



INSTITUTO SUPERIOR MINERO METALÚRGICO
“Dr. Antonio Núñez Jiménez”.
Facultad de Metalurgia - Electromecánica
Moa, Holguín

*Trabajo de Diploma para optar por el
Título de Ingeniería en Informática*

Título

Diseño e Implementación del Portal-Intranet v2.0 para el ISMM

Autor:

Ruber Enrique Serrano Saltaren

Tutor:

Dabiel González Ramos

**Moa
Julio, 2010**

Declaración de autoría

Declaro que soy el único autor de este trabajo y autorizo al Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez” y al Departamento de Informática para que hagan el uso que estimen pertinente con el mismo.

Para que así conste firmamos la presente a los _____ días del mes de _____ del 2010.

Ruber E. Serrano Saltaren

Firma autor

Ing. Dabiel González Ramos

Firma tutor

Agradecimientos

*A mis padres primero que todo, porque sin ustedes nada de esto fuese posible.
Mami, por ser el alma de todo lo que hago, eres mi fuente de inspiración y la razón del por qué siempre quiero llegar lejos para lograr que te sientas orgullosa de mi.*

Papi, por ser la fuerza que me impulsa y ser mi ejemplo a seguir, al igual que mi madre, eres lo más importante de mi vida.

A mi niña (Ruth E. Serrano), la luz que me acompaña y guía en todo camino.

A hermanos, Maryanis Gámez, Juber Serrano, Yenisel Serrano, por siempre estar ahí, ayudándome, dándome fuerzas para continuar y a mi hermanito Uber Serrano, recuerda que nunca se está en la cima, tenemos que continuar con más ímpetu para darle toda la alegría que se merecen nuestros padres.

A toda mi familia, mis abuelas, mis tíos/tías, primos, a Angel A. y Marlenis P.

A mi tutor Dabiel González Ramos, por ser un gran amigo y siempre estar ahí, ayudándome muchísimo.

El pollo (Reinier Asencio Correa) por ser el mejor de los amigos en todos los tiempos, un amigo por siempre y para siempre.

Andy (Andrés Serrano), sabes que eres un gran amigo, eres de los que no fallan, aún en los momentos más difíciles.

A Mi pequeña (Glaudis Cobas), por ser para mí una hermana más. Yeni (Yenislín Cardoza E.), por ser alguien muy especial para mí. Las quiero muchísimo.

A Eliza, Elida, Yumi, Sai y a Melicita que me matan si se me olvidan. Ja, ja, ja, ja.

A Mayito (Miguel Ramírez), un verdadero hermano en todo este período de la Universidad. A ti te deseo todo lo mejor, cuídate mi hermano.

A mis amigos: Paki (Jairo O. Blanco), Mary (María Ester), Alexi Luperón, Adrianoft (Adrián Cadena), el Yunior (Henry Campo), Chinolamon (Ramón Borges), el Rolan, Jhonny, la Java, la Mellósí, Marianny, Antonio Plano.

A mi aula Info2005 por compartir este tiempo conmigo, a mis mejores amigos Mayito, Roger, Bling, el Yoa. Ah y a los del Naruto, este es mi camino Ninja. Jajajaj...

“ No hay hermosura mayor que el agradecimiento...! Desventurado el que no sabe agradecer! “. A todos de corazón, muchas gracias.

Dedicatoria

No tengo la gracia de poder decir que soy una persona a la cual Dios le dio la virtud de expresar todo lo que siente con palabras lindas, por tal motivo, cuando analizo qué frases podrían plasmar todo el cariño y admiración que siento por ustedes, es que me doy cuenta que aún, el más versátil de los escritores, no sería capaz de expresarlo. A ustedes me debo y doy gracias por el privilegio de tener semejantes padres. Les dedico este trabajo con la esperanza de enaltecer ese orgullo que es recíproco.

Dedico también el presente trabajo a la personita que dentro de mi corazón es la luz gigantesca que ilumina mi vida, el orgullo y el brillo de mis ojos, a ti mi querida niña, Ruth E.

Resumen

La informática:

Constituye el conjunto de conocimientos científicos y de técnicas que hacen posible el tratamiento de la información por medio de computadoras. A través de esta ciencia la mayoría de los procesos que diariamente tienen lugar en la sociedad son sensibles a la automatización. En Cuba la informatización es una directiva de trabajo la cual proporciona un alto índice de eficiencia a los procesos y servicios que son objeto de automatización.

Las tecnologías de la información y las comunicaciones se han convertido en un instrumento real e indispensable para miles de instituciones a las cuales hoy en día les resulta prácticamente imposible el no poder contar con este tipo de técnica pues la misma se ha convertido en una realidad objetiva a nivel mundial.

En nuestra enseñanza el avance tecnológico y el manejo de la información mediante la electrónica, la informática y las telecomunicaciones han contribuido a que las diferencias entre la recolección de la información, su correcto procesamiento y su distribución vayan desapareciendo paulatinamente.

Nuestro Instituto no se encuentra exento de esta revolución que cada vez es más acelerada. Esta se basa en el perfeccionamiento y mejoramiento del entorno de la comunidad y en el logro de avances en el quehacer cotidiano y a su vez que este Portal-Intranet v2.0 se convierta en la puerta de entrada a todos los servicios, además de identificar nuestros usuarios se sientan identificados por el mismo.

Abstract

The computer science:

It constitutes to the group of scientific knowledge and techniques that make possible the treatment of information by means of computers. Through this science, most of the daily processes in society are sensitive to the automation. In Cuba the computerization, is a working directive which provides a high index of efficiency to the processes and services that which are objects of automation. Information technology and communication have become a real and indispensable instrument for thousands of institutions today .It is practically impossible not to implement a technique of this type in any organization. It has become an important objective in the real world.

In the world of education the technological advances and the handling of information by means of electronics, computer science and telecommunications have contributed to the differences among the gathering of the information; their correct execution and their distribution gradually disappears.

Our institute is not exempt of this revolution. It is based on its improvement of the community achieving advances in the daily chores and also achieving in turn this Portal-Intranet v2.0 which becomes the entrance door to all the services provided to our users.

INDICE

Introducción.....	1
Capítulo 1: Fundamentación teórica.....	7
Introducción:	7
1.1 Antecedentes del Internet	7
1.2 Aspectos referentes a la Intranet	8
1.3 Extranet	10
1.4 Concepto y Aspectos sobre Software Libres:	10
1.5 Las aplicaciones Web	12
1.5.1 Diferencia que existe entre un sitio Web y una Aplicación Web	13
1.6 Percepción de la Filosofía Cliente-Servidor.	14
1.7 Percepción de Servidor Web.	14
1.7.1 ¿Qué denominamos Hypertext Transfer Protocol (HTTP)?	14
1.7.2 ¿Qué denominamos Páginas Web?	15
1.8 Lenguajes de programación para el desarrollo de Aplicaciones Web ...	15
1.8.1 Hypertext Markup Language (HTML):.....	15
1.8.2 Active Server Pages (ASP).....	16
1.8.3 ASP.NET:	17
1.8.4 Java Server Pages (JSP).....	17
1.8.5 Personal Home Page (PHP)	18
1.9 Herramientas para el desarrollo de Aplicaciones Web	20
1.9.1 Sistemas gestores de bases de datos (SGBD) :	20
1.9.2 Microsoft SQL Server:.....	20
1.9.3 PostgreSQL y MySQL:.....	20
1.9.4 MySQL:	21
1.9.5 PostgreSQL:	22
1.10 Content Management System (CMS)	24
¿Qué es un CMS?	24
1.10.1 Zope.....	25
1.10.2 Joomla	26
1.10.3 Drupal	27
Características de Drupal	27
Generales.....	27
Gestión de usuarios	28
Gestión de contenido	28
Blogging	29
Plataforma.....	29
Administración y Análisis.....	30
Rendimiento y escalabilidad	30
1.11 Aspectos generales sobre las Metodologías Usadas	33
1.11.1 Metodologías tradicionales:	33
1.11.2 Metodologías Ágiles:.....	34
¿Qué propone XP?	38
Capítulo 2 Planificación y Diseño.	42
Introducción	42
2.1 Planificación.....	42

2.1.1	Concepción del sistema	42
2.1.2	Modelo de Historias de usuario del negocio	43
2.1.3	Lista de Reserva del Producto (LRP)	44
2.1.4	Historias de usuario	44
2.1.5	Lista de riesgos	45
2.1.6	Modelo de diseño	45
Capítulo 3: Desarrollo del Sistema		46
Introducción		46
3.1	Desarrollo de las Iteraciones	46
3.1.1	Iteración 1	47
Tabla 1: Historias de Usuarios de la 1ra Iteración		47
3.1.1.1	Drupal como Administrador de Contenido (CMS) para la Intranet2.0	47
Figura 1: Entorno de trabajo de Drupal		48
3.1.1.2	Instalando Drupal	51
3.1.1.3	Definición de Plantillas	56
3.1.2	Iteración 2	58
3.1.3	Iteración 3	62
¿Qué se entiende como Menú de un portal?		63
Fase de elaboración		64
3.1.4	Iteración 4	66
Importante sobre los contenidos en Drupal		66
3.1.5	Iteración 5	67
Interesante		67
Creación de Contenido		67
Gestión de contenidos		68
Publicación		68
Presentación		69
Mantenimiento y Actualización		69
Categorías		70
Agregar Contenido		70
3.1.6	Iteración 6	71
Capítulo 4: Pruebas de aceptación		73
Introducción		73
Pruebas		73
Pruebas de aceptación		73
Conclusiones del capítulo		80
Capítulo 5: Estudio de Factibilidad		81
Introducción		81
1.1	Efectos Económicos	81
1.2	Efectos directos	81
1.2.1	Efectos indirectos	82

1.2.2	Externalidades	82
1.2.3	Intangibles	82
1.3	Beneficios y Costos Intangibles en el proyecto	83
1.4	Ficha de Costo	83
Conclusiones del Capítulo		85
Conclusiones Generales		86
Recomendaciones		87
Bibliográfica		88
Anexos		¡Error! Marcador no definido.
Glosario de Términos		90

Índice de tablas

Tabla 1: Historias de Usuarios de la 1ra Iteración	47
Tabla 2: Historias de Usuarios de la 2da Iteración	58
Tabla 3: Historias de Usuarios de la 3ra Iteración	62
Tabla 4: Historias de Usuarios de la 4ta Iteración	66
Tabla 5: Historias de Usuarios de la 5ta Iteración	67
Tabla 6: Historias de Usuarios de la 6ta Iteración	71
Tabla 7: Plantilla Prueba de Aceptación	74
Tabla 8: Caso de Prueba de Aceptación definir plantilla del usuario.®	75
Tabla 9 : Caso de Prueba de Aceptación definir plantilla del administrador.®	76
Tabla 10: Caso de Prueba de Aceptación Definición de los módulos funcionales del Portal.	76
Tabla 11: Caso de Prueba de Aceptación Implementación de los módulos funcionales del Portal.	77
Tabla 12: Caso de Prueba de Aceptación Definición de los Niveles de usuarios dentro del Portal.	77
Tabla 13: Caso de Prueba de Aceptación Implementación de los Niveles de usuarios dentro del Portal.	78
Tabla 14: Caso de Prueba de Definición de los menús de navegación en el Portal.	78
Tabla 15: Caso de Prueba Implementación de los menús de navegación en el Portal.	79
Tabla 16: Caso de Prueba Definición del contenido de información.	79
Tabla 17: Caso de Prueba Implementación del contenido de información.	80
Tabla 18: Caso de Prueba Implementación de la funcionalidad personalización del Portal a nivel de Usuario.	80

Introducción

Desde la antigüedad siempre han existido personas que han tenido el anhelo del dominio de la información pues el que contaba con ésta tenía el poder de modificar la forma de pensar de un individuo o de una comunidad por completo unido al logro de grandes cambios en dependencia de sus intereses, por la propiedad de la misma de ser flexible al cambio. A raíz de esto, muchas organizaciones e instituciones se dieron a la tarea de formar en todo el mundo especialistas de la información y las comunicaciones.

La informática:

Constituye el conjunto de conocimientos científicos y de técnicas que hacen posible el tratamiento de la información por medio de computadoras. A través de esta ciencia la mayoría de los procesos que diariamente tienen lugar en la sociedad son sensibles a la automatización. En Cuba la informatización es una directiva de trabajo la cual proporciona un alto índice de eficiencia a los procesos y servicios que son objeto de automatización. En la actualidad las tecnologías y comunicación de la información cobran vida en numerosas esferas de la sociedad lo cual trae consigo una revolución similar a la de la revolución industrial. Sólo debemos analizar cómo circula la información y qué papel juegan la Internet y las Redes de Área Local (LAN) en todo esto, siendo una herramienta importante para el buen desempeño de pequeñas y grandes empresas e instituciones (PyMes), dejando atrás todo un sin número de tabúes para convertirse en una necesidad imperante del mundo moderno. De manera general se dan pasos de avances muy sólidos hacia un futuro caracterizado por el rápido intercambio de la información.

Viéndolo desde otro punto de vista nos podemos dar cuenta que la información es actualmente el recurso más importante y valioso con que se cuenta en una institución o empresa y ha dejado de ser un privilegio de unos pocos para extenderse a toda una comunidad.

Las tecnologías de la información y las comunicaciones se han convertido en un instrumento real e indispensable para miles de empresas que hoy en día les

resulta prácticamente imposible el no poder contar con este tipo de técnica, ya que esta se ha convertido en una realidad objetiva a nivel mundial.

En nuestra enseñanza el avance tecnológico y el manejo de la información mediante la electrónica, la informática y las telecomunicaciones han contribuido a que las diferencias entre la recolección de la información, su correcto procesamiento y su distribución vayan desapareciendo paulatinamente

El Instituto va de la mano con esta revolución que cada vez es más acelerada basada en el perfeccionamiento y mejoramiento del entorno de la comunidad para el logro de avances en el quehacer cotidiano y la su vez que este Portal-Intranet se convierta en la puerta de entrada a todos los servicios, además de que nuestros usuarios se sientan identificados por el mismo.

Situación Problemática

En la actualidad el Instituto cuenta con un portal intranet con las características siguientes:

PHP Construido sobre: Linux WebCenter 2.6.31-14-generic #48-Ubuntu SMP
Fri Oct 16 144:26 UTC 2009 i686

Versión de la Base de Datos: 5.1.37-1ubuntu5

Versión de PHP: 5.2.10-2ubuntu6.3

Servidor Web: Apache/2.2.12 (Ubuntu)

Interfaz servidor web a PHP: apache2handler

Versión de Joomla!: Joomla! 1.0.13 Stable

Este a su vez presenta problemas tales como:

- ✓ La incapacidad del Portal-Intranet de proveer a sus usuarios informaciones actualizadas, puesto que la difundida presenta un alto grado de desactualización.
- ✓ Insuficiencia en el grado de cobertura.

- ✓ Escaso diseño del portal. Los colores tienen que corresponderse más con la Universidad y con el territorio donde se encuentra. Debe proporcionar imágenes más referidas a los temas que se tratan para no provocar el rechazo de los usuarios, ni desviación de la atención.
- ✓ Carece de una tipografía idónea para alcanzar una buena lectura e interpretación, así como la homogeneidad en todas las páginas del portal.
- ✓ Insuficiente integración del portal con relación a los procesos y temáticas de la organización, universo de usuarios y necesidades de información que presentan.
- ✓ Insuficientes prestaciones de servicios por lo que los usuarios consideran que el portal carece de importancia.
- ✓ No ocurrencia de citas bibliográficas en la información expuesta en el portal con el fin de permitir consultas.
- ✓ Existencia de un alto grado de ambigüedades y errores ortográficos.
- ✓ Gran parte de la información difundida en el portal carece de validez, de utilidad y a la vez de usabilidad para la comunidad universitaria razón por la cual las visitas al portal no se comporten como se desea.
- ✓ La imposibilidad de aumentar el tamaño de las letras en el portal.
- ✓ Al analizar que existen varios navegadores y que el usuario puede encontrarse en cualquiera de estos el sitio actual solo permite hacer esto cuando el usuario se encuentra en Firefox.
- ✓ El portal no cuenta con la existencia de una ayuda al usuario sobre la estructura de los contenidos y la navegación del sitio.
- ✓ Existe falta de organización en la estructura lógica de los contenidos incluidos por lo que se considera que no se ha realizado una buena

arquitectura de la información. Este es un aspecto clave para el desarrollo de cualquier aplicación Web.

- ✓ Hay incoherencia y falta de homogeneidad producto de la inexactitud de los títulos lo cual ocasiona la confusión de los usuarios.
- ✓ No cuenta con un mapa del sitio que incluya de manera jerárquica y organizada todos los contenidos con enlaces activos lo que proporciona al usuario una estructura global de los contenidos.

Problema Científico

Se plantea como problema científico la incapacidad del portal intranet de convertirse en la puerta de entrada a todos los sistemas y de satisfacer las necesidades y exigencias de los usuarios finales, así como la no integración con otros sistemas del centro.

Objeto de investigación

- Sistemas y Portales para el desarrollo de Intranet.

Campo de Acción

- Portal-Intranet del ISMM.

Idea a Defender

Se considera que si se diseña e implementa un Portal Intranet que tenga la totalidad de los servicios que son necesarios e incluso algunos que son indispensables para el desarrollo cotidiano, tanto de los profesores como de los estudiantes y que permita la integración con otros sistemas del centro, se podrá alcanzar una mayor interrelación usuarios – Portal Intranet.

Objetivo general

Diseñar e Implementar un Portal Intranet basado en servicios que permita satisfacer las necesidades y demandas de la comunidad universitaria y su

interrelación con los demás sistemas que se encuentran en explotación en nuestro centro.

Objetivos Específicos

- Realizar un análisis de la información que se publicará en el portal intranet para que no halla ambigüedad y que sea precisa.
- Diseñar una interfaz amigable y con tres niveles de navegación para que sea de fácil uso para los usuarios.
- Implementar una Base de Datos que permita el fácil y preciso acceso a las informaciones del sitio.
- Desarrollar un sistema de seguridad para garantizar el acceso correcto a la información.
- Documentar toda la información posible sobre el Portal-Intranet para facilitar su mantenimiento posterior.

Tareas Específicas

- Valoración de la intranet actual.
- Realización de una encuesta con el fin de descifrar las necesidades de la comunidad universitaria.
- Realización de un estudio de las tecnologías de desarrollo de Portales-Intranet.
- Diseño de la base de datos.
- Realización del levantamiento de los requerimientos del sistema.
- Realización de un estudio exhaustivo sobre la arquitectura a desarrollar.
- Desarrollo del diseño e implementación del Portal-Intranet.
- Implementación de los métodos para darle la funcionalidad requerida al portal.
- Desarrollo del Manual de usuario.

Métodos de investigación

Para dar un correcto cumplimiento a las tareas propuestas y para la obtención de datos imprescindibles que permitieron caracterizar el fenómeno objeto de estudio se emplearon los métodos del Nivel empírico (la observación, la encuesta y las entrevistas) y del Nivel teórico (el Histórico-lógico, Análisis y síntesis, Hipotético y deductivo, Sistemático-estructural) de la investigación científica.

Una vez que se han definido los métodos se puede abordar el tema a resolver. A través de la observación vio la funcionalidad que tiene el Portal-Intranet dentro del Instituto; las encuestas tuvieron un rol fundamental al posibilitar la captura de un sin número de criterios e información para llegar a las verdaderas exigencias y necesidades de la comunidad universitaria. Las entrevistas ayudaron a determinar los principales requisitos del sistema y las perspectivas de este alto centro docente.

