista de artículos pendientes de devolución.

Fuente: Autores.

- **Lista de socios con artículos pendientes de devolución:** Esta opción redacta una lista con información de los usuarios que deben devolver artículos a la biblioteca en el futuro.



GRÁFICO 155: Lista de Socios con artículos pendientes de devolución

Fuente: Autores.

- **Listado de bibliografía prestada:** Este tipo de informe muestra la bibliografía que ha sido prestada en un determinado intervalo de tiempo, y permite seleccionar el orden de los elementos listados y el formato de salida del informe.

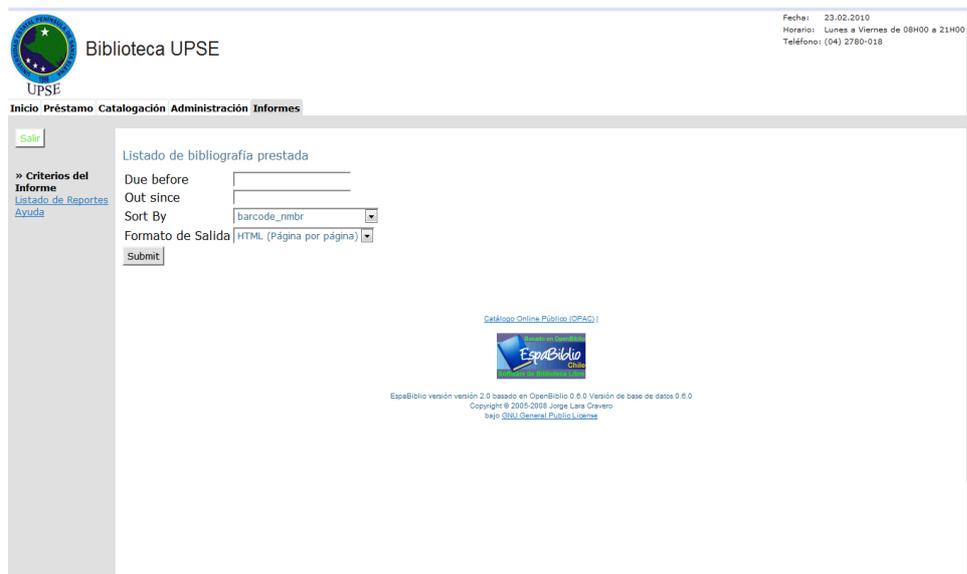


GRÁFICO 156: Listado de Bibliografía Prestada.

Fuente: Autores.

- **Peticiones de préstamo que contienen información de contacto con el socio:**
Este informe presenta una lista de las reservaciones de bibliografía que se han realizado en un intervalo de tiempo especificado por el usuario. Al igual que los demás reportes aquí también se puede seleccionar el orden de los elementos en la lista y el formato de salida de ésta.



GRÁFICO 157: Reporte de prestamos con información del socio.

Fuente: Autores.

ANEXO # 13



MANUAL DE IMPLEMENTACIÓN E INTEGRACIÓN DE CENTOS DS

INTRODUCCIÓN

CentOS Directory Server es un servidor de directorios basado en RedHat Directory Server, a manera de bifurcación desde el proyecto original tal como la distribución Fedora se deriva de RedHat Enterprise Linux.

El sistema base de CentOS DS está licenciado bajo GPLv2, y RedHat (quién posee el copyright) incluye una excepción para poder asociarlo con software no GPL.

PRINCIPALES CARACTERÍSTICAS DE CENTOS DIRECTORY SERVER

Entre las principales características que este servidor de directorios se encuentran:

1. Soporte de LDAPv3
2. Existe una documentación extensa, ya que utiliza la de RedHat Directory Server, y que puede ser accedida desde internet.
3. Soporta autenticación segura con SSLv3, TLSv1 y SASL.
4. Implementa características avanzadas de replicación.
5. Puede sincronizar usuarios y grupos con Active Directory.
6. Posee una interfaz gráfica de administración muy completa que centraliza todo lo relacionado al manejo de servicios, usuarios, grupos, logs, respaldos, etc.

PASOS PREVIOS A LA INSTALACIÓN

COMPROBANDO EL HOSTNAME

Para que no haya problemas con la instalación, es necesario que el nombre del equipo servidor (hostname) en el cual estamos realizando la instalación corresponda al nombre completo del mismo (ejm: mail.upse.edu.ec).

