



PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR



**PONTIFICIA UNIVERSIDAD CATÓLICA DEL
ECUADOR**

**FACULTAD DE INGENIERIA
ESCUELA DE SISTEMAS**

**DISERTACIÓN PREVIA A LA OBTENCION DEL
TÍTULO DE
INGENIERO EN SISTEMAS Y COMPUTACION**

**“DESARROLLO DE UN SISTEMA DE GESTIÓN DE
DOCUMENTOS EMPRESARIAL CON
HERRAMIENTAS DE SOFTWARE LIBRE MEDIANTE
EL USO DE eZ PUBLISH”**

AUTORES

**JUAN FRANCISCO ACUÑA VIVANCO
FABIÁN EDUARDO VILLAVICENCIO PÁEZ**

**DIRECTOR: ING. FABIÁN DE LA CRUZ
QUITO, 2011**



AGRADECIMIENTO

Agradecemos a Dios por la vida, a la Universidad por la educación y a la familia y amigos por todos los buenos consejos, y todo el apoyo que nos han brindado en este largo camino.



DEDICATORIA

Este trabajo lo dedico a mis padres, a mis hermanos y a todos aquellos que de una u otra forma contribuyeron e hicieron posible la culminación de esta carrera con éxito.

Juan Francisco Acuña Vivanco

Para mi padre, el ingeniero de la vida, a mi madre, la mejor maestra por su constante apoyo y cariño y a Lorena y Mateo con mucho amor.

Fabián Eduardo Villavicencio Páez



CONTENIDO

CAPÍTULO 1	I
GESTIÓN DE CONTENIDOS	1
1.1 Introducción.....	1
1.2 Los sistemas de gestión de contenidos	2
1.2.1 Creación de contenido	5
1.2.2 Gestión de información y gestión de contenidos.....	6
1.2.3 Actividades y procesos de gestión de contenidos.....	7
1.2.4 Publicación.....	9
1.2.5 Presentación	10
1.3 Necesidad de un CMS	10
1.3.1 Inclusión de nuevas funcionalidades en el web	11
1.3.2 Mantenimiento de gran cantidad de páginas.....	11
1.3.3 Reutilización de objetos o componentes.....	12
1.3.4 Páginas interactivas	12
1.3.5 Cambios del aspecto de la web	12
1.3.6 Consistencia de la web	12
1.3.7 Control de acceso	13
1.4 Gestión de Documentos.....	13
1.4.1 Que es un sistema de gestión de documentos	13
1.4.2 Aspectos básicos de la gestión.....	14
1.4.3 Un poco de historia	16
1.5 Perspectivas de desarrollo	17
1.5.1 Requerimientos	17
1.5.2 Documentación	18
1.5.3 Objetivos y planificación.....	18
1.5.4 Web semántico	18
CAPITULO 2	19
SOFTWARE LIBRE	19
2.1. Introducción.....	19
2.2. Definición	20
2.3. Categorías de Software	23



2.3.1.	Software de fuente abierta	23
2.3.2.	Software de dominio publico	24
2.3.3.	Software protegido con copyleft	25
2.3.4.	Software libre no protegido con copyleft	26
2.3.5.	Sistema GNU	26
2.3.6.	Freeware	28
2.3.7.	Shareware	28
2.3.8.	Software Privativo	29
2.3.9.	Software Privado	29
2.3.10.	Software Comercial	29
2.4.	Licencias para Software	30
2.4.1.	Compatible con GPL	30
2.4.2.	Incompatible con GPL	34
2.4.3.	Licencias para Documentación	38
2.5.	Herramientas de software libre por su funcionalidad.	40
2.5.1.	Bases de datos	41
2.5.2.	Desarrollo Web	45
2.5.3.	Modelamiento	51
2.5.4.	Gráficos	53
2.5.5.	Internet	54
2.5.6.	Ofimática	57
2.6.	Ventajas, Desventajas y Problemática del software libre	60
2.6.1.	Ventajas	60
2.6.2.	Desventajas	63
2.7.	Problemática	65
2.7.1.	Compatibilidad con Hardware	65
2.7.2.	Centros de Soporte	65
CAPITULO 3		67
CASO DE ESTUDIO: EMPRESA DE TELECOMUNICACIONES		67
3.1.	Antecedentes	67
3.1.1.	Misión	68
3.1.2.	Visión	68



3.1.3.	Valores Empresariales	68
3.2.	Descripción del Problema	69
3.2.1.	Situación Actual	69
3.2.2.	Propuesta de Automatización	71
3.2.3.	Casos de Uso a Nivel General	72
3.3.	Herramientas a utilizar	72
3.3.1.	eZ Publish	72
3.3.2.	MySQL	78
3.3.3.	PHP.....	79
3.3.4.	PowerDesigner.....	79
3.3.5.	Dreamweaver.....	80
3.3.6.	Apache	80
CAPITULO 4.....		81
ANÁLISIS Y DISEÑO DEL SISTEMA		81
4.1.	Especificación de Requerimientos de Software (SRS)	81
4.2.	Diagramas de Casos de Uso	83
4.2.1.	Ingresar al Sistema	83
4.2.2.	Administrar Incidentes.....	84
4.3.	Diagrama de Actividades del Proceso	92
4.4.	Prototipo.....	93
4.4.1.	Opciones del Menú	93
4.4.2.	Pantallas	94
4.5.	Otros Requerimientos	96
4.5.1.	Interfaces Externas	96
4.5.2.	Hardware.....	98
4.5.3.	Software	99
CAPITULO 5.....		100
DESARROLLO E IMPLEMENTACION DEL SISTEMA		100
5.	Métodos de Instalación de eZ Publish	100
5.1.1.	Asistente de Instalación	101
5.2.	Integración	120
5.3.	Modelo Entidad Relación	123



5.4.	Diccionario de Datos.....	124
5.4.1.	Listado de Tablas del Sistema.....	125
5.4.2.	Tabla: RSRC_ROBOCABL.....	125
5.4.3.	Tabla: RSRC_MEDIRECE.....	126
5.4.4.	Tabla: RSRC_CABLES.....	127
5.4.5.	Tabla: RSRC_EMPL.....	127
5.4.6.	Tabla: RSRC_TIPCAB.....	128
5.4.7.	Tabla: RSRC_BANCO.....	128
5.4.8.	Tabla: RSRC_AREA.....	128
5.5.	Codificación.....	128
5.5.1.	Registro y Seguimiento de Incidentes de Robo de Cable.....	129
5.5.2.	eZ Publish.....	133
CAPITULO 6.....		137
CONCLUSIONES Y RECOMENDACIONES.....		137
6.1.	Conclusiones.....	137
6.2.	Recomendaciones.....	140

FIGURAS

Figura 3.1:	Caso de Uso a Nivel General.....	72
Figura 3.2:	Estructura Interna eZ Publish.....	77
Figura 4.1:	Caso de Uso Administrar Incidentes.....	85
Figura 4.2:	Caso de Uso Ingresar Incidentes.....	86
Figura 4.3:	Caso de Uso Actualizar Incidentes.....	88
Figura 4.4:	Caso de Uso Eliminar Incidentes.....	89
Figura 4.5:	Caso de Uso Consultar Incidentes.....	90
Figura 4.6:	Diagrama de Actividades.....	92
Figura 4.7:	Opciones del Menú Incidentes.....	93
Figura 4.8:	Opciones del Menú Mantenimiento.....	93
Figura 4.9:	Prototipo de pantalla para el usuario de Call Center.....	94
Figura 4.10:	Prototipo de pantalla para el usuario de Seguridad Integral.....	94



Figura 4.11: Prototipo de pantalla para el Usuario de Operaciones.....	95
Figura 4.12: Prototipo de pantalla para el Usuario de Seguros.	95
Figura 4.13: Caso de Uso Envío de SMS	96
Figura 4.14: Caso de Uso Envío de Correos Electrónicos.....	97
Figura 5.1: Página de Bienvenida.....	103
Figura 5.2: Ajuste del Sistema	104
Figura 5.3: Chequeo del Sistema.....	105
Figura 5.4: Correo Saliente	106
Figura 5.5: Base de Datos	107
Figura 5.6: Inicialización de la Base de Datos	108
Figura 5.7: Soporte del Idioma.....	109
Figura 5.8: Selección del Sitio.....	110
Figura 5.9: Lista de paquetes instalados	111
Figura 5.10: Configuración de acceso al sitio.	113
Figura 5.11: Detalles del sitio.....	114
Figura 5.12: Administración del sitio.	116
Figura 5.13: Seguridad del sitio.	117
Figura 5.14: Registro del sitio.	118
Figura 5.15: Fin de la instalación.	119
Figura 5.16: Relación Objeto Contenido - Nodos.....	120
Figura 5.17: Árbol de Nodos.	121
Figura 5.18: Secciones.	121
Figura 5.19: Diagrama Integración Intranet - Sistema.	122
Figura 5.20: Modelo Entidad - Relación.....	123
Figura 5.21: Conexión a la Base de Datos.	129
Figura 5.22: Clases.....	130
Figura 5.23: Métodos.....	130
Figura 5.24: Servicios.	131
Figura 5.25: Interfaz para el usuario.	132
Figura 5.26: Datos presentados sobre el registro del Incidente.....	132
Figura 5.27: Datos sobre el incidente.	133
Figura 5.28: Pagina Principal.....	134



Figura 5.29: Apariencia Menú..... 135
Figura 5.30: Enlaces del Menú 136



CAPÍTULO 1

GESTIÓN DE CONTENIDOS

En este capítulo trataremos de dar a conocer el concepto tanto de la gestión de contenidos, la necesidad de éstos como en la gestión de documentos, los aspectos, y las perspectivas de desarrollo.

1.1 Introducción

Realizar un Web puede ser un trabajo complicado y muy laborioso si no se dispone de las herramientas adecuadas. En el pasado las herramientas eran básicamente editores que permitían generar una página, que evolucionaron para incorporar el control de la estructura de la Web y otras funcionalidades, pero en general estaban enfocadas más a la creación que al mantenimiento. En los últimos años se ha desarrollado el concepto de sistema de gestión de contenidos (*Content Management Systems* o *CMS*). Se trata de herramientas que permiten crear y mantener un Web con facilidad, encargándose de los trabajos más tediosos que hasta ahora ocupaban el tiempo de los administradores de las Web.

Teniendo en cuenta el ahorro de tiempo de desarrollo que supone la utilización de estas herramientas, sería lógico esperar que su precio fuera muy elevado. Eso es cierto para algunos productos comerciales, pero existen potentes herramientas de gestión de contenidos de acceso libre, disponibles con licencias de código abierto.



Los gestores de contenidos proporcionan un entorno que posibilita la actualización, mantenimiento y ampliación de la Web con la colaboración de múltiples usuarios. En cualquier entorno virtual ésta es una característica importante, que además puede ayudar a crear una comunidad cohesionada que participe más de forma conjunta.

El sistema de gestión de contenidos es un software que se utiliza principalmente para facilitar la gestión de Web, ya sea en Internet o en una intranet, y por eso también son conocidos como gestores de contenido Web (*Web Content Management* o *WCM*). Hay que tener en cuenta, sin embargo, que la aplicación de los CMS no se limita sólo a la Web.

1.2 Los sistemas de gestión de contenidos

Los servicios de información y documentación accesibles a través de Internet, más concretamente mediante servidores Web, están aumentando de una forma exponencial. La lógica evolución del Web desde hace más de 10 años ha ido produciendo la sustitución de páginas y documentos estáticos por documentos generados dinámicamente, merced a la interacción del usuario con la lógica de procesos y flujos de trabajo definida por los creadores del servicio y a la disponibilidad de cada vez mayores repositorios de información.

Evidentemente, se ha ido pasando progresivamente de un concepto de publicación de páginas Web, bastante simple en su origen, a esquemas



más complejos y diferenciados, fundamentados en procedimientos y técnicas basados en la gestión de información. La cada vez mayor complejidad de los servicios y de los sistemas que los soportan, ha hecho necesaria la formulación de un cuerpo teórico y práctico en el que se combinen las técnicas clásicas de gestión de información en las organizaciones con las características propias del medio ambiente digital.

Esta evolución, (que se ha acelerado durante la primera mitad de la década del 2000), ha tenido un impacto no sólo en los métodos y técnicas de gestión de información, sino también en la propia tecnología para gestión de información y, en consecuencia, en el mercado de productos y servicios. Si bien en la segunda mitad de la década de los 90 se podía diferenciar entre productos para gestión documental, para recuperación de información, entre otros desde el año 2000 se ha producido una convergencia entre todas las plataformas, de forma que en la actualidad se pueden encontrar soluciones que pretenden ser globales y ofrecer soporte a todo el proceso de gestión de información en una organización. Las herramientas para este trabajo han recibido la denominación de sistemas de gestión de contenidos (*CMS*), y se han integrado con los sistemas de gestión documental y con los de recuperación de información. A ello hay que unir que, en la concepción actual de la gestión de información, el control de los procesos es un elemento nuclear, por lo que se acompañan de sistemas de flujos de trabajo (*workflow*).



Con todo ello se puede delinear un paisaje en el cual las herramientas de gestión documental han ido incorporando a sus prestaciones las capacidades necesarias para gestionar los procesos que crean, almacenan, tratan y presentan información, en entorno digital.

Sin embargo, no por ello cabe afirmar que existe una igualdad directa entre los sistemas de gestión de contenidos y los sistemas de gestión documental. Se pueden encontrar en el mercado sistemas de gestión de contenidos que no ofrecen las prestaciones documentales que serían deseables, y viceversa. De la misma forma, no es posible igualar mediante una ecuación gestión de contenidos y gestión documental. Íntimamente relacionadas sí, pero no iguales. Sólo el estudio y la evaluación de las características y prestaciones presentes en las herramientas disponibles pueden determinar la adecuación de una solución en un contexto o problema dado. A ello cabe añadir las diferencias existentes entre la gestión de contenidos para Web y la gestión de contenidos para empresas, y que pueden encontrarse en los informes o documentos especializados que publican las propias compañías del sector. Si a esto se suma la complejidad creciente de muchos portales, tanto internos como externos, de las organizaciones, que son soportados por sistemas de gestión de contenidos, y a los procesos de publicación digital necesarios para su producción, los sistemas de gestión de contenidos se configuran como aplicaciones de varias escalas, que pueden llegar a alcanzar una alta complejidad.



En la parte humana, las herramientas para gestión de contenidos sólo se pueden entender en un entorno de trabajo en colaboración y distribuido, ya que es en este tipo de ambientes donde se puede aprovechar todo su potencial. Frente a enfoques tradicionales de tratamiento documental, pasivos, las herramientas actuales han llevado más allá los límites del tratamiento documental, ya que se han extendido hasta el proceso de creación, por su parte inicial, y al proceso de nueva publicación y de personalización, por su parte final. Esto configura ahora un esquema circular e iterativo para el tratamiento documental, a lo que se debe unir, entonces, la necesaria reflexión sobre el concepto y características de los documentos digitales. El ciclo de vida de los documentos digitales muestra significativos cambios sobre el ciclo de vida tradicional, al igual que el concepto de documento. Evidentemente, las actividades informativo/documentales deben reformularse para hacer frente al nuevo medio ambiente digital.

Una división de la funcionalidad de los sistemas de gestión de contenidos en cuatro categorías: creación de contenido, gestión de contenido, publicación y presentación.

1.2.1 Creación de contenido

Un CMS aporta herramientas para que los creadores sin conocimientos técnicos en páginas Web puedan concentrarse en el contenido.



Lo más habitual es proporcionar un editor de texto *WYSIWYG* (*What You See Is What You Get*), en el que el usuario ve el resultado final mientras escribe, al estilo de los editores comerciales, pero con un rango de formatos de texto limitado. Esta limitación tiene sentido, ya que el objetivo es que el creador pueda poner énfasis en algunos puntos, pero sin modificar mucho el estilo general del sitio Web. Hay otras herramientas como la edición de los documentos en XML, utilización de aplicaciones ofimáticas con las que se integra el *CMS*, importación de documentos existentes y editores que permiten añadir marcas, habitualmente HTML, para indicar el formato y estructura de un documento.

Un *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. Para la creación del sitio propiamente dicho, los *CMS* aportan herramientas para definir la estructura, el formato de las páginas, el aspecto visual, uso de patrones, y un sistema modular que permite incluir funciones no previstas originalmente.

1.2.2 Gestión de información y 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, cómo son los datos relativos a los documentos (versiones hechas, autor, fecha de publicación, caducidad), datos y preferencias de los usuarios.



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 imprescindible para facilitar el ciclo de trabajo con un circuito de edición que va desde el autor hasta el responsable final de la publicación. El CMS permite la comunicación entre los miembros del grupo y hace un seguimiento del estado de cada paso del ciclo de trabajo.

1.2.3 Actividades y procesos de gestión de contenidos

La gestión de contenidos debe definirse desde la perspectiva de su objetivo y actividades. Frente a la gestión de documentos, que tiene como objeto de trabajo documentos, como su propio nombre indica, la gestión de contenidos está orientada a gestionar objetos que actúan como componentes de documentos virtuales, en el contexto de lo que llama segmentación. Los objetos son tratados mediante un conjunto de procesos estructurados con la finalidad de producir publicaciones digitales. Este objetivo de la producción de publicaciones y documentos digitales por integración de componentes se puede encontrar métodos y técnicas directamente relacionadas con el desarrollo de la sociedad de la información y del conocimiento, las formulaciones teóricas han seguido a la necesidad y a su resolución tecnológica, o han discurrido en paralelo. En este sentido, la gestión de contenidos se ha beneficiado de las técnicas y procesos de producción y edición de publicaciones digitales en un primer momento.



Sin embargo, rápidamente pudo apreciarse que el proceso de publicación digital no cubría por completo todos los procesos de gestión de información necesarios para una adecuada gestión de contenidos.

El proceso de publicación digital más sencillo define un proceso en el cual se diseña un marco de integración para un conjunto de objetos, cuyo resultado final es la publicación del documento digital. Evidentemente, este proceso, que puede reflejar el esquema más simple de funcionamiento de un *weblog*, se complica si se sitúa en el contexto de una unidad o sistema de información que debe crear información digital más compleja, y que, además, debe servir a comunidades de usuarios especializados, o a diferentes comunidades de usuarios. Los productos de información digital se vuelven más complejos, y los procesos técnicos que se desarrollan sobre ellos se hacen igualmente más complejos. El trabajo en colaboración obliga a desarrollar flujos de trabajo en los que es necesario asignar diferentes roles a las personas que intervienen en el mismo. Los objetos que componen los documentos pueden proceder de diferentes fuentes y tener diferentes formatos, con lo que es necesario controlar repositorios de información. Además, los usuarios demandan servicios de apoyo a los productos de información digital como la búsqueda de información o el acceso a documentos generados anteriormente, lo que hace necesario disponer de herramientas de archivo digital.



Los cada vez más complicados procesos de gestión de contenidos demandan la formalización de sistemas de metadatos que resultan necesarios tanto para los procesos de gestión como para los de publicación o los de recuperación de información.

Esta complejidad ha producido el desarrollo de dos especializaciones principales de la gestión de contenidos: *Web Content Management (WCM)*, y *Enterprise Content Management (ECM)*. La primera especialización se centra en el entorno Web, y su objetivo principal y sus métodos se orientan a la producción de documentos e información digitales para la Web, especialmente para portales y Web corporativos. La segunda encuentra su fundamento en la idea de gestión total de la información en las organizaciones mediante la integración de toda la información necesaria para alcanzar los objetivos de la organización proveniente.

1.2.4 Publicación

Una página aprobada se publica automáticamente cuando llega la fecha de publicación, y cuando caduca se archiva para futuras referencias. En su publicación se aplica el patrón definido para 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 preocuparse por el diseño final de sus páginas.



1.2.5 Presentación

Un CMS puede gestionar automáticamente la accesibilidad de la Web, con soporte de normas internacionales de accesibilidad, y adaptarse a las preferencias o necesidades de cada usuario. También puede proporcionar compatibilidad con los diferentes navegadores disponibles en todas las plataformas y su capacidad de internacionalización lo 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 noticias se presentarían las novedades aparecidas en otra Web, 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.

1.3 Necesidad de un CMS

Se han presentado bastantes motivos para ver la utilidad de un sistema que gestione un entorno web, pero se podría pensar que no es necesario para uno relativamente pequeño o cuando no se necesitan tantas funcionalidades. Eso sólo podría ser cierto para un web con unas pocas páginas estáticas para el que no se prevea un crecimiento futuro ni muchas actualizaciones, lo que no es muy realista.



En cualquier otro caso, la flexibilidad y escalabilidad que permiten estos sistemas, justifican su utilización en prácticamente cualquier tipo de web.

Muchos usuarios particulares utilizan *CMS* gratuitos para elaborar y gestionar sus Webs personales, obteniendo Webs dinámicos llenos de funcionalidades. El resultado que obtienen es superior al de algunas empresas que se limitan a tener páginas estáticas que no aportan ningún valor añadido.

Estos son algunos de los puntos más importantes que hacen útil y necesaria la utilización de un *CMS*:

1.3.1 Inclusión de nuevas funcionalidades en el web

Esta operación puede implicar la revisión de multitud de páginas y la generación del código que aporta las funcionalidades. Con un *CMS* eso puede ser tan simple como incluir un módulo realizado por terceros, sin que eso suponga muchos cambios en la web. El sistema puede crecer y adaptarse a las necesidades futuras.

1.3.2 Mantenimiento de gran cantidad de páginas

En una Web con muchas páginas hace falta un sistema para distribuir los trabajos de creación, edición y mantenimiento con permisos de acceso a las diferentes áreas. También se tienen que gestionar los metadatos de cada documento, las versiones, la publicación y caducidad de páginas y los enlaces rotos, entre otros aspectos.



1.3.3 Reutilización de objetos o componentes

Un CMS permite la recuperación y reutilización de páginas, documentos, y en general de cualquier objeto publicado o almacenado.

1.3.4 Páginas interactivas

Las páginas estáticas llegan al usuario exactamente como están almacenadas en el servidor web. En cambio, las páginas dinámicas no existen en el servidor tal como se reciben en los navegadores, sino que se generan según las peticiones de los usuarios. De esta manera cuando por ejemplo se utiliza un buscador, el sistema genera una página con los resultados que no existían antes de la petición. Para conseguir esta interacción, los CMS conectan con una base de datos que hace de repositorio central de todos los datos de la web.

1.3.5 Cambios del aspecto de la web

Si no hay una buena separación entre contenido y presentación, un cambio de diseño puede comportar la revisión de muchas páginas para su adaptación. Los CMS facilitan los cambios con la utilización, por ejemplo, del estándar CSS (Cascading Style Sheets u hojas de estilo en cascada) con lo que se consigue la independencia de presentación y contenido.

1.3.6 Consistencia de la web

La consistencia en un web no quiere decir que todas las páginas sean iguales, sino que hay un orden (visual) en vez de caos. Un usuario nota enseguida cuándo una página no es igual que el resto de las de la misma web por su aspecto, la disposición de los objetos o por los cambios en la



forma de navegar. Estas diferencias provocan sensación de desorden y dan a entender que el web no lo han diseñado profesionales. Los CMS pueden aplicar un mismo estilo en todas las páginas con el mencionado CSS, y aplicar una misma estructura mediante patrones de páginas.

1.3.7 Control de acceso

Controlar el acceso a un web no consiste simplemente al permitir la entrada a la web, sino que comporta gestionar los diferentes permisos a cada área de la web aplicados a grupos o individuos.

1.4 Gestión de Documentos

Un sistema de Gestión de Documentos (DMS, por sus siglas en inglés) son todos aquellos programas creados para la gestión de grandes cantidades de documentos. Un ejemplo de un software de gestión de documentos, aunque muy rudimentario es el “explorador de Windows”.

1.4.1 Que es un sistema de gestión de documentos

Un sistema de gestión de documentos es un conjunto de programas, utilizado para rastrear y almacenar documentos electrónicos y/o imágenes digitales de documentos originalmente soportados en papel. El término puede ser relacionado con conceptos como sistemas de administración de contenido (CMS) y es comúnmente visto como un sistema de administración de contenido corporativo y relacionado con el término (DAM) Digital Asset Management que puede traducirse como Gestión de Recursos Digitales.