Los métodos teóricos permitieron dar calidad a la investigación al proporcionar las condiciones para la interpretación de los datos obtenidos. El histórico-lógico dio la posibilidad de entender los antecedentes del objeto de estudio y comprender la naturaleza del mismo; el análisis y síntesis contribuyó a la recopilación y procesamiento de la información obtenida a través de los métodos empíricos y así arribar a conclusiones fiables durante la investigación. El hipotético-deductivo permitió la elaboración de una hipótesis, así como realizar el análisis del desarrollo del sistema y el sistemático-estructural permitió estructurar el control interno como un sistema con sus relaciones internas y externas.

Con la culminación del estudio epistemológico sobre el tema investigado y después de los resultados arrojados por el empleo de los métodos empíricos y teóricos el autor de este Trabajo de Diploma dio como **aporte** la posibilidad de contar con un Portal-Intranet v2.0 eficiente para el instituto ISMM que sea capaz de cumplir con las expectativas de la comunidad.

Capítulo 1:

En este capítulo se verán cuestiones generales sobre desarrollo de portales Web. Para ello se ofrece una breve descripción de distintos conceptos que le darán vida a esta investigación, las herramientas a utilizar para la implementación de la aplicación, las metodologías de investigación para el desarrollo del sistema y las ventajas de las aplicaciones Web y su interrelación con los software libres.

Capítulo 1: Fundamentación teórica

Introducción:

En este capítulo se abordarán los conceptos fundamentales para el desarrollo de aplicaciones Web, se realizará un estudio de las ventajas que estas ofrecen, las características generales de los sistemas de bases de datos y de los lenguajes de programación Web, las diferentes metodologías existentes para el desarrollo de aplicaciones, y de ellas cuáles son más convenientes a utilizar. Además de la importancia del uso del software libre, y las diferencias que existen entre sitios web y aplicaciones Web.

1.1 Antecedentes del Internet

Con el objetivo de interconectar máquinas remotas en los años 60 se comenzó a investigar la creación de una red de ordenadores para el Departamento de defensa de los Estados Unidos. Esta red de comunicación le permitió al Departamento la posibilidad de intercambiar información y de forma segura con recursos bien racionalizados y bajo costo, se dio a conocer bajo el nombre de ARPAnet. Este proyecto sirvió para conectar a científicos que fueron desarrollando y ampliando cada vez más las potencialidades de la misma.

Para los años 70 las Agencias de proyectos de investigación avanzada (ARPA) se dieron a la tarea de establecer una serie de protocolos o normas para que todas las redes que cumpliesen con estas pudieran interconectarse entre sí. Esta alcanzó su mayor relevancia en 1972 cuando ARPAnet presentó en octubre del propio año una red de 40 nodos y allí se propuso su conexión con otras redes internacionales. Por este motivo en 1972 fueron conectados todas las agencias y proyectos del departamento. Pero no fue sino hasta 1980 que

debido a esta funcionalidad de interconexión de redes que se le dio el nombre de Internet.

Esta gran red mundial, también reconocida como uno de los mayores logros tecnológicos alcanzados en la historia de la humanidad, trajo consigo el nacimiento de las denominadas aplicaciones Web y sitios Web las cuales dieron un impulso definitivo a la misma y constituyeron una realidad de lo que son capaces de lograr al dejar atrás la idea de comunicar ordenadores para llegar a la comunicación entre personas. Ejemplo de ello lo constituyen los correos electrónicos tan importantes y aplicados hoy en día en cualquier institución o empresa.

De manera breve podríamos definir Internet como una red de computadoras esparcidas por todo el mundo donde cada institución adquiere, instala y le da mantenimiento a sus computadoras.

Es notable analizar que el término Internet no es sinónimo de WWW o sea Word Wide Web, sino que este se enmarca dentro de Internet, siendo el WWW uno de los tantos servicios que se prestan en la gran red de redes y consiste en un sistema de información mucho más reciente empleada por la red como medio de transmisión.

1.2 Aspectos referentes a la Intranet

Intranet es un concepto muy novedoso y posee gran futuro dentro del mundo empresarial. El término “Intranet” surge a partir del 1995 para referirse a la “Internet internas”, o también llamadas “internet empresarial”, es decir, las redes corporativas basadas en la tecnología y los servicios de Internet adaptándolos a los límites físicos y a las características particulares de cada organización; siendo el servicio WWW el corazón de la misma.

Distribuir la información dentro de cualquier organización es una tarea compleja; el asegurar que esta sea confiable y que llegue a las manos correctas y en el momento adecuado, es aún más complejo, ya que una respuesta tardía carece de importancia, lleva consigo un costo generalmente alto, mucho trabajo administrativo y organizativo.

Intranet es simplemente una red TCP/IP de una empresa u organización que enlaza a los miembros de la organización y su información de manera tal que aumente la productividad, facilite el acceso a la información y convierta la navegación por los recursos y aplicaciones, de su entorno informático, en un proceso confiable y que necesite menos recursos que los métodos tradicionales. Esta se enriquece de los protocolos y normas abiertas que surgieron a partir de Internet. Demostrar sus potencialidades y tecnologías, aplicarlas a los requerimientos específicos de una institución u organización individual, es el más noble de los principios básicos de una Intranet.

Técnicamente una Intranet posibilita reducir notablemente los costos además proporcionar muchas otras ventajas como la comunicación y transferencia de datos en un período de tiempo muy corto o relativamente corto dentro de una organización. La consolidación de datos e información a través de una sola interfaz permite la gestión de documentos, así como la publicación de informes. El correo electrónico interno es una de las tantas funcionalidades que aportan las Intranets: la capacitación y soporte en línea, además de la programación de actividades. Otra cosa muy importante es su capacidad de atender una mayor audiencia de público. Brinda además información actualizada y precisa, la credibilidad de la fuente de información e información compartida, sin importar el sistema operativo del usuario. Solución escalable al utilizar protocolos y tecnologías abiertas.

1.2.1 Servicios básicos de una Intranet:

Los servicios básicos que componen una intranet son:

Servicio a usuarios:

Estos son los que suministran los recursos y aplicaciones al usuario final. Los más importantes de este son el compartimiento y la administración de la información, la comunicación entre usuarios de la Intranet, la navegación y el acceso a las aplicaciones.

Servicios de Red:

Permiten interconectar y proporcionar el entorno de red local siendo fundamentales: la gestión de la información, la seguridad y la administración.

1.3 Extranet

Conocida como una Red Privada Virtual (VPN por sus siglas en inglés), que utiliza Internet como medio de transporte de información entre varias Intranets. En la era de las redes corporativas, las Extranets, ofrecen a las empresas una nueva forma de sacar ventaja de sus inversiones en Redes Locales e Intranets. Funcionando como redes privadas dentro de Internet, las Extranets permiten compartir información y aplicaciones entre Intranets en diferentes territorios.

1.4 Concepto y Aspectos sobre Software Libres:

La implantación y adquisición del software libre es una de las principales políticas llevadas a cabo hoy en día en nuestro país. Se pretende ayudar a convencer a nuestra comunidad informática de que con software libre bajo la licencia GNU/GPL se le pueden dar solución a disímiles problemas encontrados en la actualidad sin necesidad de invertir en software y sistemas operativos propietarios.

Software libre: Es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. El software libre suele estar disponible gratuitamente en Internet, o a precio del coste de la distribución a través de otros medios; sin embargo no es obligatorio que sea así y, aunque conserve su carácter de libre, puede ser vendido comercialmente. Análogamente, el software gratis o gratuito (denominado usualmente Freeware) incluye en algunas ocasiones el código fuente, este tipo de software no es libre en el mismo sentido que el software libre, al menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa.

No se debe confundir software libre con software de dominio público. Este último es aquel por el que no es necesario solicitar ninguna licencia y cuyos

derechos de explotación son para toda la humanidad porque 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. Si un autor condiciona su uso bajo una licencia, por muy débil que sea, ya no es dominio público.

En resumen, el software de dominio público es la pura definición de la libertad de usufructo de una propiedad intelectual que tiene la humanidad porque así lo ha decidido su autor o la ley tras un plazo contado desde la muerte de éste, habitualmente 70 años.

Entre los años 60 y 70 del Siglo XX, el software no era considerado un producto sino un añadido que los vendedores de los grandes computadores de la época (los mainframes) aportaban a sus clientes para que éstos pudieran usarlos.

A finales de los 70, las compañías iniciaron el hábito de imponer restricciones a los usuarios, con el uso de acuerdos de licencia. En 1984, Richard Stallman comenzó a trabajar en el proyecto GNU y un año más tarde fundó la Free Software Foundation (FSF). Stallman introdujo una definición para free software y el concepto de "copyleft", el cual desarrolló para dar a los usuarios libertad y para restringir las posibilidades de apropiación del software.

La licencia GNU/GPL posibilita la modificación, redistribución del software, pero únicamente bajo esa misma licencia. Y añade, que si se reutiliza en un mismo programa código "A" licenciado 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 código "A" y "B" debe de estar bajo la licencia GNU/GPL.

El término software no libre se emplea para referirse al software distribuido bajo una licencia de software más restrictiva que no garantiza las libertades del software libre. 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.

La definición de software libre no contempla el asunto del precio; un eslogan frecuentemente usado es "libre como en libertad, no como en cerveza gratis" o en inglés "Free as in freedom, not as in free beer" (aludiendo a la ambigüedad del término inglés "free"), y es habitual ver a la venta CDs de software libre como distribuciones Linux. Sin embargo, en esta situación, el comprador del CD tiene el derecho de copiarlo y redistribuirlo. El software gratis puede incluir restricciones que no se adaptan a la definición de software libre —por ejemplo, puede no incluir el código fuente, puede prohibir explícitamente a los distribuidores recibir una compensación a cambio.

Los términos "libre" (software libre) y "gratis" (software gratis) son usados únicamente dentro del movimiento del software libre, aunque están extendiéndose lentamente hacia el resto del mundo. Otros defienden el uso del término open source software (software de código abierto, también llamado de fuentes abiertas). La principal diferencia entre los términos "open source" y "free software" es que éste último tiene en cuenta los aspectos éticos y filosóficos de la libertad, mientras que el "open source" se basa únicamente en los aspectos técnicos.

1.5 Las aplicaciones Web

Una aplicación es un conjunto de páginas Web interrelacionadas entre sí que visualizan diferentes partes de la información que se quiere mostrar a través de ella, los datos se almacenan en bases de datos que a su vez estas se encuentran en servidores remotos, es decir servidores Web, a las cuales se accede por medio de un navegador Web o una terminal móvil, gracias a la lógica de negocio a través de un Navegador (Browser), que se ejecutan en el servidor y al diseño del interfaz que es transferido a dichas terminales, utilizan las Tecnologías existentes para generar contenidos dinámicos y permitir a los usuarios del sistema modificar la lógica del negocio en el servidor. Por lo que se considera que de no existir lógica de negocios en el servidor, el sistema no puede considerarse una aplicación Web, sino como un sitio Web.

Se puede decir que estas gozan de tanta popularidad debido a la practicidad del navegador Web como cliente ligero, por la facilidad que poseen para

actualizarse y mantenerse brindando servicios sin distribuir e instalar software en miles de clientes; esto contribuye a que las aplicaciones Web sean consideradas una de las mejores herramientas para divulgar, gestionar y compartir la información al provocar que cada vez su uso sea mas factible e importante porque genera una mayor eficiencia en el campo de manipulación de grandes volúmenes de información.

1.5.1 Diferencia que existe entre un sitio Web y una Aplicación Web

La arquitectura de un Sitio Web es simple. Contiene como componentes principales: el Servidor Web, una Red y un Navegador Web o cliente. Por su parte una aplicación Web incluye además la aplicación en el Servidor que es la que permite al sistema manejar lógica de negocio y tener un estado.

Ventajas:

Superan a las aplicaciones de escritorio por su propiedad de compatibilidad multiplataforma ya que muchas de las tecnologías existentes hoy en día como Java, Flash, ASP y Ajax brindan la posibilidad de desarrollar programas capaces de soportar los principales sistemas operativos y cuando de información se trata siempre es importante estar actualizados y esta es otra de las ventajas que las aplicaciones Web ofrecen. Siempre se pone a disposición del usuario la ultima versión sin necesidad de inmiscuirse en sus hábitos o desviar su atención, al brindarle la posibilidad de descargar nuevas versiones y procedimientos de instalación.

La capacidad de tener múltiples usuarios concurrentes brinda la posibilidad de que varios usuarios de la misma aplicación puedan acceder por medio de la red a distintos datos de manera segura sin que ningunos de ellos note la presencia de los otros; esto es posible gracias a que estas permiten la encriptación u ocultamiento de la información de forma que nadie mas pueda hacer uso de esta y utilizarla con el fin de perjudicar a las involucrados. Estos se hacen llamar protocolos de comunicación. Las aplicaciones Web no necesitan ser descargadas, instaladas, ni configuradas.

1.6 Percepción de la Filosofía Cliente-Servidor.

Consiste en que existen computadoras, generalmente las menos potentes, que realizan peticiones y otras con mayor potencia de procesamiento a la escucha de estas peticiones, comunicándose entre ellas por un lenguaje común llamado protocolo.

Los clientes se caracterizan por carecer de complejidad y por permitir la visualización de las respuestas de los servidores. El objetivo de esta filosofía es hacer accesible los recursos comunes al mayor número de usuarios y para ello concentra estos en un ordenador potente y accesible desde otras terminales. Toda la complejidad y los grandes volúmenes de información residen en el servidor, siendo este el encargado de proveer la seguridad y la autorización para obtener la información. El cliente interroga al servidor el cual responde. En el caso del WWW se puede decir que un servidor Web no es más que un programa que se ejecuta en una computadora configurada para gestionar información a otras computadoras clientes. Un cliente Web es un programa que permite que un usuario solicite información a este servidor Web.

1.7 Percepción de Servidor Web.

Es un programa que se ejecuta de forma continua en un servidor el cual se mantiene a la escucha de peticiones de clientes (un navegador de Internet), luego contesta estas peticiones y retorna la información solicitada por el cliente. Además utiliza el protocolo Hypertext Transfer Protocol (HTTP) el cual está diseñado para transferir lo que llamamos hipertextos, páginas web.

1.7.1 ¿Qué denominamos Hypertext Transfer Protocol (HTTP)?

El protocolo Cliente-Servidor articula los intercambios de información entre los clientes Web y los servidores HTTP sin depender del sistema operativo que se usa para atender a las necesidades del sistema global de distribución de información World Wide Web. Desde el punto de vista de las comunicaciones,

HTTP está soportado sobre el conjunto de protocolos de la arquitectura de redes TCP/IP y funciona de la misma forma que el resto de las aplicaciones.

Cliente-Servidor: un proceso servidor escucha en un puerto de comunicaciones TCP (por defecto en este caso el 80) en espera de solicitudes de conexión de los clientes Web. Una vez que se establece la conexión, el protocolo TCP se encarga de mantener la comunicación y garantizar un intercambio de datos libre de errores.

1.7.2 ¿Qué denominamos Páginas Web?

Una página Web es un documento que contiene información sobre un tema específico la cual es almacenada en un servidor remoto.

1.8 Lenguajes de programación para el desarrollo de Aplicaciones Web

1.8.1 Hypertext Markup Language (HTML):

Lenguaje estático para el desarrollo de sitios Web, en español Lenguaje de Marcas Hipertextuales, desarrollado por el World Wide Web Consortium. Textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música. En la actualidad existen disímiles servidores web pero el más utilizado, por su estabilidad y su rendimiento, es el Servidor Web Apache. Los archivos presentan las extensiones (htm, html).

Ventajas:

- Sencillo al describir hipertexto.
- Texto presentado de forma estructurada y agradable.
- Archivos pequeños.
- Despliegue rápido.
- Fácil aprendizaje del Lenguaje.
- Admisible para todos los exploradores.

Desventajas:

- Lenguaje estático.
- La interpretación de cada navegador puede ser diferente.
- Guarda muchas etiquetas que pueden convertirse en “basura” y dificultan la corrección.
- El diseño es más lento.
- Las etiquetas son muy limitadas.

1.8.2 Active Server Pages (ASP)

Es una tecnología del lado servidor desarrollado por Microsoft para páginas Web generadas dinámicamente, fue liberado por Microsoft en 1996. Para el desarrollo de páginas Web en este lenguaje es necesario tener instalado Internet Information Server (IIS). ASP no necesita ser compilado para ejecutarse. Existen varios lenguajes que se pueden utilizar para crear páginas ASP. El más utilizado es VBScript, nativo de Microsoft. ASP se puede hacer también en Perl y Jscript (no JavaScript). El código ASP puede ser insertado junto con el código HTML. Los archivos cuentan con la extensión (asp). A partir del 2002 se le denominó a las versiones anteriores a .NET como ASP clásico.

Ventajas:

- Usa Visual Basic Script, siendo fácil para los usuarios.
- Comunicación óptima con SQL Server.
- Soporta el lenguaje JScript (Javascript de Microsoft).

Desventajas:

- Código desorganizado.
- Se necesita escribir mucho código para realizar funciones sencillas.
- Tecnología privada.
- Hospedaje de sitios Web costosos.

1.8.3 ASP.NET:

Es el sucesor de la tecnología ASP. Las páginas ASP.NET se ejecutan en el servidor y generan lenguaje de marcado (como HTML, WML o XML) que se envía a un explorador móvil o de escritorio. Fue desarrollado para resolver las limitantes de su anterior versión ASP, pensado con el fin de desarrollar Web sencillas o grandes aplicaciones. Utilizan un modelo de programación compilado y basado en eventos que mejora el rendimiento y permite la separación de la lógica de aplicación y de la interfaz de usuario. Los archivos que se generan en este tienen la extensión (aspx). Para el correcto funcionamiento de las páginas es necesario tener instalado IIS con el Framework .Net. Este es proveído por Microsoft Windows 2003, solo se necesitará instalarlo en versiones anteriores.

Ventajas:

- Completamente orientado a objetos.
- Controles de usuario y personalizados.
- División entre la capa de aplicación o diseño y el código.
- Facilita el mantenimiento de grandes aplicaciones.
- Incremento de velocidad de respuesta del servidor.
- Mayor celeridad y confianza.

Desventajas:

- Mayor consumo de recursos

1.8.4 Java Server Pages (JSP)

Desarrollado por la compañía Sun Microsystems. Esta tecnología del lado del servidor basa su funcionamiento en scripts que utilizan una variante del lenguaje java. La principal ventaja de JSP frente a otros lenguajes es

que permite integrarse con clases Java (.class) lo que permite separar en niveles las aplicaciones Web al almacenar en clases java las partes que consumen más recursos (así como las que requieren más seguridad) por lo que deja la parte encargada de formatear el documento HTML en el archivo JSP. La idea fundamental es separar la lógica del negocio de la presentación de la información. Estas ventajas competitivas de JSP influyeron en la aparición de la tecnología ASP.NET perteneciente a Microsoft que en el mercado de este tipo de tecnología forma parte de la competencia de Sun Microsystems.

