



**UNIVERSIDAD DE CHILE  
FACULTAD DE CIENCIAS FISICAS Y MATEMATICAS  
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

**SISTEMA DE ALMACENAMIENTO REMOTO DE ARCHIVOS ORIENTADO  
AL TRABAJO GRUPAL**

**MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN  
COMPUTACIÓN**

**AGUSTÍN DOMINGO ALMONTE FERRADA**

**PROFESOR GUÍA:  
MARIA CECILIA BASTARRICA PIÑEYRO**

**MIEMBROS DE LA COMISIÓN:  
JOSE MIGUEL PIQUER GARDNER  
ERICH EDUARDO REIMBERG NAVARRO**

**SANTIAGO DE CHILE  
AGOSTO 2007**

## **RESUMEN**

El almacenamiento y respaldo seguro de información es una necesidad vital para las empresas actuales. La información recopilada durante el ejercicio de una empresa representa toda su historia, aprendizaje y ventaja frente a sus competidores. Por esto el poseer un repositorio centralizado y seguro para la valiosa información se vuelve un objetivo importante, como parte del plan de negocios, en cualquier empresa.

En el presente trabajo de título se aborda la implementación de un sistema de almacenamiento remoto, que oriente su diseño y funcionamiento a generar un ambiente de trabajo grupal facilitando la interacción entre los usuarios.

El mercado actual ofrece numerosas soluciones de almacenamiento de información que en conjunto ofrecen seguridad, confidencialidad y alta disponibilidad de la información de los datos. El problema de estas soluciones radica en el poco provecho que se obtiene del ambiente de trabajo grupal generado por la centralización de la información. Es por este motivo que se vuelve atractiva la idea de desarrollar un sistema que reúna las características de los distintos sistemas de almacenamiento y además de un enfoque a las operaciones hacia mejorar los procesos internos de las organizaciones usuarias.

Para el diseño e implementación del sistema se investigaron y aplicaron los últimos avances en el desarrollo de aplicaciones Web utilizándose tecnologías como webServices, AJAX, WebDav y J2EE.

Como producto final se ha obtenido un sistema que cumple con brindar las características básicas esperadas de un buen sistema de almacenamiento, como la seguridad, privacidad y disponibilidad de la información almacenada. Que además saca provecho del ambiente centralizado de trabajo generado, para introducir mejoras en la interacción de los usuarios dando mayor velocidad y flexibilidad a los procesos que involucran la información almacenada en el sistema.

## **AGRADECIMIENTOS**

A mi profesora guía María Cecilia Bastarrica

A mi familia por el apoyo incondicional

A mis amigos por la fuerza cuando faltó

Este trabajo fue parcialmente financiado por el proyecto Fondecyt 1050642.

# TABLA DE CONTENIDOS

CAPITULO 1. Introducción .....	8
CAPITULO 2. Estudio de Mercado .....	10
2.1 Soluciones Presentes en el Mercado.....	10
2.1.1 Soluciones Basadas en Servidores FTP.....	11
2.1.2 Gemelo Storage Solutions .....	11
2.1.3 Novell InternetDisk .....	12
2.2 Recopilación de Funcionalidades Deseables.....	12
CAPITULO 3. Orientación Grupal .....	14
3.1 Sistemas de Administración de Conocimiento.....	14
CAPITULO 4. Requerimientos de Software.....	16
CAPITULO 5. Diseño del Sistema.....	19
5.1 Diseño de Estructura del Sistema .....	19
5.1.1 Dominios de Almacenamiento .....	19
5.1.2 Almacenamiento de la Información .....	21
5.2 Componentes del Sistema.....	22
5.2.1 API de funciones básicas.....	22
5.2.2 Interfaces WEB.....	24
5.2.3 Encriptación de archivos .....	26
5.2.4 Servidor WEBDAV.....	27

CAPITULO 6. Tecnologías.....	29
6.1 Tecnologías en WEB.....	29
6.2 Definición de Tecnologías.....	30
6.1.1 Soporte de la estructura.....	30
6.1.2 API del sistema.....	31
6.1.3 Interfaces Web.....	32
6.1.4 Acceso al Disco Remoto en Modo Local.....	34
6.1.5 Comunicación de Interfaz de Administración.....	35
 CAPITULO 7. Implementación.....	 36
7.1 Implementación de Estructura Básica del Sistema.....	36
7.1.1 Representación de la Estructura.....	36
7.1.2 Almacenamiento de la Estructura.....	38
7.1.3 Almacenamiento de la Información.....	39
7.2 Implementación de Funciones Básicas del Sistema.....	40
7.3 Interfaces Web.....	41
7.3.1 Interfaz Principal.....	42
7.3.2 Interfaz de Administración.....	47
7.4 Transferencia de Archivos.....	47
7.4.1 Applet de subida de archivos.....	48
7.4.2 Applet de descarga de archivos.....	48
7.5 Servicios Web.....	49
7.5.1 Operaciones de administración delegada.....	49
7.5.2 Operaciones adicionales para la administración de plataforma.....	50
7.5.3 Seguridad.....	51

CAPITULO 8. RESULTADO.....	53
8.1 Descripción general del sistema .....	53
8.1.1 Interfaz de usuarios.....	54
8.1.2 Subida de archivos.....	55
8.1.3 Descarga de archivos .....	56
8.1.4 Interfaz de administración .....	58
8.1.5 Vista del disco remoto como unidad local.....	63
8.2 Funcionalidades destacadas.....	64
8.2.1 Envío de notificaciones por actualización .....	64
8.2.2 Envío de notificaciones arbitrarias .....	66
8.2.3 Compartir Carpeta .....	68
CAPITULO 9. Conclusiones y discusión.....	70
9.1 Alcances de la Solución Lograda .....	70
9.2 Tareas a seguir .....	71
Bibliografía y referencias .....	73

# INDICE DE FIGURAS

<b>Figura 1:</b> Estructuras Básicas .....	20
<b>Figura 2 :</b> Ejemplos de representaciones de árboles de directorios LDAP .....	31
<b>Figura 3:</b> Estructura definida en la implementación del sistema .....	36
<b>Figura 4:</b> Almacenamiento en Ldap de la estructura.....	38
<b>Figura 5:</b> Árbol de navegación.....	42
<b>Figura 6:</b> Tabla de archivos.....	43
<b>Figura 7:</b> Barra de operaciones.....	43
<b>Figura 8:</b> Iconos diferenciadores de archivos y carpetas.....	44
<b>Figura 9:</b> Archivo con múltiples versiones. ....	44
<b>Figura 10:</b> Arrastre de archivos.....	45
<b>Figura 11:</b> Barras de ocupación.....	45
<b>Figura 12:</b> Ventanas emergentes .....	46
<b>Figura 13:</b> Eliminación de archivos .....	47
<b>Figura 14:</b> Flujo de llamada a un webService.....	52
<b>Figura 15:</b> Flujo de llamada a webService con filtro aplicado.....	52
<b>Figura 16:</b> Visión general del sistema.....	53
<b>Figura 17:</b> Interfaz Principal .....	54
<b>Figura 18:</b> Interfaz de subida de archivos .....	55
<b>Figura 19:</b> Solicitud de contraseña para encriptación .....	56
<b>Figura 20:</b> Descarga simple de un archivo .....	57
<b>Figura 21:</b> Interfaz de descarga de archivos.....	58

<b>Figura 22:</b> Solicitud de contraseña para descriptación .....	58
<b>Figura 23:</b> Interfaz de Administración .....	59
<b>Figura 24.1:</b> Vista de dominio en interfaz de administración. ....	60
<b>Figura 24.2:</b> Vista de dominio en interfaz de administración. ....	61
<b>Figura 25.1:</b> Vista de usuarios en interfaz de administración .....	62
<b>Figura 25.2:</b> Vista de administradores en interfaz de administración .....	63
<b>Figura 26:</b> Vista de carpeta remota como local .....	64
<b>Figura 27:</b> Envío de notificación .....	65
<b>Figura 28:</b> Envío de invitación .....	66
<b>Figura 29:</b> Vista de invitación .....	67
<b>Figura 30:</b> Compartir carpeta .....	68
<b>Figura 31:</b> Configuración de lista de notificación .....	69



# CAPITULO 1. Introducción

El almacenamiento y respaldo de información es un tema importante dentro de los procesos en una empresa. No sólo porque se debe asegurar que la información esté disponible en todo momento, sino que además se posea un respaldo de ésta de forma de asegurar que ninguna parte de la información será omitida por pérdida. Generalmente la mayor parte de la información crítica de una empresa se encuentra en los computadores personales de sus empleados y su pérdida podría llegar a ser muy costosa. Es por este motivo que toma tanto valor el realizar respaldos de información de una forma sistemática.

En esta necesidad de almacenar y respaldar información, cada día adquiere un rol más importante la red, que al ser el punto de unión entre las ubicaciones distribuidas de una empresa, se presenta como el gran puente para la información. Por ejemplo, para una empresa que posea diversas sucursales alrededor del mundo que trabajan en equipo, sería interesante poder acceder a la misma información y con la misma facilidad independientemente del lugar geográfico desde donde se esté operando. Sería importante además que esta información, a pesar de estar disponible alrededor del mundo, no implicara un peligro de espionaje de negocios.

Actualmente el mercado informático ofrece múltiples soluciones que cubren la necesidad de respaldo de información [1,2,3]. Estas soluciones van desde simples servidores FTP, hasta soluciones más sofisticadas que incluyen un mayor nivel de administración sobre el espacio de respaldo. Si bien hasta los servicios más básicos encontrados cumplen con el objetivo de respaldar la información, la gran mayoría no poseen herramientas de administración sobre el espacio de almacenamiento que faciliten el trabajo grupal entre los usuarios del sistema. Por otro lado, también existen soluciones más completas que incluyen en su diseño opciones orientadas al trabajo grupal, además de funcionalidades que agregan valor al servicio como notificaciones, pero que limitan la operación inmediata desde cualquier lugar al requerir la instalación de software en el cliente para el acceso a las funciones avanzadas.

En la presente memoria se busca diseñar y desarrollar una solución de almacenamiento remoto que cumpla con los requerimientos básicos que debe brindar un servicio de respaldos, y que además agregue valor con operaciones de administración sobre el espacio de almacenamiento, un diseño pensado en favorecer el trabajo grupal dentro de una estructura jerárquica de usuarios, opciones de compartimiento flexible entre los usuarios del sistema, y un modo de operación que de confianza sobre la confidencialidad de la información almacenada.

## **CAPITULO 2. Estudio de Mercado**

El fin del trabajo a realizar es un producto para el cual existe competencia en el mercado actual, por lo que el primer paso a dar es el estudio y análisis de las soluciones disponibles, considerando en este estudio el uso de tecnología y las funcionalidades desarrolladas, para fijar una base sobre lo esperado para este tipo de productos y definir las nuevas funcionalidades que agregarán valor frente a los competidores existentes.

### **2.1 Soluciones Presentes en el Mercado**

El mercado actual ofrece numerosas soluciones para almacenar información remota, desde simples y poco novedosos servidores FTP, hasta complejos sistemas de almacenamiento que permiten al usuario numerosas operaciones de administración. Entre los dos extremos de soluciones presentes se encuentra una amplia gama de sistemas de almacenamiento, algunos como un sistema construido sobre sistemas previos como los del tipo Gmail y Whailemail que almacenan la información en correos electrónicos, y otros que se quedan en el medio por ser más elaborados que un servidor FTP pero no tan completo como los mejores sistemas.

Para el estudio necesario del mercado solo se considerarán aquellos sistemas que aporten utilidad a la definición de un sistema de almacenamiento remoto complejo.

A continuación se presentan tres soluciones destinadas al respaldo de información presentes en el mercado actual. Se estudiaron estos tres productos porque cubren el rango de soluciones existentes diferenciando tipos y funcionalidades adicionales.

### **2.1.1 Soluciones Basadas en Servidores FTP**

Si bien esta opción no representa una solución específica en el mercado se agrega como una alternativa que agrupa a muchos productos comercializados para el respaldo de información. Este tipo de solución para el respaldo es la más común y también la más simple. Como su nombre lo dice consiste en servidores FTP que permiten la transferencia de archivos mediante éste protocolo.

El nivel de operaciones que ofrece este tipo de soluciones es muy básico, limitándose a la simple transferencia de archivos. Por esto, estos sistemas no ofrecen funcionalidades agregadas a sus usuarios que permitan tener una mayor interacción.

Generalmente la comercialización de productos basados en servidores FTP consiste en la venta de un software elaborado que cumple la función de cliente FTP con alguna personalización en la interfaz del usuario.

### **2.1.2 Gemelo Storage Solutions**

Gemelo Storage Solutions[1] es una empresa dedicada a dar soluciones de respaldo. Dentro de sus productos, el más completo es el llamado Data Safe que permite respaldar información brindando además múltiples funcionalidades extras como:

- Administración centralizada de usuarios
- Acceso Web a la información almacenada (solo lectura)
- Opciones para compartir archivos y carpetas

La limitación de este producto se encuentra en que el acceso a la operación de respaldo es restringida a los computadores que tengan instalado el software diseñado para ese propósito, no permitiendo una interacción libre desde cualquier lugar a los usuarios.