COMPROBANDO LA VERSIÓN DEL SISTEMA OPERATIVO

De acuerdo a la versión de CentOS Directory Server, es necesario realizar una verificación previa a fin de garantizar que es totalmente compatible con la versión de CentOS sobre la que se va a realizar la instalación.

OBTENIENDO EL ARCHIVO DEL REPOSITORIO TESTING

Es necesario sincronizar la información de este repositorio para poder hacer la instalación.

A continuación se detallan los pasos necesarios para lograrlo:

```
# cd /etc/yum.repos.d/  
# wget -v http://dev.centos.org/centos/5/CentOS-Testing.repo
```

EXCLUYENDO UNA VERSIÓN ESPECÍFICA DE JAVA DEL REPOSITORIO DE TESTING

En el archivo que describe el repositorio es necesario agregar la línea `exclude=java-1.7.0-icedtea`, con el fin de excluir este paquete para ser instalado.

INSTALANDO ALGUNAS DEPENDENCIAS NECESARIAS

Entre las dependencias que es necesario instalar se encuentran:

- xorg-x11-xauth
- bitstream-vera-fonts
- dejavu-lgc-fonts
- urw-fonts

Para lo cual utilizamos el comando yum, tal como se observa a continuación:

```
# yum install xorg-x11-xauth bitstream-vera-fonts dejavu-lgc-fonts urw-fonts
```

INSTALACIÓN DE CENTOS DIRECTORY SERVER

Para instalar el meta paquete que contiene la distribución de CentOS Directory Server, es necesario habilitar el repositorio de testing, para que el meta paquete sea obtenido desde allí. El comando para realizar ambas operaciones es el siguiente:

```
# yum --enablerepo=c5-testing install centos-ds
```

VERIFICANDO LA VERSIÓN DE JAVA EN USO

Es de vital importancia que java se encuentre instalado en el equipo para el funcionamiento de CentOS DS, por tanto, para confirmar la versión instalada, utilizamos el siguiente comando:

```
# rpm -qa | grep java
```

```
# java -version
```

CONFIGURANDO EL SERVICIO DE DIRECTORIO

El siguiente paso es configurar nuestro servicio de directorio, para esto ejecutamos el comando `setup-ds-admin.pl` lo que nos permitirá definir todos los parámetros necesarios:

```
# /usr/sbin/setup-ds-admin.pl
```

A continuación se indican las preguntas que realiza este utilitario:

- Would you like to continue with set up? [yes]: .- ¿Desea continuar con la configuración?, a lo cual se responde con yes.

- Do you agree to the license terms? [no]: .- ¿Está de acuerdo con los términos de la licencia?, a lo cual se responde con yes.

- Would you like to continue? [no] : .- ¿Desea continuar?, esta pregunta se realiza después de realizar una revisión en el sistema para determinar posibles problemas que afecten al desempeño del servicio, si no se detecta ningún problema grave se continúa respondiendo yes.

- Choose a setup type [2]: .- Seleccione un tipo de configuración, Permite seleccionar entre dos configuraciones predefinidas (1. Express o 2. Typical) o una configuración personalizada (3. Custom), por lo que es necesario responder con el número correspondiente a la opción seleccionada.

- Computer name [mail.upse.edu.ec]: .- Nombre de la computadora, con esta opción definimos el nombre del servidor al que habrá que referenciarse cuando necesitemos acceder al servicio de directorio. Es importante verificar si el nombre mostrado es el correcto y aceptarlo, o modificarlo.

- System User [nobody]: .- Usuario del sistema.

- System Group [nobody]: .- Grupo del sistema.

- Do you want to register this software with an existing configuration directory server? [no]: .- ¿Desea registrar este software con un servidor de configuración de directorio existente?, permite que el software instalado sea configurado con un servidor de directorio existente, si este es el caso, deberá escribirse la URL de este servidor incluyendo el número de puerto utilizado para establecer la conexión.

- administrator ID [admin]: .- Identificador del administrador, permite establecer el nombre del usuario administrador del servicio de directorio.

- Password: .- Contraseña.