1.8.5 Personal Home Page (PHP)

Es el acrónimo de Hypertext Preprocessor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma con una gran librería de funciones y mucha documentación. Es también un lenguaje interpretado y enriquecido en el HTML. Fue creado originalmente en 1994 por Rasmus Lerdorf, pero como PHP está desarrollado en política de código abierto a lo largo de su historia ha tenido muchas contribuciones de otros desarrolladores. PHP es un lenguaje de programación de estilo clásico. Esto quiere decir que es un lenguaje de programación con variables, sentencias condicionales, bucles, funciones, etc. No es un lenguaje de marcas como podría ser HTML, XML o WML. A diferencia de Java o Java Script que se ejecutan en el navegador, PHP se ejecuta en el servidor, por eso permite acceder a los recursos que tenga el servidor como por ejemplo podría ser, una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador.

PHP es la gran tendencia en el mundo de Internet. Últimamente se puede observar un ascenso imparable puesto que cada día son muchísimas más las páginas Web que lo utilizan para su funcionamiento, corre en 7 plataformas, funciona en 11 tipos de servidores, ofrece soporte sobre unas 20 Bases de Datos tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite lo cual permite la creación de Aplicaciones Web muy robusta, contiene unas 40 extensiones estables y también tiene soporte para

comunicarse con otros servicios usando protocolos tales como LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (en Windows) y muchos otros.

Ventajas:

- Muy fácil de aprender.
- Se caracteriza por ser un lenguaje muy rápido.
- Soporta en cierta medida la orientación a objeto (Clases y herencia) lo que posibilita crear los formularios para la Web. No requiere definición de tipos de variables ni manejo detallado del bajo nivel.
- Es un lenguaje multiplataforma: Linux, Windows, entre otros.
- Capacidad de expandir su potencial utilizando módulos.
- Es libre por lo que se presenta como una alternativa de fácil acceso para todos, menos costes y servidores más baratos.
- Incluye gran cantidad de funciones.
- No requiere definición de tipos de variables ni manejo detallado del bajo nivel.
- Su librería estándar es realmente amplia, lo que permite reducir los llamados "costes ocultos", uno de los principales defectos de ASP.
- PHP tiene una de las comunidades más grandes en Internet. Esto permite encontrar fácilmente ayuda, documentación, artículos, noticias, y otros recursos.

Desventajas:

- Se necesita instalar un servidor Web.
- Todo el trabajo lo realiza el servidor y no delega al cliente. Por tanto puede ser más ineficiente a medida que las solicitudes aumenten de número.

- La programación orientada a objetos es aún muy deficiente para aplicaciones grandes.
- Dificulta la organización por capas de la aplicación.

1.9 Herramientas para el desarrollo de Aplicaciones Web

Case Studio: es una herramienta profesional con la que podrás diseñar tus propias bases de datos, lo que facilita herramientas para la creación de diagramas de relación, modelado de datos y gestión de estructuras (Es compatible con ambos SGBD, MySQL y PostgreSQL).

1.9.1 Sistemas gestores de bases de datos (SGBD) :

Consiste en un conjunto de programas, procedimientos y lenguajes que nos proporcionan las herramientas necesarias para trabajar con una base de datos, incorporar una serie de funciones que permita definir los registros, sus campos, sus relaciones, insertar, suprimir, modificar y consultar los datos. Actualmente existen muchos sistemas gestores de bases de datos.

1.9.2 Microsoft SQL Server:

Fabricado por Microsoft este SGBD relacional es capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea. Entre sus características se pueden destacar: soporte de transacciones, escalabilidad, estabilidad, seguridad, soporta procedimientos almacenados, incluye un potente entorno gráfico de administración, permite trabajar en modo cliente-servidor donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información. Permite también administrar información de otros servidores de datos. Es un software de tipo propietario en cuanto a su licencia.

1.9.3 PostgreSQL y MySQL:

Los Sistemas de gestión de base de datos son un tipo de software muy específico dedicado a servir de interfaz entre la base de datos, el usuario y las

aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de manipulación de datos y de consulta.

1.9.4 MySQL:

Es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. Software libre que se desarrolla en un esquema de licenciamiento dual. Perteneció a Sun Microsystems desde enero de 2008. Por un lado lo ofrece bajo la GNU GPL pero, empresas que quieran incorporarlo en productos privativos pueden comprar a la empresa una licencia que les permita ese uso. Está desarrollado en su mayor parte en ANSIC. Al contrario de proyectos como el Apache donde el software es desarrollado por una comunidad pública y el copyright del código está en poder del autor individual. MySQL está poseído y patrocinado por una empresa privada que posee el copyright de la mayor parte del código. Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. Fundado por David Axmark, Allan Larsson, y Michael Widenius. Tiene la ventaja de ser multiplataforma, es decir, que funciona sobre múltiples plataformas incluyendo AIX, BSD, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell Netware, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Vista y otras versiones de Windows. También existe MySQL para OpenVMS.

Posee múltiples motores de almacenamiento, por lo que le permite al usuario, escoger la más adecuada para cada tabla de la base de datos, agrupación de transacciones y reúne múltiples transacciones de varias conexiones para incrementar el número de transacciones por segundo. Esto ofrece un mayor rendimiento, velocidad, utilidades de administración; aunque se cuelgue, no suele perder información ni corromper los datos, mejor integración con PHP, no hay límites en el tamaño de los registros, control de acceso.

1.9.5 PostgreSQL:

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales que ha sido desarrollado de varias formas desde 1977 (ORDBMS por sus siglas en inglés); esto significa que las tablas se manejan como objetos y las tuplas son instancias de ese objeto. Comenzó como un proyecto denominado Ingres en la Universidad Berkeley de California. Ingres fue más tarde desarrollado comercialmente por la Relational Technologies/Ingres Corporation. En 1986 otro equipo dirigido por Michael Stonebraker de Berkeley continuó el desarrollo del código de Ingres para crear un sistema de bases de datos objeto-relacionales llamado Postgres. En 1996, debido a un nuevo esfuerzo de código abierto y a la incrementada funcionalidad del software, Postgres fue renombrado a PostgreSQL, tras un breve periplo como Postgres95. Se pueden crear nuevos tipos de datos, hacer herencias entre objetos. PostgreSQL es una herramienta muy potente para los desarrolladores de sistemas de bases de datos, tiene todo aquello de lo que carece MySQL. Al igual que otros proyectos, PostgreSQL no está controlado por una compañía específica, sino que cuenta con una comunidad global de desarrolladores y compañías para su desarrollo, esto posibilita que cada vez aumente sus funcionalidades. El proyecto PostgreSQL sigue actualmente un activo proceso de desarrollo a nivel mundial gracias a un equipo de desarrolladores y contribuidores de código abierto. Está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo. Posee muchas características que tradicionalmente sólo se podían ver en productos comerciales de alto calibre

Características principales:

DBMS Objeto-Relacional DBMS Objeto-Relacional: PostgreSQL aproxima los datos a un modelo objeto-relacional y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son

consultas SQL declarativas, control de concurrencia multi-versión, soporte multi-usuario, transactions, optimización de consultas, herencia, y arrays.

Altamente extensible: PostgreSQL soporta operadores, funciones métodos de acceso y tipos de datos definidos por el usuario. .

SQL comprensible: PostgreSQL soporta la especificación SQL99 e incluye características avanzadas tales como las uniones (joins) SQL92.

Integridad referencial: PostgreSQL soporta integridad referencial la cual es utilizada para garantizar la validez de los datos de la base de datos.

API flexible: La flexibilidad del API de PostgreSQL ha permitido a los vendedores proporcionar soporte al desarrollo fácilmente para el RDBMS PostgreSQL. Estas interfaces incluyen Object Pascal, Python, Perl, PHP, ODBC, Java/JDBC, Ruby, TCL, C/C++, y Pike.

Lenguajes procedurales: PostgreSQL tiene soporte para lenguajes procedurales internos e incluye un lenguaje nativo denominado PL/pgSQL. Este lenguaje es comparable al lenguaje procedural de Oracle, PL/SQL. Otra ventaja de PostgreSQL es su habilidad para usar Perl, Python, o TCL como lenguaje procedural embebido.

MVCC (Multi-Version Concurrency): MVCC, o Control de Concurrencia Multi-Versión es la tecnología que PostgreSQL usa para evitar bloqueos innecesarios.

Cliente/Servidor: PostgreSQL usa una arquitectura proceso-por-usuario cliente/servidor. Esta es similar al método del Apache 1.3.x para manejar procesos. Hay un proceso maestro que se ramifica para proporcionar conexiones adicionales para cada cliente que intente conectar a PostgreSQL. .

WAL (Write Ahead Logging): Incrementa la dependencia de la base de datos al registro de cambios antes de que estos sean escritos. Esto garantiza que en el hipotético caso de que la base de datos se caiga, existirá un

registro de las transacciones a partir del cual podremos restaurar la base de datos.

1.10 Content Management System (CMS)

¿Qué es un CMS?

Los sistemas de gestión de contenido son un software que se utilizan principalmente para facilitar la gestión de Webs, ya sea en la internet o en una intranet, y por eso también son conocidos como gestores de contenidos Web (Web Content management o WCM). Hay que tener en cuenta sin embargo que las aplicaciones de los CMS no se limitan solo a las Web.

Systems (Sistema): Se trata generalmente de un conjunto de herramientas apoyados habitualmente por una base de datos y que consiste en una serie de programas en un servidor Web, y, opcionalmente una serie de programas cliente que te permitan acceder fácilmente a esos programas en el servidor.

Content Management (Gestión de contenidos): Desde el punto de vista del usuario del sistema se trata de gestionar de forma uniforme, con un diseño centrado en el usuario, y que permita llevar a cabo fácilmente las tareas para las que ha sido diseñado.

Las principales funcionalidades de un CMS radican en la **creación de contenidos** y la **presentación de los mismos** ya que provee una serie de herramientas para que " publicar contenido " sea tan fácil como rellenar un formulario y exista una única fuente para todos ellos. Al atender a la presentación de los contenidos posibilita la publicación en múltiples formatos a partir de una sola fuente y añade metadatos para facilitar la navegación en múltiples facetas (Temporal, por categoría o autor).

Por lo tanto, un CMS provee la herramienta necesaria para gestionar todo el ciclo de vida de los contenidos: ***Creación, gestión, publicación, presentación, mantenimiento y actualización.***

1.10.1 Zope

Zope es un servidor de aplicaciones Web de código abierto escrito en el lenguaje de programación Python. Para las funciones de edición de contenidos, así como personalizaciones básicas es usado mediante un navegador web. La programación avanzada así como el desarrollo de nuevas funcionalidades requiere la edición de componentes en «file system».

Un sitio Web de Zope está compuesto de objetos en lugar de archivos como es usual con la mayoría de los otros sistemas de servidores Web. Las ventajas de usar objetos en lugar de archivos son:

- ✓ Combinan el comportamiento y los datos en una forma más natural que los archivos de texto plano.
- ✓ Alientan el uso de componentes estándares que se ocupan de una parte particular de las que forman una aplicación Web al permitir flexibilidad y buena descomposición.
- ✓ Posibilitan procesos automáticos de gestión de información.

Lo más característico de Zope es su base de datos orientada a objetos, llamada ZODB o Zope Object Database. Esta base de datos almacena objetos ordenados en un sistema similar a un sistema de ficheros pero cada objeto tiene propiedades, métodos u otros objetos. Esta aproximación es muy diferente de las base de datos relacionales habituales. Sin embargo, Zope dispone de múltiples conectores para las diferentes bases de datos relacionales y ofrece sistemas básicos de conexión y consulta abstrayéndolos como objetos.

Actualmente existen dos ramas principales, zope2 y zope3. Este último es una reimplementación del servidor zope donde se ha tratado de volcar toda la experiencia adquirida en zope2. Zope3 no trae compatibilidad hacia atrás por lo que los componentes hechos para zope2 no funcionan. Aún se está en un proceso de adaptación hacia este nuevo zope para lo cual está usando un componente llamado five, con el cual desde zope2 pueden tener la facilidad de

zope3. Constituyen ejemplos de sitios que utilizan Zope son Launchpad y schooltool.

1.10.2 Joomla

El proyecto Joomla nace en agosto de 2005 como el resultado de la división entre la compañía australiana Miró (fundadora del software libre llamado Mambo) y sus desarrolladores, a raíz de unas cláusulas que violaban los valores principales del código abierto. Joomla es un CMS (Content Management System), es decir, un sistema de administración de contenidos web. Su código es abierto y está escrito en PHP, usa bases de datos MySQL y se distribuye bajo la licencia GPL. En palabras menos técnicas, es un software libre que no paga licenciamiento y se basa en herramientas similares que no generan costos de licencias. De todas formas, hay que decir que Joomla no tiene costo (de licencia), se puede bajar y usar sin tener que pagar nada. (CARVAJAL, 2008)

Pero lo más importante es que uno es libre de acceder al código, modificarlo, mejorarlo o adaptarlo a sus necesidades. Dentro de las principales ventajas de usar Joomla como CMS, encontramos:

- ✓ La Organización del sitio Web
- ✓ Publicación de Contenidos
- ✓ Escalabilidad e implementación de nuevas funcionalidades
- ✓ Administración de usuarios
- ✓ Diseño y aspecto estético del sitio
- ✓ Navegación y menú
- ✓ Administrador de Imágenes
- ✓ Disposición de módulos modificable
- ✓ Encuestas y Publicidad
- ✓ Estadísticas de visitas y Automatización en la publicación:
- ✓ Archivo e historial
- ✓ Formatos de lectura y Envío por E-mail, etc.

1.10.3 Drupal

"Drupal" es el equivalente fonético en inglés a la palabra neerlandesa druppel que significa 'gota'. Escrito originalmente por Dries Buytaert y es el software usado para impulsar, por ejemplo, a los sitios web Debian Planet, pread Firefox y Kernel Trap. Drupal es un sistema de gestión de contenido 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. Drupal es un sistema dinámico: en lugar de almacenar sus contenidos en archivos estáticos en el sistema de ficheros del servidor de forma fija, el contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno Web. Es un programa libre, con licencia GNU/GPL, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Se destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la Web y un énfasis especial en la usabilidad y consistencia de todo el sistema. El diseño de Drupal es especialmente idóneo para construir y gestionar comunidades en Internet. No obstante, su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales disponibles hace que sea adecuado para realizar muchos tipos diferentes de sitio Web.

Contenido.

Características de Drupal

Generales

Ayuda on-line: Un robusto sistema de ayuda online y páginas de ayuda para los módulos del 'núcleo', tanto para usuarios como para administradores.

Búsqueda: Todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.

Código abierto: El código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario que otros sistemas de 'blogs' o

de gestión de contenido propietarios, es posible extender o adaptar Drupal según las necesidades.

Módulos: La comunidad de Drupal ha contribuido muchos módulos que proporcionan funcionalidades como 'página de categorías', autenticación mediante jabber, mensajes privados, bookmarks, etc.

Personalización: Un robusto entorno de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo las preferencias definidas por el usuario.

URLs amigables Drupal usa el mod_rewrite de Apache para crear URLs que son manejables por los usuarios y los motores de búsqueda.

Gestión de usuarios

Autenticación de usuarios: Los usuarios se pueden registrar e iniciar sesión de forma local o utilizando un sistema de autenticación externo como Jabber, Blogger, LiveJournal o otro sitio Drupal. Para su uso en una intranet, Drupal se puede integrar con un servidor LDAP.

Permisos basados en roles: Los administradores de Drupal no tienen que establecer permisos para cada usuario. En lugar de eso, pueden asignar permisos a un 'rol' y agrupar los usuarios por roles.

Gestión de contenido

Control de versiones: El sistema de control de versiones de Drupal permite seguir y auditar totalmente las sucesivas actualizaciones del contenido: qué se ha cambiado, la hora y la fecha, quién lo ha cambiado y más. También permite mantener comentarios sobre los sucesivos cambios o deshacer los cambios recuperando una versión anterior.

Enlaces permanentes (Permalinks): Todo el contenido creado en Drupal tiene un enlace permanente asociado a él para que pueda ser enlazado externamente sin temor de que el enlace falle en el futuro.

Objetos de Contenido (Nodos): El contenido creado en Drupal es, funcionalmente, un objeto (Nodo). Esto permite un tratamiento uniforme de la información como una misma cola de moderación para envíos de diferentes tipos, promocionar cualquiera de estos objetos a la página principal o permitir comentarios -o no- sobre cada objeto.

Plantillas (Templates): El sistema de temas de Drupal separa el contenido de la presentación al permitir controlar o cambiar fácilmente el aspecto del sitio web. Se pueden crear plantillas con HTML y/o con PHP.

Sindicación del contenido Drupal: Exporta el contenido en formato RDF/RSS para ser utilizado por otros sitios web. Esto permite que cualquiera con un 'Agregador de Noticias', tal como *NetNewsWire* o *Radio UserLand* visualice el contenido publicado en la web desde el escritorio.

Blogging

Agregador de noticias Drupal incluye un potente Agregador de Noticias para leer y publicar enlaces a noticias de otros sitios web. Incorpora un sistema de cache en la base de datos, con temporización configurable.

Soporte de Blogger API La API de Blogger permite que un sitio Drupal sea actualizado al utilizar diversas herramientas que pueden ser 'herramientas web' o 'herramientas de escritorio' que proporcionen un entorno de edición más manejable.

Plataforma

Independencia de la base de datos: Aunque la mayor parte de las instalaciones de Drupal utilizan MySQL existen otras opciones. Drupal incorpora una 'capa de abstracción de base de datos' que actualmente está implementada y

mantenida para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos.

Multiplataforma: Drupal ha sido diseñada desde el principio para ser multi-plataforma. Puede funcionar con Apache o Microsoft IIS como servidor web y en sistemas como Linux, BSD, Solaris, Windows y Mac OS X. Por otro lado, al estar implementado en PHP, es totalmente portable.

Múltiples idiomas y Localización: Drupal está pensado para una audiencia internacional y proporciona opciones para crear un portal multilingüe. Todo el texto puede ser fácilmente traducido al utilizar una interfaz web y al importar traducciones existentes o integrar otras herramientas de traducción como *GNU ettext*

Administración y Análisis

Administración vía Web: La administración y configuración del sistema se puede realizar enteramente con un navegador y no precisa de ningún software adicional.

Análisis, Seguimiento y Estadísticas: Drupal puede mostrar en las páginas web de administración informes sobre *referrals* (enlaces entrantes), popularidad del contenido o de cómo los usuarios navegan por el sitio.

Registros e Informes: Toda la actividad y los sucesos del sistema son capturados en un 'registro de eventos' que puede ser visualizado por un administrador.

Rendimiento y escalabilidad

Control de congestión: Drupal incorpora un mecanismo de control de congestión que permite habilitar y deshabilitar determinados módulos o bloques al depender de la carga del servidor. Este mecanismo es totalmente configurable y ajustable.

Sistema de Cache: El mecanismo de cache elimina consultas a la base de datos al incrementar el rendimiento y reduciendo la carga del servidor.

A pesar de que empezó como un pequeño BBS, Drupal ha llegado a ser mucho más que sólo un portal de noticias gracias a su arquitectura flexible. Se compone de una infraestructura base y un conjunto de módulos que ofrecen un amplio conjunto de funciones incluyendo sistemas de comercio electrónico, galerías de fotos, administración de listas de correo electrónico e integración de CVS. Es posible añadir módulos de terceros para modificar el comportamiento de Drupal u ofrecer nuevas funciones.

Su usabilidad se puede ver en disímiles lugares ya sea, en intranets de compañías, enseñanza en línea, comunidades de arte y administración de proyectos. Muchos piensan que la relevancia de Drupal en las comunidades de usuarios es lo que lo hace destacarse de la competencia.