1.4.2 Aspectos básicos de la gestión

Hay varias cuestiones comunes vinculadas a la gestión de documentos, tanto si es un sistema informal, un método basado en papel para una persona, como si es formal, estructurado, elaborado por computadora para muchas personas de muchas oficinas.

La mayoría de los métodos de gestión de documentos tienen las siguientes áreas:

- **Ubicación**

¿Dónde se archivarán los documentos? ¿Hacia dónde necesitará ir la gente para acceder a los documentos? Las caminatas a los depósitos de documentos o archiveros vendrían a ser los homólogos de los sistemas de navegación "en pantalla" utilizados por los sistemas de Gestión de Contenidos.

- **Llenado**

¿Cómo serán llenados los documentos? ¿Qué métodos serán utilizados para organizar o indexar los documentos y permitir su acceso posterior? Los sistemas de Gestión de Contenidos utilizarán típicamente una base de datos para archivar la información de llenado.

- **Recuperación**

¿Cómo serán encontrados los documentos? En general la recuperación se relaciona con la navegación a través de los documentos y la recuperación de cierta información específica.



- **Seguridad**

¿Cómo serán protegidos los documentos? ¿Cómo el personal no autorizado será imposibilitado de ver, modificar o destruir los documentos?

- **Recuperación tras desastres**

¿Cómo pueden los documentos ser recuperados en caso de destrucción por incendios, inundaciones o desastres naturales?

- **Período de retención**

¿Cuánto tiempo deberían los documentos ser guardados? Como las organizaciones crecen y las regulaciones se incrementan, lineamientos informales para guardar varios tipos de documentos dejan lugar a prácticas más formales para el registro de documentos.

- **Archivo**

¿Cómo pueden preservarse los documentos para su legibilidad en el futuro? Ya sea con copias o mediante papel se los podría preservar.

- **Distribución**

¿Cómo pueden los documentos estar disponibles para la gente que los necesita.

- **Flujo de trabajo**

Si los documentos necesitan pasar de una persona a otra, ¿cuáles son las reglas respecto a cómo su trabajo debería fluir.



- **Creación**

¿Cómo se crean los documentos? Esta pregunta se vuelve más importante cuando muchas personas necesitan colaborar, y surge la logística del control de la versión y la autoría.

- **Autenticación**

¿Hay alguna forma de asegurar la autenticidad de un documento? La autenticación es en parte importante para la seguridad de los documentos.

1.4.3 Un poco de historia

Comenzando en la década de los 80, un gran número de fabricantes comenzó a desarrollar sistemas para gestionar documentos de papel, lo cual incluía no sólo documentos impresos y publicados, sino también fotografías.

Más tarde, un segundo sistema fue desarrollado para gestionar documentos electrónicos; por ejemplo, todos estos documentos, o archivos, creados en computadoras y frecuentemente archivados en la computadora local del usuario. Los primeros sistemas de gestión de documentos electrónicos fueron desarrollados para gestionar archivos de tipo propietario, o un número limitado de formatos de documentos.

Muchos de estos sistemas fueron más tarde conocidos como sistemas de creación de imágenes o copias, porque sus principales habilidades eran



capturar, almacenar, indexar y recuperar archivos en formatos de imagen. Estos sistemas permitían a una organización capturar faxes y formularios, guardar copias de los documentos como imágenes, y guardar los archivos de imagen en un disco por seguridad y rapidez.

Los sistemas de gestión de documentos evolucionaron hasta que los sistemas fueron capaces de manejar cualquier tipo de formato que pudiera ser guardado en una red. Las aplicaciones crecieron y maduraron conforme fueron creciendo y madurando los documentos electrónicos, las herramientas de colaboración, seguridad y funcionalidades de auditoría.

1.5 Perspectivas de desarrollo

La gestión de contenidos y datos, y las herramientas de software libre que facilitan su implementación, son una realidad que ni puede ni debe pasar desapercibida, ya que brindan soluciones para los servicios que debe ofrecer, en el entorno digital, una unidad de información. Existen soluciones de tipo general y soluciones especializadas. Además, es innegable la existencia de un creciente mercado de productos y soporte para estas herramientas. Sin embargo, y a modo de conclusión, es necesario delinear varios factores críticos que deben tenerse en consideración:

1.5.1 Requerimientos

Si bien en el momento actual unos conocimientos técnicos de nivel medio son suficientes para poner en marcha un servicio con estas herramientas,



en un futuro cercano aumentará el requerimiento técnico, sobre todo debido al creciente volumen de información, a la complejidad de los documentos digitales y a la exigencia de niveles de seguridad.

1.5.2 Documentación

El soporte profesional para estas soluciones es un costo oculto que debe ser valorado. La documentación de administración y de usuario final todavía no alcanza, en muchos casos, el nivel de exigencia que sería deseable.

1.5.3 Objetivos y planificación

A pesar de lo que pueda parecer, todavía se siguen montando servicios de información en Internet en los que los objetivos y una correcta planificación son difuminados por el espejismo tecnológico. Sólo un correcto análisis y planificación podrán ayudar a determinar la elección de la herramienta más adecuada.

1.5.4 Web semántico

A pesar del desarrollo del XML y de sus lenguajes derivados, y de los esquemas de descripción de recursos de información y de metadatos, son todavía escasas las soluciones para gestión de contenidos que trabajan directamente con ellos. En un entorno digital enfrentado al crecimiento exponencial de la información, este es un problema al que hacer frente con celeridad.



CAPITULO 2

SOFTWARE LIBRE

En este capítulo se analizará tanto las categorías de software así como sus libertades, el sistema GNU, las clases de software libre según su funcionalidad. Además se verán las ventajas, desventajas y problemática del software libre.

2.1. Introducción

Ante la realidad del modelo económico del software propietario y el costo muy alto que representa para las empresas dentro de su presupuesto, por las licencias para cumplir los requerimientos legales, es importante analizar otros modelos desde un punto de vista económico y tecnológico.

Es así que existe una alternativa como el software libre, que gracias a varias razones tanto económicas, políticas y técnicas, ha venido haciéndose más factible gracias a que existen comunidades de usuarios y programadores a nivel mundial, respaldados por grandes empresas, que juntos han desarrollado herramientas bajo el concepto de libre, entendiéndose así, que una vez obtenido el código fuente de la aplicación, puede ser usado, copiado, estudiado, modificado y redistribuido bajo ciertas reglas y requisitos.



Estas libertades se otorgan tanto a nivel legal como técnico, y por tanto, un prerrequisito evidente es disponer del código fuente.

Pero esto no es suficiente; la licencia debe expresar claramente que todo esto está permitido, como así lo hace, por ejemplo, la Licencia Pública General(GPL).

Para el análisis de este modelo de software libre, es necesario considerar que varias empresas han aplicado modelos de calidad para automatizar sus procesos, y por lo tanto, la aplicación de software libre debe basarse inicialmente en el sistema aplicado en la empresa para la interacción de sus procesos.

Es por todo esto que las empresas tanto privadas como del gobierno están adoptando y capacitando a sus empleados en el uso de herramientas de software libre, pues solo si se logra que los usuarios se familiaricen con las herramientas, se podrá hablar de una implementación exitosa, para lo cual se demostrará la calidad y seguridad de las herramientas de software libre ya sea a nivel de escritorio como a nivel de un sistema operativo multiusuario.

2.2. Definición

Se refiere a la libertad, que la licencia otorga a los usuarios de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software, estas propiedades



se otorgan a nivel técnico y legal y por tanto un requisito fundamental es disponer del código fuente, más precisamente, se refiere a las cuatro libertades de los usuarios de software:

- **Libertad 0**

La libertad de ejecutar el programa, con cualquier propósito sea educativo social, tecnológico, entre otros.

- **Libertad 1**

La libertad de estudiar cómo funciona el programa, y adaptarlo a sus necesidades, y así conocer su funcionamiento, el conocimiento del código fuente es una condición, lo que nos permitirá descubrir las posibilidades de realizar determinadas tareas como agregar funciones o eliminarlas.

- **Libertad 2**

La libertad de distribuir copias de manera que se puede ayudar a los demás, pues se puede redistribuir sea de manera gratuita o con algún costo en algún medio de almacenamiento como un CD, DVD, o ponerlo a disposición de descarga mediante http, https o ftp.

- **Libertad 3**

La libertad de mejorar el programa, y liberar las mejoras al público de tal manera que pueda redistribuir ese software para que toda la comunidad se beneficie, el conocimiento del código fuente es una condición para esto.



Se denomina software libre a cualquier programa que cumpla estas libertades, así se puede redistribuir copias con o sin modificaciones generando ingresos, o hacerlo de forma gratuita, sin pedir permiso ni pagar para su distribución, la accesibilidad del código fuente es necesaria para cumplir estos requisitos, es una condición fundamental para el software libre.

Para que estas libertades se hagan efecto estas deben ser irrevocables, a no ser el caso que se cometan errores, pero si el desarrollador del software decide revocar la licencia entonces dejaría de ser software libre.

Sin embargo existen reglas sobre la distribución de software que sin entrar en conflicto con las libertades centrales son aceptables pues como por ejemplo la copyleft, permite que al aplicar esta norma no se añadan restricciones que nieguen al resto sus libertades centrales, lo cual nos da a entender que más bien protege las libertades del software libre.

El software libre no significa que sea “no comercial”. Cualquier programa libre estará disponible para su uso, desarrollo y distribución comercial.

El desarrollo comercial del software libre ha dejado de ser excepcional y de hecho ese software libre comercial es muy importante.



2.3. Categorías de Software

2.3.1. Software de fuente abierta

El término free dentro del concepto de software libre, puede ser interpretado como Software Libre o Software Gratis. El término gratis asustaba a las grandes empresas que de acuerdo a su ideología buscaban su expansión económica, y es así que nace el movimiento de Software de Fuente Abierta, que intenta ser un plan de mercadeo para el software libre, tomando sus características pero dejando a un lado el aspecto ideológico del software libre y así presentar modelos de negocio a las empresas.

Este modelo no ha creado una licencia específica, pero todo tipo de software que se denomine como Software de Fuente Abierta debe cumplir los siguientes criterios:

- Disponibilidad del código fuente
- Libre distribución
- Debe permitir modificaciones y trabajos derivados
- Integridad del autor del código fuente
- No a la discriminación en contra de personas o grupos
- No a la discriminación de uso en ningún campo o disciplina
- Distribución de la Licencia pues debe aplicarse a todo el software
- La licencia no debe ser específica a algún producto



- La licencia no debe restringir a otro software
- La licencia no debe estar atada a ninguna tecnología en especial.

El movimiento otorga una certificación a todo software que cumpla con estos criterios. Es además más flexible que el Software libre en el sentido que los usuarios de Software de Fuente Abierta consideran que pueden interactuar con software propietario.

La Free Software Foundation (FSF) opina que el movimiento de fuente abierta es filosóficamente diferente al de software libre, pues hace especial énfasis en los aspectos morales o éticos del software, viendo la excelencia técnica como un producto secundario deseable de su estándar ético. El movimiento Open Source ve la excelencia técnica como el objetivo prioritario, siendo la libre distribución del código fuente un medio para dicho fin. Por dicho motivo, la FSF se distancia tanto del movimiento Open Source como del término "Open Source".

2.3.2. Software de dominio publico

Es aquel software por el cual no se pide ningún tipo de licencia, es un software que no está protegido con copyright, esto quiere decir que sus derechos son para toda la humanidad, cualquiera puede hacer uso de este software siempre y cuando reconozca su autoría original.



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.

El derecho de autor es un conjunto de normas y principios que regulan los derechos morales y patrimoniales que la ley concede a los autores. El copyright comprende la parte patrimonial de los derechos de autor.

2.3.3. Software protegido con copyleft

Copyleft permite redistribuir el software, con o sin cambios, dando la libertad de copiarlo y modificarlo y redistribuir versiones modificadas y así garantiza que cada usuario tiene esa libertad, pero sin agregar restricciones propias.

El software libre protegido con copyleft, garantiza también que cada copia del software, aun si haya sido modificado, debe ser software libre y así dar a los usuarios las libertades que la filosofía del software libre implica.

El copyright lo usan los autores para restringir a los usuarios y quitarles libertades, en el movimiento del software libre se usa estos derechos reservados para garantizar estas libertades al usuario, y es por esto que el termino copyleft se puede considerarse opuesto al termino copyright en el aspecto legal.



El copyleft se convirtió en el tema clave a tratar en el conflicto ideológico entre el movimiento del código abierto y el movimiento del software libre.

2.3.4. Software libre no protegido con copyleft

El software libre no protegido con Copyleft proviene de autores que otorgan permisos para redistribuirlo y modificar así como para añadirle restricciones adicionales en estas redistribuciones.

Si un programa es libre pero no protegido con Copyleft, algunas copias o versiones modificadas pueden no ser libres completamente.

Una compañía de software puede incorporar cambios y mejoras a un software libre y distribuir esta nueva versión como un producto privativo de software.

2.3.5. Sistema GNU

Es un sistema desarrollado por el Proyecto GNU (GNU is NOT UNIX) desde 1984, que pretende ser por completo software libre, y desarrollar no solo el núcleo, sino todos los componentes del sistema operativo.

La primera aplicación para el sistema GNU fue un editor de texto llamado EMACS, que Stallman lo distribuía por Internet a través de FTP y también enviaba copias en cinta con lo cual los fondos del proyecto provenían de este negocio, luego vendrían más aplicaciones como la biblioteca C de GNU, el Shell Bourne-Again (bash), el tar GNU, el compilador C, y también se



aprovecharon otras aplicaciones libres presentes en el mercado como el formateador de textos TeX, o el sistema de ventanas X-Windows.

A principios de la década del 90 el Sistema GNU estaba casi completo a excepción del núcleo que fue desarrollado a partir de TRIX, que fue distribuido libremente por sus creadores en el MIT (Massachusetts Institute of Technology), que es totalmente compatible con la versión UNIX 7, que fue la última versión distribuida libremente de UNIX lanzada por Laboratorios Bell, antes de ser comercializada formalmente por AT&T, núcleo que ya se había trabajado en su desarrollo en 1986, pero de acuerdo a una decisión de sus programadores decidieron no utilizarlo pues no era portátil ya que solo funcionaba en computadores caros y complicados. Es por esto que se inician negociaciones con el CMU (Universidad Carnegie-Mellon), para utilizar como base el micro núcleo MACH, el cual demostró muchas dificultades.

La idea de adoptar MACH como base para el núcleo del sistema GNU fue una decisión técnica de Richard Stallman, lo cual considero luego un error, pues demoró mucho el desarrollo de HURD. Posteriormente se vienen realizando varios esfuerzos para poder portar el HURD a otro micro núcleo, la primera elección fue el microkernel L4, el cual es la siguiente versión para el núcleo MACH, el cual cuenta con similitudes al anterior, pero con ciertas mejoras.



En 1991, Linus Torvalds escribió el código para el núcleo Linux, y lo distribuyó bajo la licencia GPL, de esta forma programadores alrededor del mundo se fueron uniendo a este proyecto, colaborando a través de Internet, llegando a ser un núcleo compatible con UNIX.

En 1992, el núcleo fue combinado con el sistema GNU, lo que derivó en un sistema operativo libre y funcional conocido generalmente como GNU/LINUX, el cual tiene muchas variantes.

2.3.6. Freeware

Se refiere a programas que se puede distribuir libremente, sin costo y por tiempo ilimitado, se distribuye con una licencia de uso, la cual no permite modificar el programa, y dar el crédito a su autor, no permite su uso comercial y el código fuente no está disponible, por lo cual no se puede considerar a estos programas como software libre. Este término fue creado por Andrew Fluegelman quien quería distribuir un software sin usar los métodos tradicionales para una versión del programa PC-TALK.

2.3.7. Shareware

Es una modalidad de distribución de software en la que el usuario puede utilizar por tiempo limitado y evaluar el producto de forma gratuita pero con restricciones, para que el usuario deba comprar una licencia para adquirir el software con todas sus funcionalidades.



De igual forma el shareware no es software libre pues en muchas ocasiones el código fuente no es distribuido, y tampoco permite realizar copias sin pagar un valor por la licencia, aunque en la práctica muchos usuarios desconocen esto.

2.3.8. Software Privativo

Es un software que no es libre ni semi libre, para utilizar, modificar o redistribuir este tipo de software requiere de autorización.

2.3.9. Software Privado

Este software es desarrollado a medida o de acuerdo a los requerimientos del usuario, el cual al utilizarlo no permite liberarlo al público ni su código fuente ni como binario.

El usuario puede ser considerado, como una persona particular o una empresa, organización o compañía,

2.3.10. Software Comercial

Es aquel desarrollado por una empresa u organización para su comercialización, con lo cual pretenden obtener ganancias, por lo que el software comercial sería software privativo, pero también el software libre puede ser comercial, pues se distribuye mediante una licencia GPL, y se hacen contratos de mantenimiento.



2.4. Licencias para Software

La licencia es un contrato entre la persona o empresa que ha desarrollado un software, el propietario de los derechos y la persona o empresa que va a utilizar este software, en este caso el usuario final, que establece las restricciones en cuanto a su uso, redistribución, la cesión de derechos, modificación y bajo qué condiciones de acuerdo a ciertas normas y condiciones de acuerdo a las cláusulas del contrato.

Para clasificar una licencia se toma en cuenta ciertos aspectos:

- En general para que una licencia sea de software libre, debe cumplir las definiciones de software libre que tratan sobre las libertades que se otorgan a los usuarios finales.
- La licencia debe ser de tipo copyleft.
- Si es compatible con la Licencia GNU GPL.

2.4.1. Compatible con GPL

- **Licencia Pública General de GNU (GNU GPL)**

Es la licencia más común y más utilizada, por lo productos de software libre, pues es una licencia que cumple las libertades definidas.



Esta licencia fue hecha por la Free Software Foundation, es de tipo copyleft, pues se refiere en términos totalmente distintos a los términos de la licencias de tipo copyright.

Esta licencia permite la redistribución del software y su código fuente en todo momento, así se permite hacer modificaciones, aunque permite que solo se integren con paquetes de software que tengan la misma licencia.

Es así que los programas o las modificaciones hechas a estas licencias no pueden ser redistribuidos con otro tipo de licencia, garantizando así que no pueda ser hecho software propietario.

- **Licencia Pública General Menor de GNU (GNU LGPL)**

Es la otra licencia hecha por la Free Software Foundation, en principio estaba destinada para ser usada en librerías, y así potenciar el desarrollo sin tener las limitaciones de la GPL, es así que la L correspondió a "Library", pero fue modificada recientemente entre la versión 2 y la versión 2.1., para ser considerada una versión ligera de la licencia GNU GPL.

Esta licencia tiene un copyleft no tan fuerte, pues permite la integración con módulos de software propietario, con lo cual la FSF, recomienda su uso, solo en casos muy puntuales y excepcionales.



- **Licencia de X Window versión 11**

Esta licencia es usada para la distribución del sistema X Windows, que es la interfaz gráfica mayormente utilizada por sistemas tipo Unix y también GNU/Linux, también es conocida como Licencia MIT *Massachusetts Institute of Technology*, pues originalmente se creó en este instituto en el año de 1980.

Esta licencia no tiene copyleft, lo que permite sin restricciones que se pueda usar, copiar, modificar, integrar con otro software, vender y distribuir bajo los mismos términos.

El software que da resultado con esta licencia, puede ser tanto software libre como software no libre, y los cambios que se hace al programa original permite licenciarlos con licencias compatibles como la BSD, a la cual es muy similar o con la licencia GPL.

- **Zope Public License 2.0**

Es una licencia simple y permisiva de software libre compatible con la licencia GPL, que no tiene copyleft, es usada para la distribución de Zope, que es un servidor de aplicaciones y para otros productos relacionados, esta licencia tiene una característica adicional pues no permite el uso de las marcas registradas por Zope Corporation.



- **Licencia de Sleepycat**

Es una licencia a veces conocida también como la licencia Berkeley DB, en referencia a la base de datos que comercializó la empresa Sleepycat Software, con la cual también distribuía otros de sus productos.

Esta compañía fue adquirida en el 2006 por Oracle Corporation, quien todavía sigue utilizando la licencia con el mismo nombre a pesar de que no hace referencia al nombre en ningún documento, Oracle la usa para distribuir los productos Berkeley DB.

Esta licencia obliga a distribuir el software con el código fuente y de igual forma al usuario que lo redistribuya tiene que hacerlos bajo los mismo términos, incluidas las modificaciones, lo cual da un efecto igual al de la licencia GPL pero de forma más estricta.

- **eCos License 2.0**

Esta licencia es usada para distribuir eCos, un sistema operativo de tiempo real, compatible con varias arquitecturas, esta licencia permite la integración con módulos de software que no tienen licencia GPL, lo cual la hace muy parecida en efectos a la GNU LGPL.



2.4.2. Incompatible con GPL

- **Affero General Public License**

Esta licencia contiene una modificación de la licencia GPL versión 2.0, en la que añade una cláusula, que hace referencia a programas que ofrecen algún servicio vía Web o a través de una red de computadores, pues al usarlo esto no implica que haya sido recibido mediante una redistribución que esté con licencia, y así lo puede modificar para su uso pero como no lo está redistribuyendo no tiene la obligación de enviar el código fuente. Debido a esta restricción no es compatible con la GPL versión 2, pero de acuerdo a la Free Software Foundation **FSF**¹, es compatible con la versión 3.0, de acuerdo a como está escrita la cláusula mencionada.

- **Licencias Tipo BSD**

Esta licencia tiene su origen en las versiones de Unix que publicaba la Universidad de Berkeley en California EEUU, en la cual obliga dar el debido reconocimiento a los autores, así como permite la redistribución tanto binaria como de los códigos fuentes, aunque no sea una obligación, también permite la modificación y la integración con otros tipos de software casi sin restricciones.

¹ Why the Affero GPL: <http://www.gnu.org/licenses/why-affero-gpl.html>



Existen varias licencias de tipo similar como la licencia Apache, X Window, que son consideradas como tal en una versión mínima pues en realidad solo dan los créditos a los autores, su creación se debe a que este tipo de licencia era para software desarrollado en universidades en proyectos de investigación financiados por el estado, de lo cual parte el hecho, de que estos fondos como habían sido pagados por los impuestos, el software podía ser redistribuido sin restricciones, en forma binaria, sin necesidad de entregar el código fuente.

Esto es considerado por algunos como la real esencia del software libre, lo cual no conviene al software libre pues no necesariamente se redistribuyen aplicaciones libres bajo las definiciones que lo caracterizan.

Uno de los problemas prácticos de esta licencia, es la creación de software que no sigue un estándar en particular, pues los desarrolladores no encuentran obstáculos para la implementación de software bajo este tipo de licencia. Para la Free Software Foundation el principal problema radica en la cláusula de publicidad de esta licencia pues así al redistribuirlo sin restricciones cada usuario que lo modifique y redistribuya puede cambiar esta cláusula generando de esta forma gran cantidad de licencias que deberían cada una tener un nombre diferente.



Para evitar esto, se modificó la licencia en lo que se conoce como la Licencia BSD modificada, la cual elimina la cláusula de publicidad resultando ser una licencia más permisiva y compatible con la GNU GPL.

- **Licencia OpenSSL**

Es una combinación de dos licencias, la una es la misma licencia y la otra es la licencia de SSLeay que es una implementación libre y no comercial de SSL 2.0 y 3.0.

De esta forma se deben cumplir las cláusulas establecidas en las dos licencias, lo que da como efecto una licencia permisiva y sin copyleft pero de igual forma que la licencia BSD, contiene una cláusula de publicidad lo cual hace que esta licencia no sea compatible con la licencia GNU/GPL.

- **Licencia de Apache**

Licencia permisiva y sin copyleft, es con la cual se distribuyen la mayoría de productos apache, en la versión 1.1, no es compatible con la GNU/GPL versión 2, pues no obliga a la redistribución del código fuente, lo cual cambia en la licencia de Apache versión 2, la cual ya es compatible con la versión 3 de la GNU GPL, pues es más sencillo de usar esta licencia para proyectos que no sean de la Apache Software Foundation, permitiendo que la licencia sea incluida por referencia en lugar de estar en cada archivo.