- Password (confirm): .- Confirmación de contraseña.

- Administration Domain [upse.edu.ec]: .- Dominio Administrado.

- Directory server network port [389]: .- Puerto de red del servidor de directorio, establece el número de puerto mediante el cual se establecerá la conexión con el servidor de directorio.

- Directory server identifier [mail]: .- Identificador del servidor de directorio.

- Suffix [dc=upse, dc=edu, dc=ec]: .- Sufijo.

- Directory Manager DN [cn=Directory Manager]: .- DN del Administrador de directorio.

- Password: .- Contraseña.

- Password (confirm) : .- Confirmación de contraseña.

- Administration port [9830] : .- Puerto de administración, número de puerto mediante el cual se establecerá la conexión con el servidor para tareas de administración.

- Are you ready to set up your servers? [yes]: .- ¿Está listo para configurar sus servidores?, en este punto el utilitario ha recopilado toda la información necesaria para la configuración, por tanto, al responder yes, se van a aplicar todos los parámetros definidos.

ALGUNAS COMPROBACIONES POSTERIORES

A continuación se presentan algunos comandos que permitirán comprobar el correcto funcionamiento del servidor instalado.

Verificar el inicio del servicio de directorio:

```
# /etc/init.d/dirsrv status
```

Verificar si se activó el puerto 389:

```
# netstat -tln | grep 389
```

Comprobar si se inició el servicio de administración:

```
# /etc/init.d/dirsrv-admin status
```

Esta última verificación es sumamente importante ya que a través de este servicio podremos logearnos a la aplicación de administración cuando sea necesario, desde donde se tiene control del árbol del directorio, se puede realizar operaciones como

detención, inicio y reinicio del servicio de directorio (naturalmente también se puede hacer desde una línea de comandos), entre otras cosas.

LANZANDO LA CONSOLA DE ADMINISTRACIÓN

Para abrir la consola de administración, utilizamos el comando `centos-idm-console` tal como se muestra a continuación:



GRÁFICO 158: Lanzando la consola de administración

En llamada al comando indicado, puede observarse el puerto 9830 el cual se asignó durante el proceso de configuración, en caso de haberse asignado otro, entonces este deberá ser indicado al ejecutar el comando.

Como resultado de este comando, se presenta una ventana de autenticación en la cual se ingresa el nombre del usuario (por defecto `admin`, a menos que se haya especificado otro durante el proceso de configuración), tal como se observa a continuación:

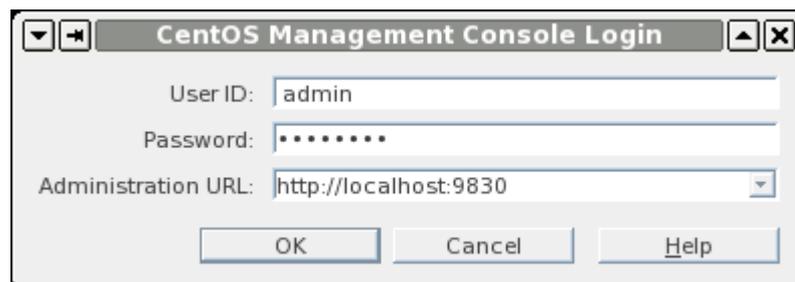
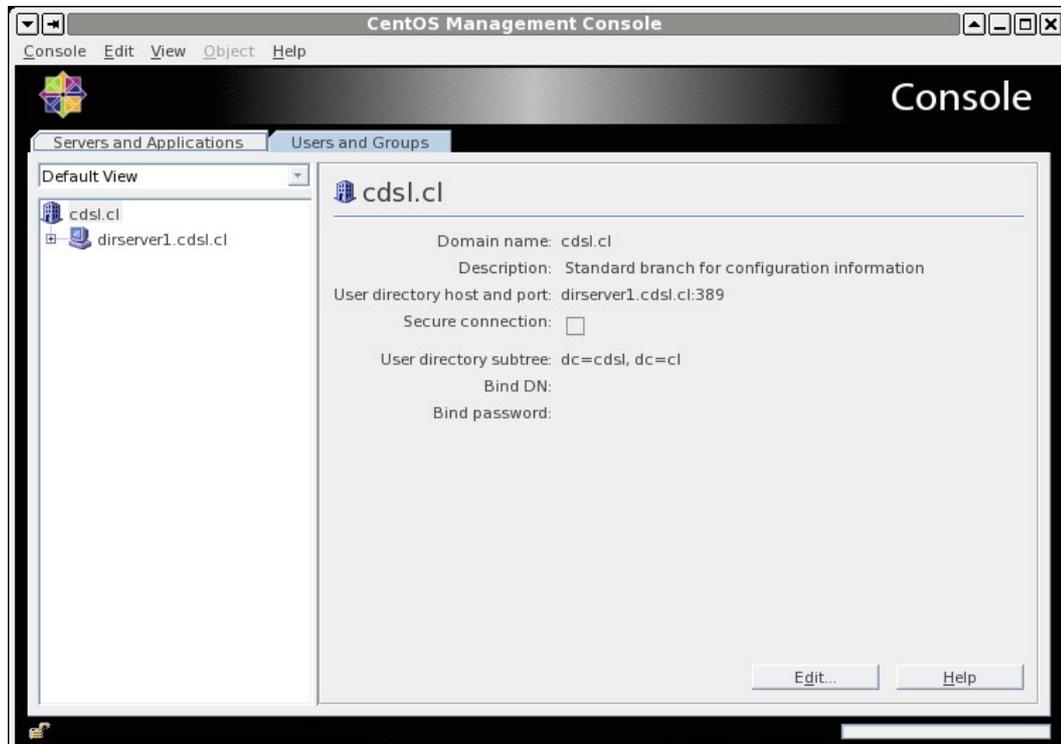