Es un gestor de contenidos multipropósito que puede usarse para aplicaciones como:

Portales Comunitarios

Foros de discusión

Sitios Web corporativos

Aplicaciones de Intranet

Sitios personales o blogs

Aplicaciones de comercio electrónico

Directorio de recursos

Sitios de redes sociales

Módulos

En Drupal se pueden ampliar sus funcionalidades mediante extensiones llamadas módulos programados por su comunidad de usuarios. Entre los más

importantes que no están incluidos en la distribución oficial se pueden destacar:

Drupal

Views

Content Construction Kit (CCK)

Token

Pathauto

File Field

Administration de menú

ImageField

ImageAPI

ImageCache

Distribuciones de Drupal

Hay varias distribuciones de Drupal personalizadas. Muchas de ellas están previamente pre empaquetadas con módulos de software de terceros, pero algunas contienen modificaciones del núcleo.

Referencias

1. ↑ Sobre Drupal en Drupal Hispano
2. ↑ Módulos libres contribuidos por la comunidad
3. ↑ Utilización de los módulos de Drupal

Enlaces externos

*Colabora en Commons. Wikimedia Commons alberga contenido multimedia sobre Drupal. Commons

* Sitio Web de Drupal

* Sitio Web de Drupal Hispano

Drupal vs Joomla

Atendiendo a la construcción del Sitio: La flexibilidad y potencia de Drupal es significativamente más potente y mucho más flexible. Las propiedades que ofrece (Views, CCK, Panels, etc.) le dan al mismo una enorme ventaja sobre Joomla porque simplemente no puede tener esa flexibilidad con Joomla. Los

desarrolladores de Joomla cuando lo diseñaron no tuvieron en cuenta la flexibilidad. Puedes construir algo muy rápido con Joomla pero toparán con un muro antes o temprano.

Resumen:

Utiliza Joomla si quieres tener un sitio bonito de forma rápida y puedes afrontar tener un sistema más lento, una categorización de contenido rígida y unas opciones de configuración y de diseño limitadas.

Utiliza Drupal si quieres alto rendimiento, escalabilidad, buen gestor de contenido y una flexibilidad de diseño significativa.

Al tener en cuenta las anteriores características y al analizar que se utilizaran las tecnologías cliente/servidor sobre plataforma Web con el objetivo de llevar a cabo el correcto desarrollo del proyecto Portal-Intranet v2.0, se escogió como lenguaje de programación Web a PHP dado a las propiedades del mismo de ser portable, eficiente y de libre acceso, y como SGBD a PostgreSQL por sus potencialidades. Además de que la política que lleva a cabo hoy nuestra universidad respalda a estas herramientas y lenguajes de programación por su condición de software libre con el fin de poder darle reparación y ofrecer nuevas actualizaciones futuras.

1.11 Aspectos generales sobre las Metodologías Usadas

1.11.1 Metodologías tradicionales:

Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo cierta resistencia a los cambios impuestos externamente. Proceso mucho más controlado, con numerosas políticas/normas, existe un contrato prefijado, el cliente interactúa con el equipo de desarrollo mediante reuniones, grupos grandes y posiblemente distribuidos, más artefactos, más roles, la arquitectura del software es esencial y se expresa mediante modelos. Constituye un ejemplo de esta metodología:

RUP: El Proceso Unificado de Rational (RUP) publicado en 1998. Tiene sus raíces desde el 1967 con la metodología de Ericsson Approach la cual constituye la pionera en la historia de desarrollo basados en componentes; es en esta que se utiliza el concepto de caso de uso. RUP por su parte es también una metodología de desarrollo de Software, la cual basa su fundamento en que todo proceso de desarrollo de software debe pasar por cuatro fases.

Las principales características de este proceso unificado son:

Dirigido por casos de uso, centrado en la arquitectura, es iterativo e incremental. Y además, este utiliza un solo lenguaje de modelación (UML).

UML (*Unified Model Language*): Es un lenguaje de modelado de propósito general que pueden usar todos los modeladores. No tiene propietario y está basado en el común acuerdo de una gran parte de la comunidad informática. Constituye una técnica utilizada para el modelado de diferentes tipos de sistemas para describir una información bien detallada sobre el funcionamiento de los mismos mediante símbolos estándares.

Además está compuesto por una amplia gama de diagramas o artefactos que nos permiten graficar los procesos así como comprender el funcionamiento de estos.

1.11.2 Metodologías Ágiles:

Manifiesto para el Desarrollo de Software Ágil, una declaración de los principios y valores comunes de los procesos ágiles. Hay también un deseo de colaborar más en el futuro para animar más, tanto a tecnólogos como a gente de negocios, para usar y requerir acercamientos ágiles al desarrollo de software. Hay un artículo en una revista de desarrollo de software que es un comentario y una explicación del manifiesto.

Se define metodologías ágiles a la prioridad de satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor: a dar la bienvenida a los cambios, es decir, que se capturan los cambios para que el

cliente tenga una ventaja competitiva, a la entrega frecuentemente de software que funcionen desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas. La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto. Se construye el proyecto en torno a individuos motivados. Se le brinda el entorno y todo el apoyo que necesiten sus desarrolladores así como el voto de confianza a ellos para conseguir finalizar el trabajo. El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo. El software que funciona es la medida principal de progreso. Los procesos ágiles contribuyen a un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante. En esta se consideran conceptos tales como:

- ✓ “ La atención continua a la calidad técnica y al buen diseño mejora la agilidad.”
- ✓ “ La simplicidad es esencial.”
- ✓ “ Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.”

Las metodologías ágiles se basan en heurísticas provenientes de prácticas de producción de código, especialmente preparados para cambios durante el proyecto, impuestas internamente (por el equipo), estos procesos son menos controlados, con pocos principios, en ellos no existe un contrato tradicional o al menos es bastante flexible, El cliente es parte del equipo de desarrollo, grupos pequeños (<10 integrantes) y trabajando en el mismo sitio; además, estos cuentan con la ventaja de que a partir de ellos se generan pocos artefactos, pocos roles, menos énfasis en la arquitectura del software.

Constituyen ejemplos de metodologías ágiles:

Crystal Methodologies:

Conjunto de metodologías para el desarrollo de software caracterizadas por estar centradas en las personas que componen el equipo y la reducción al

máximo del número de artefactos producidos. Desarrolladas por Alistair Cockburn. El desarrollo de software se considera un juego cooperativo de invención y comunicación.

- ✓ Limitado por los recursos a utilizar.
- ✓ El equipo de desarrollo es un factor clave por lo que se deben invertir esfuerzos en mejorar sus habilidades y destrezas, así como tener políticas de trabajo en equipo definidas.

Dynamic Systems Development Method (DSDM):

DSDM empezó en Gran Bretaña en 1994 como un consorcio de compañías del Reino Unido que querían construir sobre RAD [R. del T. Desarrollo Rápido de Aplicaciones] y desarrollo iterativo. Empezó con 17 fundadores, ahora tiene más de mil miembros y ha crecido fuera de sus raíces británicas. Siendo desarrollado por un consorcio tiene un sabor diferente a muchos de los otros métodos ágiles. Tiene una organización de tiempo completo que lo apoya con manuales, cursos de entrenamiento, programas de certificación y demás. También lleva una etiqueta de precio lo que ha limitado la investigación sobre su metodología.

Adaptive Software Development (ASD):

Promovido por Jim Highsmith. Sus principales características son:

- ✓ Iterativo.
- ✓ Orientado a los componentes software más que a las tareas y tolerante a los cambios.
- ✓ El ciclo de vida que propone tiene tres fases esenciales:
 - Especulación.
 - Colaboración y aprendizaje.

En la primera de ellas se inicia el proyecto y se planifican las características del software; en la segunda, se desarrolla las características y finalmente en la tercera, se revisa su calidad y se entrega al cliente. La revisión de los componentes sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo.

Feature-Driven Development (FDD):

Define un proceso iterativo que consta de 5 pasos. Las iteraciones son cortas (hasta 2 semanas). Se centra en las fases de diseño e implementación del sistema partiendo de una lista de características que debe reunir el software. Sus impulsores son Jeff De Luca y Peter Coad.

Lean Development (LD):

Definida por Bob Charretes en los años 80 y utilizada en numerosos proyectos de telecomunicaciones en Europa. Los cambios se consideran riesgos, pero si se manejan adecuadamente se pueden convertir en oportunidades que mejoren la productividad del cliente. Su principal característica es introducir un mecanismo para implementar dichos cambios.

PROGRAMACIÓN EXTREMA (EXTREME PROGRAMMING, XP)

XP:

Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software por lo que promueve el trabajo en equipo, y se preocupa por el aprendizaje de los desarrolladores al propiciar un buen clima de trabajo. Se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. Se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes y donde existe un alto riesgo técnico. Esta metodología se basa en:

- Pruebas Unitarias: se basa en las pruebas realizadas a los principales procesos de tal manera que al adelantar en algo hacia el futuro se pueden hacer pruebas de las fallas que pudieran ocurrir. Es como si se adelantara a obtener los posibles errores.
- Refabricación: se basa en la reutilización de código para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.

- Programación en pares: Consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto:

Mientras uno conduce, el otro consulta el mapa.

¿Qué propone XP?

- Empieza en pequeño y añade funcionalidad con retroalimentación continua.
- El manejo del cambio se convierte en parte sustantiva del proceso.
- El costo del cambio no depende de la fase o etapa.
- No introduce funcionalidades antes que sean necesarias.

SCRUM:

Esta metodología se basa en una filosofía del desarrollo ágil, creado por Hirotaka Takeuchi e Ikujiro Nonaka en el año 1986 para el desarrollo de software. SCRUM aunque puede ser usado para otro tipo de proyectos y tiene demostrada efectividad en otras áreas, generalmente es funcional solo para desarrollos de software porque para eso fue diseñado. La idea es desarrollar aplicaciones mucho más rápido y eficazmente.

Pero entonces la pregunta es:

¿Qué significa SCRUM?

Pues la palabra se usa en rugby y prácticamente significa melé. Si ustedes han visto un juego de rugby cuando va a iniciar una jugada se aglomeran los equipos empujándose con tal de hacerse de la pelota para recuperarla y pasársela a otro que espera atrás. Pues eso es SCRUM no son siglas ni nada por el estilo.

Lo interesante de la melé es que si un jugador del equipo se cae se viene abajo todo el equipo, así que deben de estar coordinados para que todos vayan a la misma velocidad.

Esta metodología está basada entre muchas bajo estas premisas:

- a) Los individuos por encima de los procesos y herramientas.
- b) En entregar soluciones por encima de reportes de seguimiento.
- c) A dar respuesta a los cambios en lugar de ceñirse a seguir un plan.

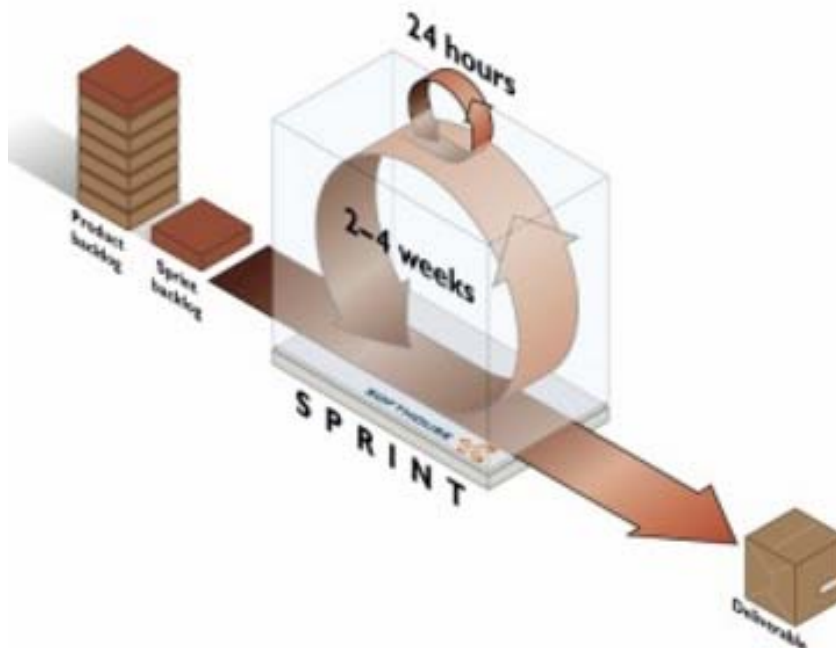
La literatura de SCRUM se enfoca principalmente en la planeación iterativa y el seguimiento del proceso. Es muy cercana a las otras metodologías ágiles en muchos aspectos y debe funcionar bien con las prácticas de código de la XP.

Al no existir una metodología universal que sea capaz de dar solución fiable y eficiente a cualquier proyecto de desarrollo de software es el porqué se debe tener en cuenta el análisis exhaustivo de todas las existentes y ver cuáles son las ventajas que nos ofrecen cada una de ellas para que se pueda adaptar al contexto del proyecto actual, al atender a los recursos materiales y capital humano, tiempo en que se enmarca el desarrollo del mismo, características generales y específicas de nuestro sistema a desarrollar. Generalmente las metodologías tradicionales han tratado de tomar la delantera al abordar la mayor cantidad de situaciones de contexto en cuanto a desarrollo de software se refiere por lo que exige un mayor esfuerzo para adaptarlas, sobre todo en proyectos pequeños y requisitos funcionales y no funcionales sumamente cambiantes, donde las metodologías ágiles son las que nos proporcionan una mejor solución para este tipo de proyectos que contienen estas características. Un factor a tener en cuenta en cualquier trabajo es la complejidad la cual brinda una cualidad muy importante y estas son metodologías sencillas tanto por su aprendizaje como por su aplicación lo cual trae consigo el ahorro de los costes.

Se considera que por todo lo anteriormente expuesto y por las características y necesidades para el desarrollo de este tipo de proyecto es conveniente la

utilización de SCRUM y XP como metodologías por lo que se ejemplifican las principales.

- ✓ Entregas periódicas en ciclos cortos, es decir, dar pasitos pequeños pero seguros. Los **Sprint** de SCRUM se ajustan perfectamente a esta idea. Se necesitaba algo para controlar mejor el estado del desarrollo en un momento determinado, algo que permita llegar a la oficina por la mañana y saber si vas reservando mesa en un restaurante para celebrarlo o vas preparando las maletas para poner tierra de por medio antes de la tempestad. En este caso las **reuniones de SCRUM** diarias y las **gráficas de burndown** son herramientas simples pero que pueden resultar muy efectivas.
- ✓ **Pruebas unitarias**, integración continua, definir una guía de estilo, refactorizaciones, muchas de las practicas de XP nos parecían un buen modo de mejorar la calidad de nuestros desarrollos.
- ✓ **Movilidad** de los miembros del equipo y **propiedad colectiva del código**. Tanto SCRUM como XP promueven la idea de equipos con miembros que se puedan hacer cargo de cualquier tarea en cualquier parte del proyecto frente a la idea de especialistas. Muy importante para evitar los problemas de dependencia y conseguir que todos los miembros del equipo tengan una visión general del proyecto y no se queden en sus respectivas islas.



Conclusiones

Este capítulo servirá para dar ayuda a todo aquel interesado en los aspectos importantes de esta aplicación así como en la bibliografía consultada para su buen desarrollo. Por otra parte hace referencia a la importancia que tienen hoy en día las Tecnologías de Información y las Comunicaciones (TIC), para el buen flujo de la información mediante Redes LAN y hacia las técnicas que se le aplican actualmente a los portales Web, además del manejo y acceso confiable a las bases de datos por medio del Portal-Intranet. Se plantea una descripción de una solución propuesta para la actualización de la información dentro del Portal-Intranet.

Capítulo 2 Planificación y Diseño.

Introducción

En dicho capítulo veremos las cuestiones referentes a los parámetros principales del Portal-Intranet los cuales se ven en la medida en que pueda permitir el portal la gestión y visualización de la información a la que se accederá por medio de la intranet del ISMM debido a que esta es la naturaleza del sitio y que además constituye el punto de partida para la implementación de la interfaz de la aplicación. Además, se verán las historias de usuarios realizadas por el cliente y a su vez la planificación de entrega para su implementación.

2.1 Planificación

La planificación constituye la primera fase de la metodología XP. En esta se definen todos los procesos a automatizar, así como el tiempo que se determinará para realizar cada una de estas aplicaciones. Por tal motivo, es la encargada de generar toda la documentación correspondiente a la concepción inicial del sistema, su definición, además incluye algunos vinculados a la primera fase de los procesos de ingeniería de Software. Por otra parte involucra documentos relacionados con la estimación inicial de esfuerzos y la valoración de los riesgos. Las planillas surgen a partir de cada actividad en específico y constituyen un baluarte para la documentación del Software, estas cobran suma importancia ya que en ellas se detalla la razón de su existencia y la descripción de los objetivos a tener en cuenta.

2.1.1 Concepción del sistema

Primer documento generado en la fase de Planificación-Definición. Este recoge los detalles relacionados con las actividades, entrevistas y/o encuestas al cliente. Este documento, además de reflejar la visión general del producto a implementar, también recoge los diferentes roles que intervendrán en el desarrollo del software. Se documenta el tipo de proyecto al que pertenece así como la especificación del Polo Productivo y su clasificación. Se recoge

además cuáles herramientas serán utilizadas para el desarrollo de la aplicación, el alcance que va a tener, una descripción de los involucrados en el negocio, cuáles son los motivos de la necesidad del desarrollo del software y la propuesta de solución. Esta es la plantilla principal dentro de la documentación por ser la guía para los demás documentos que se generan durante el ciclo de desarrollo de software. Para el caso del presente Portal se realizó una encuesta a partir del sistema de encuesta lo que permitió que esta información fuese accesible a toda la comunidad universitaria y a su vez confiable lo cual dio la posibilidad de saber qué se tiene y qué hace falta para el desarrollo del Portal-Intranet v2.0; estos serán aún mas detallados con posterioridad en las demás plantillas que se generan de las metodologías ágiles en cuestión. A partir de los resultados arrojados por la encuesta realizada se determinó que el producto a desarrollar debe estar acorde con las necesidades imperantes para los usuarios finales, para ello se realizó un análisis de todos los sistemas que se deben integrar al portal con el objetivo de brindarle más funcionalidades al mismo: los colores del sitio deben estar en correspondencia con las características del territorio donde se encuentra enmarcada la universidad, la forma de visualizar el contenido debe estar de manera que el acceso a ellos sea rápido y comprensible por lo que se llevó cabo una correcta arquitectura de la información (AI) para que todo el contenido estuviese al alcance en tan solo tres niveles de navegación.

Roles: Analista y Cliente.

2.1.2 Modelo de Historias de usuario del negocio

La plantilla del Modelo Historias de usuario del negocio es un artefacto que se genera del Juego de la planificación, luego de estar definida la concepción del sistema, se hace mucho más fácil comprender el negocio.

Se definen las características específicas del negocio, así como la forma en que interactúa el sistema con el cliente y viceversa. El Modelo de negocio cuando se trabaja con metodologías ágiles, es diferente al ya conocido en el proceso unificado, ya que en este caso se trabaja con historias de usuarios en

vez de con casos de uso; pero independientemente de los cambios técnicos que puedan existir, el negocio se modela igual en cualquier metodología.

Roles: Analista

2.1.3 Lista de Reserva del Producto (LRP)

La plantilla de Lista de Reserva del Producto es el primer artefacto generado en la etapa de captura de requisitos. Está conformada por una lista priorizada que define el trabajo que se va a realizar en el proyecto. Cuando un proyecto comienza es muy difícil tener claro todos los requerimientos sobre el producto. Sin embargo, suelen surgir los más importantes que casi siempre son más que suficientes para una iteración.