### **2.1.3 Novell InternetDisk**

Novell InternetDisk[2] es sin duda una de las mejores soluciones de almacenamiento remoto disponibles en la actualidad. Su diseño cumple con tener soporte para crear comunidades dentro del sistema lo que permite un trabajo grupal. Además posee diversas funcionalidades que lo hacen destacarse frente a los demás sistemas, algunas de las capacidades que ofrece este sistema son las siguientes:

- Visualizar y operar el disco remoto como si fuera local
- Capacidad para enviar notificaciones
- Compartir carpetas entre usuarios
- Visualización Web del disco remoto
- Bloquear carpetas con el uso de claves
- Realizar respaldos de carpetas locales

Como se puede notar, este es un producto bastante completo, sin embargo la gran limitación que posee es que su operación esta basada en la instalación de software en el cliente. El punto de movilidad de este sistema lo da la interfaz web que provee. Sin embargo, las operaciones que se pueden realizar a través de este medio están restringidas a simples transferencias de archivos dependiendo de la instalación de software local para acceder a mas operaciones.

## **2.2 Recopilación de Funcionalidades Deseables**

Frente a las posibilidades que ofrece el mercado se pueden definir las funcionalidades deseables que debería tener el sistema a desarrollar para ser competitivo y estar a la par con lo que esperaría un usuario del sistema. Estas funcionalidades están compuestas por las que ofrecen las alternativas presentadas.

- Administración centralizada de usuarios
- Acceso Web a la información almacenada
- Opciones para compartir archivos y carpetas
- Visualizar y operar el disco remoto como si fuera local
- Capacidad para enviar notificaciones
- Bloquear carpetas con el uso de claves
- Realizar respaldos de carpetas locales

Al menos estas funcionalidades debe tener el sistema a desarrollar para que pueda competir frente a las soluciones alternativas. Además, se puede tomar como ventaja el hecho que no existe una solución que cumpla con todas estas características funcionales y que haga uso de la última tecnología en aplicaciones Web, por lo que el sistema a desarrollar puede explotar una gran mejora en la usabilidad utilizando los últimos avances en el desarrollo de interfaces.

## **CAPITULO 3. Orientación Grupal**

La orientación grupal que se busca darle al sistema no contempla la habilitación de edición colaborativa de documentos. El fin del sistema no es la edición en línea de contenidos por múltiples usuarios sino brindar una plataforma segura en donde se pueda compartir contenidos.

Se espera que el sistema logre agilizar la interacción entre los usuarios permitiendo el envío de notificaciones ante nuevos contenidos y la definición de grupos de interés a la espera de cambios. Por ejemplo, un grupo de vendedores desearía ser notificado sobre la actualización de la lista de precios, el grupo de interés lo formarían los vendedores y el nuevo contenido la lista de precios.

Como orientación grupal se requiere que el sistema incentive la comunicación y división de tareas entre los usuarios. Para esto un buen ejemplo base a seguir para determinar directrices de diseño es un sistema de administración de conocimiento.

### **3.1 Sistemas de Administración de Conocimiento**

Los sistemas de administración de conocimiento se enfocan en actividades administrativas que incentivan la creación, reunión, organización y difusión del conocimiento dentro de una empresa[25].

Las características de un sistema como el mencionado son de gran utilidad al diseñar un sistema en que se busca mejorar la interacción entre los usuarios de un repositorio de información.

El artículo “Knowledge management systems: Issues, Challenges, and Benefits”[21] presenta un estudio de los sistemas de administración del conocimiento existentes

mostrando sus falencias, desafíos y beneficios. Se considerara este estudio para definir propiedades deseables del sistema a desarrollar.

#### Llaves para el desarrollo de un sistema de administración de conocimientos

- Reunir la información de forma fácilmente usable.
- Asegurar la confidencialidad del usuario.
- Involucrar a los usuarios en la mantención de la información.
- Determinar responsabilidades para la administración del conocimiento almacenado.
- Definir los requisitos estructurales necesarios para el sistema.
- Mantener el sistema actualizado con respecto a las tecnologías que utiliza.
- Mantener la información segura en Internet.

Entre los beneficios mencionados de un sistema de administración del conocimiento destacamos los siguientes:

- Facilita y agiliza la comunicación
- Incentiva la participación
- Reduce los tiempos de resolución de problemas

Considerando las directrices mencionadas en el artículo se diseñará un sistema para facilitar la comunicación entre los usuarios, un fácil manejo, mantención y administración de acceso a la información.



## CAPITULO 4. Requerimientos de Software

Si bien actualmente el mercado presenta variadas soluciones de almacenamiento, desde las más básicas hasta las más completas y robustas, es interesante crear una solución que no sólo provea las funciones disponibles en el mercado, sino una aplicación que sea orientada para facilitar el trabajo en equipo, con funcionalidades especialmente dirigidas hacia este objetivo. La aplicación estará pensada para aquellos usuarios que desean poder tener el control de los datos de empresa y las acciones sobre éstos, desde cualquier lugar del mundo. Para cumplir este último requisito, es indispensable que la aplicación pueda ser accedida desde cualquier plataforma, sin la necesidad de instalar software.

A continuación se definen los requisitos que debe cumplir el sistema para lograr los objetivos planteados y al mismo tiempo ser competitivo frente a las soluciones existentes en la actualidad.

<b>Requisito</b>	<b>Descripción</b>
REQ.1	El sistema debe estar disponible para ser utilizado desde cualquier lugar a través de Internet.
REQ.2	Las funcionalidades incluidas en el sistema no deben requerir la instalación de software adicional en el equipo del usuario
REQ.3	El sistema debe permitir la transferencia de información entre carpetas remotas o entre el disco local y el remoto.
REQ.4	Se debe permitir el envío de notificaciones de actualización de

	<p>contenidos, incluyendo un link directo a la carpeta remota donde se encuentra el archivo actualizado.</p>
REQ.5	<p>Se debe dar la posibilidad de compartir contenidos almacenados en el disco remoto utilizando o no clave de acceso con usuarios o no usuarios del sistema.</p>
REQ.6	<p>El sistema debe definir permisos de acceso sobre las carpetas remotas. En particular definir para cada usuario si esta habilitado para leer, escribir o crear dentro de una carpeta remota.</p>
REQ.7	<p>Se debe permitir la administración de los permisos existentes sobre una carpeta remota por parte del usuario dueño de la misma.</p>
REQ.8	<p>El tamaño de los archivos a transferir no se debe ver limitado por el sistema.</p>
REQ.9	<p>El sistema debe permitir la existencia de versiones del mismo archivo, dando la posibilidad de configurar el numero de versiones que se soportan.</p>
REQ.10	<p>El sistema debe tener una orientación al trabajo grupal permitiendo compartir información almacenada en un espacio común.</p>
REQ.11	<p>La transferencia de archivos entre el equipo local y disco remoto debe considerar la protección de los datos mediante métodos de encriptación</p>
REQ.12	<p>El sistema debe permitir la visualización de los contenidos almacenados de manera local en la forma de un disco extra en el sistema operativo.</p>

REQ.13	El sistema debe poseer una interfaz de acceso para usuarios administradores en donde se centralice la administración de usuarios.
REQ.14	El sistema debe poseer una interfaz de administración completa en donde se permita crear/modificar/eliminar grupos e integrantes, y administrar el espacio asignado.
REQ.15	La interfaz de administración del sistema debe poder aislarse de la aplicación principal para evitar el intento de ingreso de usuarios no autorizados.

## **CAPITULO 5. Diseño del Sistema**

Ya definidos los requerimientos impuestos sobre el sistema se procede a plantear el diseño que los cumplirá. Para esto se presenta el diseño de la estructura interna del sistema y el caso particular de cada componente del sistema.

### **5.1 Diseño de Estructura del Sistema**

Una de las bases en el planteamiento del sistema es que favorezca el trabajo grupal, por esto su diseño debe ser dirigido a este mismo objetivo básico. Además, se deben considerar la aplicación de funcionalidades existentes en las soluciones disponibles y de nuevas funcionalidades como:

- Accesos restringidos por usuarios a carpetas
- Confidencialidad de la información almacenada
- Envío de invitaciones a personas ajenas al sistema para la visualización de archivos

Con el fin de cumplir con todos los requisitos mencionados, a continuación se define el diseño requerido para la estructura base del sistema.

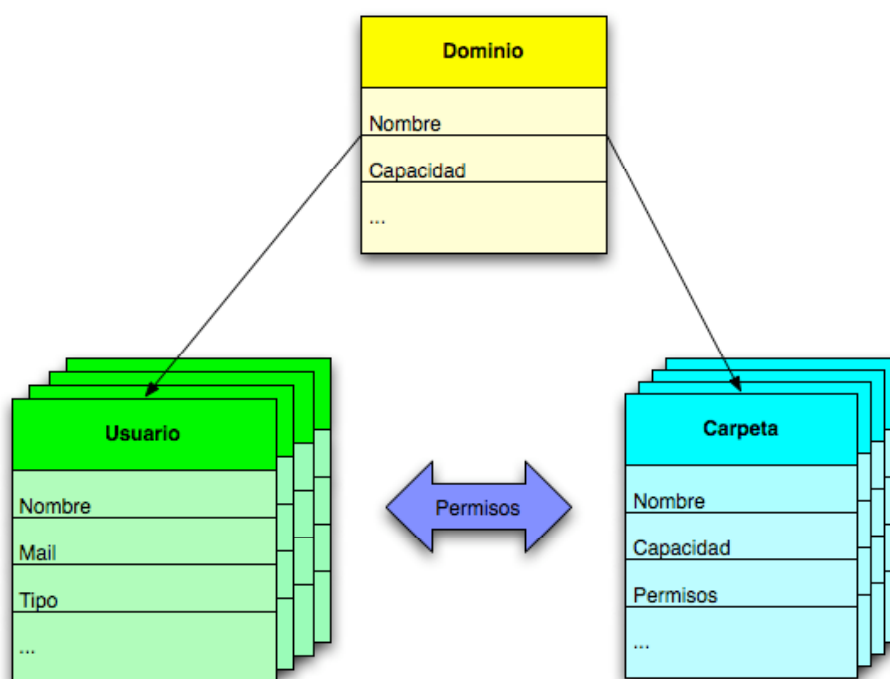
#### **5.1.1 Dominios de Almacenamiento**

Para cumplir con el requisito de funcionalidades grupales, se crearon tres niveles de detalle dentro del sistema: dominios, carpetas y usuarios. Los dominios permiten definir una agrupación de usuarios que comparten un mismo espacio de disco, que a su vez estará conformado por carpetas. Si se lleva esto a la práctica los dominios conformaran la empresa

que contiene a sus empleados, los que pueden compartir operaciones dentro del dominio manipulando la información almacenada en las carpetas.

En base a este diseño la estructura del sistema será dividida en dominios de almacenamiento. Para cada dominio existirá un árbol de directorios (carpetas) independiente asociado a un conjunto de usuarios con permiso para acceder y participar del dominio.

Cada uno de estos tres componentes estará caracterizado por propiedades como se puede apreciar en la figura 1.



**Figura 1:** Estructuras Básicas

La figura muestra las componentes de la estructura básica del sistema junto con sus propiedades e interacciones.

Para resguardar la información entre los usuarios de un dominio, se definen permisos para los usuarios. Los permisos utilizados son definidos a nivel de carpetas y cada

carpeta tendrá asignados permisos de acceso para cada usuario del dominio. Los niveles de permisos son tres:

- Lectura
- Escritura
- Creación (o Administración)

Un usuario con permiso de creación sobre una carpeta está autorizado para crear subcarpetas, modificar sus propiedades y asignar permisos sobre ellas. Además un usuario podría no tener ninguno de los tres permisos mencionados para alguna carpeta en particular, con lo que el acceso a ésta le sería negado.

Dentro del dominio los usuarios además son caracterizados por el rol que cumplen, permitiendo así la existencia de usuarios del tipo básico y administradores. Los usuarios administradores poseen permiso de creación sobre toda la estructura de directorios del dominio, cumpliendo un rol de moderadores. Estos usuarios administradores tienen además la posibilidad de administrar su grupo por medio de la interfaz de administración del sistema a describir mas adelante.

### **5.1.2 Almacenamiento de la Información**

La información respaldada en el disco remoto será almacenada directamente en el sistema de archivos. Los archivos almacenados soportarán la existencia de versiones previas, para esto el sistema reconocerá la solicitud de respaldo de un archivos un nombre previamente creado y lo grabará como una nueva versión del archivo ya existente. Este tipo de características junto a otras como el usuario dueño y el estado de encriptación del archivo será almacenado para cada archivo como propiedades independientes para cada versión.

Por su parte, como fue mencionado antes, la información de carpetas creadas remotamente formara parte de la estructura básica de la solución por lo que será almacenada junto el resto de las componentes del sistema.

## **5.2 Componentes del Sistema**

### **5.2.1 API de funciones básicas**

La primera componente a definir del sistema es el conjunto de funciones que posibilitarán la interacción de las interfaces de usuario con la estructura interna. El grupo de funciones se divide en dos conjuntos, uno que contiene las funciones asociadas a la interfaz normal de uso y otro contenedor de las funciones a utilizar en labores administrativas del sistema.

#### **5.2.1.1 Funciones Principales**

Para habilitar la interfaz de usuarios se diseñó una API de funciones que permite:

- Crear carpetas
- Asignar permisos sobre las carpetas
- Listar el contenido de una carpeta
- Verificar índices de ocupación dentro de una carpeta
- Cambiar propiedades de una carpeta
- Crear archivos en el disco
- Enviar notificaciones de cambios
- Compartir una carpeta a usuarios externos

Con estas operaciones se cubren las acciones de los usuarios sobre el disco remoto.