GRÁFICO 159: Autenticación de la consola de administración

Finalmente tenemos acceso a la consola de administración.



En esta parte se creó un usuario, el cual tiene por DN (distinguished name, o nombre distintivo por decirlo de alguna forma) lo siguiente :

```
"uid=lvivero,ou=people,dc=upse,dc=edu,dc=ec"
```

Entonces, utilizando una de las herramientas que provee `openldap-clients.i386`, `ldapsearch`, se realiza una búsqueda simple, especificando el DN y password del usuario, de la siguiente forma:

```
$ ldapsearch -x -D "uid=lvivero,ou=people,dc=upse,dc=edu,dc=ec" -w L34rn1n6  
-b "ou=people,dc=upse,dc=edu,dc=ec" "objectclass=*"
```

dando como resultado algo similar a lo siguiente:

```
# extended LDIF  
#  
# LDAPv3  
# base <ou=people,dc=upse,dc=edu,dc=ec> with scope subtree  
# filter: objectclass=*<br># requesting: ALL<br>#<br># People, upse.edu.ec<br>dn: ou=People, dc=upse, dc=edu, dc=ec<br>objectClass: top<br>objectClass: organizationalunit<br>ou: People<br># LVivero, People, upse.edu.ec<br>dn: uid=LVivero,ou=People, dc=upse, dc=edu, dc=ec<br>mail: morenisco@noc-root.net<br>uid: LVivero<br>givenName: Luis<br>objectClass: top<br>objectClass: person<br>objectClass: organizationalPerson<br>objectClass: inetorgperson<br>sn: Vivero<br>cn: Luis Vivero<br># search result<br>search: 2<br>result: 0 Success<br># numResponses: 3<br># numEntries: 2
```

Como se observa, se obtuvo la salida fruto de la búsqueda, permitida por un bind correcto.

Si uno ingresa mal el password, o si la autenticación no funciona, entonces obtendríamos lo siguiente:

```
$ ldapsearch -x -D "uid=lvivero,ou=people,dc=upse,dc=edu,dc=ec" -w L34rn1n7  
-b "ou=people,dc=upse,dc=edu,dc=ec" "objectclass=*"
```

```
ldap_bind: Invalid credentials (49)
```

ANEXO # 14



GUÍA DE IMPORTACIÓN DE MATERIAL BIBLIOGRÁFICO CON ESTÁNDAR UNIMARC

INTRODUCCIÓN

El formato UNIMARC es un estándar internacional utilizado en la catalogación de material bibliográfico.

Mediante este estándar se permite normalizar la información descriptiva de los materiales contenidos en bibliotecas.