Esta lista puede crecer y modificarse a medida que se obtienen más conocimientos acerca del producto y del cliente con la restricción de que sólo puede cambiarse entre iteraciones. El objetivo es asegurar que el producto definido al terminar la lista es el más correcto, útil y competitivo posible y para esto debe acompañar los cambios en el entorno y el producto. La lista puede estar conformada por requerimientos técnicos y del negocio, funciones, errores a reparar, defectos, mejoras y actualizaciones tecnológicas requeridas.

Roles: Analista y Cliente

2.1.4 Historias de usuario

Las historias de usuario son la técnica utilizada en XP para especificar los requisitos del software lo que equivaldría a los casos de uso en el proceso unificado. Las mismas son escritas por los clientes como las tareas que el sistema debe hacer y su construcción depende principalmente de la habilidad que tenga para definir las. Estas se encargan de dirigir la construcción de las pruebas de aceptación, elemento clave en XP (deben generarse una o más pruebas para verificar que la historia ha sido correctamente implementada) y son utilizadas para estimar tiempos de desarrollo. En este sentido, sólo proveen detalles suficientes para hacer una estimación razonable del tiempo que llevará implementarlas. En el momento de implementar una historia de

usuario se debe detallar a través de la comunicación con el cliente. En esta plantilla los campos de puntos estimados y puntos reales se llenan, luego del desarrollo de la actividad de estimación de esfuerzo, donde se decide qué tiempo se le dedicará a cada historia de usuario.

Las historias de usuario proporcionan ventajas, tales como:

- ✓ Están escritas en lenguaje del cliente, por lo que es muy fácil su comprensión.
- ✓ Especifican cada uno de los requisitos del sistema sin necesidad de documentaciones extensas.
- ✓ Reflejan todas las características del sistema.
- ✓ Si se definen correctamente, guían el proceso de implementación.

Roles: Analista y Cliente

2.1.5 Lista de riesgos

La plantilla de Lista de riesgos es el documento que se genera de la actividad de Valoración de riesgos. En ella quedan definidos los posibles riesgos que actuarán sobre el proceso de desarrollo de software, así como la estrategia trazada, además de un plan de contingencia que describe qué curso seguirán las acciones si el riesgo se materializa.

Roles: Gerente

2.1.6 Modelo de diseño

La plantilla del Modelo de diseño es el documento que se genera del diseño con las metáforas donde se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto. En XP no se enfatiza la definición temprana de una arquitectura estable para el sistema. Dicha arquitectura se asume de forma evolutiva y los posibles inconvenientes que se generarían por no contar con ella explícitamente en el comienzo del proyecto se solventan con la existencia de una metáfora.

Roles: Diseñador

Capítulo 3: Desarrollo del Sistema

Introducción

En este capítulo se construye la solución propuesta de forma iterativa, tal y como indica la metodología XP. Una iteración es una entrega del proyecto al cliente que está incompleta, pero tiene implementadas algunas funcionalidades. La siguiente iteración es otra entrega con alguna funcionalidad más y así sucesivamente hasta llegar a la iteración final en la que se entrega el software acabado. En este proyecto vale destacar que el desarrollo de las iteraciones ha sido ajustado de forma tal que al final de cada una de ellas se tenga un entregable del sistema y se vaya cumpliendo con los requerimientos establecidos en la Lista de Reserva del Producto.

3.1 Desarrollo de las Iteraciones

Durante la Fase de Exploración y Planificación se detallaron las Historias de Usuario correspondientes a cada una de las iteraciones a desarrollar y se tuvo en cuenta las prioridades y restricciones de tiempos previstas por el cliente. Para darle cumplimiento a cada Historias de Usuario primeramente se realiza una revisión del plan de iteraciones y de ser necesario se realizan modificaciones. Dentro del contenido de este plan se descomponen las Historias de Usuario en Tareas Ingeniería (TI) o tareas de programación, asignándole de esta forma un equipo de desarrollo (o una persona) que será el responsable de su implementación. Con ellas se pretende cumplir con las funcionalidades básicas que luego conformarán las funcionalidades generales de cada historia. Las tareas no tienen que necesariamente ser entendidas por el cliente, pueden ser escritas en lenguajes técnicos, pues las mismas son usadas únicamente por los programadores.

3.1.1 Iteración 1

En esta iteración se le dio cumplimiento a la implementación de las Historias de Usuario que se consideraron de importancia primordial para el desarrollo y funcionamiento básico del sistema con el fin de obtener una versión inicial del producto con algunas de las funcionalidades básicas del usuario.

Historias de Usuario	Tiempo de Implementación (semanas)	
	Estimación	Real
Instalación de Apache	0.125	0.125
Instalación de PHP	0.125	0.125
Instalación y configuración de PostgreSQL	0.25	0.25
Instalación del Core de Drupal	0.25	0.25
Configuración básica del Core de Drupal	0.25	0.25
Definir plantilla del usuario	1	0.7
Definición plantilla del administrador	1	1

Tabla 1: Historias de Usuarios de la 1ra Iteración

3.1.1.1 Drupal como Administrador de Contenido (CMS) para la Intranet2.0

Como se ha planteado en capítulos anteriores el diseño de Drupal es especialmente idóneo para construir y gestionar comunidades en Internet. No obstante, su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales disponibles, hace que sea adecuado para realizar muchos tipos diferentes de sitio Web.

Preparación del Servidor

Antes de iniciar la instalación del CMS Drupal se hace necesario tener preparado un servidor Web con las características idóneas para el funcionamiento estable y eficiente del Core (núcleo básico).

Para ello se dispone de un servidor con Sistema Operativo¹ Debian 5 (distribución Lenny) que debe disponer además de los paquetes necesarios para permitir el alojamiento de las páginas web (Servidor Web), un intérprete del lenguaje del Core de Drupal (PHP) y un motor de almacenamiento de los datos del CMS (Sistema de Gestión de Base de Datos).

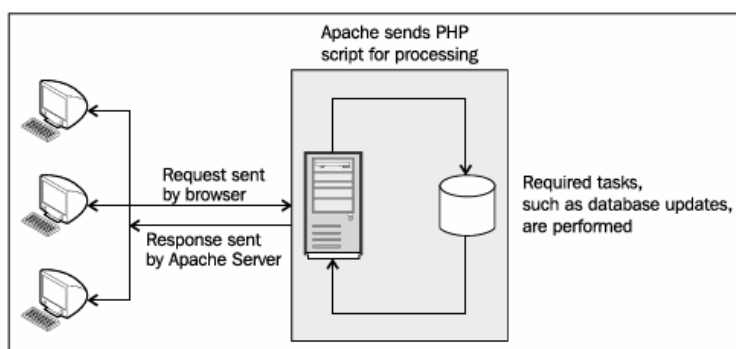


Figura 1: Entorno de trabajo de Drupal

Apache como servidor web de alojamiento

El servidor HTTP Apache es un software (libre) servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows, Macintosh y otras que implementan el protocolo HTTP/1.1 y la noción de sitio virtual.

¹ El sistema operativo en que se instale el CMS es a conveniencia del usuario o según la política que rige la institución que implementara el sistema. Puede ser GNU/Linux, Windows, Mac.

- ✓ Modular
- ✓ Open source
- ✓ Multi-plataforma
- ✓ Extensible
- ✓ Popular (fácil conseguir ayuda/suporte)
- ✓ Gratuito

PostgreSQL como Sistema de Gestión de Datos

La instalación de PostgreSQL dentro del sistema Debian difiere en dependencia de la versión que se vaya a utilizar. En el caso de aquellos que lo instalan a partir de los repositorios² sería para la versión 8.3:

```
sudo apt-get install postgresql-8.3 (Se instala el paquete Postgres)
sudo apt-get install libapache2-mod-auth-pgsql php5-pgsql (Paquetes adicionales
para conectar con Apache como servidor web y PHP como intérprete de
lenguaje)
sudo /etc/init.d/apache2 restart (Reiniciar apache)
sudo /etc/init.d/postgresql-8.3 restart (Reiniciar postgres)
```

Pero se puede descargar desde el sitio oficial de descargas del propio PostgreSQL (<ftp://ftp.postgresql.org/pub/source/>). Las versiones más actuales para el sistema operativo en cuestión en el momento en que se desarrolló esta investigación es la versión 8.4. Los pasos se describen a continuación:

1. *Descargar la versión estable más actual desde el sitio oficial:*
<ftp://ftp.postgresql.org/pub/source/>

Nota: preferentemente descargar un ejecutable para Debian, .deb o .bin, para este caso se escogió postgresql-8.4.1-1-linux.bin

2. *Damos permisos de ejecución al archivo descargado:*

```
sudo chmod 755 postgresql-8.4.1-1-linux.bin
```

3. *Instalamos el servidor postgres y seguimos los pasos de la instalación:*

```
sudo ./ postgresql-8.4.1-1-linux.bin
```

² Un **repositorio**, **depósito** o **archivo** es un sitio centralizado donde se almacena y mantiene información digital, habitualmente bases de datos o archivos informáticos. En el caso de GNU/Linux es donde se almacenan todos los paquetes de software necesarios.

Hasta aquí solo se ha instalado el PostgreSQL como motor de almacenamiento o sistema de gestión de base de datos. Sin embargo es necesario realizar las configuraciones pertinentes a modo de seguridad y para permitir conexiones remotas al propio gestor, pues por defecto solo es administrable desde el host local.

1. Editar dentro del directorio PostgreSQL/8.4/data/ el archivo `pg_hba.conf`:
`sudo nano /opt/PostgreSQL/8.4/data/pg_hba.conf`

2. Cambiar los valores que se encuentran en la sección:

```
# TYPE DATABASE USER CIDR-ADDRESS METHOD
```

Y permitir las conexiones remotas:

```
# TYPE DATABASE USER CIDR-ADDRESS METHOD

# "local" is for Unix domain socket connections only
local all all ident
# IPv4 local connections:
host all all 127.0.0.1/16 md5
host all all 10.28.0.0/16 md5
# IPv6 local connections:
host all all ::1/128 md5
```

3. Configurar el usuario con el cual se conectara el cliente al servidor. Se debe editar el archivo PostgreSQL/8.4/data/pg_ident.conf:

```
sudo nano /opt/PostgreSQL/8.4/data/pg_ident.conf
```

4. Agregar el usuario de la forma:

```
# MAPNAME SYSTEM-USERNAME PG-USERNAME
postgres postgres postgres
```

Donde mapname es el nombre a mapear, sistema-username es el nombre de usuario dentro del sistema y pg-username es el nombre de usuario dentro del servidor PostgreSQL.

5. Por último se debe reiniciar el servidor de datos:

```
sudo /etc/init.d/postgresql-8.3 restart
```

Ahora solo resta crear la Base de Datos que almacenará todo los datos gestionados desde el CMS:

1. Se utiliza pgadmin III como herramienta case para administrar el servidor de datos:

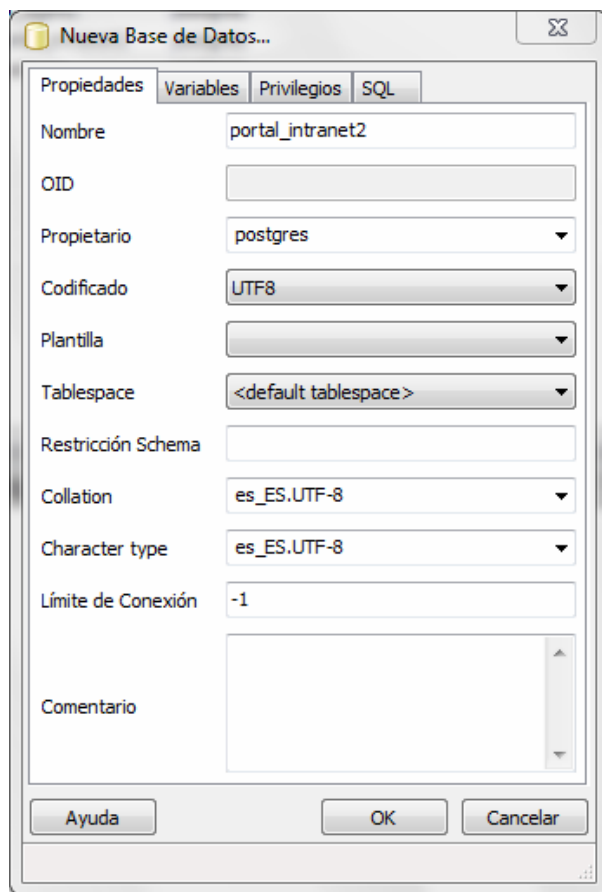


Figura 2: Creando la Base de Datos desde pgadmin III

Nombre de la Base de Datos: portal_intranet2
Usuario propietario dentro del SGBD: postgres

3.1.1.2 Instalando Drupal

Instalación del Sistema Base (Core de Drupal)

Ya se tiene un escenario correctamente preparado y listo para permitir la instalación de Drupal para lo cual se deben seguir los siguientes pasos:

1. Descargue versión más reciente de Drupal 6. Lo cual creara un archivo llamado (drupal-6.x.tar.gz).
2. Descomprima el archivo y descargue en algún directorio de su PC. Lo cual creara una carpeta llamada (drupal-6.x) que contiene las carpetas y archivos de instalación de Drupal.
3. Descargue versión más reciente de la traducción para Drupal 6 (en este caso 6.x-1.0). Lo cual creara un archivo llamado (es-6.x-1.0.tar.gz).
4. Descomprima el archivo descargado en algún directorio de su PC. Lo cual creara varias carpetas y archivos txt.

5. Copie los archivos y carpetas de traducción dentro de la carpeta principal de Drupal (drupal-6.x) y reemplace los archivos originales.

Subir archivos de Drupal 6.x al Servidor

1. Para instalar Drupal en sitio remoto. A través de Filezilla suba todos los archivos existentes dentro de la carpeta (drupal-6.x) a la carpeta raíz de su sitio remoto (servidor externo).
2. Para instalar Drupal en modo local. Copie todos los archivos existentes dentro de la carpeta (drupal-6.x) a la carpeta raíz de su servidor.

Nota: Si utilizó otro paquete de instalación en modo local, vea la ayuda para saber cuál es su carpeta raíz.

3. **Acceda a la carpeta que contiene Drupal** escribiendo la dirección de su sitio en la barra de exploración de su navegador web, ejemplo:

En sitio remoto: <http://www.nombremisitio.com>.

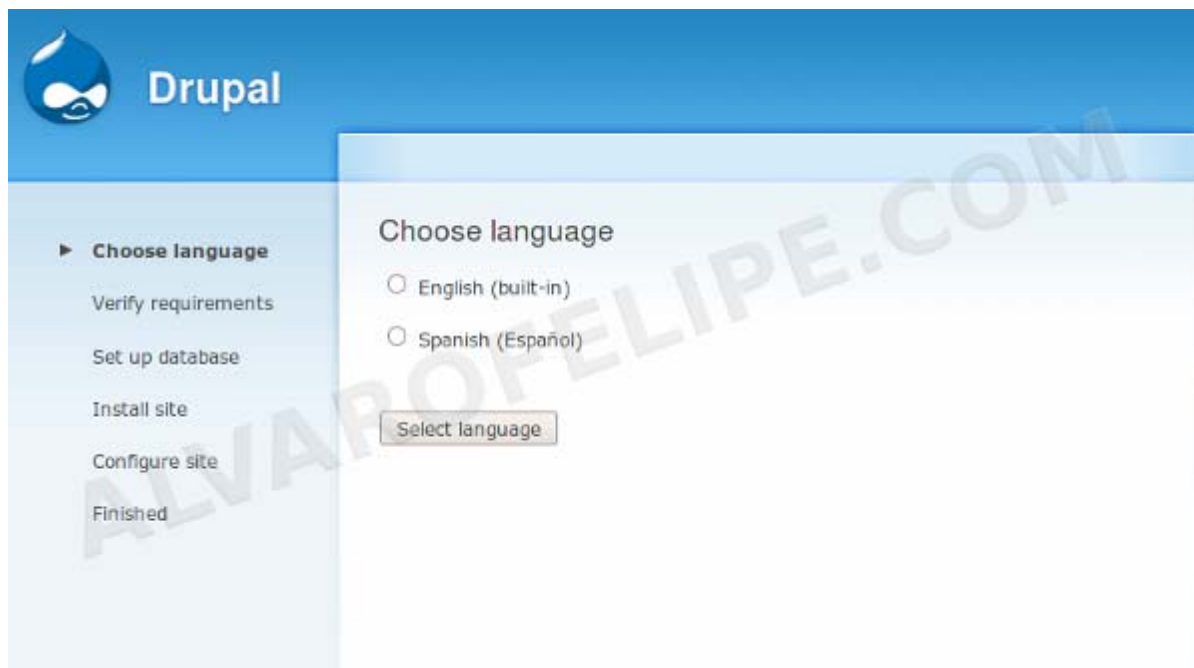
En modo local: <http://localhost>

En su navegador, pulse la opción Ir a la URL (para Actualizar) y siga las instrucciones.

Configuración Básica

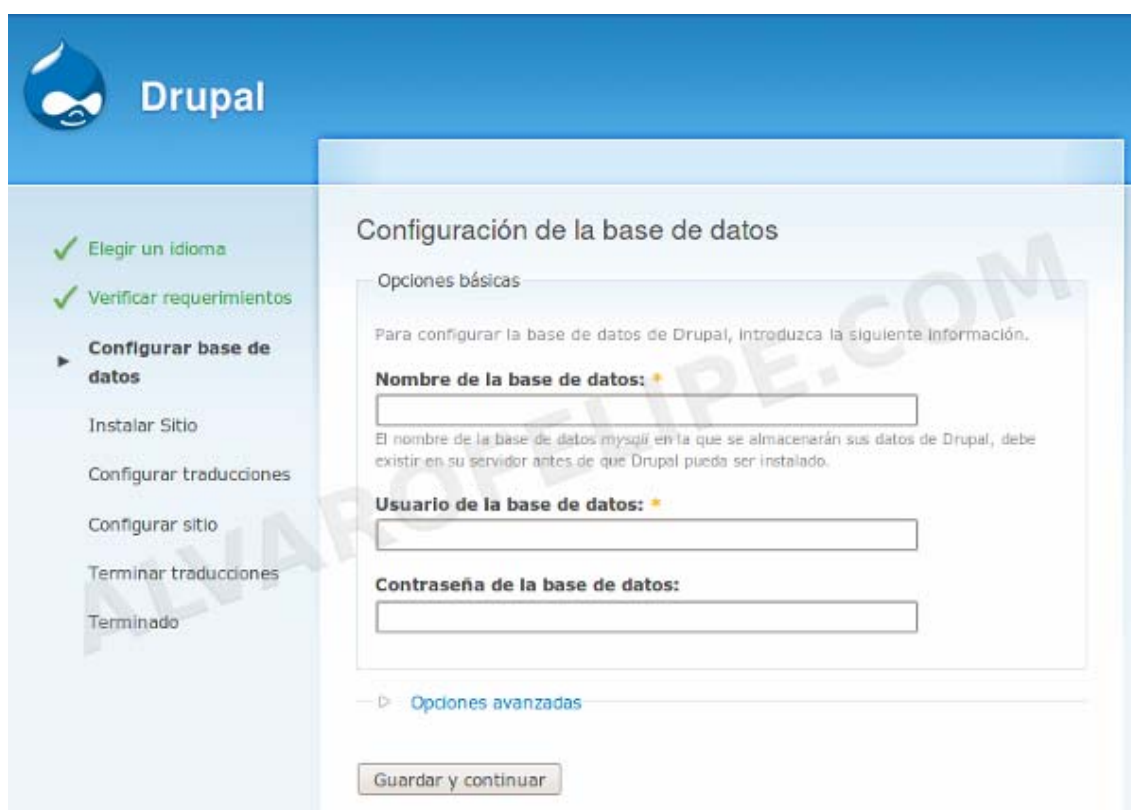
Una vez que se ha abierto la URL del DRUPAL en el navegador inicia el proceso de configuración básica para el nombrado CMS donde se especifica, entre otras cuestiones, el lenguaje a utilizar y dónde almacenar los datos.