#### **5.2.1.2 Funciones de Administración**

Para la administración del sistema, se proveerá de una interfaz que permitirá la creación de dominios y usuarios junto con la capacidad de modificar propiedades de éstos.

Ésta interfaz se apoya sobre el uso de una API de aprovisionamiento que provee las siguientes funciones sobre los dominios y usuarios.

### **Dominios**

- Crear dominio
- Borra dominio
- Lista de dominios existentes
- Verifica si el dominio existe
- Obtiene atributos del dominio
- Modifica atributos del dominio
- Verifica atributos del dominio

### **Usuarios**

- Crea usuario en dominio
- Borra usuario de dominio
- Lista de usuarios de un dominio
- Verifica si un usuario existe en el dominio
- Obtiene atributos de un usuario
- Modifica atributos de un usuario
- Verifica atributos de un usuario

Las funciones disponibles dentro de la administración son divididas, al momento de la implementación, en dos niveles, el nivel de administración central y el de administración delegada. Esto se hace con el fin de diferenciar los niveles de administración general del sistema, de la administración que un usuario administrador de dominio pueda hacer sobre las características de su dominio y usuarios.



## **5.2.2 Interfaces WEB**

Para asegurar la disponibilidad del sistema desde cualquier lugar, sin necesidad de instalación de software en el cliente, es necesario que el sistema disponga de todas sus funcionalidades por medio de una interfaz Web

La interfaz Web debe conectar las operaciones, gatilladas por los usuarios, hacia la estructura interna del sistema. Además de cumplir un rol funcional se debe tener en cuenta que tan fácil de usar resulta la interfaz a los usuarios finales para incentivar su uso.

Como fue definido en la especificación de requerimientos es necesario disponer de dos interfaces Web, una para operaciones básicas y otra para realizar la administración del sistema.

### **5.2.2.1 Interfaz de Usuarios**

Esta interfaz cumple con realizar las operaciones comunes sobre el sistema. Dentro de éstas operaciones se consideran las funcionalidades relacionadas con el respaldo y recuperación de información, cambios en la estructura de directorios del espacio de almacenamiento, modificación de permisos y propiedades de las carpetas remotas, envío de invitaciones y notificaciones a usuarios, y toda operación que se efectúe sobre el espacio asignado para un dominio de respaldo.

En cuanto al diseño, uno de los objetivos es lograr una interfaz lo más similar posible a una aplicación local. Para lograr esto se imitará el comportamiento de un explorador de archivos típico de un sistema operativo, simulando la vista de archivos con un árbol de navegación, una barra de herramientas y ventanas emergentes para la solicitud de información adicional requerida por las aplicaciones.

### **5.2.2.2 Interfaz de Administración**

La interfaz de administración busca cumplir los requisitos definidos para las modificaciones sobre grupos de usuarios definidos y la administración del espacio de almacenamiento. Para esta interfaz se requieren dos tipos de acceso, uno para la administración de un dominio en particular y otro para la administración general del sistema, transversal al sistema, sobre la totalidad de usuarios y dominios de almacenamiento.

#### **➤ Administración Central**

La administración central es el acceso para administradores generales, no asociados a un dominio en particular, sino administradores de la plataforma del sistema. Desde acá se podrán realizar las siguientes operaciones:

- Creación de dominios, y asignación de cuotas
- Mantenimiento de dominios y cuotas
- Listado de dominios configurados, sus cuotas, y nivel de ocupación.

Además de estas operaciones, el administrador central podrá realizar cualquier modificación específica a un dominio como si fuera un administrador del dominio en cuestión.

#### **➤ Administración Delegada (Administradores de Dominio)**

La administración delegada está pensada para ser utilizada por el o los usuarios administradores de un dominio. Desde esta administración, el usuario administrador del dominio podrá realizar las siguientes operaciones:

- Creación de usuarios
- Mantenición de usuarios
- Eliminación de usuarios
- Activación y desactivación de usuarios
- Cambio de claves de acceso
- Listado de usuarios creados
- Visualización del nivel de ocupación del dominio

Considerando que por seguridad es más conveniente ubicar la aplicación de administración en un servidor diferente al que usan los usuarios básicos para acceder al sistema, se implementó esta aplicación utilizando métodos remotos. En base a esto, se definió una capa de servicios Web que podrán ser llamados desde otro servidor, en donde estará la administración, haciendo el enlace a las funciones provistas por la API de aprovisionamiento.

### **5.2.3 Encriptación de archivos**

Uno de los requerimientos mas importantes definidos corresponde al resguardo de la privacidad de la información almacenada en el disco remoto. Para esto se definió la necesidad de contar con un medio que permita el respaldo de archivos utilizando un medio de encriptación. Para cumplir el requerimiento se definen dos interfaces especiales de interacción que permitirán encriptar la transferencia de la información desde el momento en que comienza el flujo desde el computador del usuario, de esta manera todo el proceso de encriptación estará fuera del sistema dando mas confiabilidad al usuario quien será el único que manipula los archivos descriptados.

#### **5.2.3.1 Interfaz de subida de archivos**

Esta aplicación debe funcionar incrustada dentro de la aplicación Web. Proveerá de una interfaz avanzada para subir archivos en la que se permitirá:

- Subir múltiples archivos
- Encriptar los archivos utilizando una clave asignada por el usuario[17 18]
- Evitar la transferencia de información libre, sin encriptar, desde el usuario hasta el sistema de respaldo.
- Enviar notificaciones de cambios

Esta interfaz de transferencia de archivos solicitará una contraseña al usuario que luego utilizará para encriptar el archivo a subir en el computador del usuario antes de enviarlo. Con esto se evita que la clave de encriptación pase por el sistema y evitando la posible captura de información confidencial en durante el paso por la red.

#### **5.2.3.2 Interfaz de descarga de archivos**

Esta aplicación será complementaria al desarrollo de la interfaz de subida. Al momento de solicitar la descarga de información encriptada almacenada en el sistema, se solicitará la clave necesaria para desencriptar con la que se desencriptará el archivo de forma local luego de transmitir el archivo.

#### **5.2.4 Servidor WEBDAV**

Para que el disco remoto pueda ser utilizado a modo de un disco local, como ofrecen las soluciones actuales, es necesario implementar una funcionalidad de servidor que tenga esta característica.

El estándar para este tipo de soporte es WebDav[3], que permite la transferencia de información y contenidos de un sistema remoto para ser visualizado de manera local a través de HTTP. Esta característica es implementada como una componente más sobre la el sistema dando las funciones de un servidor WebDav. La funcionalidad de cliente WebDav esta cubierta dentro del sistema operativo Windows, dentro de éste se incluye un cliente

webDav por lo que se cumple con el requisito de no necesitar instalar software adicional para su funcionamiento.

# CAPITULO 6. Tecnologías

## 6.1 Tecnologías en WEB

Desde hace algunos años la Web ha experimentado un cambio en la naturaleza de sus contenidos. En un principio la Web se componía de paginas estáticas que desplegaban contenidos a sus usuarios, sin necesidad de interactuar con él. Últimamente se ha marcado un cambio de tendencia que ha llevado a la Web a redefinir su utilidad convirtiéndose en una plataforma de aplicaciones[26]. Este nuevo enfoque se enmarca dentro de lo que se ha llamado Web 2.0, en donde se diseña la Web enfocándose en la interacción con sus usuarios.

Tomando en cuenta la tendencia a cambiar el tipo de contenidos de la Web, debemos enfocar el diseño del sistema hacia una aplicación orientada a dar una operabilidad al usuario lo mas similar posible a una aplicación local. Con este fin los navegadores han adquirido un rol muy importante al soportar la descarga de aplicaciones integradas a los servidores Web que permite ejecutar aplicaciones de manera local al usuario con las ventajas en velocidad de respuesta y mejores en cuanto a las posibilidades de interacción que esto conlleva.

Dentro de los lenguajes mas utilizados en los últimos tiempos en el desarrollo de aplicaciones Web esta javascript. Una ventaja de javascript frente a otros lenguajes es que la gran mayoría de los navegadores incluye soporte para ejecutar aplicaciones en éste lenguaje sin necesidad de instalar componentes adicionales. Junto con el creciente uso de javascript en el desarrollo de aplicaciones, han surgido numerosos Frameworks<sup>1</sup> que

---

<sup>1</sup> Framework: Estructura de soporte definida sobre la cual otro proyecto de software puede ser organizado y desarrollado

facilitan el desarrollo, al utilizarse como una base sobre la cual programar aplicaciones mediante funciones ya definidas para la definición de interfaces y la comunicación con el servidor. En el punto referido a las interfaces web de este capítulo, se aborda en más detalle la utilidad y los diversos Frameworks existentes.

Por el lado del servidor los cambios si bien han existido tal vez han sido menos notorios para el usuario final. Dentro de estos cambios se encuentran definiciones de estándares para el desarrollo de aplicaciones Web como J2EE, además del desarrollo de diversos frameworks orientados a facilitar la integración entre aplicaciones que funcionan en el computador de usuario final y el servidor web de contenidos.

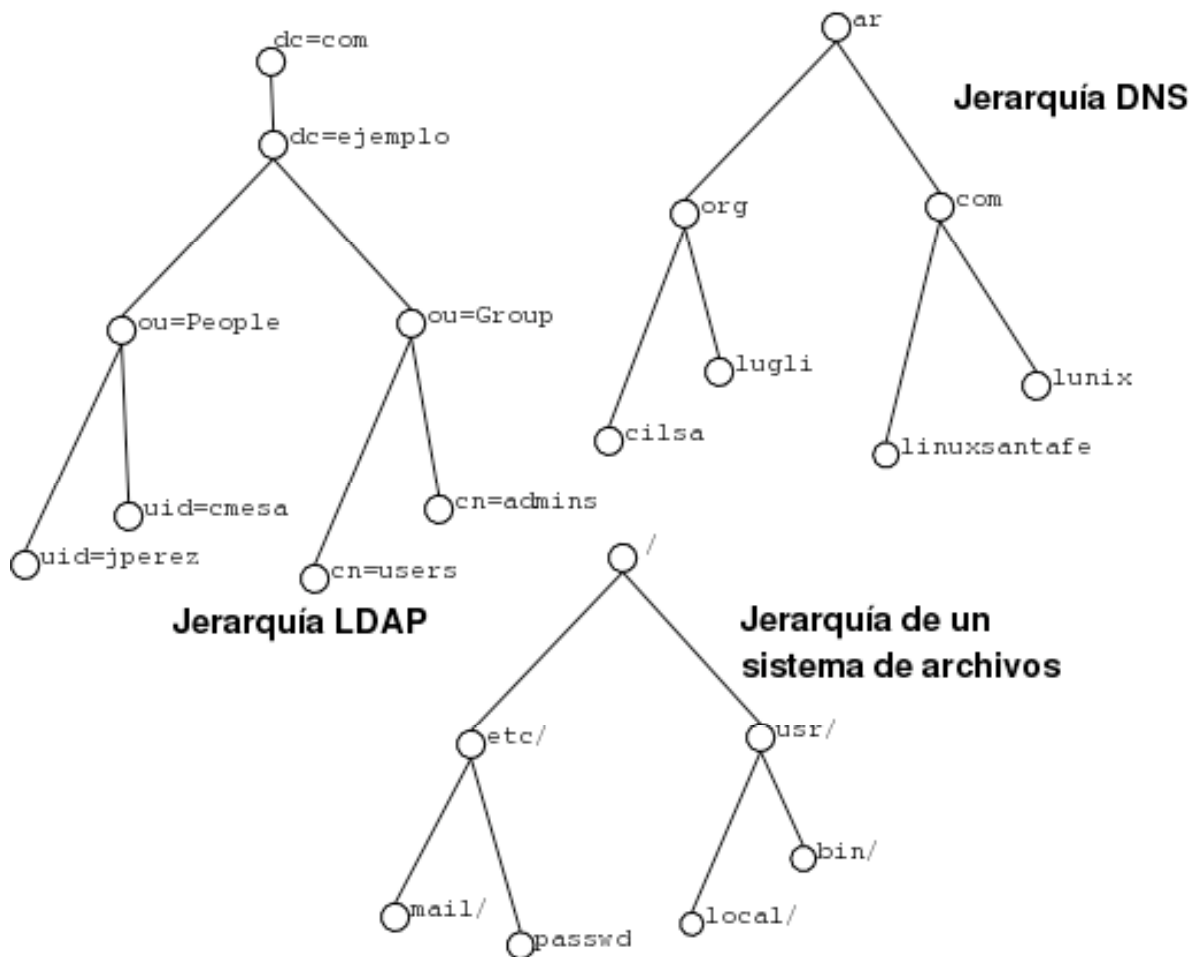
## **6.2 Definición de Tecnologías**

Frente a las funcionalidades deseables de agregar en el sistema fue preciso investigar la unión de tecnologías que permite su implementación de manera eficaz y de acuerdo a los estándares actuales, de manera de lograr un software atractivo y competitivo frente a las opciones existentes. Para este objetivo se dividió el sistema para definir la tecnología a utilizar en cada componente.

### **6.1.1 Soporte de la estructura**

Uno de los requisitos básicos impuestos al diseño es la capacidad de cubrir operaciones derivadas de la interacción entre los usuarios. Para esto en el punto de diseño anterior se presentó una estructura de dominios, que agrupara usuarios y el árbol de directorios asociado a él. Al unir esta condición junto a la existencia de propiedades en estos tres niveles de la estructura, surge la idea de poseer una estructura jerárquica que permita dar atributos a sus componentes. Con esto se podría agrupar un grupo de usuarios y carpetas bajo un dominio, junto a sus propiedades.

