

# UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

Facultad de Contaduría y Administración  
Facultad de Ciencias Químicas e Ingeniería  
Maestría en Tecnologías de la Información y la Comunicación



**“Sistema estratégico de gestión documental digital colaborativo”.**

TESIS QUE PARA OBTENER EL GRADO DE  
MAESTRO EN TECNOLOGÍAS DE LA INFORMACIÓN Y LA COMUNICACIÓN

*Presenta:*

LUIS FERNANDO DÍAZ MÉNDEZ.

*BAJO LA DIRECCIÓN DE:*

DR. EDUARDO AHUMADA TELLO

*CO DIRECCIÓN:*

MTRO. CARLOS ERNESTO GONZÁLEZ SALAS

TIJUANA, BAJA CALIFORNIA, MÉXICO.

JUNIO 2015

***A mi esposa María Isabel por su valioso apoyo.***

***A mi padre por preguntarme todos los días sobre mi avance y***

***A mi hermana Gaby que la quiero mucho.***



## **Agradecimientos.**

Deseo dedicar este espacio hacia aquellas personas que fueron parte importante para que este proyecto fuera llevado a cabo, como todo tiene una raíz, quisiera comenzar con casa, es con el aliento de mi familia, en especial de mi padre y hermana, así como mi esposa Maria Isabel, su impulso me han llevado a seguir descubriendo caminos que me ayudan a superarme profesionalmente.

Además, no puedo dejar de mencionar a mi segunda casa, la organización objeto de la presente investigación, en especial atención al Dr. Adolfo Solís, por llevar las riendas de la empresa en la que me he desenvuelto los últimos años y que con su dirección ha sido posible que florezcan los proyectos de TI.

Adicionalmente no me cabe más que agradecer a los profesores y tutores de la Universidad Autónoma de Baja California, como el Dr. Eduardo Ahumada Tello por el tiempo otorgado a pulir el presente documento, así como a la Dra. Margarita Ramírez Ramírez y la Mtra Nora Del Carmen Osuna Millán por darme la oportunidad de pertenecer a la MTIC, así como a catedráticos que me llevan de la mano en la obtención de nuevos conocimientos.

Luis Fernando Díaz Méndez

## 1. Resumen.

El presente trabajo es el resultado del desarrollo e implementación de un sistema repositório de información, el cual permite agregar documentos digitalizados a través de una interface Web. Lo anterior con el objetivo de mitigar los tiempos de acceso ocasionados por los volúmenes altos de información en la organización. A través de una exposición del panorama actual de la gestión documental, se ubica en contexto el sistema propuesto y la metodología para diseño e implementación de sistemas de gestión de documentos y registros (DIRKS del inglés *Designing and Implementing Recordkeeping Systems*), la cual es utilizada para la obtención de las funcionalidades y como columna vertebral para la planeación del proyecto. Con la implementación del repositório se logró estructurar la información de tal forma que existieron mejoras en los tiempos de consulta, sin hacer grandes cambios en los hábitos de los usuarios. Por último se obtienen las conclusiones derivadas de la ejecución del proyecto, así como algunas recomendaciones.

En la Ilustración 1 se observa de forma grafica la estructura del presente documento donde cada recuadro representa un capítulo.

Ilustración 1. Esquema del documento.



Fuente: Elaboración propia.

Ilustración que muestra gráficamente la estructura del documento en capítulos, con el cual se puede observar un mapa del presente documento.

## **1. Abstract.**

The present work is the result of the development and the implementation of a document repository that allows adding digital documents through a web interface. This, in order to mitigate access times caused by high volumes of information in the organization. Through an exposure of the current background of document management, the proposed system is placed in context, also the Designing and Implementing Recordkeeping Systems (*DIRKS*) methodology used to obtain the requirements, as well as backbone for project planning. With the implementation of the repository, the information is in a structured way, so there were improvements in retrieve times without making major changes in user habits. Finally the conclusions derived from the execution of the project are obtained, as well as some recommendations.

## Índice general.

Capítulo I. ....	15
1.1. Introducción.....	15
1.2. Contexto del problema. ....	16
1.3. Planteamiento del problema. ....	22
1.4. Justificación.....	24
1.5. Objetivos. ....	27
1.6. Alcance del trabajo. ....	28
Capítulo II Marco teórico. ....	29
2.1 Definición de información.....	29
2.2 La información digital en las organizaciones. ....	30
2.2.1 Ciclo de vida del documento.....	31
2.3 Definición de repositorio digital de información. ....	32
2.4 Tipos de repositorios.....	33
2.5 Alternativas de software para la implementación de un repositorio de información. ....	34
2.5.1 Software comercial. ....	35
2.5.1.1 Gratuitos. ....	35
2.5.1.2 Bajo licenciamiento.....	36
2.5.2 Software de código abierto.....	37
2.5.3 Software propio. ....	39
2.6 Herramientas para el desarrollo del repositorio de información digital. ....	39
2.6.1 Servidor web. ....	40
2.6.2 Procesamiento del lado del servidor, <i>backend</i> .....	41
2.6.3 Procesamiento del lado del cliente, <i>frontend</i> .....	44
2.6 Herramientas para el entorno de desarrollo. ....	47
Capítulo III Metodología. ....	50
3.1 Investigación aplicada al desarrollo de software.....	50
3.2 Metodología DIRKS. ....	50
3.2.1 Investigación preliminar.....	52
3.2.2 Análisis de las actividades de la organización.....	53
3.2.3 Identificación de los requisitos. ....	53
3.2.4 Evaluación de los sistemas existentes. ....	54

3.2.5	Identificación de estrategias para cumplir los requisitos. ....	54
3.2.6	Diseño de un sistema de gestión de documentos de archivo. ....	55
3.2.6.1	El diseño de la base de datos. ....	56
3.2.6.2	El diagrama de casos de uso general, así como la descripción de cada uno de ellos. ....	56
3.2.6.3	Prototipo de cada funcionalidad. ....	56
3.2.6.4	Diagrama general de arquitectura del software. ....	57
3.2.6.5	Diagramas de secuencia. ....	57
3.2.6.6	Diseño de los casos de prueba. ....	57
3.2.7	Implementación de un sistema de gestión de documentos de archivo. ....	59
3.2.8	Revisión posterior a la implementación. ....	59
Capítulo IV	Desarrollo. ....	60
4.1	Investigación preliminar. ....	60
4.2	Análisis de las actividades de la organización. ....	61
4.3	Identificación de los requisitos. ....	64
4.4	Evaluación de los sistemas existentes. ....	65
4.5	Identificación de estrategias para cumplir los requisitos. ....	67
4.6	Diseño de un sistema de gestión de documentos de archivo. ....	70
4.6.1	Diagrama de caso de uso general y descripción de casos de uso. ....	70
4.6.2	Descripción detallada de caso de uso y prototipo. ....	72
4.6.3	Diseño de la base de datos propuesto. ....	82
4.6.4	Prototipo de las funcionalidades. ....	83
4.6.5	Diagrama general de arquitectura del software. ....	88
4.6.6	Diagramas de secuencia. ....	90
4.6.7	Diseño de los casos de prueba. ....	93
4.6.8	Ejecución de la fase de desarrollo. ....	94
4.6.9	Resultado de las pruebas a las funcionalidades. ....	110
4.7	Implementación del sistema de gestión de documentos de archivo. ....	112
4.7.1	Configuración adicional del servidor. ....	113
Capítulo V	Resultados. ....	116
5.1	Puntos a evaluar. ....	116
5.2	Resultados de las evaluaciones. ....	116
Capítulo VI	Conclusiones y recomendaciones. ....	121

Bibliografía.....	123
ANEXO 1. Cronograma de actividades.....	129
ANEXO 2. Diseños de caso de prueba.....	130
ANEXO 3 Requerimientos de infraestructura de TI para la ejecución del software.....	138

## Índice de Ilustraciones.

Ilustración 1. Esquema del documento. ....	5
Ilustración 2 Uso de repositorios en el mundo según opendoar.org al 12 de noviembre de 2013. ....	18
Ilustración 3. Relación entre costo y beneficio de un ECM. ....	36
Ilustración 4 Uso de lenguajes de programación del lado del servidor al 6 de diciembre de 2013. ....	43
Ilustración 5. Tecnologías agrupadas bajo el concepto de Ajax. ....	45
Ilustración 6 Uso de librerías Javascript en sitios web al 6 de diciembre de 2013. ....	46
Ilustración 7 Administración de sitios Adobe Dreamweaver CS5. ....	48
Ilustración 8 Herramientas adicionales de SQLYog. ....	49
Ilustración 9 Metodología DIRKS (Designing and Implementing Recordkeeping Systems).....	51
Ilustración 10 Modelo de trabajo incremental. ....	55
Ilustración 11. Organigrama de la organización simplificado. ....	60
Ilustración 12. Plan de la calidad resumido. ....	62
Ilustración 13 Proceso para proporcionar información digital de Soporte al Área de la Calidad.....	62
Ilustración 14. Proceso para visualizar la documentación del expediente. ....	63
Ilustración 15. Mecanismo para consulta de la información electrónica del expediente. ....	66
Ilustración 16. Etapas de desarrollo del software mediante el modelo incremental. ....	68
Ilustración 17. Proyecto agendado en el gestor DotProject de la organización. ....	69
Ilustración 18. Diagrama de caso de uso general. ....	71
Ilustración 19. Diagrama Entidad Relación de la base de datos. ....	82
Ilustración 20. Prototipo inicio de sesión. ....	83
Ilustración 21. Prototipo cierre de sesión. ....	83
Ilustración 22. Prototipo recordar contraseña. ....	84
Ilustración 23. Prototipo crear carpeta. ....	84
Ilustración 24. Prototipo crear archivo. ....	85
Ilustración 25. Prototipo obtener elementos. ....	85
Ilustración 26. Prototipo selección de elementos. ....	85
Ilustración 27. Prototipo mover elementos. ....	85
Ilustración 28. Prototipo borrar elemento. ....	86
Ilustración 29. Prototipo cambiar nombre a elemento. ....	86
Ilustración 30. Prototipo cambiar permisos. ....	87
Ilustración 31. Prototipo notificar nuevo elemento. ....	87
Ilustración 32. Prototipo enviar elemento a principal. ....	88
Ilustración 33. Prototipo elementos desde expedientes. ....	88
Ilustración 34. Arquitectura MVC utilizada para el desarrollo del sistema. ....	89
Ilustración 35. Componentes del sistema en la arquitectura MVC. ....	89
Ilustración 36. Diagrama de secuencia crear carpeta. ....	90
Ilustración 37. Diagrama de secuencia recordar contraseña. ....	91
Ilustración 38. Diagrama de secuencia obtener elementos. ....	91

Ilustración 39. Diagrama de secuencia mover elemento. ....	92
Ilustración 40. Diagrama de secuencia notificar nuevo elemento. ....	93
Ilustración 41. Estructura de los archivos fuentes del proyecto.....	95
Ilustración 42. Resultado del desarrollo de la funcionalidad inicio de sesión.....	95
Ilustración 43. Resultado del desarrollo de la funcionalidad recordar contraseña.....	95
Ilustración 44. Función Javascript para validar forma HTML.....	96
Ilustración 45. Resultado del desarrollo de la funcionalidad cerrar sesión.....	97
Ilustración 46. Método para finalizar la sesión de la clase LoginController. ....	97
Ilustración 47. Despliegue de mensaje de una sesión terminada. ....	98
Ilustración 48. Resultado del desarrollo de la funcionalidad crear carpeta. ....	98
Ilustración 49. Resultado del desarrollo de la funcionalidad crear archivo.....	98
Ilustración 50. Función del controlador LoginController para guardar elemento carpeta. ....	99
Ilustración 51. Procesamiento que reciben los archivos en el controlador.....	100
Ilustración 52. Resultado del desarrollo de la funcionalidad obtener elementos.....	101
Ilustración 53. Resultado del desarrollo de la funcionalidad selección de elementos.....	101
Ilustración 54. Función Javascript que obtiene elementos mediante AJAX. ....	101
Ilustración 55. Ruta de la ubicación actual dentro del repositorio.....	102
Ilustración 56. Resultado del desarrollo de la funcionalidad mover elementos ....	102
Ilustración 57. Resultado del desarrollo de la funcionalidad renombrar elemento.....	102
Ilustración 58. Resultado del desarrollo de la funcionalidad borrar elementos. ....	103
Ilustración 59. Inicio de la transacción en la base de datos repositorio.....	104
Ilustración 60. Resultado del desarrollo de la funcionalidad cambiar permisos.....	105
Ilustración 61. Aviso de permiso negado al visualizar un elemento.....	106
Ilustración 62. Verificación de permisos efectivos. ....	106
Ilustración 63. Correo electrónico recibido por la funcionalidad enviar notificación. ....	107
Ilustración 64. Resultado del desarrollo de la funcionalidad enviar notificación.....	108
Ilustración 65. Enlaces a elementos desde la página principal.....	108
Ilustración 66. Resultado del desarrollo de la funcionalidad para agregar archivos desde detalles de expedientes.....	109
Ilustración 67. Herramienta de búsqueda de elementos. ....	110
Ilustración 68. Acceso al servidor web donde se alojará la aplicación. ....	113
Ilustración 69. Instalación de URL Rewrite 2.0 mediante la consola de IIS. ....	114
Ilustración 70. Contenido del archivo web.config en el directorio raíz usado por URL Rewrite de IIS. ...	114
Ilustración 71. Configuración autenticación anónima deshabilitada en el IIS.....	115
Ilustración 72. Acceso al sistema repositorio a través de la aplicación de escritorio.....	117
Ilustración 73 Comparación de tiempos de espera antes del repositorio.....	119
Ilustración 74. Accesos a los documentos del ISO 9001. ....	120

## Índice de tablas.

Tabla 1. Lista de acrónimos usados comúnmente.....	34
Tabla 2. Uso de tecnologías en las eras web. ....	42
Tabla 3. Resultados esperados en cada etapa de la metodología DIRKS. ....	52
Tabla 4. Formato para casos de prueba.....	58
Tabla 5. Los 10 archivos con mayor número de kilobytes en el sistema de expedientes. ....	64
Tabla 6. Listado de casos de uso. ....	72
Tabla 7. Descripción de inicio de sesión. ....	73
Tabla 8. Descripción cierre de sesión.....	73
Tabla 9. Descripción recordar contraseña. ....	74
Tabla 10. Descripción crear carpeta. ....	75
Tabla 11. Descripción crear archivo.....	75
Tabla 12. Descripción obtener elementos.....	76
Tabla 13. Descripción selección de elementos. ....	77
Tabla 14. Descripción mover elementos.....	77
Tabla 15. Descripción borrar elementos.....	78
Tabla 16. Descripción cambiar nombre. ....	78
Tabla 17. Descripción cambiar permisos. ....	79
Tabla 18. Descripción notificar nuevo elemento. ....	80
Tabla 19. Descripción enviar elemento a página principal. ....	80
Tabla 20. Descripción cargar elementos desde expedientes.....	81
Tabla 21. Extensiones permitidas en la funcionalidad crear archivo.....	99
Tabla 22. Tabla de permisos efectivos para los elementos. ....	104
Tabla 23. Formato del reporte de resultados de los casos de prueba. ....	110
Tabla 24. Resultados de las pruebas de las fases 1 y 2. ....	111

## Índice de siglas y abreviaturas.

(X)HTML	eXtensible HyperText Markup Language (Lenguaje de Marcas de Hipertexto Extensible).	DOM	Document Object Model (Modelo de Objetos del Documento).
AJAX	<i>Asynchronous JavaScript and XML (Javascript y XML Asíncrono).</i>	ECM	Enterprise Content Management (Gestión de Contenidos Empresariales).
API	Application Programming Interface (Interfaz de Programación para Aplicaciones).	EDMS	Electronic Document Management Systems (Sistemas de Gestión Documental).
ASP.NET	Active Server Pages .NET (Páginas Activas del Servidor en .NET)	ERMS	Electronic Records Management System (Sistema de Gestión de Registros Electrónicos).
Blogs	Expresión proveniente de <i>Web Log</i> , (Bitácora <i>Web</i> ).	FTP	<i>File Transfer Protocol (Protocolo de Transferencia de Archivos).</i>
BPMN	Business Process Model and Notation (Modelos y Notación de Procesos de Negocio).	GUI	Graphical User Interface (Interfaz Gráfica de Usuario).
BSD	Berkeley Software Distribution (Distribución de Software Berkeley).	HTML	HyperText Markup Language (Lenguaje de Marcas de HiperTexto).
CASE	<i>Computer Aided Software Engineering (Herramientas de Ingeniería Asistida por Computadora).</i>	HTTP	<i>Hypertext Transfer Protocol (Protocolo de Transferencia de Hipertexto).</i>
CBUC	<i>Consorci de Biblioteques Universitàries de Catalunya (Consortio de Bibliotecas Universitarias de Cataluña).</i>	IIS	Internet Information Services (Servicios de Información para Internet).
CGI	Common Gateway Interface (Interfaz de Entrada Común).	ISO	International Organization for Standardization (Organización Internacional de Normalización).
CSS	Cascading Style Sheets (Hojas de Estilo en Cascada).	LAECSP	Ley de Acceso Electrónico de los Ciudadanos a los Servicios Públicos.
DBMS	Data Base Management System (Sistema de Gestión de Base de Datos).	MDC	Memòria Digital de Catalunya (Memoria Digital de Cataluña).
DER	Diagrama Entidad Relación.	MDX	Materials Docents en Xarxa (Materiales Docentes en Red).
DHTML	<i>Dynamic HyperText Markup Language (Lenguaje de Marcas de Hipertexto Dinámico).</i>	MIT	Massachusetts Institute of Technology (Instituto Tecnológico de Massachusetts).
DIRKS	Designing and Implementing Recordkeeping Systems (Diseño e Implementación de Sistemas de Gestión de Registros).	MoReq	<i>Model Requirements for the management of electronic records (Modelo de Requisitos Para la Gestión de Documentos Electrónicos y Registros).</i>
DLM Forum	<i>Document Lifecycle Management Forum (Foro de Gestión del Ciclo de Vida del Documento).</i>	MVC	Model View Controller (Modelo Vista Controlador).
		NARAS's	<i>National Archives and Records Administration (Archivos Nacionales y Administración de Registros).</i>

*NFS*

*Network File System (Sistema de Archivos de Red).*

*openDOAR*

*Directory of Open Access Repositories (Directorio de Repositorios de Acceso Abierto).*

*PDF*

*Portable Document Format (Formato de Documento Portátil).*

*PHP*

*PHP Hypertext Pre-processor (Pre-procesador de Hipertexto PHP).*

*RACO*

*Revistes Catalanes amb Accés Obert (Revistas Catalanas de Acceso Abierto).*

*RECERCAT*

*Dipòsit de la Recerca de Catalunya (Depósito de la Investigación de Cataluña).*

*ROI*

*Return Of Investment (Retorno de Inversión).*

*SGDEA*

*Sistemas de Gestión de Documentos Electrónicos de Archivo.*

*SIAB*

*Sistema Integral para la Automatización de Bibliotecas.*

*SMB*

*Server Message Block (Bloque de Mensajes de Servidor).*

*SQL*

*Structured Query Language (Lenguaje de Consulta Estructurado).*

*SWF*

*Abreviación de Shockwave Flash.*

*TCP*

*Transmission Control Protocol (Protocolo de Control de Transmisión).*

*TDR*

*Tesis Doctorals en Xarxa (Tesis Doctorales en Red).*

*TI*

*Tecnologías de la Información.*

*UML*

*Unified Modeling Language (Lenguaje Unificado de Modelado).*

*URL*

*Uniform Resource Locator (Localizador Uniforme de Recursos).*

*USB*

*Universal Serial Bus (Bus Universal en Serie).*

*WebDAV*

*Web Distributed Authoring and Versioning (Edición y Versionado Distribuidos Sobre la Web).*

*XML*

*eXtensible Markup Language (Lenguaje de Marcas Extensible).*

## Capítulo I.

### 1.1. Introducción.

La organización Grupo Farías a la que se hace referencia en el presente trabajo es una firma de servicios legales ubicada en la ciudad de Tijuana, Baja California, México, la cual se especializa en el ramo fiscal. En el año 2005 la organización comenzó a incentivar el desarrollo de las tecnologías de la información (TI) dentro de la empresa para así hacer frente a la automatización de los procesos enfocados al servicio.

Las organizaciones buscan la manera de reducir el espacio físico destinado al almacenamiento de documentos, por tal motivo, la migración a sistemas de información (SI) se vuelve una opción viable (Cervantes de la Fuente, 2014). Durante los últimos años se ha identificado un importante incremento en la información digitalizada dentro de la organización, detectándose oportunidades de mejora dentro de este rubro encaminadas a solucionar los altos tiempos de acceso a la documentación digital.

La automatización de los procesos dentro de las organizaciones se ha convertido en un desafío en las últimas décadas. Esto ha ocasionado la aparición de procesos a realizar de manera colaborativa, ya que permiten abordar más eficazmente los problemas si se realizan en conjunto (Gutiérrez, et al., 2006), por lo que es una de las características deseables de los sistemas actuales.

El software que brinde soporte a estas problemáticas, por lo tanto, debe ser capaz de inmiscuirse bien en los procesos clave de gestión de información para permitir un flujo de trabajo como menciona Osuna Alarcón y De La Cruz Gómez (2010). Con la implementación de SI, las organizaciones pueden lograr importantes mejoras pues automatizan procesos, proporcionan información de apoyo y facilitan lograr ventajas en el terreno de los negocios (Cohen Karen & Asín Lares, 2000).

## 1.2. Contexto del problema.

La competitividad en las empresas de servicios hace necesario que éstas busquen la forma de atraer clientes potenciales, una de las formas de lograr esto es aumentar la calidad de los servicios de tal forma que busquen un diferencial o valor agregado respecto a sus competidores (Cohen Karen & Asín Lares, 2000). No en vano el interés por las normas internacionales ha venido en incremento, tal como se observa en el sitio oficial de la Organización Internacional de Normalización (ISO del inglés *International Organization for Standardization*) en su encuesta de certificaciones (2013). La organización adoptó el estándar ISO 9001 y actualmente se encuentra certificada bajo esta norma (Sitio Web Grupo Farías, 2014).

El objetivo de la implementación de esta norma en la organización es pulir los procesos para la oportuna atención al cliente, centrándose en la mejora continua de sus procesos. Las TI tienen cabida en este sentido, pues la empresa requiere de herramientas que las ayuden a automatizar estos procesos para conseguir sus objetivos.

Según los registros, los servicios que ofrece la organización pueden generar información cuantiosa por efecto de las diferentes instancias o etapas en las que se incurren. Si bien contar con documentación impresa es requerido dentro de los procesos de la organización, el uso de diferentes TI han venido a requerir que un porcentaje de dicha documentación se encuentre digitalizada.

El propósito principal de que dicha información se encuentre digitalizada es atender de forma oportuna al cliente tal como es el propósito de la norma 9001, ésta a su vez en su apartado 4.2, Requisitos de la documentación (ISO, 2008), menciona el control que debe existir sobre los documentos dentro de la organización. Además según estudios realizados existe hasta un 40% la posibilidad de dar respuestas

apropiadas a los clientes, por lo que una correcta gestión documental contribuye a mejorar la eficiencia de la organización (D'Alòs-Moner, 2006).

Así pues, como menciona Novelle López (2012), la gestión documental ha pasado de ser algo invisible para las organizaciones, a presentar una importante estrategia en la toma de decisiones y de los procesos, pues como comenta el Dr. Dearstyne en Read y Lea Ginn (2011), las organizaciones tienden a ser más dependientes de la documentación electrónica. El uso de TI para el manejo de todo tipo de archivos digitales ha propiciado que la gestión de los documentos pase de los estantes hacia el almacenamiento digital.

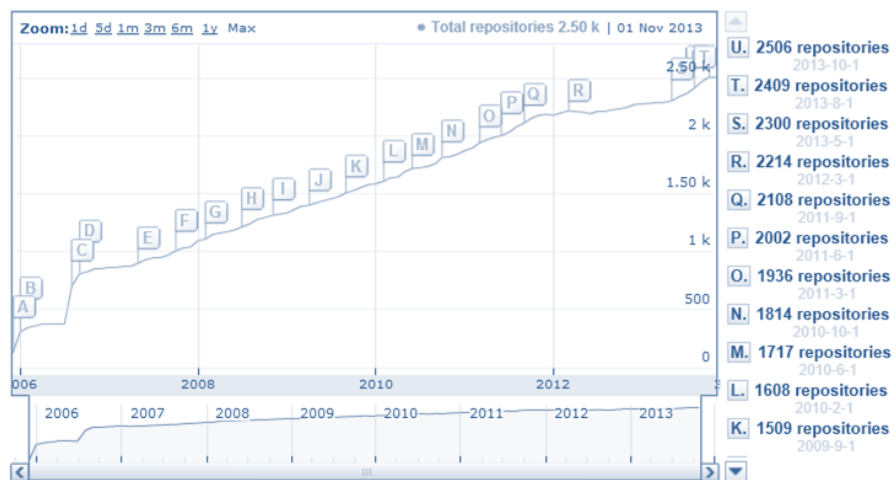
El tratamiento de la documentación digital no es algo nuevo, pero si ha salido a flote en los últimos tiempos el concepto de gestión de la documentación (*Document Management*) como “tema estrella” (D'Alòs-Moner, 2011), esto como principal desencadenador la cantidad de información digital producto de la revolución industrial de datos. Debido a lo anterior a continuación se exponen ejemplos de las soluciones que han dado distintas organizaciones a este tema.

Las universidades son organizaciones que necesitan el resguardo de información como por ejemplo, tesis, artículos, material didáctico, llamados comúnmente repositorios de conocimiento (Adame, Llórens, & Schorr, 2013), por ello a la hora de realizar una implementación se realizan análisis sobre las prestaciones de un software para este fin, llevando a unos casos evaluar software de código abierto o comercial. Tal es el caso de la evaluación que fue llevada a finales del 2004 por parte de bibliotecarios e informáticos del Consorcio de Bibliotecas Universitarias de Cataluña (*CBUC del catalán Consorci de Biblioteques Universitàries de Catalunya*) y algunas universidades catalanas para la elección en conjunto de un repositorio institucional, evaluándose las opciones *ARNO*, *Cdsware*, ahora llamado CDS Invenio, *DSpace*, *EPrints*, *Fedora*, *i-TOR* y *MyCoRe* (De la Vega, 2008), todos ellos propuestos en un informe del

Open Source Institute, al final decidieron la utilización de DSpace, en primera instancia por su adopción mundial del 40% de los repositorios en el planeta así como por su facilidad de uso y administración.

Tal es el auge de este tipo de repositorios en el mundo que las base de datos han ido en incremento en los últimos años, Véase Ilustración 2.

Ilustración 2 Uso de repositorios en el mundo según opendoar.org al 12 de noviembre de 2013.



Fuente: OpenDOAR (2013).

Ilustración que muestra la tendencia de la base de datos del sitio opendoar.org, en la cual se refleja el aumento de repositorio en el mundo a través del tiempo desde el año 2006.

Sin este tipo de repositorios, las organizaciones se tornarían menos participativas, como el caso que presenta Concepción Báez (2008), debido a que los integrantes podrían decantarse por medios alternativos para resguardar dicha información como pueden ser *Wikis*, *Blogs*, o repositorios basados en la nube, afectando procesos establecidos.

Si se toma la definición de (Concepción Báez, 2008), sobre documentos finales o terminados, conviene traer al contexto entonces que no es un documento final, en esta misma referencia se cita que son aquellos documentos que aún no están listos, es decir que pudiesen haber modificaciones a ellos producto de la colaboración. Es importante entonces definir qué tipo de documentos serán resguardados en los repositorios, ya que podrían ser diferentes los documentos guardados por una

editorial de revistas que por una organización médica. Otro ejemplo de elección de repositorio se observa en la editorial de revistas catalanas de acceso abierto (*RACO del catalán Revistes Catalanes amb Accés Obert*), donde se evaluaron *DSpace+CWIS*, *uPortal*, *ContentDM* y *Lodel* y finalmente se seleccionó *Open Journal Systems (OJS)*, por ser una opción viable para revistas y publicaciones que requieren revisión, De la Vega (2008).

Las organizaciones gubernamentales son otro tipo de entidades que generan gran cantidad de información, cada año crece un 60% la cantidad de información digital según Giralt, Vidal-Pijoan, & Pérez-Soler (2011), por lo que se hace más evidente garantizar su autenticidad, integridad, así como facilitar su acceso y disponibilidad, además de la necesidad de perseverar la documentación en las organizaciones es inherente hacerlo debido a las leyes o normas que puedan estar regidos los países, donde comúnmente abarcan varios años o incluso décadas.

Debido a esto los gobiernos invierten esfuerzos en implementar sistemas que les permitan llevar a cabo esta labor, tal es el ejemplo del archivo del ayuntamiento de Barcelona, que basados en una buena referencia del gobierno de Amsterdam deciden implementar el *Hitachi Content Platform* de *Hitachi Data Systems*, el cual brinda una alta integridad a los documentos ahí resguardados, además de ser altamente escalable, ya que soporta protocolos abiertos tales como: *NFS*, *SMB*, *HTTP*, *WebDAV*; además del establecimiento de políticas de acceso (Giralt, Vidal-Pijoan, & Pérez-Soler, 2011).

Por lo tanto, la selección del software para el resguardo de la documentación digital debe ser con base en las necesidades específicas de la organización, pues el software en sí, no realizará el trabajo, si no que se tienen que crear políticas y procedimientos de digitalización para su ingreso al sistema, como es el caso del ayuntamiento de Barcelona (Giralt, Vidal-Pijoan, & Pérez-Soler, 2011), además se deben establecer los tiempos en los que se realizarán, ya que deben ser acordes a los flujos de trabajo de las organizaciones o en pocas palabras “entendibles por la organización” (D’Alòs-Moner, 2011).

Es evidente que todas estas implementaciones han sido producto del aumento de información digital y de la necesidad de resguardarla y clasificarla, no en vano la base de datos de *openDOAR* ha ido en constante crecimiento desde 2005, en donde se aprecia hasta un 250% de aumento a nivel mundial en lo que a repositorios de conocimiento se refiere (OpenDOAR, 2013). Si a esto sumamos que las organizaciones poco a poco se están dando cuenta de que la gestión de documentación no es una simple cuestión de guardar archivos informáticos, sino que es un producto de sus procesos que podría llevarlos a aspirar a una mejora continua, estaríamos posiblemente en un tiempo donde se atiende más este tema.

Tal evidente es la importancia del resguardo de información que ha dado nacimiento a normas y estándares ISO para la gestión de documentos en las organizaciones tal como el compendio de buenas prácticas en gestión de documentos, la norma ISO 15489, la cual tiene como objeto regular “la gestión de documentos que producen las organizaciones, ya sean públicas o privadas, con fines externos o internos”, de forma que se aplica a la gestión de los documentos, con independencia de su formato o soporte y de las tecnologías utilizadas, creados o recibidos por una organización en el transcurso de sus actividades (Alonso, García, & Lloveras, 2007).

Así mismo la norma 30300 y 30301 pautan la implementación de un Sistema de Gestión de la Documentación. Incluso, la Comisión Europea y el *DLM Forum* han ido más allá, planteando el modelo *Model Requirements for the management of electronic records (MoReq)*, el cual trata las capacidades necesarias para la gestión de documentos electrónicos de archivo mediante programas informáticos y tradicionales, actualmente MoReq en su versión 2010 es certificable, de hecho, a octubre 2013 se están evaluando sistemas de *gestión de documentos electrónicos de archivo (SGDEA)* o ERMS del inglés *Electronic Records Management System* que cumpla con este modelo (Sitio Web MoReq2, 2013).

MoReq2010 gestiona registros, que según la definición ISO 15489 de Palma (2008) lo definen como:

*Información creada o recibida, conservada como información y prueba, por una organización o un individuo en el desarrollo de sus actividades, o en virtud de sus obligaciones legales.*

Existen otras características que son generalmente esperadas por este tipo de sistemas, según un estudio realizado a bibliotecarios acerca de lo que esperaban de un software ERMS era: Gestión del flujo de trabajo, Control de préstamos, Manejo de estadísticas, Guardado de información administrativa, Información financiera, Interoperabilidad (Collins & Grogg, 2011). Como se puede apreciar cada de una de estas tareas está estrechamente unida a las operaciones administrativas de la organización, es decir, este software es más que un repositorio, es toda una solución que ayuda a la organización a desempeñar su objetivo por lo que su selección debería hacerse mediante un estudio y análisis como menciona Osuna Alarcón & De La Cruz Gómez (2010).

### 1.3. Planteamiento del problema.

En la organización el intercambio de información digitalizada es fundamental como proceso operativo y es resguardada celosamente por ser un recurso fundamental para la prestación del servicio, la información digital generada dentro de la organización y la resultante en entidades externas se tiene que gestionar durante la vida del servicio contratado.

Además, se encuentra la proporcionada por los clientes, la cual es requisito para el cumplimiento del servicio, pues bien, sumados las tres fuentes: información generada de los procesos internos, información proporcionada por los clientes e información generada por las dependencias en donde se tramitan los asuntos, resulta vasta información que la organización tiene que gestionar, por lo cual busca un mecanismo que le permita hacerlo con eficiencia.

Dicha información es resguardada de forma digital en el sistema de expedientes donde permite consultarse, sin embargo, el volumen de información ocasiona que el tiempo de recuperación sea alto para la realización de esta tarea, pues existe la carencia de anexar más de un documento digital por expediente y el personal tiene que esperar a que el documento sea transferido al medio de visualización. Además, la colaboración se ve afectada, pues no existen mecanismos para que el personal sea avisado cuando existe nueva documentación digitalizada en el sistema.

Adicionalmente, la colaboración entre el personal de la empresa requiere el intercambio de archivos constante, de forma natural suele elegirse el correo electrónico, aun cuando el archivo sea destinado al compañero del siguiente escritorio, esto, la mayoría de las veces es funcional, pero no cuando el correo electrónico presenta fallas o son archivos que exceden los límites del servicio.

Con base al problema planteado las preguntas que se tendrían que realizar para la realización de esta investigación son:

¿Qué funcionalidades se podrían desarrollar para socavar las debilidades del sistema de expedientes en relación a la gestión de los archivos digitales?.

¿Cómo se podría ayudar a la organización a aumentar su eficiencia en sus operaciones de intercambio de archivos digitales?.

¿Cómo se podría lograr un entorno colaborativo para el intercambio de archivos digitales entre las áreas de la organización?.

¿Cómo se podría tener la información digitalizada de tal forma que el proceso de obtención y anexo de los archivos sea en un periodo más corto de tiempo?.

¿Qué aspectos de seguridad se deben considerar para el desarrollo del sistema que gestione los documentos digitales?.

#### 1.4. Justificación.

Dentro de la cantidad de información que genera la organización, la más importante es la relacionada al expediente, pues al ser una empresa de servicios jurídicos, ésta es vital para el seguimiento a los servicios contratados. Por lo tanto, el mecanismo para almacenamiento y recuperación de la información debe ser ágil, de tal forma que los tiempos en recuperación sean cortos; debe ser colaborativo, de modo que permita el intercambio de documentos entre las áreas y además, debe proveer mecanismos de seguridad.

El desarrollo e implementación de un sistema que auxilie a llevar las tareas mencionadas en el párrafo anterior es deseable, pues los mecanismos actuales no son los más óptimos, esto puede repercutir en una atención al cliente más lenta y contraria a lo que exige la norma ISO 9001 implementada en la organización.

En una primera instancia dentro de la organización fue suficiente contar con la documentación digitalizada concentrada en un solo documento digital, pero con el aumento en el volumen de información conforme avanza el servicio ralentiza los procesos de gestión de la documentación. Un aspecto que se solventaría con esta investigación es que se daría la función de tener estructurada la información digital en documentos seccionados, puesto que permitiría tener varios archivos digitales por expediente y así se disminuirían los tiempos de espera de transmisión de los archivos a través de la red, lo que se traduciría en una mejor respuesta al cliente.