PASO 1: Escoger el lenguaje.



Se escoge idioma español

PASO 2: Configurar el acceso a la Base de Datos (BD) anteriormente creada.



Nota: en opciones avanzadas se puede especificar además el host donde se encuentra la BD, así como el prefijo para las tablas.

PASO 3: Configurar nombre del sitio y cuenta de administración.

The screenshot shows the Drupal installation configuration interface. On the left, a sidebar contains a list of steps: 'Elegir un idioma', 'Verificar requerimientos', 'Configurar base de datos', 'Instalar Sitio', 'Configurar traducciones', 'Configurar sitio' (highlighted), 'Terminar traducciones', and 'Terminado'. The main content area is titled 'Configurar sitio' and contains several sections: 1. A red warning box: 'Se han realizado todos los cambios necesarios a ./sites/default y ./sites/default/settings.php, por lo que ahora debe remover los permisos de escritura sobre estos archivos a fin de evitar riesgos de seguridad. Si no está seguro sobre cómo hacerlo, por favor consulte el manual en línea.' 2. A text prompt: 'Para configurar su sitio web, por favor proporcione la siguiente información.' 3. 'Información del sitio' section: 'Nombre del sitio web:' with a text input containing 'localhost'; 'e-mail del sitio:' with an empty text input. Below this is a note: 'La dirección De en los correos automáticos enviados durante el registro y tras la solicitud de nueva contraseña, y otros avisos. (Use una dirección que termine con el dominio de su sitio para ayudarle a evitar que su correo sea etiquetado como spam.)' 4. 'Cuenta del administrador' section: A note: 'La cuenta de administrador tiene acceso completo al sitio; recibirá automáticamente todos los permisos y puede realizar cualquier actividad administrativa. Esta será la única cuenta capaz de ejecutar ciertas actividades, de modo que guarde a buen recaudo sus credenciales.' 'Usuario:' with an empty text input. A note: 'Se permite la utilización de espacios; los signos de puntuación no están permitidos a excepción de puntos, guiones altos y guiones bajos.' 'Dirección de correo electrónico:' with an empty text input. A note: 'Todos los correos del sistema serán enviadas a esta dirección. La dirección de correo no se hace pública y solo se usará si desea recibir una nueva contraseña o desea recibir ciertas noticias o notificaciones por correo electrónico.' 'Contraseña:' with an empty text input. 'Confirmar contraseña:' with an empty text input. 5. 'Configuración del servidor' section: 'Zona horaria predefinida:' with a dropdown menu showing 'Sábado, Febrero 20, 2010 - 13:46 -0500'. A note: 'Por defecto, las fechas en este sitio se mostrarán en la zona horaria seleccionada.' 'URLs limpios:' with radio buttons for 'Desactivado' (selected) and 'Activado'. A note: 'Esta opción hace que Drupal prepare URL "limpios" (esto es, sin incluir ?q= en el URL). La configuración de su sistema no soporta actualmente esta característica. La página del manual sobre URL limpios tiene más información sobre posibles problemas.' 'Actualizar notificaciones:' with a checked checkbox for 'Verificar actualizaciones automáticamente'. A note: 'Con esta opción marcada, Drupal le notificará cuando estén disponibles nuevas versiones. Esto mejorará de modo importante la seguridad de su sitio y lo recomendamos ampliamente. La opción requiere que su sitio envíe periódicamente información anónima sobre los componentes instalados a drupal.org. Para más información por favor vea la información sobre notificación de actualizaciones.' At the bottom of the main content area is a 'Guardar y continuar' button.

PASO 4: Esperar a que Drupal finalice la configuración



El nuevo sitio creado (todavía vacío), debe verse así:



3.1.1.3 Definición de Plantillas

Plantilla del Usuario

Cuando se desea una flexibilidad de diseño significativa a la hora de implementar y/o modificar una plantilla, Drupal resulta ser un CMS muy potente ya que parece llevarse todas las palmas dada su flexibilidad para desarrollar las plantillas y su uso intensivo de CSS y XHTML; la responsabilidad de llevar a cabo esto encaja del lado del Administrador.

Drupal 6 llama a las funciones de temas y plantillas **theme hooks**. Un theme hook es contenido que puede ser modificado por un tema y usualmente es el resultado de tus módulos.

En Drupal 6 es necesario definir qué funciones para temas o plantillas utilizarás por anticipado; es decir, debes registrar los theme hooks. Esto permite que sea más eficiente al evitar **descubrir** los theme hooks en tiempo de ejecución. Usa el nuevo hook_theme para registrar theme hooks, esta función retorna un array con las funciones de temas o plantillas que tu módulo ofrece y las variables disponibles. Para reemplazar una función de tema o plantilla basta solo con un poco de conocimientos de CSS y PHP.

Para poder visualizar los cambios efectuados en la plantilla es necesario borrar el registro de temas ya que guarda todo lo que sabe de los Theme hooks en este registro. La manera más común de realizar esta función es visitando:

[Administer > Site configuration > Performance](#) *y haz clic en Clear cached data.*

Para definir una o más regiones para bloques en la plantilla basta solo con añadir estas líneas al archivo.info de la plantilla usuario:

```
name = nombre del Tema  
description = Descripción del Tema  
version = VERSION  
core = 6.x  
engine = phptemplate  
stylesheets[all][] = style.css  
stylesheets[print][] = print.css  
regions[left] = left sidebar  
regions[right] = right sidebar  
regions[content] = content  
regions[header] = header  
regions[footer] = footer  
regions[social] = social
```

Nota: Las opciones que empiezan con **regiones** es texto simple pero la sintaxis es similar a la definición de un array en PHP. Estas líneas le dicen al tema qué regiones tendrá disponibles para bloques, los nombres entre corchetes serán los nombres de las variables en las plantillas, **regions[*left*]** creará una variable *\$left*, por ejemplo.

¿Qué hacer ante los problemas más frecuentes?

1. Cuando las modificaciones realizadas en la plantilla del usuario no se visualizan en el navegador.
 - a) visite *Administer > Site configuration > Performance* y haz clic en *Clear cached data* y borra el registro de Temas.
2. Se trata de seleccionar los temas por prioridades y Drupal se comporta de forma extraña o genera conflictos
 - a) Las plantillas se deben instalar en alguno de estas tres carpetas
 - ✓ *sites/default/themes*- Para las instalaciones simples.
 - *sites/all/themes* - En instalaciones de tipo multisite
 - *sites/nombredelsitio/themes*- Para instalaciones multisite las plantillas que estén aquí podrán ser utilizados solamente por el sitio especificado.

Plantilla del Administrador

Una vez que ya se ha realizado un análisis de la plantilla del usuario es aún más sencillo de comprender cómo funciona la plantilla del administrador al no diferir mucho de la plantilla del usuario; es más, la diferencia de esta plantilla consiste en que para generarla se tienen en cuenta un sin número de cuestiones relacionadas con el administrador al dejar atrás lo referente a la plantilla del usuario, es decir, se tienen que ver que las funcionalidades de esta plantilla son mucho más extensas y por ende, debe ofrecer mucho más información, pero siempre se debe recordar los pasos a realizar en la misma y cuáles son los problemas más frecuentes con los que se puede encontrar en el transcurso de la implementación.

3.1.2 Iteración 2

Esta iteración surge con el objetivo de dar cumplimiento a las HU de definición e implementación de los módulos funcionales del Portal. La mayor relevancia de esta iteración es que mediante esta el sitio Web cobrará vida y será la encargada de extender las funcionalidades del sitio en cuestión.

Historias de Usuario	Tiempo de Implementación (semanas)	
	Estimación	Real
Definición de los módulos funcionales del Portal	1	0.7
Implementación de los módulos funcionales del Portal	2	2

Tabla 2: Historias de Usuarios de la 2da Iteración

Nótese que esta iteración no hace alusión a los módulos que se encuentran dentro del paquete Core-Obligatorio y el del Core-Opcional. Puesto que los módulos que conforman estos paquetes son los que trae Drupal predeterminado en su instalación. Por tal motivo se hablará de los módulos que conforman los demás paquetes que se utilizaron para el desarrollo del presente proyecto.

¿Qué hacer cuando se necesita una funcionalidad determinada?

Cuando esta funcionalidad se encuentra en un módulo contribuido, se descarga y se coloca en la carpeta **modules**. Esto puede provocar conflictos en Drupal a la hora de seleccionar los módulos por prioridades, ya que los que están dentro de la carpeta modules son considerados parte del Core del Drupal. Teniendo en cuenta esto, para que estos cobren vida se deben instalar en algunas de estas carpetas:

- ✓ *sites/default/modules* - Para las instalaciones simples.
- *sites/all/modules* - En instalaciones de tipo multisite (varios sitios)

Con vista a lograr el sitio satisfaga las necesidades de la comunidad y que contengan las funcionalidades necesarias y suficientes se definieron 6 agrupaciones de módulos principales, dentro de estos se aglomeran una serie de módulos (valga la redundancia) para llevar a cabo dichas funcionalidades, los cuales se verán a medida de que se vaya generando su dependencia en el sitio. Estos son:

- CCK
- ImageCache
- Otros
- User Interface
- Views
- User Login Bar

CCK: Es una de las herramientas más potentes que ofrece Drupal y es considerado como una de las principales ventajas de este CMS ante los otros, este a su vez anida módulos como:

Content: Este módulo permite al administrador definir nuevos tipos de contenidos, este módulo es la base para el funcionamiento de otros módulos con los cuales se amplía la funcionalidad del paquete CCK:

Content Copy: Permite activar la habilidad de importar y exportar definiciones de campos.

Content Permissions: Cambiar los permisos del nivel de campo para los campos de CCK.

Embedded Audio Field: Define un tipo de campo para exponer los deferentes tipo de música como de fiesta u otros tipos de audio.

Embedded Image Field: Define un tipo de campo para exponer imágenes desde el proveedor como Flickr o Photobucket.

Embedded Media Field: Provee un artefacto para integrar módulos a varios contenidos como videos, imágenes y audios cercanos.

Embedded Media Thumbnail: Permite almacenar campos de media.

Embedded Video Field: Define un tipo de campo para mostrar partes de videos como *YouTube* y *Google Video*.

Embedded Wave Field: Define un tipo de campo para mostrar movimientos desde proveedores como Google y otros.

Fieldgroup: Crea un grupo para mostrar los campos de CCK.

Node Reference: Define un tipo de campo para hacer referencia a un nodo desde otro.

Number: Define un campo de tipo numérico.

Option Widgets: Define una selección entre check box y radio button en una caja de texto y un campo de numérico.

Text: Define un tipo de campo para textos simples.

User Reference: Define un tipo de campo para hacer referencia a un usuarios desde un nodo.

El paquete ImageCache anida un conjunto de módulos cuya función es el tratamiento de las imágenes que serán expuestos en el sitio desarrollado por Drupal. Los módulos que conforman este paquete son:

ImageAPI: Brinda múltiples juegos de herramientas.

ImageAPI GD2: EL constructor GD2 usa un soporte de procesamiento de imágenes.

ImageAPI ImageMagick: Brinda una línea de commando ImageMagick.

ImageCache: Manipulador de imagen dinámicas y cache.

ImageCache UI: ImageCache Interfaz de usuario.

Dentro del paquete Otros se agrupan los módulos que se han tratado para brindar una mayor funcionalidad al lado del administrador, es decir, la importancia de este paquete radica en que da la posibilidad al administrador de extender sus funcionalidades en cuanto al tratamiento de taxonomías, URL, imágenes y demás. Estos módulos son los que se describen a continuación:

Embedded Inline Media: Permite a los usuarios agregar dentro de áreas de textos archivos media.

Global Redirect: Buscador para un alias par alas actuales URL y redirecciona 301 si es encontrado. Detiene la subida de contenido duplicado cuando el módulo **Path** se encuentra activado.

IMCE: Un navegador y gestor de subidas de documentos que reconoce directorios personales y cuotas por usuario.

Pathauto: Proporciona un método para que los módulos puedan crear automáticamente alias de URLs del contenido que gestionan, es decir, en vez de tener `http://www.examples.com/node/1`, sería `http://www.examples.com/noticias/titulo-de-la-noticia`.

Read More link: Activa la opción de leer más link para mostrar en nodos

Taxonomy breadcrumb: Activa trazas por vocabularios y términos creados en la taxonomía y permite una asociación de un nodo con un término.

Token: Proporciona una API compartida para el remplazo de comodines textuales con datos concretos (es decir remplazar caracteres especiales como acentos y otros a la hora de crear un alias de una URL).

Token actions: Proporciona versiones mejoradas de las acciones del núcleo de Drupal usando el módulo de comodines Token.

TokenSTARTER: Provee un tokens adicional y una base en la cual se puede construir tu tokens.

Transliteration: Provee la transfiguración para entrada de textos en UTF-8 (formato unicode que admite todo tipo de caracteres) y nombres de archivos.

En el paquete **User Interface** se encuentran los módulos:

IMCE Wysiwyg API bridge: Activa IMCE como plugin para integrar editores client-side via Wysiwyg API.

Wysiwyg: Permite a los usuarios editar contenidos con editores cliente-site.

Como su nombre indica el paquete **Views es el encargado de almacenar los módulos correspondientes a las vistas del sitio, es son:**

Views: Create customized lists and queries from your database.

Views exporter: Permite exporter múltiples vistas a la vez.

Views UI: Permite administrar interfaz de vistas, con estos módulos se puede crear o editar tus vistas.

3.1.3 Iteración 3

Historias de Usuario	Tiempo de Implementación (semanas)	
	Estimación	Real
Definición de los Niveles de usuarios dentro del Portal	0.5	0.5
Implementación de los Niveles de usuarios dentro del Portal	0.5	0.5
Definición de los menús de navegación en el Portal	0.5	0.5
Implementación de los menús de navegación en el Portal	1	0.7

Tabla 3: Historias de Usuarios de la 3ra Iteración

Los niveles de usuarios se definen con el objetivo de brindar al sitio una mayor seguridad ya que estos determinan como será la navegación de cada usuario, a que información tendrán acceso, en fin describen el proceder de todos en su tránsito por el Portal-Intranet. Esto tiene como principal importancia que se logra tener un sitio organizacional que le permite en todo momento al Webmaster evitar que los usuarios no autorizados hagan uso de una determinada información.

Por tal motivo para el presente proyecto se determinaron 3 niveles de usuarios o roles de usuarios, estos son:

Administrador ó Webmaster: Este dispondrá de todos los privilegios con que cuenta el Portal-Intranet, tendrá autoridad sobre todos los demás usuarios, tendrá propiedades como:

- ✓ Podrá Insertar, eliminar y modificar el perfil de cualquier usuario que navegue por dicho Portal-Intranet.
- ✓ Podrá incluir servicios en el sitio
- ✓ Podrá eliminar y modificar cualquiera de los servicios de los que dispone el Portal-Intranet.
- ✓ Podrá insertar, eliminar y modificar cualquier información difundida a partir del Portal-Intranet.

- ✓ Tendrá derechos para actualizar la interfaz del sitio.
- ✓ Podrá actualizar complementos de contenido multimedia, tales como imágenes, audio y videos.

Usuarios Editores: Estos podrán acceder la mayoría de las aéreas con que cuenta el portal, la diferencia de un usuario editor y el Webmaster consiste a que estos no podrán modificar los perfiles de los usuarios del portal, ni insertar nuevos servicios al mismo. Tendrán la posibilidad insertar y editar artículos en el sitio.

Usuarios Autenticados: Los usuarios autenticados son aquellos solo podrán acceder a la información a la cual tengan previo autorizo, podrán hacer uso de los servicios que brinda la intranet y determinados momentos tendrán la posibilidad de agregar comentarios a informaciones del sitio.

¿Que se entiende como Menú de un portal?

Se entiende como menú la interfaz predeterminada de Drupal, usted notará muchos enlaces dados como etiquetas. Estos son conocidos en el sistema del menú como "las tareas locales", y son dados como etiquetas por defecto, sin embargo otras presentaciones son posibles. La función local de tareas lo mismo que otras opciones del menú en la mayoría de respetos. Es conveniente que los nombres de estas tareas deberían ser abruptamente verbos si es posible. Además, una tarea local "predeterminada" debería ser provista para cada set, Al visitar la opción del menú del padre de una tarea local, la tarea local predeterminada será dada como si es seleccionada. Esto provee para un usuario normal de la etiqueta la experiencia. Esta tarea predeterminada es especial adentro que se relaciona no para su camino provisto, pero para su padre el camino del artículo en lugar de eso. El camino de la tarea predeterminada se usa sólo para colocar eso apropiadamente en la jerarquía del menú.

Para un mejor entendimiento de lo que se esta hablando es necesario analizar a que se hace referencia cuando se habla de Menú. Entiéndase como menú en Drupal a la parte que permite agrupar las principales informaciones a la que se

puede acceder una vez que se esta dentro de la interfaz de un sitio Web, por otra parte, los mismos dan la posibilidad de crear submenús, es decir, estos submenús tienen herencia y como explica la lógica de esta palabra, los submenús heredan los atributos del menú padre, como URL y demás. Además estos por lo general se encuentran en una parte visible y a la que se puede acceder fácilmente, otra de sus principales características es que casi siempre están presentes en todas las vistas generadas por el sitio. EL menú mas popular en los sitios Web es el menú inicio (home para los sitios de habla inglesa).

Fase de elaboración

En la fase de elaboración de los menús es importante y de carácter obligatorio en un sitio gestionado por el CMS Drupal tener en cuenta determinados campos, los cuales contienen la valiosa información de los menús durante todo el periodo de vida de estos en el sitio, estos son:

Path: Almacena el parcial de la URL a la pagina para este ítem del menú, la información se almacena de tipo string

Title: Almacena el titulo que este ítem del menú tendrá en el menú. La información se almacena de tipo string.

Type: Constante que denomina el tipo del ítem del menú, la información es almacenada de tipo integer.