Al investigar se encontró que lo mas utilizado para representar estructuras de usuarios son los directorios LDAP[11], que dan la posibilidad de almacenar datos y asociar atributos a ellos. Este tipo de estructuras también permiten almacenar la estructura de carpetas con sus propiedades manteniendo control sobre la pertenencia de cada componente al representarse en un árbol. La Figura 2 muestra ejemplos de representaciones de directorios LDAP.



**Figura 2 :** Ejemplos de representaciones de árboles de directorios LDAP

### 6.1.2 API del sistema

Debido a la naturaleza jerárquica de la estructura del sistema, el lenguaje a utilizar debe facilitar la representación de cada componente. Por esto es una buena opción la utilización de un lenguaje orientado a objetos como Java.



Por otro lado, considerando que la API de funciones necesaria para el funcionamiento del sistema requiere funcionar en el entorno de un servidor Web, se vuelve conveniente utilizar para su implementación la tecnología J2EE[14 20]. Con esto se aprovecha la portabilidad que brinda el lenguaje Java, además de la confianza por el probado uso de esta tecnología en el desarrollo de aplicaciones Web.

Por otro lado, la interfaz Web, de gran importancia al ser la cara del sistema frente a los usuarios, debe proveer una interacción fácil y atractiva. Para esto se tuvo que definir una tecnología que facilite la actualización de la interfaz frente a solicitudes de contenidos del servidor, volviendo este proceso mas rápido de manera que se asimile a una aplicación local. Para este fin existen numerosos frameworks, basados en distintos lenguajes, que integran interfaces Web con las funcionalidades disponibles en el servidor. En el caso de este sistema el lenguaje seleccionado es Java por lo que el número de frameworks disponibles se reduce a los siguientes:

- DWR
- Echo
- Google Web ToolKit
- IT Mill ToolKit
- ZK
- ThinWire

La selección del framework mas conveniente debe ir en relación a la tecnología utilizada para construir la interfaz, para poder asegurar la integración de ambas tecnologías se completará esta definición junto con la requerida para la interfaz Web.

### **6.1.3 Interfaces Web**

Para el diseño de las interfaces Web, dada la utilización de Java como el lenguaje en el servidor, se consideró conveniente y ventajoso la utilización de tecnología JSP[19 20] (Java Server Pages), incluida dentro de J2EE, para el desarrollo base de las paginas Web del sistema. Esta tecnología es muy utilizada en la actualidad en el desarrollo de

aplicaciones Web y permite una unión simple entre la interfaz Web y las funciones definidas en el servidor.

Además, para dar mayor usabilidad y lograr un aspecto de interfaz local, se utilizó la tecnología AJAX[22], con ésta tecnología se obtienen interfaces Web mas rápidas, al reducir la transferencia de información, y con componentes que asimilan los que se podrían encontrar en una aplicación local.

Para facilitar el desarrollo de aplicaciones basadas en AJAX existen numerosos Frameworks que brindan un grupo de componentes con las que se puede construir desde la interfaz hasta el modelo de interacción con el usuario. La idea es seleccionar un framework basado en javascript para poder construir una interfaz mas atractiva y en un menor tiempo al utilizar las componentes definidas por el framework.

Algunos de los Frameworks presentes en la actualidad, y basados en javascript, son los siguientes:

- Prototype
- Dojo
- QooxDoo
- Yahoo UI
- TIBCO General Interface

De este grupo de frameworks destacan por la capacidad para generar interfaces amigables QooxDoo[23] y TIBCO GI[24]. Estos otorgan una gran variedad de componentes predefinidas para la construcción de interfaces basadas en AJAX.

En una primera instancia, se decidió utilizar para el desarrollo el Framework provisto por QooxDoo. Los motivos de la elección fueron la gran variedad de componentes y controles desarrollados sobre ellos que permitían una interfaz muy atractiva y adaptable por el usuario. Luego, durante la etapa de implementación, se notó lentitud en el desarrollo de las interfaces por la dificultad de posicionar cada componente en la interfaz y la poca

flexibilidad a los cambios, cada cambio requería cambiar demasiado código. Por esta razón, se reimplementaron las interfaces utilizando TIBCO GI, que al ser distribuido con un IDE de desarrollo facilita la creación de las interfaces mediante un modelo gráfico de diseño basado en el arrastre de componentes y con un control de cambios limitados dentro del mismo IDE.

Por el lado de la comunicación con el servidor TIBCO GI ofrece integración con WebServices. Además, TIBCO recomienda la utilización de DWR como framework para la comunicación entre sus interfaces y el servidor Web, por lo que la interacción se hizo utilizando el mencionado Framework.

#### **6.1.4 Acceso al Disco Remoto en Modo Local**

Una forma estándar de dar acceso a un disco remoto es mediante la utilización de un servicio WebDAV según el estándar definido en [3 - 9]. Al utilizar este estándar, el usuario puede acceder a la vista del disco remoto como si fuera una unidad local utilizando cualquier cliente soportado. En particular Windows y las últimas distribuciones de Linux incluyen dentro de su instalación un cliente WebDAV estándar lo que permitirá al usuario operar sin necesidad de instalar software adicional, y para sistemas como OSX de Mac la conexión con carpetas remotas mediante WebDAV viene integrada a su sistema de exploración de archivos. La visión local del disco que dará un servidor WebDAV permitirá al usuario copiar, pegar y eliminar archivos del disco remoto.

La manera más simple de implementar esta funcionalidad es mediante la extensión HTTP descrita en [9] para WebDAV. Para esto es necesaria la utilización de un servidor WebDAV, que cumpla con las especificaciones mencionadas, funcionando integrado al conjunto de servicios que correrán en el servidor Web del sistema y que además se integre de manera correcta con la estructura definida para el sistema.

El servidor web de apache permite agregar la funcionalidad webDav fácilmente agregando un módulo en su configuración. Sin embargo esta solución no es posible de usar

en el sistema a desarrollar debido a que se necesita tener total control sobre las cuotas de almacenamiento, lo que obliga a manejar toda operación de respaldo a través del sistema sin delegar nada. Por esta razón se deberá implementar un servidor webDav integrado a la estructura básica del sistema cumpliendo los estándares definidos en [9] .

### **6.1.5 Comunicación de Interfaz de Administración**

Uno de los requerimientos definidos solicita la que la interfaz de administración del sistema pueda ser alojada en un servidor distinto del usado para la interfaz principal, esto para dar mayor seguridad al sistema. Para cumplir este requerimiento fue necesario definir la tecnología a utilizar para comunicar la interfaz de administración con el servidor donde se aloje la estructura principal del sistema.

Un estándar muy utilizado actualmente es el de servicios remotos contenidos dentro de la tecnología de Servicios Web[12 13]. Esta tecnología permite definir funciones en un servidor que quedan expuestas para ser llamadas remotamente desde otros servidores. Para implementar los servicios remotos se utilizará la implementación provista por Apache AXIS[15], que se distribuye con herramientas para una fácil creación de las clases necesarias y su publicación en el servidor.

# CAPITULO 7. Implementación

## 7.1 Implementación de Estructura Básica del Sistema

El primer punto en la implementación del sistema es la creación de las clases que representen la estructura básica obtenida del diseño planteado. En base a esta implementación se realizó luego la definición de las funciones necesarias para la correcta interacción entre las componentes.

### 7.1.1 Representación de la Estructura

Para la representación de la estructura se definió una estructura de clases basada en las necesidades reunidas por los requisitos y el diseño planteado. La figura 3 muestra un diagrama con las clases implementadas con éste propósito.

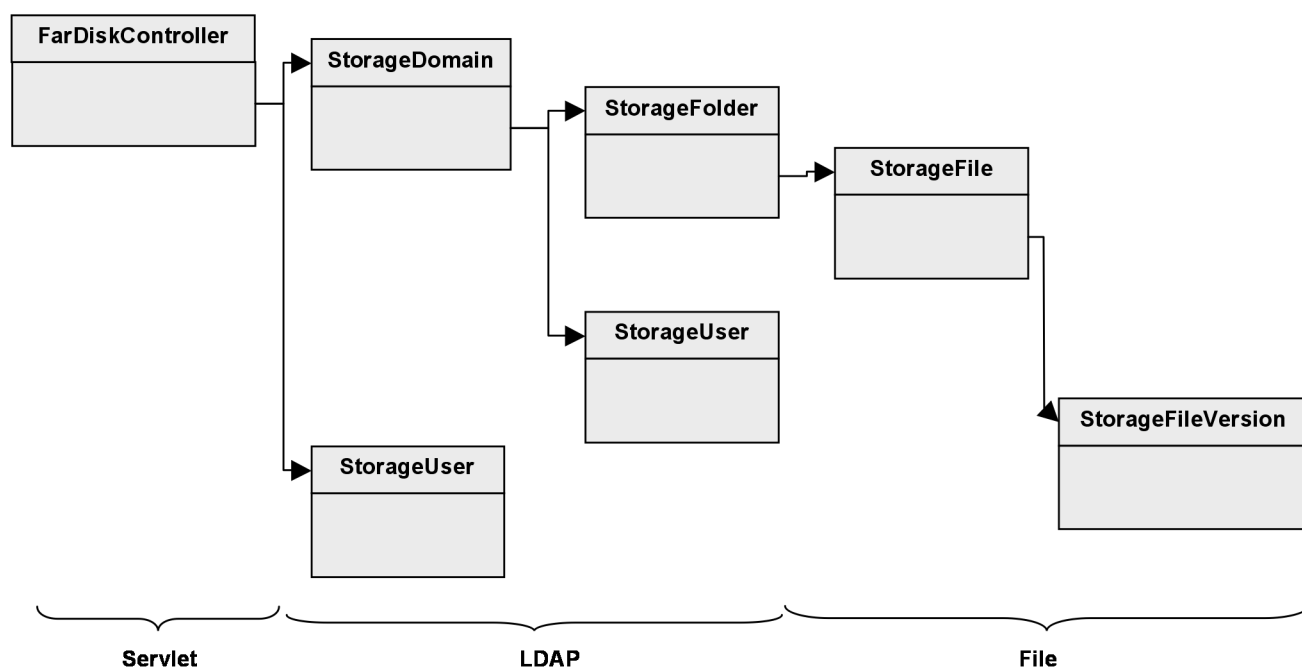


Figura 3: Estructura definida en la implementación del sistema

En la figura se ve una división por niveles de las clases, este nivel corresponde a la ubicación que toma la componente a la que representa cada clase implementada.

- Servlet: nivel administrado por el contenedor Web
- LDAP: nivel almacenado en el servidor LDAP
- File: nivel almacenado en el sistema de archivos. En este nivel se almacenaran los archivos contenidos por el sistema.

A continuación se detalla el rol que cumple cada clase en la estructura del sistema.

**FarDiskController:** Clase base para la interacción y configuración del sistema. Esta clase es manejada por el contenedor Web para dar acceso a las distintas funciones implementadas y componentes requeridas por el usuario.

**StorageDomain:** Clase representativa de un dominio de almacenamiento. Su ubicación y definición esta en el servidor LDAP.

**StorageUser:** Clase representativa de un usuario del sistema. Los usuarios pueden ser de dos tipos, usuarios de un dominio (que están asociados al dominio al que pertenecen) y usuarios de plataforma que realizaran labores administrativas.

**StorageFolder:** Clase que representa una carpeta dentro de un dominio. Al igual que los dominios y usuarios, su definición y propiedades se ubican en el servidor LDAP.

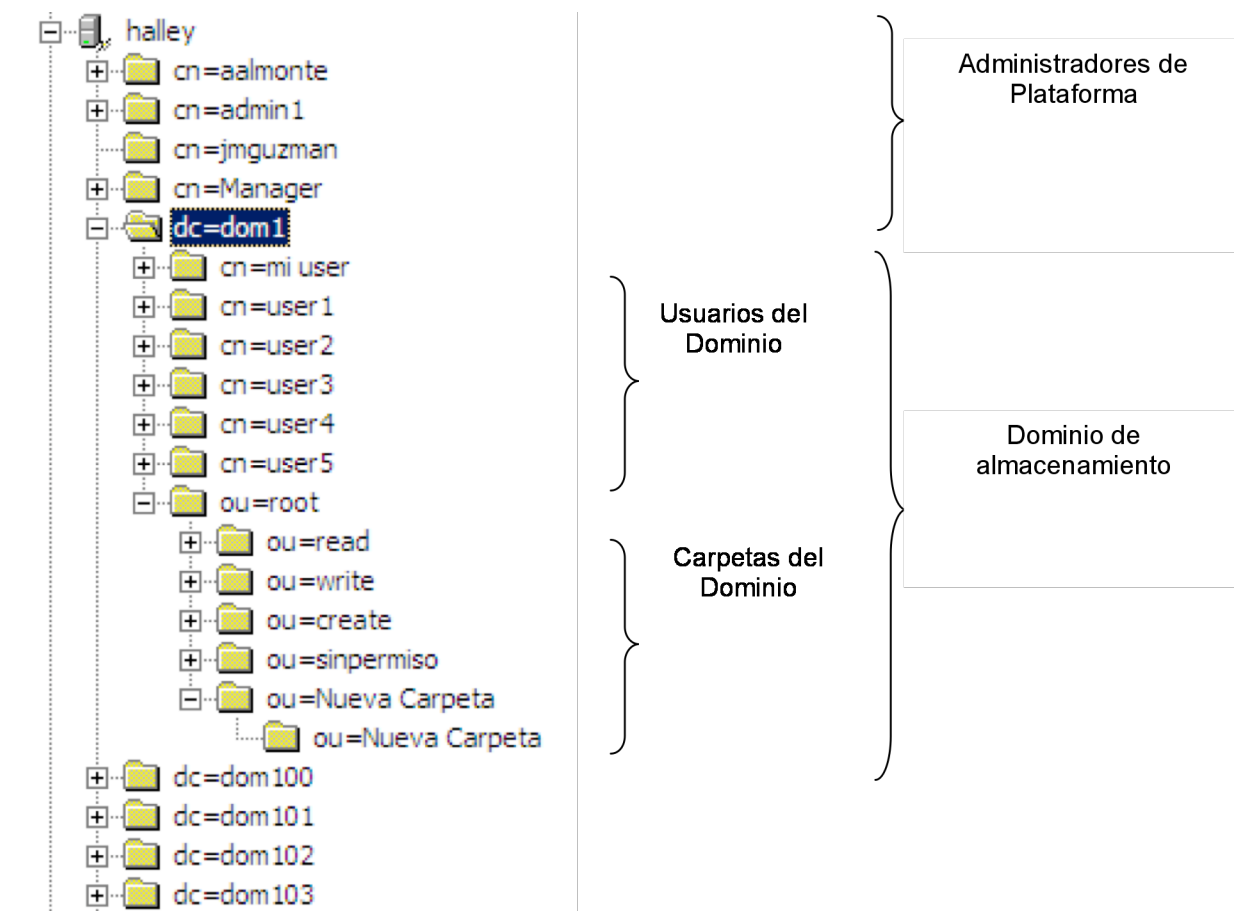
**StorageFile:** Clase que representa un archivo almacenado en el sistema. Los archivos almacenados en el sistema de archivos se recuperan a través de una instancia de esta clase.