El sistema también se utilizaría para fomentar la colaboración en la organización, ya que las personas podrían compartir información digitalizada y así crear equipos de trabajo, como ejemplo, el sistema enviaría avisos a los integrantes del grupo notificando que se modificó un archivo o se agregó

información reciente. Sumado a esto, podría solucionarse el problema de los altos tiempos de espera que en ocasiones se producen por cuestiones de lentitud del correo electrónico, pues es común que el personal intercambie archivos y con este sistema se reduciría el uso de mecanismos tradicionales, tales como, el correo electrónico o medios de almacenamiento USB.

La integración con los sistemas existentes es vital, pues si se busca disminuir los pasos en las operaciones de búsqueda y compartición de documentos, sería contraproducente crear nuevos mecanismos para el acceso al sistema, por lo que una integración con los sistemas actuales es una funcionalidad buscada.

En la organización los archivos digitales son producto de los procesos de la organización y por ello deben ser resguardados celosamente, por lo que el almacenamiento con proveedores externos no es deseable por la empresa, los motivos generales son el posible robo de información o posible visualización por parte de terceros debido a problemas de seguridad en los servicios, como las expuestas por Kholia & Przemysław (2013).

Una alternativa sería la opción de licenciamiento, sin embargo habría que tomar en cuenta costos de licenciamiento y adquisición de hardware adicional, así como evaluar el impacto en sus procesos, por lo que solventar este tipo de costos se vuelve complejo para pequeñas y medianas empresas, como menciona Goodyear (2013).

Además como comenta David (2012) un sistema web propio puede tener la propiedad de ser escalable y así ajustarse a la mejora continua que exige la organización derivado del sistema de calidad implementado.

A manera de resumen a continuación se exponen los puntos de relevancia de la necesidad de desarrollar esta solución software para la organización:

- Preocupación de la organización actual por la información digitalizada.
- Seguridad de la información digitalizada.
- Altos tiempos de acceso a la información digitalizada por falta de una estructura adecuada.
- Altos tiempo por el área dedicada a digitalizar los archivos por las limitaciones del sistema actual.
- Intercambio de archivos por métodos tradicionales anticuados que se traducen en pérdida de tiempo de espera.
- Se requiere un mecanismo para que el cliente pueda recibir respuesta rápidamente por parte de su contacto directo y no por personal secundario.
- Se busca evitar una no conformidad en el servicio por falta atención a un cliente o por lentitud.
- Ahorro en costos de licenciamiento de software comercial y de hardware.

## 1.5. Objetivos.

Objetivo general.

Desarrollar e implementar una solución software que permita el almacenamiento y consulta de información digitalizada de los registros de la organización, de tal forma que la información sea resguardada conforme a las necesidades y que permita un entorno colaborativo en la organización.

Objetivos particulares.

- Resguardo de los archivos digitales de los expedientes de forma estructurada.
- Permitir el intercambio de archivos dentro de la organización a través de los sistemas existentes.
- Interfaz gráfica que permita agregar múltiples archivos a través de una navegación sencilla y ágil para el usuario.
- Reducir los tiempos de acceso a la información digital de los expedientes.
- Proporcionar acceso integral con los demás sistemas de la organización.

En esencia, desarrollar e implementar un mecanismo para que la información pueda ser recuperada de una forma ágil, es requerido para colocar a la empresa en competencia que el entorno exige y que la ayude a cumplir sus objetivos de mejora continua y que disminuya tiempos de proceso y aumente el trabajo colaborativo en la organización.

## 1.6. Alcance del trabajo.

La metodología seleccionada proporciona las pautas para el desarrollo de un sistema de gestión de documentos en una organización, durante la obtención de los requerimientos se tomarán en cuenta varios aspectos en relación a la gestión de la documentación, tales como, el tipo de información que se desea resguardar, procedimientos que la generan, quienes tienen acceso y mediante análisis de los sistemas existentes. Con base en lo anterior se obtendrán las funcionalidades, se desarrollarán y se implementarán, considerando el resguardo de documentación final relativa a la prestación del servicio, es decir, aquella que es resguardada en el expediente físico.

Sin embargo, en la última fase, la metodología menciona la evaluación posterior a la implementación del sistema, en ésta se contempla la retroalimentación de la organización a mediano plazo, misma que podría producir nuevas funcionalidades o modificaciones. Esto último escapa del alcance de la presente investigación en cuanto al desarrollo de las funcionalidades que se produzcan dentro de esta etapa, pero se mencionarán en la etapa de resultados a manera de trabajos futuros.

El sistema será concebido en su diseño para ser implementado con la metodología existente dentro de la empresa ubicada en la ciudad de Tijuana, Baja California, sin embargo, ya que ésta cuenta con certificación a nivel nacional, podría ampliarse su implementación y así abarcar otras oficinas en el país, pues el sistema ISO implementado abarca a todas.

## Capítulo II Marco teórico.

### 2.1 Definición de información.

El objeto principal al que se hace referencia en esta investigación en todo momento es la información digital en la organización, por ello conviene definir qué es lo que queremos gestionar, Cohen Karen & Asín Lares (2000), definen información como: “Producto de datos que dentro de un contexto tienen un significado para alguien”. Pressman (2010) define a la información de forma similar como hechos asociados a un contexto. Por lo que con base en estas definiciones se puede deducir que la información es el producto de datos asociados que tienen un significado, ésta puede provenir de varios recursos de la organización donde se estén ejecutando procesos que produzcan datos, por lo tanto dicha información es un activo importante de la organización, pues es como el aceite que hace trabajar a la organización y que la mantiene funcionando correctamente (Escobar, 2006).

Si bien existen diversos sujetos que pueden producir información, en esta investigación se referirá a la asociada a las organizaciones de servicios, ya que la aplicación de esta investigación va orientada a dicho sector. Además del interés intrínseco de las organizaciones por su información como activo propio, puede haber varios aspectos para requerir la conservación de su información tales como: legales, de procedimiento con los estándares implementados, evidencia o como entradas de sus procesos operativos, el objetivo es que dicha información digital no se corrompa y sea accesible a lo largo del tiempo (Boté Vericad, 2012).

Pero la definición de información simple y llana puede no darnos un contexto suficiente para dimensionar el problema, por lo tanto es necesario incorporar el término información electrónica o en su forma acumulada por medio de documentos electrónicos, para este caso conviene citar la definición

de la Ley de Acceso Electrónico de los Ciudadanos a los Servicios Públicos (Boletín Oficial del Estado, 2007) de España, la cual la define como:

*Información de cualquier naturaleza en forma electrónica, archivada en un soporte electrónico según un formato determinado y susceptible de identificación y tratamiento diferenciado.*

El hecho que el gobierno español haya promulgado la Ley de Acceso Electrónico de los Ciudadanos a los Servicios Públicos (LAECSP) o el gobierno de Estados Unidos de América haya planificado el NARAS's *Open government* 2012-2014, así como diversas instituciones educativas (García-Morales, 2013), puede darnos la idea como las organizaciones se están volcando a la nueva era de la documentación electrónica con fines administrativos y será donde las empresas tendrían que considerar para beneficio de sus procesos.

## **2.2 La información digital en las organizaciones.**

García-Morales (2013) comenta que en el siglo XXI, el verdadero cambio de paradigma radica en la evolución desde la gestión electrónica de documentos a la gestión de documentos electrónicos. En el pasado, las empresas se preocupaban por gestionar los documentos digitales que tenían en papel, es decir el medio electrónico era solo un soporte adicional; hoy en día con la automatización y usos de las TI muchos documentos originales y válidos en el ámbito legal se encuentran de forma pura en medios digitales, como el caso de las facturas electrónicas, donde el documento válido para efectos legales es un archivo XML.

En la era actual, las empresas no pueden plantearse su competición competitiva en el mercado sin hacer esfuerzos por desarrollar una rápida y eficaz capacidad de adaptación (Goitia, Sáenz-de-Lacuesta, & Bilbao, 2008). Con el buen recibimiento que han tenido los sistemas de información en las

organizaciones se ha pasado a una era en la que la comunicación es mayormente por medios tecnológicos de información, esto da como producto que el intercambio que antes se hacía por medios impresos ahora sea en mayor grado de forma digital, esto orilla a que las organizaciones tengan en proporción su información en soporte digital, en estos casos las organizaciones deben prestar atención al hecho de implementar mecanismos adecuados para el control de su archivo electrónico.

Como lo comenta García-Morales (2013), no solo es preocuparse por la gestión documental si no ir más allá incorporando una visión global que cubra las necesidades de la organización, pero que también las ayude a sobrellevar sus procesos y que sean verdaderas dinámicas de cambio y de mejora continua.

### **2.2.1 Ciclo de vida del documento.**

En el año de 1940 Philip Coolidge Brooks introdujo el concepto "*life cycle of records*" Así creó uno de los paradigmas de la archivística contemporánea, de enorme influencia hasta nuestros días, que en esencia viene a señalar que el documento tiene una vida similar a la de un organismo biológico, el cual nace (fase de creación), vive (fase de mantenimiento y uso) y muere (fase de expurgo) menciona Cruz Mundet (2011). Aunque como menciona este mismo autor, existen otras clasificaciones, por ejemplo la de Ira A. Penn el cual los clasificaba con base a un rol o función operacional en un momento determinado, estas fases fueron: activos, semiactivos e inactivos.

El ciclo de vida de los documentos se convirtió en la base conceptual de la gestión de documentos, que en la definición de Artel Ricks "se extiende al ciclo de vida completo de los documentos, desde su producción hasta su eliminación final o su envío al archivo para su conservación permanente" (Cruz Mundet, 2011).

La presente investigación vendría a agregar valor al ciclo de vida del documento dentro de la fase de mantenimiento y uso, puesto que en esta fase es donde la organización tiene un uso intensivo del

documento y es aquí donde generalmente los controles de los sistemas de la calidad como ISO 9001 intervienen con más frecuencia, puesto que son requeridos como entradas de procesos encargados de brindarle el servicio al cliente.

### 2.3 Definición de repositorio digital de información.

En algunas referencias se observa que la definición de repositorio va en caminata a los de contenido académico, tal como se menciona en la definición de repositorio del Sitio Web EcuRed (2013). Aunque se observa que Adam (2008) hace referencia a éstos como una herramienta de ayuda a las organizaciones, sin especificar.

En el portal de la Universidad de Salamanca (Sitio Web USAL, 2013) define repositorio de la siguiente forma:

*“Un repositorio, depósito o archivo es un sitio web centralizado donde se almacena y mantiene información digital, habitualmente bases de datos o archivos informáticos”*

Ferreras Fernández (2010) define repositorio institucional como el conjunto de servicios para almacenar y hacer accesibles materiales de investigación de una institución. Una gran cantidad de repositorios en el mundo generalmente son de entidades gubernamentales, investigación, universidades entre otras, como se observa en el Ranking Web de Repositorios (Ranking Web Of Repositories, 2014). Esto podría deberse a la necesidad de automatización que existe actualmente de los procesos, servicios y gestión de las bibliotecas, requiriendo para ello sistemas integrales (Arriola Navarrete & Butrón Yáñez, 2008). Este tipo de sistemas los define como un sistema integral para la automatización de bibliotecas (SIAB) o Integrated Library Systems (ILS por sus siglas en inglés) (Arriola Navarrete & Butrón Yáñez, 2008).

Los repositorios son una herramienta de la gestión de documental (Document Management, DM) la cual se define como es el conjunto de productos y servicios orientados a la gestión de repositorios que permiten el control y la revisión de los documentos electrónicos localizados en toda la organización (Albalat, 2012).

## 2.4 Tipos de repositorios.

Osuna Alarcón & De La Cruz Gómez (2010) menciona que desde el año 2000 se ha producido una convergencia entre todas las herramientas de gestión de la información, de forma que en la actualidad encontramos soluciones globales que ofrecen soporte a todo el proceso de gestión de información en cualquier entidad, institución u organización. Estas herramientas han recibido la denominación de Sistemas de Gestión de Contenidos (CMS del inglés *Content Management Systems*), integrándose los Sistemas de Gestión Documental (EDMS del inglés *Electronic Document Management Systems*) con los de recuperación de información. Este mismo autor menciona la importancia de diferenciar entre los CMS y los EDMS, para lo cual menciona que:

La gestión de documentos tiene como objeto de trabajo los documentos como su propia nombre lo indica, en cambio la gestión de contenidos gestiona objetos que actúan como componentes de documentos virtuales, los objetos son tratados mediante un conjunto de procesos estructurados con la finalidad de producir publicaciones (Osuna Alarcón & De La Cruz Gómez, 2010).

En la bibliografía, existen diferentes denominaciones para tipos de sistemas que se ocupan de guardar documentos y registros de forma electrónica, entre ellos Adam (2008) menciona los mostrados en la Tabla 1.

**Tabla 1. Lista de acrónimos usados comúnmente.**

<b>Acrónimo</b>	<b>Significado</b>
<b>EDRMS</b>	Electronic Document and Records Management System
<b>EDMS</b>	Electronic Document Management System
<b>EDM</b>	Electronic Document Management
<b>ERM</b>	Electronic Records Management
<b>ERMS</b>	Electronic Records Management System
<b>DMS</b>	Document Management System
<b>DIP</b>	Digital Image Processing
<b>ECM</b>	Enterprise Content Management
<b>RM</b>	Records Management
<b>DM</b>	Document Management

Fuente: Adam (2008).

Tabla que muestra los acrónimos comúnmente aceptados para referirse a sistemas encargados de gestionar documentación electrónica.

En estos casos, estos sistemas necesitan de un repositorio de documentos, ya que son donde éstos guardan los documentos electrónicos que están bajo gestión. El repositorio debería de ser algún mecanismo centralizado para los documentos de la organización que permita a los usuarios recuperarlos a través de una funcionalidad de búsqueda o de navegación (Adam, 2008).

Como comenta Osuna Alarcón & De La Cruz Gómez (2010), la selección de un repositorio debería hacerse mediante un estudio y análisis de la organización que lo adopte, de los objetivos de la misma, de los procesos y recursos de información que requiere.

## **2.5 Alternativas de software para la implementación de un repositorio de información.**

Debido al boom de la información digital, en la última década se ha percibido un crecimiento en la cantidad de software orientado a este propósito y encontramos desde poderosas herramientas de gestión y digitalización de documentos hasta robustos sistemas en la nube, todos nacidos de la misma necesidad de la gestión de información.

### 2.5.1 Software comercial.

Existe una variedad de soluciones producidas por compañías de software especializadas que tienen años de experiencia en la arena de la gestión de documentos y registros (Adam, 2008). En esta sección se nombrarán algunas de ellas.

#### 2.5.1.1 Gratuitos.

Si bien existen sistemas populares basados en la nube tales como Dropbox (M.R. de Dropbox Inc), OneDrive, antes Skydrive (M.R. de Microsoft Corporation) y Google Drive (M.R. Google Inc), tanto en versiones de uso gratuito como comercial que podrían cumplir el objetivo de resguardo de información electrónica, una organización puede determinar no utilizar estos servicios por dos cuestiones, preocupación por la confidencialidad de la información y la responsabilidad derivada de incidentes relacionados con la infraestructura (ENISA, 2009).

Uno de los motivos por lo que la confidencialidad de la información se puede ver afectada, es debido a que las concentraciones masivas de recursos y de datos constituyen un objetivo más atractivo para los atacantes (ENISA, 2009). Consientes de los anterior varios autores han dado a conocer las vulnerabilidades de las que podrían ser víctimas los servicios en la nube, por ejemplo, Kholia & Przemysław (2013) presentaron una vulnerabilidad en la autenticación que podría comprometer información resguardada en este tipo de servicios.

Otro punto importante que recae dentro del aspecto de seguridad es la comunicación entre cliente y proveedor, pues estas grandes empresas de hospedaje cuentan con muchos clientes que no siempre podrían atender en un momento inmediato, o bien, si trasladamos a costos esta problemática, podría suponer un costo adicional el contar con un servicio exclusivo de atención técnica.

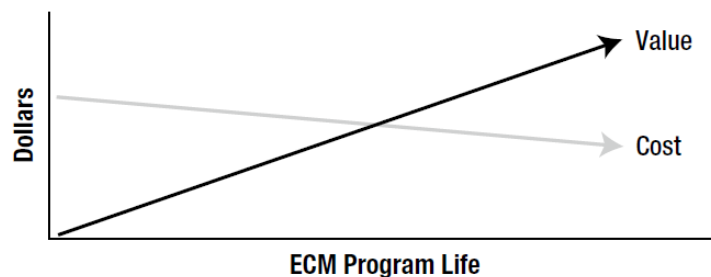
En cuanto a costos en este tipo de servicios de almacenamiento en la nube se reducen y conllevan flexibilidad, por lo que la migración a la computación en nube es una opción irresistible para muchas PYME. Sin embargo para que la computación en la nube alcance todo su potencial, se debe ofrecer solidez en la seguridad de la información (ENISA, 2009).

### 2.5.1.2 Bajo licenciamiento.

Muchas compañías han desarrollado y empujan el mercado hacia las soluciones de gestión de contenidos empresariales (ECM del inglés *Enterprise Content Management*). La ventaja de los ECM es que permiten a las organizaciones adquirir de forma modular funcionalidades tales como gestión de documentos, gestión de registros, gestión de contenidos Web, herramientas de colaboración, herramientas de flujo de trabajo entre otros (Adam, 2008).

Una parte a considerar acerca de estos sistemas ECM es su costo, si se deja de lado a organizaciones pequeñas y de estructura sencilla, un ECM podría ser costoso en un inicio debido al trabajo de análisis y diseño previo que se tiene que realizar para la implementación en una organización mediana. Aunque el costo de lo invertido (ROI del inglés *Return Of Investment*) retornará conforme avance el tiempo agregando valor (Goodyear, 2013). Ver Ilustración 3.

Ilustración 3. Relación entre costo y beneficio de un ECM.



Fuente: Goodyear (2013).

Ilustración que muestra la relación costo-beneficio durante la vida de un ECM en la organización.

Dentro del software comercial se encuentra Microsoft *SharePoint* como una plataforma que la organización puede utilizar para gestionar la información que adquiere o produce para después usarla en sus operaciones. En su versión inicial del 2001 comenzó como una plataforma de gestión de documentos y colaboración para los usuarios que proveía de un repositorio centralizado más rico en características que el compartimiento simple de archivos a través de la red (Goodyear, 2013).

Ahora SharePoint evolucionó para ofrecer una mejor experiencia para la colaboración entre usuarios en una organización, permitiendo ser una plataforma para almacenaje de varios tipos de información durante todo el ciclo de vida, desde la creación hasta la búsqueda, retención y disposición (Goodyear, 2013).

Xerox *DocuShare* de Xerox Corporation provee una familia de productos para la gestión de documentos y registros que incluyen, *Xerox DocuShare Enterprise Content Manager*, *Xerox DocuShare CPX Enterprise Content and Process Manager* y *Xerox DocuShare Records Manager*. Xerox DocuShare ECM permite implementar el flujo de los documentos, control de versiones, colaboración entre otras facilidades de publicación. DocuShare Records Manager puede ser integrado para ofrecer una completa solución EDRMS (Adam, 2008).

### 2.5.2 Software de código abierto.

La mayor ventaja de los desarrollos de software de código abierto encaminados a repositorios de información digital es la adaptabilidad, tal como lo menciona De La Vega (2008). Como menciona Fresno Chávez (2012) la tendencia mundial en repositorios es implementarlos por medio de software de código libre, sin embargo, el hecho de mencionar que sea de código abierto no significa que sea gratuito, pues conlleva gastos de soporte y personalización (Rodríguez, 2013).

Dentro del software que mayormente se usa para gestión de bibliotecas digitales de tipo SIAB es DSpace con el 41.5% de los repositorios a nivel mundial (OpenDOAR, 2014). Adicional a DSpace Fresno Chávez (2012) menciona también Eprints, CDSware, Fedora, i-TOR, y MyCoRe como los más conocidos y utilizados en la actualidad, este mismo autor vuelve a señalar a DSpace y Fedora como dentro de los más citados para bibliotecas digitales, además se habría que adicionar a Greenstone a esta lista.

El software de código abierto como menciona anteriormente, es comúnmente más utilizado para repositorios de bibliotecas de instituciones académicas, un ejemplo de esto es la Biblioteca Digital de Cataluña la cual cuenta con 5 repositorios: TDX para tesis, RACO para revistas, RECERCAT para documentos de investigación, MDC para imágenes, y MDX para material docente, cuatro de los cinco repositorios están construidos bajo software libre (Anglada I. de Ferrer, Reoyo Tudo, Cambras, & De La Vega, 2008).

Una alternativa adicional de tipo ECM es Alfresco. Con más de veinte años de experiencia en sistemas de gestión de contenidos el equipo de Documentum de EMC Corporation crean Alfresco como plataforma de escala empresarial para gestión y distribución de contenido, de los cuales se incluyen sitios Web, documentos, registros y multimedia. Alfresco es una sistema ECM de código abierto creado en el año 2005 como una opción alternativa al software propietario en un mercado de \$4,000 millones de dólares (Caruana, Newton, Farman, Uzquiano, & Roast, 2010).

Otras características de Alfresco que lo identifican son sus capacidades de control de los contenidos, tales como la gestión de metadatos, control de versiones, gestión del ciclo de vida, flujo de trabajo, búsqueda, asociaciones con otro contenido, comentarios y muchos más. Esto permite que las organizaciones encuentren información oportuna dentro, además permite publicarla a través de la Web o cualquier otro canal que sea apropiado para los usuarios (Caruana, Newton, Farman, Uzquiano, & Roast, 2010).

### 2.5.3 Software propio.

Como es el caso de esta investigación, la organización podría promover el desarrollo y puesta en marcha de su propio repositorio. La gran ventaja de desarrollar uno en casa a adquirirlo comercialmente es que la organización puede diseñarlo y desarrollarlo conforme a las necesidades específicas y objetivos establecidos (Adam, 2008). Desarrollar este tipo de sistema tiene sus particularidades, pues generalmente los usuarios finales contribuyen a lo largo de todo el proceso, además de que estos sistemas deben tomar ventaja del ambiente Web (Wen, Ye, & Lin, 1998).

Una de las ventajas de desarrollar un sistema en casa son los costos que puede haber entre licenciar un software comercial y desarrollar uno en la organización. Supóngase una organización mediana donde se tuviera que implementar el sistema por el proveedor. Se tendrían que contemplar costos de adquisición del producto, licencias de usuario, costos de consulta y mantenimiento, podría decirse que sería conveniente considerar desarrollar un sistema en casa si esto conlleva un costo benéfico (Adam, 2008).

## 2.6 Herramientas para el desarrollo del repositorio de información digital.

La presente investigación desembocará en una aplicación bajo una plataforma web, por lo que conviene traer a mención las herramientas que se consideran adecuadas para el desarrollo del proyecto, en este sentido es necesario poner en contexto donde se llevará a cabo el desarrollo e implementación, ya que existen lenguajes de programación como ASP.NET cuyas herramientas de programación pueden significar un costo en comparación con el lenguaje PHP donde existe un mayor número de opciones tanto gratuitas como de licenciamiento. En este sentido, las organizaciones quizás cuenten con herramientas que sean proporcionadas por motivos de licenciamiento o simplemente por común acuerdo del equipo de trabajo.

### 2.6.1 Servidor web.

Una parte fundamental de un sistema web es el servidor web. Un servidor web es un programa que se ejecuta en segundo plano en un equipo y que escucha en un puerto TCP/IP particular para detectar peticiones entrantes y establecer la comunicación con el navegador mediante el Localizador Uniforme de Recursos (URL del inglés *Uniform Resource Locator*). El tráfico de subida del navegador hacia el servidor Web consiste principalmente en peticiones mediante el protocolo de transferencia de hipertexto (HTTP del inglés *Hypertext Transfer Protocol*) de archivos específicos, sin embargo existen mecanismos como las interfaces de entrada común (CGI del inglés *Common Gateway Interface*), donde a través de éstos los usuarios pueden suministrar información que es recibida por el servidor y entonces se ejecutan una secuencia de comandos CGI que define como usar dicha información. Dicho uso podría ser por ejemplo la consulta a un servidor de base de datos, realizar una transacción financiera en línea o cualquier otro uso (Zacker, 2001).

Existen una gran variedad de programas de aplicación que pueden trabajar como servidores Web ejecutándose sobre múltiples sistemas operativos. Sin embargo, en el caso de *Windows Server* es una buena alternativa instalar el servidor Web que ya viene incluido con el sistema operativo, pues tiene un buen rendimiento, saca ventaja de la gran integración que tiene con el sistema operativo y es una opción gratuita (Poratti, 2004).

Debido a que la organización en el año 2004 inicio la adopción de software basado en la web, ésta adquirió equipo en el cual se ejecutaba Windows en su versión server 2003 estándar, satisfechos con los resultados de este sistema, se migró a Windows server 2008 R2, en la organización siempre se ha utilizado el servidor *Web Internet Information Services (IIS)* en medida por tener una integración con el sistema operativo nativa, ser modular y estar conformado por más de 40 características que le permiten

ser extensible, además de ofrecer la dualidad de poder interpretar el lenguaje propietario de Microsoft .NET Framework, PHP, entre otros, así como posibilidad de administrar servicios como FTP, WebDav desde una misma consola (Volodarsky, 2007). Por lo que IIS da un sentido de dualidad al poder integrar lenguajes propietarios como de acceso abierto. En cuanto a aspectos de seguridad de IIS versión 7.5, está construido con base en IIS versión 6.0 el cual ha logrado registros de seguimiento de confiabilidad y seguridad altas, con un solo boletín de seguridad crítico desde su lanzamiento (Volodarsky, 2007).

#### 2.6.2 Procesamiento del lado del servidor, *backend*.

Dentro de las aplicaciones Web es posible distinguir aquellas que son catalogadas como sitios web estáticos y dinámicos, un sitio Web estático es aquel que no utiliza lenguajes dinámicos y devuelve la misma respuesta siempre. Para lograr un sitio Web dinámico se tendría que utilizar un lenguaje de programación que permita modificar la petición en tiempo real y enviarla al navegador, por lo que dicho sitio Web podría comportarse con base a la información ingresada por el usuario (Minera, 2010).

La Web ha venido evolucionando no solo debido a cambios tecnológicos, si no que van de la mano con la evolución de los hábitos de los usuarios, las tendencias en los modos de navegación, las necesidades del mercado y aspectos culturales (De Luca, 2011). Benítez (2013), menciona un artículo de Driscoll (2011) que divide a la Web en décadas, lo que nos permite dar un panorama de su evolución y hacia dónde se dirige. Básicamente el artículo distingue tres eras, ver Tabla 2.

**Tabla 2. Uso de tecnologías en las eras web.**

1990-1999	2000-2009	2010 - ¿?
Era HTML	Era LAMP (Linux Apache MySQL PHP)	Era Javascript
Se caracterizaba por una web con páginas estáticas y generalmente estaban escritas en un solo lenguaje como HTML.	Se caracteriza por la introducción de tecnologías de base de datos y de lenguajes que operan junto al servidor web para ofrecer información cambiante y actualizada (dinamismo) a las páginas. Dicho dinamismo es mayormente del lado del servidor.	Es una evolución de las anteriores donde se incorpora dinamismo del lado del cliente, en conjunción con la del servidor, para ofrecer interfaces ricas y ágiles al usuario. Se caracteriza por el uso de frameworks para el manejo del frontend, AJAX y librerías Javascript para el manejo del <i>DOM</i> .

Fuente: Elaboración propia con datos de Driscoll (2011).

Tabla que muestra el uso de las tecnologías en la web en diferentes eras y como ha ido cambiando desde la introducción de internet para ofrecer una experiencia más rica al usuario.

Bajo el contexto de las eras de la web ejemplificada en la Tabla 2, sirve de base para determinar que tecnologías se pueden utilizar en la construcción del repositorio de información digital dada su característica de ser un sistema web, de tal forma que el producto sea acorde a la cultura actual de los usuarios en donde buscan un rol activo y de participación, como menciona De Luca (2011).

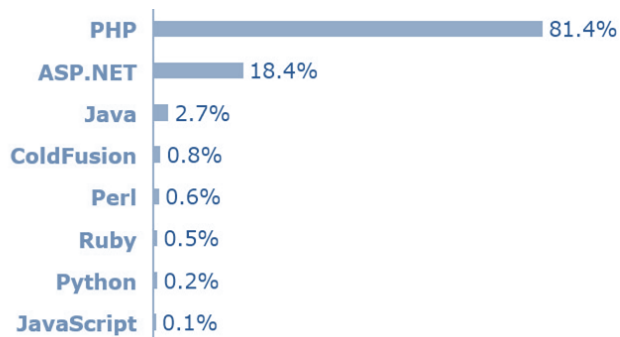
Como menciona Driscoll (2011) con la evolución de las eras Web, dentro de la era LAMP se introdujo el uso de las bases de datos en las aplicaciones Web. El uso de bases de datos representa ventajas, por ejemplo la independencia de los datos y su tratamiento, es decir el cambio de datos no implica cambio de programas y viceversa, lo que induce menor costo de mantenimiento, otra es la coherencia en los resultados, ya que se disminuye la redundancia en el acceso a los datos por parte de los usuarios (Vázquez, 2006).

MySQL es una de las aplicaciones más usadas en servidores web para la administración de bases de datos. Su adopción puede deberse por tres factores que la caracterizan: su velocidad, su estabilidad y su facilidad de uso. Si deseamos incorporar a nuestro servidor web una completa herramienta para contar con infinitas posibilidades de interacción con el usuario (por ejemplo, sistemas de validación, foros de discusión, libros de visitas, etc.), MySQL Server será la mejor opción (Vázquez, 2006).

En cuanto a la lógica de negocio y procesamiento en el servidor (*Backend*), la utilización de una base de datos es requerido para la presente investigación, puesto que es necesario el resguardo de información de usuarios, roles, metadatos de los archivos, entre otros. Además, como parte de las buenas prácticas en el desarrollo del *Backend* de los cuales hace mención Benítez (2013), el acceso a datos físicos debe ser en lo posible utilizando una capa de abstracción de datos.

Otra herramienta a considerar es el referente al lenguaje de programación del lado del servidor que extraerá la información de la base de datos para enviarla al servidor web y éste al cliente. Por lo que los lenguajes de programación *PHP*, *Python* y *Ruby* son opciones abiertas generalmente usadas, en tanto que las opciones comerciales son *ASP.NET*, *Java*, *Coldfusion* como muestra W3Techs (2013). Ver Ilustración 4.

Ilustración 4 Uso de lenguajes de programación del lado del servidor al 6 de diciembre de 2013.



Fuente: (W3Techs, 2013)

Ilustración que muestra el porcentaje de uso de los lenguajes de lado del servidor donde se observa una importante diferencia del lenguaje PHP respecto al cercano ASP.NET.

PHP es uno de los lenguajes más utilizados en la actualidad, se utiliza mayormente para desarrollos de sitios Web pero para muchos es ya un lenguaje de propósito general (Minera, 2010). La reputación de PHP generalmente no es la mejor como lenguaje de alta usabilidad, de hecho como se menciona en Benítez (2013), PHP obtiene la calificación más baja en usabilidad, sin embargo, con el uso de buenas

prácticas en el desarrollo como las que menciona Guzel (2011) y con la adopción de una arquitectura por capas como menciona Pressman (2010) se podría sobrellevar esta debilidad del lenguaje. Dentro de las ventajas de PHP se pueden citar las siguientes (Minera, 2010):

- Es libre y gratuito.
- Disponible en variedad de plataformas y sistemas operativos.
- Soporte para múltiples base de datos.
- Evolución constante gracias a su comunidad.
- Facilidad de aprendizaje.

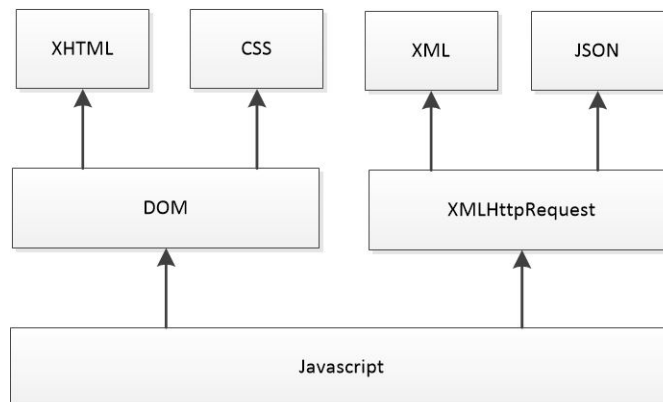
### 2.6.3 Procesamiento del lado del cliente, *frontend*.

Las hojas de estilos en cascada (CSS del inglés *Cascading Style Sheets*), han resultado la herramienta para dar formato y color a los contenidos de la Web. Así, cualquier documento en el lenguaje de marcas de hipertexto (HTML del inglés *HyperText Markup Language*) es formateado con CSS, las hojas de estilo nacen de la necesidad de separar el contenido de la presentación, de tal forma que un mismo contenido pueda ser presentado en diferentes dispositivos, servicios, contextos o aplicativos (Collell Puig, 2014).

Con la entrada de las aplicaciones de la Web 2.0 los usuarios pasaron de un rol pasivo a buscar alternativas de participación que permitieran una web más social (De Luca, 2011). El principio de la Web 2.0 es el uso de herramientas y tecnologías que permitan participar de manera más activa al usuario, interactuando con el sitio Web, evitando demoras entre peticiones cliente y servidor que dificulten la experiencia de usuario (Minera, 2007).

Para lograr lo anterior la tecnología *AJAX* viene a apoyar a la Web 2.0, ya que uno de los objetivos de Ajax es acercar la funcionalidad de las aplicaciones Web a la interactividad ofrecida por las aplicaciones de escritorio (Minera, 2007). En realidad es una nueva forma de utilizar tecnologías ya existentes como (X)HTML, CSS, XML, *Javascript*, DHTML entre otras (Ver Ilustración 5); el término fue concebido por *Jesse James Garret* como acrónimo de *Asynchronous JavaScript and XML* (Minera, 2007). Lo interesante de *AJAX* es que permite actualizar el contenido de una página web servida al cliente sin necesidad de volver a cargar todo el contenido, para ello se hace uso del objeto *XMLHttpRequest* para enviar y recibir datos entre el navegador y el servidor web asíncronamente (Minera, 2007).

Ilustración 5. Tecnologías agrupadas bajo el concepto de Ajax.



Fuente: Eguíluz Pérez (2008).

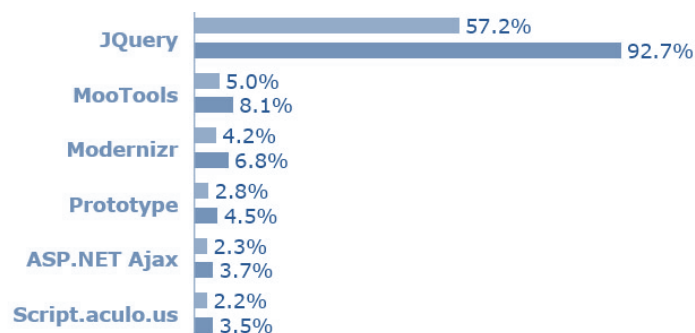
Ilustración que muestra aquellas tecnologías en las que el término AJAX se soporta.

Como puede observarse en la Ilustración 5, *Javascript* es el pilar de la tecnología *Ajax*, este es el lenguaje de programación utilizado en el desarrollo de aplicaciones web por parte del cliente (Collell Puig, 2014). Nace gracias a *Netscape Corporation* en 1995, paralelamente Microsoft crea el lenguaje *JScript* el cual es muy similar pero con sus diferencias, tales como la forma de interactuar con el DOM, el sistema de eventos, entre otros menciona Collell Puig.

Al principio la programación de lado del cliente era terriblemente difícil, puesto que había que trabajar con cada una de las especificaciones para los diferentes navegadores, lo que provocaba que el código que se generaba fuera poco sólido y mantenible. Para solucionar estos problemas de interacción del lenguaje con los navegadores, nacieron bibliotecas cuyo objetivo es conseguir una interfaz de programación para aplicaciones (API del inglés *Application Programming Interface*) común a los diferentes navegadores (Collell Puig, 2014).