- **IBM Public License Versión 1.0**

Esta licencia permite la redistribución de los binarios con la condición de que de alguna forma se entregue también los códigos fuentes, pero bajo la misma licencia.

Lo que caracteriza a esta licencia y la hace incompatible con la GNU/GPL, es que obliga al usuario que redistribuye el software con las modificaciones realizadas a licenciar las patentes.

- **Mozilla Public License 1.1**

Esta licencia tiene su origen de la primera licencia libre que tuvo el Netscape Navigator, de la cual se hicieron modificaciones para distribuir el programa bajo su propia licencia, libre la cual no es compatible con la GPL, pues no tiene un copyleft fuerte y además según la FSF, no se puede integrar un módulo con licencia GPL, con otro módulo de un software con una licencia de este tipo.

A pesar de que no es compatible con la GPL, esta licencia en su versión 1.1 tiene una excepción que cabe resaltar, pues permite que para una parte o varias partes de un software se puedan escoger entre esta licencia y otras licencias.



2.4.3. Licencias para Documentación

- **Licencia de Documentación Libre de GNU**

Esta licencia se aplica de acuerdo a la Free Software Foundation para la documentación como los manuales, entre otros documentos de tipo técnico o didáctico, esta licencia tiene copyleft.

Su origen se debe a que al darse cuenta de que un documento que acompaña al programa no es lo mismo que el software, de esta forma se debe poner a disposición de quien requiera una copia del documento que sea transparente y legible para el usuario, quiere decir que no esté codificada como un archivo sin formato para un conjunto mínimo de herramientas, además de las copias que estarían codificadas, lo que daría el mismo efecto de los binarios que se entregan y su respectivo código fuente.

- **Licencia de Documentación de FreeBSD**

Licencia de documentación libre y permisiva pero que a diferencia de la primera no tiene copyleft, aun así es compatible con la Licencia de Documentación Libre de GNU.

- **Licencia de Documentación Común de Apple**

Esta licencia es de documentación libre pero incompatible con la Licencia de Documentación Libre de GNU, pues no permite añadir otros términos o condiciones de los que ya se encuentran en esta licencia.



- **Licencia de Publicación Abierta**

Es una licencia de documentación libre siempre y cuando el propietario de los derechos de autor no haga uso de las cláusulas que hacen referencia a las Opciones de Licencia que se presentan, pues al hacerlo esta licencia deja de ser libre.

De igual forma si al redistribuirlo no se usan estas opciones, se pone en riesgo de que en el software resultante, el usuario decida cambiar de opinión sobre estas opciones lo cual dejaría sin efecto este software como libre.

- **Licencias de Creative Commons**

En el año del 2001 se fundó Creative Commons una organización sin fines de lucro, con la finalidad de fomentar recursos de forma intelectual puesto a disposición de la comunidad de diversas maneras, la cual es dirigida por expertos en propiedad intelectual, derecho, informática.

Uno de los trabajos más reconocidos fue una serie de licencias creadas no para software, sino para trabajos literarios, artísticos, didácticos, como se había mencionado recursos de forma intelectual.



2.5. Herramientas de software libre por su funcionalidad.

En la actualidad y con el desarrollo del Sistema GNU, que luego fue combinado con el núcleo Linux, la intención del proyecto GNU es la de integrar a este sistema herramientas que en su totalidad cumplan con las definiciones del software libre.

El núcleo Linux, además permitió la creación y desarrollo de distribuciones que utilizan las librerías del proyecto GNU, y la interfaz gráfica que brinda el sistema X Windows.

Con el desarrollo de estas distribuciones y de acuerdo a la funcionalidad que estas ofrecen al usuario y sus necesidades, estos paquetes de software incluyen también una gran variedad de herramientas, llegando cada vez más a un alto grado de robustez, poniéndose en el nivel de sistemas operativos de software propietario, marcando una alta competencia.

De acuerdo a esto el proceso de migración a herramientas de software libre se da no solo sobre sistemas operativos libres, sino también sobre software propietario en este caso sobre Microsoft Windows y para lo cual existen alternativas a los más conocidos paquetes informáticos, lo cual es un primer paso que los usuarios que quieran utilizar herramientas de software libre, puedan hacerlo instalando sobre Windows y posteriormente migrando



completamente el sistema operativo a una distribución basada en Linux y así conocer las ventajas tanto económicas, como a nivel de seguridad, aprendiendo a trabajar en entornos diferentes, mostrando al usuario que existen alternativas para sustituir a herramientas de software propietario.

En este capítulo se tratara principalmente sobre herramientas para el desarrollo de aplicaciones, así como para la manipulación de imágenes y para el usuario final aplicaciones para la navegación en Internet y herramientas de escritorio, para la elaboración de documentos, hojas de cálculo y presentaciones.

2.5.1. Bases de datos

Para el manejo de datos de una forma estructurada y para mantener la información de una forma ordenada, es necesaria una infraestructura más sofisticada y así garantizar las ventajas que ofrecen los Sistemas Gestores de Base de Datos, como la consistencia, seguridad, manejo de transacciones y mínimos tiempos de respuesta.

De esta forma existen herramientas más livianas enfocadas al usuario final que de igual forma les permite un manejo de datos organizado.



En el mercado informático existen bases de datos propietarias y base de datos de código abierto como MySql y Postgresql, que últimamente han crecido permitiendo más funciones acercándose a las características que prestan software propietario como Oracle y SQL Server.

- **MySQL**

Es una base de datos relacional, que cumple las funciones de un servidor para el acceso de varios usuarios a la vez a varias base de datos, es la base de datos de código abierto más popular, que pone su código fuente disponible bajo la licencia GNU GPL, y en caso de que no se requiera esta licencia el usuario puede adquirir una licencia privativa que les permita su uso. MySQL pertenece y es patrocinada por una empresa privada con fines de lucro, MySQL AB, una empresa sueca, que fue adquirida por Sun Microsystems una subsidiara de Oracle Corporation. A partir del código fuente de MySQL, se han creado otros paquetes de base de datos que como Drizzle, fue creado a partir del código fuente de MySQL 6.0 y fue presentado antes de que Sun Microsystems fuese adquirida por Oracle. Drizzle es distribuido bajo la versión GNU GPL versión 2.0.

Varios proyectos de software libre incluyen a MySQL dentro de sus componentes como es el caso de WAMP, LAMP, paquetes que ofrecen una solución completamente funcional, para el desarrollo de aplicaciones, bajo un servidor Web.



También es utilizado en sitios Web de gran acceso y que manejan grandes cantidades de información, como Flickr, Facebook, Google (aunque no para búsquedas), Nokia, Youtube y Wikipedia.

El código de MySQL es desarrollado en C y C++, trabaja en diferentes plataformas como AIX, FreeBSD HP-UX, Linux, Mac OS X, Novell NetWare, Solaris, Symbian and Microsoft Windows. La mayoría de lenguajes de programación incluyen librerías para establecer la conexión a MySQL.

Para la administración de MySQL, una de las herramientas más conocidas catalogada como software libre es phpMyAdmin, escrita en php con la finalidad de administrar a través de la Web

Su rápido crecimiento en su desarrollo y funcionalidades se debe a la dedicación e inversión del grupo de trabajo de MySQL, así como a la gran comunidad que contribuya y da soporte al proyecto, dejando así un sistema gestor de base de datos lo suficientemente maduro para ser considerado como una alternativa para su uso en producción.

- **Postgresql**

Es una base de datos relacional orientada a objetos de software libre que se distribuye bajo la licencia BSD.



Más conocido como Postgres, esta base de datos no pertenece a una empresa privada, pero tiene una gran comunidad y organizaciones que soportan su desarrollo, que fue hecho en C, esta comunidad es conocida como PostgreSQL Global Development Group. Entre sus clientes más importantes se encuentran Yahoo, MySpace, hi5, Skype.

En la actualidad la empresa privada EnterpriseDB provee soporte a nivel empresarial para Postgresql, con su producto Postgres Plus Standard Server, una base de datos relacional, desarrollada con paquetes adicionales de software libre y que se enfoca al mercado empresarial, esta empresa también ofrece soporte para el desarrollo de aplicaciones basadas en Postgresql, de igual forma con su versión Advanced permite simplificar la migración de datos desde otras bases de datos.

La herramienta ideal para la administración vía Web de Postgresql, es phpPgAdmin, que es una herramienta de software libre.

La situación actual de las base de datos de software libre o código abierto, garantizan su estabilidad y continuo desarrollo y mejoramiento a lo largo de los años, permitiendo al usuario optar por estas herramientas como una viable solución a base de datos de software propietario.



2.5.2. Desarrollo Web

El Internet que en principios fue creado como un proyecto de investigación de tecnologías para el envío de paquetes a través de computadores que estén conectados a esta red y así comunicarse de una forma más fácil, se convirtió con el tiempo en una vía de comunicación a la cual se conectaban cada vez más personas, lo cual permitió a las empresas publicar todos sus servicios mediante aplicaciones a través de la Web, dando el acceso a sus clientes para que realicen sus transacciones. Para presentar estos servicios el Internet utiliza un sistema de documentos de hipertexto llamado WWW (World Wide Web), que mediante vínculos un usuario puede visualizar sitios Web que contengan texto, imágenes y contenido multimedia.

Para que la Web pueda realizar estas acciones, la W3C, un consorcio internacional que define las reglas para que los servidores en Internet puedan comunicarse efectivamente, creo tres estándares principales:

- **URL (Uniform Resource Locators):** Es una secuencia de caracteres o un nombre que identifica únicamente a diferentes recursos en la Internet, lo que permite localizarlos fácilmente.
- **HTTP (Hypertext Transfer Protocol):** Es un protocolo que permite la comunicación y envío de información entre varios servidores a través de la



Web, simplificando su funcionamiento para atender miles de peticiones independientemente de las características de hardware y software de estos servidores.

- **HTML (Hypertext Markup Language):** Es un lenguaje que mediante marcas permite la creación y elaboración de páginas Web, definiendo así la estructura para presentar el contenido en formato de texto, ayudándose de contenido multimedia.

Es así que a lo largo del tiempo y con la creación de estándares y técnicas para que las empresas puedan ofrecer mejores servicios y una mejor imagen se han creado herramientas que permiten que el Internet sea hoy por hoy, uno de los principales medios de presentación a nivel mundial.

- **WampServer**

Es una herramienta que permite levantar toda la infraestructura completa de un servidor Web, para el desarrollo de páginas HTML básicas y avanzadas con el lenguaje PHP. Su nombre es un acrónimo conformado por las herramientas que utiliza:



Windows, sistema operativo.

Apache, servidor Web

MySQL, base de datos relacional

Php, lenguaje de programación.

También en este paquete se incluye phpMyAdmin como visualizador y administrador de las base de datos de MySQL.

La alternativa para instalar sobre Linux, se denomina LAMP, haciendo referencia a las mismas herramientas y de igual forma existen alternativas para otras plataformas como MAMP (Macintosh), SAMP (Solaris), FAMP (FreeBSD).

La instalación de WAMP, es sencilla y configura automáticamente todos sus componentes y la integración y activación de librerías de MySQL sobre PHP, y para que Apache interprete el lenguaje de programación.

- **XAMPP**

Es un servidor que se puede instalar por el momento en 4 plataformas Linux, Windows, Mac OS X y Solares, de igual forma tiene incluido MySQL como gestor de base de datos, Apache como servidor Web y con la particularidad que Apache interpreta los lenguajes de programación PHP y Perl, el programa está publicado bajo la licencia GNU.



- **Bluefish**

Es un editor para desarrollo Web, que se enfoca en la creación de sitios Web dinámicos, destinado para programadores y diseñadores, funciona sobre plataformas compatibles con estándares POSIX (Portable Operating System Interface), que permite generalizar las interfaces de los sistemas operativos para que una misma aplicación pueda ejecutarse en distintas plataformas, es un proyecto de desarrollo de software libre, liberado bajo la licencia GNU GPL. Con esta herramienta se puede realizar el desarrollo en varios lenguajes como CSS, Javascript para HTML avanzado, XML, generalmente para la publicación de servicios Web, PHP, C, Java, SQL, Perl, JSP, Python, Ruby y shell.

- **Kompozer**

Es un editor de páginas Web, WYSIWYG (What You See Is What You Get), permitiendo ver directamente el resultado final que aplica a editores de texto con formato, que ofrece las mismas funcionalidades que Dreamweaver, liberado bajo tres licencias, MPL 1.1, GPL 2.0, LPGL 2.1, es un editor basado en el código fuente del editor discontinuado Nvu, mantenido por la misma comunidad de usuarios que creó esta herramienta, proyecto el cual pertenece a SourceForge.



- **PHP**

Es un lenguaje de programación interpretado, para aumentar la funcionalidad sobre sitios Web estáticos con solamente código HTML, que permite funciones más avanzadas, con la creación de sitios Web dinámicos. Su nombre proviene de un acrónimo recursivo, PHP hypertext preprocessor, tiene una sintaxis sencilla parecida a la del lenguaje C, es multiplataforma y en sus últimas versiones es orientado a objetos.

La ventaja de PHP, es que todas sus librerías son de software libre, así mismo está publicado bajo la licencia PHP 3.0.1, que es considerada por la Free Software Foundation como software libre.

PHP funciona de forma que una vez hecha la solicitud de la página al servidor que interpreta el código, la ejecuta y devuelve el resultado que generalmente es HTML.

- **Firebug**

Es un componente de Firefox que permite analizar, editar el código fuente HTML, CSS y Javascript y DOM (Document Object Model) además de monitorear el rendimiento de las páginas web, este componente es un plug-in de Mozilla tanto de software libre como open source, bajo la licencia BSD.



- **Apache**

Es un servidor Web de código abierto, desarrollado por la colaboración de una comunidad auspiciada por la Apache Software Foundation y desarrollado bajo el proyecto HTTP Server, httpd, que implementa el protocolo HTTP 1.1, además de introducir el concepto de sitio virtual.

Actualmente es el servidor web más utilizado y con más aceptación, pues dentro de sus ventajas esta que funciona sobre diferentes plataformas, es modular, de código abierto, publicado bajo la licencia de Apache la cual no es considerada compatible con la GNU GPL versión 2.0 por la Free Software Foundation, sin embargo en la GNU GPL versión 3.0 existen una cláusula que permite la compatibilidad con la licencia de Apache.

- **Mono**

Es un conjunto de herramientas basadas en GNU/Linux y compatibles con .NET para el desarrollo de software, es un proyecto iniciado por Ximian, que fue adquirida por Novell, quienes continúan impulsando y dando soporte. Funciona sobre distintas plataformas incluida Windows.

Esta implementación está basada en los estándares para C# de ECMA, una organización privada de estándares para sistemas de información y comunicación.



- **Eclipse**

Es un ambiente de desarrollo de software multilenguaje que incluye un IDE y varios plugins que le dan más funciones.

Escrito en Java, permite el desarrollo de aplicaciones en Java, además de otros lenguajes dependiendo del plugin, por ejemplo C, C++, PHP, Ruby, Python, Perl, de esta forma el IDE adquiere su nombre de acuerdo al plugin, componiéndose su nombre con la primera letra de acuerdo al lenguaje seguido de las iniciales correspondientes a Developments Tools, por ejemplo para PHP, Eclipse PDT. Es publicado bajo la licencia Eclipse, la cual no es compatible con la licencia GNU GPL, aunque es considerada como una licencia de software libre por la Free Software Foundation.

2.5.3. Modelamiento

Existen varias herramientas que permite visualizar, especificar los diagramas que durante el desarrollo de software se crean, haciendo estos diagramas más funcionales y reales describiendo funciones y procesos del requerimiento del usuario. UML (Unified Modeling Language) define estándares para el modelado de sistemas en su mayoría para el desarrollo de software aunque también para el campo de la mecánica, electrónica y educación, cuenta con el soporte del OMG (Object Management Group), una organización que define estándares para varias tecnologías.



- **ArgoUML**

Es una herramienta de modelamiento UML desarrollada en Java, que tiene soporte para todos los diagramas UML versión 1.4, su ventaja al ser desarrollada en Java, es que puede ser ejecutada en cualquier plataforma que soporte Java, esta aplicación es liberada bajo la licencia Pública de Eclipse.

ArgoUML, es un proyecto de código abierto, auspiciada por Tigris.org, y aunque ha ganado premios de desarrollo de software, todavía no implemente completamente el estándar UML.

- **Umbrello UML Modeller**

Es una herramienta de software libre que permite crear diagramas de software en el formato Standard de UML

Su código nativamente fue escrito para UNIX, es parte de KDE, un entorno de escritorio amigable para el usuario final y es publicado bajo la licencia GNU GPL. Permite realizar la reingeniería inversa desde varios lenguajes de programación como por ejemplo: C++, Delphi, Java y a su vez también exporta a varios lenguajes, pues soporta todos los diferentes diagramas de UML.



2.5.4. Gráficos

Es más común en este campo el uso de herramientas de Adobe, que han alcanzado una estabilidad para la edición de gráficos e imágenes en distintos formatos, y que además con la última adquisición de las herramientas de Macromedia, Adobe ha potenciado de manera general sus productos, integrando la edición de imágenes estáticas, el diseño gráfico, con la realización de animaciones y movimiento en 3D. Sin embargo en el campo de Software Libre existen también herramientas que alcanzado un nivel de madurez gracias a la colaboración participativa de una gran comunidad, permitiendo hacer animaciones de alta calidad, puramente con herramientas de software libre.

- **Gimp**

Es una herramienta para la edición de imágenes y retoque de fotografías, presentándose como una seria alternativa a Adobe Photoshop.

Es parte del proyecto GNU y su nombre proviene del acrónimo (GNU IMAGE MANIPULATION PROGRAM), es distribuido bajo la GNU GPL, la versión más reciente se ejecuta en la mayoría de sistemas operativos, incluyendo los Linux, Mac OS X, y Windows.



- **Blender**

Es una aplicación de software libre para la creación y edición de gráficos en 3D, así como también aplicaciones interactivas como juegos, películas animadas y efectos especiales.

Blender está disponible para la mayoría de sistemas operativos, bajo la licencia GNU GPL, no necesita de mayores características de hardware debido al poco tamaño de instalación.

2.5.5. Internet

Para el usuario final una vez que el Internet se convirtió en una herramienta esencial de investigación y consulta, así como de presentación para los servicios de una empresa, también es una herramienta para el ocio, lo cual provoca generalmente exponerse el usuario a una serie de amenazas y posibles ataques, exponiendo la información publicada.

Sea el caso de uso que se le dé al Internet estas amenazas están presentes, más para las herramientas de software propietario que como ha quedado demostrado, en el caso de Internet Explorer es una de las herramientas más vulnerables en el sentido tecnológico, por la existencias de virus y troyanos a través de la red global.



Dado que para el software libre, la colaboración y participación viene de una gran comunidad de programadores, se han desarrollado herramientas que permiten al usuario final, herramientas más seguras, que permiten en muchos de los casos, navegar por Internet de una manera más segura y estable, incluyendo más funcionalidades haciéndolas una seria alternativa.

- **Mozilla Firefox**

Es un navegador Web de software libre y código abierto, publicado bajo tres licencias, MPL, GNU GPL Y GNU, LPGL, proviene de Mozilla Application Suite, un conjunto de herramientas que incluyen un navegador, un programa de correo y noticias, un editor de páginas Web y un cliente de chat, cuyo código actualmente fue renombrado a SeaMonkey.

Es desarrollado por Mozilla Corporation, y actualmente es el segundo navegador más utilizado en el mundo detrás de Internet Explorer, una de sus ventajas es que muchas funcionalidades pueden ser añadidas mediante plugins y extensiones desarrollados por terceros.

Funciona sobre la mayoría de plataformas y versiones de sistemas operativos Linux, Mac OS X y Windows, lo cual es una ventaja pues al usar Firefox sobre Windows es un primer paso a una transición para el uso de más herramientas de software libre.



Para que Firefox sea una herramienta más segura, no soporta la ejecución de controles ActiveX, pues las páginas Web que requieren estos controles solo se ejecutan en versiones de Windows, además de que con la creación y ejecución de estos controles se han hecho el blanco para software malicioso

- **Thunderbird**

Es un cliente de correo y noticias de software libre y código abierto, de igual forma que Mozilla Firefox, publicado bajo tres licencias, MPL, GNU GPL y GNU, LPGL.

Desarrollada por Mozilla Messaging Inc, una subsidiaria con fines de lucro, de Mozilla Foundation, es multiplataforma y funciona sobre Windows, Linux, Mac OS X, entre otros, además que su código fuente como está disponible puede ser compilado para ser ejecutado en una variedad de sistemas operativos.

- **FileZilla**

Es un cliente FTP, de código abierto, publicado bajo la licencia GNU GPL, es multiplataforma y soporta FTP, SFTP y FTPS.

- **Amsn**

Es un cliente de mensajería instantánea, cuyo nombre proviene de Alvaro's Messenger, que soporta únicamente el protocolo MSN, a diferencia de otras herramientas de este tipo, que soportan también otras redes, como Yahoo y



ICQ, esto permite dedicarse a emular las funcionalidades de Windows Live Messenger así como su apariencia.

Funciona sobre una variedad de plataformas que soportan Tcl/Tk, que es un lenguaje de scripts, dentro de estas plataformas están Windows, Mac OS X, y las distintas distribuciones de Linux, además de Maemo, sistema operativo que usa Nokia en sus últimos modelos de Tablets.

- **Pidgin**

Es un cliente de mensajería instantánea multiplataforma y que se conecta a varias redes (multiprotocolo), originalmente conocido como Gaim, utiliza la librería libpurple, para conectarse desde un solo programa a distintas redes de mensajería instantánea. Es una aplicación de software libre publicada bajo la licencia GNU GPL.

2.5.6. Ofimática

Uno de los puntos críticos a la hora de realizar una migración a software libre en empresas no solamente de tecnología, es la elección de una herramienta básica de trabajo que también es básica para la educación en general.

Sin embargo que existen varios paquetes informáticos que prestan están funcionalidades enfocadas al usuario final, la más usada es Microsoft Office, por todas las atractivas funciones y características tan interesante, siendo su



único problema el costo por licencia que se debe adquirir para su uso al ser software propietario, para lo cual no todos los clientes tienen acceso u oportunidad económica, lo que provoca hacer uso de la piratería para conseguir este paquete informático.

Constituye uno de los grandes avances logrados por el movimiento del software libre la creación de una herramienta que de las mismas funcionalidades que presente las mismas características para ofrecer un producto de calidad que pueda competir con las herramientas del paquete informático de Microsoft Office como Word, Excel y PowerPoint.

- **OpenOffice**

Es un conjunto de herramientas de software libre entre las que se incluyen un procesador de texto, una hoja de cálculo, creador de presentaciones y base de datos. Su código fuente esta publicado bajo la LPGL versión 3.0, es multiplataforma y esta disponibles en varios idiomas.

OpenOffice soporta el estándar OpenDocument Format (ODF) aprobado por la ISO (International Organization for Standarization) y la IEC (International Electrotechnical Commission), el cual es un formato de archivos para el almacenamiento de documentos de paquetes ofimáticas.



Originalmente nombrada como StarOffice fue desarrollada en sus inicios por StarDivision que fue adquirida por Sun Microsystems, la cual fue adquirida por Oracle Corporation, que actualmente lidera el desarrollo del conjunto de aplicaciones de OpenOffice, apoyada por otras grandes empresas como Novell, RedHat, IBM, Google entre otras.

Dentro de sus componentes encontramos herramientas que ofrecen las funcionalidades básicas que ofrece Microsoft Office:

- **Writer**, es un procesador de texto que puede imprimir archivos en el formato PDF sin ningún tipo de software adicional
- **Calc**, es una hoja de cálculo.
- **Impress**, es un programa para crear presentaciones.
- **Base**, Es un sistema de administración de Base de Datos.
- **Draw**, es un editor de gráficos vectoriales,
- **Math**, una herramienta para la creación y edición de fórmulas matemáticas.