PROCESO DE IMPORTACIÓN DE INFORMACIÓN

El proceso de importación de información, consistió en una serie de pasos que permitieran almacenar la información proveída por la biblioteca universitaria en la base de datos utilizada por la aplicación OpenBiblio. A continuación se especifica la secuencia de estos pasos.

OBTENCIÓN DE LA INFORMACIÓN FUENTE

La información fue obtenida en una hoja de cálculo en formato XLS.

IDENTIFICACIÓN DE LOS CAMPOS CONTENIDOS EN LA HOJA DE CÁLCULO

En la hoja de cálculo proveída, se identificó cada uno de los campos que describían el material bibliográfico, entre los que se encontraban: un identificador único, el nombre del material, el ISBN, el nombre del autor, el nombre de la editorial, el año de edición, entre otros.

LOCALIZACIÓN DE CÓDIGOS MARC PARA LAS COLUMNAS DE DATOS

Al almacenar la información de la base de datos de la aplicación OpenBiblio, se necesitaba ubicar el código MARC que diferenciaría a cada uno de los atributos del material bibliográfico, para lograr este fin accedimos a las siguientes páginas que proveían dichos códigos.

<http://www.loc.gov/marc/bibliographic/>

<http://www.loc.gov/marc/bibliographic/examples.html>

EXPORTANDO LOS DATOS

La primera fase de exportación consistió en tomar de la hoja de cálculo grupos de 1000 registros y con ellos crear un archivo de texto separado por tabuladores y delimitado por comillas dobles.

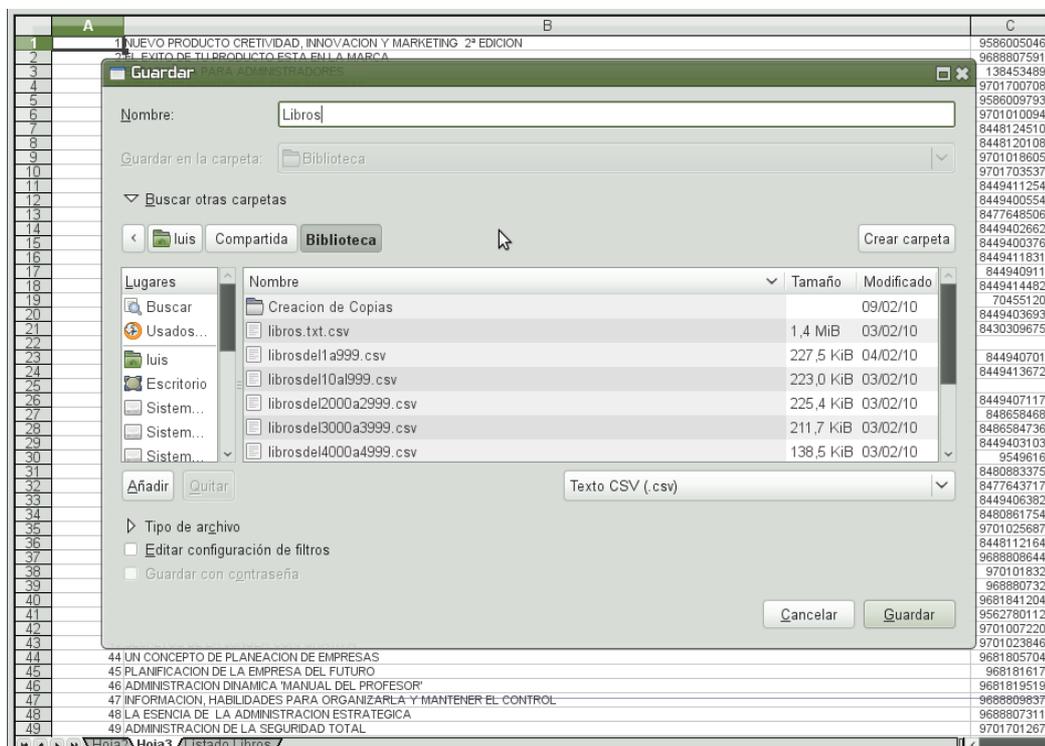


GRÁFICO 161: Exportando a archivos planos

CODIFICANDO LA INFORMACIÓN

Para aplicar la codificación MARC se utilizó una aplicación llamada MarcEdit 5.2 la cual permite generar un archivo de datos con codificación MARC a partir de un archivo plano.