Dentro de las librerías *Javascript* más importantes se encuentran *JQuery*, *Prototype*, *Mootools* que permiten una manipulación del DOM e interacción con el usuario a través de eventos (Collell Puig, 2014). De esta forma se ofrecen interfaces ricas e interactivas al usuario. En la Ilustración 6 se observa el uso de las principales librerías *Javascript* a nivel mundial, donde *JQuery* ha sido adoptada por la mitad de los sitios web en el mundo (Buckler, 2012).

Ilustración 6 Uso de librerías Javascript en sitios web al 6 de diciembre de 2013.



Fuente: (W3Techs, 2013)

Ilustración que muestra el porcentaje de uso de los lenguajes de lado del cliente donde se observa una importante diferencia de la librería *JQuery* respecto a otras sus más cercanas.

Con las tecnologías antes mencionadas se logra que el usuario reciba la información que desea rápidamente y con interfaces dinámicas que proveen al usuario una experiencia como la búsqueda por las aplicaciones de la Web 2.0.

## 2.6 Herramientas para el entorno de desarrollo.

Teniendo en cuenta que tanto el lenguaje HTML como CSS y Javascript son lenguajes que se pueden escribir en cualquier editor de textos y que no necesitan compilación, son claras las ventajas que ofrecen algunas herramientas de desarrollo, tales como las siguientes: identificación de código con diferentes colores, sistema de ayuda en las sintaxis y también otras facilidades relacionadas con la construcción de código (De Luca, 2011).

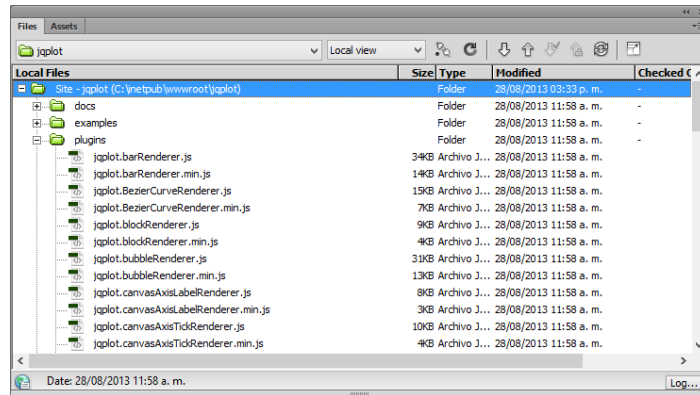
Dentro de las herramientas What You See Is What Get (WYSIWYG) para lenguajes y tecnologías Web, sin duda el más famoso es Adobe Dreamweaver (De Luca, 2011). Dreamweaver provee ocho vistas de espacio de trabajo para satisfacer diferentes tipos de necesidades y desarrolladores (Axzo Press, 2011), pudiéndose escoger la vista combinada de diseño y código, si se trabaja directamente con el código se podrá tener mayor control sobre el desarrollo que se esté realizando (De Luca, 2011).

Eventualmente dentro de un desarrollo Web se llegan a tener una gran cantidad de archivos y recursos que son parte del proyecto, tales como: páginas Web HTML, imágenes, archivos CSS u otros, por lo que es importante definir la estructura del sitio para empezar con el pie derecho (Bruce, Ray, & Ness, 2011).

Existen muchas opiniones acerca de la forma correcta de organizar la estructura de un proyecto Web, algunos gustan de seccionar todo en carpetas y subcarpetas, otros prefieren estructuras más sencillas y concentrar grandes cantidades de archivos en menos carpetas, cada desarrollador tiene su método ideal de organización conforme a su experiencia (Bruce, Ray, & Ness, 2011).

*Adobe Dreamweaver* ayuda a gestionar la estructura de los proyectos adecuadamente, ya que desde un inicio se puede crear carpetas agrupadas que contenga los archivos del proyecto (Ver Ilustración 7).

### Ilustración 7 Administración de sitios Adobe Dreamweaver CS5.



Fuente: Elaboración propia, Adobe Dreamweaver es propiedad de Adobe Systems Incorporated.

Ilustración que muestra el administrador de archivos de sitio de *Adobe Dreamweaver CS6* mediante el cual permite obtener una vista jerárquica de las fuentes del proyecto.

Un entorno IDE como *Dreamweaver* permite un mantenimiento del sistema ameno ya que cuenta con poderosas herramientas edición de código, búsqueda de texto en todo el sitio, reemplazo de código, cliente FTP, desde una interfaz visual.

En cuanto al tratamiento de la lógica de negocio y procesamiento en el servidor (*Backend*), la utilización de un sistema de gestión de bases de datos (DBMS del inglés *Data Base Management System*) es deseable. La instalación de MySQL no incluye herramientas para las tareas de administración comunes, tales como el monitoreo del servidor y comparación de datos entre servidores, no así en sus versiones comerciales, es por ello que existen herramientas tanto de terceros como de código abierto para auxiliar en estas cuestiones (Schwartz, Zaitsev, & Tkachenko, 2012). Hay ocasiones en que será más cómodo sacar provecho de ciertos beneficios como realizar el diseño de la base de datos y sus tablas a través de este tipo de herramientas (Minera, 2010).

Las herramientas con interfaz gráfica de usuario (GUI del inglés *Graphical User Interface*), ayudan a ejecutar sentencias en lenguaje de consulta estructurado (SQL del inglés *Structured Query Language*), crear tablas, usuarios, entre otras tareas, que ayudan agilizar el trabajo y disminuir la incidencia de

errores. Dentro de este rubro de herramientas se citan como populares las siguientes: MySQL Workbench, SQLyog, phpMyAdmin, Adminer (Schwartz, Zaitsev, & Tkachenko, 2012).

SQLYog es una de las GUI más populares con muchas características que otras no tienen, aunque solo está disponible para *Microsoft Windows*. Cuenta con versiones bajo licenciamiento como gratuita (Schwartz, Zaitsev, & Tkachenko, 2012). Ver Ilustración 8.

Ilustración 8 Herramientas adicionales de SQLYog.



Fuente: Elaboración propia.

Ilustración que muestra la interfaz grafica de la herramienta para gestión de base de datos llamada *SQLYog* mediante la cual se puede trabajar con las base de datos *MySQL* mediante objetos gráficos y que cuenta con una versión gratuita llamada *SQLYog Community Edition*.

## Capítulo III Metodología.

### 3.1 Investigación aplicada al desarrollo de software.

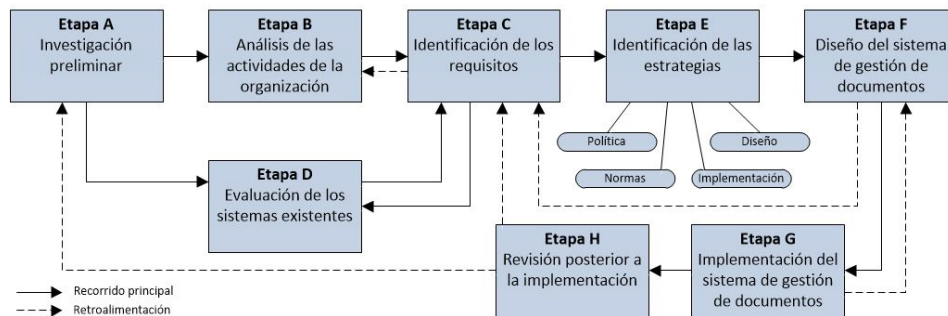
La presente investigación será aplicada al ramo de las TI, ya que este conocimiento producirá un sistema software para resolver un problema actual dentro de la organización. El desarrollo del software sería apoyado por la metodología para diseño e implementación de sistemas de gestión de documentos y registros (*DIRKS del inglés Designing and Implementing Recordkeeping Systems*) mencionada en la primera parte de la norma ISO 15489 apartado 8.4. Lo anterior mediante la aplicación de la metodología incremental del ciclo del software para el desarrollo de las funcionalidades, durante este lapso se utilizarán herramientas de ingeniería asistida por computadora (*CASE del inglés Computer Aided Software Engineering*) para el apoyo en este ciclo.

### 3.2 Metodología DIRKS.

Como investigación orientada a un desarrollo de producto software y dado que éste tiene el objetivo de fungir como sistema de gestión de documentos se considerará la metodología *DIRKS* (Ilustración 9), esta metodología es parte del apartado 8.4 de la norma ISO 15489 la cual es un compendio de buenas prácticas que ayuda a orientar a todas aquellas organizaciones que persiguen la mejora en la gestión documental.

La metodología *DIRKS* está destinada a garantizar que los registros y la gestión documental estén firmemente basados en las necesidades comerciales de la organización (Font Aranda, Ruiz Rodríguez, & Mena Mugica, 2012). No está concebida de un modo lineal, las tareas pueden realizarse en diferentes etapas, de forma reiterada, parcial o gradualmente, de acuerdo con las necesidades de la organización, los requisitos formales de conformidad y los cambios operados en el entorno de la organización y de la gestión de documentos de archivo (Alonso, García, & Lloveras, 2007).

**Ilustración 9 Metodología DIRKS (Designing and Implementing Recordkeeping Systems).**



Fuente: (Sitio Web EcuRed, 2011).

Ilustración que muestra las fases de las que se compone la metodología DIRKS mencionada en la norma ISO 15489 apartado 8.4.

Para la realización de la presente investigación se centrará la atención en las etapas *DIRKS*, ya que como metodología se basa en metodologías tradicionales de diseño de sistemas (Sitio Web EcuRed, 2011), por lo que es una metodología válida tanto para la construcción de un sistema de gestión documental como para la revisión de otros ya existentes (Font Aranda, Ruiz Rodríguez, & Mena Mugica, 2012). La organización cuenta con procedimientos específicos para el trato de la documentación. Es precisamente en los procedimientos de la organización, sus políticas, sus normas, sus valores en los que la metodología hace hincapié, siempre con el objetivo de que el software vaya enfocado a las necesidades comerciales de la organización. Por ello es importante determinar los requisitos funcionales en este tipo de software con base en lo mencionado, ya que de ello depende una armonía con los procedimientos y sistemas actuales.

Por lo consiguiente en la Tabla 3 se menciona como auxiliará cada de las etapas DIRKS en la investigación.

**Tabla 3. Resultados esperados en cada etapa de la metodología DIRKS.**

Etapa	Resultado
A. Investigación preliminar.	Examen de la estructura de la organización, su entorno legal y normativo así como los factores críticos y debilidades en relación a la gestión de documentos.
B. Análisis de las actividades de la organización.	Esquemas donde se representarán los flujos de los procesos derivado del análisis de procedimientos o por medio de entrevistas.
C. Identificación de los requisitos.	Se determinará de acuerdo a los procedimientos implementados por ISO 9001 las necesidades de evidencia de información, además de la tabla de acceso y seguridad a los mismos. Esto puede ser modelado por medio de casos de uso mediante UML.
D. Evaluación de los sistemas existentes.	Obtención de funcionalidades del sistema.
E. Identificación de estrategias para cumplir los requisitos.	En esta etapa se obtendrá la planeación del proyecto software de acuerdo a las políticas, normativas, procedimientos de la organización.
F. Diseño de un sistema de gestión de documentos de archivo.	Asegurar que el producto software sea acorde a los requisitos y análisis previamente realizados, mediante la metodología de desarrollo iterativa incremental.
G. Implementación de un sistema de gestión de documentos de archivo.	Implementación de la investigación de forma sistemática y de acuerdo a lo planificado en la etapa E.
H. Revisión posterior a la implementación.	Recopilación de información acerca de la penetración del producto software en las actividades de la organización y/o procedimientos. En esta investigación este punto escapa del alcance, sin embargo se realizarían pruebas de software y con la ayuda de la metodología incremental se apegaría a los requisitos de la organización.

Fuente: Elaboración propia con información de Alonso, García, & Lloveras (2007) y UNE-ISO (2005).

Tabla que muestra las etapas de DIRKS y en qué forma vendrán a apoyar al proyecto, mostrando para ello el resultado esperado para cada una de las fases.

Con las etapas de la A a la D se obtendrá un diagnóstico de la organización que permita obtener las funcionalidades del sistema, la etapa E servirá para obtener la planeación del proyecto de acuerdo a los lineamientos de la organización, en la siguiente etapa será realizado el diseño del sistema conforme a las necesidades detectadas y posteriormente en la G será desarrollado e implementado.

### 3.2.1 Investigación preliminar.

Durante la investigación preliminar se estudiará la estructura de la organización que permita determinar las áreas involucradas en el manejo de la información, así como entender el ciclo de vida de los

documentos que serán resguardados en el repositorio. Por último se indagará como se lleva en la actualidad la gestión de los documentos digitales de los expedientes y cuáles son sus requisitos para el cumplimiento del servicio.

### **3.2.2 Análisis de las actividades de la organización.**

En esta etapa se analizarán los procedimientos de la organización que estén involucrados en el resguardo de la información, esto permitirá generar diagramas de modelos y notación de procesos de negocio (BPMN del inglés *Business Process Model and Notation*) que ayude a detectar las funcionalidades del software para el suministro y recuperación de la información. En caso de que se necesite profundizar en algún procedimiento, entonces se recurrirá a la entrevista con los involucrados de las áreas de acuerdo al organigrama de la organización. La información se obtendrá de tres fuentes principales, las cuales son:

- Procesos de la empresa.
- Sesiones de grupo.
- Escenarios.

### **3.2.3 Identificación de los requisitos.**

Se detectarán las necesidades de resguardo de información para determinar qué tipo de información será resguardada en el repositorio y con qué objetivo se encontrará en éste. Además, se obtendrán los requisitos de seguridad para el acceso a los documentos.

### **3.2.4 Evaluación de los sistemas existentes.**

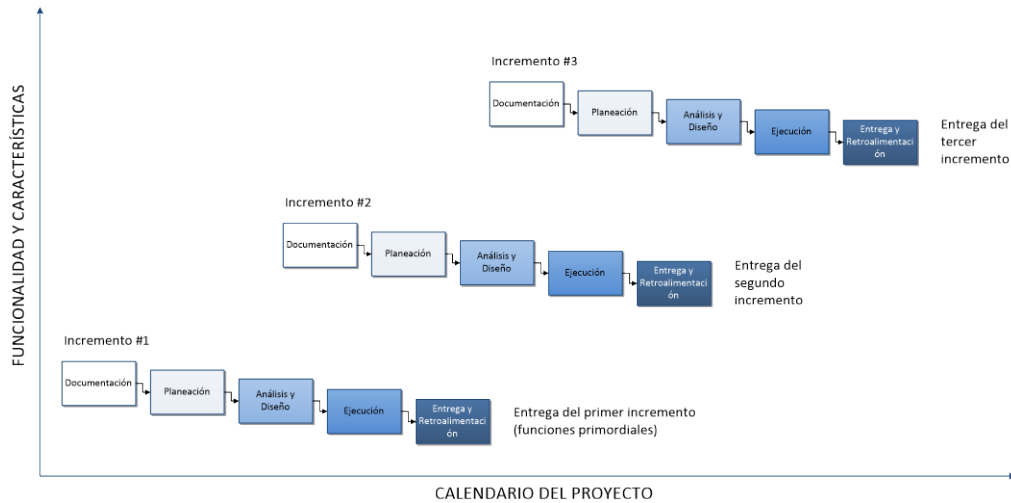
Con base en el diagnóstico de las primeras etapas A, B y C se determinará de qué forma los sistemas actuales cumplen con los requisitos de resguardo de información. Una vez recolectados los resultados de estas etapas se estará en condiciones de obtener las funcionalidades requeridas para el resguardo de la información.

### **3.2.5 Identificación de estrategias para cumplir los requisitos.**

Con base en las políticas, normativas, procedimientos de la organización se realizará la planeación del sistema de gestión documental, esto tendrá como resultado un cronograma de actividades el cual se adjuntará como ANEXO 1 dentro de la investigación.

Para llevar a cabo la planeación para el desarrollo del sistema se realizará bajo el modelo de trabajo incremental, es decir se entregarán partes funcionales del software mediante etapas (Ilustración 10), este modelo permite trabajar paralelamente con otras actividades conforme avanza el proyecto e incluso con otras actividades dentro de la empresa (Pressman, 2010). Además que en este tipo de aplicaciones tienden a ser trabajadas bajo este modelo (García, 2011), tal vez por la característica que menciona Cáceres & Marcos (2001), donde la parte iterativa permite reajustar y hacer cambios en los requisitos.

Ilustración 10 Modelo de trabajo incremental.



Fuente: (Pressman, 2010).

Ilustración que muestra la forma de trabajo para el desarrollo de las funcionalidades, la cual será llevada de forma incremental.

En cada incremento se entregará una parte funcional del sistema, por lo que el primer incremento se enfocará a desarrollar las funciones primordiales, tales como el control de acceso, después se entregarán funciones relativas a la administración de los documentos y así sucesivamente hasta agotar las funcionalidades planeadas.

### 3.2.6 Diseño de un sistema de gestión de documentos de archivo.

Se realizará la representación gráfica del sistema con base en los requerimientos detectados, es decir, la conceptualización del sistema por medio de modelos gráficos con ayuda del lenguaje unificado de modelado (UML del inglés *Unified Modeling Language*) para la representación de los requisitos funcionales. Para ello se utilizarán las siguientes técnicas de modelado de sistemas:

### ***3.2.6.1 El diseño de la base de datos.***

Menciona Pressman (2010) que si los requerimientos del software incluyen la necesidad de crear, ampliar o hacer interfaz con una base de datos, el equipo del software tal vez elija crear un modelo de datos como parte del modelado general de los requerimientos. El diagrama entidad-relación (DER) aborda dichos aspectos y representa todos los datos que se introducen, almacenan, transforman y generan dentro de una aplicación. El diagrama DER será desarrollado en este punto.

### ***3.2.6.2 El diagrama de casos de uso general, así como la descripción de cada uno de ellos.***

Se obtendrá el diagrama de caso de uso general a partir de las funcionalidades detectadas, así mismo dentro de este apartado se generará la relación de los casos de uso detectados y así como las descripciones para cada uno de ellos.

### ***3.2.6.3 Prototipo de cada funcionalidad.***

Uno de los atributos de una aplicación web es la usabilidad, el cual Pressman (2010) la define como “medición cualitativa de la facilidad y eficiencia con la que un humano emplea las funciones y características que ofrece el producto de alta tecnología”, menciona además que si los mecanismos de la interfaz están bien diseñados, el usuario se desliza por la interacción a un ritmo suave que hace que el trabajo se realice sin esfuerzo. Pero si la interfaz fue mal concebida, el usuario avanza y retrocede, y el resultado final es frustración y poca eficiencia en el trabajo.

Por lo anterior dicho se realizarán prototipos de cada funcionalidad detectada. Los prototipos serán realizados con la herramienta para diseño de interfaces *Balsamiq Mockups* (M.R. de

*Balsamiq Studios*). Con esto se desea obtener una validación de la interfaz y así pulir las funcionalidades acorde a las necesidades.

#### ***3.2.6.4 Diagrama general de arquitectura del software.***

Menciona Pressman (2010) sobre la importancia de la arquitectura, pues resalta las primeras decisiones que tendrán un efecto profundo en todo el trabajo de ingeniería de software siguiente y también importante en el éxito último del sistema como entidad operacional. Por esta razón se desarrollarán diagramas de arquitectura que ayuden a analizar la efectividad del diseño y que ayuden a decidir las herramientas adecuadas para su construcción.

#### ***3.2.6.5 Diagramas de secuencia.***

Con base a las descripciones de los casos de uso se obtendrán los diagramas de secuencia de aquellas funcionalidades que requieran mayor ejemplificación.

#### ***3.2.6.6 Diseño de los casos de prueba.***

En esta parte se diseñarán los casos de prueba para testear las funcionalidades que se vayan desarrollando, se realizará como mínimo un caso de prueba por funcionalidad, para ello se utilizará el formato mostrado en la Tabla 4.

Tabla 4. Formato para casos de prueba.

Numero de caso de prueba: << >>			
Título: Título de caso de prueba			
Área		Subarea	
Prioridad	Tipo	Frecuencia	Tiempo estimado
Descripción:			
Propósito de la prueba:			
Condiciones iniciales:			
Pasos:			
1.			
2.			
3.			
Resultados esperados:			
Notas:			

Fuente: Basado de Page, Jhonston, & Rollison (2009).

Tabla que muestra el formato que se tomará para la redacción de los casos de prueba a ejecutar una vez que hayan sido desarrollados.

El formato contiene un título con el cual identificar al caso de prueba junto con un consecutivo único, una descripción, los pasos a seguir para ejecutar la prueba, así como los resultados esperados al finalizar la prueba.

Además se enfocará a ejecutar el modelo de desarrollo del sistema, considerando las etapas:

- Ejecución (Codificación del software), Esta subfase se enfocará a ejecutar lo planeado en el análisis y diseño del sistema por medio de herramientas de desarrollo de software con base en los casos de uso definidos en el análisis. Debido a que las funcionalidades del sistema serian plasmadas en casos de uso, en la primera fase se llevarán a cabo las referentes al módulo de seguridad y así sucesivamente hasta agotar cada una de los casos de uso identificados.
- Validación, se verificará que el sistema realice correctamente las funcionalidades definidas y desarrollados y en su defecto realizar correcciones o ajustes que pudiesen haberse suscitado. Para ello se tomarían en cuenta los casos de prueba planificados en el diseño y se tomará nota de los resultados para su aborde.

### **3.2.7 Implementación de un sistema de gestión de documentos de archivo.**

Esta etapa será la estrategia para la implementación del sistema dentro de la organización y los ajustes necesarios a los sistemas hardware y software existentes. Se especificará qué roles serán los involucrados en el uso del nuevo sistema.

### **3.2.8 Revisión posterior a la implementación.**

Como menciona Peis Arredondo (2004), esta etapa tiene el objetivo medir la eficacia del sistema tras un periodo prudente de prueba, esto con la intención de detectar posibles deficiencias y subsanarlas. Esta revisión se irá realizando de forma periódica, ya que con los resultados de esta etapa pueden surgir nuevas funcionalidades no previstas en un inicio.

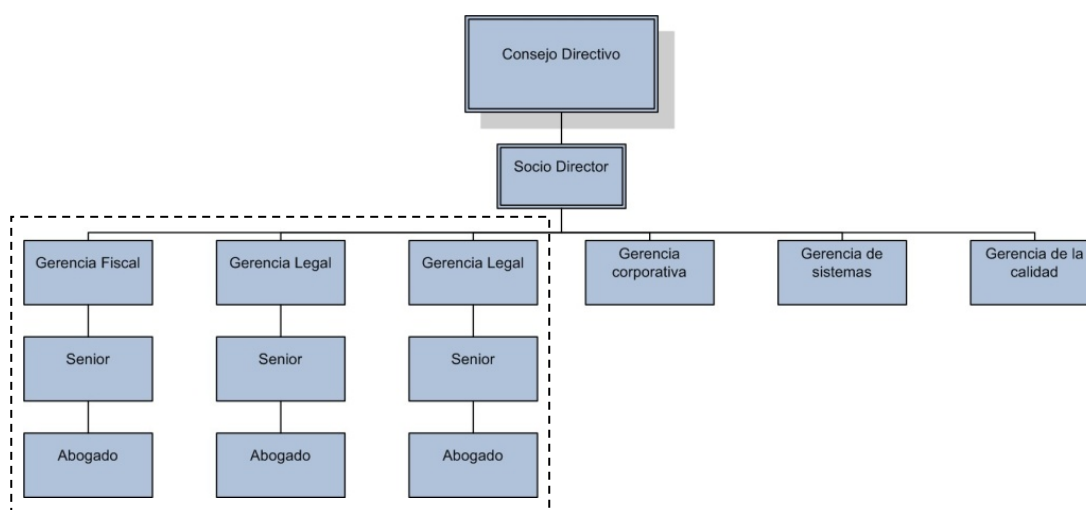
Como se mencionó en el alcance, esta etapa escapa del presente trabajo, sin embargo se dejarán plasmadas recomendaciones que surjan dentro del periodo de implementación y puesta en marcha del proyecto.

## Capítulo IV Desarrollo.

### 4.1 Investigación preliminar.

Grupo Farías nace hace más de dos décadas y se especializa en el ámbito fiscal de servicios legales, es la primera organización certificada en servicios legales de defensa fiscal, por la norma ISO 9001:2008. La certificación de Grupo Farías, Abogados, está enfocada específicamente a los servicios de asesoría y defensa fiscal (Sitio Web Grupo Farías, 2014).

Ilustración 11. Organigrama de la organización simplificado.



Fuente: Elaboración propia.

Ilustración que muestra un organigrama simplificado de la organización donde se identifica que parte de ella es la que está generando la información a gestionar.

La organización está conformada por áreas legales especiales, como se muestra en el recuadro de la Ilustración 11, así como el área administrativa. Ambas áreas convergen en los procesos dedicados a la prestación del servicio, por lo que ambas requieren que los recursos de información estén al alcance, en la organización se identificaron dos recursos principales de documentos, los cuales son:

- Registros del sistema de gestión de la calidad y
- Documentación del expediente.

Los recursos de documentos del expediente fue el objeto de esta investigación. Si bien el software no se diseñó para restringir el tipo de documentos que puede contener, la necesidad primordial es el control de los documentos del expediente, pues son los que más volumen de información generan, además de estar estrechamente ligados a la prestación del servicio.

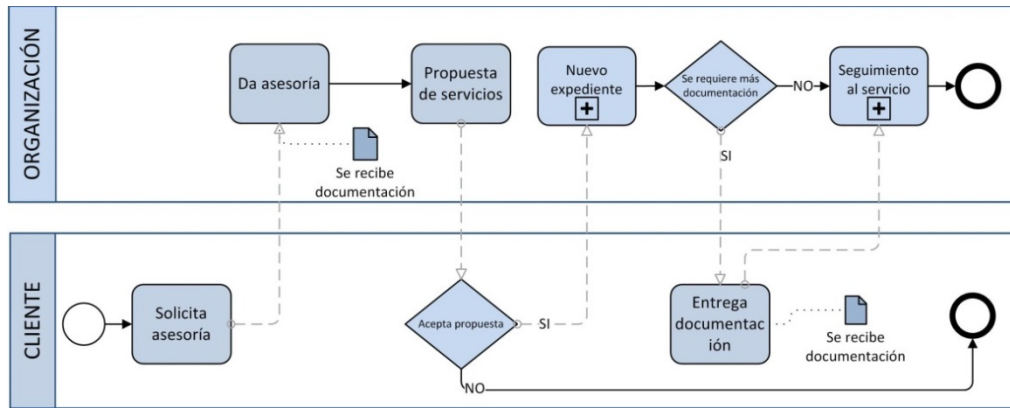
La documentación del expediente se ve iniciada en la mayoría de los casos desde la aceptación del servicio, el cliente entrega documentación que es requerida por la organización para la prestación del servicio, ésta es integrada a su expediente físico el cual se identifica mediante una clave y un folio que es asignado por el sistema de expedientes.

Para el control de los registros la organización estipula dentro del Procedimiento 3, Control de los registros, apartado 5.3, la disponibilidad que debe existir a través de los sistemas informáticos, por lo que dicha documentación debe estar al alcance del personal. Así mismo dentro de este mismo procedimiento, dentro del apartado 5.4, menciona el tiempo de retención de los registros, que en ciertas circunstancias puede ser de hasta 5 años.

## **4.2 Análisis de las actividades de la organización.**

La organización ejecuta el Plan de la Calidad durante el tiempo que dura la prestación del servicio, en él se identificaron dos momentos en los que se recibe información del cliente: antes de la elaboración de la propuesta de servicios y después de la aceptación. El Plan de la Calidad se resume en la Ilustración 12, dicha documentación es resguardada en carpetas físicas hasta el momento en que el cliente acepta el servicio, en este punto se digitaliza y se agrega al sistema de expedientes bajo el formato de documento portátil (PDF del inglés *Portable Document Format*).

Ilustración 12. Plan de la calidad resumido.

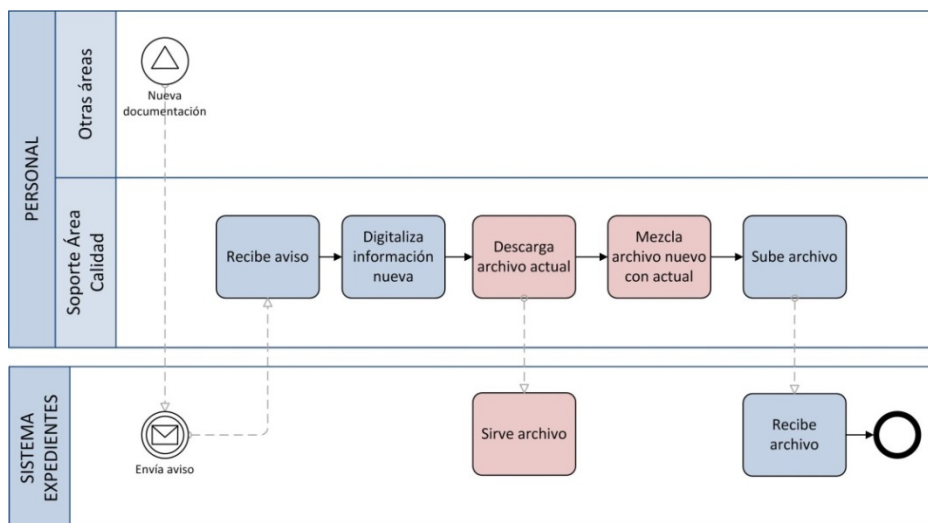


Fuente: Elaboración propia.

Ilustración que muestra el flujo de información dentro del plan de la calidad, donde se identifican los procesos que generan información.

El personal específicamente encargado de agregar dicha información al sistema es el puesto de Soporte al Área de la Calidad, sin embargo dicha actividad también puede ser apoyada por otras áreas como Atención a Clientes. El procedimiento que se realiza para proporcionar información digitalizada se detalla en la Ilustración 13.

Ilustración 13 Proceso para proporcionar información digital de Soporte al Área de la Calidad.



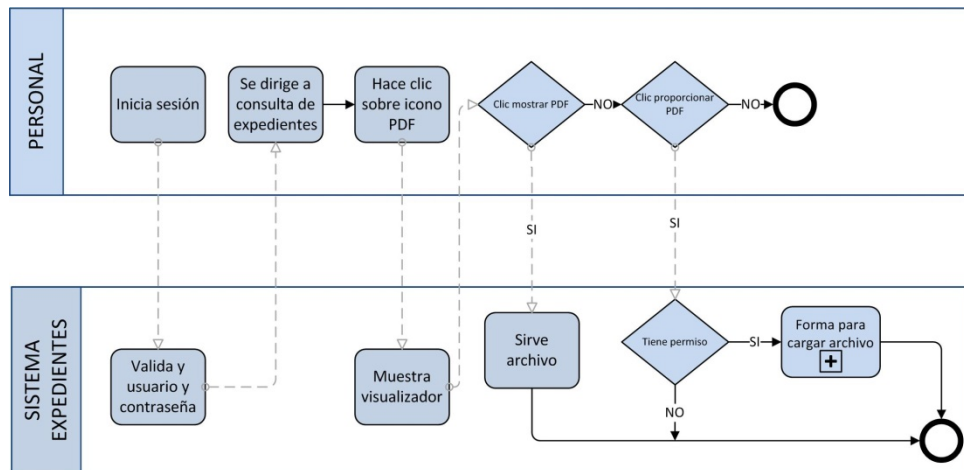
Fuente: Elaboración propia con datos de la organización.

Ilustración donde se obtienen los pasos que tiene que realizar el personal en la actualidad para adjuntar un archivo al sistema de expedientes, los pasos marcados en rojo representan tiempo de espera para la persona que ejecuta el proceso.

La notificación que recibe el personal para ejecutar el proceso de la Ilustración 13 puede ser por medio del aviso de nuevo expediente, o por el aviso de solicitud de escaneo que se recibe mediante correo electrónico a través del sistema de expedientes. Además se observó que para añadir nueva información al archivo PDF, se tiene que descargar el archivo actual, anexar la información y volverlo a proporcionar en el sistema, que son los pasos marcados en rojo en la Ilustración 13.

El personal accede a los documentos del expediente por la necesidad de dar cumplimiento al servicio. El proceso que tiene que realizar el personal para visualizar el archivo PDF se muestra en la Ilustración 14. Los usuarios solo pueden obtener la información una vez que se han identificado con sus credenciales en el sistema de expedientes.

**Ilustración 14. Proceso para visualizar la documentación del expediente.**



Fuente: Elaboración propia.

Ilustración que muestra los pasos necesarios para visualizar el archivo digital del expediente.

Con el objetivo de entender las actividades de la organización respecto a la obtención de los archivos electrónicos del expediente, se realizaron entrevistas a los usuarios donde se detectaron escenarios, los cuales se describen a continuación:

En un escenario suscitado en la organización, un usuario A solicitó el envío vía correo electrónico la sentencia digitalizada de un expediente, por lo que otro usuario B procedió a obtenerla del sistema de expedientes, se detectó que el usuario B tuvo que invertir un periodo de tiempo considerable en encontrar lo solicitado dentro de un archivo de 200 hojas, esto se debe a que algunos oficios no cuentan con un título que los identifique plenamente, por lo que el usuario tiene que invertir tiempo en leer, lo que retrasa la respuesta del usuario B.

### 4.3 Identificación de los requisitos.

Es en la documentación del cliente donde en los últimos dos años se ha detectado un aumento en la cantidad de información. Esto ocasiona que los documentos digitales cuenten con tamaños de archivo exponenciales (Tabla 5), lo que dificultaba su administración y la colaboración entre el personal de la organización.

**Tabla 5. Los 10 archivos con mayor número de kilobytes en el sistema de expedientes.**

Fecha de modificación	Tipo	Tamaño
18/03/2014	PDF	4,670,256
27/02/2013	PDF	388,592
17/07/2013	PDF	341,964
09/01/2013	PDF	323,680
24/03/2014	PDF	236,607
28/02/2014	PDF	216,335
22/11/2012	PDF	194,176
25/09/2012	PDF	172,799
21/02/2014	PDF	165,739
26/03/2014	PDF	153,572

Fuente: Elaboración propia con datos de la organización al 31 de marzo de 2014.

Tabla que muestra información de tamaño de los diez archivos digitales de expediente con mayor tamaño a la fecha.

Dentro de los análisis que se realizaron a los documentos, se detectó que algunos tipos de asuntos son propensos a generar más documentación, los cuales son los de tipo demandas y amparos, en menor

medida opiniones y constitución de empresas. En el caso de los expedientes de gran volumen, se debe a que la información proviene de hasta tres fuentes distintas, las cuales son:

- Información proporcionada por el cliente (Antecedentes).
- Información generada por la organización en la prestación del servicio.
- Información generada por las dependencias externas donde se llevan a cabo las gestiones.

En los expedientes que no se genera demasiada información es porque en su mayor parte es concebida dentro de la organización.