De esta manera OpenOffice reduce en gran cantidad el dominante mercado que estaba liderado por Microsoft Office, siendo unos de los objetivos principales en la migración a software libre en las entidades gubernamentales y privadas, pues al representar un ahorro por el nulo costo de licencias, la migración se centra en la resistencia al cambio que presenta



el usuario final por la costumbre adquirida de usar lo más comúnmente se encuentra en el mercado informático.

2.6. Ventajas, Desventajas y Problemática del software libre

Existe una alta competitividad entre software libre y software propietario donde cada cual exhibe sus ventajas y desventajas para cumplir con el único objetivo de hacer llegar su producto al usuario final, la diferencia se da en la forma como cada uno realiza el mercadeo de sus paquetes informáticos y en el caso del software libre como financiar sus proyectos, y para el caso del software propietario, sencillamente vender más, aunque después de realizada la venta, se descuide de alguna manera u otra al usuario final, quien termina haciendo una gran inversión.

2.6.1. Ventajas

- **Participación libre del usuario**

Como el usuario para el software propietario solo termina siendo el cliente final para obtener sus ganancias, muchas veces el soporte no permite al usuario participar por completo en este proceso, lo que para el software libre va mucho más allá, convirtiendo al cliente final en un actor más de la cadena de producción, brindándole soporte mediante una comunidad, para que el a su vez pueda brindar soporte, de manera libre y no necesariamente gratis, beneficiando a la sociedad directamente, pues entra a capacitar, a colaborar, a enseñar y a seguir aprendiendo a la vez, y en última instancia proyectando



sus servicios profesionales para dar soporte a grandes empresas, como las empresas del gobierno en muchos países que han escogido como ideología política y social el uso de software libre.

- **Ahorro en el Licenciamiento**

Bajo la filosofía de software libre y al enfocarnos en el concepto de free como libre y no como gratis, todo paquete informático libre se distribuye a la final con una licencia, la cual al basarse en la libertades definidas, abre una puerta a la empresas para que al no tener que pagar un valor por las licencias, puedan realizar una mejor inversión en el soporte y mantenimiento del software personalizada para el usuario final.

- **Altos niveles de seguridad**

Al tener acceso al código fuente, la comunidad de desarrolladores y las empresas que auspician a los paquetes informáticos de software libre tiene más libertad para trabajar sobre la seguridad y posibles errores que hagan vulnerable al software, con una colaboración directa entre ellos a través del Internet, para así dar soluciones y hacer mejoras en el código fuente del paquete informático.

- **Reutilización de recursos**

Ante la creación de paquetes informáticos de software libre existentes y publicados en Internet, el tiempo de desarrollo para nuevas aplicaciones se hace más corto, pues al tener acceso al código fuente, los desarrolladores,



pueden reutilizar el código en su totalidad o partes de él, para solucionar problemas puntuales y no tener que empezar desde cero.

- **Productos Multiplataforma**

Originalmente muchos paquetes informáticos de software libre fueron desarrollados especialmente para plataformas GNU/Linux y ambiente Unix, bajo lenguajes como Java, con lo cual luego les permitió funcionar bajo cualquier plataforma que soporte el lenguaje con el cual fueron desarrollados, lo cual en el caso de que fuera desarrollado para sistemas operativos propietarios, el costo del desarrollo para su funcionamiento sobre otras plataformas sería más costoso.

- **Características de Hardware menores**

De acuerdo a las necesidades y requerimientos que se tenga sobre la funcionalidad del servidor, se pueden instalar distribuciones de Linux, que no requieren características de hardware sofisticadas, pues presentan la opción de ser instaladas sin la interfaz gráfica, para que el servidor pueda ser administrado vía comandos.

- **Relación con varios proveedores.**

Una vez que el usuario final tenga acceso al código fuente, en el caso de que quiera realizar modificaciones a la aplicación, no tiene que depender únicamente de un solo proveedor, en este caso del desarrollador de la aplicación, sino que puede libremente escoger en el mercado cualquier



proveedor disponible y con los conocimientos suficientes para ofrecer mantenimiento al código o distintos servicios sobre la aplicación, este soporte no quiere decir que sea gratuito.

2.6.2. Desventajas

- **Altos costos de migración e implantación**

Dado que los paquetes o herramientas a instalar en una empresa que esté atravesando un proceso de migración a Software Libre, constituyen como algo nuevo tanto a nivel tecnológico, como a nivel de los usuarios finales, la implementación y la capacitación a usuarios representa un alto costo, partiendo del hecho de que generalmente el usuario puede ofrecer resistencia al cambio y a la costumbre adquirida de trabajar anteriormente con otras herramientas.

- **Tiempos de aprendizajes altos**

Generalmente algunas herramientas de software libre y en el caso más específico de las distribuciones de Linux a nivel de sistema operativo, van más destinadas a usuarios más experimentados en el campo de la informática, especialmente en el desarrollo del software, pues al no tener acceso a una interfaz gráfica el control y administración se lo realiza vía comandos, incluso contando con una interfaz gráfica muchas veces se tiene que realizar acciones de configuración vía comandos, por lo cual, para usuarios menos experimentados resultaría un aprendizaje un poco más



lento, pues se empezaría la enseñanza con conceptos básicos de computación lo cual llevaría más tiempo para que el usuario tenga conocimientos generales sobre la herramienta para su uso.

- **Compatibilidad**

De acuerdo a las funciones específicas que realice el usuario final en la organización donde presta sus servicios, utiliza software específico para llevar a cabo sus actividades y labores, generalmente estos paquetes informáticos que utiliza funcionan sobre software propietario, lo cual dificulta su operabilidad sobre otras plataformas. La solución es buscar alternativas de herramientas dentro del software libre, para que el usuario siga realizando sus actividades con el menor impacto posible, sin embargo muchas herramientas no prestan aun las respectivas facilidades, por lo cual es necesario instalar un emulador, que muchas veces no ofrece altos grados de compatibilidad.

- **Estabilidad**

Una gran cantidad de herramientas de software libre, al no tener auspicios de grandes empresas, depende solamente de la buena voluntad y tiempo libre de la comunidad de desarrolladores que la soportan, lo cual ha provocado en muchos casos, productos que no terminan de ofrecer alto grado de estabilidad, para la absoluta confianza del usuario.



2.7. Problemática

2.7.1. Compatibilidad con Hardware

Actualmente existen muchas distribuciones de Linux, que ya integran una gran cantidad de drivers de mainboard principalmente, lo cual permite que al instalarlo no sea necesario contar con el cd de drivers entregado por el fabricante de la placa, de igual forma existen distribuciones que como Ubuntu están enfocadas también a ambientes de escritorio para usuarios finales dándole una amplitud más grande incluso para el reconocimiento de drivers en computadores portátiles.

Sin embargo en el momento de querer instalar dispositivos para usos más específicos como el caso de escáneres y cámaras Web, la instalación y reconocimiento de drivers se complica dependiendo de la marca y modelo, dando como resultado en muchos casos, una larga búsqueda por Internet, y un alto tiempo en la configuración para su utilización.

2.7.2. Centros de Soporte

En el caso de que se requiera soporte para alguna herramienta en particular, la solución más sencilla es la búsqueda en Internet en una gran cantidad de foros, lo cual no es una atención personalizada de un técnico especializado, por lo que muchas veces no se encuentra la respuesta o solución esperada, de acuerdo a esto, son muy pocos los centros de soporte que puedan prestar



atención sobre cierto producto informático.

Solamente para el caso de productos auspiciados por grandes compañías, es más factible encontrar el soporte esperado, el cual muchas veces no es tan especializado dependiendo del problema que se haya presentado.



CAPITULO 3

CASO DE ESTUDIO: EMPRESA DE TELECOMUNICACIONES

En este capítulo se conocerá a la empresa internamente, su misión, visión y valores, también se describirá el problema que enfrenta al centralizar la información, de acuerdo a su situación actual.

Se realizará una propuesta de automatización, realizando un diagrama de los casos de uso a nivel general, además se conocer detalles técnicos de la herramientas con las que cuenta la empresa para el desarrollo del software.

3.1. Antecedentes

Con la finalidad de brindar un mejor servicio a todos los ecuatorianos, y conectar a todo el país con redes de telecomunicaciones de última tecnología, se crea la empresa de telecomunicaciones del estado, la cual tiene el mayor porcentaje de la cobertura total a nivel nacional, y que en los últimos años ha realizado fusiones estratégicas, con otras empresas a nivel regional para dar una mayor cobertura y potencializar los servicios de telefonía fija e Internet.



Además la fusión con la empresa de servicios móviles, y los planes de ofrecer servicios de telefonía IP, para transmisión de datos, Internet y televisión, permitirá potenciar la cartera de productos enfocando los esfuerzos empresariales en el empaquetamiento de servicios y en convergencia de tecnologías.

3.1.1. Misión

Nos conectamos a todo el mundo integrando nuestro país a la nueva tecnología, mediante la provisión de soluciones de Internet y soluciones Web innovadoras, con recurso humano capacitado.

3.1.2. Visión

Ser la empresa líder del país, por la eficiencia en la integración de soluciones y la atención que ofrece a sus clientes.

3.1.3. Valores Empresariales

- Trabajo en equipo
- Integridad
- Compromiso con el servicio
- Cumplimiento de objetivos
- Responsabilidad social



3.2. Descripción del Problema

Ante la creciente tecnología, los servicios que ofrece la empresa de telecomunicaciones, y la cobertura que alcanza en todas las regiones del Ecuador, se presentan inconvenientes en la instalación del cableado para la red telefónica así como la fibra óptica en el caso del servicio de Internet.

Actualmente al tratar de recopilar la información y los documentos, para realizar un reporte unificado para el presidente Ejecutivo no cuadran los valores, ni los datos, que proporcionan las distintas áreas involucradas: Call Center, Seguridad Integral, Gerencia de Accesos, Gerencia Administrativa, Asuntos Procesales, pues no existe un repositorio común ni un flujo para realizar el seguimiento.

De esta forma a continuación se presenta la situación actual de la empresa para el manejo de información de siniestros que afecte a los abonados, proceso que se va a automatizar de acuerdo a la propuesta tecnológica presentada.

3.2.1. Situación Actual

Los incidentes de robo de cable son reportados al call center, ellos registran dichos incidentes en hojas electrónicas (Excel), las cuales reportan al área de seguridad integral mediante correo electrónico, igualmente las hojas



electrónicas son entregadas impresas y así se envía los documentos a cada departamento involucrado en cada etapa del proceso.

Los departamentos, con la información proporcionada, verifican en el lugar del siniestro la veracidad del incidente, reportando mediante hojas electrónicas enviadas como archivo adjunto por correo electrónico al área de operaciones.

Una vez confirmado el siniestro se obtiene la información técnica sobre la infraestructura afectada lo cual se pasa al área de seguros para que realicen el análisis de la cobertura en el caso de que se verifique que existe cobertura. De esta forma cada área maneja su respectiva información sobre un mismo siniestro.

Para entregar un reporte solicitado por el gerente general, sobre los siniestros presentados en un lapso de tiempo, se consolida la información por parte de las áreas involucradas.

Por tal motivo se solicita que estos ingresos de información lo realice cada una de estas áreas indicadas.



3.2.2. Propuesta de Automatización

En base a lo indicado es necesario el desarrollo de un sistema informático para registrar y dar seguimiento cuando sucede el siniestro de Robo de Cable. El sistema permitirá el registro de la información de los distintos casos de Robo de Cable.

En este proceso se involucran las áreas de Call Center, Seguridad Integral, Gerencia de Accesos, Gerencia Administrativa y Asuntos Procesales. De esta manera se podrá realizar un seguimiento de cada caso por parte de las autoridades de la empresa de Telecomunicaciones.

Garantizando que la información no se duplique ni se altere y se ingrese en el menor tiempo posible con la ayuda de mensajes de correo y SMS, que alerten a las áreas respectivas de sus gestiones.

3.2.3. Casos de Uso a Nivel General

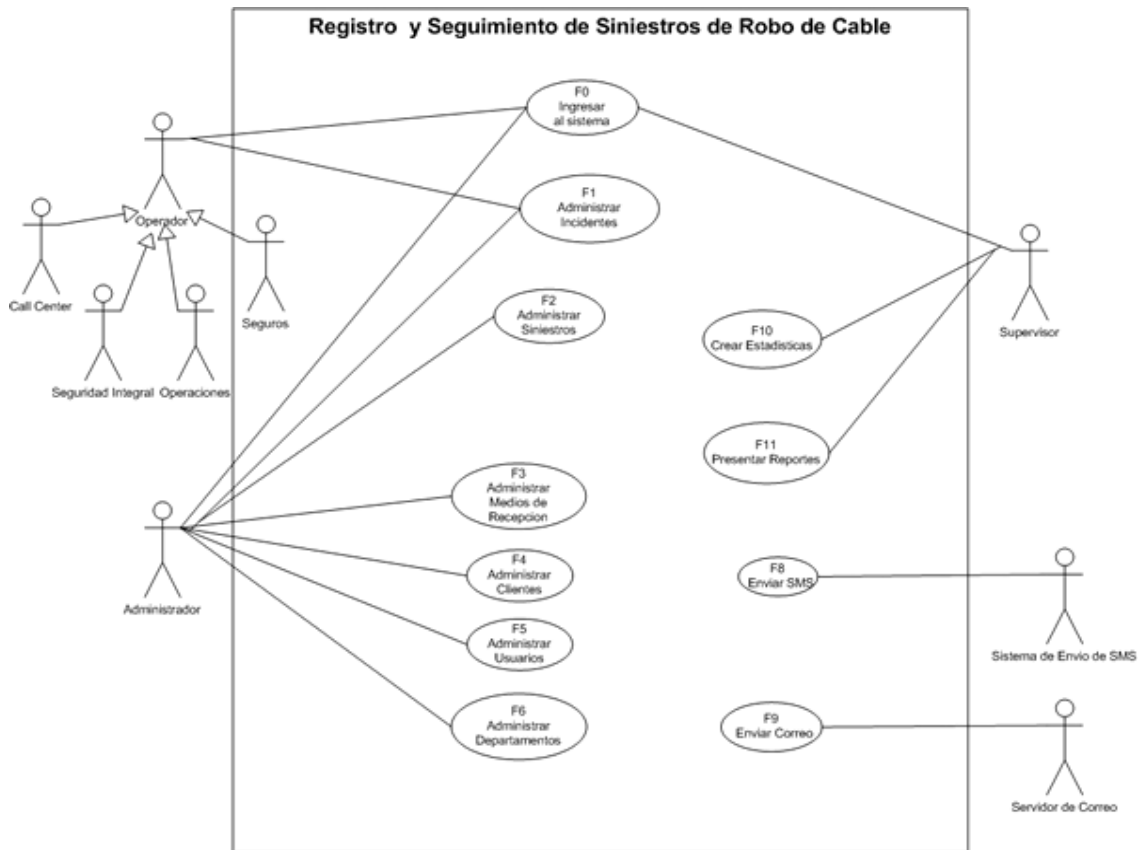


Figura 3.1: Caso de Uso a Nivel General

Elaborado por: Juan Acuña, Fabián Villavicencio P.

Los casos de uso a detalle se presentan en el siguiente Capítulo.

3.3. Herramientas a utilizar

Para la presente disertación se usaran las siguientes herramientas informáticas disponibles en la empresa de telecomunicaciones.

3.3.1. eZ Publish

Es un gestor de contenidos empresarial distribuido como open source, con la



cual se pueden desarrollar aplicaciones desde páginas Web personales hasta sitios corporativos en varios idiomas, de una forma dinámica, incluye acceso multiusuarios basado en perfiles, además de funciones para comercio electrónico y comunidades en línea.

Su característica principal es la habilidad de crear varios sitios desde una sola instalación de eZ Publish.

La administración se la realiza vía Web browser, cuenta con un editor de texto que permite dar formato al contenido, sin necesidad de altos conocimientos en HTML, incluye también versionamiento del contenido, librería de archivos de formato de audio y video, además de la posibilidad de subir varios archivos a la vez.

eZ Publish para la publicación utiliza un componente llamado Website Toolbar, que es una barra de herramientas que se despliega directamente en el browser una vez que el usuario inicia sesión, dando la opción de publicar, mover, reemplazar e incluso traducir el contenido.

Presenta también extensiones que permiten incrementar la funcionalidad, siendo así un sistema muy flexible para el desarrollo de componentes y bibliotecas.



- **Requerimientos**
- **Plataforma de Referencia**

Para la ejecución del sistema se presenta a continuación las características para una plataforma más segura y optimizada para proyectos críticos, la cual está certificada y probada por eZ Publish Enterprise¹.

Sistema Operativo	<ul style="list-style-type: none">• Debian 5.0. Linux 2.6
Servidor Web	<ul style="list-style-type: none">• Apache 2.2.14
DBMS	<ul style="list-style-type: none">• MySQL 5.0.51^a
PHP	<ul style="list-style-type: none">• PHP 5.2.12

- **Plataformas Aprobadas**

Para la ejecución de eZ Publish, se describe a continuación una lista de plataformas adecuadas, para un correcto funcionamiento².

Sistema Operativo	<ul style="list-style-type: none">• Linux 2.6• Opensolaris• Solaris 10 Intel con Sun Webstack• Windows 2008 (IIS 7)
Servidor Web	<ul style="list-style-type: none">• Apache 2.2.x• MS IIS 7

¹ <http://ez.no/eZPublish/Requirements>

² <http://ez.no/eZPublish/Requirements>



DBMS	<ul style="list-style-type: none">• MySQL 5.x• PostgreSQL 8• Oracle 11g
PHP	<ul style="list-style-type: none">• PHP 5.2.1 -> PHP 5.2.13• PHP 5.3
Manejador Grafico	<ul style="list-style-type: none">• ImageMagick >= 6.4.x• PHP extension GD2

- **Estructura Interna**

eZ Publish es una aplicación orientada a objetos, la cual se compone de 3 diferentes capas:

- **Librerías**

Corresponden a los principales elementos que constituyen el sistema, son en general clases en PHP, que pueden ser reutilizadas. No son de ninguna forma dependientes del kernel, aunque algunas librerías están interconectadas entre sí.



- **Núcleo (Kernel)**

Como su nombre lo indica es el núcleo del sistema, que cuida toda la funcionalidad en el más bajo nivel, como es el manejo de contenido, los accesos y los flujos, el núcleo se compone de varios motores que se basan y hacen uso de la librerías en general.

- **Módulos.**

Un módulo de eZ Publish pone a disposición una interfaz HTTP la cual puede ser usada para interacción con el sistema basada en Web.

Algunos módulos ofrecen una interfaz para la funcionalidad con el núcleo, mientras que otros son más o menos independientes del núcleo

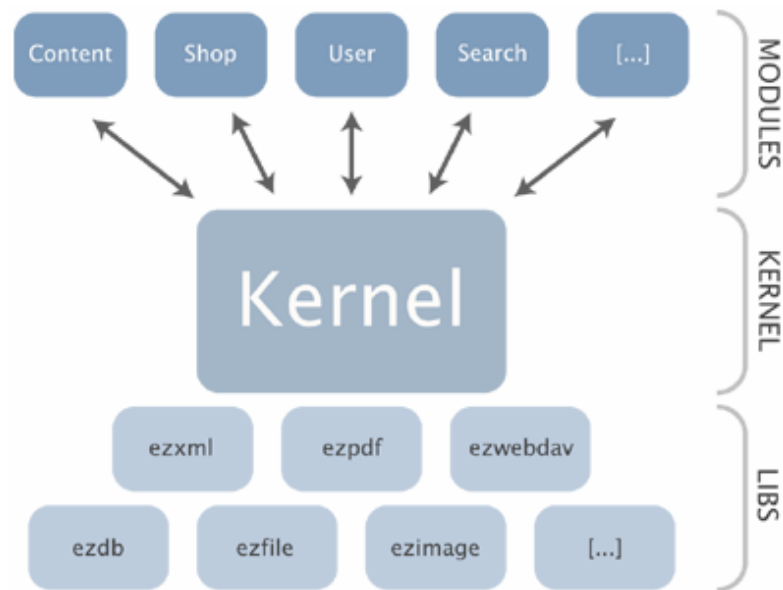


Figura 3.2: Estructura Interna eZ Publish¹

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Licenciamiento.**

eZ Publish es completamente open source sea bajo la GNU GPL o bajo la nueva licencia BSD. Además cuenta con licencias propietarias, para explotar todas las funcionalidades que presenta la herramienta, así enumeramos las posibilidades que se encuentran para las licencias:

- eZPUL, eZ Proprietary Use License, Es la licencia tradicional pagada para el usuario final.

¹ <http://doc.ez.no/eZ-Publish/Technical-manual/4.4/Concepts-and-basics/The-internal-structure-of-eZ-Publish>



- eZRL, eZ Rebrand License, permite a los proveedores de software vender y distribuir software propietario bajo su propia marca basada en eZ Publish.
- eZPEL, eZ Proprietary Extension License, permite a los proveedores de software vender extensiones propietarias de eZ Publish.

De esta forma, Telcoweb, adquirió eZ Publish con la licencia eZPUL, con lo cual en el plazo de un año la empresa tiene acceso a nuevos parches y correcciones sobre el código original de eZ Publish y también utilizar la licencia para un solo sitio Web.

3.3.2. MySQL

Como ya se había indicado en el capítulo anterior MySQL, es una de las bases de datos más populares y de mayor uso en el mundo por las múltiples características que presenta, siendo la base de datos de código abierto más popular.

Telcoweb al ser una empresa pública y de acuerdo al decreto de estado que obliga a las empresas a utilizar software libre, ha realizado un plan de migración para las distintas aplicaciones que han sido desarrolladas para cambiar a MySQL, y definir como un estándar para base de datos.



3.3.3. PHP

Uno de los lenguajes de programación mas utilizados para el desarrollo de sitios Web, pues permite potenciar los sitios Web estáticos, además de que el código es interpretado.

De igual forma la empresa para cumplir con el decreto de estado, ha escogido utilizar PHP, como lenguaje de programación para la mayoría de sitios Web. Además de que eZ Publish es desarrollada con PHP, y de esta forma permitiría una mejor integración entre las diferentes aplicaciones y las extensiones que distribuye la herramienta.

3.3.4. PowerDesigner

Es una herramienta de modelamiento empresarial, que corre nativamente como aplicación bajo Windows y de forma de plugin bajo eclipse.

Aunque PowerDesigner sea una herramienta bajo una licencia propietaria, es usada en la empresa de telecomunicaciones, por todas las características que ofrece, así como por sus años en el mercado convirtiéndola en una de las herramientas más usadas en el mundo, pues permite diseñar bases de datos soportando la mayoría de sistemas gestores de base de datos.



3.3.5. Dreamweaver

Es una aplicación para el desarrollo Web, que soporta diferentes tecnologías, con ciertas funciones especializadas en PHP y MySQL.

Telcweb adquirió el software justamente por la especialidad que ofrece en el desarrollo de aplicaciones bajo PHP y MySQL, así como las características de sincronización, transferencia de archivos, la previsualización de sitios Web en los diferentes Web browsers instalados así como en adobe device central, especializado en el desarrollo de sitios Web para dispositivos móviles, todo esto bajo su editor estilo WYSIWYG.

3.3.6. Apache

Es un servidor Web utilizado en plataformas de aplicaciones como LAMP y WAMP, mezcla que definen la infraestructura de un servidor Web.

Para la empresa de telecomunicaciones, la decisión de utilizar Apache, se debió primero a la recomendación técnica de eZ Publish, al ser desarrollada con PHP, debido a su bajo costo de adquisición e implementación, representando un conjunto de soluciones para un servidor de aplicaciones estable que permita el correcto rendimiento de las aplicaciones desarrolladas con eZ Publish.



CAPITULO 4

ANALISIS Y DISEÑO DEL SISTEMA

En este capítulo se conocerá la definición de SRS¹, y su función en el desarrollo de software, así como se pondrá en detalle los casos de uso principales, que se utilizan para las funcionalidades centrales del sistema, como Ingresar al sistema y Administrar Incidentes, también se presentara el diagrama de actividades que es un complemento de los casos de uso.