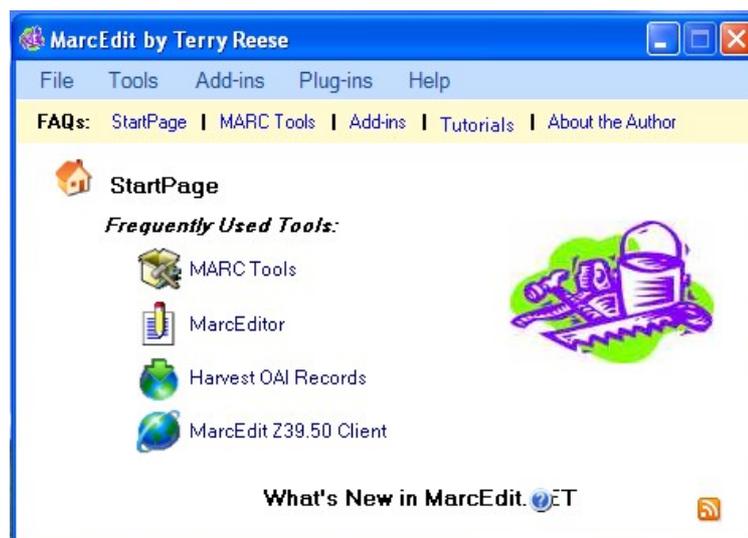


GRÁFICO 162: Interface principal de la aplicación MarcEdit

El procedimiento consiste en utilizar la herramienta “Delimited Text Translator” ubicada en el menú “Add-ins”.

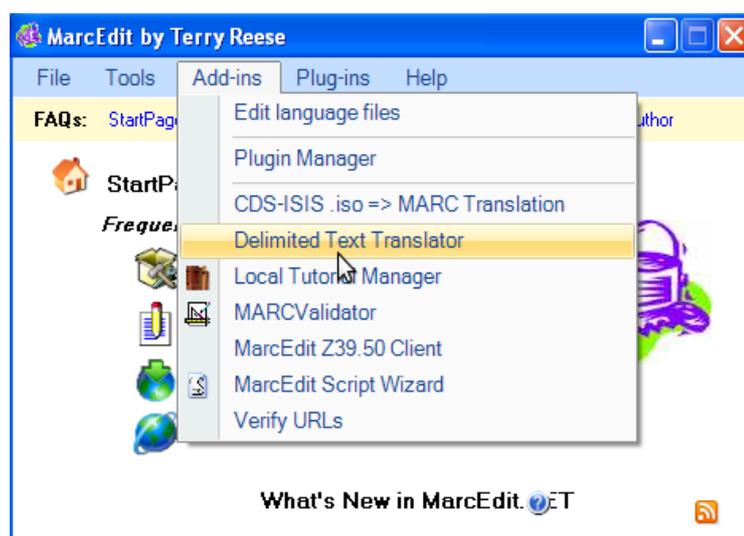


GRÁFICO 163: Opción Delimited Text Translator

Esta opción permite seleccionar el archivo que será utilizado para generar la información válida que posteriormente será subida a OpenBiblio.

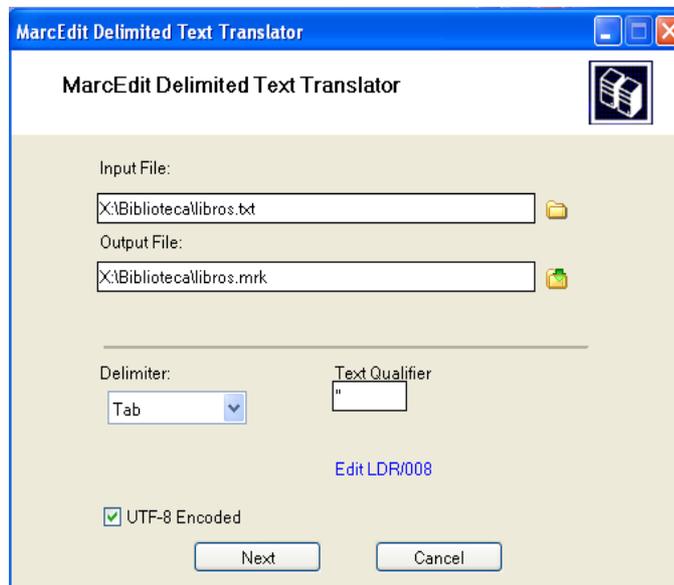


GRÁFICO 164: Selección del archivo a convertir

Se debe especificar el código MARC correspondiente a cada una de las columnas contenidas en el archivo, las cuales fueron previamente obtenidas.