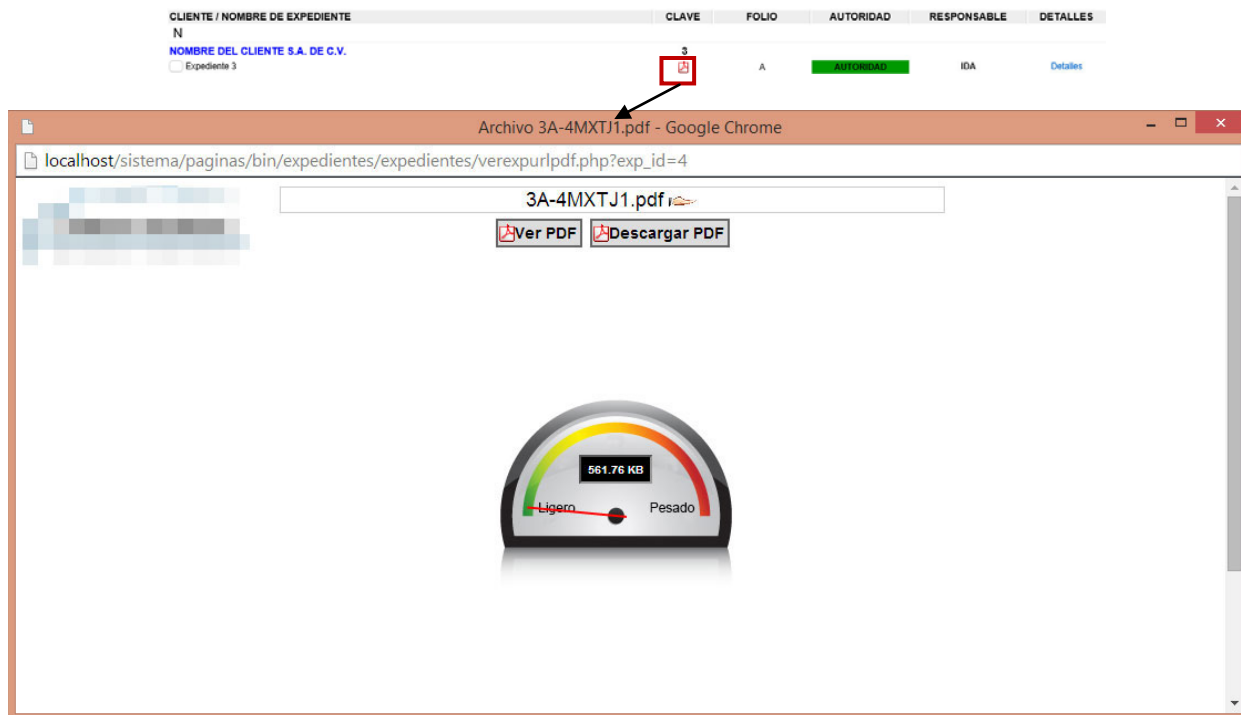
En cuanto a la seguridad, se detectó que todos los usuarios con acceso al sistema de expedientes se encuentran habilitados para consultar la información, sin embargo, solo los usuarios de ciertos roles pueden proporcionar el archivo electrónico PDF en el sistema, los cuales son: Administradores y Atención al Cliente.

El intercambio de archivos entre el personal también es llevado con frecuencia en la organización, por lo que éste recae de forma natural en el correo electrónico y a través de memoria externas USB, sin embargo el personal ha mencionado que en ocasiones hay retrasos en el servicio de correo, por lo que una solución sería el intercambio desde un servicio propio, ya que al momento de la investigación se contaba con el servicio de correo con un proveedor.

#### **4.4 Evaluación de los sistemas existentes.**

Según los resultados obtenidos del análisis, se determinó que la organización cumple con su procedimiento especificado en su documentación ISO en cuanto a la disposición de los documentos del expediente en los sistemas actuales, esto debido a que existe el mecanismo para consultarse a través del sistema de expedientes, ver Ilustración 15.

Ilustración 15. Mecanismo para consulta de la información electrónica del expediente.



Fuente: Sistema de expedientes de la organización.

Ilustración que muestra la funcionalidad para visualizar el archivo digital del expediente la cual cuenta con dos botones, ver archivo o descargar, así como una gráfica visual acerca del tamaño del archivo.

En otro apartado del sistema de expedientes se detectó un repositorio de documentos para formularios, cartas, procedimientos, entre otros, que son referentes a la documentación ISO de la organización, los archivos son mostrados de forma de árbol jerárquico, sin embargo no es posible agregar archivos o carpetas a través de una interface, si no solo a través del sistema de archivos del servidor.

Además, bajo la modalidad de *Propuestas de Mejora*, se detectó la petición número 126, la cual hace relación a la necesidad de digitalizar la documentación desde antes de la aceptación del servicio (Ver Ilustración 12), situación que no era posible. Asimismo, bajo la petición número 110 se expresa la necesidad de proporcionar más de un archivo al registro de expediente.

En este orden de ideas fue que se detectaron oportunidades de mejora en cuanto a la gestión de los documentos, puesto que si el archivo del expediente se dividiera, los tiempos de transferencia de datos y de búsqueda de información se disminuirían. Para esto, se diseñó el sistema de tal forma que permitiera tener varios archivos ordenados en forma de árbol jerárquico, así la organización pudo estructurar la información de acuerdo a sus necesidades.

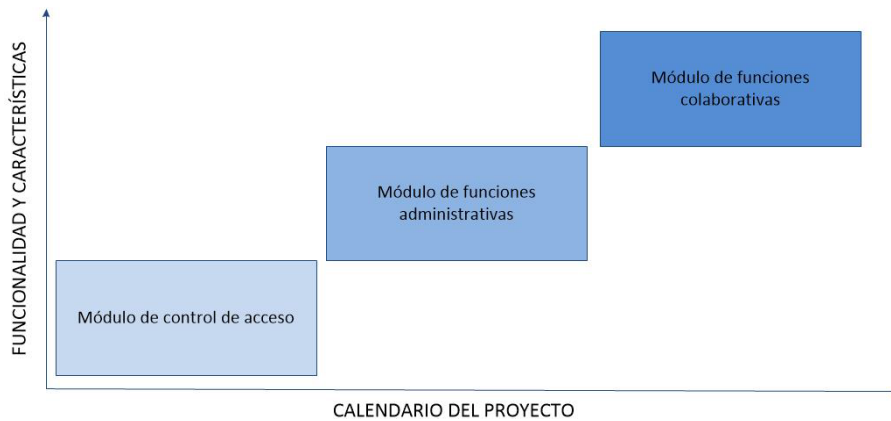
En relación a la seguridad de los documentos se detectó que era necesario que cada archivo o carpeta tuviese un mecanismo para el control de los archivos para cada usuario, ya que así podría ser utilizado para espacios de trabajo colaborativo. Además, se consideró el elemento de seguridad para los archivos que vayan a ser resguardados, de tal forma que un usuario sin autenticarse no pueda acceder a ellos escribiendo la URL completa de la ubicación en el navegador.

En la necesidad de compartir los archivos, se propuso que el repositorio permitiera enviar un enlace a un archivo recién compartido para que sea descargado desde la página principal del sistema de expedientes, ya que así es susceptible que la persona lo visualice. Además, para efectos de una mejor integración con los sistemas existentes, se ideó la implementación de la función para proporcionar archivos desde el sistema de expedientes, de modo que el flujo de trabajo para el usuario resultara transparente entre un sistema y otro.

#### **4.5 Identificación de estrategias para cumplir los requisitos.**

Para la ejecución del desarrollo de las funcionalidades, se agruparon funcionalidades en tres módulos: módulo de control de acceso, módulo de funciones administrativas y módulo de funciones colaborativas. Al concluir las primeras dos etapas, se colocó el producto en el servidor de producción, la tercera etapa se anexaría posteriormente, pues dentro de las primeras dos fases fue donde recayeron las funcionalidades de mayor interés para el cliente. Esto se puede apreciar en la Ilustración 16.

Ilustración 16. Etapas de desarrollo del software mediante el modelo incremental



Fuente: Elaboración propia.

Ilustración que muestra las etapas en las cuales se desarrollaron las funcionalidades del sistema.

En el módulo de control de acceso se llevaron a cabo actividades referentes al control de acceso al sistema. Cabe mencionar que en la organización ya existía una base de datos de usuarios con sus respectivos roles, por lo que se tomó para esta investigación.

En el módulo de funciones administrativas se agruparon aquellas funcionalidades encargadas de gestionar la documentación contenida en el repositorio, tales como archivos y carpetas. Las actividades dentro de este módulo fueron:

- Creación de carpetas y archivos.
- Mostrar contenido de una carpeta.
- Permitir seleccionar archivos y carpetas para mostrar el menú de acciones.
- Opciones para mover, borrar, renombrar y cambiar permisos a carpetas y archivos.

Por último en el módulo de funciones colaborativas se agruparon aquellas que permiten al personal de la organización estar informados en todo momento de los cambios de la documentación digital de su interés, por ejemplo:

- Notificación de nuevo archivo o carpeta al personal.
- Enviar un archivo a la pantalla principal de los usuarios del sistema de expedientes.
- Permitir cargar archivos desde los detalles del expediente.

Dentro de la organización existe un procedimiento para el control de las actividades de desarrollo de software, los proyectos deben agendarse en un gestor de proyectos. La cartera de proyectos de TI es gestionada por el software de código libre dotProject (Sitio Web dotProject, 2004), el cual es un sistema web que permite agregar las diferentes tareas del proyecto, así como horas estimadas y recursos que son requeridos para cada actividad.

Por lo que la organización requirió que el proyecto de desarrollo de software fuera dado de alta en el gestor de proyectos de TI de la organización para así generar el enlace con el personal, Ver Ilustración 17. Las fases del proyecto fueron registradas también, pues son de interés para la organización, ya que le permitieron observar el avance del proyecto en cada una de sus etapas.

Ilustración 17. Proyecto agendado en el gestor DotProject de la organización.

The screenshot shows the dotProject web application interface. At the top, there's a navigation bar with 'projects list', 'edit this project', 'organize tasks', and 'reports'. Below this is a 'Sistema repositorio de información' section with 'Detalles' and 'Summary' tabs. The 'Detalles' tab shows project information like 'Company: Grupo Farías', 'Start Date: 03/09/2013', and 'Actual End Date: 09/10/2014'. The 'Summary' tab shows 'Status: In Progress', 'Priority: normal', and 'Project Hours: 1874'. Below the summary is a 'Description' field. At the bottom, there's a 'Tasks' section with a table of tasks. The table has columns for 'Task Name', 'Task Owner', 'Assigned Users', 'Start Date', 'Duration', and 'Finish Date'. The tasks are listed in a hierarchical manner, including 'Elección del framework a utilizar', 'Fase 1', 'Recibir inicio de sesión con la b.d. de ingreso', 'Recibir contraseña', 'Fase 2', 'Crear nuevo elemento', 'Mostrar contenido de elementos', 'Selección de elementos', 'Mover elementos', 'Borrar elementos', 'Cambiar nombre a elementos', 'Cambiar permisos a elementos', 'Puesta en marcha en servidor', 'Fase 3', 'Notificar via mail de nuevo elemento', 'Enviar elemento a pantalla principal', 'Carga de elementos desde el sistema de expedientes', and 'Puesta en marcha en servidor'.

Task Name	Task Owner	Assigned Users	Start Date	Duration	Finish Date
Elección del framework a utilizar para a elaboración del proyecto	dial	dial (100%)	24/02/2014 02:00 pm	36 hours	18/03/2014 06:00 pm
Fase 1	dial	dial (100%)	07/04/2014 01:45 pm	1 hours	07/04/2014 06:00 pm
Recibir inicio de sesión con la b.d. de ingreso	dial	dial (100%)	16/02/2014 01:00 pm	20 hours	14/03/2014 06:00 pm
Recibir contraseña	dial	dial (100%)	22/04/2014 03:30 pm	24 hours	25/04/2014 01:30 pm
Fase 2	dial	dial (100%)	02/05/2014 01:00 pm	69 days	04/08/2014 06:00 pm
Crear nuevo elemento (archivos v carpetas)	dial	dial (100%)	05/05/2014 01:00 pm	10 days	19/05/2014 06:00 pm
Mostrar contenido de elementos	dial	dial (100%)	19/05/2014 03:30 pm	10 days	02/06/2014 11:30 am
Selección de elementos	dial	dial (100%)	02/06/2014 09:30 am	10 days	16/06/2014 08:30 am
Mover elementos	dial	dial (100%)	16/06/2014 09:45 am	10 days	30/06/2014 08:45 am
Borrar elementos	dial	dial (100%)	30/06/2014 01:00 pm	4 days	04/07/2014 11:00 am
Cambiar nombre a elementos	dial	dial (100%)	04/07/2014 01:00 pm	8 days	15/07/2014 06:00 pm
Cambiar permisos a elementos	dial	dial (100%)	15/07/2014 03:00 pm	12 days	31/07/2014 11:00 am
Puesta en marcha en servidor	dial	dial (100%)	31/07/2014 03:00 pm	5 days	07/08/2014 11:00 am
Fase 3	dial	dial (100%)	07/08/2014 09:45 am	40 days	02/10/2014 08:45 am
Notificar via mail de nuevo elemento	dial	dial (100%)	08/08/2014 10:00 am	10 days	21/08/2014 06:00 pm
Enviar elemento a pantalla principal	dial	dial (100%)	21/08/2014 10:00 am	10 days	03/09/2014 06:00 pm
Carga de elementos desde el sistema de expedientes	dial	dial (100%)	03/09/2014 10:15 am	20 days	01/10/2014 08:15 am
Puesta en marcha en servidor	dial	dial (100%)	02/10/2014 10:15 am	5 days	09/10/2014 08:15 am

Fuente: Elaboración propia con datos de la organización.

Ilustración que muestra las actividades capturadas en el gestor de proyectos de la organización llamado dotProject.

El gestor de proyectos de TI se actualizó con cada funcionalidad realizada para obtener un avance del proyecto en general, visualizándose así el porcentaje de avance. La organización fue notificada vía correo electrónico de la finalización de una tarea mediante la función proporcionada por el gestor de proyectos.

Ahora bien, si bien se mencionó que la documentación a controlar era la referente al expediente, se propuso que se tomara como prueba piloto el adoptar el nuevo repositorio para reemplazar el de documentos ISO, ya que así el personal se fue empapando con el nuevo sistema poco a poco.

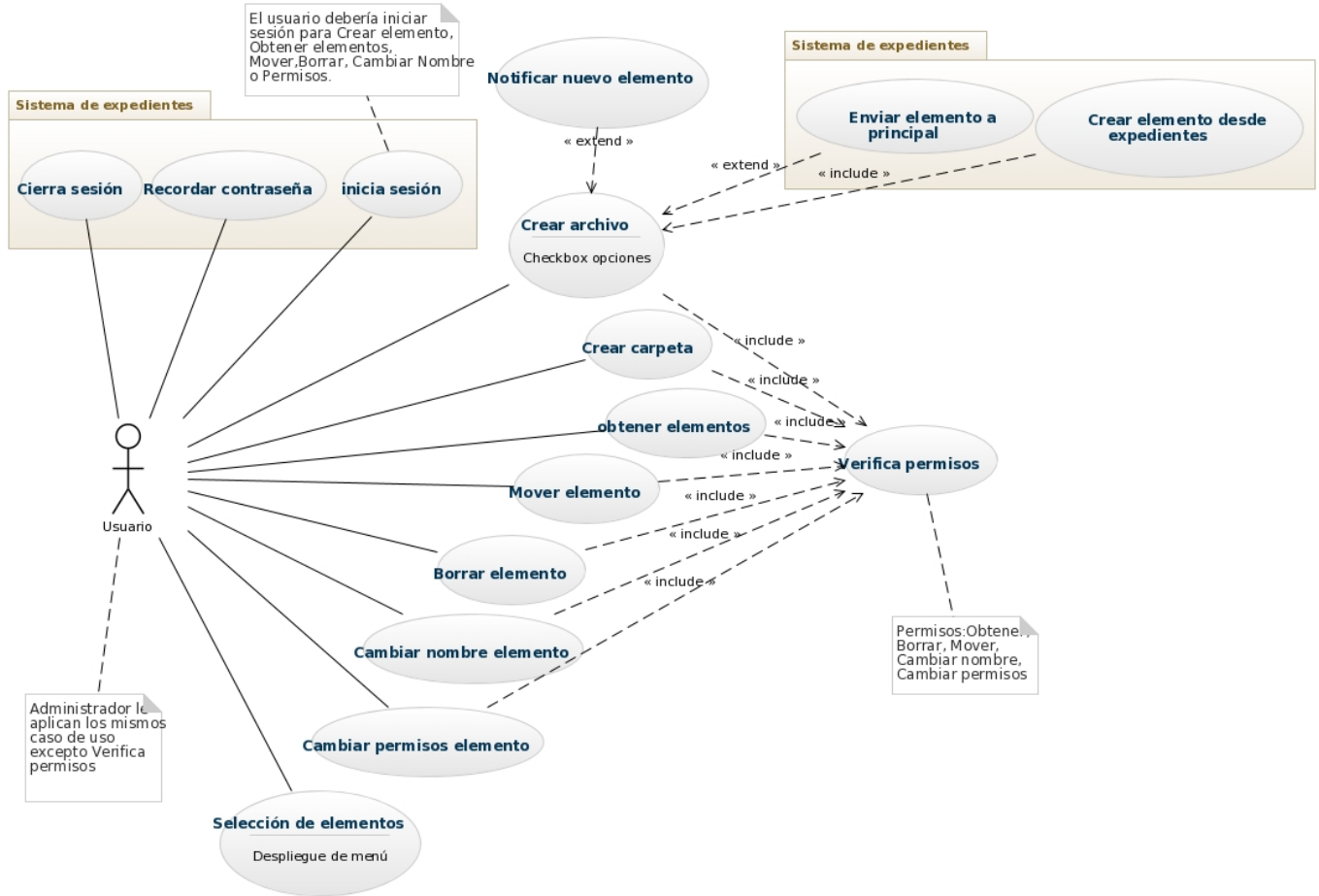
#### **4.6 Diseño de un sistema de gestión de documentos de archivo.**

Una vez obtenidos los requerimientos y como parte del cronograma del proyecto se obtuvieron aquellos modelos que permitieron detallar las funcionalidades obtenidas en la ejecución de las etapas anteriores.

##### **4.6.1 Diagrama de caso de uso general y descripción de casos de uso.**

En el diagrama de caso de uso general (Ver Ilustración 18) se muestran los requerimientos funcionales detectados durante el análisis, en él se muestran las funcionalidades desarrolladas en las tres fases.

Ilustración 18. Diagrama de caso de uso general.



Fuente: Elaboración propia.

Ilustración que muestra el caso de uso general donde se identifican las funcionalidades que se identificaron a partir de las primeras etapas de la metodología.

Las funcionalidades mostradas en grupo y tituladas *sistema de expedientes* de la Ilustración 18, se refiere a aquellas que necesitaron interactuar con bases de datos de la organización debido a usuarios y roles existentes. El usuario que sea administrador puede acceder a cualquier elemento del repositorio, cualquier otro rol debe tener permiso explícito para ver, cambiar nombre, mover, borrar según sea el caso.

Derivado del diagrama de caso de uso mostrado en la Ilustración 18, se obtuvo la siguiente tabla con un listado de los caso de uso que fueron programados para cada una de las tres fases del proyecto (Ver Tabla 6).

Tabla 6. Listado de casos de uso.

ID	MÓDULO	CASO DE USO
1	Fase 1	Inicio de sesión
2	Fase 1	Cerrar sesión
3	Fase 1	Recordar contraseña
4	Fase 2	Crear carpeta
5	Fase 2	Crear archivo
6	Fase 2	Obtener elementos
7	Fase 2	Selección de elementos
8	Fase 2	Mover elemento
9	Fase 2	Borrar elemento
10	Fase 2	Cambiar nombre de elemento
11	Fase 2	Cambiar permisos
12	Fase 3	Notificar de nuevo elemento
13	Fase 3	Enviar elemento a la página principal
14	Fase 3	Cargar elemento desde sistema de expedientes

Fuente: Elaboración propia.

Tabla que contiene las funcionalidades que se desarrollaron así como la fase en la que se llevaron a cabo.

Las fases fueron programadas de manera incremental en el gestor de proyectos de la organización, al término de la segunda fase se realizó la instalación del sistema en el servidor de producción, ya que hasta este momento las funciones de mayor peso se habían realizado y probado.

#### 4.6.2 Descripción detallada de caso de uso y prototipo.

Las siguientes tablas contienen las descripciones de cada caso de uso, las cuales muestran la precondición, flujo básico, flujo alternativo, postcondición y opcionalmente excepciones y reglas del negocio que tuvieron que considerarse para el desarrollo del software, la descripción de los casos de uso se basó en Fontela (2012).

**Tabla 7. Descripción de inicio de sesión.**

Nombre de funcionalidad:	Inicio de sesión	Número de caso de uso ID1
Precondición	El usuario debe solicitar un elemento del repositorio, Ej. <a href="http://&lt;&lt;ip servidor&gt;&gt;/repositorio/elemento/">http://&lt;&lt;ip servidor&gt;&gt;/repositorio/elemento/</a>	
Flujo básico	<ol style="list-style-type: none"> <li>1. Introduce nombre de usuario.</li> <li>2. Introduce contraseña.</li> <li>3. Se da clic en botón de iniciar sesión o presiona la tecla Entrar.</li> <li>4. Se verifican los datos.</li> <li>5. Se dirige al usuario al elemento solicitado mostrando su contenido.</li> </ol>	
Flujo alternativo	<ol style="list-style-type: none"> <li>1.1 Se alerta si la tecla Mayúsculas esta activada.</li> <li>2.1 Se alerta si la tecla Mayúsculas esta activada.</li> <li>4.1 Si el usuario o la contraseña es incorrecta se muestra un mensaje.</li> <li>5.1 Si el elemento solicitado no existe se redirige al nodo principal.</li> <li>5.2 Si no tiene permiso sobre el elemento solicitado se muestra mensaje y se redirige al nodo principal.</li> </ol>	
Postcondición	Se mostrarán los elementos del nodo principal.	
Excepción	El usuario tal vez no recuerde su inicio de sesión.	
Reglas del negocio		

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad inicio de sesión.

En la Tabla 7 se describe la funcionalidad para iniciar sesión en el sistema, ésta vino a satisfacer el requisito de seguridad para el repositorio, debido a que con esta funcionalidad es posible restringir el acceso a los archivos en el repositorio. El mecanismo implementado crea una variable de sesión para validar y el identificador del usuario, posteriormente en cada controlador se valida que exista dicha variable. El usuario teclea sus credenciales que le fueron asignados en el sistema de expedientes para poder acceder al repositorio.

**Tabla 8. Descripción cierre de sesión.**

Nombre de funcionalidad:	Cierre de sesión	Número de caso de uso ID2
Precondición	El usuario debe iniciar sesión.	
Flujo básico	<ol style="list-style-type: none"> <li>1. El usuario debe presionar el botón de cerrar sesión ubicado en la parte superior derecha.</li> <li>2. El sistema envía al usuario a la pantalla de inicio de sesión.</li> <li>3. Se muestra un mensaje que dice "ha cerrado su sesión correctamente".</li> </ol>	
Flujo alternativo	1.1 El usuario cierra la aplicación por medio del botón cerrar de la ventana.	
Postcondición	La sesión es destruida y el usuario tendrá que volver a iniciar sesión para poder obtener elementos.	
Excepción	Si el usuario había seleccionado "recordar inicio de sesión" ya no será válida y tendrá que volver a proporcionar sus credenciales.	
Reglas del negocio		

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad llamada cerrar sesión en el sistema repositorio.

En la Tabla 8 se describe la funcionalidad llamada cierre de sesión, la cual tiene como objetivo destruir las variables creadas a partir del inicio de sesión. El botón para cerrar sesión aparece si existe una sesión activa, después de que el usuario lo presiona lo redirige al inicio de sesión y muestra un mensaje de confirmación.

En la Tabla 9 se describe la funcionalidad recordar contraseña la cual permite al usuario obtener sus credenciales en caso que éste las olvide, el usuario selecciona el enlace recordar contraseña y posteriormente teclea su correo electrónico corporativo asociado a su cuenta, entonces el sistema busca en la base de datos la información y la envía al correo electrónico asociado.

**Tabla 9. Descripción recordar contraseña.**

Nombre de funcionalidad:	Recordar contraseña	Número de caso de uso ID3
Precondición	Ninguna.	
Flujo básico	<ol style="list-style-type: none"> <li>1. El usuario escribe su correo corporativo en la caja de texto.</li> <li>2. El usuario debe hacer clic sobre el botón recordar contraseña.</li> <li>3. El sistema envía el usuario y contraseña al correo registrado en la cuenta.</li> </ol>	
Flujo alternativo	<ol style="list-style-type: none"> <li>2.1 Se escribe un correo en formato no válido y el sistema muestra mensaje.</li> <li>3.1 El sistema no encuentra el correo proporcionado y muestra mensaje.</li> </ol>	
Postcondición	El sistema muestra un mensaje con el resultado de la operación.	
Excepción	El usuario recibe su usuario y contraseña al correo electrónico registrado.	
Reglas del negocio	Si hay dos usuarios con el mismo correo.	

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad recordar contraseña.

La funcionalidad para crear una carpeta dentro del repositorio está descrita en la Tabla 10. Esta funcionalidad permite crear varias secciones para organizar los diferentes archivos, así, es posible tener en forma estructurada la información para permitir una recuperación agilizada. Además, esta funcionalidad resuelve la problemática de un archivo por expediente, ahora se crea una carpeta que identifica al expediente y ahí se colocan los archivos seccionados.

**Tabla 10. Descripción crear carpeta.**

Nombre de funcionalidad:	Crear carpeta	Número de caso de uso ID4
Precondición	El usuario debe iniciar sesión.	
Flujo básico	<ol style="list-style-type: none"> <li>1. El usuario hace clic sobre el botón crear carpeta.</li> <li>2. El usuario debe teclear el nombre de la carpeta.</li> <li>3. El sistema verifica que el nombre de la carpeta sea válido.</li> <li>4. El usuario hace clic sobre el botón Aceptar.</li> <li>5. El sistema obtiene la ubicación actual (elemento padre).</li> <li>6. El sistema guarda el elemento en la base de datos con el nombre dado por el usuario.</li> <li>7. El sistema avisa al usuario con un mensaje de la finalización del proceso.</li> </ol>	
Flujo alternativo	<ol style="list-style-type: none"> <li>3.1 El usuario teclea un nombre no válido.</li> <li>3.2 El sistema avisa al usuario con un mensaje de que no es nombre válido.</li> <li>5.1 El usuario no tiene permiso de crear elementos en la ubicación actual.</li> <li>5.2 El sistema avisa que no tiene permiso.</li> </ol>	
Postcondición	Se obtienen elementos de la ubicación actual.	
Excepción	Solo se permite crear una carpeta si el usuario tiene permiso sobre la ubicación actual.	
Reglas del negocio		

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad para crear carpetas dentro del sistema.

La funcionalidad crear archivo (Tabla 11) describe la función que permite adjuntar archivos al repositorio mediante un formulario Web. El usuario debe tener permisos sobre la posición actual para poder realizar esta acción. La funcionalidad no limita el número de archivos que se pueden seleccionar pero si el tipo de archivo.

**Tabla 11. Descripción crear archivo.**

Nombre de funcionalidad:	Crear archivo	Número de caso de uso ID5
Precondición	El usuario debe iniciar sesión.	
Flujo básico	<ol style="list-style-type: none"> <li>1. El usuario hace clic sobre el botón crear archivo.</li> <li>2. El sistema muestra una ventana que permite seleccionar un archivo de la computadora del usuario, en dicha ventana hay casillas para notificar nuevo elemento vía correo y para enviar a página principal.</li> <li>3. El sistema muestra la cantidad en Kb a cargar en el sistema.</li> <li>4. El usuario debe presionar el enlace Cargar para iniciar,</li> <li>5. El sistema muestra el progreso de avance y tiempo estimado de finalización.</li> <li>6. El sistema guarda en la base de datos los elementos con el nombre dado por el archivo.</li> <li>7. El sistema envía mensaje de resultado.</li> </ol>	
Flujo alternativo	<ol style="list-style-type: none"> <li>2.1 El usuario cancela la operación haciendo clic en Cancelar..</li> <li>2.2 El sistema cierra la ventana y muestra los elementos de la ubicación actual.</li> <li>3.1 El usuario puede seleccionar otro archivo para cargar.</li> <li>3.2 El sistema actualiza la cantidad en Kb a cargar.</li> </ol>	
Postcondición	Se obtienen elementos de la ubicación actual.	
Excepción	Solo se permite crear archivo si el usuario tiene permiso sobre la ubicación actual.	
Reglas del negocio		

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad crear archivo, la cual permitirá agregar más de un archivo en el sistema.

**Tabla 12. Descripción obtener elementos.**

Nombre de funcionalidad:	Obtener elementos	Número de caso de uso ID6
Precondición	El usuario debe iniciar sesión.	
Flujo básico	<ol style="list-style-type: none"> <li>1. El usuario hace clic sobre una carpeta.</li> <li>2. El sistema lee de la base de datos los elementos asociados.</li> <li>3. El sistema devuelve una lista de los elementos.</li> <li>4. El usuario puede ver la lista de elementos en pantalla.</li> </ol>	
Flujo alternativo	<ol style="list-style-type: none"> <li>1.1 El usuario hace clic sobre un archivo.</li> <li>2.1 El sistema obliga la descarga del archivo y le pregunta al usuario donde desea guardarlo, si es un archivo que se puede desplegar en el navegador el sistema muestra una ventana donde se muestra el contenido del archivo.</li> <li>2.2 El usuario guarda el archivo con el gestor de descargas del navegador.</li> </ol>	
Postcondición	Se muestra la lista de elementos de la ubicación deseada.	
Excepción	El usuario debe tener permiso de obtener elementos sobre la ubicación.	
Reglas del negocio		

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad obtener elementos, mediante esta función los usuarios pueden desplazarse sobre las carpetas y leer archivos.

La funcionalidad obtener elementos (Tabla 12) permite navegar entre los elementos de tipo carpeta de forma jerárquica, cada ocasión que el usuario hace clic sobre una carpeta, el repositorio obtiene los elementos asociados y los despliega mediante iconos gráficos. Si el usuario hace clic sobre un elemento de tipo archivo, entonces se verifica el tipo de archivo, si es un archivo que el navegador puede desplegar, por ejemplo, aquellos en los que existen complementos para ser visualizados a través del navegador web como *PDF* o *SWF*, el sistema abre una ventana donde se muestra el contenido, en caso contrario el sistema fuerza la descarga del archivo.

En la Tabla 13 se describe la selección de elementos, esta funcionalidad permite mostrar el menú de acciones posibles después que el usuario hace clic sobre la casilla de selección de un archivo o carpeta, en este sentido, se muestran las acciones: cambiar nombre, mover, cambiar permisos y borrar si se selecciona uno y mover, cambiar permisos y borrar si se seleccionan varios. Si posteriormente el usuario

deselecciona los elementos, el menú vuelve a su estado original y muestra solo subir archivo y crear carpeta.

**Tabla 13. Descripción selección de elementos.**

Nombre de funcionalidad:	Selección de elementos	Número de caso de uso ID7
Precondición	El usuario debe iniciar sesión. El usuario debe haber obtenido elementos de una carpeta.	
Flujo básico	1. El hace clic en la casilla de selección de un elemento. 2. El sistema muestra un menú de opciones de: Cambiar de nombre, Mover, Borrar, Cambiar permisos. 3. El usuario deselectiona el elemento seleccionado. 4. El sistema oculta el menú.	
Flujo alternativo	1.1 El usuario hace clic sobre la casilla en más de un elemento. 2.1 El sistema muestra un menú con opciones de: Mover, Borrar, Cambiar permisos. 3.1 El usuario deselectiona los elementos.	
Postcondición	Se muestran opciones de Cambiar de nombre, Mover, Borrar, Cambiar permisos según corresponda.	
Excepción		
Reglas del negocio		

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad selección de elementos, mediante esta funcionalidad el usuario podrá activar el menú de acción para realizar diferentes tareas administrativas.

**Tabla 14. Descripción mover elementos.**

Nombre de funcionalidad:	Mover elemento	Número de caso de uso ID8
Precondición	El usuario debe iniciar sesión. El usuario debe tener seleccionado al menos un elemento.	
Flujo básico	1. El usuario hace clic sobre el botón Mover. 2. El sistema muestra en una ventana una lista de ubicaciones de destino posibles. 3. El usuario selecciona una ubicación de la lista. 4. El usuario hace clic sobre el botón aceptar. 5. El sistema actualiza la base de datos. 6. El sistema cierra la ventana y Obtiene elementos de la ubicación actual.	
Flujo alternativo	3.1 El usuario hace clic sobre Cancelar. 3.2 El sistema cierra la ventana y obtiene elementos de la ubicación actual.	
Postcondición	El sistema obtiene elementos de la ubicación actual.	
Excepción	4.1 El usuario no tiene permiso sobre la ubicación destino. 4.2 El sistema muestra mensaje de negación de permiso. 4.3 El sistema obtiene elementos de la ubicación actual.	
Reglas del negocio	El usuario debe tener permiso sobre la ubicación destino. El usuario debe tener permiso Mover sobre los elemento(s) seleccionado(s)	

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad mover elemento, la cual permite cambiar de ubicación un archivo o carpeta.

La funcionalidad mover elementos (Tabla 14) permite reubicar un elemento archivo o carpeta hacia otra ubicación, para ello se tienen que seleccionar los elementos a mover y en el menú de acciones hacer clic sobre el botón mover, en seguida el sistema muestra los elementos carpeta que el usuario puede seleccionar como destino. La funcionalidad para borrar elemento de la Tabla 15, permite al usuario eliminar archivos y carpetas del repositorio, si éste cuenta con el permiso sobre los elementos.

**Tabla 15. Descripción borrar elementos.**

Nombre de funcionalidad:	Borrar elemento	Número de caso de uso ID9
Precondición	El usuario debe iniciar sesión.	
	El usuario debe tener seleccionado al menos un elemento.	
Flujo básico	<ol style="list-style-type: none"> <li>1. El usuario hace clic sobre Borrar.</li> <li>2. El sistema muestra una ventana que pide confirmación.</li> <li>3. El usuario confirma haciendo clic sobre Aceptar.</li> <li>4. El sistema actualiza la base de datos.</li> <li>5. El sistema envía mensaje de resultado.</li> <li>6. El sistema obtiene elementos de la ubicación actual.</li> </ol>	
Flujo alternativo	<ol style="list-style-type: none"> <li>3.1 El sistema verifica que el usuario no tiene permiso sobre un elemento seleccionado.</li> <li>3.2 El sistema genera mensaje de aviso de negación de permiso.</li> </ol>	
Postcondición	El sistema obtiene elementos de la ubicación actual.	
Excepción	El usuario debe tener permiso Borrar sobre los elementos.	
Reglas del negocio	Los archivos no deben ser borrados físicamente del disco duro.	

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad borrar elemento, la cual permite eliminar un elemento del sistema.

**Tabla 16. Descripción cambiar nombre.**

Nombre de funcionalidad:	Cambiar nombre	Número de caso de uso ID10
Precondición	El usuario debe iniciar sesión.	
	El usuario debe tener seleccionado solo un elemento.	
Flujo básico	<ol style="list-style-type: none"> <li>1. El usuario hace sobre el botón Cambiar nombre.</li> <li>2. El sistema muestra ventana donde el usuario debe escribir el nuevo nombre.</li> <li>3. El usuario escribe el nombre nuevo para el elemento.</li> <li>4. El sistema verifica que sea nombre válido.</li> <li>5. El usuario hace clic sobre aceptar.</li> <li>6. El sistema actualiza la base de datos y envía mensaje de resultado.</li> <li>7. El sistema obtiene elementos de la ubicación actual.</li> </ol>	
Flujo alternativo	<ol style="list-style-type: none"> <li>4.1 El nombre no es válido el sistema muestra un mensaje para que el usuario lo corrija.</li> <li>5.1 El sistema verifica que el usuario no tiene permisos sobre el elemento.</li> <li>5.2 El sistema envía mensaje de negación de permisos.</li> </ol>	
Postcondición	El sistema obtiene elementos de la ubicación actual.	
Excepción	El usuario debe tener el permiso cambiar nombre sobre el elemento seleccionado.	
Reglas del negocio		

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad cambiar nombre, la cual permite modificar el nombre a un elemento, siendo éste el atributo más importante como mecanismo de identificación.

La funcionalidad cambiar nombre (Tabla 16) permite al usuario cambiar este atributo de un elemento archivo o carpeta, ya que es posible que al momento de crear el archivo, éste tenga un nombre poco representativo para el contenido o simplemente por alguna omisión.

Cambiar permisos (Tabla 17) permite modificar los permisos válidos para los elementos que el usuario tenga seleccionados al momento, para que esta opción proceda, el usuario debe tener el permiso sobre los elementos que se vayan a modificar.