**StorageFileVersion:** Clase que representa una versión específica de un StorageFile. Esta clase permite mantener varias versiones de un mismo archivo. Esta es la clase que representa a cada archivo en el sistema de archivos remoto.

### 7.1.2 Almacenamiento de la Estructura

Como fue definido la estructura básica del sistema se compone de los dominios, usuarios y carpetas. Esta estructura tiene un carácter jerárquico y como se definió en el capítulo de diseño, se utilizará un directorio LDAP para su almacenamiento.

En la figura 4 se muestra una vista del árbol formado por las componentes del sistema.



**Figura 4:** Almacenamiento en Ldap de la estructura

Como lo muestra la figura en un primer nivel se ubican los administradores de plataforma y los dominios. Dentro de cada dominio se ubican los usuarios pertenecientes al dominio en particular y sus la estructura de directorios.

### **7.1.3 Almacenamiento de la Información**

Si bien, el almacenamiento de la información no requirió de un desarrollo mayor al indicado para su representación en el sistema, si se tuvo que definir la forma en que se almacenarían los archivos en el sistema de archivos. En particular las definiciones necesarias involucran como representar las propiedades de los archivos en el sistema.

Los archivos almacenados por el sistema tienen en principio tres propiedades a representar:

- Nombre del usuario que subió y por lo tanto dueño del archivo
- Versión del archivo almacenado
- Es un archivo encriptado o no.

La implementación resuelta para representar las propiedades de cada archivo consiste en que al momento de almacenar el archivo en el disco remoto, se agregarán tags al final del nombre del archivo con el contenido de las propiedades, para todas aquellas que no sean directamente obtenibles de un archivo común. Por ejemplo, si el usuario aalmonde decide almacenar el archivo memoria.doc en el disco remoto encriptando su contenido, y para ese archivo existen dos versiones anteriores, entonces el sistema lo almacenará con el nombre “memoria.doc,u=aalmonde,v=3,e=1”, de esta manera se define una notación que el sistema conoce para recuperar las propiedades. Además utilizando esta implementación se permite agregar nuevas propiedades a los archivos en el futuro.



## 7.2 Implementación de Funciones Básicas del Sistema

Junto con la definición de clases para la representación de la estructura interna del sistema, se definieron funciones para realizar la interacción y definición de los componentes del sistema.

La definición de las funciones se hizo dentro de la clase correspondiente a cada nivel del sistema. Cada componente del sistema maneja sus funciones relacionadas desde la clase implementada para su representación. En base a esto se encuentran funciones definidas en cada nivel de los componentes del sistema:

- Usuarios
- Dominios
- Carpetas
- Archivos

### Funciones a nivel de Usuario

- Autenticación de usuarios
- Cambio de atributos
- Eliminación del usuario
- Obtención de dominio de almacenamiento
- Verificación de permisos

### Funciones a nivel de Dominio

- Agregar nuevos usuarios al dominio
- Obtención de usuarios del dominio
- Eliminación del dominio
- Obtención de referencia a la carpeta base del dominio

### Funciones a nivel de Carpetas

- Creación de nuevas carpetas
- Modificación de cuotas asignadas en carpetas
- Obtención de archivos contenidos
- Eliminación de archivos
- Envíos de invitaciones y notificaciones para visitar la carpeta

#### Funciones a nivel de Archivos

- Creación de una nueva versión de un archivo
- Eliminación de una versión particular de un archivo
- Cambiar la ubicación de un archivo
- Obtener atributos del archivo

## **7.3 Interfaces Web**

Como se mencionó antes, las interfaces Web fueron desarrolladas utilizando el Framework TIBCO General Interface. La utilización de este framework permitió construir interfaces muy funcionales y de gran similitud con aplicaciones locales.

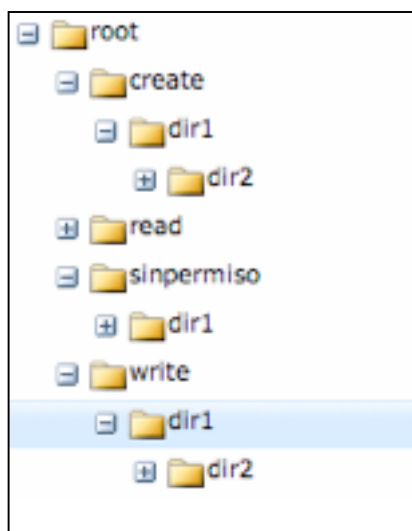
El mayor énfasis en cuanto a diseño de interfaz se hizo en la interfaz principal, en donde todos los usuarios ingresarán para respaldar su información. Por otro lado la interfaz de administración, también fue construida utilizando el mismo framework, pero su diseño es mas simple limitándose solo a dar visualización a las operaciones mencionadas para la labor de administración.

### 7.3.1 Interfaz Principal

La interfaz principal fue dotada de funcionalidades especiales en busca de lograr una similitud con un navegador de archivo local. Algunas de las funcionalidades agregadas se describen a continuación:

#### Árbol de navegación de directorios

Para la navegación a través de las carpetas remotas se imitó un árbol de directorios típico de aplicaciones locales. En la figura 5 se ve el resultado de esta adaptación.



**Figura 5:** Árbol de navegación.

## Tabla de archivos

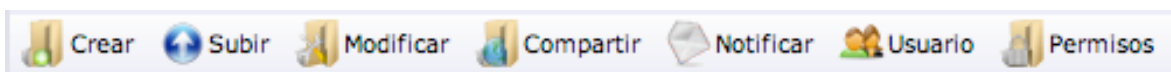
Para la visualización de los archivos y carpetas contenidos en el directorio actual se creo una tabla con columnas ordenables, facilitando con esto las búsquedas en caso de muchos archivos.

Nombre	Fecha	Versión	Tamaño	Usuario	Eliminar
dir1			0		✖
InformeCC69EV1.1	Jul 8, 2007	1	245,248	user1	✖
Memoria0.8	Jul 8, 2007	1	437,760	user1	✖
Memoria0.96.doc	Jul 8, 2007	1	490,496	user1	✖
Memoria0.97.doc	Jul 8, 2007	1	488,448	user1	✖
Memoria0.97.pdf	Jul 8, 2007	3	924,580	user1	✖
Pac.jpg	Jul 8, 2007	1	592,426	user1	✖

**Figura 6:** Tabla de archivos

## Barra de Operaciones

Al igual que en una aplicación de escritorio, se dotó a la interfaz de una barra superior de botones representando cada operación disponible. Los botones se activan y desactivan dependiendo de los permisos que tenga el usuarios en la carpeta visualizada.



**Figura 7:** Barra de operaciones

## Uso de iconos para indicar encriptación

Para diferenciar archivos encriptados de los normales y carpetas con capacidad de encriptación se utilizaron iconos especiales como indicadores. En la figura 7 se da una muestra de los iconos utilizados.

Nombre	Fecha	Versión	Tamaño	Usuario
carpetaNueva			0	
dir1			0	
InformeCC69EV1.1	Jul 8, 2007	1	245,248	user1
Memoria0.8	Jul 8, 2007	1	437,760	user1
Memoria0.96.doc	Dec 31, 1969	1	0	user1
Memoria0.97.doc	Jul 8, 2007	1	488,448	user1
Memoria0.97.pdf	Jul 8, 2007	3	924,580	user1
Memoria0.97.pdf				user1
Memoria0.97.pdf				user1
Memoria0.97.pdf				user1
Pac.jpg	Jul 8, 2007	1	592,426	user1

**Figura 8:** Iconos diferenciadores de archivos y carpetas

### Visualización especial para archivos con versiones

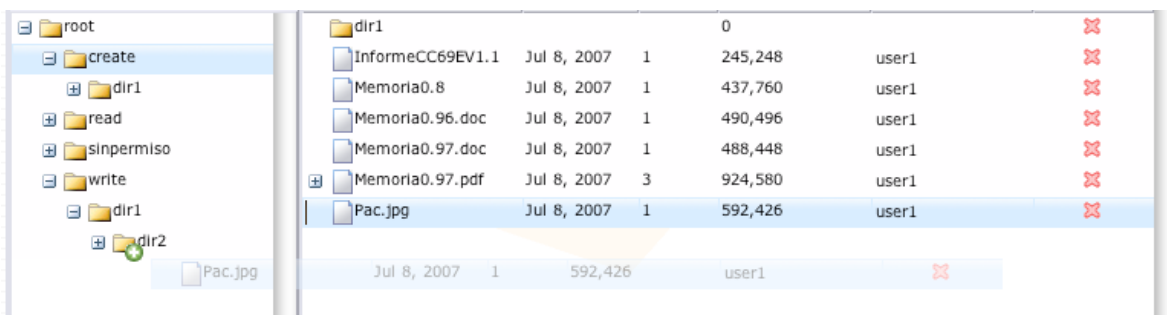
Algo importante en la implementación de la interfaz fue lograr un acceso fácil e intuitivo a las versiones anteriores de los archivos. Para esto se ideó un tipo de archivo desplegable para mostrar las versiones almacenadas. En la figura 8 se muestra un archivo con sus versiones.

Memoria0.97.pdf	Jul 8, 2007	3	924,580	user1	✘
Memoria0.97.pdf	Jul 8, 2007	3	924,580	user1	✘
Memoria0.97.pdf	Jul 8, 2007	4	924,580	user1	✘
Memoria0.97.pdf	Jul 8, 2007	5	924,580	user1	✘

**Figura 9:** Archivo con múltiples versiones.

### Mover archivos de carpeta

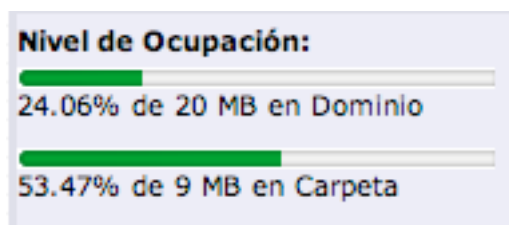
Para dar mayor facilidad de uso a la interfaz se implementó el soporte para mover archivos entre carpetas utilizando el arrastre con el Mouse.



**Figura 10:** Arrastre de archivos

### **Barras de ocupación**

Para mantener informado al usuario sobre la capacidad de almacenamiento que resta en su dominio y carpeta actual, se incluyó en la interfaz un componente indicador de la ocupación. En este componente se muestra de manera actualizada la capacidad libre que queda tanto en el dominio como en la carpeta visualizada.



**Figura 11:** Barras de ocupación.

### **Ventanas emergentes especiales**

Las operaciones disponibles en la interfaz requieren solicitar información al usuario para ejecutarse, esta solicitud se hace a través de una ventana emergente. Para evitar la confusión y pérdida de ventanas emergentes desde la aplicación, se implementó un sistema de ventanas emergentes embebidas dentro de la misma ventana principal, la figura 11 muestra las ventanas emergentes utilizadas. Con esta característica se evita la generación de

ventanas emergentes en diferentes instancias del navegador y además se evita lidiar con configuraciones que bloquean la creación de ventanas emergentes.