Además se conocerá el prototipo del sistema mediante pantallas para las opciones del menú, así como los campos en cada pantalla que serán ingresados en cada etapa del proceso.

Por último se presentará las características técnicas mínimas necesarias para el correcto rendimiento de la aplicación web, tanto de hardware y software a nivel del servidor y de las estaciones de trabajo para el cliente final.

4.1. Especificación de Requerimientos de Software (SRS)

Es la descripción de los requerimientos de un sistema, que se realiza previo al trabajo de diseño y desarrollo, define de forma precisa el paquete informático y asegura tanto al cliente como al área de sistemas, los

¹ Software Requirements Specification



requerimientos desde esa perspectiva.

El documento SRS presenta en un lenguaje explícito y claro todas las funciones y características de un sistema, además debe definir el alcance y las limitaciones que el sistema debe soportar.

Es importante considerar que el SRS contiene también los requerimientos funcionales, conocidos como los casos de uso, que definen las interacciones que tendrán el cliente con el sistema, además de los requerimientos no funcionales, que definen en cambio restricciones en el diseño y la implementación del sistema.

Un documento bien elaborado de SRS, es la clave para alcanzar 4 metas:

- Provee una retroalimentación con el cliente.
- Separa el problema en pequeños componentes
- Sirve como una entrada para el SDS¹.
- Es un documento para la validación y pruebas sobre el sistema.

El SRS es realizado en las primeras etapas del desarrollo de software, en donde la información proporcionada es agrupada de acuerdo a los requerimientos que se necesitan que son los que definen la funcionalidad y

¹ Software Design Specification



a los que no, que son los requerimientos no funcionales.

El levantamiento de información se lo puede realizar mediante visitas, entrevistas, cuestionarios y a veces a través análisis del retorno de la inversión.

4.2. Diagramas de Casos de Uso

A continuación procederemos a presentar los casos de uso a detalle, de acuerdo a la Figura 3.1 del capítulo anterior.

4.2.1. Ingresar al Sistema

- **Identificador:** F0
- **Descripción:** Este caso de uso se utiliza siempre que se requiera ingresar al sistema, ya que presta todas las funcionalidades para la autenticación del usuario.
- **Actores:** Operador, Administrador, Supervisor.
- **Flujo Principal:**
 1. El actor inicia el aplicativo.
 2. El sistema presenta la ventana de acceso.
 3. El actor ingresa los datos de su usuario y password.
 4. El actor da clic sobre el botón Aceptar.
 5. El sistema valida la información ingresada, el usuario y el password.



6. El sistema presenta el menú principal.

- **Flujo Alterno**

1. No existe

- **Excepciones**

Causa	Acción
Usuario no valido	Verificar la información ingresada, el usuario y el password y volver a intentar.

4.2.2. Administrar Incidentes

- **Identificador:** F1

- **Descripción:** Este caso de uso se utiliza para administrar los incidentes registrados en el estado en que se encuentren, con la posibilidad de Ingresar, Actualizar, Eliminar y Consultar incidentes de acuerdo al actor, estos casos de uso se los vera más a detalle a continuación.

- **Actores:** Operador, Administrador

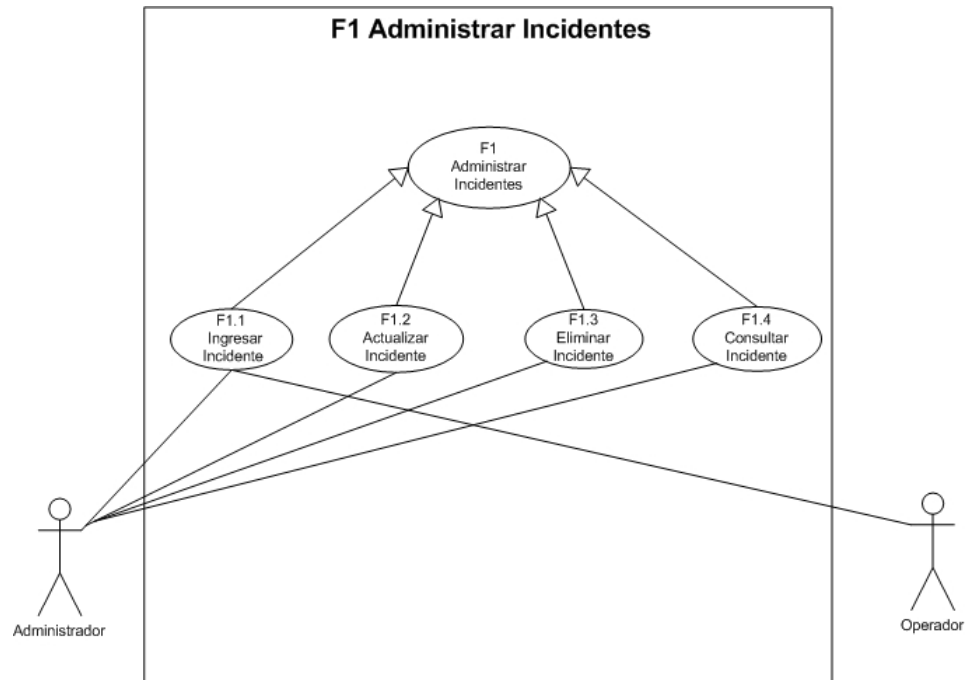


Figura 4.1: Caso de Uso Administrar Incidentes

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Ingresar Incidentes**

- **Identificador:** F1.1
- **Descripción:** Este caso de uso se utiliza para registrar nuevos incidentes, notificados por los clientes.

- **Actores:** Operador, Administrador.



Figura 4.2: Caso de Uso Ingresar Incidentes

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Flujo Principal:**

1. El actor ingresa a la opción en el menú de Registro de incidentes.
2. El sistema presenta la pantalla con los campos para llenar con la información respectiva.
3. El sistema presenta el número de incidente que se va a registrar.
4. El sistema presenta la fecha y hora actual del sistema, datos con los que se va a ingresar el incidente.
5. El sistema realiza consultas para presentar valores en los campos de clientes, Medios de Recepción, Tipos de Incidencia.
6. El actor ingresa el código del cliente en el campo donde se solicita la información respectiva.
7. El sistema verifica la existencia del cliente en la base de datos con el código del cliente ingresado.
8. El sistema presenta los datos del cliente en pantalla, nombres, apellidos, número de teléfono, email.



9. El actor selecciona el medio de recepción
10. El actor ingresa la fecha y hora del posible incidente.
11. El actor ingresa el posible lugar del incidente.
12. El actor ingresa las observaciones.
13. El actor ingresa las acciones tomadas por el cliente.
14. El actor selecciona el tipo de incidente.
15. El actor presiona el botón grabar.
16. El sistema ingresa el incidente.

- **Flujo Alternativo**

1. El sistema presenta lo que se detalla en el caso de uso F4.1, Ingreso de Clientes.

- **Excepciones**

Causa	Acción
Seleccione el Medio de Recepción del Siniestro	Escoger un valor de la lista en el campo de Medios de Recepción.
Ingrese el posible lugar del siniestro	Llenar el campo de Lugar del Siniestro
Ingrese las acciones tomadas	Llenar el campo de Acciones Tomadas
Ingrese el tipo de incidente	Escoger un valor de la lista en el campo de Tipo de Siniestro.

- **Actualizar Incidentes**

- **Identificador:** F1.2
- **Descripción:** Este caso se utilizara para modificar los datos de los incidentes que ya han sido registrados, en cualquier estado en el que se encuentren.
- **Actores:** Administrador

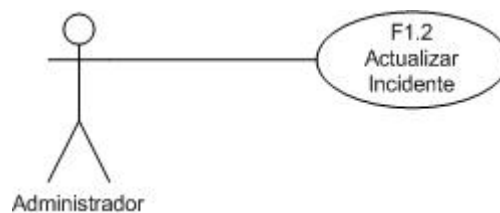


Figura 4.3: Caso de Uso Actualizar Incidentes

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Flujo Principal:**

1. El actor ingresa a la opción en el menú de Ver Incidentes.
2. El sistema presenta en pantalla los incidentes ingresados.
3. El actor ingresa a la opción para modificar el siniestro.
4. El sistema presenta en pantalla los datos previamente ingresados para el incidente.
5. El actor modifica los datos.
6. El actor presiona sobre el botón grabar.
7. El sistema modifica el incidente.

- **Flujo Alterno**

No aplica

- **Excepciones**

Causa	Acción
No existe el incidente	Ingresar Incidente.

- **Eliminar Incidentes**

- **Identificador:** F1.3

- **Descripción:** Este caso de uso se utilizara para eliminar un incidente.

- **Actores:** Administrador



Figura 4.4: Caso de Uso Eliminar Incidentes

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Flujo Principal:**

1. El actor ingresa a la opción en el menú de Ver Incidentes.
2. El sistema presenta en pantalla los incidentes ingresados.
3. El actor ingresa a la opción para modificar el siniestro.
4. El sistema presenta en pantalla los datos previamente ingresados para el incidente.

5. El actor presiona sobre el botón Eliminar.

6. El sistema elimina el incidente.

- **Flujo Alterno**

No aplica.

- **Excepciones**

Causa	Acción
El incidente no puede ser eliminado.	Eliminar primero información relacionada

• **Consultar Incidentes**

- **Identificador:** F1.4

- **Descripción:** Este caso de uso se utilizara para realizar consultas de incidentes.

- **Actores:** Administrador.



Figura 4.5: Caso de Uso Consultar Incidentes

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Flujo Principal:**

1. El actor ingresa a la opción en el menú de Ver Incidentes.

2. El sistema presenta en pantalla los campos para ingresar los



criterios de búsqueda.

3. El actor ingresa los parámetros de búsqueda.
4. El sistema presenta en pantalla los incidentes
5. El actor ingresa a la opción para Ver los datos del siniestro.

- **Flujo Alternativo.**

No aplica.

- **Excepciones**

Causa	Acción
La búsqueda no devolvió resultados	Ingrese el incidente.

4.3. Diagrama de Actividades del Proceso

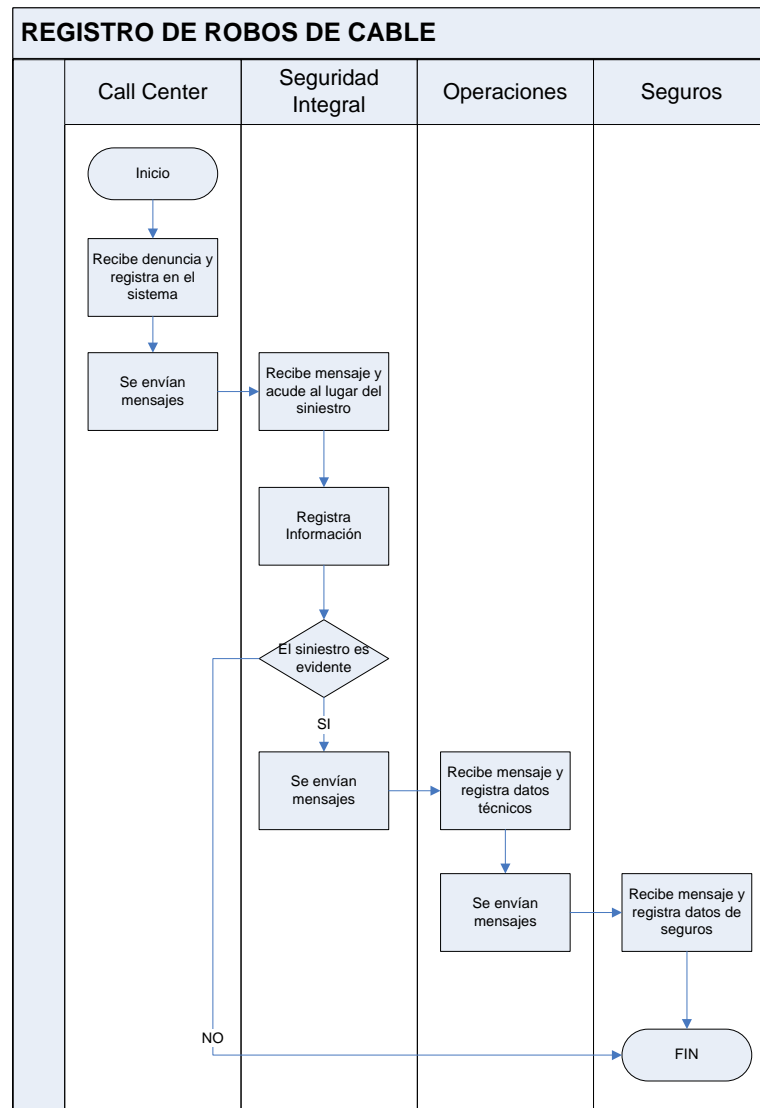


Figura 4.6: Diagrama de Actividades

Elaborado por: Juan Acuña, Fabián Villavicencio P.

4.4. Prototipo

4.4.1. Opciones del Menú

- **Opción Incidentes**



Figura 4.7: Opciones del Menú Incidentes

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Opción Mantenimiento de Parámetros**

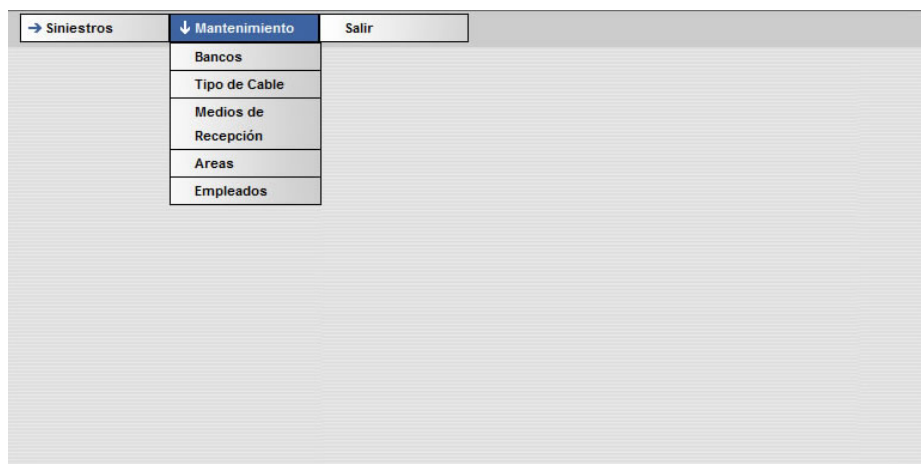


Figura 4.8: Opciones del Menú Mantenimiento

Elaborado por: Juan Acuña, Fabián Villavicencio P.

4.4.2. Pantallas

- **Usuario Call Center**

Call Center

Número de Caso Fecha Hora

Nombre de Denunciante

Telefono de Denunciante Medio de Recepción

Fecha de Siniestro Lugar de Siniestro

Observaciones

Acciones Tomadas

Origen de Denuncia

Figura 4.9: Prototipo de pantalla para el usuario de Call Center.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Usuario Seguridad Integral**

Seguridad Integral

Confirmación del siniestro SI NO

Observaciones

Figura 4.10: Prototipo de pantalla para el usuario de Seguridad Integral.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Usuario Operaciones**

Operaciones

Ruta	<input type="text"/>	Central	<input type="text"/>		
Ruta/Listones/Serie	<input type="text"/>	Número de Cables Robados	<input type="text"/>		
Longitud de Cable 1	<input type="text"/>	Pares de Cable 1	<input type="text"/>	Tipo de Cable 1	<input type="text"/>
Longitud de Cable 2	<input type="text"/>	Pares de Cable 2	<input type="text"/>	Tipo de Cable 2	<input type="text"/>
Longitud de Cable 3	<input type="text"/>	Pares de Cable 3	<input type="text"/>	Tipo de Cable 3	<input type="text"/>
Número de Abonados Afectados	<input type="text"/>				
Observaciones	<input type="text"/>				

Figura 4.11: Prototipo de pantalla para el Usuario de Operaciones.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Usuario Seguros**

Seguros

Número GAC	<input type="text"/>	Fecha de Recepción GAC	<input type="text"/>
Valor de Siniestro GAC	<input type="text"/>	Cobertura	<input checked="" type="radio"/> SI <input type="radio"/> NO
Número de Aseguradora	<input type="text"/>	Fecha de Recepción Denuncia	<input type="text"/>
Fecha de Envío a Seguro	<input type="text"/>	Fecha de Recepción Reconocimiento	<input type="text"/>
Número de Cheque	<input type="text"/>	Valor del Cheque	<input type="text"/> USD
Valor del Siniestro	<input type="text"/>	Deducible	<input type="text"/>
RASA	<input type="text"/>	Anticipo de Pago	<input type="text"/>
Valor Indeminizado	<input type="text"/>		
Observaciones	<input type="text"/>		

Figura 4.12: Prototipo de pantalla para el Usuario de Seguros.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

4.5. Otros Requerimientos

4.5.1. Interfaces Externas

- **Envío de SMS**

- **Identificador:** F8
- **Descripción:** Este caso de uso se utilizara cuando el sistema realice el envío de mensajes de texto con notificaciones a los usuarios.
- **Actores:** Sistema de Envío de SMS, Sistema



Figura 4.13: Caso de Uso Envío de SMS

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Flujo Principal:**

1. El sistema consulta los números de los teléfonos celulares de los usuarios del sistema.
2. El sistema consulta la información para armar el mensaje.
3. El sistema envía el número de celular con el texto del mensaje al sistema de envío de SMS.
4. El sistema de SMS envía el mensaje de texto a los usuarios del sistema.
5. El sistema de SMS, envía una confirmación al sistema de Siniestros.

- **Flujo Alterno.**

El sistema ejecuta el caso de uso F9.

- **Excepciones**

Causa	Acción
El Mensaje de Texto no fue enviado	Volver a enviar el SMS desde el sistema de Incidentes.

- **Envío de Correos Electrónicos**

- **Identificador:** F9

- **Descripción:** Este caso de uso se utilizara cuando el sistema realice el envío de correos electrónicos con información sobre el incidente a los usuarios.

- **Actores:** Servidor de Correo Electrónico, Sistema

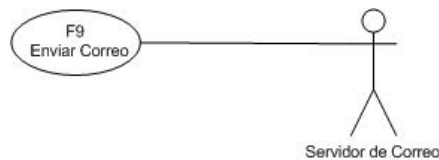


Figura 4.14: Caso de Uso Envío de Correos Electrónicos

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Flujo Principal:**

1. El sistema consulta las direcciones del correo electrónico a los usuarios del sistema.
2. El sistema consulta la información sobre el siniestro para enviarlo como texto en el correo.



3. El sistema envía los datos al servidor de correo electrónico.
4. El servidor de correo electrónico envía el correo a los usuarios del sistema.
5. El servidor de correo electrónico, envía una confirmación al sistema de Siniestros.

- **Flujo Alternativo.**

No aplica.

- **Excepciones**

Causa	Acción
El Mensaje de Correo no fue enviado	Volver a enviar el email desde el servidor de correo electrónico

4.5.2. Hardware

Para el correcto rendimiento del sistema los requerimientos mínimos de hardware tanto para el lado del servidor como para las estaciones de trabajo son las siguientes:

• **Servidor**

- Intel Xeon Dual Core 2.0 Ghz
- 4GB RAM
- 500GB Disco Duro



- **Cliente**
 - Procesador Pentium IV de 1.5 Ghz
 - Disco Duro de 40 Gb.
 - 1Gb en Memoria RAM
 - Ethernet NIC 100/100 Full Duplex
 - Monitor Resolución 1024 x 768

4.5.3. Software

El sistema fue desarrollado con PHP, por lo cual al ejecutarse mediante un browser, es necesario contar con los siguientes requerimientos de software tanto para el servidor como para la ejecución en las estaciones de trabajo.

- **Servidor**
 - PHP 5.1.6
 - MySql 5.1
 - PhpMyAdmin
 - Apache Server

- **Cliente**
 - Internet Explorer 7, Firefox Mozilla.



CAPITULO 5

DESARROLLO E IMPLEMENTACION DEL SISTEMA

En este capítulo conoceremos los diferentes métodos para realizar una instalación de eZ Publish, revisaremos en detalle y paso a paso el método Normal de Instalación, como método más común para todos los usuarios. Se presentará el Diagrama Entidad – Relación del sistema, el diccionario de datos, y ejemplos de la codificación orientada a objetos que se utilizó para el desarrollo, así como también se explicará cómo se personalizó el estilo de eZ Publish.

5. Métodos de Instalación de eZ Publish

eZ Publish puede ser instalado por 3 diferentes métodos:

- **Normal**

Este es el método más común y más recomendado, requiere de un ambiente apropiado previamente instalado con un servidor web y una base de datos. A continuación es necesario descargar una distribución de eZ Publish, descomprimirlo en una carpeta, para luego a través de un servidor web, iniciar el asistente de instalación, el cual realizara una serie de preguntas que automáticamente configurará eZ Publish.



- **Manual**

Este es un método para usuarios más experimentados, el cual no tiene un asistente de instalación, tan solo requiere un sistema que ya tenga instalado un servidor web y una base de datos. De igual forma eZ Publish necesita ser descargado para luego descomprimir en una carpeta en el servidor web, para así realizar la configuración realizando cambios en algunos archivos y haciendo cambios manualmente en la base de datos.

- **Automática**

Este método también conocido como kick-start, de igual forma es para usuarios más experimentados, es la forma automática del método de Instalación Normal, y está destinado para administradores del sistema que deseen realizar instalaciones pre configuradas, mediante poca interacción con el asistente de instalación Web.

De igual forma eZ Publish necesita ser descargado para luego descomprimir en una carpeta en el servidor web, pero en vez de ir paso a paso directamente en el asistente de instalación web, es necesario configurar el archivo kickstart.ini, para luego iniciar el asistente.

5.1.1. Asistente de Instalación

Es una herramienta que está diseñada para facilitar la instalación inicial del sistema. Puede ser iniciado utilizando un browser, para seguir con lo pasos



de instalación hasta que sean completados, para luego abrir la página principal del sistema.

El asistente de instalación no guarda ni modifica ningún dato, hasta la confirmación en el paso final, por cuestiones de seguridad, pues en cualquier momento a través de la URL, se pueden modificar estos valores, es así que se puede regresar a pasos previos para modificar configuración, sin alterar la información ingresada.

- **Página de Bienvenida**

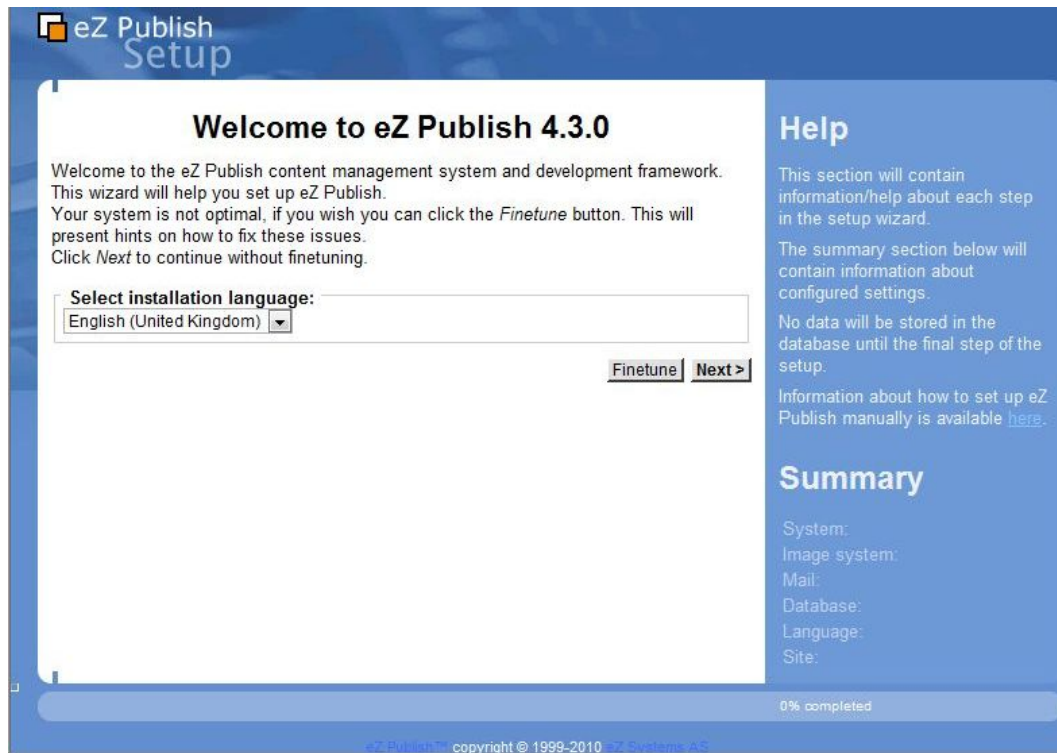


Figura 5.1: Página de Bienvenida

Elaborado por: Juan Acuña, Fabián Villavicencio P.