Para el caso del presente proyecto fueron definidos por el AI(Arquitecto de la Información) del equipo de desarrollo los siguientes como menús y submenús:

Menús primarios:

1. Inicio
2. Universidad
 - 2.1. Misión/ Visión
 - 2.2. Historia
 - 2.3. Sobre Antonio Núñez Jiménez
 - 2.4. Objetivos Estratégicos

- 2.5. Consejo Universitario
- 2.6. Carreras Rectoras

- 3. Estructura
 - 3.1. Administración
 - 3.2. Facultades
 - 3.3. Centros de Estudios
- 4. Formación
 - 4.1. Moodle
 - 4.2. Clases Pregrado
- 5. Biblioteca
 - 5.1. Biblioteca ISMM
 - 5.2. Otras Bibliotecas
- 6. Servicios
 - 6.1. Correo Estudiantes
 - 6.2. Correo Profesores
 - 6.3. Ftp Central
 - 6.4. Encuestas Online
 - 6.5. Comunidad ISMM
 - 6.6. Recuperador SIGENU
 - 6.7. Cambio de Contraseña

Menús secundarios:

- 1. Avisos
- 2. Sitios de Interés
- 3. Prensa nacional
- 4. Destacamos
- 5. Relevantes

3.1.4 Iteración 4

Historias de Usuario	Tiempo de Implementación (semanas)	
	Estimación	Real
Definición del contenido de información	1.5	1

Tabla 4: Historias de Usuarios de la 4ta Iteración

Esta iteración es la responsable de llevar a cabo un análisis de los contenidos de información y la agrupación de estos según criterios comunes, por lo que se realizara un estudio primeramente sobre las propiedades y su comportamiento en el CMS Drupal.

Importante sobre los contenidos en Drupal

- *Enlaces permanentes (Permalinks)* Todo el contenido creado en Drupal tiene un enlace permanente asociado a él para que pueda ser enlazado externamente sin temor de que el enlace falle en el futuro.
- *Búsqueda* todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- *Objetos de Contenido (Nodos)* El contenido creado en Drupal es, funcionalmente, un objeto (Nodo). Esto permite un tratamiento uniforme de la información, como una misma cola de moderación para envíos de diferentes tipos, promocionar cualquiera de estos objetos a la página principal o permitir comentarios -o no- sobre cada objeto.
- Drupal separa el contenido de la presentación permitiendo controlar o cambiar fácilmente el aspecto del sitio web.

- *Sindicación del contenido* Drupal exporta el contenido en formato RDF/RSS para ser utilizado por otros sitios web. Esto permite que cualquiera con un 'Agregador de Noticias', ejemplo *NetNewsWire*.

3.1.5 Iteración 5

Esta iteración hace alusión a las cuestiones relacionadas con la implementación de la información dentro del portal, por tal motivo se debe hacer un análisis de que sucede internamente con la gestión del contenido implementado en este CMS.

Historias de Usuario	Tiempo de Implementación (semanas)	
	Estimación	Real
Implementar los contenidos de Información dentro del Portal	3	3

Tabla 5: Historias de Usuarios de la 5ta Iteración

Interesante

Creación de Contenido

Primeramente es necesario entender que casi todo el contenido que se da de alta a través de Drupal es tratado internamente como un tipo de entidad genérica llamada nodo.

Un nodo es prácticamente cualquier texto que se introduce en Base de Datos y se muestra a través de la web. Esto permite que a cualquier tipo de contenido se le pueda asociar de forma individual una serie de atributos comunes, independientemente que sea una noticia, un artículo o el tema de un foro.

Los nodos se organizan asociándoles categorías. Por ejemplo, se puede indicar que un nodo es de la categoría "Noticias". De esta forma se consigue que se puedan listar fácilmente todas las noticias dadas de alta en la web. Pero además, el nodo se podría relacionar con los términos "Deportes" y "Fútbol" para que se sepa el tipo de información que contiene.

Además los CMS aportan herramienta para que los creadores de paginas web sin conocimientos técnicos puedan concentrarse en el contenido, por lo general proporcionan un editor de texto WYSIWYG, en el que el usuario visualiza el resultado final mientras escribe, al estilo de los editores comerciales

Hay otras herramientas como la edición de los documentos en XML, utilización de aplicaciones ofimáticas con la que se integra el CMS, importación de documentos existentes y editores que permiten añadir marcas, por lo general HTML, para indicar el formato y estructura del documento.

El CMS puede incorporar una o varias de estas herramientas, pero siempre tendría que proporcionar un editor WYSIWYG por su facilidad de uso y la comodidad de acceso desde cualquier ordenador con un navegador y acceso a Internet.

Gestión de contenidos

Los documentos creados se depositan en una base de datos central donde también se guardan el resto de datos de la Web, como son los datos relativos a los documentos (versiones hechas, autor, fecha de publicación, caducidad, etc.). Datos y preferencias del usuario, la estructura de la Web, etc.

La estructura de la Web se puede configurar con una herramienta que habitualmente, presenta una visión jerárquica del sitio y permite modificaciones. Mediante esta estructura se puede asignar un grupo a cada área, con responsables, editores, autores y usuarios con diferentes permisos, eso es necesario para facilitar el ciclo de trabajo (Workflow) con un circuito de edición que va desde el autor hasta el responsable final de la publicación.

Publicación

Una página aprobada se publica automáticamente cuando llega la fecha de publicación o cuando un usuario con previos permisos desea activarla mediante el panel de administración; y cuando caduca, o el administrador o usuario con permisos del sistema la desactiva, se archiva para futuras referencias. En su publicación se aplica el patrón definido por toda la web o para la sección concreta donde está situada, de forma que el resultado final es

un sitio Web con un aspecto consistente en todas sus páginas. Esta separación entre contenido y forma permite que se pueda modificar el aspecto visual de un sitio web sin afectar a los documentos ya creados y libera a los autores de preocupaciones por el diseño final de su página.

Presentación

Un CMS puede gestionar automáticamente la accesibilidad a la Web, con soporte de normas internacionales de accesibilidad como WAI, y adaptarse a las preferencias o necesidades de cada usuario. También puede proporcionar compatibilidad con los diferentes navegadores disponibles en todas las plataformas (Windows, Linux, Mac, Palm, etc.) y su capacidad de internacionalización le permite adaptarse al idioma, sistema de medidas y cultura del visitante.

El sistema se encarga de gestionar muchos otros aspectos como son los menús de navegación o la jerarquía de la página actual dentro de la web, añadiendo enlaces de forma automática. También gestiona todos los módulos, internos o externos, que incorpore al sistema. Así por ejemplo con un módulo de publicidad se mostraría un anuncio o mensaje animado, y con un módulo de foro se podría mostrar, en la página principal, el título de los últimos mensajes recibidos. Todo eso con los enlaces correspondientes y, evidentemente, siguiendo el patrón que los diseñadores hayan creado.

Mantenimiento y Actualización

Fácilmente actualizable, otorga una autonomía plena a los administradores del sistema para poder editar, añadir, modificar y eliminar documentos y secciones, así como una escalabilidad ilimitada en todos sus contenidos. Está diseñado para permitir definir distintos niveles de administración, de forma tal que el administrador principal pueda definir autores locales con autorización para introducir datos solo en determinadas áreas y que estos datos deban ser validados por el administrador general o un supervisor para ser publicados, o que no necesiten esta validación.

Categorías

Las categorías se utilizan para organizar el contenido. Y se implementan mediante vocabularios, términos, y las relaciones que se establecen entre ellos, que pueden ser simples o jerárquicas, incluyendo jerarquías múltiples. A este tipo de organización se le denomina taxonomía.

Los vocabularios se dan de alta en Drupal a través del menú *administrar->categorías->añadir vocabulario*

Una vez creado el vocabulario se puede seleccionar de la lista de categorías y añadirle términos a través de la opción de menú

administrar->categorías->añadir término.

Agregar Contenido

El contenido de la web se da de alta a través de la opción de menú **crear contenido nuevo**. No tiene mucho misterio. Un simple formulario para la introducción del texto con unas pocas opciones acerca del autor, formato y comentarios. Y por supuesto, el desplegable para la selección de un término del vocabulario "Secciones".

En el menú **administrar->contenido** se muestra una lista con todo el contenido dado de alta en la web y varias opciones muy sencillas de configurar.

3.1.6 Iteración 6

Historias de Usuario	Tiempo de Implementación (semanas)	
	Estimación	Real
Implementar funcionalidad de personalización del Portal a nivel de Usuario	3	3

Tabla 6: Historias de Usuarios de la 6ta Iteración

Con el propósito de facilitar la visualización y el acceso a la información necesaria, Drupal como CMS brinda la herramienta necesaria para personalizar el entorno Web de los usuarios de un sitio determinado, Drupal logra tal funcionalidad por defecto partiendo de la instalación del core obligatorio y de cores opcionales.

La personalización del sitio puede verse de formas diferentes, es decir, esta funcionalidad se puede lograr internamente, analizando el rol de cada uno de los usuarios dentro del sitio, los cuales están determinados por su naturaleza, desde la misma concepción del sistema (usuarios autenticados, usuarios editores y usuarios administradores) o partiendo de los usuarios del dominio. El presente trabajo propone realizar esta funcionalidad por medio de los usuarios del dominio dado a las características actuales del centro, ya que se cuenta con un dominio, lo que facilita que desde el mismo instante que el usuario acceda por medio a la red, puede hacer pleno uso del Portal-Intranet y por tanto modificaciones en su perfil.

Una vez que el usuario se encuentra dentro del Portal-Intranet, para lograr dicha funcionalidad debe acceder a **mi cuenta** y configurar la visualización del mismo.

El usuario podrá controlar la visualización sobre los bloques **contenido popular, cumulus tag cloud, banner mundial de futbol 2010, y contador de visitas**.

Notése que no se hace alusión a la totalidad de los bloques, esto se debe a que solo el administrador del sitio o usuarios con previo autorizo por el Webmaster podrán realizar modificaciones o controlar la visualización de determinados bloques como ***Reflexiones de Fidel.***

Conclusiones

Con el desarrollo de este capítulo se ha dado una breve descripción de todas las iteraciones presentes para este proyecto, las cuales sirvieron para entender las funcionalidades del mismo. Uno de los puntos fuertes de Drupal es la facilidad con la que puede extenderse o modificarse su funcionamiento. No es frecuente encontrar software que lo ponga tan fácil. Se debe señalar que hacer un proyecto sencillo es relativamente fácil si se tiene cierta experiencia como programador y se sabe leer el modelo de datos. Drupal da bastantes facilidades a la hora de construir formularios, acceder a base de datos, e integrar el funcionamiento de un módulo dentro del funcionamiento normal de Drupal.

Capítulo 4: Pruebas de aceptación.

Introducción.

El presente capítulo servirá para dar una visión de cómo se comportan en el proyecto las pruebas de aceptación, recuérdese que estas son anteriormente confeccionadas por el cliente y ya en esta etapa del proyecto es que el cliente realiza la comprobación sobre cada HU llevada a cabo, acción que le permite al cliente tener y brindar al equipo de desarrollo una valoración de la aplicación, es decir, si la aplicación se concluyo satisfactoriamente o no.

Pruebas.

En la metodología XP es esencial el desarrollo de las pruebas, esto permite constantemente probar la calidad del código. Cada vez que se quiere implementar las funcionalidades que tendrá el software, XP propone una redacción sencilla de prueba, para ser pasada por el código posteriormente. El desarrollo constante de las pruebas permite que se desarrolle un producto con mayor calidad dando a los programadores una mayor seguridad en el trabajo que desarrollan.

Pruebas de aceptación.

Las pruebas de aceptación se realizan con el propósito de validar de un sistema, en cuanto a si cumple o no con la funcionalidad esperada, y permitir al usuario de dicho sistema que determine su aceptación. Estas pruebas son definidas por el cliente y elaboradas por el equipo de desarrollo y se tiene siempre en cuenta que la aprobación final del producto corresponde al cliente.

Esto proporciona ventajas como:

- ✓ Son el termómetro de los desarrolladores, fundamentalmente de los programadores a la hora de medir la calidad de su trabajo.
- ✓ Garantizan la entrega de un producto con calidad que responde a las necesidades del cliente.

Durante las iteraciones las HU seleccionadas serán traducidas a pruebas de aceptación. En ellas se especifican, desde la perspectiva del cliente, los escenarios para probar que ha sido implementada correctamente. Esta puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. El objetivo final es garantizar que las funcionalidades requeridas por el cliente hayan sido cumplidas. Una HU no se considera completa hasta que no ha pasado por sus pruebas de aceptación.

Caso de Prueba de Aceptación	
Código Caso de Prueba: [Inicial del proyecto-número de la HU a la que pertenece la prueba-número de la prueba.]	Nombre Historia de Usuario: [Nombre de la HU a realizar prueba.]
Nombre de la persona que realiza la prueba: [Nombre y apellidos.]	
Descripción de la Prueba: [Descripción de la prueba realizada.]	
Condiciones de Ejecución: [Condiciones necesarias para poder realizar la prueba.]	
Entrada / Pasos de ejecución: [Serie de pasos necesarios para lograr la realización de la HU, y así realizar la prueba.]	
Resultado Esperado: [Que cumpla con las restricciones del producto.]	
Evaluación de la Prueba: [Satisfactoria o no satisfactoria.]	

Tabla 7: Plantilla Prueba de Aceptación

A continuación se describen los casos de Pruebas de Aceptación que fueron ejecutados por el cliente (ISMM). Este emite para cada una su criterio final, lo cual ofrece una visión general sobre la satisfacción o no del cliente.

Caso de Prueba de Aceptación	
Código Caso de Prueba: PI-5-1	Nombre Historia de Usuario: Definir plantilla del usuario.
Nombre de la persona que realiza la prueba: ISMM.	
Descripción de la Prueba: Se define la plantilla del usuario, Hojas de Estilo CSS, Tipografía de letras, imágenes y texto	
Condiciones de Ejecución: Ejecutar la aplicación desde un navegador Web (se recomienda varios navegadores para comprobar la compatibilidad) y verificar que el sistema carga correctamente	
Entrada / Pasos de ejecución: Abrir el navegador y poner la URL donde se encuentra alojado el Sistema	
Resultado Esperado: se definió correctamente la plantilla del usuario.	
Evaluación de la Prueba: Satisfactoria	

Tabla 8: Caso de Prueba de Aceptación definir plantilla del usuario.®

Caso de Prueba de Aceptación	
Código Caso de Prueba: PI-6-1	Nombre Historia de Usuario: Definir plantilla del Administrador.
Nombre de la persona que realiza la prueba: ISMM.	
Descripción de la Prueba: Se define la plantilla del administrador, Hojas de Estilo CSS, Tipografía de letras, imágenes y texto	
Condiciones de Ejecución: Ejecutar la aplicación desde un navegador Web (se recomienda varios navegadores para comprobar la compatibilidad) y verificar que el sistema carga correctamente.	
Entrada / Pasos de ejecución: Abrir el navegador, poner la URL donde se encuentra alojado el Sistema, y autenticarse como administrador.	
Resultado Esperado: Se definió correctamente la plantilla del administrador.	
Evaluación de la Prueba: Satisfactoria	

Tabla 9 : Caso de Prueba de Aceptación definir plantilla del administrador.®

Caso de Prueba de Aceptación	
Código Caso de Prueba: PI-7-1	Nombre Historia de Usuario: Definición de los módulos funcionales del Portal.
Nombre de la persona que realiza la prueba: ISMM.	
Descripción de la Prueba: Se definen todos los módulos funcionales con los que contará el Portal-Intranet.	
Condiciones de Ejecución: Ejecutar la aplicación desde un navegador Web (se recomienda varios navegadores para comprobar la compatibilidad) y verificar que el sistema carga correctamente	
Entrada / Pasos de ejecución: El cliente debe acceder al Portal-Intranet por medio de un navegador, introduciendo la URL donde se encuentra alojado el sistema.	
Resultado Esperado: El portal ejecuta correctamente los módulos funcionales.	
Evaluación de la Prueba: Satisfactoria	

Tabla 10: Caso de Prueba de Aceptación Definición de los módulos funcionales del Portal.

Caso de Prueba de Aceptación	
Código Caso de Prueba: PI-8-1	Nombre Historia de Usuario: Implementación de los módulos funcionales del Portal.
Nombre de la persona que realiza la prueba: ISMM.	
Descripción de la Prueba: Se realizan varias pruebas al portal con el objetivo de comprobar las funcionalidades de los módulos implementados .	
Condiciones de Ejecución: Ejecutar la aplicación desde un navegador Web (se recomienda varios navegadores para comprobar la compatibilidad) y verificar que el sistema ejecuta correctamente todas sus funcionalidades.	
Entrada / Pasos de ejecución: El cliente debe acceder al Portal-Intranet por medio de un navegador, introduciendo la URL donde se encuentra alojado el sistema.	
Resultado Esperado: El portal ejecuta correctamente todas las acciones realizadas.	
Evaluación de la Prueba: Satisfactoria	

Tabla 11: Caso de Prueba de Aceptación Implementación de los módulos funcionales del Portal.

Caso de Prueba de Aceptación	
Código Caso de Prueba: SP-9-1	Nombre Historia de Usuario: Definición de los Niveles de usuarios dentro del Portal.
Nombre de la persona que realiza la prueba: ISMM.	
Descripción de la Prueba: Se definen todos los niveles de usuarios que tendrá el Portal-Intranet.	
Condiciones de Ejecución: El Usuario debe encontrarse dentro del Portal, y comprobar a que privilegios tiene derechos, estos pueden ser modificar información, acceder a distintas áreas, etc.	
Entrada / Pasos de ejecución: El usuario debe autenticarse en el Portal por medio de un navegador, poniendo la URL donde se encuentra alojado el sistema	
Resultado Esperado: El sistema debe permitir acceder solo a las áreas donde se le halla dado un previo autorizo.	
Evaluación de la Prueba: Satisfactoria	

Tabla 12: Caso de Prueba de Aceptación Definición de los Niveles de usuarios dentro del Portal.

Caso de Prueba de Aceptación	
Código Caso de Prueba: SP-10-1	Nombre Historia de Usuario: Implementación de los Niveles de usuarios dentro del Portal.
Nombre de la persona que realiza la prueba: ISMM.	
Descripción de la Prueba: Se realizan pruebas de acceso a distintas áreas a partir de distintos niveles de navegación.	
Condiciones de Ejecución: El Usuario debe encontrarse dentro del Portal, y comprobar a que privilegios tiene derechos, estos pueden ser modificar información, acceder a distintas áreas, etc.	
Entrada / Pasos de ejecución: El usuario debe autenticarse en el Portal por medio de un navegador, poniendo la URL donde se encuentra alojado el sistema	
Resultado Esperado: El sistema debe permitir acceder solo a las áreas donde se le halla dado un previo autorizo.	
Evaluación de la Prueba: Satisfactoria	

Tabla 13: Caso de Prueba de Aceptación Implementación de los Niveles de usuarios dentro del Portal.

Caso de Prueba de Aceptación	
Código Caso de Prueba: SP-11-1	Nombre Historia de Usuario: Definición de los menús de navegación en el Portal.
Nombre de la persona que realiza la prueba: ISMM.	
Descripción de la Prueba: Se definen todos los menús de navegación con los que contara el Portal-Intranet.	
Condiciones de Ejecución: El Usuario debe acceder al sitio por medio de un navegador y visualizar los menús que contenga el sitio.	
Entrada / Pasos de ejecución: El usuario debe autenticarse en el Portal por medio de un navegador, poniendo la URL donde se encuentra alojado el sistema	
Resultado Esperado: El sitio cuenta con todos los menús de navegación que se definieron.	
Evaluación de la Prueba: Satisfactoria	

Tabla 14: Caso de Prueba de Definición de los menús de navegación en el Portal.

Caso de Prueba de Aceptación	
Código Caso de Prueba: SP-12-1	Nombre Historia de Usuario: Implementación de los menús de navegación en el Portal.
Nombre de la persona que realiza la prueba: ISMM.	
Descripción de la Prueba: Se comprueba que el sitio cuenta con los menús necesarios y que estos siempre están disponibles en nuestro navegador, además de que conservan las propiedades previamente definidas.	
Condiciones de Ejecución: El Usuario debe encontrarse dentro del Portal, y verificar que puede hacer uso de los menús de navegación sin problema alguno.	
Entrada / Pasos de ejecución: El usuario debe autenticarse en el Portal por medio de un navegador, poniendo la URL donde se encuentra alojado el sistema	
Resultado Esperado: El sistema permite visualizar en todo momento los menús de navegación del Portal-Intranet.	
Evaluación de la Prueba: Satisfactoria	

Tabla 15: Caso de Prueba Implementación de los menús de navegación en el Portal.