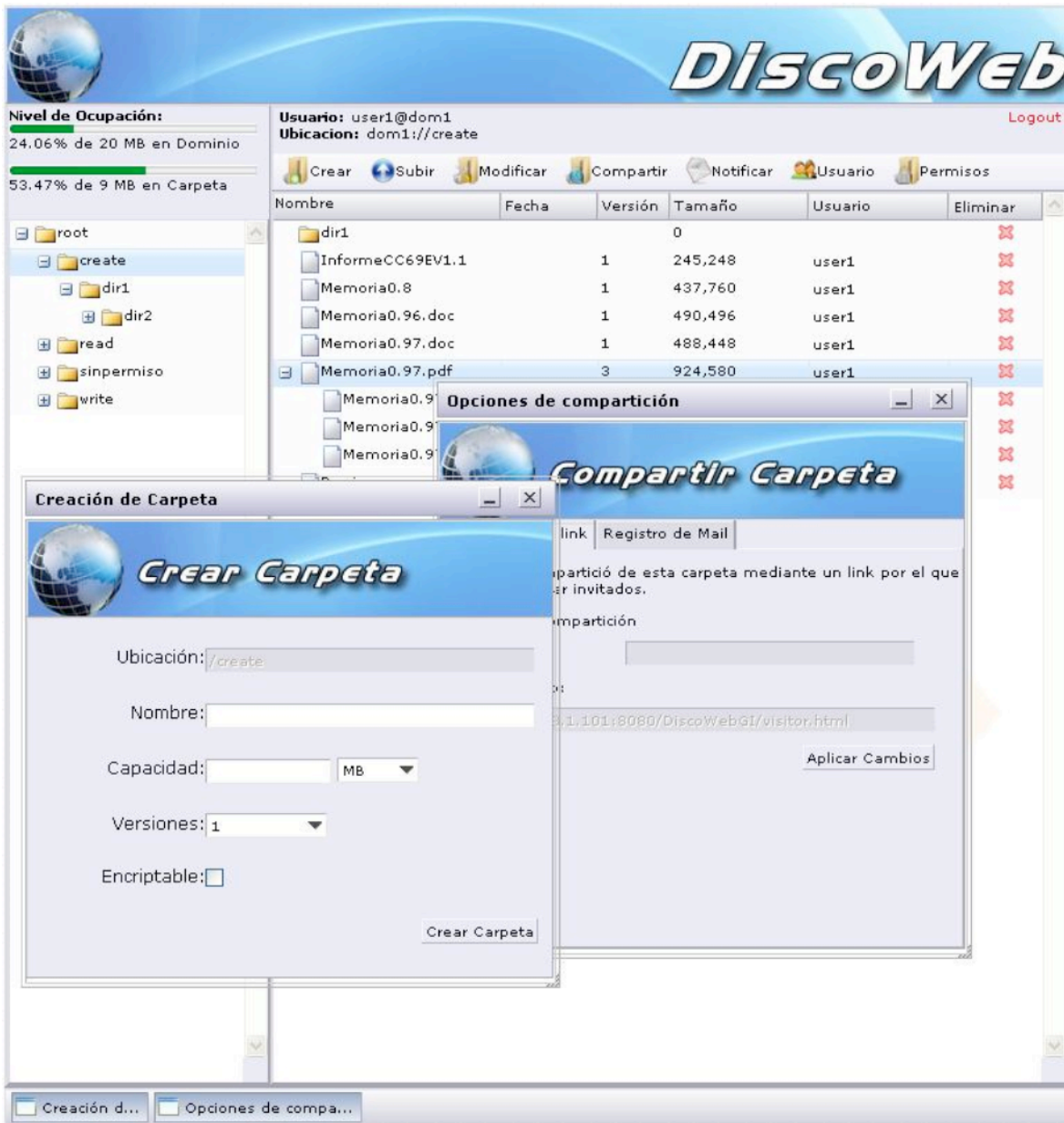






Figura 12: Ventanas emergentes

Además, se puede ver en la figura 11 como se agregó a la interfaz una barra de tareas para almacenar las ventanas emergentes activas y minimizadas.

### Eliminación directa de archivos

Para una operación mas rápida de los archivos se agrego un botón de eliminación junto a cada entrada en la tabla de archivos.

 InformeCC69EV1.1	Jul 8, 2007	1	245,248	user1	
 Memoria0.8	Jul 8, 2007	1	437,760	user1	

**Figura 13:** Eliminación de archivos

### **Doble clic**

Emulando el efecto de un doble clic en una aplicación local, se agregó la acción de abrir las carpetas y descargar archivos con un doble clic.

## **7.3.2 Interfaz de Administración**

Como se mencionó, el caso de la interfaz de administración es mas simple que el implementado para la principal. En este caso se reutilizaron algunas funcionalidades aplicadas en la interfaz principal aplicándolas a las funcionalidades de administración.

## **7.4 Transferencia de Archivos**

La implementación de las interfaces de transferencia de archivos se realizó utilizando applets. Al desarrollarse en este lenguaje, se permite implementar una aplicación que pueda encriptar localmente utilizando librerías y métodos definidos[17,18], que este contenida dentro de la interfaz Web y con los permisos necesarios para acceder al disco local con la posibilidad de leer, para el caso de subida, y escribir archivos para el caso de descarga.



### **7.4.1 Applet de subida de archivos**

Para almacenar archivos en el disco remoto se implemento un applet que funciona embebido dentro de la interfaz Web. Este applet permite subir múltiples archivos a la vez, con la posibilidad de encriptar sus contenidos. Además, una vez que los archivos han sido respaldados con éxito, el applet permite el envío de un mensaje de notificación de actualización de contenidos.

Para el caso de almacenar archivos encriptados, el applet solicita el ingreso de una clave que será utilizada para generar la llave de encriptación. Esta encriptación se produce completamente de manera local por lo que el sistema no podrá conocer nunca la clave de encriptación. Para mayor comodidad se permite al usuario definir una clave y recordarla para sus próximos respaldos, la clave recordada quedará almacenada de manera local y nunca en el servidor remoto.

Además del estado final de la operación de respaldo, la interfaz compuesta por el applet proporciona durante el tiempo de subida de los archivos, información estadística sobre la velocidad promedio y actual a la que se esta realizando el respaldo, tiempos restantes, y porcentaje de subida de los archivos.

### **7.4.2 Applet de descarga de archivos**

Para el caso en que se desea descargar un archivo, se implementó un applet mucho mas simple que para el caso de subida. Para este caso solo se necesitó implementar una aplicación que fuera capaz de desencriptar un archivo a partir de una contraseña ingresada y almacenar el archivo, ya desencriptado, en el disco local en la ubicación indicada por el usuario.

Para descargar un archivo encriptado el usuario debe ingresar la misma clave que se utilizo en el proceso de encriptación, de otro modo la descarga le será denegada. Si el

usuario, al momento de encriptar el archivo, decidió recordar su clave, el applet de descarga le consultará si desea utilizar la clave almacenada evitando tener que reingresar la llave comúnmente utilizada.

## 7.5 Servicios Web

La implementación de los servicios remotos necesarios para integrar la interfaz de administración con el sistema fue desarrollado utilizando la implementación de servicios web provista por apache con el proyecto AXIS[24] siguiendo la metodología definida en [15].

El desarrollo consistió principalmente en la definición de dos clases contenedoras de las funciones remotas. Una clase contiene los métodos disponibles para el nivel de administración delegada y la otra complementa estos métodos con operaciones que permiten una administración completa de la plataforma. La idea de esta separación es limitar el acceso a administradores de dominio solo a una de las clases mientras un administrador de plataforma tiene acceso a ambos conjuntos de operaciones.

### 7.5.1 Operaciones de administración delegada

Las operaciones dirigidas a la administración delegada se concentraron en la clase UserServer. En esta clase se definieron operaciones que permiten la administración sobre un dominio en particular con el fin de limitar la acción del administrador a su propio dominio. Las operaciones permitidas son las siguientes:

- **Creación de un nuevo usuario.** Se permite la creación de un nuevo usuario en el dominio, siempre y cuando las propiedades del dominio indiquen que el número de usuarios permitidos no se ha alcanzado.
- **Eliminación de Usuarios.**

- **Activar Usuario.** Cambia el estado de un usuario, marcándolo como activo o inactivo. Solo los usuarios activos pueden ingresar al sistema, utilizando esta operación el administrador puede deshabilitar el acceso de un usuario en particular.
- **Modificar usuario.** Modificar las propiedades que definen a un usuario.
- **Identificar un usuario.** Permite la identificación de un usuario en el sistema. Esta función se utiliza en el login para limitar el acceso.
- **Cambiar claves de acceso.** Permite la mantención de las claves de acceso de los usuarios. Como es una operación de administración, no exige el ingreso de la contraseña anterior.
- **Obtener la lista de usuarios del dominio.** Útil para desplegar los usuarios creados en la interfaz.
- **Obtener propiedades del dominio.** En este caso de administración las propiedades del dominio son solo informativas y no pueden ser modificadas.

## 7.5.2 Operaciones adicionales para la administración de plataforma

Para completar la integración de la interfaz de administración se definió la clase AdminServer que contiene operaciones exclusivas para administradores de plataforma del sistema. Un administrador de la plataforma, además de valerse de estas funciones para operar, también tiene acceso a todas las funciones definidas para la administración delegada. Las operaciones disponibles en esta clase son las siguientes:

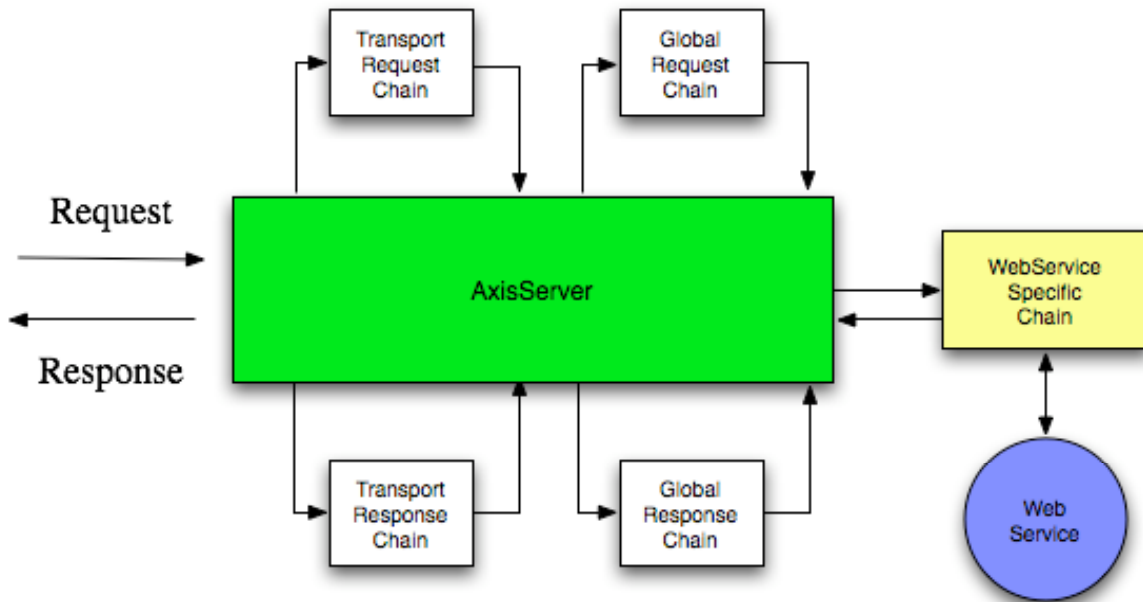
- **Creación de un administrador de plataforma.**
- **Eliminación de administrador.**
- **Modificación de administrador.**
- **Cambiar claves de acceso.**
- **Cambiar estado de administrador.** Cambia el estado entre activo e inactivo para bloquear el acceso a la interfaz.

- **Creación de dominio.**
- **Eliminar dominio.**
- **Modificar dominio.** Modificar las propiedades de un dominio, por ejemplo, la cuota de almacenamiento o el número máximo de usuarios permitido.
- **Obtener lista de dominios y administradores.** Utilizado para desplegar información en la interfaz.

### 7.5.3 Seguridad

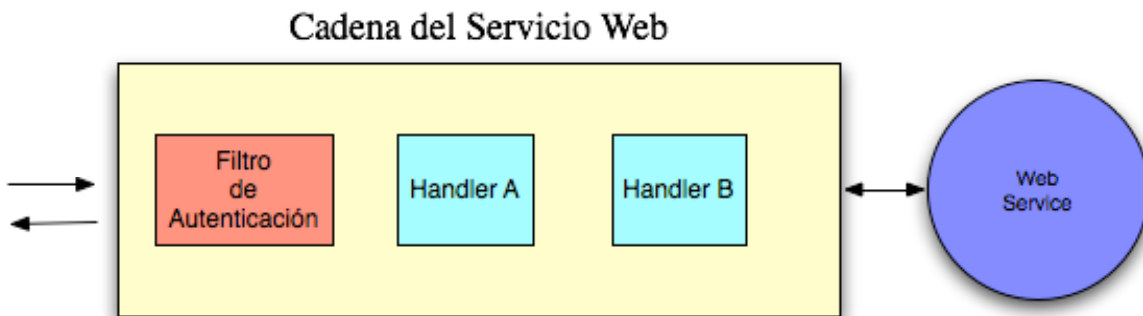
Uno de los motivos para incluir la tecnología de servicios web en el sistema fue prevenir exponer la interfaz de administración a la totalidad de los usuarios ubicándola en el mismo servidor. Pero al habilitar los servicios web en el servidor, si bien quitamos el acceso a la interfaz de administración, quedan expuestas todas las funciones de administración de la plataforma. Esto sería un problema si no se implementara una política de seguridad para el acceso a las funciones.

Para limitar el acceso a las operaciones expuestas a través de los servicios remotos se implementó un filtro de autenticación que se ubica entre las funciones y los requerimientos. Los requerimientos en AXIS se ven como un flujo de información que pasa a través de cadenas de procesamiento. En la figura 14 se pueden ver las distintas cadenas que aplican cuando el servidor AXIS recibe un requerimiento.