**Tabla 17. Descripción cambiar permisos.**

Nombre de funcionalidad:	Cambiar permisos	Número de caso de uso ID11
Precondición	El usuario debe iniciar sesión. El usuario debe seleccionar al menos un elemento.	
Flujo básico	1. El usuario hace clic sobre cambiar permisos. 2. El sistema muestra ventana donde se muestran los usuarios y permisos seleccionables. 3. El usuario selecciona un usuario y los permisos Ver (Obtener elementos). 4. El usuario da clic en el botón Aceptar para aplicar cambios. 5. El sistema actualiza permisos en la base de datos. 6. El sistema muestra mensaje de resultado.	
Flujo alternativo	4.1 El sistema verifica que el usuario no tiene permiso para cambiar permisos sobre el elemento. 4.2 El sistema genera mensaje de denegación de permiso.	
Postcondición	El sistema obtiene elementos de la ubicación actual.	
Excepción	El usuario debe tener permiso de cambiar permiso sobre los elementos.	
Reglas del negocio	El usuario Administrador puede cambiar permisos sobre cualquier elemento.	

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad para cambiar permisos a los elementos, mediante esta función se pueden establecer el nivel de acceso para cada usuario o grupos de usuarios.

La funcionalidad de la Tabla 18 fue parte de la fase tres del desarrollo las cuales son las funcionalidades para lograr un entorno de colaborativo, ésta tiene como objetivo informar a los usuarios cuando nuevos elementos son agregados a las carpetas por medio de un historial y enviar a los usuarios un aviso mediante correo electrónico si el usuario marca la casilla para ello, con esto se pretende que las personas estén informadas acerca de si se ha proporcionado nueva información a un expediente o para estar enterados de los archivos creados y modificados al estar navegando entre las carpetas.

**Tabla 18. Descripción notificar nuevo elemento.**

Nombre de funcionalidad:	Notificar nuevo elemento	Número de caso de uso ID12
Precondición	El usuario debe iniciar sesión. El usuario debe haber creado un archivo.	
Flujo básico	1. El usuario hace clic sobre el botón crear archivo. 2. El sistema muestra una ventana que permite seleccionar un archivo de la computadora del usuario, en dicha ventana hay casillas para notificar nuevo elemento vía correo y para enviar a página principal. 3. El sistema muestra la cantidad en Kb a cargar en el sistema. 4. El usuario debe presionar el enlace Cargar para iniciar, 5. El sistema muestra el progreso de avance y tiempo estimado de finalización. 6. El sistema guarda en la base de datos los elementos con el nombre dado por el archivo. 7. El sistema envía mensaje de resultado. 8. El sistema muestra del lado derecho un resumen de los cambios efectuados.	
Flujo alternativo	2.1 El usuario marca la casilla notificar nuevo elemento. 6.1 El sistema envía vía correo electrónico la notificación de nuevo elemento a usuarios con permiso de Ver.	
Postcondición	El sistema obtiene elementos de la ubicación actual.	
Excepción	Si el envío de correo marca error se muestra mensaje.	
Reglas del negocio		

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad notificar nuevo elemento, la cual envía un aviso vía correo electrónico cuando un archivo es agregado.

**Tabla 19. Descripción enviar elemento a página principal.**

Nombre de funcionalidad:	Enviar elemento a página principal	Número de caso de uso ID13
Precondición	El usuario debe iniciar sesión. El usuario debe crear un archivo.	
Flujo básico	1. El usuario hace clic sobre el botón crear archivo. 2. El sistema muestra una ventana que permite seleccionar un archivo de la computadora del usuario, en dicha ventana hay casillas para notificar nuevo elemento vía correo y para enviar a página principal. 3. El usuario selecciona un archivo y marca casilla enviar a principal. 4. El sistema muestra la cantidad en Kb a cargar en el sistema. 5. El usuario debe presionar el enlace Cargar para iniciar, 6. El sistema muestra el progreso de avance y tiempo estimado de finalización. 7. El sistema guarda en la base de datos los elementos con el nombre dado por el archivo. 8. El sistema envía mensaje de resultado. 9. El usuario accede a la página principal del sistema de expedientes. 10. El usuario ve en la página principal un enlace al archivo recién creado en el repositorio. 11. El usuario hace clic sobre el enlace. 12. El sistema actualiza la base de datos para no mostrar más en la página principal el elemento recién creado.	
Flujo alternativo	10.1 El usuario navega a otra sección del sistema de expediente sin hacer clic sobre elemento recién creado. 10.2 El usuario regresa a la página principal del sistema de expedientes. 10.3 El elemento enviado a principal sigue apareciendo.	
Postcondición	El sistema obtiene elementos de la ubicación actual.	
Excepción		
Reglas del negocio	Si son varios archivos los creados en el repositorio en el sistema de expedientes mostrar agrupados.	

Fuente: Elaboración propia.

Tabla que muestra la descripción del caso de uso para la funcionalidad enviar elemento a principal, la cual permite agregar un enlace directo hacia un archivo recién creado.

La funcionalidad para enviar un elemento a principal (Tabla 19), coloca un enlace en la página principal del sistema de expedientes, esto con el objetivo de que los usuarios puedan crear áreas de trabajo compartido y estén notificados al momento de los cambios efectuados en la documentación del expediente. La funcionalidad es una estrategia de inclusión al sistema ya que la página principal es el punto inicial de acceso a los sistemas de la organización.

**Tabla 20. Descripción cargar elementos desde expedientes.**

Nombre de funcionalidad:	Cargar elemento desde expedientes	Número de caso de uso ID14
Precondición	El usuario debe iniciar sesión.	
Flujo básico	El usuario debe acceder a los detalles del expediente. 1. El usuario hace clic sobre icono proporcionar archivo. 2. El sistema muestra una ventana que permite seleccionar un archivo de la computadora del usuario, en dicha ventana hay casillas para notificar nuevo elemento vía correo y para enviar a página principal. 3. El sistema muestra la cantidad en Kb a cargar en el sistema. 4. El usuario debe presionar el enlace Cargar para iniciar, 5. El sistema muestra el progreso de avance y tiempo estimado de finalización. 6. El sistema guarda en la base de datos los elementos con el nombre dado por el archivo. 7. El sistema envía mensaje de resultado.	
Flujo alternativo		
Postcondición		
Excepción		
Reglas del negocio	Los archivos se cargarán a la carpeta identificada con la RUE del expediente, si no hay una carpeta el sistema creará una.	

Fuente: Elaboración propia.

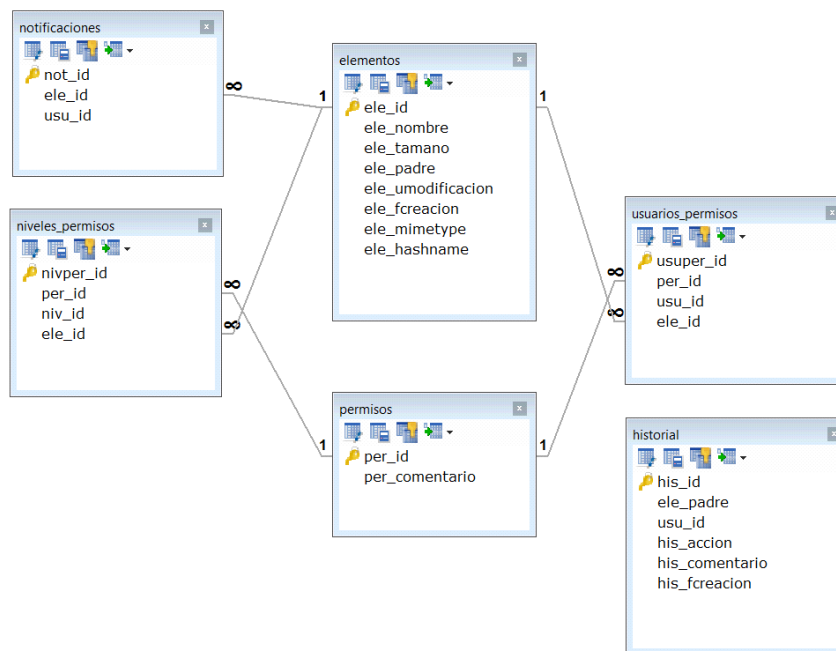
Tabla que muestra la descripción del caso de uso para la funcionalidad cargar elemento desde expedientes, la cual permite agregar archivos al repositorio desde el sistema de expedientes.

Como parte de la estrategia para solventar el problema de los documentos digitales del expediente mencionado en la sección 4.4, se propuso la funcionalidad descrita en la Tabla 20, con esto, el personal puede ver y proporcionar archivos desde el sistema de expedientes, por lo que ahora se pueden tener varios archivos seccionados conforme a las necesidades de la organización.

#### 4.6.3 Diseño de la base de datos propuesto.

Las funcionalidades definidas permitieron dar forma al resguardo físico de datos a empleado por el sistema, por ello, en esta etapa se llevó a cabo el diseño de la base de datos de acuerdo a las necesidades de las funcionalidades, el DER se puede observar en la Ilustración 19.

Ilustración 19. Diagrama Entidad Relación de la base de datos.



Fuente: Elaboración propia.

Ilustración que muestra el diseño de la base de datos del sistema repositorio, el cual se conforma de 6 tablas.

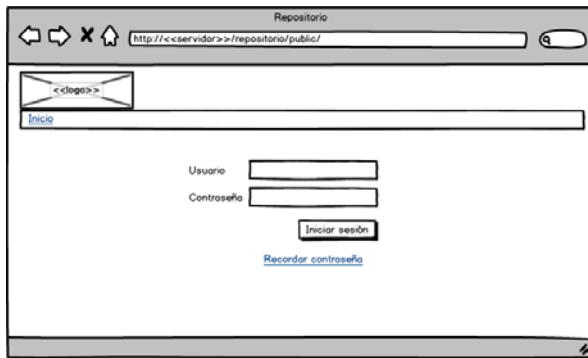
Como se observa en la Ilustración 19, la tabla elementos es la encargada de resguardar los registros de los elementos del sistema, a esta tabla se le relacionan los permisos efectivos, que pueden aplicarse por usuario o por grupos. Cabe mencionar que en este modelo no se muestran entidades utilizadas por

algunas funcionalidades del sistema, por ejemplo, el inicio de sesión, ya que éstas se encuentran en bases de datos independientes que ya se encontraban en la organización.

#### 4.6.4 Prototipo de las funcionalidades.

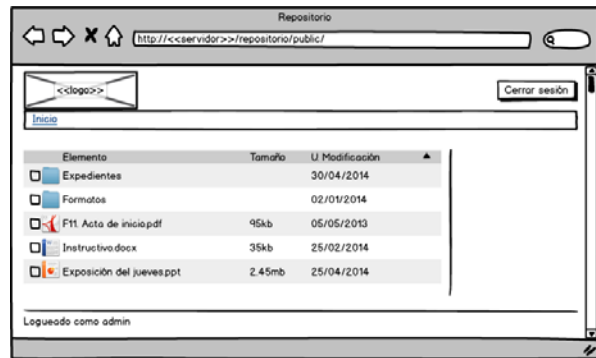
En esta etapa se inició el desarrollo del *Frontend* del proyecto, para ello se realizaron prototipos de las funcionalidades las cuales permitieron detallar la posición del contenido en la pantalla del usuario, así como obtener la opinión de la organización.

Ilustración 20. Prototipo inicio de sesión.



Fuente: Elaboración propia.

Ilustración 21. Prototipo cierre de sesión.



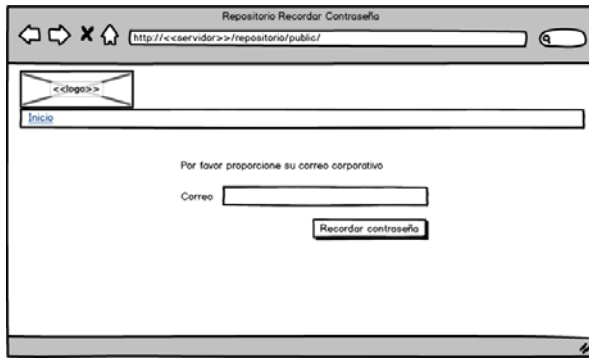
Fuente: Elaboración propia.

En la Ilustración 20 se muestra el prototipo de la funcionalidad para iniciar sesión.

En la Ilustración 21 se muestra el prototipo de la funcionalidad para cerrar sesión en el sistema.

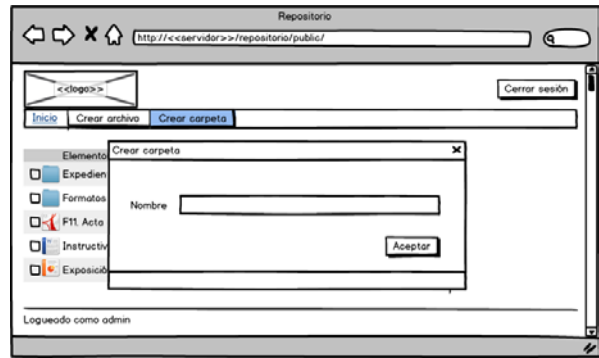
En la Ilustración 20 se muestra el prototipo de la funcionalidad inicio de sesión, se muestran dos campos, uno para el nombre de usuario y otro para la contraseña, el usuario debe presionar el botón iniciar sesión. En la parte inferior del botón se muestra un enlace para acceder a la funcionalidad recordar contraseña. En la Ilustración 21 se muestra como fue el prototipo de la funcionalidad cierre de sesión, la cual se representó mediante un botón en la parte superior derecha en todas las vistas del sistema.

Ilustración 22. Prototipo recordar contraseña.



Fuente: Elaboración propia.

Ilustración 23. Prototipo crear carpeta.



Fuente: Elaboración propia.

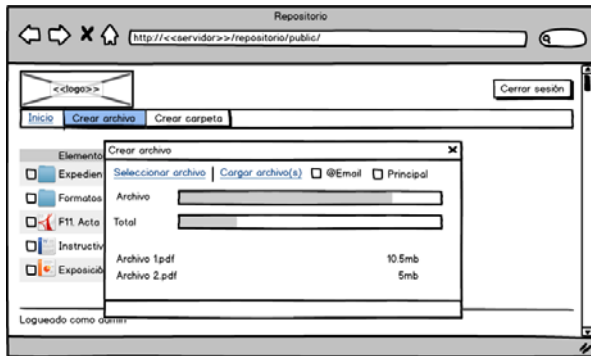
En la Ilustración 22 se muestra el prototipo de la funcionalidad para recordar contraseña.

En la Ilustración 23 muestra el prototipo de la funcionalidad para crear una carpeta.

En la Ilustración 22 se muestra el prototipo para recordar las credenciales del usuario, dichas credenciales son administradas desde el sistema de expedientes. El usuario tiene que teclear su correo electrónico corporativo y presionar el botón “Recordar Contraseña”, el mensaje de resultado se implementó para ser mostrado en la parte inferior.

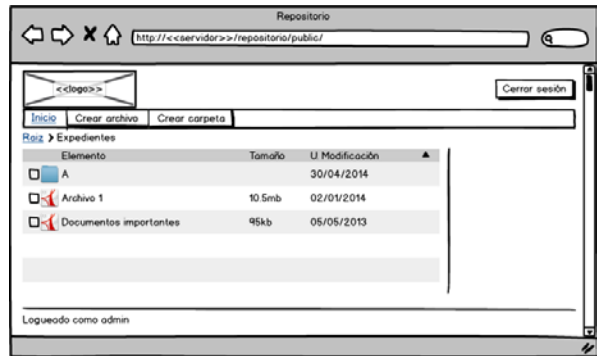
En la Ilustración 23 se muestra la funcionalidad correspondiente a la fase 2 del desarrollo, las carpetas permiten estructurar la información de forma jerárquica. Para crear una carpeta el usuario debe desplazarse a la ubicación deseada por medio de enlaces que invocan la funcionalidad obtener elementos de la Ilustración 25, entonces el usuario debe presionar el botón “Crear Carpeta” del menú de acción ubicado en la parte superior.

Ilustración 24. Prototipo crear archivo.



Fuente: Elaboración propia.

Ilustración 25. Prototipo obtener elementos.



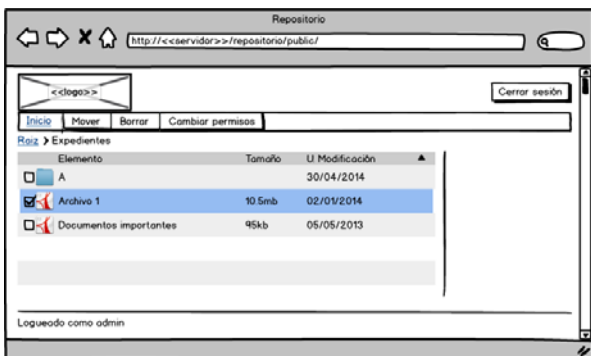
Fuente: Elaboración propia.

En la Ilustración 24 se muestra el prototipo de la funcionalidad para crear un archivo.

En la Ilustración 25 se muestra el prototipo de la funcionalidad para obtener elementos.

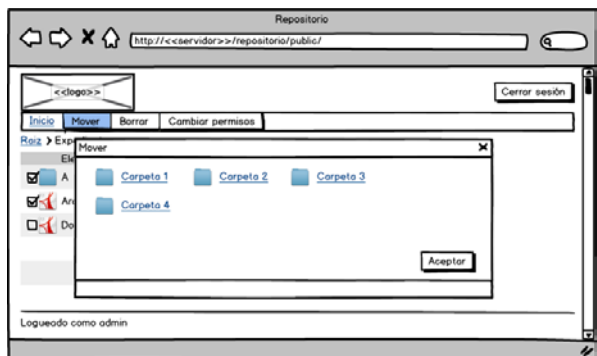
En la Ilustración 24 se muestra la funcionalidad para crear un archivo, ésta despliega una ventana dentro del navegador que muestra el formulario para agregar los archivos que se desean transferir, en esa misma ventana el usuario tiene que presionar el botón llamado “Subir” para iniciar la operación.

Ilustración 26. Prototipo selección de elementos.



Fuente: Elaboración propia.

Ilustración 27. Prototipo mover elementos.



Fuente: Elaboración propia.

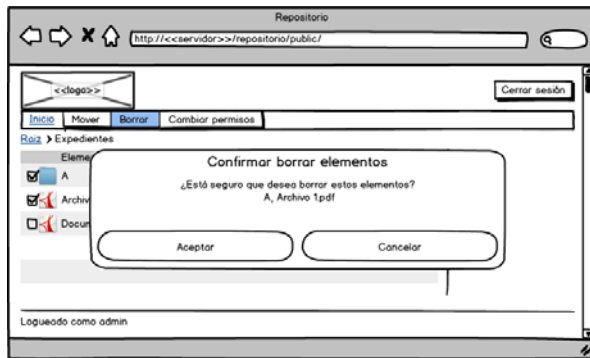
En la Ilustración 26 se muestra el prototipo de la funcionalidad para seleccionar un elemento.

En la Ilustración 27 se muestra el prototipo de la funcionalidad para mover un elemento.

En la Ilustración 26 se muestra la funcionalidad llamada selección de elementos, se subraya el elemento con color azul al momento de activar la casilla de selección, así mismo el menú de acción en la parte superior cambia con las posibles opciones a realizar.

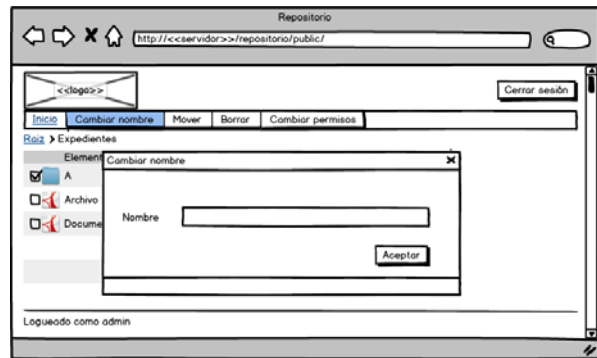
En la Ilustración 27 se muestra la funcionalidad para mover elementos, el usuario debe seleccionar los elementos a mover y presionar “Mover” del menú de acción para invocar una ventana con los destinos posibles, al seleccionar un destino debe hacer clic en “Aceptar”.

Ilustración 28. Prototipo borrar elemento.



Fuente: Elaboración propia.

Ilustración 29. Prototipo cambiar nombre a elemento.



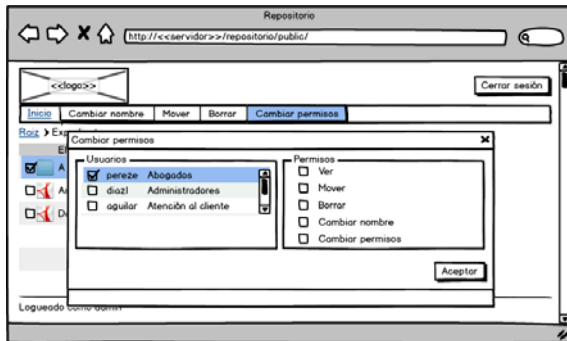
Fuente: Elaboración propia.

En la Ilustración 28 se muestra el prototipo de la funcionalidad para borrar un elemento.

En la Ilustración 29 se muestra el prototipo de la funcionalidad para cambiar nombre a un elemento.

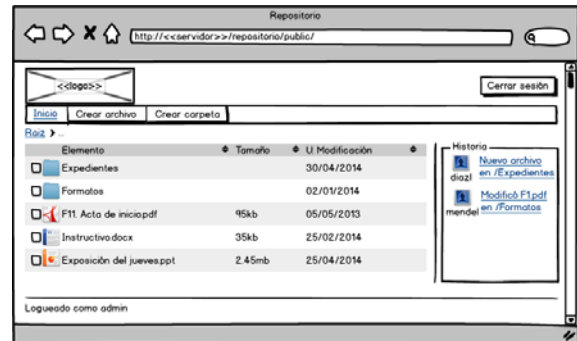
Para borrar un elemento se tiene que hacer la selección de los elementos mediante la funcionalidad de la ilustración Ilustración 28, posteriormente el usuario debe presionar el botón “Borrar” para ejecutar la acción. Para renombrar un elemento el requisito es que solo uno esté seleccionado, acto seguido se debe presionar “Renombrar” como en la Ilustración 29.

Ilustración 30. Prototipo cambiar permisos.



Fuente: Elaboración propia.

Ilustración 31. Prototipo notificar nuevo elemento.



Fuente: Elaboración propia.

En la Ilustración 30 se muestra el prototipo de la funcionalidad para borrar un elemento.

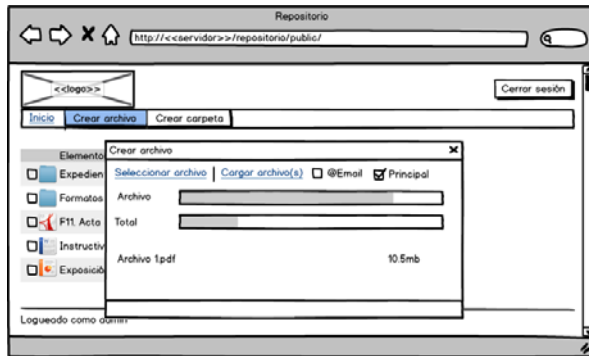
En la Ilustración 31 se muestra el prototipo de la funcionalidad para borrar un elemento.

Cabe recordar que para poder acceder a un elemento de tipo carpeta o archivo, el usuario debe contar con los permisos para hacerlo, la funcionalidad de la Ilustración 30 permite gestionar los permisos para los elementos, para realizar cambios el usuario debe tener el permiso para hacerlo. A través de un mensaje al momento de presionar el botón "Permisos" le notificará al usuario si se aplicaron los permisos o no.

La Ilustración 31, Ilustración 32 e Ilustración 33 son las funcionalidades de la tercera fase del desarrollo, para notificar nuevo elemento y enviar elemento a principal, el usuario debe acceder a la funcionalidad crear archivo y activar las casillas para invocar estas funciones.

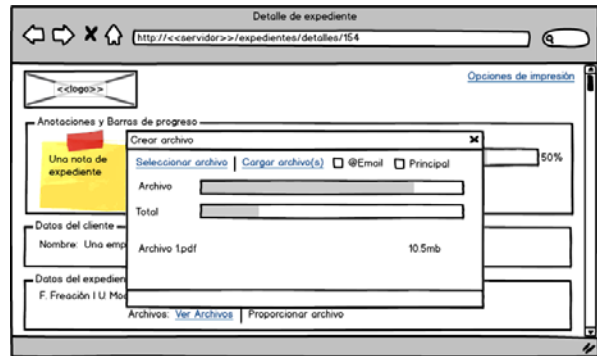
Por último la funcionalidad cargar elementos desde expedientes se ideó para permitir agregar archivos desde los detalles del sistema de expedientes para facilitar la integración con los procesos actuales.

Ilustración 32. Prototipo enviar elemento a principal.



Fuente: Elaboración propia.

Ilustración 33. Prototipo elementos desde expedientes.



Fuente: Elaboración propia.

En Ilustración 32 se muestra el prototipo de la funcionalidad para enviar elemento a principal.

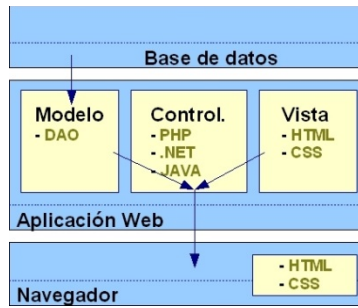
En la Ilustración 33 se muestra el prototipo de la funcionalidad para agregar archivos desde el sistema de expedientes.

#### 4.6.5 Diagrama general de arquitectura del software.

Pressman (2010) cita que las aplicaciones web deben construirse con el empleo de capas en las que se tomen en cuenta distintas preocupaciones; en particular, deben separarse los datos de la aplicación de los contenidos de ésta (nodos de navegación), y éstos, a su vez, deben separarse con toda claridad del aspecto y la sensación de la interfaz (páginas). Menciona que los autores sugieren realizar esta separación en tres capas, desacoplando la interfaz de la navegación y del comportamiento de la aplicación.

Siguiendo la recomendación anterior, el sistema fue desarrollado usando la arquitectura Modelo Vista Controlador (MVC), esto permite tener un orden en el código fuente separando la lógica de negocio y la interfaz. El diseño de la arquitectura se observa en Ilustración 34.

**Ilustración 34. Arquitectura MVC utilizada para el desarrollo del sistema.**

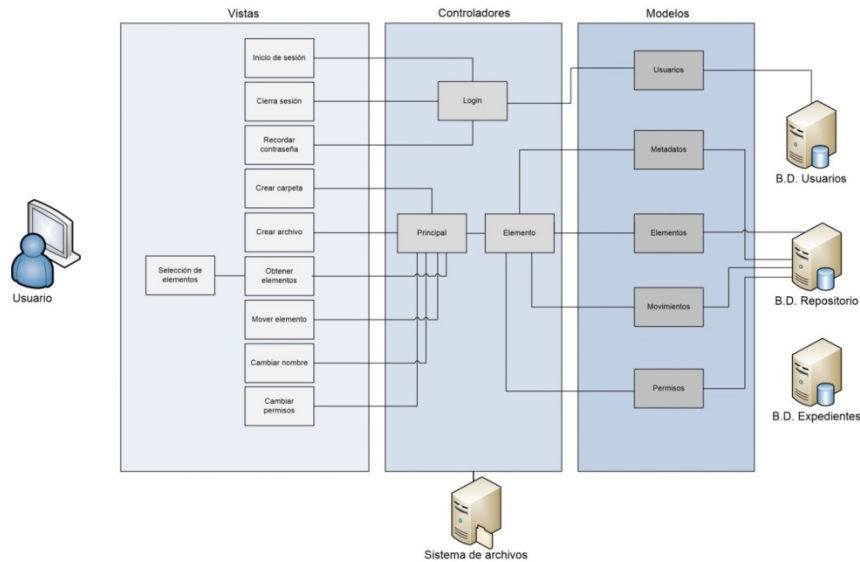


Fuente: San Martín (2007).

Ilustración que detalla la interacción de los componentes bajo una arquitectura Modelo Vista Controlador para un sistema Web, misma que fue usada en el sistema repositorio.

En la Ilustración 35 se muestra la relación entre los componentes del sistema bajo la arquitectura MVC, la cual fue seleccionada como modelo para el desarrollo del software. El sistema tuvo que hacer uso de dos bases de datos existentes, la base de datos de usuarios y la del sistema de expedientes. Además de los recursos de base de datos, el sistema hace uso del sistema de archivos del servidor, pues es donde se resguardan los archivos que son proporcionados mediante el controlador Elemento.

**Ilustración 35. Componentes del sistema en la arquitectura MVC.**



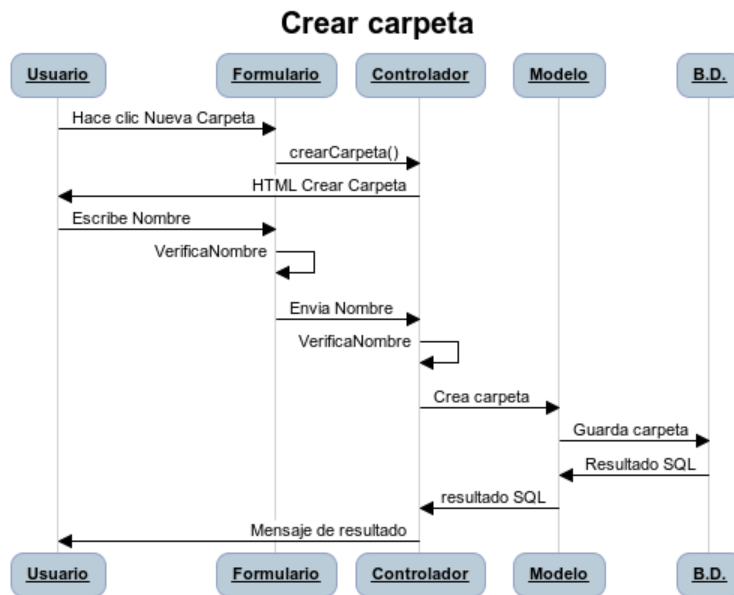
Fuente: Elaboración propia.

Ilustración que muestra la interacción de componentes del sistema repositorio mediante una arquitectura MVC.

#### 4.6.6 Diagramas de secuencia.

En esta sección se integran los diagramas de secuencia UML de aquellas funcionalidades más relevantes o que necesitaron de un análisis detallado. Los diagramas fueron creados con una herramienta gratuita basada en la Web llamada *websequence diagrams* (Sitio Web Web Sequence Diagrams, 2014).

Ilustración 36. Diagrama de secuencia crear carpeta.

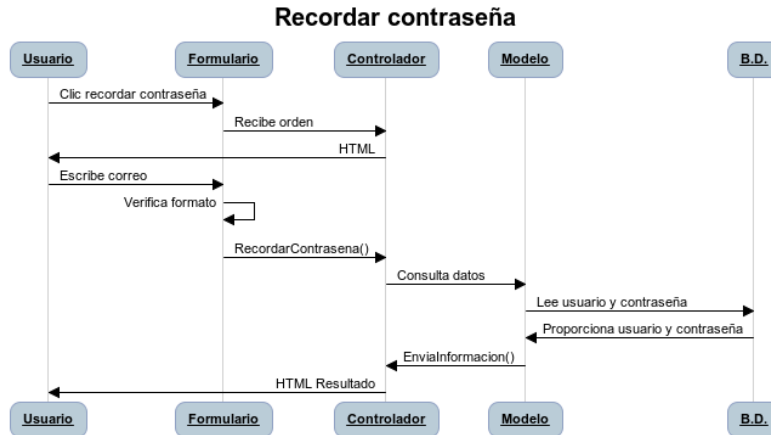


Fuente: Elaboración propia.

Ilustración que muestra el diagrama de secuencia de la funcionalidad crear carpeta.

En la Ilustración 36, la funcionalidad crear carpeta accede a la base de datos para guardar el nombre del elemento deseado y el nodo padre que lo contiene, sin embargo antes de esta acción, se verifican los parámetros para limpiarlos de posibles ataques de inyección de código SQL y de HTML.

Ilustración 37. Diagrama de secuencia recordar contraseña.

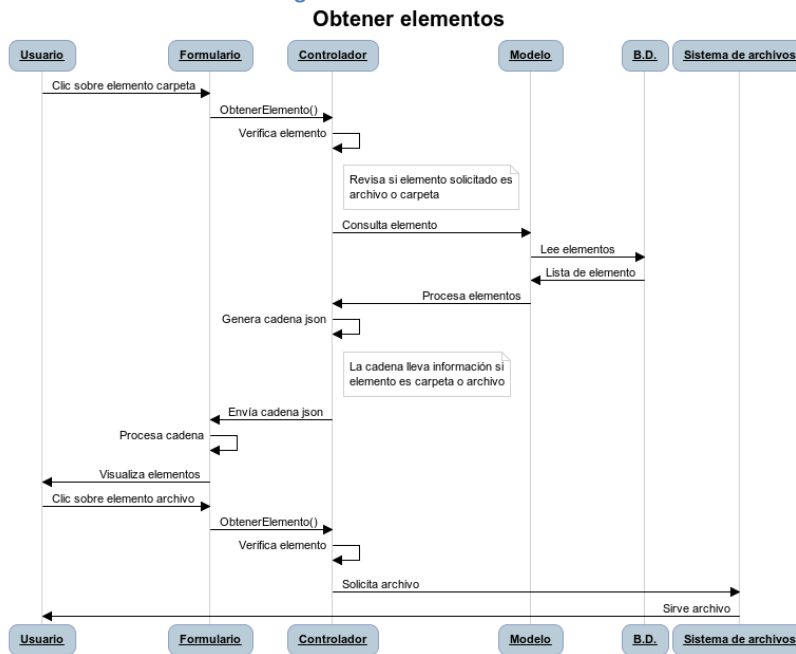


Fuente: Elaboración propia.

Ilustración que muestra el diagrama de secuencia de la funcionalidad recordar contraseña.

En la Ilustración 37 la funcionalidad recordar contraseña obtiene de la base de datos de expediente las credenciales del usuario, esto realizando una búsqueda con el correo electrónico que se proporcione, al final muestra un mensaje de resultado.

Ilustración 38. Diagrama de secuencia obtener elementos.



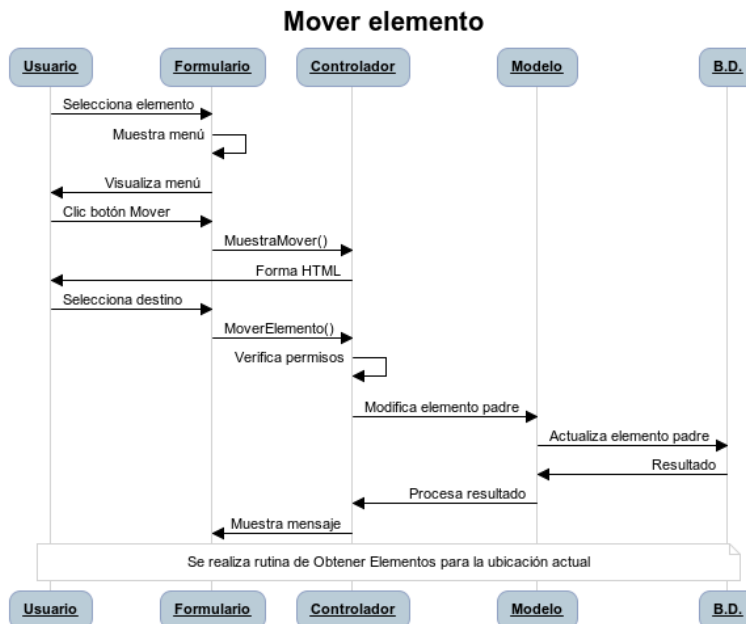
Fuente: Elaboración propia.

Ilustración que muestra el diagrama de secuencia de la funcionalidad obtener elementos.

En la Ilustración 38 se observa la secuencia de la funcionalidad obtener elementos, esta función posiblemente sea la más utilizada dentro del repositorio, ya que es ejecutada cada vez que el usuario navega entre los elementos carpetas. Es por esto que se planeó utilizar AJAX para devolver la lista de elementos y así solo actualizar parte del documento HTML, lo que permite un uso de los recursos equilibrado.