Es la página principal del asistente de instalación de eZ Publish, permite seleccionar el idioma que será utilizado en el proceso de instalación, el asistente realiza una revisión del sistema para conocer si es óptimo, para esto existe un botón en la parte inferior llamado Finetune o Ajuste Preciso.

- **Ajustes del sistema**



Ajuste preciso del sistema

Hay algunas cuestiones que deberían estar arregladas para tener un rendimiento óptimo. Una lista esta presentada abajo. Cada sección contiene una descripción y una solución sugerida.

Una vez que esos temas estan arreglados, puede hacer click en el botón *Ajuste preciso* para continuar. Se volverá a ejecutar la comprobación del sistema. Si todo esta correcto, el configurador pasará al paso siguiente. Si los temas no estan arreglados la página de ajuste preciso del sistema volverá a aparecer.

Si no quiere arreglar esos temas haz click en *Next*.

Cuestiones

- 1. PHP no registra variables de entorno**
PHP no está configurado para registrar variables de entorno en la variable global `$_ENV`. Algunas extensiones pueden estar usando `$_ENV`, recomendamos corregirlo a no ser que tengas control total sobre todas las extensiones que usas!
Para solucionarlo, edita el archivo `php.ini` y añade `E` a todas las configuraciones de `variables_order` y reinicia tu servidor web.
- 2. Falta la extensión cURL**
Se recomienda activar la extensión PHP cURL, sinó algunas funcionalidades que requieren un proxy o SSL no funcionarán.
Para activar las funciones PHP cUR debes compilar PHP con soporte para ello. Configura PHP con `--with-curl`. Hay más información sobre este tema disponible en php.net.
- 3. Programa ImageMagik perdido**

Ayuda

Se esta visualizando la página de ajuste preciso del sistema. Significa que unos temas podrían estar arreglado para mejorar el rendimiento o las funcionalidades.

Esos temas no necesitan estar arreglados. eZ Publish funcionará correctamente sin ellos.

Sumario

Sistema: OK
Sistema de imagen: ImageGD
Mail:
Base de datos:
Idioma:
Sitio:

Figura 5.2: Ajuste del Sistema

Elaborado por: Juan Acuña, Fabián Villavicencio P.

Este paso proporcionará más información sobre detalles de configuración que es necesario que sean arregladas para que eZ Publish tenga un rendimiento óptimo.

Al avanzar este paso en el asistente, si existen detalles importantes que corregir, se abrirá la página en donde se realizará una revisión del sistema.

- **Chequeo del sistema**

Chequeo del sistema

Hay algunas cuestiones importantes que han de resolverse. La lista se presenta abajo. Cada sección tiene una descripción y una solución recomendada o sugerida.

Una vez que los problemas / cuestiones se hayan solucionado, puedes hacer clic en el botón *Siguiente* para continuar. La comprobación de sistema arrancará de nuevo. Si todo es correcto, la configuración continuará hasta el estadio siguiente. Ante cualquier problema volverá a aparecer la página de comprobación del sistema.

Algunas cuestiones pueden ser ignoradas al marcar la(s) casilla(s) *ignora esta prueba*; aún así, no es recomendable.

También se puede ajustar el sistema de forma precisa, haz clic en *Ajuste preciso* en lugar de *Siguiente* si quiere ver las opciones de ajuste preciso.

Cuestiones

1. Configuración de la zona horaria

Estás usando la zona horaria predeterminada, UTC. Es importante que fijas tu zona horaria para asegurarte de que las fechas y horas se van a gestionar correctamente. Para hacer esto, fija la configuración `date.timezone` en `php.ini`.

Ejemplo de configuración `php.ini`:

```
date.timezone = "Europe/Oslo"
```

Para ver una lista de zonas horarias válidas, puedes ver [Lista de Zonas Horarias Soportadas](#) en la documentación del PHP.

De forma alternativa, si no tienes acceso para modificar `php.ini`, puedes cambiar la zona horaria en `config.php`. La zona fijada en `config.php` sobrescribirá la fijada en `php.ini`.

Ejemplo de configuración `config.php`:

```
<?php
date_default_timezone_set( "Europe/Oslo" );
?>
```

Si quieres conservar UTC como tu zona horaria, marca **Ignorar esta prueba** abajo

Ayuda

Se está mostrando las página del comprobador del sistema. Esto quiere decir que hay algunos problemas o cuestiones.

Estos temas se han de resolver o solucionar. Si no, eZ Publish no funcionará correctamente.

Los problemas están habitualmente relacionados con el sistema de archivos y se pueden solucionar con un corta y pega los comandos sugeridos en el sistema de intérprete de comandos.

Sumario

Sistema:
Sistema de imagen: ImageGD
Mail:
Base de datos:
Idioma:
Sitio:

Figura 5.3: Chequeo del Sistema

Elaborado por: Juan Acuña, Fabián Villavicencio P.

En el caso que los temas presentados no necesiten ser arreglados, se abrirá la página donde se configurará los parámetros correspondientes al servidor de correos.

- **Correo saliente**

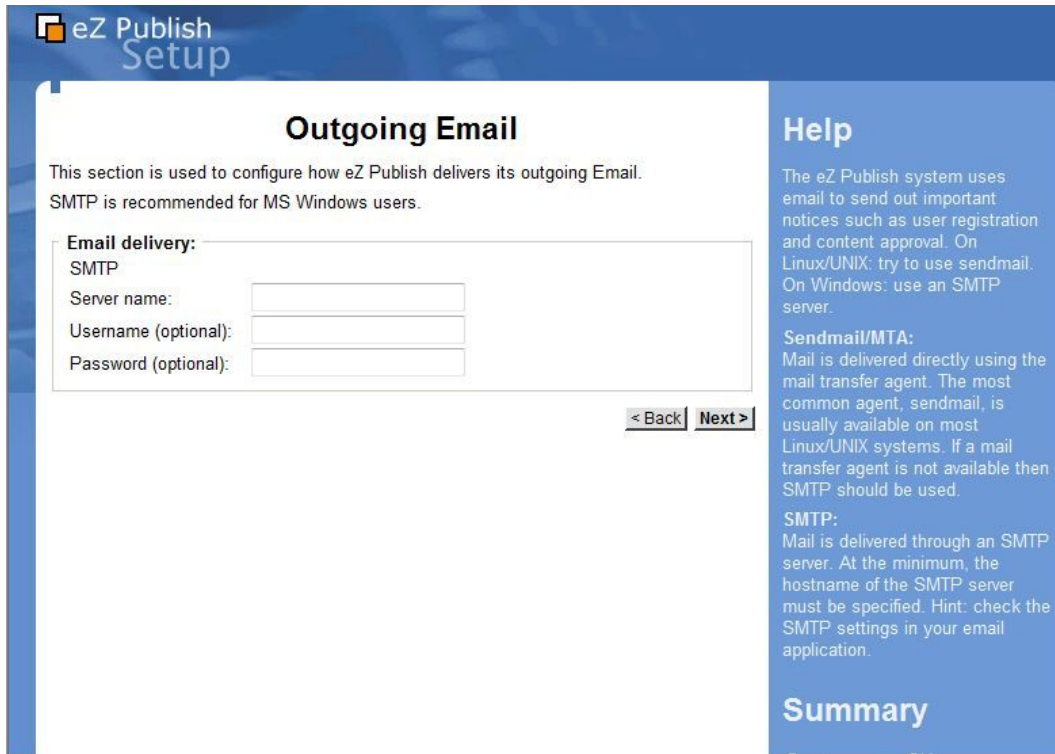


Figura 5.4: Correo Saliente

Elaborado por: Juan Acuña, Fabián Villavicencio P.

eZ Publish utiliza el correo electrónico para enviar mensajes como el registro de usuarios y la aprobación de contenido, en este paso se configura la forma como eZ Publish realizará la entrega de correo saliente, mediante dos opciones, la una en Linux con SendMail, o a través de un servidor SMTP en Windows.

- **Base de Datos**

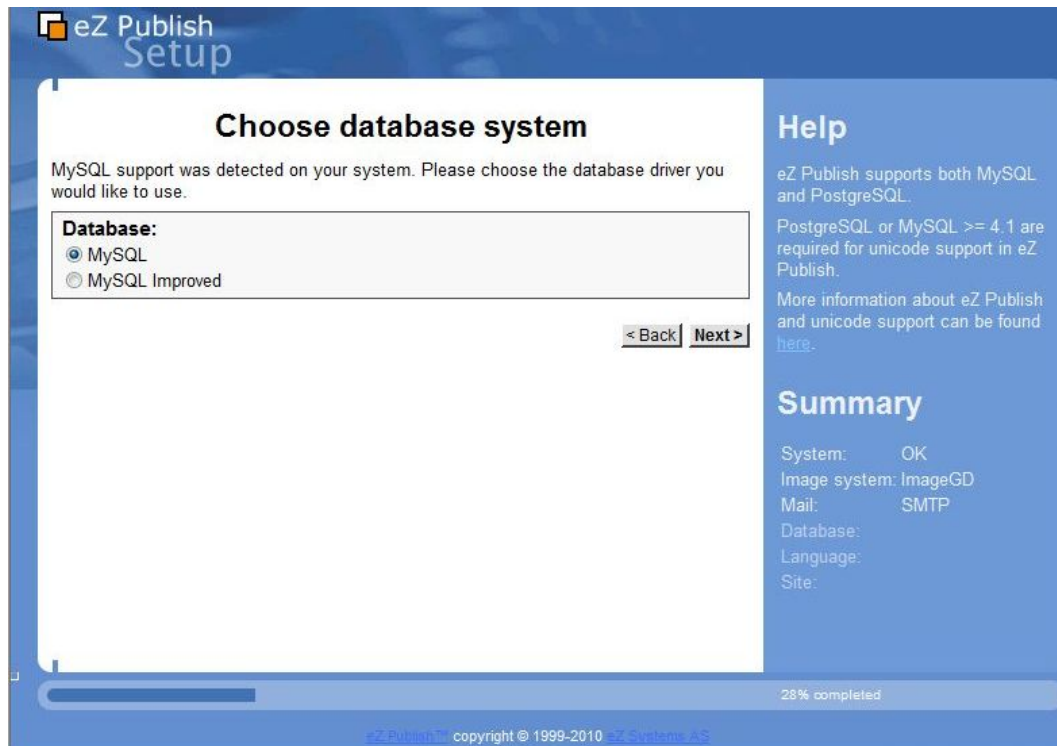


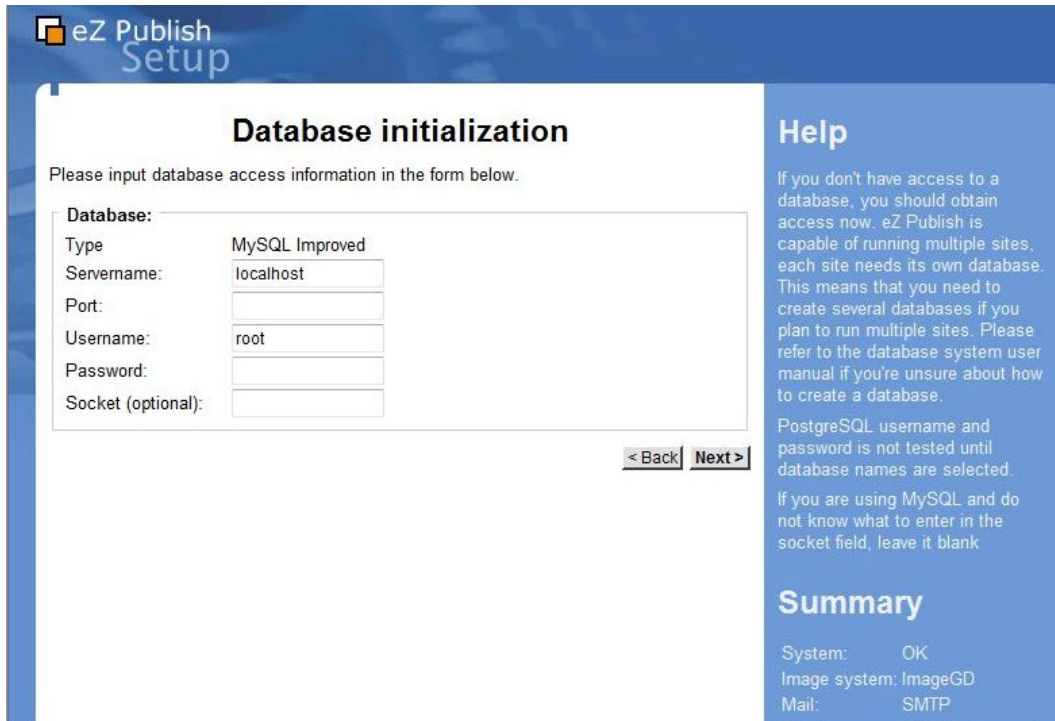
Figura 5.5: Base de Datos

Elaborado por: Juan Acuña, Fabián Villavicencio P.

El asistente detectará las bases de datos que están disponibles para PHP, en el caso de que MySQL y PostgreSQL estén instalados y soportadas por PHP, se abrirá un cuadro de diálogo que permitirá escoger la base de datos, en el caso de que detecte solamente el soporte para una base de datos, eZ Publish la escogerá, y permitirá configurar en el siguiente paso.

Si la extensión MySQLi está habilitada en PHP, la opción MySQL Improved se presentará para poder ser seleccionada, y en el caso de realizar la instalación sobre MySQL, es recomendado utilizar la segunda opción.

- **Inicialización Base de Datos**



eZ Publish Setup

Database initialization

Please input database access information in the form below.

Database:

Type	MySQL Improved
Servername:	<input type="text" value="localhost"/>
Port:	<input type="text"/>
Username:	<input type="text" value="root"/>
Password:	<input type="password"/>
Socket (optional):	<input type="text"/>

< Back | Next >

Help

If you don't have access to a database, you should obtain access now. eZ Publish is capable of running multiple sites, each site needs its own database. This means that you need to create several databases if you plan to run multiple sites. Please refer to the database system user manual if you're unsure about how to create a database.

PostgreSQL username and password is not tested until database names are selected.

If you are using MySQL and do not know what to enter in the socket field, leave it blank

Summary

System:	OK
Image system:	ImageGD
Mail:	SMTP

Figura 5.6: Inicialización de la Base de Datos

Elaborado por: Juan Acuña, Fabián Villavicencio P.

En este paso es necesario proveer información del nombre del servidor donde se ejecuta el motor de base de datos, así como el usuario y la contraseña para realizar la conexión, el asistente no pasará a la siguiente pantalla mientras no compruebe y realice una conexión a la base de datos en el caso de MySQL, para el caso de PostgreSQL, se verificará la información ingresada en un paso más adelante.

Para el caso de Oracle y aunque la extensión de eZ Publish para soportar la conexión a la base de datos este instalada, el asistente no permitirá utilizar

esta opción. Esta configuración se la debe realizar manualmente como se describe en la respectiva documentación.

- **Soporte del Idioma**

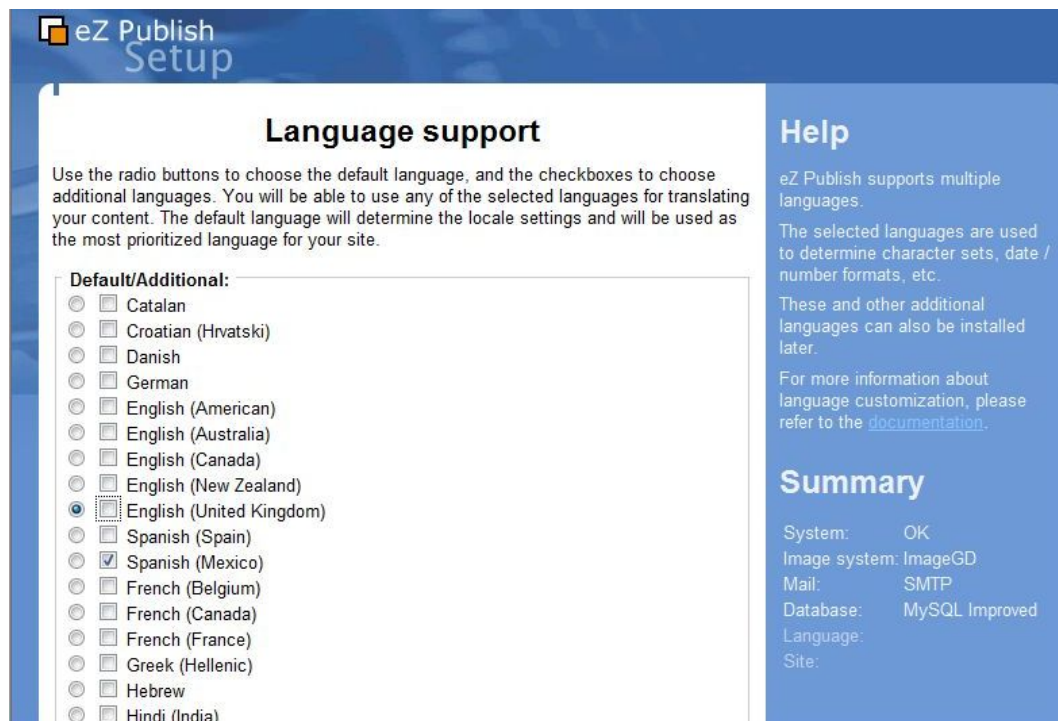


Figura 5.7: Soporte del Idioma

Elaborado por: Juan Acuña, Fabián Villavicencio P.

En este paso el asistente le permitirá escoger el lenguaje de configuración para el sitio que se va a instalar, el asistente automáticamente selecciona el idioma de acuerdo al país que está configurado para el explorador por default.

Además del lenguaje por defecto que se escoge para la instalación el asistente presenta la opción para escoger los lenguajes adicionales, lo cual

es opcional, esto se lo realiza con el radio button para el lenguaje por defecto, y con el checkbox para los lenguajes adicionales.

Todos los lenguajes serán añadidos al sistema, y permitirán posteriormente al finalizar el asistente, crear y traducir contenido con todos estos lenguajes instalados.

- **Selección del Sitio**

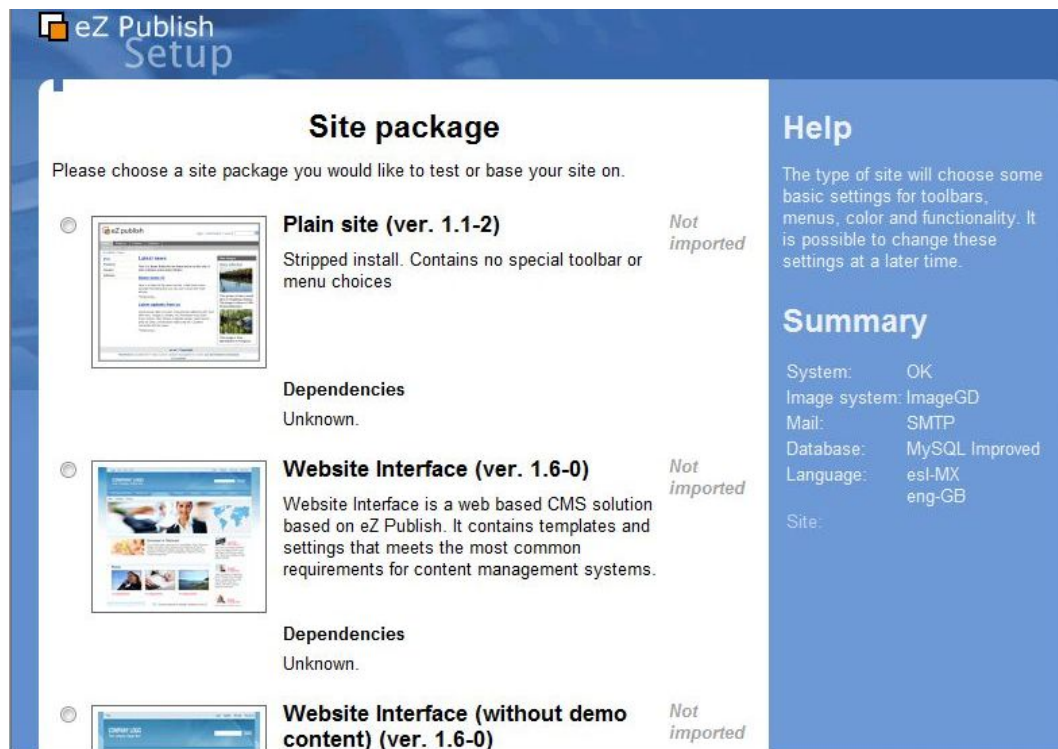


Figura 5.8: Selección del Sitio

Elaborado por: Juan Acuña, Fabián Villavicencio P.

En este punto el asistente permitirá escoger entre los diferentes paquetes que el asistente obtiene de repositorios remotos e internos, dando la opción

de escoger uno. Estos paquetes están destinados a proveer ejemplos para propósitos de demostración y aprendizaje. Sin embargo también es posible partir de estos ejemplos para realizar desarrollos de sitios más robustos.

Al seleccionar la opción en la pantalla, el asistente automáticamente descargará el paquete y toda la lista de los paquetes dependientes, que han sido exitosamente importados mostrándolos en la siguiente pantalla.

- **Lista de paquetes instalados**

Site package

Package 'ezflow_site' and it's dependencies have been downloaded successfully. Press 'Next' to continue.

eZ Flow (ver. 2.1-0) *Imported*

The eZ Flow extension to eZ Publish enables editors to build complex page layouts and pre-plan the publication schedule to ensure a constant flow of rich content.

Dependencies

- ezwt_extension (ver.1.2): *Imported*
- ezstarrating_extension (ver.1.1): *Imported*
- ezgmaplocation_extension (ver.1.1): *Imported*
- ezwebin_extension (ver.1.6): *Imported*
- ezflow_extension (ver.2.1): *Imported*
- ezflow_classes (ver.2.1): *Imported*
- ezflow_democontent (ver.2.1): *Imported*
- ezflow_design (ver.2.1): *Imported*

Help

The type of site will choose some basic settings for toolbars, menus, color and functionality. It is possible to change these settings at a later time.

Summary

System: OK
Image system: ImageGD
Mail: SMTP
Database: MySQL Improved
Language: esl-MX, eng-GB
Site: ezflow_site

< Back | Next >

50% completed

Figura 5.9: Lista de paquetes instalados

Elaborado por: Juan Acuña, Fabián Villavicencio P.



En este paso se presentará el paquete y sus dependencias instaladas, que son extensiones que ofrecen funcionalidades para satisfacer distintas necesidades del usuario.

eZ Publish es un Framework, que permite el desarrollo de nuevas extensiones, para añadir al paquete de instalación normal.