**Figura 14:** Flujo de llamada a un webService

La implementación consistió en ubicar un eslabón extra en la cadena de ejecución del llamado al método remoto (ver Figura 15), esto se logro agregando un manejador adicional en la cadena especifica a la ejecución del servicio web publicado. De esta forma todo requerimiento debe pasar antes por el filtro de autenticación. El filtro construido recupera el usuario y contraseña seteado al momento de efectuar el llamado desde la interfaz de administración, realizando la validación contra los usuarios del sistema.



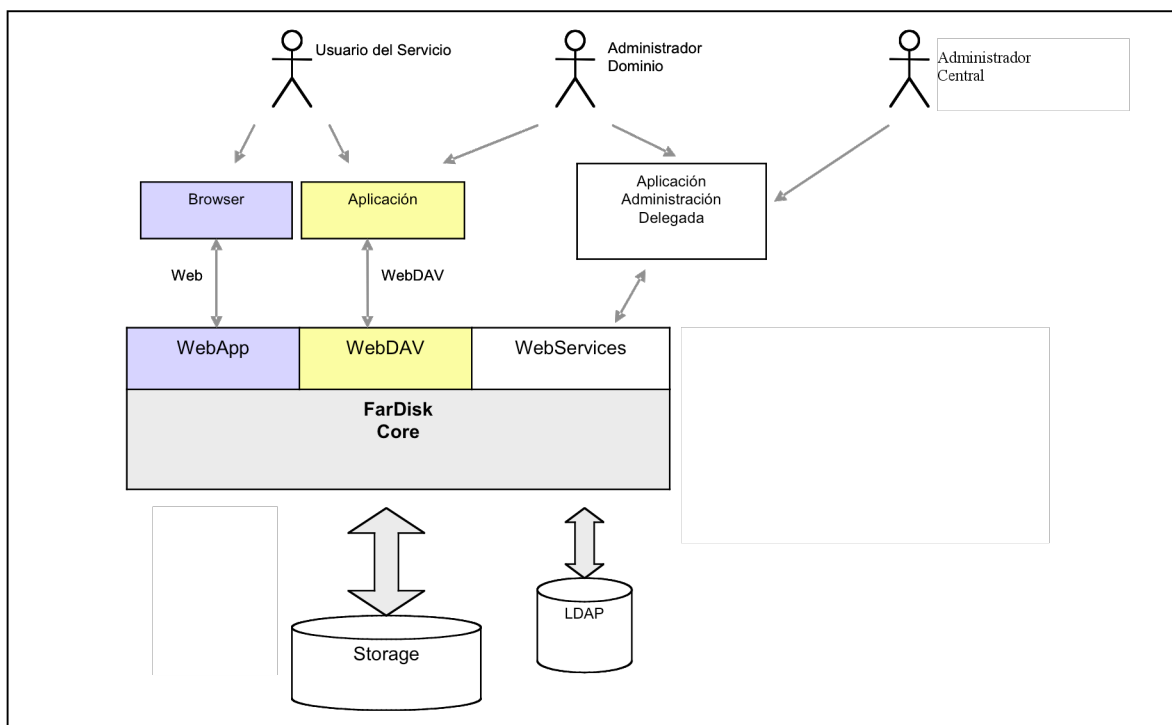
**Figura 15:** Flujo de llamada a webService con filtro aplicado

## CAPITULO 8. RESULTADO

A continuación se presenta el resultado final obtenido de la implementación del sistema de almacenamiento remoto. Se presentará la vista de las principales funcionalidades y componentes del sistema logrado.

### 8.1 Descripción general del sistema

La figura 16 muestra una visión general del sistema. En ella se ven las componentes que dan la funcionalidad a la aplicación y las interacciones y accesos al sistema.



**Figura 16:** Visión general del sistema

### 8.1.1 Interfaz de usuarios

La figura 17 muestra la interfaz principal de acceso de los usuarios al sistema. En la misma figura se distinguen los sectores descritos en la etapa de implementación, visualizándose ahora como toman posición dentro de la interfaz.

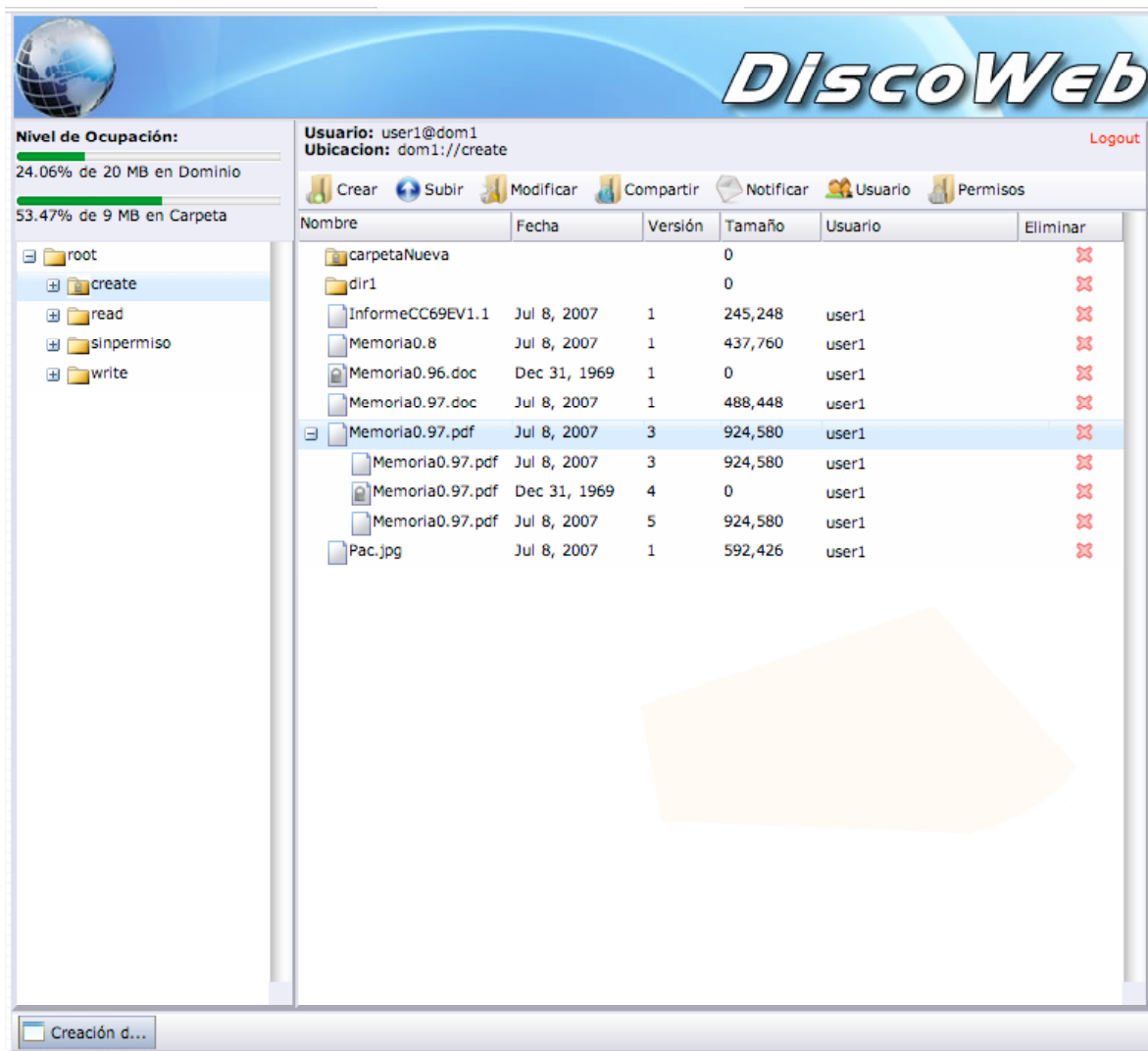


Figura 17: Interfaz Principal

## 8.1.2 Subida de archivos

En la figura 18 se ve la interfaz de subida de archivos. Esta interfaz se despliega como una ventana emergente dentro de la interfaz principal.

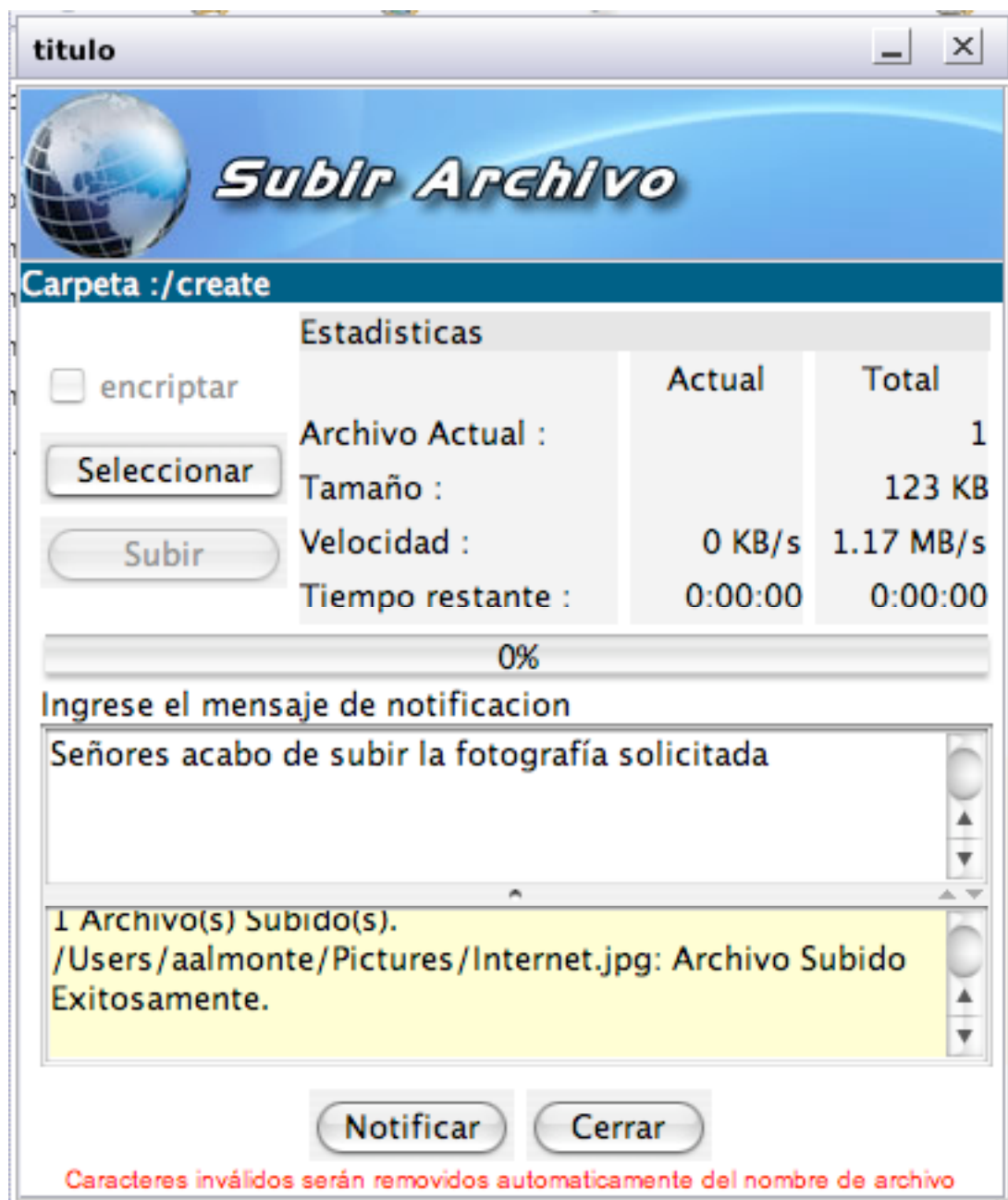
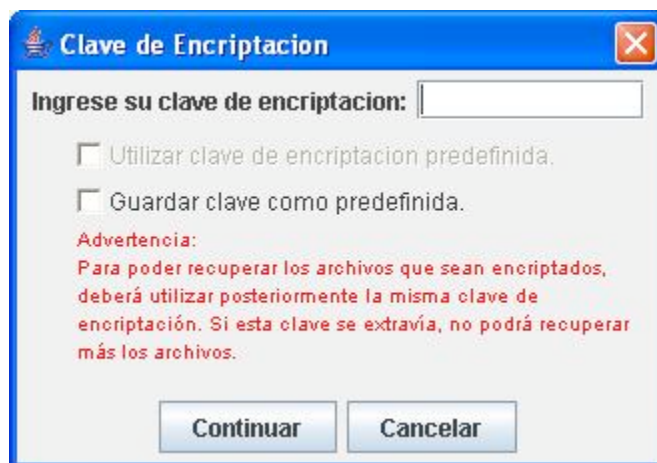


Figura 18: Interfaz de subida de archivos



Al seleccionar la opción encriptar, para subir un archivo encriptado, se despliega la ventana de solicitud de contraseña mostrada en la figura 19. También se ve en esta figura la opción de recordar la contraseña ingresada.