En el diagrama de secuencia de la Ilustración 39 muestra como interactúa la funcionalidad mover elementos, el usuario selecciona uno o varios elementos que se proporcionan como parámetro al controlador, además del nodo destino, el controlador verifica que en el nodo destino no exista un elemento con un nombre igual, que se tengan los permisos sobre los elementos a mover, entre otras verificaciones.

Ilustración 39. Diagrama de secuencia mover elemento.

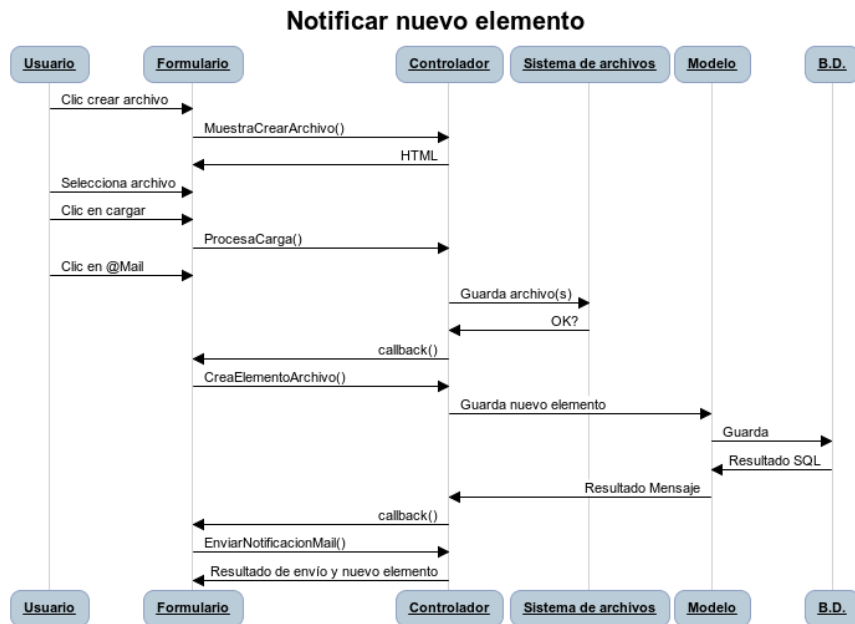


Fuente: Elaboración propia.

Ilustración que muestra el diagrama de secuencia de la funcionalidad mover elemento.

Por último en el diagrama de secuencia para notificar nuevo elemento (Ilustración 40), se invoca cuando se termina de cargar los archivos si la casilla *@Mail* está marcada dentro de la funcionalidad crear archivos, entonces el sistema envía un aviso vía correo electrónico a los usuarios que tienen acceso a la ubicación que contiene los archivos.

Ilustración 40. Diagrama de secuencia notificar nuevo elemento.



Fuente: Elaboración propia.

Ilustración que muestra el diagrama de secuencia de la funcionalidad notificar nuevo elemento.

#### 4.6.7 Diseño de los casos de prueba.

Los casos de prueba a validar se diseñaron de acuerdo a lo esperado en la descripción de los casos de uso, se ejecutaron una vez que se desarrollaron las funcionalidades para verificar que el software realizaba lo acordado, si se suscitaba algún error o el sistema hacía algo no planeado, se tomaba nota en el registro y se comentaba el resultado. Cabe mencionar que estas pruebas fueron ejecutadas por usuarios ajenos al desarrollo del sistema del departamento de sistemas de la organización. El diseño final de los casos de prueba se puede observar en el ANEXO 2 del presente documento.

Los resultados de los casos de prueba se abordarán en el punto 4.6.8, etapa posterior al desarrollo de las funcionalidades.

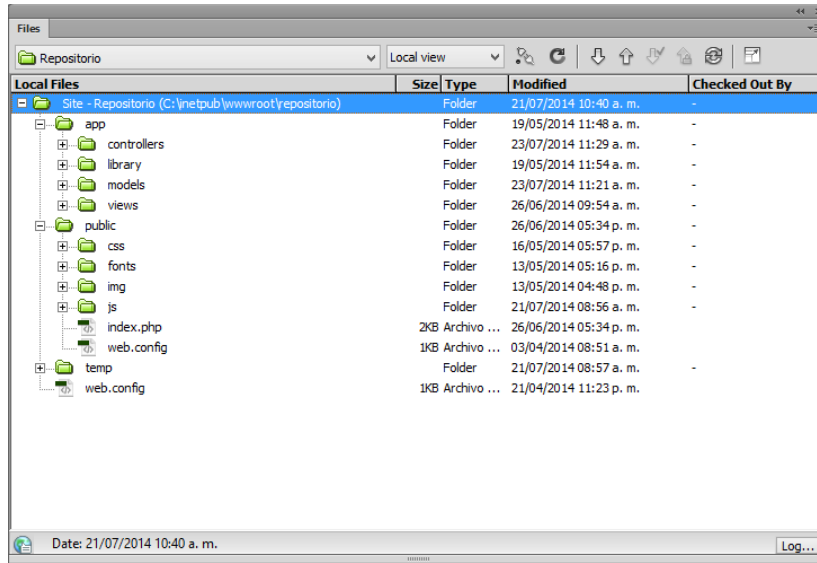
#### 4.6.8 Ejecución de la fase de desarrollo.

Para el desarrollo del software repositorio se seleccionó el lenguaje de lado del servidor PHP, debido a la aceptación por parte del mercado como se observa en W3Techs (2013), así mismo se eligió el *framework Phalcon PHP* ya que permite trabajar con la arquitectura MVC y por obtener buenos resultados en las calificaciones de rendimiento según se observa en el sitio Web TechEmpower (2014). Adicional a esto, el lenguaje de programación PHP ya es utilizado por la organización para los sistemas existentes, lo que aumentaba la compatibilidad con el servidor IIS y permitió una adaptación mínima.

Cabe mencionar que tanto el lenguaje PHP como el *framework Phalcon* cuentan con licenciamiento tipo distribución de software Berkeley (BSD del inglés *Berkeley Software Distribution*), por lo que se puede utilizar para efectos comerciales. Para la representación de los elementos HTML del lado del *frontend* se utilizó el *framework Bootstrap* ya que permite aplicar diseños preestablecidos para cada elemento además de contar con licencia del Instituto Tecnológico de Massachusetts (MIT del inglés Massachusetts Institute of Technology), por lo que se puede utilizar libremente.

Para la gestión de los archivos fuentes se utilizó *Adobe Dreamweaver* mediante la creación de un sitio y un servidor local para las pruebas, los archivos fuentes fueron ordenados conforme la estructura de directorios predeterminada del *framework* (Sitio Web Phalcon PHP, 2014), la cual se puede apreciar en la Ilustración 41.

Ilustración 41. Estructura de los archivos fuentes del proyecto.

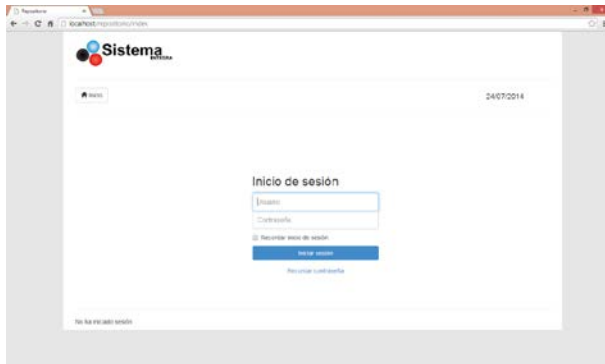


Fuente: Elaboración propia.

Ilustración que muestra la estructura de directorios del sistema repositorio, la cual es la predeterminada por el *framework* *Phalcon* PHP.

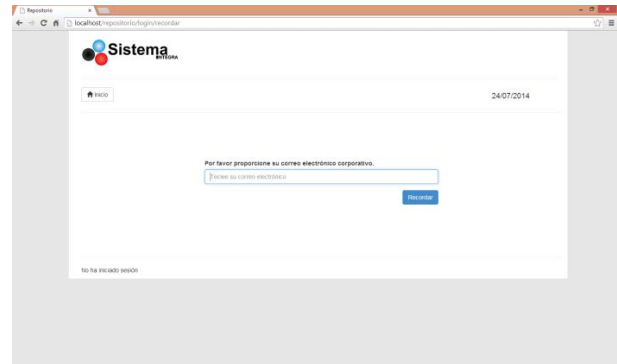
En la primera fase se obtuvieron las funcionalidades para iniciar sesión, recordar contraseña y cerrar sesión, según lo planificado. Los resultados se exponen a continuación.

Ilustración 42. Resultado del desarrollo de la funcionalidad inicio de sesión.



Fuente: Elaboración propia.

Ilustración 43. Resultado del desarrollo de la funcionalidad recordar contraseña.



Fuente: Elaboración propia.

Ilustración que muestra la funcionalidad para iniciar sesión en el sistema visualizada a través del navegador Google Chrome.

Ilustración que muestra la funcionalidad recordar contraseña donde el usuario debe teclear su correo electrónico.

En la Ilustración 42, se solicita al usuario un usuario y contraseña, debido a que se utilizó la misma base de datos de usuarios y roles existente del sistema de expedientes, los usuarios no tienen que recordar

nuevas credenciales, lo que facilitó la adopción del sistema. Las formas HTML son parte del *frontend* del sistema, se utilizó la librería *Javascript JQuery* para la validación de los datos, un ejemplo se puede observar en la Ilustración 44. La elección de esta librería se debió a la aceptación del mercado mostrada en W3Techs (2013) y debido a su amplia documentación y funciones brindadas.

Ilustración 44. Función Javascript para validar forma HTML.

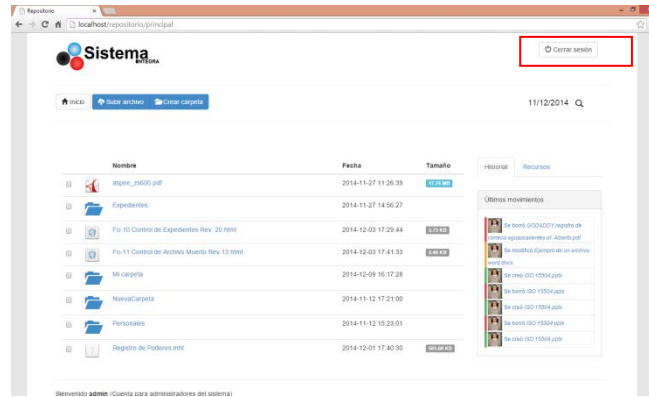
```
1 //FUNCION PARA VALIDAR EL FORMULARIO
2 $('#formallogin').submit(function () {
3
4     //OBTENIENDO CAMPOS DEL FORMULARIO
5     var username = $.trim($('#username').val());
6     var password = $.trim($('#password').val());
7
8     //VERIFICAR DATOS DE FORMULARIO
9     if (username === '') {
10        alert('Por favor proporcione su nombre de usuario');
11        return false;
12    }
13    if (password === '') {
14        alert('Por favor proporcione su contraseña');
15        return false;
16    }
17
18    //SI TODAS LAS VALIDACIONES SON CORRECTAS
19    $('#btniniciar').attr('disabled', true);
20    $('#btniniciar').html('Por favor espere...');
21});
```

Fuente: Elaboración propia.

Ilustración que muestra la función *Javascript* para validar la forma para iniciar sesión en el sistema, misma que hace uso de la librería *Javascript JQuery*.

En la Ilustración 43, el usuario proporciona su correo electrónico y el sistema ejecuta una función que hace llegar su usuario y contraseña al correo electrónico registrado, cabe mencionar que el control de usuarios se realiza en el sistema de expedientes, como se ha mencionado.

Ilustración 45. Resultado del desarrollo de la funcionalidad cerrar sesión.



Fuente: Elaboración propia.

Ilustración que muestra la funcionalidad cerrar sesión en la parte superior derecha, la cual es representada mediante un botón llamado “cerrar sesión” que al ser presionado elimina las variables de sesión.

En la Ilustración 45 se muestra el resultado de la funcionalidad cerrar sesión, la cual es un botón en la parte superior derecha que al ser presionado destruye las variables de sesión. Para esto se invoca el método `cerrarSesion` del objeto `LoginController`.

Ilustración 46. Método para finalizar la sesión de la clase `LoginController`.

```
1 public function cerrarSesionAction()  
2 {  
3     $this->session->remove('auth');  
4     $this->flash->success('Ha terminado su sesión en el sistema correctamente');  
5     return $this->dispatcher->forward(array('controller' => 'index', 'action' =>  
6     'index'));  
}
```

Fuente: Elaboración propia.

Ilustración que muestra la función que elimina las variables de sesión creadas por el sistema para autenticar a los usuarios en los controladores.

En la línea tres de la Ilustración 46 se muestra como se destruye la variable de sesión que guarda los datos del usuario, tales como el id del usuario, el nombre de usuario y el nivel. Después en la línea cuatro se establece un mensaje el cual es mostrado en la vista, ver Ilustración 47.

Ilustración 47. Despliegue de mensaje de una sesión terminada.

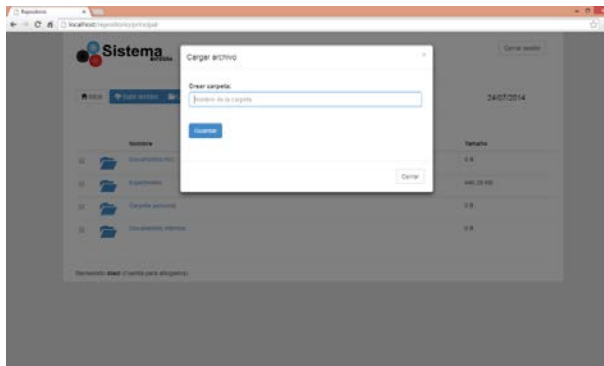


Fuente: Elaboración propia.

Ilustración que muestra el resultado de presionar el botón para cerrar sesión en el sistema, un mensaje confirma al usuario el resultado de la acción.

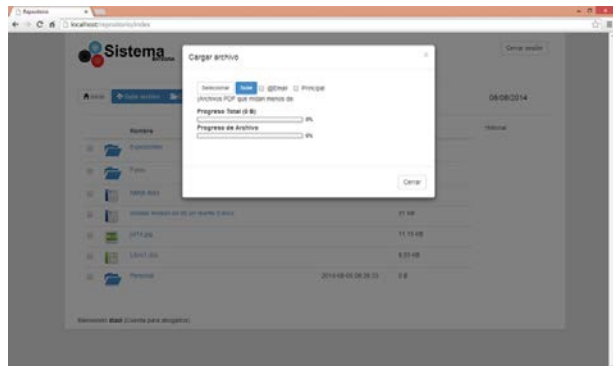
En la siguiente fase se desarrollaron las funciones administrativas, por lo que el resultado se puede apreciar en las siguientes ilustraciones.

Ilustración 48. Resultado del desarrollo de la funcionalidad crear carpeta.



Fuente: Elaboración propia.

Ilustración 49. Resultado del desarrollo de la funcionalidad crear archivo.



Fuente: Elaboración propia.

Ilustración que el formulario para una crear carpeta el cual es un solo campo donde se escribe el nombre del elemento.

Ilustración que muestra el formulario para subir archivos al sistema haciendo uso del *plugin* Fancybox (Kirschner, 2014).

En la Ilustración 48 el usuario teclea el nombre de la carpeta que desea crear, internamente la vista envía al controlador la ubicación actual y nombre, esto se puede apreciar en la línea 5 y 6 de Ilustración 50, posteriormente en la línea 15 se guarda el elemento en la base de datos.

Ilustración 50. Función del controlador LoginController para guardar elemento carpeta.

```

1 public function agregarCarpetaAction(){
2     //SE DESHABILITA LA VISTA
3     $this->view->disable();
4     //SE RECIBEN LOS VALORES DEL FORMULARIO
5     $sele_nombre = $this->request->getPost('ele_nombre', 'string');
6     $sele_padre = $this->request->getPost('ele_padre', 'int');
7
8     //SE CREA EL OBJETO ELEMENTO PARA GUARDARLO EN LA BD
9     $elemento = new Elementos();
10    $elemento->ele_nombre = $sele_nombre;
11    $elemento->ele_tamano = 0;
12    $elemento->ele_padre = $sele_padre;
13    //SE INTENTA GUARDAR EN LA BD
14    try{
15        if($elemento->save() == false) {
16            ...

```

Fuente: Elaboración propia.

Ilustración que muestra parte del código fuente de la funcionalidad para agregar una carpeta, misma que se encuentra contenida dentro de una función en el controlador ElementoController.

En la funcionalidad de la Ilustración 49, el usuario debe hacer clic sobre el botón “Subir archivo”, en seguida se muestra la forma donde se pueden proporcionar uno o varios archivos, cabe mencionar que solo se permiten archivos con las extensiones especificadas en la Tabla 21.

Tabla 21. Extensiones permitidas en la funcionalidad crear archivo.

Representación visual	Extensiones
	*.jpg, *.jpeg
	*.png
	*.pdf
	*.doc, *.docx
	*.xls, *.xlsx
	*.ppt, *.pptx
	*.swf
	*.html

Fuente: Elaboración propia con uso de iconos KFaenza v0.8.9.

Tabla que muestra la iconografía utilizada para mostrar la lista de elementos al navegar entre las diferentes carpetas, el uso de archivos SWF y HTML fue solicitado por la organización para el uso de documentos oficiales y formularios.

Los iconos representados en la Tabla 21 son los utilizados por la vista para desplegar los elementos, dichos iconos están liberados bajo la licencia GPL y pueden ser usados para fines institucionales (Free Software Foundation, 2014).

Una vez que la transferencia de los archivos hacia el servidor se completa, se muestra el resultado para cada uno de los archivos. Existe la posibilidad de que el sistema muestre errores como resultado, esto se debe a que no se permiten que dos archivos tengan el mismo nombre dentro de una carpeta u otros relacionados a la comunicación con el servidor Web. Los archivos son guardados en el disco duro del servidor con un nombre generado mediante marcas de tiempo y variables aleatorias procesadas mediante el algoritmo *Secure Hash Algorithm* versión 1 (SHA1). En la Ilustración 51 se muestra el procesamiento que recibe el archivo por el controlador *ElementoController*.

Ilustración 51. Procesamiento que reciben los archivos en el controlador.

```
1 //REVISAR SI HAY ARCHIVOS
2 if($this->request->hasFiles() == true){
3     //SE COLOCAN TODOS LOS ARCHIVOS EN UN ARRAY
4     $uploads = $this->request->getUploadedFiles();
5     //SE PROCESA CADA ARCHIVO
6     foreach($uploads as $upload){
7         //SE OBTIENE TIPO DE ARCHIVO
8         $finfo = new finfo(FILEINFO_MIME_TYPE);
9         $fres = $finfo->file($upload->getTempName());
10        $hashname = sha1(uniqid(rand(), true));
11        //SE VERIFICA QUE EL TIPO ESTE DENTRO DE LOS PERMITIDOS
12        $validarExtension = $this->revisarExtension($fres);
13        if($validarExtension['valido'] != 0){
14            //SE CREA EL OBJETO ELEMENTO PARA GUARDARLO EN LA BD
15            $elemento = new Elementos();
16            ...

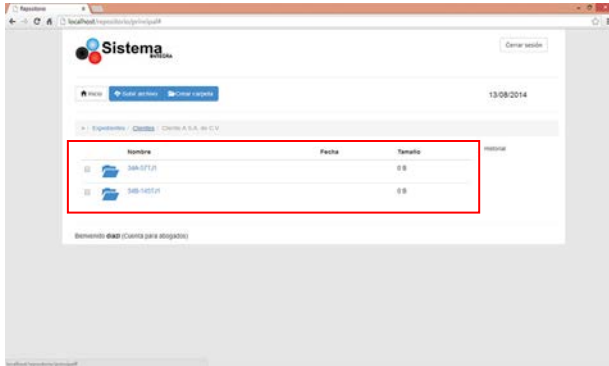
```

Fuente: Elaboración propia.

Ilustración que muestra parte de la función encargada de procesar los archivos enviados por el formulario encargado de agregar dichos elementos al sistema.

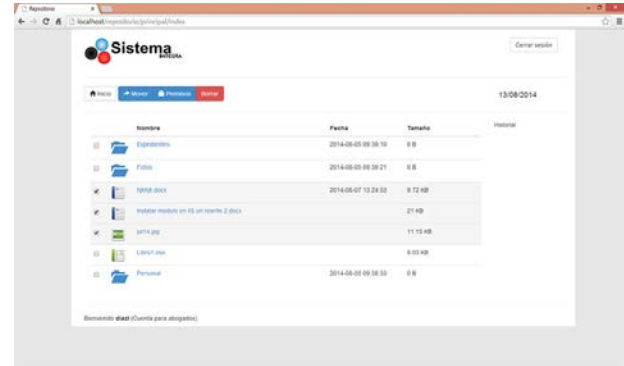
En la línea 10 de la Ilustración 51 se muestra como es renombrado el archivo, con el nombre generado, se guarda en el disco duro del servidor Web, posteriormente en la línea 13 se verifica la extensión del archivo de forma segura.

Ilustración 52. Resultado del desarrollo de la funcionalidad obtener elementos.



Fuente: Elaboración propia.

Ilustración 53. Resultado del desarrollo de la funcionalidad selección de elementos.



Fuente: Elaboración propia.

En la ilustración se observa con el área marcada en rojo la sección que se actualiza al navegar entre las carpetas.

En la ilustración se observa el cambio de color que indica cuales elementos se encuentran seleccionados.

La Ilustración 52 muestra el resultado de la funcionalidad obtener elementos, ésta es ejecutada cada vez que el usuario hace clic sobre el hipervínculo de un elemento carpeta, para invocarla se ejecuta una función Javascript (Ilustración 54) que a su vez ejecuta el método *obtenerElementos* en el controlador, mismo que devuelve la lista de elementos mediante AJAX a la vista.

Ilustración 54. Función Javascript que obtiene elementos mediante AJAX.

```
1 //FUNCION PARA OBTENER LISTA DE ELEMENTOS DE UN NODO DADO
2 function obtenerElementosJS(nodo){
3   var er_nodo = /^[0-9]+$;/
4   var nodomostrar = (!er_nodo.test(nodo))? 0 : nodo;
5   nodoActual = nodomostrar;
6   $.ajax({
7     url: "<?php echo $this->url->get("elemento/obtenerElementos")
8     >?"+"/"+nodomostrar,
9     type: 'POST',
10    data: { nodo: nodomostrar },
11    success: function(data){
12      $('#contenedorElementos').html(data);
13      obtenerBreadcrumbJS(nodomostrar);
14      menu();
15    }
16  });
17 }
```

Fuente: Elaboración propia.

En la ilustración se observa parte del código *Javascript* encargado de obtener los elementos, ejecutando para ello la función del controlador *ElementoController* la función *obtenerElementos*.

Como se observa en la Ilustración 54 en la línea 12, la función para obtener elementos ejecuta un método que obtiene la ruta actual de la ubicación del usuario dentro del árbol jerárquico, lo que permite que el usuario navegue con más fluidez por medio de hipervínculos dentro del repositorio, ver Ilustración 55.

**Ilustración 55. Ruta de la ubicación actual dentro del repositorio.**



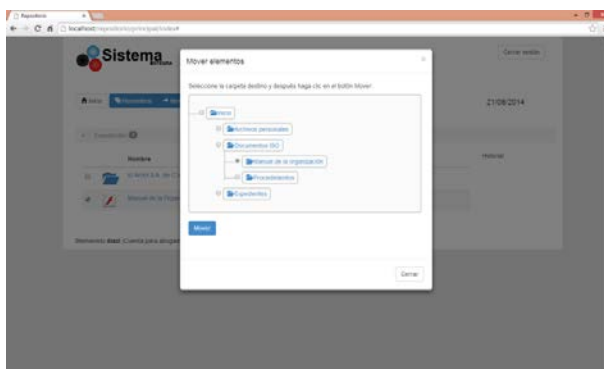
Fuente: Elaboración propia.

Ilustración que muestra la ruta actual al navegar entre las diferentes carpetas, misma que se actualiza en la vista cada vez que el usuario hace clic sobre un elemento carpeta.

Adicional a esto, la ruta muestra el total de elementos de la carpeta actual a través de un número contenido en un óvalo.

La Ilustración 53 muestra la funcionalidad selección de elementos, la cual es llamada en el momento que el usuario hace clic en la casilla de selección del elemento. El sistema cambia el menú de acción ubicado en la parte superior dependiendo del número de elementos seleccionados, en caso de que el usuario deseccione los elementos el menú vuelve a su estado original.

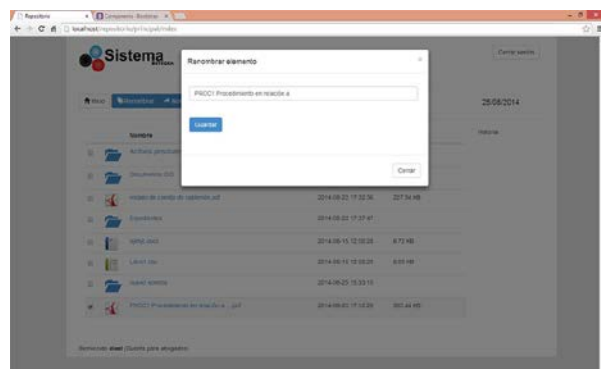
**Ilustración 56. Resultado del desarrollo de la funcionalidad mover elementos**



Fuente: Elaboración propia.

Ilustración que muestra el formulario para seleccionar un destino a los elementos que se encuentran seleccionados.

**Ilustración 57. Resultado del desarrollo de la funcionalidad renombrar elemento.**



Fuente: Elaboración propia.

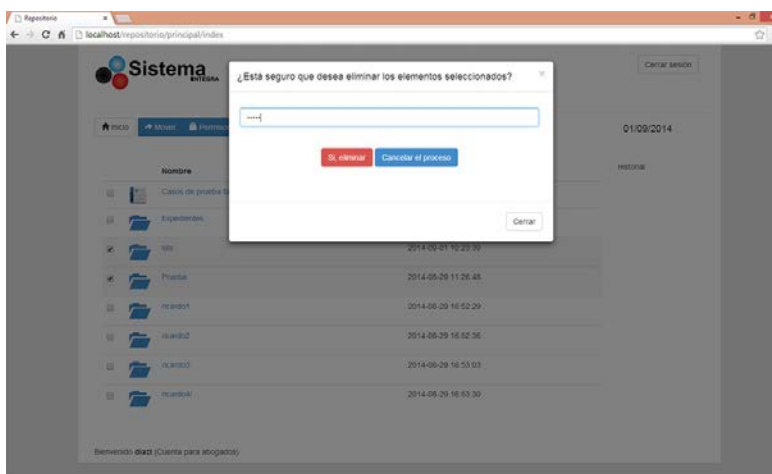
Ilustración que muestra el formulario compuesto de un solo campo para renombrar al elemento seleccionado.

En la Ilustración 56 se muestra el resultado de la funcionalidad mover elementos, el usuario selecciona los elementos a mover de la vista principal, entonces al presionar el botón mover del menú de acción se abre una ventana donde el usuario selecciona el destino. Para esto el controlador verifica cada uno de los elementos para que cumplan los siguientes requisitos:

- No se puede mover una carpeta a niveles inferiores dentro de la misma ramificación.
- No se puede mover un elemento a una carpeta que ya contenga otro con el mismo nombre.

Para renombrar un elemento (Ilustración 57) debe estar seleccionado solo uno, el usuario presiona el botón correspondiente en el menú de acciones y entonces se muestra una ventana con el formulario donde se proporciona el nuevo nombre, predeterminadamente en el campo nombre se coloca el nombre actual.

**Ilustración 58. Resultado del desarrollo de la funcionalidad borrar elementos.**



Fuente: Elaboración propia.

Ilustración que muestra el formulario para confirmar la eliminación de los elementos seleccionados, el usuario debe escribir su contraseña para validar la eliminación.

En la Ilustración 58 se muestra la funcionalidad borrar, la cual consiste en seleccionar los elementos deseados y hacer clic sobre el botón borrar, entonces se abre una ventana de confirmación donde se escribe la contraseña del usuario que ejecuta la acción, si la autenticación es válida entonces el controlador elimina los elementos de la base de datos y archivos del disco duro. Para ello el controlador

crea una transacción con la base de datos para garantizar que todas las operaciones sean realizadas

Ilustración 59.

**Ilustración 59. Inicio de la transacción en la base de datos repositorio.**

```
1 if($usu->usu_password == $password){  
2 //SE INICIA LA TRANSACCION  
3 $this->dbrepositorio->begin();
```

Fuente: Elaboración propia.

Ilustración que muestra el inicio de una transacción en la base de datos para la funcionalidad borrar elementos, así se garantiza que todos los elementos seleccionados y sus elementos contenidos sean eliminados.

En la última tarea de esta fase se desarrolló la funcionalidad para la gestión de los permisos, dicha funcionalidad permite controlar el acceso a los diferentes elementos, los permisos efectivos para cada elemento recayeron en cuatro posibles grupos, Sin acceso, Control, Modificar y Lectura. Ver Tabla 22.

**Tabla 22. Tabla de permisos efectivos para los elementos.**

Grupo	Permisos efectivos
<b>Sin acceso</b>	Sin ningún tipo de acceso al elemento
<b>Control</b>	Cambiar permisos, borrar, mover, renombrar, crear archivos y carpetas, ver su contenido.
<b>Modificar</b>	Mover, renombrar, crear archivos y carpetas, ver su contenido.
<b>Lectura</b>	Solo ver.

Fuente: Elaboración propia.

Tabla que muestra los permisos efectivos para cada grupo seleccionable en el formulario para cambiar permisos a los elementos.

Para cambiar el permiso a un elemento el usuario puede seleccionar uno o más de un elemento de la vista, acto seguido a través del menú de acción debe hacer clic sobre el botón “permisos”, entonces el sistema mostrará la funcionalidad como se muestra en la Ilustración 60.

Ilustración 60. Resultado del desarrollo de la funcionalidad cambiar permisos.

Permisos por niveles	Permisos por usuarios
Cuenta para administradores del sistema	Control
Cuenta para abogados	Lectura
Cuenta para administracion y cobranza	Lectura
Cuenta para usuarios invitados	SIN ACCESO
Cuenta para atención al cliente	Lectura

Fuente: Elaboración propia.

Ilustración que muestra el formulario para cambiar los permisos a los elementos que en su momento se encuentren seleccionados, en la imagen se observa activada la modalidad de permisos por niveles.

Como se observa en la Ilustración 60, existen dos formas de otorgar permisos a los elementos, una es llamada “permisos por niveles” y otra “permisos por usuarios”, mediante la primera se especifican los permisos para todos los usuarios que pertenezcan al nivel señalado.

En cambio sí se activan los permisos por usuario, entonces tendrá que especificar el tipo de permiso para cada usuario del sistema. Una vez que se está satisfecho con los cambios realizados, se debe hacer clic sobre el botón “Guardar” y un mensaje confirmará la aplicación.

Habría que mencionar que predeterminadamente se asignan los permisos de la ubicación actual al agregar un archivo o carpeta. Así mismo, mediante el controlador de permisos, se verifica que el usuario tenga acceso al elemento, por ejemplo en la Ilustración 61 se observa el aviso que se recibe si no se tiene permiso de lectura sobre un elemento.

**Ilustración 61. Aviso de permiso negado al visualizar un elemento.**

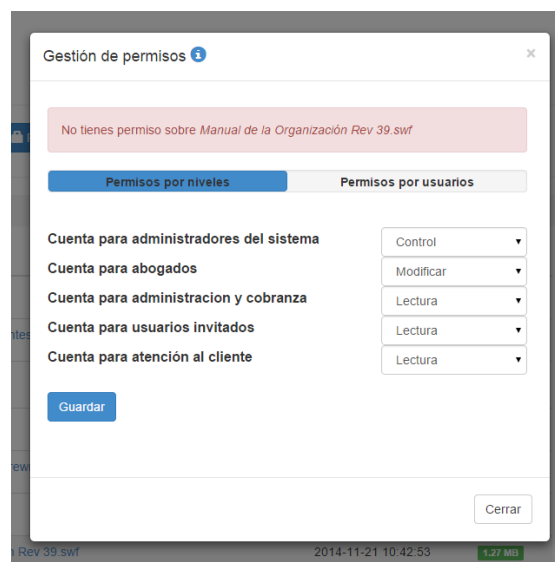


Fuente: Elaboración propia.

Ilustración que muestra el mensaje de permiso negado a un usuario que trata de visualizar el contenido de un archivo en formato PDF y que no autorización.

En otras funcionalidades del sistema también se verifica que el usuario tenga permiso de realizar cambios, tal es el caso el ejemplo de la Ilustración 62, donde el usuario puede acceder a la funcionalidad, sin embargo si se realizan cambios, el sistema mostrará un aviso.

**Ilustración 62. Verificación de permisos efectivos.**

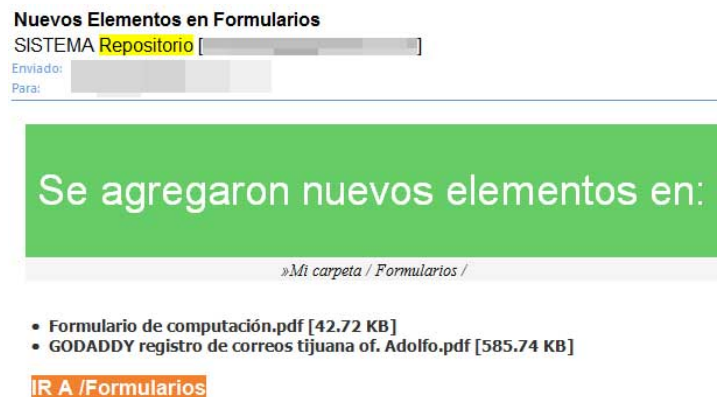


Fuente: Elaboración propia.

Ilustración que muestra el resultado de intentar cambiar los permisos sobre algún elemento cuando no se tiene el permiso de control.

La tercera fase del desarrollo vino a cerrar el ciclo de desarrollo, en esta fase se realizaron las últimas tres funcionalidades que vendrían a apoyar los aspectos de colaboración e integración. La funcionalidad para notificar nuevo elemento permite que los usuarios reciban un correo electrónico si se agregaron elementos en alguna carpeta a la que tienen acceso, ver Ilustración 63.

**Ilustración 63. Correo electrónico recibido por la funcionalidad enviar notificación.**

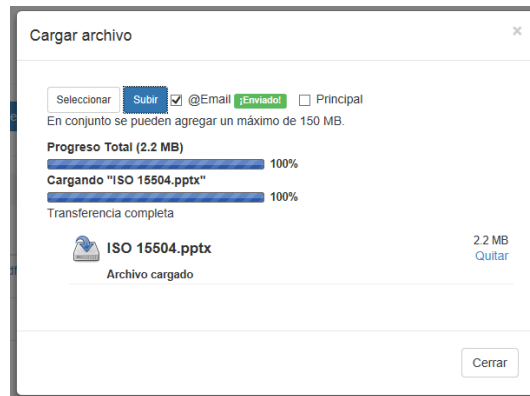


Fuente: Elaboración propia.

Ilustración que muestra un correo electrónico enviado por la funcionalidad para notificar nuevos elementos, se observa que el aviso contiene la ubicación donde se agregaron los archivos, el nombre, así como un enlace hacia dicha ubicación.

Para indicar al sistema que se desea notificar a los usuarios, se debe hacer clic sobre la casilla “@Email” al agregar un archivo, una vez transferidos los archivos al servidor, entonces se iniciará el proceso de notificación mediante una función tipo *callback*, el resultado se notifica al usuario mediante una marca color verde si fue satisfactorio o color rojo si ocurrió algún error, ver Ilustración 64.

Ilustración 64. Resultado del desarrollo de la funcionalidad enviar notificación.

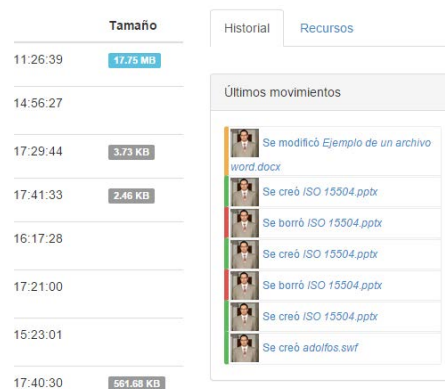


Fuente: Elaboración propia.