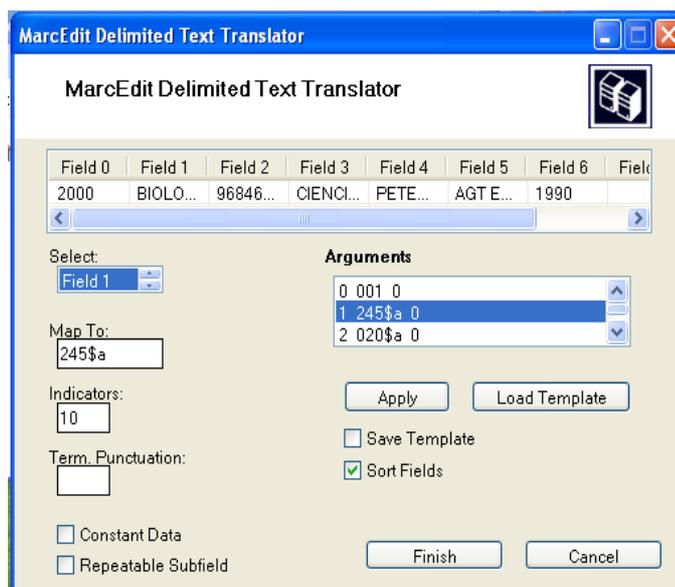


GRÁFICO 165: Especificando los código MARC

La aplicación permite la generación de una plantilla con la asignación de los códigos MARC a cada una de las columnas de datos, por tanto, puede utilizarse esta plantilla para las siguientes grupos de 1000 registros a convertir siempre que no se cambie el orden de los datos.

IMPORTANDO LOS DATOS EN OPENBIBLIO

Finalmente con el archivo de datos, sólo basta con utilizar la opción de importación proveída por la aplicación OpenBiblio, que para nuestro caso se encuentra en la ruta:

http://www.upse.edu.ec/openbiblio/catalog/upload_usmarc_form.php

The screenshot shows the OpenBiblio web interface. At the top left is the UPSE logo (Universidad Politécnica del Estado de los Andes) and the text 'Biblioteca UPSE'. To the right, contact information is listed: 'Fecha: 16.05.2010', 'Horario: Lunes a Viernes de 08H00 a 21H00', and 'Teléfono: (04) 2780-018'. Below this is a navigation menu with tabs: 'Inicio', 'Préstamo', 'Catalogación', 'Administración', and 'Informes'. The 'Catalogación' tab is active. On the left side, there is a sidebar with a 'Salir' button and several links: 'Búscar material', 'Nuevo material', '» Subir datos Marc', 'Consultar Biblioteca del Congreso', and 'Ayuda'. The main content area contains a form for data importation. It includes a 'Test de importación' section with radio buttons for 'Verdadero' and 'Falso' (selected). Below this is a text input field for 'Entrada de archivo USMarc:' and an 'Examinar...' button. A 'Predeterminados:' section contains three dropdown menus: 'Colección:' (set to 'General'), 'Tipo de material:' (set to 'Libros'), and 'Mostrar en OPAC:' (set to 'Si'). At the bottom of the form is a button labeled 'Archivo Almacenado'. In the bottom right corner, there is a link to 'Catálogo Online Público (OPAC)' and a logo for 'EspaBiblio Chile' with the text 'Basado en OpenBiblio' and 'Software de Bibliotecas Libres'.

GRÁFICO 166: Importación en OpenBiblio

GLOSARIO DE TÉRMINOS

Autenticación

Acto de establecimiento o confirmación de algo (o alguien) como auténtico, es decir que reclama hecho por, o sobre la cosa son verdadero. La autenticación de un objeto puede significar (pensar) la confirmación de su procedencia, mientras que la autenticación de una persona a menudo consiste en verificar su identidad. La autenticación depende de uno o varios factores.