- **Configuración de acceso al sitio**

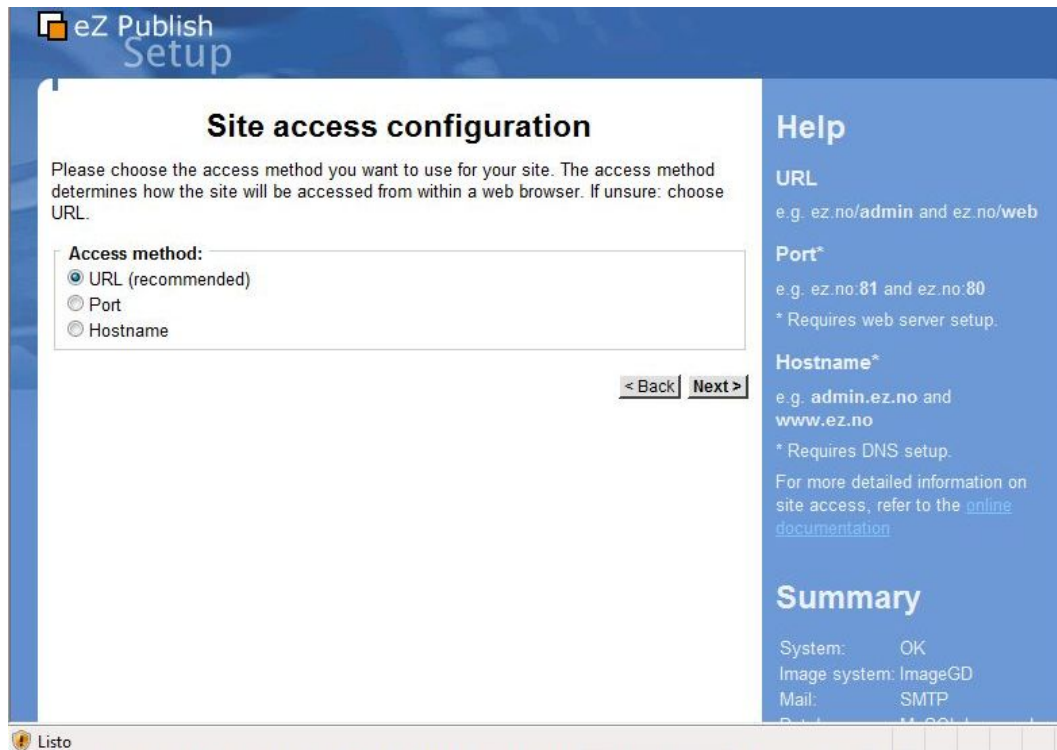


Figura 5.10: Configuración de acceso al sitio.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

Este paso permite la configuración del método de acceso que se debería usar una vez que eZ Publish reciba una solicitud, para lo cual existen tres opciones:

URL:

Cuando este método es usado, eZ Publish selecciona el sitio al cual debería acceder basado en el contenido de la URL, específicamente en el texto que viene luego de index.php, esta es la opción por defecto y la más genérica pues no requiere configuración adicional.

PORT:

En esta opción eZ Publish selecciona el sitio al cual debe acceder basado en el puerto que está unido al nombre del host, esta opción requiere configuración adicional en el web server y en el firewall.

HOST NAME:

Para esta opción cada sitio tendrá un único host name por ejemplo para el sitio destinado al público, `www.sitepublic.com` y para el sitio de administración `admin.sitepublic.com`, esta opción requiere configuración adicional en el DNS.

- **Detalles del sitio**

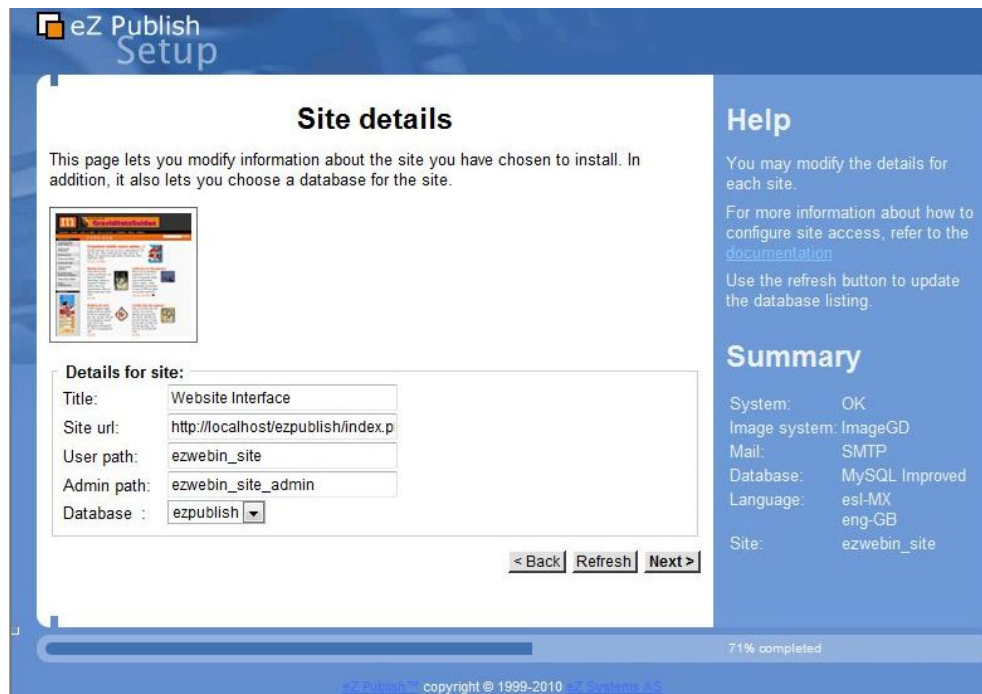


Figura 5.11: Detalles del sitio.

Elaborado por: Juan Acuña, Fabián Villavicencio P.



En este paso, se puede modificar información relacionada con el sitio que se está instalando, los valores de los campos de la ruta del usuario y de la ruta del administrador dependen del método de acceso que se escoja, si es por puerto, los valores serán numéricos, en el caso del método de acceso URL, los valores de los campos pueden contener solo letras, dígitos y guión bajo.

En el caso de seleccionar el método de acceso por host name, se permite ingresar caracteres adicionales como guiones, puntos y comas, sin embargo para este método no se acepta guiones bajos.

Para el campo Base de Datos se visualizara los nombres de las bases a las cuales tenga privilegios el usuario de acuerdo a la información ingresada en el punto 5.1.1.6, en la inicialización de la base de datos.

- **Administración del Sitio**

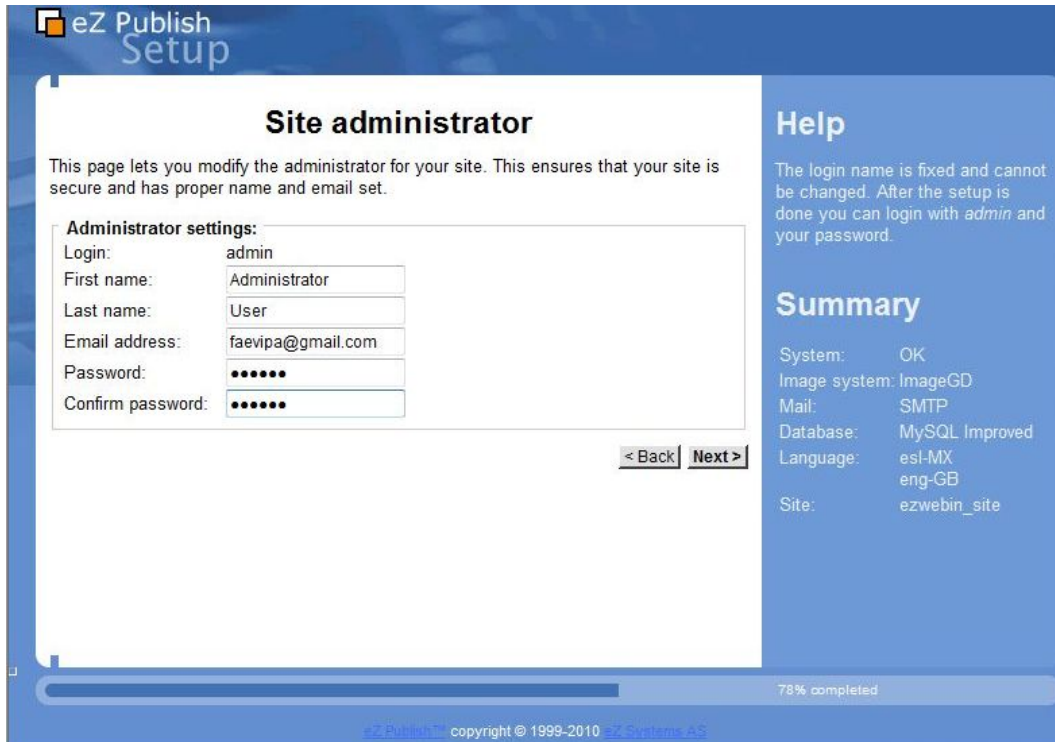


Figura 5.12: Administración del sitio.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

En esta sección, se puede realizar modificaciones básicas para poner más seguridad en el sitio, para proteger de accesos no deseados.

- **Seguridad del Sitio**

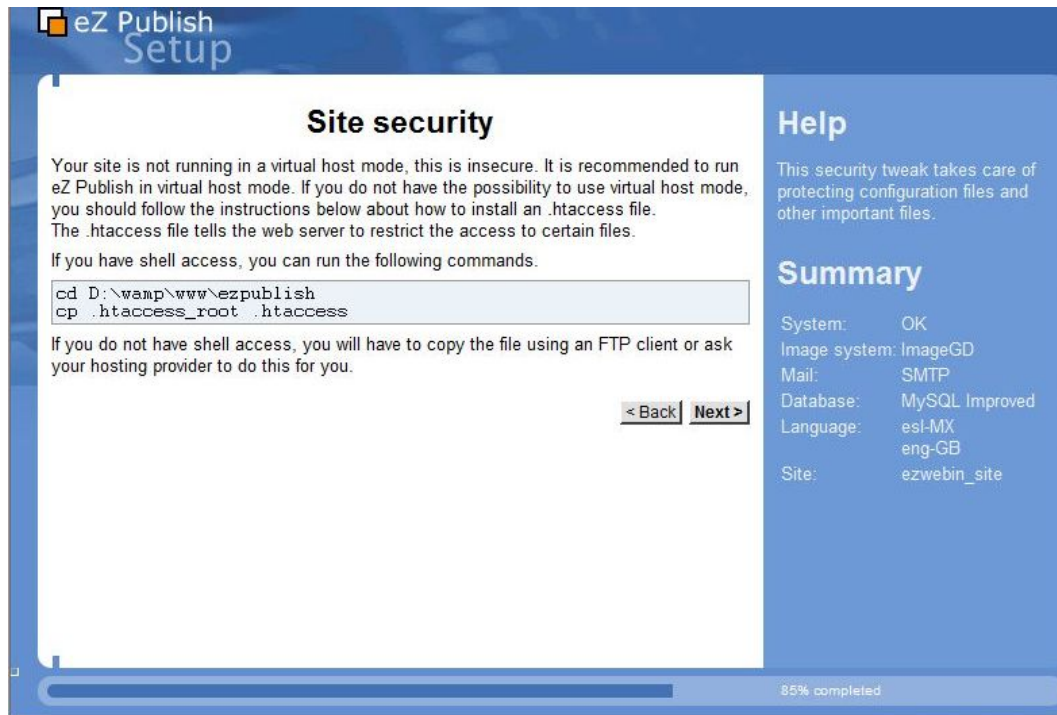
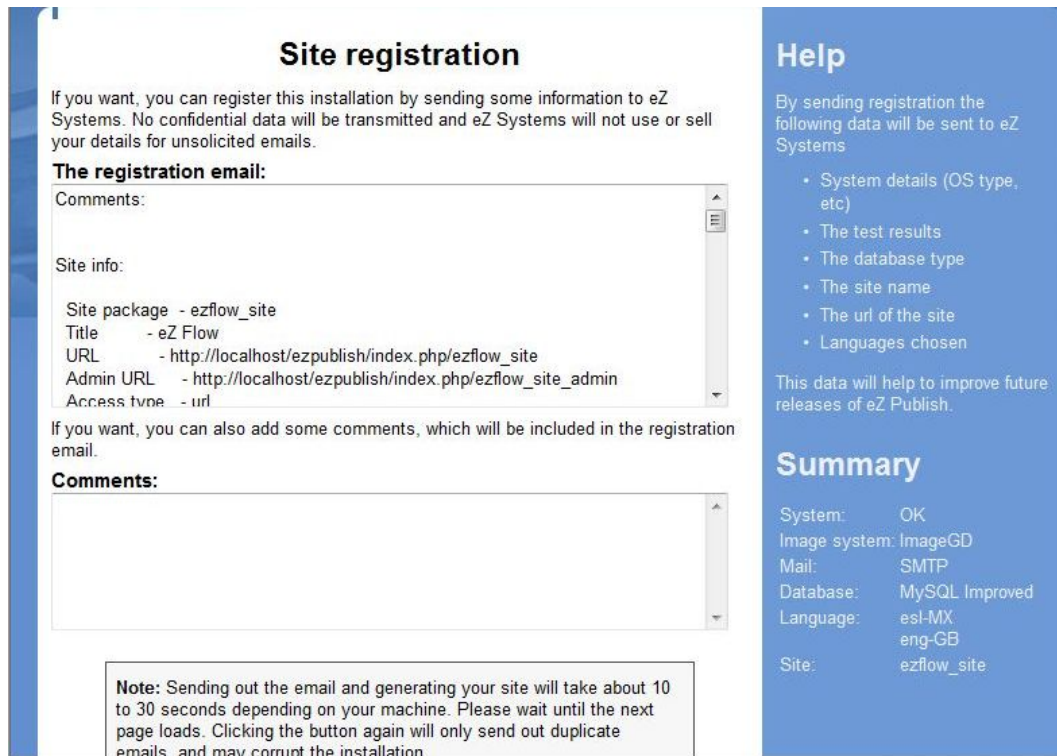


Figura 5.13: Seguridad del sitio.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

En el caso de que el asistente detecte posibles causas que puedan afectar a la seguridad del sistema, se presenta recomendaciones adicionales sobre el uso de un archivo de configuración htaccess, para restringir el acceso a directorios, restringir el acceso a IPs, manejar errores del servidor, crear redirecciones, en otras características.

- **Registro del Sitio**



Site registration

If you want, you can register this installation by sending some information to eZ Systems. No confidential data will be transmitted and eZ Systems will not use or sell your details for unsolicited emails.

The registration email:

Comments:

Site info:

Site package - ezflow_site
Title - eZ Flow
URL - http://localhost/ezpublish/index.php/ezflow_site
Admin URL - http://localhost/ezpublish/index.php/ezflow_site_admin
Access tvnoe - url

If you want, you can also add some comments, which will be included in the registration email.

Comments:

Note: Sending out the email and generating your site will take about 10 to 30 seconds depending on your machine. Please wait until the next page loads. Clicking the button again will only send out duplicate emails. and may corrupt the installation.

Help

By sending registration the following data will be sent to eZ Systems

- System details (OS type, etc)
- The test results
- The database type
- The site name
- The url of the site
- Languages chosen

This data will help to improve future releases of eZ Publish.

Summary

System: OK
Image system: ImageGD
Mail: SMTP
Database: MySQL Improved
Language: esl-MX
eng-GB
Site: ezflow_site

Figura 5.14: Registro del sitio.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

Este paso permite enviar al sistema información a eZ Systems, para cuestiones de estadísticas y mejoras al producto, por lo tanto datos no confidenciales serán transmitidos:

- Detalles del sistema
- Resultados de pruebas
- Tipo de base datos utilizados
- Nombre del sitio
- La URL del sitio

- Los idiomas seleccionados.
- **Fin de la instalación**

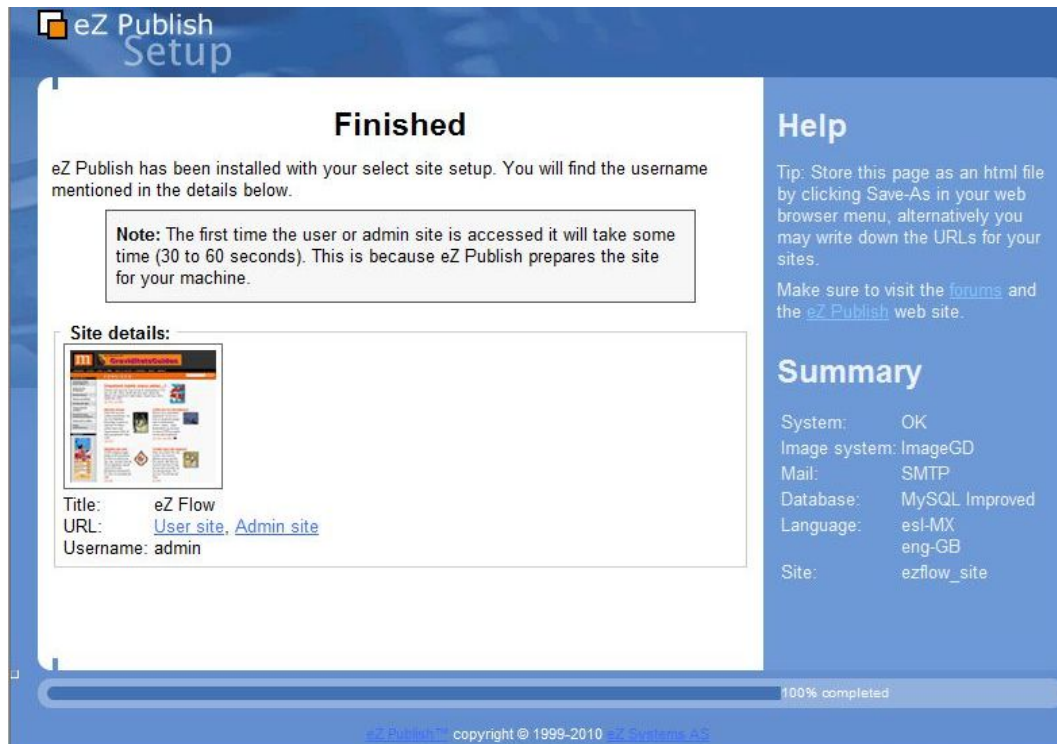


Figura 5.15: Fin de la instalación.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

Este es el paso final de la instalación, en el cual se presentan enlaces para acceder tanto al sitio del usuario, como al sitio del administrador.

Se puede configurar que se pueda reiniciar el asistente de instalación después de haber terminado, así una vez que se intente acceder al sitio, se vuelva a abrir la página de bienvenida del asistente.

5.2. Integración

eZ Publish presenta dentro de sus características, una muy importante, es que con una instalación de eZ Publish, se puede implementar varios sitios, para el caso de la Empresa de Telecomunicaciones, se ha implementado eZ Publish, como Portal Web, de cara a los usuarios y al público en general, así como Intranet para los empleados y usuarios internos.

A su vez eZ Publish de acuerdo a su estructura de contenidos, permite la definición de la estructura del sitio, pues los contenidos están incluidos en nodos



Figura 5.16: Relación Objeto Contenido - Nodos.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

Los nodos a su vez se organizan en estructuras de árbol.

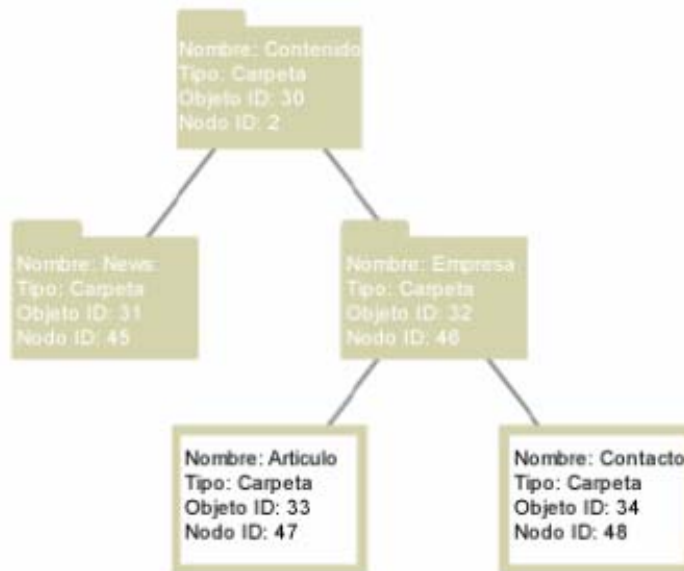


Figura 5.17: Árbol de Nodos.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

Los nodos se organizan en secciones, para dividir el árbol de nodos.

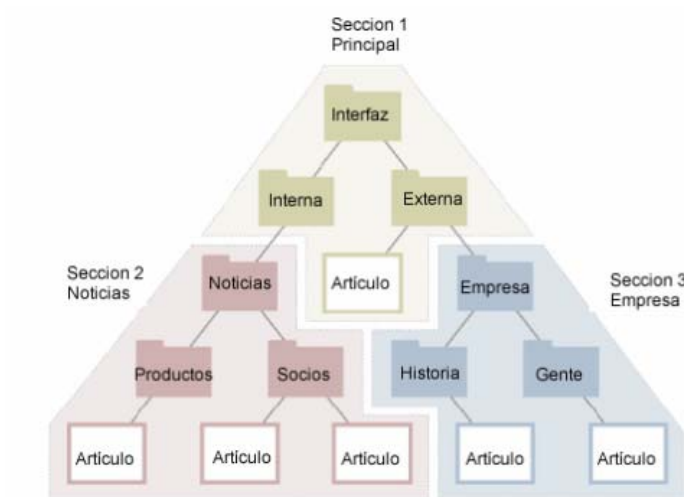


Figura 5.18: Secciones.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

De esta forma, la estructura del sitio puede contener ítems del menú, carpetas, artículos, comentarios, productos.

Para la integración con el Sistema de Autenticación de Usuarios, se creará un ítem en el menú, que será parte de la estructura del sitio, en este caso de la Intranet.

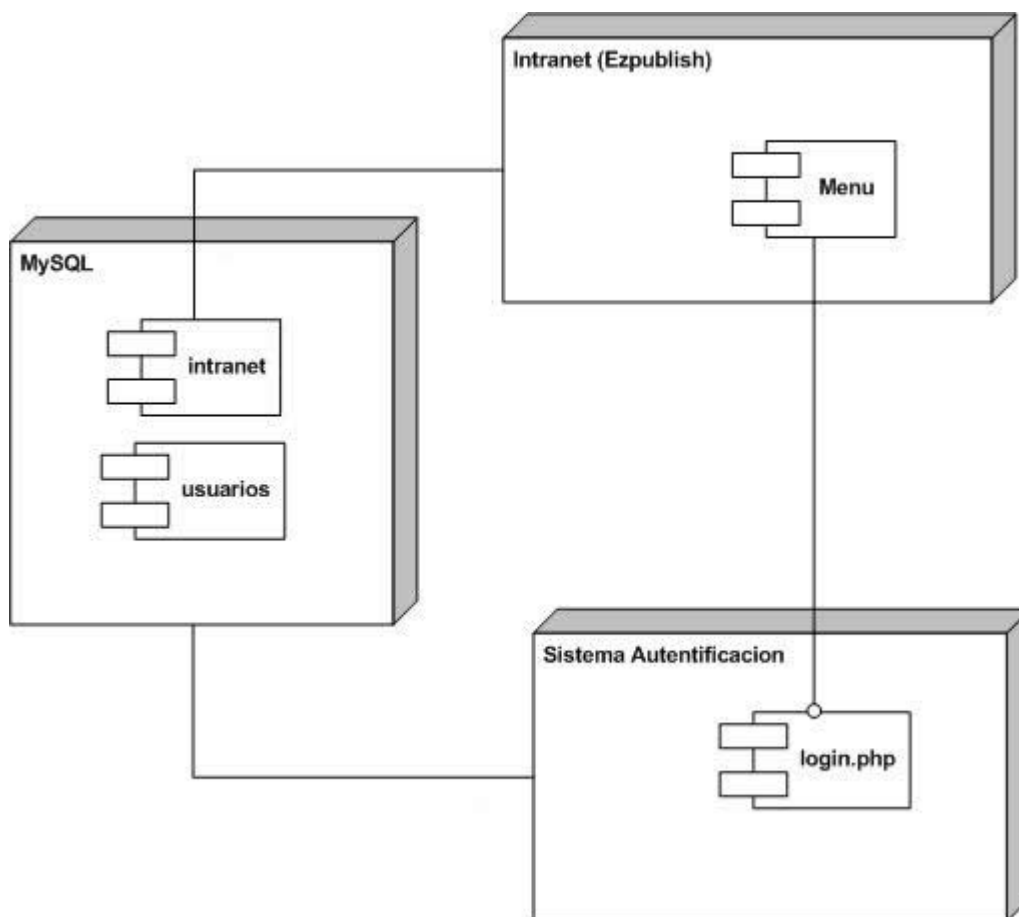


Figura 5.19: Diagrama Integración Intranet - Sistema.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

Dentro del diseño de la Intranet, una vez definida la estructura del sitio, se podrá visualizar el menú, en la parte superior del sitio Web.