Ilustración que muestra un aviso con una etiqueta color verde indicando que el envío de la notificación fue satisfactoria, en caso contrario se mostraría una etiqueta color rojo.

La casilla llamada “principal” en la función para crear archivos, permite agregar un enlace que direcciona al usuario a la ubicación donde se encuentran los nuevos elementos. Estos enlaces se encuentran en las notificaciones del sistema de expedientes. Mientras tanto en la página principal del repositorio se muestran aquellos elementos recientemente creados, borrados o modificados. Para lo anterior, se identifica cada movimiento con una franja verde si el elemento fue creado, amarillo si fue modificado y rojo se fue eliminado, ver Ilustración 65.

Ilustración 65. Enlaces a elementos desde la página principal.



Fuente: Elaboración propia.

Ilustración que muestra enlaces directos a la ubicación donde se encuentran los elementos recién agregados, modificados o eliminados.

Como se muestra en la Ilustración 65, si el usuario hace clic sobre el nombre de algún elemento, el sistema lo conduce a la ubicación que lo contiene. Adicionalmente si el usuario coloca el cursor sobre el avatar del usuario se muestra la fecha y hora que se realizó la operación.

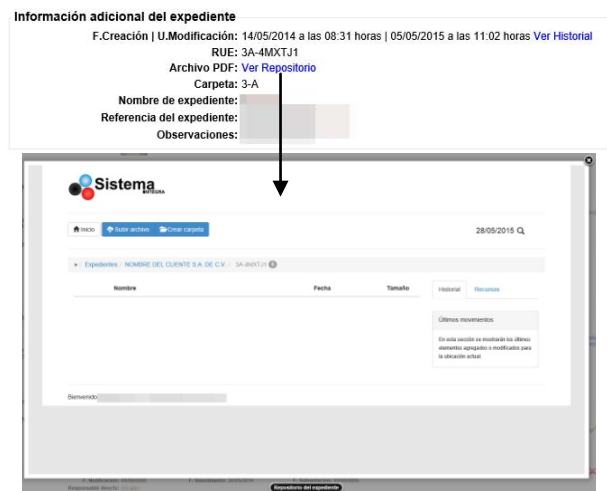
Por último la funcionalidad para agregar archivos desde el sistema de expedientes se implementó mediante un buscador de elementos y mediante acceso en el sistema de expedientes, ver Ilustración 66. El acceso directo usa la clave única de expediente (RUE) para buscar en el repositorio el elemento carpeta con dicho nombre, si se encuentra, entonces el sistema redirige al usuario a esa ubicación, si no se encuentra se brinda la opción de crearla.

**Ilustración 66. Resultado del desarrollo de la funcionalidad para agregar archivos desde detalles de expedientes.**

Si no existe repositorio para el expediente:



Si existe repositorio para el expediente:



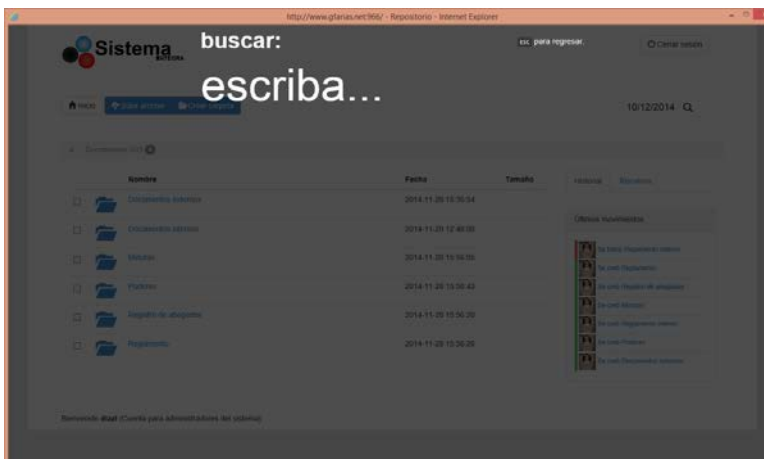
Fuente: Elaboración propia.

Ilustración que muestra la funcionalidad para agregar archivos desde el sistema de expedientes, si el expediente no cuenta con espacio dentro del repositorio se brinda la opción de crearlo, si ya lo tiene entonces se redirige.

Además, la herramienta de búsqueda también permite encontrar elementos a través de criterios textuales, ver Ilustración 67, el usuario debe invocar dicha función a través del icono de la lupa ubicada

en la parte superior derecha, entonces el sistema superpone una capa donde el usuario puede escribir el nombre del elemento buscado.

Ilustración 67. Herramienta de búsqueda de elementos.



Fuente: Elaboración propia.

Ilustración que muestra en funcionamiento la herramienta de búsqueda de elementos, el usuario escribe sobre una capa superpuesta y presiona la tecla entrar, o bien, la tecla de escape para cancelar la acción.

#### 4.6.9 Resultado de las pruebas a las funcionalidades.

Los casos de prueba definidos en la sección 4.6.6 se ejecutaron para verificar que el software realizara lo esperado, para ello se empleó un formato en una hoja de cálculo que funcionó como reporte el cual contenía la información mostrada en la Tabla 23.

Tabla 23. Formato del reporte de resultados de los casos de prueba.

RESULTADOS DE LOS CASOS DE PRUEBA				
CASO DE PRUEBA #	NOMBRE PERSONA QUE EJECUTÓ	DEPARTAMENTO	RESULTADO	OBSERVACIONES
###	Nombre de la persona que ejecutó	Departamento	PASÓ ó NO PASÓ	Comentario acerca del resultado

Fuente: Elaboración propia.

Tabla que muestra el formato a seguir para anotar el resultado de los casos de prueba que van a ser ejecutados en las funcionalidades.

Una vez que fueron ejecutadas las pruebas, se obtuvieron los comentarios de la persona encargada de llevarlas a cabo, el resultado de la ejecución se puede observar en la Tabla 24.

Tabla 24. Resultados de las pruebas de las fases 1 y 2.

CASO DE PRUEBA #	NOMBRE PERSONA QUE EJECUTÓ	DEPARTAMENTO	RESULTADO	OBSERVACIONES
0	Coordinador de sistemas	Sistemas	PASÓ	
1	Coordinador de sistemas	Sistemas	PASÓ	
2	Coordinador de sistemas	Sistemas	PASÓ	
3	Coordinador de sistemas	Sistemas	NO PASÓ	No se pudo enviar la información de la cuenta de administrador al correo sistema@xxxxxx.com. Error: SMTP connect() failed.
4	Coordinador de sistemas	Sistemas	PASÓ	Acepta caracteres especiales, quizá restringir un poco ya que puede crear una carpeta con nombre %&((&("."))=\$^^?
5	Coordinador de sistemas	Sistemas	PASÓ	
6	Coordinador de sistemas	Sistemas	PASÓ	No permite subir un archivo con extensión no permitida. Pero si el archivo es un pdf y se cambia su extensión a exe, si permite subirlo. En Chrome se inmoviliza la pantalla y se debe de cerrar, en Explorer simplemente no deja seleccionar un archivo no permitido (Windows 7)
7	Coordinador de sistemas	Sistemas	PASÓ	
8	Coordinador de sistemas	Sistemas	PASÓ	
9	Coordinador de sistemas	Sistemas	PASÓ	
10	Coordinador de sistemas	Sistemas	PASÓ	
11	Coordinador de sistemas	Sistemas	PASÓ	
12	Coordinador de sistemas	Sistemas	NO PASÓ	No permite hacer dos veces la operación de dar permisos.
13	Coordinador de sistemas	Sistemas	PASÓ	Se despliegan correctamente los últimos eventos
14	Desarrollador	Sistemas	NO PASÓ	Marca una notificación de error al enviar el correo electrónico.
15	Desarrollador	Sistemas	PASÓ	

Fuente: Elaboración propia.

Tabla que muestra el resultado de la ejecución de los caso de prueba a las diferentes funcionalidades del sistema, las mismas se encuentran con comentarios originales de las personas que fueron encargadas de ejecutarlas.

La ejecución de las pruebas dio como resultado el que se refinaran las funcionalidades donde hubo inconsistencias, por ejemplo la prueba con el número 3 de la Tabla 24, se refiere a la funcionalidad para recordar contraseña, en este sentido se detectó que la prueba no pasó debido a un bloqueo del puerto 25 que impidió la comunicación con el servidor de correo, sin embargo en producción este problema no

se replicó, no obstante la funcionalidad fue adecuada para permitir especificar puerto TCP para la comunicación, en caso de que sea requerido en el futuro.

Las pruebas 4 y 6 que presentaron anotaciones, se tomaron en cuenta para su consideración, en el caso de la prueba 6 se verificó la funcionalidad crear archivo y se encontró que se obtenían resultados diferentes en navegadores distintos, por lo que se trabajó a nivel controlador para corregir lo mencionado.

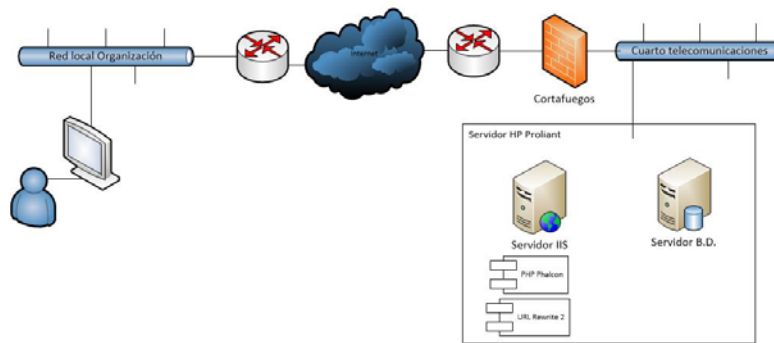
Con respecto a la prueba 12 se detectó que por omisión no se había activado una función del controlador PermisoController, por lo que se suscitaban dichos eventos, sin embargo con la corrección se logró obtener el resultado esperado.

El problema de la prueba 12 requirió cambiar la versión de la librería PHPMailer, ya que en el desarrollo se utilizó la versión más reciente, sin embargo no es compatible del todo con el proveedor de correo electrónico actual, por lo que fue requerido regresar a una versión anterior para solventar el problema.

#### **4.7 Implementación del sistema de gestión de documentos de archivo.**

El sistema fue puesto en marcha en un servidor Web ejecutando *Microsoft Windows Server 2008 R2* con IIS activado, ya que este servidor web es el utilizado por la organización desde el 2005. El acceso al servidor Web se realiza a través de internet, pues el cuarto de telecomunicaciones y la red de la organización se encuentran en ubicaciones diferentes, lo anterior puede apreciarse en la Ilustración 68.

Ilustración 68. Acceso al servidor web donde se alojará la aplicación.



Fuente: Elaboración propia.

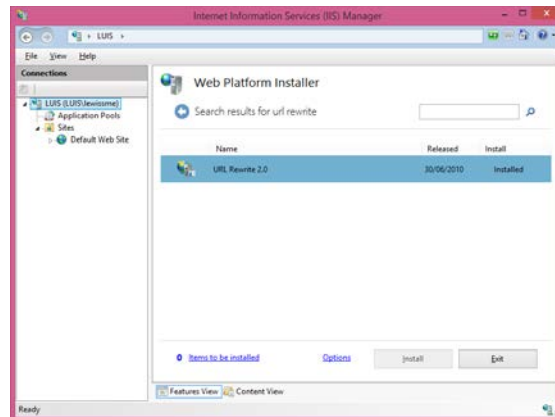
Ilustración que expone el acceso al sistema repositorio por parte de la organización.

Como se mencionó en la sección 4.5, una vez instalado y terminada la segunda fase del proyecto, se llevó a cabo la prueba piloto con los documentos del sistema ISO, por lo que las personas ahora acceden a estos documentos a través del repositorio. Para esto se agregaron los enlaces necesarios en la página principal del sistema integra, a la cual se tiene acceso mediante una aplicación de escritorio. En cuanto al personal encargado de proporcionar la información digital, no hubo necesidad de hacer cambios y se continuó trabajando de la misma forma.

#### 4.7.1 Configuración adicional del servidor.

El servidor contaba con la versión de PHP 5.3.11, sin embargo se tuvo que actualizar a la versión 5.3.28, ya que en la etapa de Desarrollo y Pruebas se observaron inconsistencias con el *framework Phalcon* PHP. Así mismo, fue requerido instalar un módulo adicional de IIS llamado *URL Rewrite 2.0*, esto a través del Instalador de Plataforma Web (*Microsoft Platform Installer*) integrada en la consola de administración de IIS, ver Ilustración 69.

Ilustración 69. Instalación de URL Rewrite 2.0 mediante la consola de IIS.



Fuente: Elaboración propia.

Ilustración que muestra la consola de administración del servidor Web IIS donde se hace uso de la plataforma *Web Platform Installer* para añadir el módulo *URL Rewrite 2.0*.

Si bien el *framework* no obliga el uso de URL amigables, se decidió su utilización, pues permite invocar los controladores y enviar parámetros a éstos por medio de la URL, de tal forma que para el usuario represente una navegación sencilla y fácil de recordar. Para ello se colocaron dos archivos con extensión (\*.config) en la carpeta raíz del proyecto y otro en el directorio *public* de la aplicación, el contenido de la carpeta raíz se observa en la Ilustración 70.

Ilustración 70. Contenido del archivo *web.config* en el directorio raíz usado por URL Rewrite de IIS.

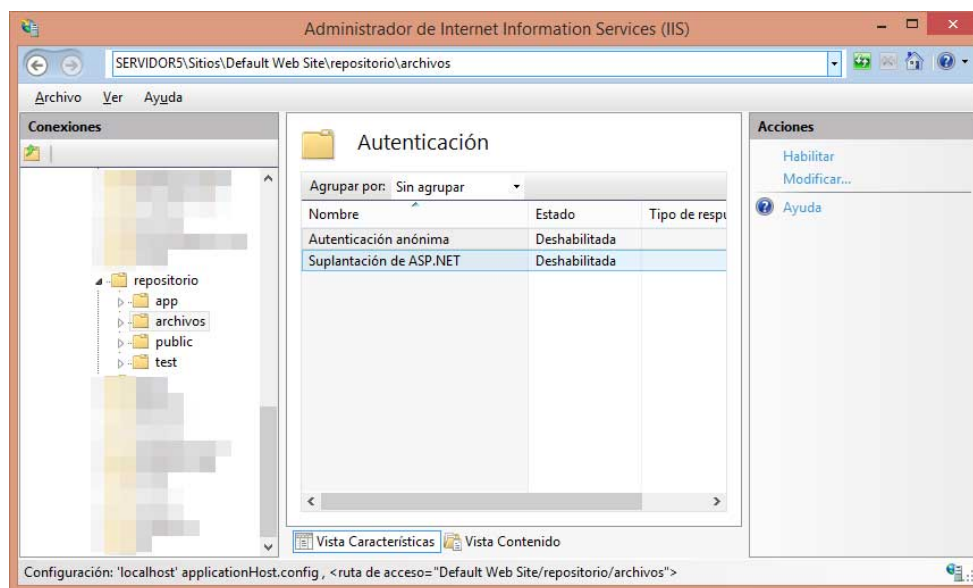
```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <configuration>
3 <system.webServer>
4 <rewrite>
5 <rules>
6 <rule name="Main Rule2" stopProcessing="true">
7 <match url="(.*)" />
8 <conditions logicalGrouping="MatchAll" trackAllCaptures="false">
9 <add input="{REQUEST_FILENAME}" matchType="IsFile" negate="true" />
10 <add input="{REQUEST_FILENAME}" matchType="IsDirectory" negate="true" />
11 </conditions>
12 <serverVariables />
13 <action type="Rewrite" url="public/index.php?_url={R:1}" appendQueryString="true"
logRewrittenUrl="false" />
14 </rule>
15 </rules>
16 </rewrite>
17 </system.webServer>
18 </configuration>
```

Fuente: Basado en Skvorc (2013).

Ilustración que muestra el contenido del archivo *web.config* usado por el módulo URL Rewrite 2.0 mediante el cual se logra el funcionamiento de las rutas por defecto de Phalcon PHP.

Considerando la seguridad de los archivos que son transferidos del sistema repositorio hacia el servidor, se configuraron permisos para evitar el acceso Anónimo y directo a los archivos, ya que se deshabilitó en el servidor Web la autenticación anónima para el directorio que los contiene (Ver Ilustración 71). Con esto, si el usuario intenta descargar un archivo a través de la URL mostrará un mensaje que dirá acceso denegado, es decir, solo se podrán acceder a través de la aplicación repositorio.

Ilustración 71. Configuración autenticación anónima deshabilitada en el IIS.



Fuente: Elaboración propia.

Ilustración que muestra la consola de administración del servidor Web IIS en la función para deshabilitar el acceso anónimo a los elementos archivo que sean agregados mediante el sistema repositorio.

Una vez realizado lo anterior, fue necesario modificar el archivo de configuración de PHP, esto para habilitar las extensiones necesarias para el proyecto, tales como: `php_pdo_mysql.dll`, ya que el *framework* utiliza esta librería para el acceso a datos, `extension=php_fileinfo.dll` para verificar el formato de los archivos agregados al servidor de forma segura y por último, la librería `php_phalcon.dll`, que es la necesaria para hacer uso del *framework*.

## Capítulo V Resultados.

### 5.1 Puntos a evaluar.

Como consecuencia del desarrollo del sistema repositorio y su posterior puesta en marcha en la organización, se evaluaron los objetivos planteados en el inicio de la investigación, tal como el objetivo general, así como los específicos, obteniéndose los resultados detallados en la presente sección.

Los puntos a evaluar fueron los siguientes:

- A. “Desarrollar e implementar una solución software que permita el almacenamiento y consulta de información digitalizada de los registros de la organización, de tal forma que la información sea resguardada estructuralmente y que permita un entorno colaborativo”.
- B. Resguardo de los archivos digitales de los expedientes de forma estructurada.
- C. Permitir el intercambio de archivos dentro de la organización a través de los sistemas existentes.
- D. Interfaz gráfica que permita agregar múltiples archivos a través de una navegación sencilla y ágil para el usuario.
- E. Reducir los tiempos de acceso a la información digital de los expedientes.
- F. Proporcionar acceso integral con los demás sistemas de la organización.

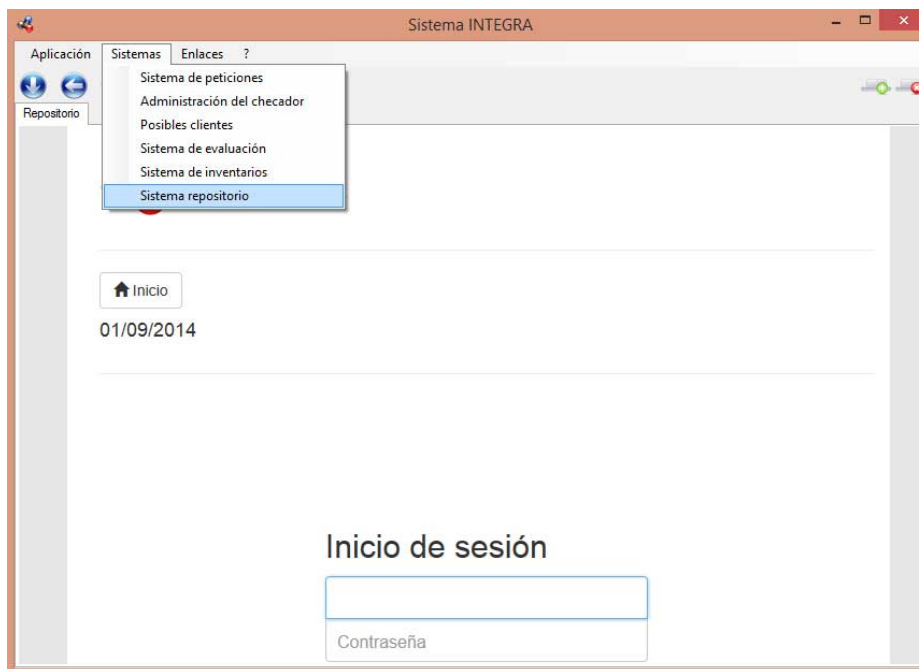
### 5.2 Resultados de las evaluaciones.

Con base a los objetivos se evaluaron los puntos de la sección anterior se obtuvieron los resultados que se exponen a continuación.

A) Se logró desarrollar y colocar en marcha el sistema repositorio a nivel de producción en la organización, el sistema se nombró “repositorio”, este mismo nombre se le otorgó al enlace mediante el

cual los usuarios pueden tener acceso. El enlace es una aplicación de escritorio la cual incorpora un navegador Web integrado, ver Ilustración 72.

**Ilustración 72. Acceso al sistema repositorio a través de la aplicación de escritorio.**



Fuente: Elaboración propia.

Ilustración que muestra la aplicación de escritorio mediante la cual el personal de la organización accede a los distintos sistemas y la cual fue añadido un enlace adicional hacia el repositorio.

Cabe mencionar que conforme a la propuesta mencionada en el punto 4.5 de la página 67, se agregaron los documentos del sistema de gestión de la calidad al repositorio, de esta forma se logro que los usuarios se familiarizaran con el nuevo sistema paulatinamente.

B) El sistema repositorio permite el resguardo de los archivos de forma de árbol jerárquico, donde un elemento que tiene la función de un nodo, puede contener varios elementos de tipo archivo u otros nodos. Por lo que es posible organizar los documentos como si se tratara de un sistema de archivos convencional, donde se puede navegar fácilmente a través de enlaces de hipertexto entre los diferentes nodos.

C) Con la funcionalidad llamada “enviar elemento a principal” y “notificar nuevo elemento” se consiguió cumplir este objetivo. A través de estas funciones, los usuarios pueden ingresar archivos al sistema y después notificar vía correo electrónico, o mediante un enlace, al personal que cuente con permiso de lectura sobre los archivos.

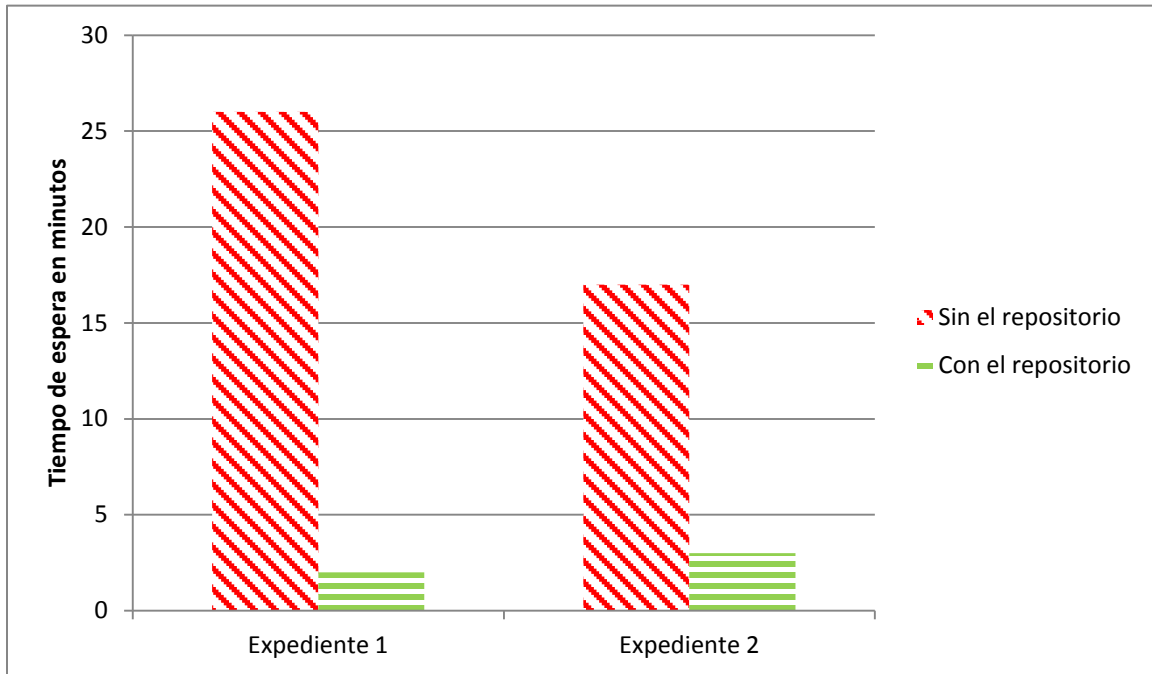
Con esto se mitiga la problemática mencionada en el punto 4.3, en el sentido de que los usuarios ahora pueden enviarse archivos de mayor tamaño, que los permitidos mediante correo electrónico convencional, además de contar con un mecanismo alternativo al uso de memorias externas USB.

D) Una de las problemáticas de mayor peso en esta investigación, era la referente a la limitación del sistema de expedientes para poder agregar más de un archivo a la documentación del expediente, por ello se agregó como objetivo particular. Como resultado se obtuvo que fue posible llevar a cabo con éxito la funcionalidad para agregar múltiples archivos, esto con la implementación del componente *Fancyupload* el cual cuenta con licencia tipo MIT (Kirschner, 2014).

E) Al brindar la posibilidad de gestionar el archivo digital de un expediente en varios de ellos, no solo se redujeron tiempos de acceso a los mismos, en algunos casos se abrió la puerta a que fueran administrados por el personal encargado, ya que anteriormente ciertos archivos superaban hasta los 4 *Gigabytes* de tamaño, por lo que su descarga y administración se tornaba complicado.

Para ejemplificar lo anterior, se tomaron medidas de tiempo para el proceso para proporcionar información digital del expediente mostrado en la Ilustración 13, se observó una mejoría en los tiempos de acceso a la información, Ver Ilustración 73, por lo que el tiempo total del proceso se redujo.

Ilustración 73 Comparación de tiempos de espera antes del repositorio.



Fuente: Elaboración propia.

Ilustración que muestra un gráfico comparativo de tiempos de acceso a la información antes de la implementación del repositorio y posterior a su puesta en marcha.

Así mismo, el poder seccionar en varios archivos, permitió crear apartados donde se resguarda contenido que es utilizado con más frecuencia de aquel que no lo es, lográndose que el usuario pueda encontrar la información intuitivamente.

F) El sistema repositorio se integró a los sistemas existentes en la organización, prueba de ello es el acceso al sistema, el cual se realiza a través de la aplicación de escritorio de la Ilustración 72. Además, en las funcionalidades para iniciar sesión en el sistema, se utilizaron las mismas variables de sesión que son utilizadas en sistemas anteriores, por lo que los usuarios pueden acceder a ambos con las mismas credenciales.

La funcionalidad llamada “cargar elemento desde sistema de expedientes” acopla de mejor forma ambos sistemas, ya que una vez cargados los archivos de los expedientes, se pueden consultar desde los

detalles de los expedientes sin tener que pasar a otra ventana o aplicación, por lo que el cambio entre un sistema y otro es transparente para el usuario.

Como se mencionó en la sección [4.5](#), se llevó a cabo la estrategia para integrar el sistema a las actividades diarias de la organización, para ello se agregó al sistema repositorio los documentos del sistema de la calidad ISO 9001 para su consulta, además, se colocaron accesos directos en la página principal del sistema de expedientes, con lo que se obtuvo que la adopción del nuevo sistema fuera transparente.

**Ilustración 74. Accesos a los documentos del ISO 9001.**



Fuente: Elaboración propia.

Ilustración que muestra la parte de la página principal del sistema de expedientes, donde se observan los enlaces a los documentos del ISO 9001.

## Capítulo VI Conclusiones y recomendaciones.

Con la implementación del sistema repositorio de documentos digitales dentro de la organización en la ciudad de Tijuana, se lograron consumir los objetivos inicialmente planteados, ya que se observaron mejoras en los tiempos de acceso a la información y por lo tanto, se disminuyeron los tiempos muertos en algunas de las operaciones relacionadas a la gestión de los documentos digitales.

El sistema repositorio, permite gestionar los documentos digitales de los expedientes en varios archivos electrónicos, dando mayor agilidad en su administración, además la integración con los sistemas y bases de datos existentes permitió una adopción transparente para los usuarios.

Como se mencionó en el alcance, esta implementación fue llevada a cabo en las instalaciones de la ciudad de Tijuana, Baja California, sin embargo al ser una organización certificada en ISO 9001 y con la finalidad de tener homologados los procesos, la implementación en las demás ubicaciones es inminente, para ello, será necesario verificar que se cuenta con la infraestructura de TI requerida para la ejecución del sistema, la cual se plasma en el ANEXO 3, así como las configuraciones necesarias del punto 4.7.1.

Además de la implementación hacia otras regiones dentro de la organización, es una posibilidad la apertura del sistema hacia entidades externas como clientes, ya que podría lograrse el intercambio de archivos de gran tamaño en donde el correo electrónico convencional no lo permite, con lo que se lograría el uso de infraestructura propia. Para esto, es posible que el sistema requiera adecuaciones en los controladores de visualización de los elementos para agregar mecanismos de seguridad, pues serían vistos por personas que no cuentan con credenciales registradas en el sistema.

Aun cuando dentro de las funcionalidades se integró la capacidad de buscar elementos dentro del repositorio, tal vez en el futuro se requiera de un sistema independiente para la indización de elementos, puesto que este tipo de sistemas ofrecen mayor rendimiento frente a las funciones incorporadas del

motor de base de datos utilizado, en este sentido el sistema Sphinx se integra muy bien con MySQL (Galbraith, 2009), por lo que sería una opción a tomar en cuenta.

## Bibliografía.

- Adam, A. (2008). *Implementing electronic document and record management systems*. Taylor & Francis Group, LLC.
- Adame, S. I., Llórens, L., & Schorr, M. (2013). Retrospectiva de los repositorios de acceso abierto y tendencias en la socialización del conocimiento. *Revista Electrónica de Investigación Educativa*, 148-162.
- Albalat, J. A. (2012). *Gestión documental*. Obtenido de Slideshare: <http://es.slideshare.net/josealbalat/gestion-documental-14736045>
- Alonso, J. A., García, M., & Lloveras, M. R. (Diciembre de 2007). *La norma ISO 15489: un marco sistemático de buenas prácticas de gestión documental en las organizaciones*. Obtenido de <http://eprints.rclis.org/12263/>
- Anglada I. de Ferrer, L., Reoyo Tundo, S., Cambras, J., & De La Vega, R. (Junio de 2008). *Los repositorios electrónicos cooperativos de la Biblioteca Digital de Cataluña*. Obtenido de Portal de la Facultad de Biblioteconomía y documentación : <http://bid.ub.edu/20angla2.htm>
- Arriola Navarrete, Ó., & Butrón Yáñez, K. (2008). *Sistemas integrales para la automatización de bibliotecas basados en software libre: Una oportunidad para la automatización de bibliotecas*. Acimed.
- Axzo Press. (2011). *Dreamweaver CS5*. Axzo Press.
- Benítez, C. A. (2013). *Sistemas web escalables*. Buenos Aires: Dalaga; Fox Andina.
- Boletín Oficial del Estado. (2007). *Ley de Acceso Electrónico de los Ciudadanos a los Servicios Públicos*. Obtenido de LAECSP: [http://www.boe.es/diario\\_boe/txt.php?id=BOE-A-2007-12352](http://www.boe.es/diario_boe/txt.php?id=BOE-A-2007-12352)
- Boté Vericad, J.-J. (2012). *Propuesta de un modelo de preservación digital para pequeñas y medianas instituciones sanitarias*. Barcelona.
- Bruce, B., Ray, J., & Ness, R. (2011). *Teach yourself Adobe Dreamweaver CS5 in 24 hours*. Pearson Education.
- Buckler, C. (Agosto de 2012). *jQuery Is Used on Half of All Websites*. Recuperado el Febrero de 2014, de Site Point: <http://www.sitepoint.com/jquery-used-on-50-percent-of-websites/>
- Cáceres, P., & Marcos, E. (2001). *Procesos Ágiles para el Desarrollo de Aplicaciones Web. I Taller sobre Ingeniería del Software Orientada al Web*. Almagro (Ciudad real).

- Caruana, D., Newton, J., Farman, M., Uzquiano, M. G., & Roast, K. (2010). *Alfresco, Practical Solutions for Enterprise Content Management*. Wiley Publishing, Inc.
- Cervantes de la Fuente, J. Á. (2014). *Tesis organización de documentos en español, digitalizados y semánticamente relacionados*. Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional.
- Cohen Karen, D., & Asín Lares, E. (2000). *Sistemas de información para los negocios, Un enfoque de toma de decisiones*. México, D.F.: McGraw-Hill.
- Collell Puig, J. (2014). *CSS3 y Javascript Avanzado*. FUOC. Fundació per a la Universitat Oberta de Catalunya.
- Collins, M., & Grogg, J. (1 de Marzo de 2011). Building A Better ERMS. *Library Journal*, 22-28.
- Concepción Báez, C. (2008). Wiki, intranet y repositorios documentales. *ACIMED - Centro Nacional de Informacion de Ciencias Medicas*.
- Cruz Mundet, J. R. (2011). *Administración de documentos y archivos: Textos fundamentales*. Madrid.
- D'Alòs-Moner, A. (Mayo-Junio de 2006). La gestión documental: aspectos previos a su implementación. *El profesional de la información*, 222-226.
- D'Alòs-Moner, A. (Agosto de 2011). Gestión de documentos: sonrisas y lágrimas. *El profesional de la información*, 456-459.
- David, J. (Enero de 2012). *5 Razones para Tener una Aplicación Web Propia en Mi empresa*. Obtenido de <http://blog.rhiss.net/5-razones-para-tener-una-aplicacion-web-propia-en-mi-empresa.html>
- De la Vega, R. (2008). Software libre en repositorios de e-información. *El Profesional de la Información*, v.17, n. 1, pp. 49-55.
- De Luca, D. (2011). *HTML 5*. Buenos Aires: Fox Andina.
- Driscoll, M. (2011). *Node.js and the JavaScript Age*. Recuperado el Noviembre de 2013, de <http://gigaom.com/cloud/node-js-and-the-javascript-age>
- Eguíluz Pérez, J. (2008). *Introducción a Ajax*. librosweb.es.
- ENISA. (2009). *Computación en nube. Beneficios, riesgos y recomendaciones para la seguridad de la información*. Agencia Europea de Seguridad de las Redes y de la Información ENISA.
- Escobar, S. (2006). *La información: un recurso valioso para su empresa*. Obtenido de Slideshare: <http://www.slideshare.net/sergioges/la-informacin-un-recurso-valioso-para-su-empresa>
- Ferreras Fernández, T. (2010). *Preservación digital en repositorios institucionales*. Salamanca.

- Font Aranda, O., Ruiz Rodríguez, A., & Mena Mugica, M. (2012). Diagnóstico sobre la gestión documental y de archivos en la Universidad Central Marta Abreu de las Villas. Cuba: Caso de estudio. . *Revista española de Documentación Científica*, 573-598.
- Fontela, C. (2012). *UML Modelado de Software para Profesionales*. Alfaomega.
- Free Software Foundation. (2014). *Preguntas frecuentes acerca de las licencias de GNU*. Obtenido de GNU.org: <http://www.gnu.org/licenses/gpl-faq.es.html#GPLRequireSourcePostedPublic>
- Fresno Chávez, C. (2012). Bibliotecas o bibliotecas digitales con Greenstone. Obtenido de <http://www.greenstone-la.org/recursos-sobre-greenstone/95-bibliotecas-o-repositorios-digitales-con-greenstone>
- Galbraith, P. (2009). *Developing Web Applications with Apache, MySQL, memcached, and Perl*. Wiley Publishing, Inc.
- García, F. (Mayo de 2011). *Modelado Conceptual de Aplicaciones Web*. Obtenido de Slideshare: <http://www.slideshare.net/grialusal/modelado-conceptual-de-aplicaciones-web>
- García-Morales, E. (2013). *Gestión de documentos en la e-administración*. El profesional de la información.
- Giralt, O., Vidal-Pijoan, C., & Pérez-Soler, C. (2011). Seguridad de los documentos de archivo: estudio de caso del Archivo del Ayuntamiento de Barcelona. *El profesional de la información*, 202-205.
- Goitia, S., Sáenz-de-Lacuesta, S., & Bilbao, M. (2008). Implantación de sistemas de información empresarial. (17), 540-545.
- Goodyear, S. (2013). *Practical Sharepoint 2013 Enterprise Content Manager*. Apress.
- Gutiérrez, F., Penichet, V., Isla, J., Montero, F., Lozano, M., Gallud, J., & Rodríguez, M. (Enero de 2006). Un Marco Conceptual para el Modelado de Sistemas Colaborativos Empresariales. España.
- Guzel, B. (30 de Marzo de 2011). *Top 15+ Best Practices for Writing Super Readable Code*. Obtenido de <http://code.tutsplus.com/tutorials/top-15-best-practices-for-writing-super-readable-code--net-8118>
- ISO, S. C. (2008). *Norma Internacional ISO 9001, traducción oficial*. Ginebra, Suiza.
- Kholia, D., & Przemysław, W. (2013). *Looking Inside the (Drop) Box*. Obtenido de USENIX Advanced Computing Systems Association: <https://www.usenix.org/conference/woot13/workshop-program/presentation/kholia>
- Kirschner, H. (Abril de 2014). *FancyUpload - Swiff meets Ajax (v3.0)*. Obtenido de digitald.de: <http://digitald.de/project/fancyupload/>
- Minera, F. (2007). *Ajax*. Gradi S.A.