Certificado Digital

Un certificado digital es un documento digital mediante el cual un tercero confiable (una autoridad de certificación) garantiza la vinculación entre la identidad de un sujeto o entidad y su clave pública.

Código Fuente

El código fuente de un programa informático (o software) es un conjunto de líneas de texto que son las instrucciones que debe seguir la computadora para ejecutar dicho programa. Por tanto, en el código fuente de un programa está descrito por completo su funcionamiento.

Escalable

La propiedad deseable de un sistema, una red o un proceso, que indica su habilidad para extender el margen de operaciones sin perder calidad, o bien manejar el crecimiento continuo de trabajo de manera fluida, o bien para estar preparado para hacerse más grande sin perder calidad en los servicios ofrecidos.

Framework

Conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular, que sirve como referencia para enfrentar y resolver nuevos problemas de índole similar. En el desarrollo de software, un framework es una estructura conceptual y tecnológica de soporte definida, normalmente con artefactos o módulos de software concretos, en base a la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros programas para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Hipertexto

Se refiere a un tipo de texto electrónico, estructurado en una serie de bloques conectados entre sí. Esos bloques de información se almacenan en nodos, que se conectan mediante enlaces. El resultado es una red hipertextual, más o menos compleja, que el usuario puede recorrer con distintos itinerarios.

HTML (HyperText Markup Language)

Lenguaje de Marcas de Hipertexto Lenguaje para elaborar páginas Web. Permite especificar al navegador la estructura o el formato de los documentos de hipertexto del WWW.

Intranet

Básicamente una Intranet es una red tipo Internet pero de uso interno. Por ejemplo, una intranet podría ser la red corporativa de una empresa que utilizara protocolo TCP/IP y servicios similares como WWW para el intercambio de información y la comunicación interna.

LAN (Local Area Network) o Red de Área Local.

Red que conecta ordenadores y recursos situados físicamente cerca, desde unos pocos metros a un kilómetro de distancia (por ejemplo, la red de un edificio o de una oficina). Se caracterizan por ser redes muy rápidas, desde 4 Mbps hasta 600 Mbps. Las LANs son redes privadas que se emplean para compartir recursos (impresoras, por ejemplo) e intercambiar información.

Single Sign-On

Procedimiento de autenticación que habilita al usuario para acceder a varios sistemas con una sola instancia de identificación.

WWW (World Wide Web) o Telaraña Alrededor del Mundo.

Es una forma de organizar y acceder a la información multimedia on-line distribuida en Internet. Soporta los protocolos tradicionales de Internet. Desde el punto de vista del usuario es una interfaz gráfica que sirve como punto de entrada a todo tipo de información que es presentada mediante hipermedia (en forma de páginas Web) y que es accesible a través de hiperenlaces.

En el WWW los documentos, formados por textos, imágenes y sonidos, que están almacenados en ordenadores dispersos por todo el planeta, se interconectan entre sí. Para acceder a ellos hay que contar con un programa navegador, que sabe cómo interpretar los contenidos de las páginas Web.

ABREVIATURAS

Término	Inglés	Español
GNU	GNU is Not Unix	GNU No es Unix
GPL	GNU General Public License	Licencia Pública General de GNU
BSD	Berkeley Software Distribution	Distribución de Software Berkeley
MPL	Mozilla Public License	Licencia Pública de Mozilla
ILS	Integrate Library System	Sistema Integrado de Biblioteca
CMS	Content Managment System	Sistema de Administración de Contenido
PHP	PHP Hypertext Pre-processor	Pre-procesador de Hipertexto PHP
JDK	Java Development Kit	Herramientas de Desarrollo Java
CAS	Central Authentication Service	Servicio de Autenticación Centralizada
SSL	Secure Sockets Layer	Protocolo de Capa de Conexión Segura
LDAP	Lightweight Directory Access Protocol	Protocolo Ligero de Acceso a Directorios
JVM	Java Virtual Machine	Máquina Virtual de Java
OPAC	Online Public Access Catalog	Catálogo en línea
MARC	Machine Readable Cataloging	Catalogación legible por máquina
XML	Extensible Markup Language	Lenguaje de marcas extensible