5.3. Modelo Entidad Relación

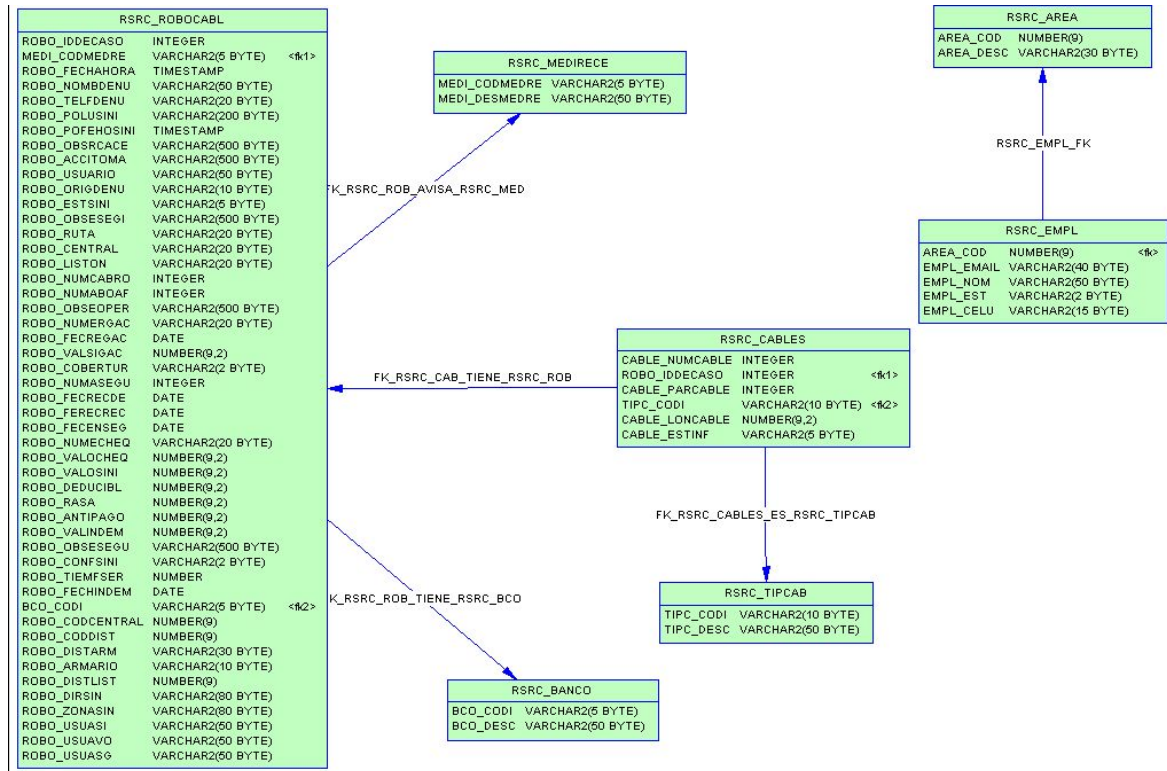


Figura 5.20: Modelo Entidad - Relación.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

5.4. Diccionario de Datos.

Nombre	Descripción
RSRC_ROBOCABL	Almacena la información de los incidentes ingresados en todas las etapas del flujo
RSRC_MEDIRECE	Almacena los valores de Medios de Recepción
RSRC_CABLES	Almacena la información técnica de los cables para cada incidente.
RSRC_EMPL	Almacena la información de los empleados a los cuales les va a llegar las notificaciones según vaya avanzando cada etapa en el flujo.
RSRC_TIPCAB	Almacena información técnica sobre los tipos de cable.
RSRC_BANCO	Almacena información sobre los bancos para la información ingresada por el área de seguros, para la cobertura en el caso de que la haya.
RSRC_AREA	Almacena información sobre las distintas áreas que intervienen en el flujo.

**5.4.1. Listado de Tablas del Sistema.****5.4.2. Tabla: RSRC_ROBOCABL**

ROBO_NUMABOAF	Número de abonados afectados	NUMBER(6)
ROBO_OBSEOPER	Observaciones Operaciones	VARCHAR2(500)
ROBO_NUMERGAC	Numero de GAC	VARCHAR2(10)
ROBO_FECREGAC	Fecha de recepción de GAC	DATE
ROBO_VALSIGAC	Valor del siniestro GAC	VARCHAR2(8)
ROBO_COBERTUR	Cobertura	VARCHAR2(2)
ROBO_NUMASEGU	Numero de aseguradora	VARCHAR2(20)
ROBO_FECRECDE	Fecha de recepción de denuncia	DATE
ROBO_FERECREC	Fecha de recepción de reconocimiento	DATE
ROBO_FECENSEG	Fecha de envío a seguro	DATE
ROBO_NUMECHEQ	Numero de cheque	VARCHAR2(15)
ROBO_VALOCHEQ	Valor de cheque	NUMBER(8,2)
ROBO_VALOSINI	Valor de siniestro	NUMBER(8,2)



ROBO_DEDUCIBL	Deducible	NUMBER(8,2)
ROBO_RASA	RASA	VARCHAR2(20)
ROBO_ANTIPAGO	Anticipo de pago	NUMBER(8,2)
ROBO_VALINDEM	Valor indemnizado	NUMBER(8,2)
ROBO_OBSESEGU	Observaciones Seguros	VARCHAR2(500)

5.4.3. Tabla: RSRC_MEDIRECE

Nombre	Descripción	Tipo de Dato
MEDI_CODMEDRE	Código de medio de recepción	VARCHAR2(2)
MEDI_DESMEDRE	Descripción de medio de recepción	VARCHAR2(30)

**5.4.4. Tabla: RSRC_CABLES**

Nombre	Descripción	Tipo de Dato
CABLE_NUMCABLE	Numero de Cable	INTEGER
ROBO_IDDECASO	ID de Caso	INTEGER
CABLE_PARCABLE	Par de Cable	INTEGER
TIPC_CODI	Tipo de cable	VARCHAR2(2)
CABLE_LONCABLE	Longitud del Cable	INTEGER
CABLE_ESTINF	Longitud Final del Cable	FLOAT

5.4.5. Tabla: RSRC_EMPL

Nombre	Descripción	Tipo de Dato
AREA_COD	Código de Área	INTEGER
EMPL_EMAIL	Email del Empleado	INTEGER
EMPL_NOM	Nombre del Empleado	INTEGER
EMPL_EST	Estado del Empleado	INTEGER
EMPL_CELU	Numero Celular del Empleado	FLOAT

5.4.6. Tabla: RSRC_TIPCAB

Nombre	Descripción	Tipo de Dato
TIPC_CODI	Código del Tipo de Cable	VARCHAR2(10)
TIPC_DESC	Descripción del Tipo de Cable	VARCHAR2(50)

5.4.7. Tabla: RSRC_BANCO

Nombre	Descripción	Tipo de Dato
BCO_CODI	Código del Tipo de Cable	VARCHAR2(10)
BCO_DESC	Descripción del Tipo de Cable	VARCHAR2(50)

5.4.8. Tabla: RSRC_AREA

Nombre	Descripción	Tipo de Dato
AREA_COD	Código del Área	NUMBER(10)
AREA_DESC	Descripción del Área	VARCHAR2(50)

5.5. Codificación

A continuación se presentará las funciones a nivel general, desarrolladas bajo conceptos programación orientada a objetos y programación por capas. A nivel de eZ Publish, se presentará la codificación a nivel de la hoja de estilos, que define la plantilla para el sitio.

5.5.1. Registro y Seguimiento de Incidentes de Robo de Cable.

- **Conexión a la Base de Datos**

```
1 <?php
2 /*Se incluye la libreria para realizar la conexion
3 a la base de datos mediante la libreria ADODB.*/
4 include_once('../lib/adodb/adodb.inc.php');
5 require_once('../lib/adodb/includes/functions.inc.php');
6 /*Se define los parametros para realizar la conexion
7 a la base de datos indicada.*/
8 $dbdriver='mysql';
9 $server='localhost';
10 $user=usuario;
11 $password='clave';
12 $database='siniestro';
13 /*Se establece la conexión con la base de datos,
14 dando la opción de realizar un debug*/
15 $cnnBaseDatos= ADONewConnection($dbdriver);
16 $cnnBaseDatos->Connect($server, $user, $password, $database);
17 $cnnBaseDatos->debug = true;
18 ?>
```

Figura 5.21: Conexión a la Base de Datos.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Clases**

Dentro de una aplicación que interactúa con una base de datos se construye clases por cada una de las tablas manteniendo el nombre de la tabla. Los atributos de las clases son los mismos que los atributos de las tablas.

```
1 <?php
2 /*Se incluye el archivo de la conexión a la base de datos*/
3 require_once('../config/conexion/conex.inc.php');
4 /*Se define la clase que realizara la interaccion con la tabla
5 RSRC_AREA*/
6 class RSRC_AREA
7 {
8 // ATRIBUTOS
9
10 var $AREA_COD; //integer,9,Codigo Area
11 var $AREA_DESC; //text,30,Descripcion Area
12 |
13 }
14 ?>
```

Figura 5.22: Clases.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- Métodos

En cuanto a lo que tiene que ver a los métodos de las clases se ha colocado un nombre para identificar la acción que va a realizar la función con un nombre que identifique la tabla, enviando los parámetros que sean necesarios. Estas acciones son scripts, INSERT, UPDATE, DELETE, SELECT, sobre la tabla, para lo cual se define el query.

```
1 <?php
2 function fnInsertarRSRC_AREA($AREA_COD,$AREA_DESC)
3 {
4 global $cnnBaseDatos;
5 $strSql = sprintf("INSERT INTO RSRC_AREA(AREA_COD,AREA_DESC) VALUES (%s,%s)",
6 GetSQLValueString($AREA_COD,"integer"),
7 GetSQLValueString($AREA_DESC,"text"));
8 $rsRSRC_AREA=$cnnBaseDatos->Execute($strSql);
9 return ($rsRSRC_AREA) ? 1 : 0;
10 }
11 ?>
```

Figura 5.23: Métodos.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Servicios**

Para cada uno de los métodos de la clase se definen funciones, creando archivos que son identificados como servicios, los cuales serán utilizados dentro de la interfaz para cada pantalla.

```
1 <?php
2 require_once('./clases/rsrc_area.php');
3
4 function fnInsertarRSRC_AREA($AREA_COD,$AREA_DESC)
5 {
6     $objRSRC_AREA=new RSRC_AREA();
7     $rsRSRC_AREA=$objRSRC_AREA->fnInsertarRSRC_AREA($AREA_COD,$AREA_DESC);
8     return $rsRSRC_AREA;
9 }
10 |
11 ?>
```

Figura 5.24: Servicios.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

- **Interfaz para el usuario**

Para cada usuario y de acuerdo al perfil se presentara los incidentes almacenados en la base de datos, de acuerdo al estado en que se encuentre, aquí se utilizará los servicios, para realizar en este caso consultas y ciertas acciones dependiendo del perfil.

```
1 <?php
2 /*Se incluye el archivo donde se definio el servicio que contiene las funciones
3 que utilizan las funciones*/
4 |
5 require('./servicios/srsrc_roboabl.php');
6
7 /*De acuerdo al perfil se realiza la consulta de los incidentes dependiendo del estado*/
8
9 if ($sesPerfil=='OPERADORSI'){
10 $rsRSRC_ROBOCABL = fnSConsultarRSRC_ROBOCABL('ROBO_IDDECASO',3);
11 }
12 if ($sesPerfil=='OPERADORVO'){
13 $rsRSRC_ROBOCABL = fnSConsultarRSRC_ROBOCABL('ROBO_IDDECASO',4);
14 }
15 if ($sesPerfil=='OPERADORSG'){
16 $rsRSRC_ROBOCABL = fnSConsultarRSRC_ROBOCABL('ROBO_IDDECASO',5);
17 }
18 if ($sesPerfil=='ADMRSRC'){
19 $rsRSRC_ROBOCABL = fnSConsultarRSRC_ROBOCABL('ROBO_IDDECASO',6);
20 }
21 ?>
```

Figura 5.25: Interfaz para el usuario.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

De acuerdo al servicio que se utilice se utilizará los parámetros que retorne la función para desplegar los datos en pantalla, o para dar mensajes de aviso en caso de inserción, actualización o eliminación de datos.

CASO		REGISTRO		
#	MEDIO DE RECEPCION	USUARIO	FECHA	HORA
1	FIJO	Prevencion Fraude	22/09/2008	09:54:10
2	FIJO	Rodrigo Nicolas Guerrero Najera	08/12/2008	15:34:48
3	FIJO	Maria Paulina Jaramillo Correa	30/10/2009	09:11:37
4	FIJO	Call Center	04/09/2010	13:32:42

Figura 5.26: Datos presentados sobre el registro del Incidente.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

NOTIFICADOR DEL SINIESTRO		SINIESTRO					
NOMBRE	TELEFONO	LUGAR	FECHA/HORA	ACCIONES TOMADAS	ORIGEN	ESTADO	
FABIAN VILLAVICENCIO	2597159	PRUEBA	21/09/2008 20:00	PRUEBA	EXTERNO	FINALIZADO	CONSULTAR
JUANA CAMPOVERDE	022690445	LA ECUATORIANA. CIUDADELA	08/12/2008 03:27	SE CONFIRMA INFORMACION, SE INFORMA AL	EXTERNO	SEGURIDAD INTEGRAL	VERIFICAR
JUANA VITERI	087540722	MAÑOSCA Y VASCO DE CONTRERAS POR	30/10/2009 09:02	SE HA NOTIFICADO A ARMILET SR.	EXTERNO	FINALIZADO	CONSULTAR
JUAN ACUÑA	2972076	SAN CARLOS	04/09/2010 01:31	LLAMADA	EXTERNO	REQUIERE VERIFICACION	REGISTRAR

Figura 5.27: Datos sobre el incidente.

Elaborado por: Juan Acuña, Fabián Villavicencio P.

5.5.2. eZ Publish

Para realizar cambios en el diseño del sitio, una vez instalado mediante el asistente, se puede realizar modificaciones a nivel de archivos CSS.

- **classes-colors.css**

Sobre escribe los estilos definidos por el archivo content.css para todos los estilos relacionados con el contenido

- **site-colors.css**

Sobre escribe los estilos definidos por el archivo pagelayout.css para todos los estilos no relacionados con el contenido.

Ambos archivos están vacíos originalmente y están localizados en la carpeta donde se realizó la instalación de eZ Publish, dentro de la siguiente carpeta:

`var/storage/packages/ez_systems/eswebin_design_files_default_file/`

Se toma como referencia esta ubicación, asumiendo que la instalación de eZ Publish se la realizó utilizando la extensión ezwebin correspondiente al paquete Website Interface.

Se puede editar estos archivos existentes o de igual forma se puede crear dos nuevos archivos y guardarlos como `classes-colors.css` y como `site-colors.css`. Todas las imágenes relacionadas con el diseño se pueden guardar en una carpeta llamada `images`, y referirse a esta ruta dentro de los archivos `css` creados.

- **Personalizando el diseño de la página principal.**

El estilo de la página esta definido con el id "page", de esta forma en la hoja de estilo `site-colors.css` se puede personalizar el ancho, la imagen de fondo, por ejemplo:

```
1 div#page {  
2 margin:0pt auto;  
3 text-align:left;  
4 width:770px;  
5 }
```

Figura 5.28: Pagina Principal

Elaborado por: Juan Acuña, Fabián Villavicencio P.

De esta forma, el resto de estilos quedarían definidos por la hoja de estilos `pagelayout.css`.

Mediante estos conceptos también se puede modificar la apariencia del menú superior que está definido con el id “topmenu” para el tag html ``, aquí se puede modificar el fondo existente con uno nuevo, y cambiar el ancho del menú también.

```
1  div#topmenu ul {  
2  background-image url(images/mainmenu-bg.gif);  
3  background-repeat: repeat-x;  
4  height: 21px;  
5  }  
6  |  
7  div#topmenu ul, div#topmenu li {  
8  height: 21px;  
9  }
```

Figura 5.29: Apariencia Menú

Elaborado por: Juan Acuña, Fabián Villavicencio P.

También es importante cambiar el estilo del tag <a>, pues cada texto en el menú es un enlace.

```
1  div#topmenu li a {
2  background-color:transparent;
3  color:#4F5241;
4  display:block;
5  float:left;
6  height:15px;
7  line-height:10px;
8  border-right:none;
9  padding:2px 15px 0px;
10 text-decoration:none;
11 }
12
13 div#topmenu li.selected {
14 background-image:url(images/mainmenu-bg.gif);
15 }
16
17 div#topmenu li {
18 background-image:url(images/mainmenu-bg.gif);
19 border-right:1px solid #C3C4B6;
20 }
21
22 div#mainmenu a:hover, div#mainmenu li.current a:hover,
23 div#submenu a:hover, div#mainmenu li.current a.menuhover:hover,
24 div#mainmenu a.menuhover:hover, div#mainmenu li.current li a:hover {
25 background-color:#F15E22;
26 background-image:none;
27 color:#FFFFFF;
28 }
29
30 div#topmenu li a:hover {border-bottom:none;
31 background-color:#F15E22;
32 background-image:none;
33 color:#FFFFFF;
34 }
```

Figura 5.30: Enlaces del Menú

Elaborado por: Juan Acuña, Fabián Villavicencio P.



CAPITULO 6

CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones

- Se requiere de una metodología adecuada para el levantamiento de información en un desarrollo de software, para de esta manera lograr un buen análisis de requerimientos.
- Un administrador de contenidos o CMS es una aplicación que permite actualizar el contenido de un sitio Web. Además de esto, muchos desarrolladores tienen productos propios y a medida. Es decir, es un sistema de gestión de contenidos en línea que permite modificar la información rápidamente desde cualquier computadora conectada a Internet, simplificando las diferentes tareas de creación, distribución, presentación y mantenimiento de contenidos en la red.
- La definición de procesos ayuda notablemente en el desarrollo de software.
- La implementación de un CMS es mucho más óptima y baja de costos, a diferencia de la manera convencional de realizar el sistema/portal.



- El CMS eZ Publish es un sistema de gestión de contenidos de código abierto (Open Source) que permite el desarrollo de aplicaciones web dinámicas, personalizadas y escalables, que se puede comunicar fácilmente con otros programas de gestión para compartir información entre sistemas. Cuenta con una amplia gama de funcionalidades predefinidas que hacen posible desarrollar soluciones complejas en plazos de tiempo reducidos. También cuenta con extensiones personalizables para prácticamente cualquier necesidad en la gestión del contenido.
- La combinación de las dos tecnologías usadas para el desarrollo, tanto MySQL como base de datos y PHP como lenguaje de programación son tecnologías de uso libre, por tanto no se necesita de algún pago previo de licencia, en el caso de que sean utilizados por personales naturales, pues para una empresa privada o pública, al ser necesario soporte y mantenimiento se puede adquirir licencias de versiones que al tener un costo, involucra un servicio personalizado.
- La organización del proyecto, los pasos a seguir definidos y la documentación del proceso, son ventajas que se tienen al utilizar un estándar de desarrollo de software.



- Para el uso de un CMS no es necesario de algún sistema operativo específico ya que al estar presente en la red, este se hace accesible desde cualquier plataforma (Windows, Mac, Linux, etc.).
- Las restricciones en cuanto al código no deberían existir, pero si deberían estar sujetas a leyes, el orden y el reconocimiento sobre el autor, es algo que si debe reconocer no terminando esto en un monopolio, el cual termina con la competencia y limita la creatividad, dificultando la integración con módulos tanto de hardware como de software.
- Existen muchas maneras de ganarse la vida sin tener que perder de vista las libertades acerca del software libre, como lo son integración hardware software, la enseñanza, servicios de soporte y mantenimiento, la distribución de programas, o los donativos inclusive.
- El software libre es una puerta abierta a la alternativa que ofrece la libertad de compartir conocimiento, ya que al ser testigos de lo que las leyes de los gobiernos alrededor del mundo pretenden, que es aumentar las restricciones sobre los recursos de los ordenadores, llegando al punto de restringir los documentos, el software y por ultimo invadiendo la privacidad de las personas, para lo cual acceden a información privada, dejando de lado la libertad, llegando ya a ser esto no solo una lucha tecnológica de ideas, sino



una lucha política y legal entre una ideología claramente establecida como lo es el software libre y un gran negocio altamente lucrativo como lo es el software propietario.

- Algunas personas y entidades, con ciertos intereses, confunden intencionadamente a los potenciales clientes y usuarios nombrándolo "software gratuito", "código abierto", o "no comercial". Esto es debido a que aprovechan, en busca de sus propios intereses, una interpretación parcial de las libertades mencionadas o la mala traducción de la expresión original "*free software*" ya que en inglés "*free*" corresponde tanto a "libre" como a "gratuito".

6.2. Recomendaciones

- Se recomienda hacer un estudio previo de la metodología de desarrollo de software para la aplicación de cualquier proyecto. Cada metodología tiene sus ventajas para determinados proyectos, esto influye en los resultados finales y en la forma de trabajo de las dos partes involucradas (desarrollador – cliente).
- Se recomienda considerar la seguridad con punto fundamental para la integridad del sistema y su información.



- Definir un cronograma de trabajo, planificando objetivos definidos, tiempos y demás recursos. Utilizar la experiencia de proyectos anteriores para definir estas actividades es una de las maneras de reducir el riesgo de errores en la planificación, más aun cuando se trabaja en conjunto con otras partes dentro de un proyecto.
- El alcance del sistema es un punto dentro de la definición del proyecto ya que es aquí donde se determina donde el proyecto termina. Se recomienda que tanto los desarrolladores y el cliente estén de acuerdo con el alcance, el cual debe ser claro y sin ambigüedades para evitar conflictos entre las partes involucradas.
- Se recomienda que para aprovechar mejor las funcionalidades del sistema y su correcto uso, se capacite a los usuarios del sistema al final del proceso de desarrollo. Esto ayudará a que los usuarios se familiaricen con el nuevo sistema y que vean las ventajas que existen frente al sistema anterior.
- Las empresas tanto privadas como públicas deben adoptar y capacitar a sus empleados en el uso de herramientas de software libre, pues solo así se lograra que los usuarios se familiaricen con su uso demostrando la calidad y seguridad de las herramientas de software libre ya sea a nivel de escritorio como a nivel de un sistema operativo multiusuario.



- Aunque estemos involucrados en el mundo de la tecnología, sea como desarrolladores o como administradores, es necesario también conocer la parte legal que involucra en este caso al software y a los sistemas desarrollados, conociendo los términos legales ya mencionados como son las patentes, el copyright, y la propiedad intelectual, lo cual nos permitiría no caer en ignorancia y cometer errores que vayan en contra de nuestros ideales.
- En sí la convivencia sería el panorama ideal entre los desarrolladores, sea de software libre o software propietario, lo cual se daría en la práctica pero en la teoría, esta convivencia se ve mermada por la parte legal que establece claramente las diferencias y las restricciones a las cuales se deben atener miembros de ambas comunidades.
- Las herramientas y aplicaciones más populares no son generalmente las mejores, ni las más estables, como por ejemplo el caso de Joomla, una de los CMS más utilizados para sitios web, sobre el cual se han reportado errores en su código fuente e incompatibilidad entre los módulos instalados, por lo cual no ofrece garantías para el desarrollo de sitios web más complejos. Por lo cual como ingenieros con acceso a un mundo informático a través del internet, es necesario investigar y buscar alternativas que alimenten el aprendizaje para encontrar diferentes soluciones informáticas.



- Recomendamos, que como parte de la malla curricular se tome en cuenta al software libre dentro de alguna materia para estudio de la carrera, de igual forma que se de a conocer sobre la gestión de contenidos y gestión de documentos, pues actualmente se han desarrollado varias herramientas que realizan estas funciones bajo la modalidad de software libre.