- Minera, F. (2010). *PHP 6*. Argentina: Gradi; Foz Andina.
- Novelle López, L. (2012). *Archivos de empresa, nociones básicas sobre historia, contenido y gestión*.
- OpenDOAR. (Octubre de 2013). *Growth of the OpenDOAR Database - Worldwide*. Obtenido de The Directory of Open Access Repositories - OpenDOAR:  
<http://www.opendoar.org/onechart.php?clID=&ctID=&rtID=&clID=&IID=&potID=&rSoftWareName=&search=&groupby=r.rDateAdded&orderby=&charttype=growth&width=600&height=350&caption=Growth%20of%20the%20OpenDOAR%20Database%20-%20Worldwide>
- OpenDOAR. (Octubre de 2014). *Usage of Open Access Repository Software - Worldwide*. Obtenido de OpenDOAR:  
<http://opendoar.org/onechart.php?clID=&ctID=&rtID=&clID=&IID=&potID=&rSoftWareName=&search=&groupby=r.rSoftWareName&orderby=Tally%20DESC&charttype=pie&width=600&height=300&caption=Usage%20of%20Open%20Access%20Repository%20Software%20-%20Worldwide>
- Osuna Alarcón, M., & De La Cruz Gómez, E. (Junio de 2010). Los sistemas de gestión de contenidos en Información y Documentación.
- Page, A., Jhonston, k., & Rollison, B. (2009). *How we test software at Microsoft*. Microsoft Press.
- Palma, M. d. (Diciembre de 2008). *La especificación MoReq2*. Recuperado el Noviembre de 2013, de Slideshare: <http://www.slideshare.net/radarik/la-especificacion-mo-req2-modelo-de-requisitos-para-erms>
- Peis Arredondo, E. (2004). *Página personal Eduardo Peis*. Recuperado el Noviembre de 2013, de Sitio web de Universidad de Granada: [www.ugr.es/~epeis/docencia/archivistica/tema7.ppt](http://www.ugr.es/~epeis/docencia/archivistica/tema7.ppt)
- Poratti, G. (2004). *Windows Server 2003*. Buenos Aires: MP Ediciones.
- Pressman, R. (2010). *Ingeniería de software un enfoque práctico*. México: McGraw-Hill.
- Ranking Web Of Repositories*. (2014). Recuperado el 20 de 11 de 2013, de [repositories.webometrics.info](http://repositories.webometrics.info)
- Read, J., & Lea Ginn, M. (2011). *Records Management*. South-Western, Cengage Learning.
- Rodríguez, D. (2013). *La empresa sin papeles. Guía para trabajar con documentos electrónicos con Acrobat XI Pro*. España: Alfaomega.
- San Martín, E. (Abril de 2007). *MVC - Modelo Vista Controlador*. Recuperado el Marzo de 2014, de ProgramacionWeb.net: <http://www.programacionweb.net/articulos/articulo/mvc-modelo-vista-controlador/>
- Schwartz, B., Zaitsev, P., & Tkachenko, V. (2012). *High Performance MySQL*. O'Reilly.

Sitio Web dotProject. (2004). *dotproject Open Source Project and Task Management Software*. Obtenido de <http://www.dotproject.net/>

Sitio Web EcuRed. (24 de Noviembre de 2011). Recuperado el 20 de Noviembre de 2013, de DIRKS: <http://www.ecured.cu/index.php/DIRKS>

Sitio Web EcuRed. (2013). Recuperado el Octubre de 2014, de Repositorio: <http://www.ecured.cu/index.php/Repositorio>

Sitio Web Grupo Farías. (2014). Obtenido de Certificados en servicios de defensa fiscal: <http://www.grupofarias.com/esp/iso.php>

Sitio Web MoReq2. (Octubre de 2013). *MoReq2*. Obtenido de [www.moreq2.eu](http://www.moreq2.eu)

Sitio Web Phalcon PHP. (Abril de 2014). Obtenido de Tutorial 1: Aprendamos usando ejemplos: <http://docs.phalconphp.com/es/latest/reference/tutorial.html#estructura-de-directorios>

Sitio Web TechEmpower. (Febrero de 2014). *Framework Performance Comparison*. Obtenido de TechEmpower Web: <http://www.techempower.com/benchmarks/#section=data-r6&hw=i7&test=query&l=sg>

Sitio Web USAL. (Octubre de 2013). *Portal bibliotecario de USAL*. Recuperado el 18 de Noviembre de 2013, de Portal de información bibliográfica de la Universidad de Salamanca: [http://bibliotecabiologia.usal.es/tutoriales/catalogos-repositorios-bibliosvirtuales/repositorios\\_digitaes.html](http://bibliotecabiologia.usal.es/tutoriales/catalogos-repositorios-bibliosvirtuales/repositorios_digitaes.html)

Sitio Web Web Sequence Diagrams. (2014). Obtenido de WebSequenceDiagrams.com - Draw and Edit Sequence Diagrams in seconds: [www.websequencediagrams.com](http://www.websequencediagrams.com)

Skvorc, B. (Diciembre de 2013). *Phalcon on Windows Azure*. Recuperado el Febrero de 2014, de Sitepoint: <http://www.sitepoint.com/phalcon-windows-azure/>

*The ISO Survey of Management System Standard*. (2013). Recuperado el 27 de Marzo de 2014, de Sitio Web ISO.org: <http://www.iso.org/iso/home/standards/certification/iso-survey.htm?certificate=ISO%209001&countrycode=AF>

UNE-ISO. (2005). Proyecto UNE-ISO 15489/1. Información y documentación. Gestión de documentos. Parte 1: Generalidades. *Revista Española de Documentación Científica*, 28(1).

Vázquez, P. (2006). *Creación de sitios Web*. Usershop.

Volodarsky, M. (2007). *Explore el servidor web de Windows Vista y más allá*. Recuperado el 28 de Noviembre de 2013, de <http://msdn.microsoft.com/es-es/magazine/cc163453.aspx>

W3Techs. (2013). *Usage of JavaScript libraries for websites*. Obtenido de W3Techs - extensive and reliable web technology surveys: [http://w3techs.com/technologies/overview/javascript\\_library/all](http://w3techs.com/technologies/overview/javascript_library/all)

W3Techs. (2013). *Usage of server-side programming languages for websites*. Obtenido de W3Techs - extensive and reliable web technology surveys:  
[http://w3techs.com/technologies/overview/programming\\_language/all](http://w3techs.com/technologies/overview/programming_language/all)

Wen, H., Ye, D., & Lin, B. (1998). Intranet document management systems. *Internet Research: Electronic Networking Applications and Policy*.

Zacker, C. (2001). *Redes. Manual de referencia*. Madrid: McGraw-hill / Interamericana de España.

## ANEXO 1. Cronograma de actividades.

El cronograma se encuentra estructurado en dos grandes etapas, la primera referente a la concepción del proyecto y la segunda al proceso de desarrollo del software, en éste último, se mencionan las tres fases del desarrollo. La primer fase se refiere al primer incremento representada por el módulo de seguridad, la segunda por las funciones administrativas y por último las funciones colaborativas.

ID	Task Mode	Task Name	Duration	Start	Finish	Predecessors
1		Sistema repositorio de información colaborativo	296 days	Mon 03/06/13	Mon 21/07/14	
2		Anteproyecto	133 days	Mon 03/06/13	Wed 04/12/13	
3		Selección del proyecto	1 day	Mon 03/06/13	Mon 03/06/13	
4		Objetivos	1 day	Tue 04/06/13	Tue 04/06/13	3
5		Justificación	1 day	Wed 05/06/13	Wed 05/06/13	4
6		Investigación preliminar	5 days	Tue 04/06/13	Mon 10/06/13	3
7		Cronograma	2 days	Tue 11/06/13	Wed 12/06/13	6
8		Presentación de protocolo	1 day	Wed 23/10/13	Wed 23/10/13	
9		Marco teórico	30 days	Thu 24/10/13	Wed 04/12/13	8
10		Proyecto desarrollo de software	106 days	Mon 24/02/14	Mon 21/07/14	
11		Fase Uno - Módulo de seguridad	30 days	Mon 24/02/14	Fri 04/04/14	
12		Análisis de las actividades	5 days	Mon 24/02/14	Fri 28/02/14	
13		Análisis de procesos de la organización	5 days	Mon 24/02/14	Fri 28/02/14	9
14		Identificación de los requisitos	3 days	Mon 03/03/14	Wed 05/03/14	13
15		Identificación de las estrategias para cumplir con los requisitos	3 days	Thu 06/03/14	Mon 10/03/14	14
16		Diseño y Desarrollo del sistema de gestión de documentos	16 days	Tue 11/03/14	Tue 01/04/14	
17		Modelado de la base de datos	3 days	Tue 11/03/14	Thu 13/03/14	15
18		Inicio de sesión	3 days	Fri 14/03/14	Tue 18/03/14	17
19		Pantalla principal y navegación	7 days	Wed 19/03/14	Thu 27/03/14	18
20		Cierre de sesión	3 days	Fri 28/03/14	Tue 01/04/14	19
21		Implementación del sistema de gestión de documentos	1 day	Wed 02/04/14	Wed 02/04/14	
22		Puesta en marcha de la primera fase en producción (servidor)	1 day	Wed 02/04/14	Wed 02/04/14	20
23		Revisión posterior a la implementación	2 days	Thu 03/04/14	Fri 04/04/14	
24		Pruebas al modulo de seguridad	2 days	Thu 03/04/14	Fri 04/04/14	22
25		Fase Dos Módulo de administracion	43 days	Mon 07/04/14	Wed 04/06/14	
26		Análisis de las actividades	2 days	Mon 07/04/14	Tue 08/04/14	
27		Escenarios y entrevistas	2 days	Mon 07/04/14	Tue 08/04/14	11
28		Identificación de los requisitos	2 days	Wed 09/04/14	Thu 10/04/14	27
29		Identificación de las estrategias para cumplir con los requisitos	2 days	Fri 11/04/14	Mon 14/04/14	28
30		Diseño y Desarrollo del sistema de gestión de documentos	32 days	Tue 15/04/14	Wed 28/05/14	
31		Crear nuevo elemento en el repositorio	5 days	Tue 15/04/14	Mon 21/04/14	29
32		Mostrar contenido de un elemento	5 days	Tue 22/04/14	Mon 28/04/14	31
33		Selección de elementos	5 days	Tue 29/04/14	Mon 05/05/14	32
34		Mover elemento	5 days	Tue 06/05/14	Mon 12/05/14	33
35		Borrar elemento	2 days	Tue 13/05/14	Wed 14/05/14	34
36		Cambiar nombre a elemento	4 days	Thu 15/05/14	Tue 20/05/14	35
37		Cambiar permisos a elemento	6 days	Wed 21/05/14	Wed 28/05/14	36
38		Implementación del sistema de gestión de documentos	1 day	Thu 29/05/14	Thu 29/05/14	37
39		Puesta en marcha de la segunda fase en producción (servidor)	1 day	Thu 29/05/14	Thu 29/05/14	37
40		Revisión posterior a la implementación	2 days	Fri 30/05/14	Mon 02/06/14	
41		Pruebas al modulo de administración	2 days	Fri 30/05/14	Mon 02/06/14	39
42		Capacitación	2 days	Tue 03/06/14	Wed 04/06/14	41
43		Fase Tres Módulo de herramientas colaborativas	32 days	Thu 05/06/14	Fri 18/07/14	
44		Análisis de las actividades	3 days	Thu 05/06/14	Mon 09/06/14	
45		Escenarios y entrevistas	3 days	Thu 05/06/14	Mon 09/06/14	25
46		Identificación de los requisitos	2 days	Tue 10/06/14	Wed 11/06/14	45
47		Identificación de las estrategias para cumplir con los requisitos	2 days	Thu 12/06/14	Fri 13/06/14	46
48		Diseño y Desarrollo del sistema de gestión de documentos	20 days	Mon 16/06/14	Fri 11/07/14	
49		Notificar via mail de nuevo elemento a usuarios	5 days	Mon 16/06/14	Fri 20/06/14	47
50		Enviar un elemento a la pantalla principal de un usuario	5 days	Mon 23/06/14	Fri 27/06/14	49
51		Carga de elementos desde el sistema de expedientes	10 days	Mon 30/06/14	Fri 11/07/14	50
52		Implementación del sistema de gestión de documentos	1 day	Mon 14/07/14	Mon 14/07/14	
53		Puesta en marcha de la tercera fase en producción (servidor)	1 day	Mon 14/07/14	Mon 14/07/14	51
54		Revisión posterior a la implementación	2 days	Tue 15/07/14	Wed 16/07/14	
55		Pruebas al modulo de herramientas colaborativas	2 days	Tue 15/07/14	Wed 16/07/14	53
56		Capacitación	2 days	Thu 17/07/14	Fri 18/07/14	55
57		Entrega de documentación	1 day	Mon 21/07/14	Mon 21/07/14	56

Ilustración A1. Cronograma de actividades.

## ANEXO 2. Diseños de caso de prueba.

En esta sección se muestran los casos de prueba ejecutados durante la fase de pruebas.

Numero de caso de prueba: 000			
Título: Autenticación correcta			
Área		Subarea	
Inicio de sesión		Fase 1	
Prioridad	Tipo	Frecuencia	Tiempo estimado
1	Funcionalidad	Milestone	2min
<b>Descripción:</b> El controlador iniciar sesión debe permitir escribir un usuario y contraseña y validar que dicha información sea la registrada en la base de datos, posteriormente dirigir al usuario al contenido del sistema.			
<b>Propósito de la prueba:</b> Verificar que se pueda realizar el inicio de sesión con credenciales registradas en la base de datos.			
<b>Condiciones iniciales:</b> La prueba debe realizarse con una cuenta de usuario registrada.			
<b>Pasos:</b> <ol style="list-style-type: none"><li>1. Abra el navegador de internet</li><li>2. En la barra de direcciones escriba <code>http://&lt;&lt;servidor&gt;&gt;/repositorio/index/</code>.</li><li>3. Escriba el usuario "yo" y contraseña "1234".</li><li>4. Presione el botón iniciar sesión.</li><li>5. Escriba el usuario "admin" y contraseña "1234".</li></ol>			
<b>Resultados esperados:</b> <ul style="list-style-type: none"><li>• El sistema debe mostrar un mensaje de error de inicio de sesión al proporcionar las credenciales "yo" y "1234".</li><li>• Una vez proporcionadas las correctas se debe enviar al usuario a la página principal.</li></ul>			
<b>Notas:</b>			

Numero de caso de prueba: 001			
Título: Solicitar inicio de sesión para obtener elementos			
Área		Subarea	
Inicio de sesión		Fase 1	
Prioridad	Tipo	Frecuencia	Tiempo estimado
1	Funcionalidad	Por controlador	2min
<b>Descripción:</b> En cada controlador deberá solicitar inicio de sesión si no existe una sesión válida cuando el usuario solicite obtener elementos directamente por la URL.			
<b>Propósito de la prueba:</b> Verificar que el sistema valide el inicio de sesión en cada controlador.			
<b>Condiciones iniciales:</b> La prueba debe realizarse con una cuenta de usuario registrada.			
<b>Pasos:</b> <ol style="list-style-type: none"><li>1. Abra el navegador de internet</li><li>2. En la barra de direcciones escriba <code>http://&lt;&lt;servidor&gt;&gt;/repositorio/principal/index/100</code>.</li><li>3. Escriba usuario "admin" y contraseña "1234".</li><li>4. Presione el botón iniciar sesión.</li></ol>			
<b>Resultados esperados:</b>			

- El sistema debe redirigirlo a iniciar sesión al solicitar el elemento.
- Una vez iniciado sesión el sistema debe mostrar el contenido de elemento solicitado.

Notas:

Numero de caso de prueba: 002

Título: Cierre de sesión efectivo

Área	Subarea		
Cierre de sesión	Fase 1		
Prioridad	Tipo	Frecuencia	Tiempo estimado
1	Funcionalidad	Milestone	2min

**Descripción:** El botón cerrar sesión debe cumplir la función de destruir la sesión activa de PHP y evitar que otro usuario que llegue a usar la computadora la retome.

**Propósito de la prueba:**

Verificar que el botón de cerrar sesión elimine la sesión PHP.

**Condiciones iniciales:**

Debe iniciar sesión.

**Pasos:**

1. Abra el navegador de internet.
2. Ingrese la dirección del sistema.
3. Inicie sesión con el nombre de usuario "admin" y contraseña "1234".
4. Navegue dentro del sistema a una carpeta y copie la dirección url al portapapeles.
5. Presione el botón cerrar sesión.
6. Pegue la dirección url anteriormente copiada y presione Ir.

**Resultados esperados:**

- El sistema debe redirigirlo a iniciar sesión al solicitar la url pegada.

Notas:

Numero de caso de prueba: 003

Título: No recuerdo mi contraseña

Área	Subarea		
Recordar contraseña	Fase 1		
Prioridad	Tipo	Frecuencia	Tiempo estimado
1	Funcionalidad	Milestone	2min

**Descripción:** La función de recordar contraseña ayuda a los usuarios a obtener su nombre de usuario y contraseña.

**Propósito de la prueba:**

Verificar que el usuario reciba correctamente su usuario y contraseña.

**Condiciones iniciales:**

Debe ser un usuario previamente registrado.

**Pasos:**

1. Abra el navegador de internet y dirijase a la url del sistema.
2. Presione en el enlace Recordar inicio de sesión.
3. Escriba el correo electrónico registrado en el sistema para la cuenta "admin" (sistema@grupofarias.com), en el campo Correo.
4. Presione el botón Recordar Contraseña.

**Resultados esperados:**

- Debe llegar un correo electrónico a la cuenta del usuario con su usuario y contraseña.

Notas:

Numero de caso de prueba: 004			
Título: Crear muchas carpetas			
Área		Subarea	
Crear carpeta		Fase 2	
Prioridad	Tipo	Frecuencia	Tiempo estimado
1	Funcionalidad	Milestone	15min
<b>Descripción:</b> Crear carpetas dentro del sistema.			
<b>Propósito de la prueba:</b> Verificar que el controlador funcione correctamente en el momento de crear varias carpetas consecutivamente.			
<b>Condiciones iniciales:</b> El usuario debe iniciar sesión.			
<b>Pasos:</b>			
1. Abra el navegador de internet y dirijase a la url del sistema.			
2. Inicie sesión con usuario: "admin" y contraseña "1234".			
3. Presione el botón crear carpeta.			
4. Proporcione un nombre válido.			
5. Presione el botón aceptar.			
6. Repita el paso 4 y 5 por un total de 10 veces.			
<b>Resultados esperados:</b>			
<ul style="list-style-type: none"> <li>El sistema debe permitir crear las carpetas y mostrarlas correctamente en la vista.</li> </ul>			
<b>Notas:</b>			

Numero de caso de prueba: 005			
Título: Crear un archivo en el sistema			
Área		Subarea	
Crear archivo		Fase 2	
Prioridad	Tipo	Frecuencia	Tiempo estimado
1	Funcionalidad	Milestone	~5min
<b>Descripción:</b> El botón crear archivo debe permitir transferir archivos de la computadora del usuario hacia el servidor y reflejarse en la vista del elemento actual.			
<b>Propósito de la prueba:</b> Verificar que el botón de crear archivo permita subir varios archivos.			
<b>Condiciones iniciales:</b> El usuario debe iniciar sesión en el sistema. El usuario debe estar sobre una carpeta con permiso de modificar.			
<b>Pasos:</b>			
1. Abra el navegador de internet.			
2. Ingrese la dirección del sistema.			
3. Inicie sesión con el usuario "admin" y contraseña "1234".			
4. Presione el botón crear archivo.			
5. Seleccione tres archivos de entre 5 y 10mb cada uno.			
6. Presione el enlace Cargar archivos.			
<b>Resultados esperados:</b>			
<ul style="list-style-type: none"> <li>El sistema debe mostrar la cantidad total de kb a transferir.</li> <li>El sistema debe transferir los archivos al servidor.</li> <li>Al terminar de transferir el sistema debe obtener (refrescar) los elementos de la posición actual.</li> </ul>			
<b>Notas:</b>			

Numero de caso de prueba: 006			
Titulo: Intentar engañar al sistema con archivos de extensiones no permitidas			
Área		Subarea	
Crear archivo		Fase 2	
Prioridad	Tipo	Frecuencia	Tiempo estimado
2	Funcionalidad	Milestone	~5min
<b>Descripción:</b> En controlador debe validar que no se puedan crear ciertos tipos de archivos.			
<b>Propósito de la prueba:</b> Verificar que al cargar archivo se valide su tipo de extensión y se muestre error en caso de que sea uno no permitido.			
<b>Condiciones iniciales:</b> El usuario debe iniciar sesión en el sistema.			
<b>Pasos:</b>			
1. Abra el navegador de internet.			
2. Ingrese la dirección del sistema.			
3. Inicie sesión con el usuario "admin" y contraseña "1234".			
4. Presione el botón crear archivo.			
5. Seleccione dos archivos de formato pdf, en su computadora, cambie la extensión de un archivo ejecutable (.exe) a extensión pdf y selecciónelo.			
6. Presione el enlace Cargar archivos.			
<b>Resultados esperados:</b>			
<ul style="list-style-type: none"> <li>• El sistema debe mostrar la cantidad total de kb a transferir.</li> <li>• Al terminar la transferencia, el sistema debe marcar error para el archivo que se cambió la extensión.</li> </ul>			
<b>Notas:</b>			

Numero de caso de prueba: 007			
Titulo: Obtener elementos			
Área		Subarea	
Obtener elementos		Fase 2	
Prioridad	Tipo	Frecuencia	Tiempo estimado
1	Funcionalidad	Milestone	2min
<b>Descripción:</b> Permitir obtener elementos relacionados a una carpeta.			
<b>Propósito de la prueba:</b> Verificar que la navegación dentro de los elementos carpeta sea mostrada correctamente en la vista por el controlador.			
<b>Condiciones iniciales:</b> El usuario debe iniciar sesión en el sistema.			
<b>Pasos:</b>			
1. Abra el navegador de internet.			
2. Ingrese la dirección del sistema.			
3. Inicie sesión con el nombre de usuario "admin" y contraseña "1234".			
4. Haga clic sobre un elemento carpeta (si no hay carpetas cree una y añada archivos).			
5. Haga clic sobre el botón Inicio del menú de acción.			
<b>Resultados esperados:</b>			
<ul style="list-style-type: none"> <li>• El sistema debe mostrar los elementos relacionados a la carpeta.</li> <li>• El sistema debe mostrar los elementos de la raíz.</li> </ul>			
<b>Notas:</b>			

Numero de caso de prueba: 008				
Título: Seleccionar y deseleccionar elementos del sistema hace que cambie el menú				
Área		Subarea		
Selección de elementos		Fase 2		
Prioridad	Tipo	Frecuencia	Tiempo estimado	
1	Funcionalidad	Milestone	2min	
<b>Descripción:</b> Permitir seleccionar elementos y observar cómo cambia el menú.				
<b>Propósito de la prueba:</b> Verificar que el menú cambie de acuerdo al número de elementos seleccionado diferenciando si el usuario tiene seleccionado uno o muchos.				
<b>Condiciones iniciales:</b> El usuario debe iniciar sesión en el sistema.				
<b>Pasos:</b>				
1. Abra el navegador de internet.				
2. Ingrese la dirección del sistema.				
3. Inicie sesión con el nombre de usuario "admin" y contraseña "1234".				
4. Haga clic sobre la casilla de selección de un elemento.				
5. Haga clic sobre la casilla de selección de otro elemento.				
<b>Resultados esperados:</b>				
<ul style="list-style-type: none"> <li>El sistema debe marcar con un sombreado los elementos seleccionados.</li> <li>El sistema debe cambiar el menú de acción en la parte superior al momento de seleccionar dos elementos.</li> </ul>				
<b>Notas:</b>				

Numero de caso de prueba: 009				
Título: Mover elementos a otra ubicación				
Área		Subarea		
Mover elementos		Fase 2		
Prioridad	Tipo	Frecuencia	Tiempo estimado	
1	Funcionalidad	Milestone	2min	
<b>Descripción:</b> Permitir seleccionar elementos y moverlos a otra carpeta.				
<b>Propósito de la prueba:</b> Verificar que el sistema permita mover dos elementos a otra ubicación.				
<b>Condiciones iniciales:</b> El usuario debe iniciar sesión en el sistema.				
<b>Pasos:</b>				
1. Abra el navegador de internet.				
2. Ingrese la dirección del sistema.				
3. Inicie sesión con el nombre de usuario "admin" y contraseña "1234".				
4. Cree dos elementos carpeta llamados Carpeta 1 y Carpeta 2.				
5. Vaya a Carpeta 1.				
6. Cree un elemento carpeta llamado Hijo en Carpeta 1				
7. Cree un archivo dentro de Carpeta 1.				
8. Seleccione Hijo y el archivo recién creados.				
9. Haga clic en Mover y seleccione el destino Carpeta 2.				
10. Haga clic en ir a ubicación.				
<b>Resultados esperados:</b>				
<ul style="list-style-type: none"> <li>El sistema debe mostrar en Carpeta 2 los elementos Hijo y el archivo.</li> </ul>				
<b>Notas:</b>				

Numero de caso de prueba: 010				
Título: Eliminar un elemento del sistema				
Área		Subarea		
Borrar elementos		Fase 2		
Prioridad	Tipo	Frecuencia	Tiempo estimado	
1	Funcionalidad	Milestone	2min	
<b>Descripción:</b> El sistema debe permitir eliminar elementos del sistema.				
<b>Propósito de la prueba:</b> Verificar que el sistema permita borrar los elementos creados en la Prueba 008.				
<b>Condiciones iniciales:</b> El usuario debe iniciar sesión en el sistema.				
<b>Pasos:</b>				
1. Abra el navegador de internet.				
2. Ingrese la dirección del sistema.				
3. Inicie sesión con el nombre de usuario "admin" y contraseña "1234".				
4. Seleccione el elemento Carpeta 2.				
5. Seleccione el elemento Hijo.				
6. Haga clic sobre el botón Eliminar y confirme la eliminación proporcionando su contraseña.				
7. Vaya a la Raíz y seleccione Carpeta 2.				
8. Haga clic sobre Eliminar y confirme.				
<b>Resultados esperados:</b>				
<ul style="list-style-type: none"> <li>• El sistema debe pedir confirmación antes de eliminar los elementos.</li> <li>• El sistema debe actualizar la ubicación actual sin los elementos eliminados.</li> </ul>				
<b>Notas:</b>				

Numero de caso de prueba: 011				
Título: Cambiar nombre a un elemento				
Área		Subarea		
Cambiar nombre a elemento		Fase 2		
Prioridad	Tipo	Frecuencia	Tiempo estimado	
1	Funcionalidad	Milestone	2min	
<b>Descripción:</b> El sistema debe permitir cambiar nombre a un elemento carpeta o archivo.				
<b>Propósito de la prueba:</b> Verificar que el sistema permita cambiar nombre a una carpeta.				
<b>Condiciones iniciales:</b> El usuario debe iniciar sesión en el sistema.				
<b>Pasos:</b>				
1. Abra el navegador de internet.				
2. Ingrese la dirección del sistema.				
3. Inicie sesión con el nombre de usuario "admin" y contraseña "1234".				
4. Cree una carpeta llamada CarpetaX.				
5. Seleccione CarpetaX.				
6. Haga clic en Cambiar nombre.				
7. Proporcione el nombre Carpeta100 y de clic en Aceptar.				
<b>Resultados esperados:</b>				
<ul style="list-style-type: none"> <li>• El sistema debe actualizar los elementos de la ubicación actual y mostrar el elemento con el nuevo nombre.</li> </ul>				
<b>Notas:</b>				

Numero de caso de prueba: 012				
Título: Cambiar permisos				
Área		Subarea		
Cambiar permisos		Fase 2		
Prioridad	Tipo	Frecuencia	Tiempo estimado	
1	Funcionalidad	Milestone	2min	
<p><b>Descripción:</b> El sistema debe permitir cambiar los permisos efectivos que tiene un usuario sobre un elemento carpeta o archivo.</p> <p><b>Propósito de la prueba:</b> Verificar que el sistema permita cambiar los permisos de un elemento carpeta.</p> <p><b>Condiciones iniciales:</b> El usuario debe iniciar sesión en el sistema. Debe haber creado dos usuarios llamados Usuario1 y Usuario2.</p> <p><b>Pasos:</b></p> <ol style="list-style-type: none"> <li>1. Abra el navegador de internet.</li> <li>2. Ingrese la dirección del sistema.</li> <li>3. Inicie sesión con el usuario1 y contraseña 1234.</li> <li>4. Cree una carpeta llamada MiCarpeta dentro de la carpeta "Personales".</li> <li>5. Seleccione MiCarpeta.</li> <li>6. Haga clic sobre Cambiar permisos.</li> <li>7. Seleccione "permisos por usuarios", al usuario1 asigne el permiso Control y al usuario2 SIN ACCESO.</li> <li>8. Cierre sesión de usuario1 e Inicie sesión con usuario2.</li> <li>9. Haga clic sobre MiCarpeta.</li> </ol> <p><b>Resultados esperados:</b></p> <ul style="list-style-type: none"> <li>• El sistema debe mostrar la ventana para cambiar permisos.</li> <li>• El sistema debe denegar el acceso al Usuario2 a MiCarpeta.</li> </ul>				
Notas:				

Numero de caso de prueba: 013				
Título: Notificar cambios en elementos en la Raíz del sistema.				
Área		Subarea		
Notificar nuevo elemento		Fase 3		
Prioridad	Tipo	Frecuencia	Tiempo estimado	
1	Funcionalidad	Milestone	3min	
<p><b>Descripción:</b> El sistema debe mostrar en la Raíz del sistema las actualizaciones a los elementos.</p> <p><b>Propósito de la prueba:</b> Verificar que se muestre en el historial de cambios cuando se crea una nueva carpeta.</p> <p><b>Condiciones iniciales:</b> El usuario debe iniciar sesión en el sistema. El usuario debe tener permiso de crear una carpeta sobre la ubicación actual.</p> <p><b>Pasos:</b></p> <ol style="list-style-type: none"> <li>1. Abra el navegador de internet.</li> <li>2. Ingrese la dirección del sistema.</li> <li>3. Inicie sesión en el sistema con el usuario admin y contraseña 1234.</li> <li>4. Cree una carpeta llamada NuevaCarpeta. Después cree otra carpeta o archivo dentro de NuevaCarpeta.</li> <li>5. Dirijase a la Raíz del sistema.</li> </ol> <p><b>Resultados esperados:</b></p> <ul style="list-style-type: none"> <li>• El sistema debe mostrar en la sección historia la bitácora de los nuevos elementos creados.</li> </ul>				
Notas:				

Numero de caso de prueba: 014			
Título: Notificar vía correo electrónico acerca de nuevos archivos.			
Área		Subarea	
Enviar elemento a principal		Fase 3	
Prioridad	Tipo	Frecuencia	Tiempo estimado
1	Funcionalidad	Milestone	5min
<p><b>Descripción:</b> El sistema tiene la función al momento de crear un archivo de marcar una casilla si se desea enviar una notificación a los usuarios que tienen permiso sobre los elementos.</p> <p><b>Propósito de la prueba:</b> Verificar que el correo electrónico sea enviado y que se muestre en el cliente sin errores.</p> <p><b>Condiciones iniciales:</b> El usuario debe iniciar sesión en el sistema. El usuario debe tener permiso de crear un archivo sobre la ubicación.</p> <p><b>Pasos:</b></p> <ol style="list-style-type: none"> <li>1. Abra el navegador de internet.</li> <li>2. Ingrese la dirección del sistema.</li> <li>3. Inicie sesión en el sistema.</li> <li>4. Diríjase a una carpeta donde tenga permiso de modificar.</li> <li>5. Haga clic sobre Nuevo archivo.</li> <li>6. Seleccione un archivo y marque la casilla @Email.</li> <li>7. haga clic en Cargar archivo y espere a que se cargue.</li> </ol> <p><b>Resultados esperados:</b></p> <ul style="list-style-type: none"> <li>• El sistema debe mostrar una notificación con el resultado del envío.</li> <li>• En caso de éxito se deberá recibir un correo electrónico con el nombre de los elementos agregados y su ubicación, además de un enlace.</li> </ul>			
Notas:			

Numero de caso de prueba: 015			
Título: Cargar elementos desde expedientes.			
Área		Subarea	
Cargar elementos desde expedientes		Fase 3	
Prioridad	Tipo	Frecuencia	Tiempo estimado
1	Funcionalidad	Milestone	5min
<p><b>Descripción:</b> Se debe permitir crear un archivo desde el sistema de expedientes.</p> <p><b>Propósito de la prueba:</b> Verificar que se cargue un archivo desde los detalles de un expediente.</p> <p><b>Condiciones iniciales:</b> El usuario debe iniciar sesión en el sistema. El usuario debe tener permiso de crear un archivo sobre la ubicación de resguardo de archivos de Expedientes.</p> <p><b>Pasos:</b></p> <ol style="list-style-type: none"> <li>1. Inicie sesión en el sistema de expedientes.</li> <li>2. haga clic sobre los detalles de cualquier expediente.</li> <li>3. Haga clic sobre el enlace del repositorio.</li> <li>4. Seleccione un archivo y haga clic en Cargar archivo.</li> <li>5. Diríjase al sistema repositorio y vaya al elemento que contiene información de los Expedientes.</li> </ol> <p><b>Resultados esperados:</b></p> <ul style="list-style-type: none"> <li>• El sistema debe mostrar el archivo que creó desde el sistema de expedientes.</li> </ul>			

### ANEXO 3 Requerimientos de infraestructura de TI para la ejecución del software.

En esta sección se muestran los requerimientos tanto de hardware como de software donde el sistema fue probado y arrojó resultados favorables para su ejecución, la finalidad de esto es dejar plasmado el entorno requerido para el funcionamiento del sistema. Las características marcadas con asterisco son requeridas para su cumplimiento como se especifica.

#### Requisitos de infraestructura hardware

Característica	Valor
<b>Espacio en la carpeta de publicación del servidor Web *</b>	30 Megabytes, este tamaño incrementará en cuanto se agreguen archivos al sistema.

#### Requisitos software

Característica	Versión
<b>Servidor Web</b>	IIS 8.5
<b>PHP *</b>	5.3.28 o superior.
<b>Módulos de PHP activados *</b>	php_fileinfo.dll, php_pdo_mysql.dll
<b>MySQL</b>	5.5
<b>Framework Phalcon PHP DLL *</b>	1.3.1
<b>Módulo URL Rewrite para IIS *</b>	2.0
<b>Bases de datos *</b>	Acceso a la base de datos de usuarios “integra” y “gfarias”, para lo cual el archivo de configuración del repositorio deberá ser revisado.
<b>Permisos *</b>	Permisos de escritura para el usuario encargado de ejecutar los procesos de IIS, generalmente IIS_IUSRS.