Caso de Prueba de Aceptación	
Código Caso de Prueba: SP-13-1	Nombre Historia de Usuario: Definición del contenido de información.
Nombre de la persona que realiza la prueba: ISMM.	
Descripción de la Prueba: Se define el contenido que será difundido en el sitio. Además de como será visualizado este.	
Condiciones de Ejecución: El Usuario debe encontrarse dentro del Portal, y contar con la información que será visualizada.	
Entrada / Pasos de ejecución: El usuario debe autenticarse en el Portal por medio de un navegador, poniendo la URL donde se encuentra alojado el sistema y organizar el contenido según criterios comunes.	
Resultado Esperado: El sistema permite gestionar el contenido de información sin problemas.	
Evaluación de la Prueba: Satisfactoria	

Tabla 16: Caso de Prueba Definición del contenido de información.

Caso de Prueba de Aceptación	
Código Caso de Prueba: SP-14-1	Nombre Historia de Usuario: Implementación del contenido de información.
Nombre de la persona que realiza la prueba: ISMM.	
Descripción de la Prueba: Se comprueba que el sitio visualiza sin problemas el contenido almacenado en las bases de datos central.	
Condiciones de Ejecución: El Usuario debe encontrarse dentro del Portal, y comprobar que cuenta con las herramientas necesarias para la gestión del contenido.	
Entrada / Pasos de ejecución: El usuario debe autenticarse en el Portal por medio de un navegador, poniendo la URL donde se encuentra alojado el sistema y además comprobar que puede gestionar la información del sitio.	
Resultado Esperado: El sistema permite con facilidad gestionar la	

información del sitio.
Evaluación de la Prueba: Satisfactoria

Tabla 17: Caso de Prueba Implementación del contenido de información.

Caso de Prueba de Aceptación	
Código Caso de Prueba: SP-15-1	Nombre Historia de Usuario: Implementación de la funcionalidad personalización del Portal a nivel de Usuario.
Nombre de la persona que realiza la prueba: ISMM.	
Descripción de la Prueba: Se comprueba que la implementación de la funcionalidad de personalización del portal a nivel de usuario funciona correctamente.	
Condiciones de Ejecución: El Usuario debe encontrarse dentro del Portal.	
Entrada / Pasos de ejecución: El usuario debe autenticarse en el Portal por medio de un navegador, poniendo la URL donde se encuentra alojado el sistema, una vez dentro el usuario debe comprobar que puede personalizar su interfaz de usuario.	
Resultado Esperado: El sistema permite que el usuario pueda realizar cambios en su perfil.	
Evaluación de la Prueba: Satisfactoria	

Tabla 18: Caso de Prueba Implementación de la funcionalidad personalización del Portal a nivel de Usuario.

Conclusiones del capítulo

Con la culminación de este capítulo el cliente se asegura de que el producto desarrollado cumple con las funcionalidades para lo cual fue concebida al realizar una prueba para cada una de las HU implementada con la valoración de cada una de ellas. Las pruebas realizadas arrojaron resultados satisfactorios tanto para el cliente como para el equipo de desarrollo.

Capítulo 5: Estudio de Factibilidad

Introducción

Con el desarrollo de este capítulo se pretende ofrecer un análisis sobre el estudio de factibilidad del proyecto para lo cual se utilizó la técnica de Análisis de Costo - Beneficio la cual plantea que la conveniencia de la ejecución de un proyecto se determina por la observación conjunta de dos factores:

1. El costo, que involucra la implementación de la solución informática, adquisición y puesta en marcha del sistema hardware/software y los costos de operación asociados.
2. La efectividad, que se entiende como la capacidad del proyecto para satisfacer la necesidad, solucionar el problema o lograr el objetivo para el cual se ideó, es decir, un proyecto será más o menos efectivo con relación al mayor o menor cumplimiento que alcance en la finalidad para la cual fue ideado (costo por unidad de cumplimiento del objetivo).

La técnica de Análisis de Costo - Beneficio, tiene como objetivo fundamental proporcionar una medida de los costos en que se incurren en la realización de un proyecto informático, y a su vez comparar dichos costos previstos con los beneficios esperados de la realización de dicho proyecto.

1.1 Efectos Económicos

Los efectos económicos pueden clasificarse como:

- ✓ Efectos directos.
- ✓ Efectos indirectos
- ✓ Efectos externos
- ✓ Intangibles

1.2 Efectos directos

- ✓ Positivos
 - Se gestiona la información necesaria a la que los usuarios finales del sitio podrán acceder.

- Mayor integración usuario- artefactos, ya que por medio de este el usuario siente necesidades de interactuar con el portal, debido a que este facilita y le brinda la información necesaria.
- Se cuenta con una herramienta capaz de mantener la seguridad e integridad de la información difundida.
- Permite al usuario estar informado y organizado ante cualquier tarea tanto de ámbito nacional como institucional.
- Facilita a usuarios con determinadas roles difundir información por medio del portal.

✓ Negativos

- Para usar la aplicación es necesario la utilización de un ordenador conectado a la red, paralelo a los gastos de consumo de electricidad y mantenimiento que conlleva.

1.2.1 Efectos indirectos

- Los efectos económicos observados que pudiera repercutir sobre otros mercados no son perceptibles.

1.2.2 Externalidades

- Se contará con una herramienta que permitirá a los usuarios finales acceder a la información necesaria, además de acceder a servicios tales como el correo, SIGENU y demás, de una forma segura y rápida.

1.2.3 Intangibles

En la valoración económica siempre hay elementos perceptibles por una comunidad como perjuicio o beneficio, pero al momento de ponderar en unidades monetarias esto resulta difícil o prácticamente imposible. A fin de medir con precisión los efectos, deberán considerarse dos situaciones:

1.3 Beneficios y Costos Intangibles en el proyecto

Costos:

- ✓ Resistencia al cambio.

Beneficios:

- ✓ Mayor comodidad, organización e información para los usuarios.
- ✓ Mayor integración usuarios-artefactos.
- ✓ Mejora en la calidad y visibilidad de la información.

1.4 Ficha de Costo

Para determinar el costo económico del proyecto se utilizará el procedimiento para elaborar Una Ficha De Costo de un Producto Informático [Dra. Ana María Gracia Pérez, UCLV]. Para la elaboración de la ficha se consideran los siguientes elementos de costo, desglosados en moneda libremente convertible y moneda nacional.

Costos en Moneda Libremente Convertible:

- Costos Directos.
 1. Compra de equipos de cómputo: No procede.
 2. Alquiler de equipos de cómputo: No procede.
 3. Compra de licencia de Software: No procede.
 4. Depreciación de equipos: \$ 60.78.
 5. Materiales directos: No procede.

Total: \$ 60.78CUC
- Costos Indirectos.
 1. Formación del personal que elabora el proyecto: No procede.
 2. Gastos en llamadas telefónicas: No procede.
 3. Gastos para el mantenimiento del centro: No procede.

4. Know How: No procede.

5. Gastos en representación: No procede.

Total: \$0.00.

▪ Gastos de distribución y venta.

1. Participación en ferias o exposiciones: No procede.

2. Gastos en transportación: No procede.

3. Compra de materiales de propagandas: No procede.

Total: \$0.00.

Costos en Moneda Nacional:

▪ Costos Directos.

1. Salario del personal que laborará en el proyecto: \$100.00.

2. El 5% del total de gastos por salarios se dedica a la seguridad social: No procede.

3. El 0.09% de salario total, por concepto de vacaciones a acumular: No procede.

4. Gasto por consumo de energía eléctrica: \$ 5.94.

5. Gastos en llamadas telefónicas: No procede.

6. Gastos administrativos: No procede.

▪ Costos Indirectos.

1. Know How: \$ 108,75.

Total: \$ 214.69MN

Como se hizo referencia anteriormente, la técnica seleccionada para evaluar la factibilidad del proyecto es la Metodología Costo-Efectividad. Dentro de esta metodología la técnica de punto de equilibrio aplicable a proyectos donde los beneficios tangibles no son evidentes el análisis se basa exclusivamente en los costos. Para esta técnica es imprescindible definir una variable discreta que haga variar los costos. Teniendo en cuenta que el costo para este proyecto es despreciable, tómesese como costo el tiempo empleado por el Webmaster para mantener el sitio actualizado en todo momento:

Conclusiones del Capítulo

Este capítulo realizó el estudio de factibilidad mediante La Metodología Costo Efectividad (Beneficio), analizó los efectos económicos, los beneficios y costos intangibles; además, se calculó el costo de ejecución del proyecto mediante la ficha de costo arrojando como resultado \$ 60.78 CUC. y \$ 214.69 MN por lo que se demostró la factibilidad del proyecto.

Conclusiones Generales

Por medio de la culminación del proyecto se concluyó:

- ❖ Se pudo desarrollar una aplicación web dinámica capaz de gestionar la información necesaria de la que demanda la comunidad universitaria del ISMM.
- ❖ El desarrollo del proyecto permitió realizar una valoración sobre los conceptos principales asociados al objeto de estudio y campo de acción.
- ❖ Estuvo presente el cliente vinculado al área de producción, durante el período de planeación y desarrollo del software.
- ❖ Se realizó el análisis y diseño de la aplicación a partir de las características del territorio en el que se encuentra enmarcado la Institución y las necesidades de la comunidad.
- ❖ Se identificó cuáles eran los principales procesos que debía proveer el Portal-Intranet v2.0. Para ello se definieron los requerimientos funcionales y no funcionales del sistema los cuales se deben tener en cuenta en la construcción del mismo, así como las fases de exploración y planificación del sistema como parte de la metodología de desarrollo utilizada.
- ❖ Las pruebas de aceptación dieron una visión de la valoración del cliente sobre el trabajo desarrollado, este resultó ser satisfactorio ya que se cumplió con la totalidad de las Historias de Usuarios.
- ❖ Con el análisis del estudio de la factibilidad se demostró los beneficios que trajo la realización del proyecto para el Instituto al ser este de \$ 60.78 CUC. y \$ 214.69 MN.

Recomendaciones

- ❖ Se recomienda que el Portal-Intranet v2.0 en versiones posteriores incluya dentro del sitio el tema relacionado con la economía, ya que en la actualidad el sitio solo hace alusión al tema de control Interno.

- ❖ Incentivar el estudio de nuevas herramientas para proporcionarle mayor funcionalidad al Portal-Intranet.

- ❖ Ampliar la visión del Portal-Intranet.

Bibliográfica

Abrahamsson, P., Salo, O., Ronkainen, J., Warsta, J. Agile software development methods Review and analysis. VTT Publications. 2002.

Álvarez, M. ¿Qué es ASP? [en línea], 2007. [Consultado 25-01-10]. Disponible en: <http://www.DesarrolloWeb.com>

Álvarez, M. ¿Qué es JSP? [en línea], 2007. [Consultado 25-01-10]. Disponible en: <http://www.DesarrolloWeb.com>

Cambrá, Pedro, Los cinco principales errores al empezar con Drupal y su solución, [en línea], 21/02/2009 a las 13:19, [Consultado 12/02/2010]. disponible en: <http://cambrico.net>

Cuenca Aguilar, Yeidel, Aplicación Web para la contabilización del pago de estipendio estudiantil y la contabilidad presupuestada, Trabajo de Diploma, ISMM, 2006.

Flower, Martin, Texto original: [The New Methodology](#), Traducción: [Alejandro Sierra](#), marzo/abril de 2003. [Consultado 15/03/2010]. Disponible en: <http://newMethodology.es>

González Ramos, Dabiel, *GESTOR DOCENTE DE INFORMACION VIRTUAL*, Trabajo de Diploma, Cujae, 2007.

Jeffries, R., Anderson, A., Hendrickson, C. Extreme Programming Installed. Addison-Wesley. 2001

Lamber Santana, Liexna, *INTRANET DE LA EMPRESA RENE RAMOS LATOURT MODULO DE ECONOMIA ENERGETICA*, Trabajo de Diploma, ISMM, 2009.

Marin E. Lucian, Como Instalar y configurar PostgreSQL en Debian, [en línea], 2009, consultado [Consultado 23/02/2010]. disponible en: www.wordpress.com.

Martin, Cesar, 8 razones para usar Drupal, [en línea], 27-07-2009, Disponible en: www.alzado.org

Meza Martínez, Jorge Iván, personalizar formulario de búsqueda, [en línea], 13/08/2009- 3:57pm, [Consultado 23/02/2010]. disponible en <http://blog.jorgeivanmesa.com>.

Monografías “Arquitectura Cliente - Servidor”, 2009. Disponible en: <http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml>.

Núñez Torres, Edgar, *SISTEMA AUTOMATIZADO PARA LA GESTION DE LA INFORMACION DEL CONTROL INTERNO EN EL ISMM*, Trabajo de Diploma, ISMM, 2009.

Pérez Roldán, Dayron, Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informática, Trabajo de Diploma, UCI, 2008.

Poppendieck M., Poppendieck T. *Lean Software Development: An Agile Toolkit for Software Development Managers*. Addison Wesley. 2003.

Reyero Jose A, Características de Drupal Drupal Hispano, [en línea], Julio 17, 2005 - 21:38. , [Consultado 25-02-10]. Disponible en: www.drupal.org

Rincon Sanchez, Carlos, Drupal vs Joomla, [en línea], 2009, [Consultado 15/03/2010]. Disponible en: <http://carlos.rinconsanchez.com>

Rodríguez Corbea, M. *La Metodología XP Aplicable al Desarrollo del Software Educativo en Cuba*. Trabajo de Diploma. Universidad de las Ciencias Informáticas, UCI”, 2007

Romero Navarro, Mayelin, *Modulo de Administracion de la Intranet de Serconi*, Trabajo de Diploma, ISMM, 2007.

Sanchez, Ismael, Metodología SCRUM para la dirección de proyectos informáticos, Miércoles 10/Junio/2009 , [Consultado 15/03/2010]. disponible en: www.ejecucion.wordpress.com

Schwaber K., Beedle M., Martin R.C. *Agile Software Development with SCRUM*. Prentice Hall. 2001.

Vera, Alejandro, creando módulos de Drupal, [en línea], 21 /04/2008, [Consultado 23/02/2010]. Disponible en <http://www.almejo.cl>

Wiki Drupal [Consultado 23/2/2010]. Disponible en: <http://es.wikipedia.org/wiki/Drupal>

Wiki Zope. [Consultado 23/2/2010]. Disponible en: <http://es.wikipedia.org/wiki/Zope>

Glosario de Términos

Artefacto: En tecnología, es un dispositivo concebido y fabricado, sea de modo artesanal o industrial, por una o más personas.

Calidad: La palabra calidad tiene múltiples significados. La calidad de un producto o servicio es la percepción que el cliente tiene del mismo. Es una fijación mental del consumidor que asume conformidad con un producto o servicio determinado, que solo permanece hasta el punto de necesitar nuevas especificaciones. La calidad es un conjunto de propiedades inherentes a un objeto que le confieren capacidad para satisfacer necesidades implícitas o explícitas.

Ciclo de vida: Es un proceso por el cual los analistas de sistemas, los ingenieros de software, los programadores y los usuarios finales elaboran sistemas de información y aplicaciones informáticas.

Cliente: Persona, organización o grupo de personas que encargan la construcción de un producto software.

Desarrollo incremental: Forma de reducir la repetición del trabajo en el proceso de desarrollo y dar oportunidad de retrasar la toma de decisiones en los requisitos hasta adquirir experiencia con el sistema. Es una combinación del Modelo de Cascada y Modelo Evolutivo.

Estándares: Es una especificación que regula la realización de ciertos procesos o la fabricación de componentes para garantizar la interoperabilidad.

Herramientas: Son los ambientes de apoyo necesario para automatizar las prácticas de Ingeniería de Software.

Iteraciones: En el contexto de un proyecto se refieren a la técnica de desarrollar y entregar componentes incrementales de funcionalidades de un negocio. Una iteración resulta en uno o más paquetes atómicos y completos del trabajo del proyecto que pueda realizar alguna función tangible del negocio. Múltiples iteraciones contribuyen a crear un producto completamente integrado.

Metodología ágil: Nuevo enfoque metodológico orientado a la gente y los resultados.

Metodología de desarrollo: Es una versión amplia y detallada de un ciclo de vida COMPLETO de desarrollo de sistemas que incluye: Reglas, procedimientos, métodos, herramientas, funciones individuales y en grupo por cada tarea, productos resultantes, normas de Calidad.

Metodologías tradicionales: Metodologías basadas en procesos.

Procedimiento: Son los mecanismos de gestión que soportan a los métodos: El control de los proyectos, el control de la calidad.

Proceso: secuencia de actividades que tienen un marcado inicio y fin.

Proyecto de desarrollo: Elemento organizativo a través del cual se gestiona el desarrollo de software. El resultado de un proyecto es una versión de un producto.

Pruebas de aceptación: son las pruebas realizadas por el cliente para validar el software.

Requisitos: Capacidades, condiciones o cualidades que el sistema debe cumplir y tener.

Sprint: Equivale a una iteración, en la metodología Scrum.

Usuario: Persona encargada de utilizar el sistema, obteniendo algún beneficio.

Validación: no es más que verificar que un producto determinado cumple con los requisitos que fueron pactados con el cliente.

Libre: O Software Libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software.

Protocolo: Conjunto de normas que rigen un determinado proceso de comunicación.

Imagen: Grupo de configuraciones específicas que se extraen de una PC para distribuirlas en un grupo de PC.

TCP/IP: Transfer Control Protocol / Internet Protocol. Son los protocolos que se utilizan en Internet para transmitir datos. El TCP está orientado a la conexión que establece una línea de diálogo entre el emisor y el receptor antes de que se transfieran los datos. El IP trata cada paquete de forma independiente e incluye en la cabecera información adicional para así controlar la información. Estos protocolos garantizan que la comunicación entre dos aplicaciones es precisa.

PHP (PHP Hypertext Pre-processor): Lenguaje de programación para el desarrollo de webs dinámicas, con sintaxis parecida a la de C. Originalmente se conocía como Personal Home Page tools, herramientas para páginas personales (en Internet).

HTTP: Protocolo de transferencia de hipertexto.

FTP: Protocolo de transferencia de archivo.

Servidor: Computadora central de un sistema de red que provee servicios y recursos (programas, comunicaciones, archivos, etc.) a otras computadoras (clientes) conectadas a ella.

Herramienta CASE: Computer Aided Software Engineering, son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas ayudan en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores.

Apache: Servidor de páginas Web de código abierto para diferentes plataformas (UNIX, Windows, etc.). **CMS (Content Management System):** Sistema que facilita la gestión de contenidos en todos sus aspectos: creación, mantenimiento, publicación y presentación. También se conoce como Web Content Management (WCM) sistema de gestión de contenido de webs.

GPL (General Public License): Licencia que permite el uso y modificación del código para desarrollar software libre, pero no propietario.

Metadatos: Datos que describen otros datos. Son datos altamente estructurados que describen información, el contenido, la calidad, la condición y otras características de los datos.

WAI: Es una rama del World Wide Web Consortium que vela por la accesibilidad de la Web.