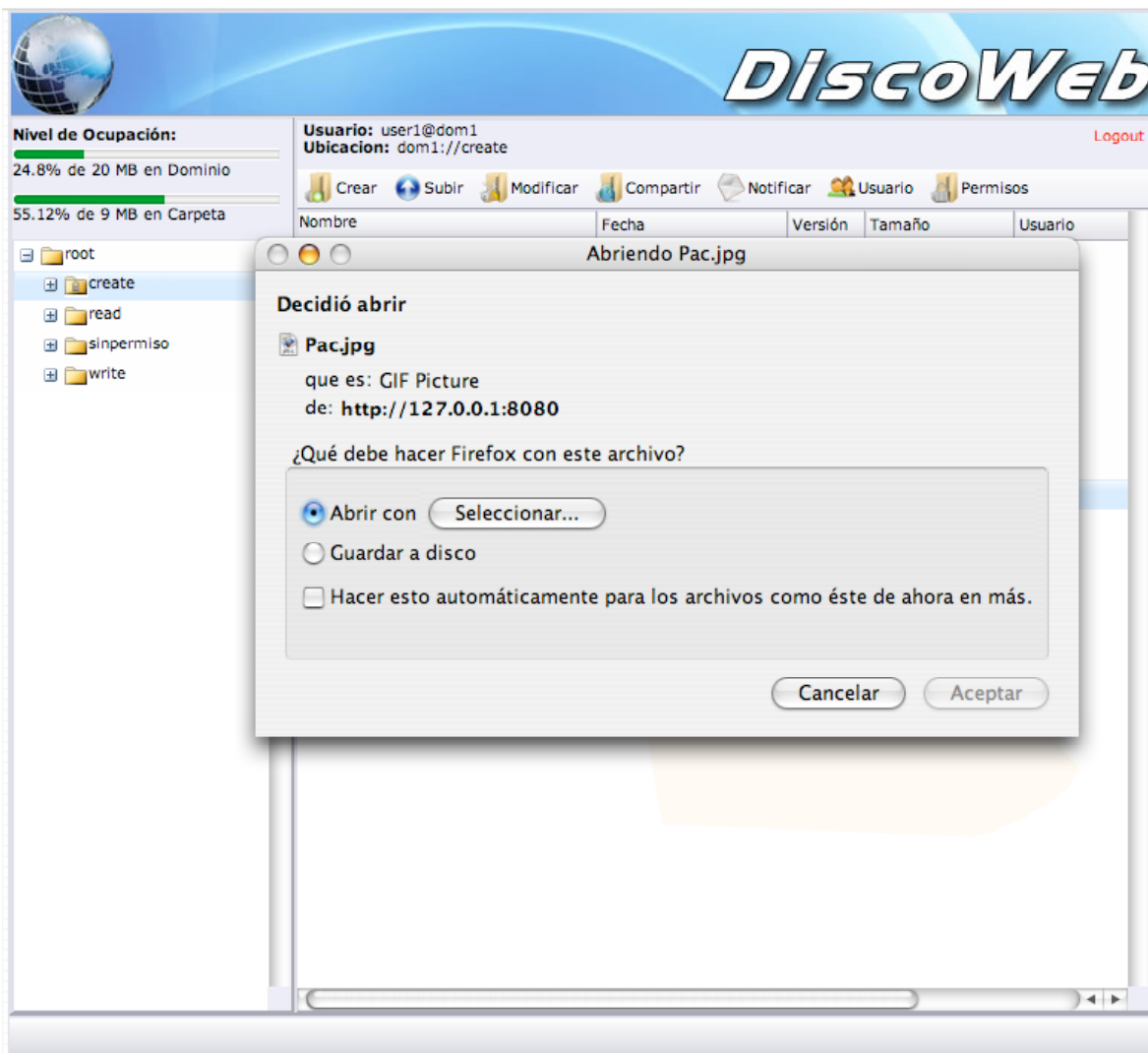


**Figura 19:** Solicitud de contraseña para encriptación

Si el usuario selecciona la opción para guardar la contraseña como predefinida, se almacena la contraseña de manera local y en las próximas operaciones de encriptación, tanto para encriptar como para desencriptar, el usuario tendrá la posibilidad de utilizar la contraseña guardada.

### 8.1.3 Descarga de archivos

Las descargas de archivos se dividen en dos tipos, descargas de archivos encriptados y descargas de archivos no encriptados. Para el caso de un archivo no encriptado, la descarga se comporta como si estuviera descargando un contenido desde cualquier pagina. La figura 20 muestra la descarga simple de un archivo no encriptado.



**Figura 20:** Descarga simple de un archivo

Para el caso de la descarga de un archivo encriptado, se despliega la interfaz mostrada en la figura 21. Esta interfaz solicitará la contraseña para desencriptar el archivo como se muestra en la figura 22. Además desde la interfaz de descarga se permite seleccionar el lugar de destino para el archivo a descargar.



Figura 21: Interfaz de descarga de archivos

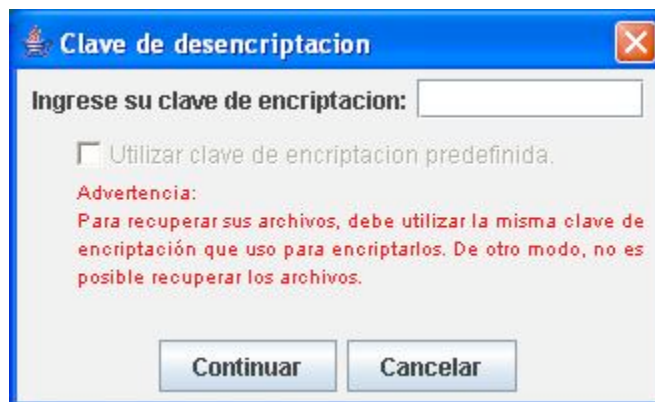
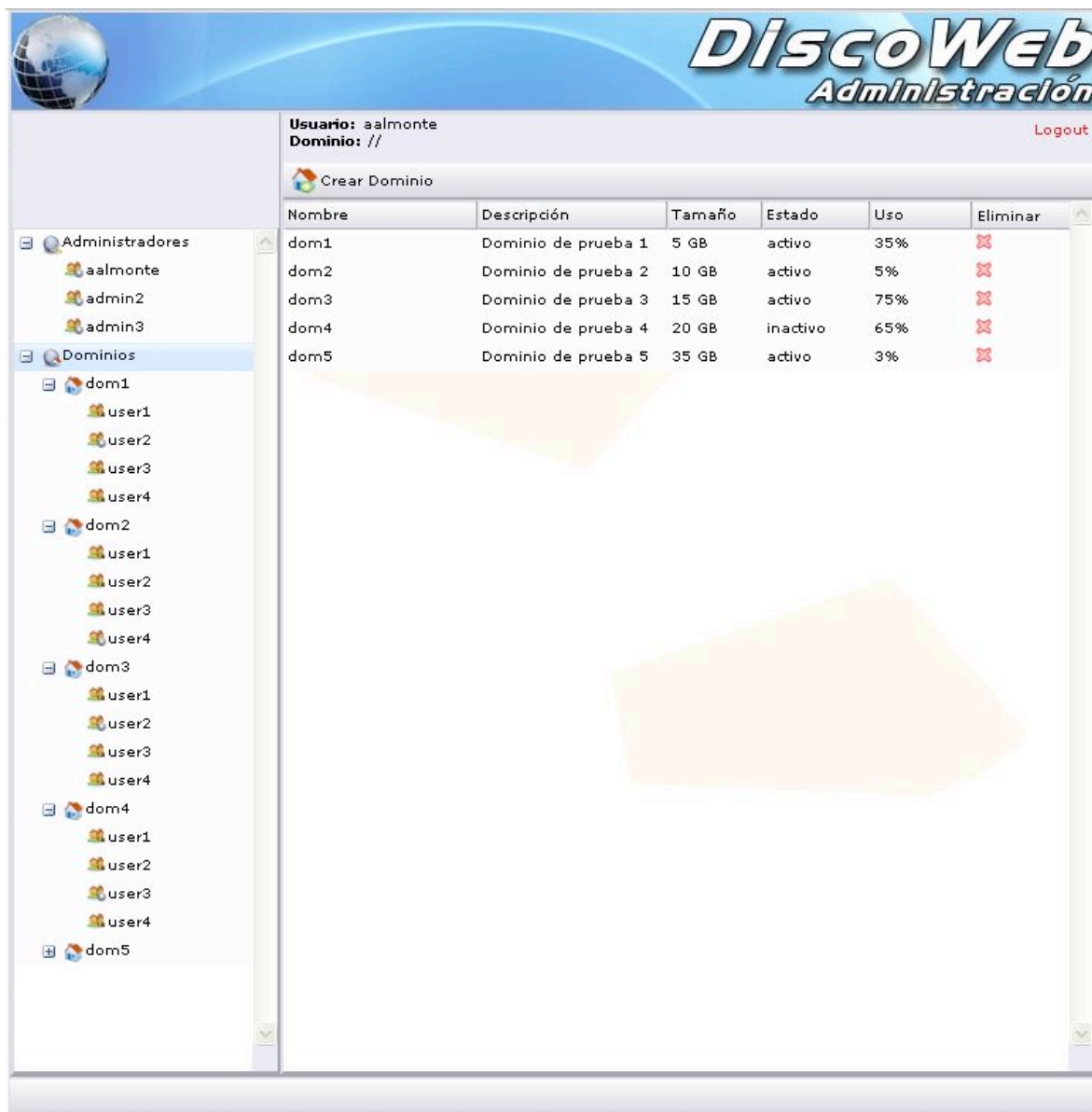


Figura 22: Solicitud de contraseña para descriptación

#### 8.1.4 Interfaz de administración

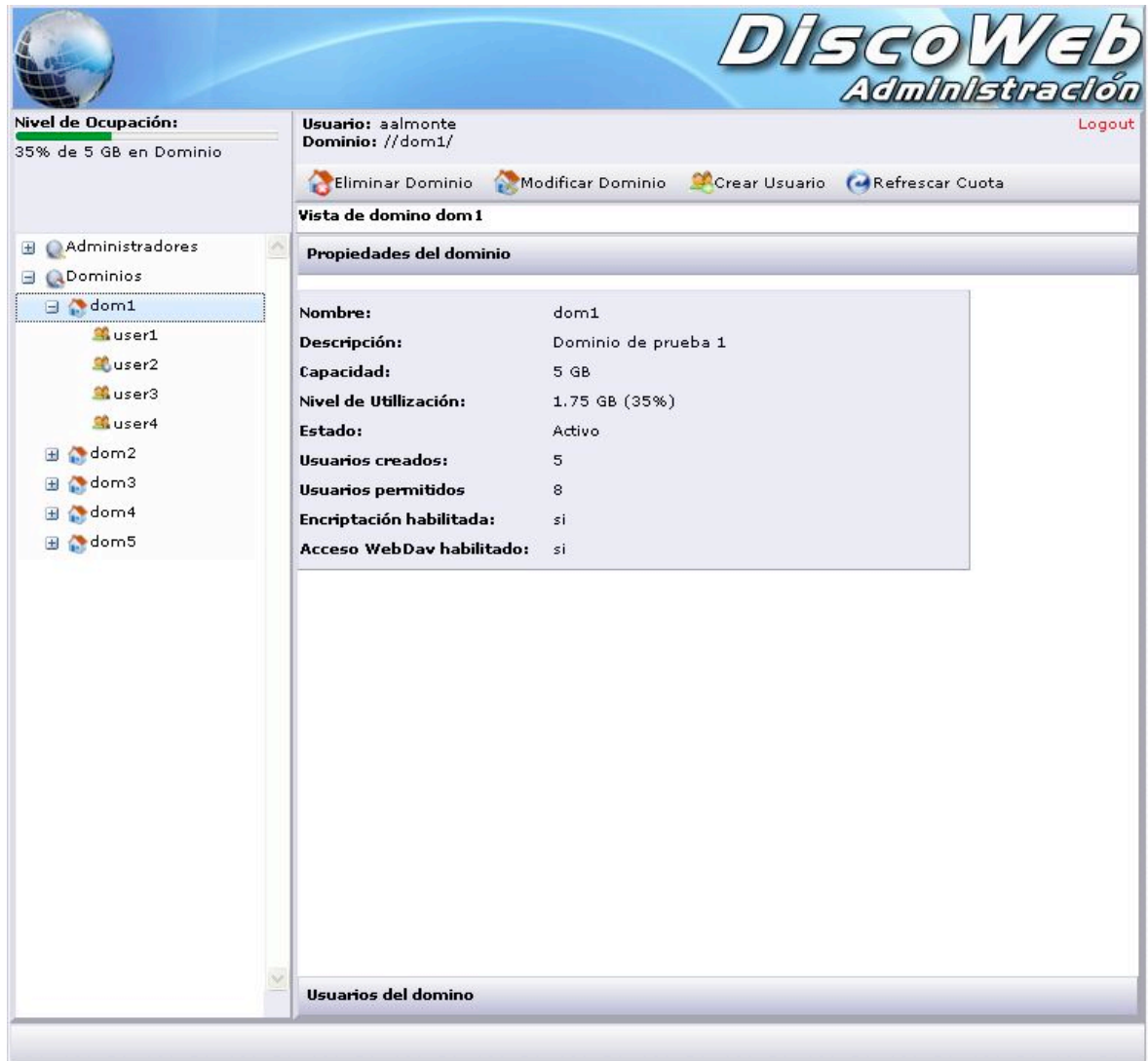
El ingreso a la interfaz de administración se diferencia en los casos de usuarios administradores de plataforma y administradores de dominio. Para el primer caso todas las opciones de la interfaz están disponibles, mientras que para un administrador de dominio

solo estarán habilitadas las operaciones que realicen cambios en el interior de su dominio, sin la posibilidad de modificar las características del dominio al que pertenece. La figura 23 muestra la interfaz de administración, mas vistas de la misma interfaz se aprecian en las figuras 24 y 25.



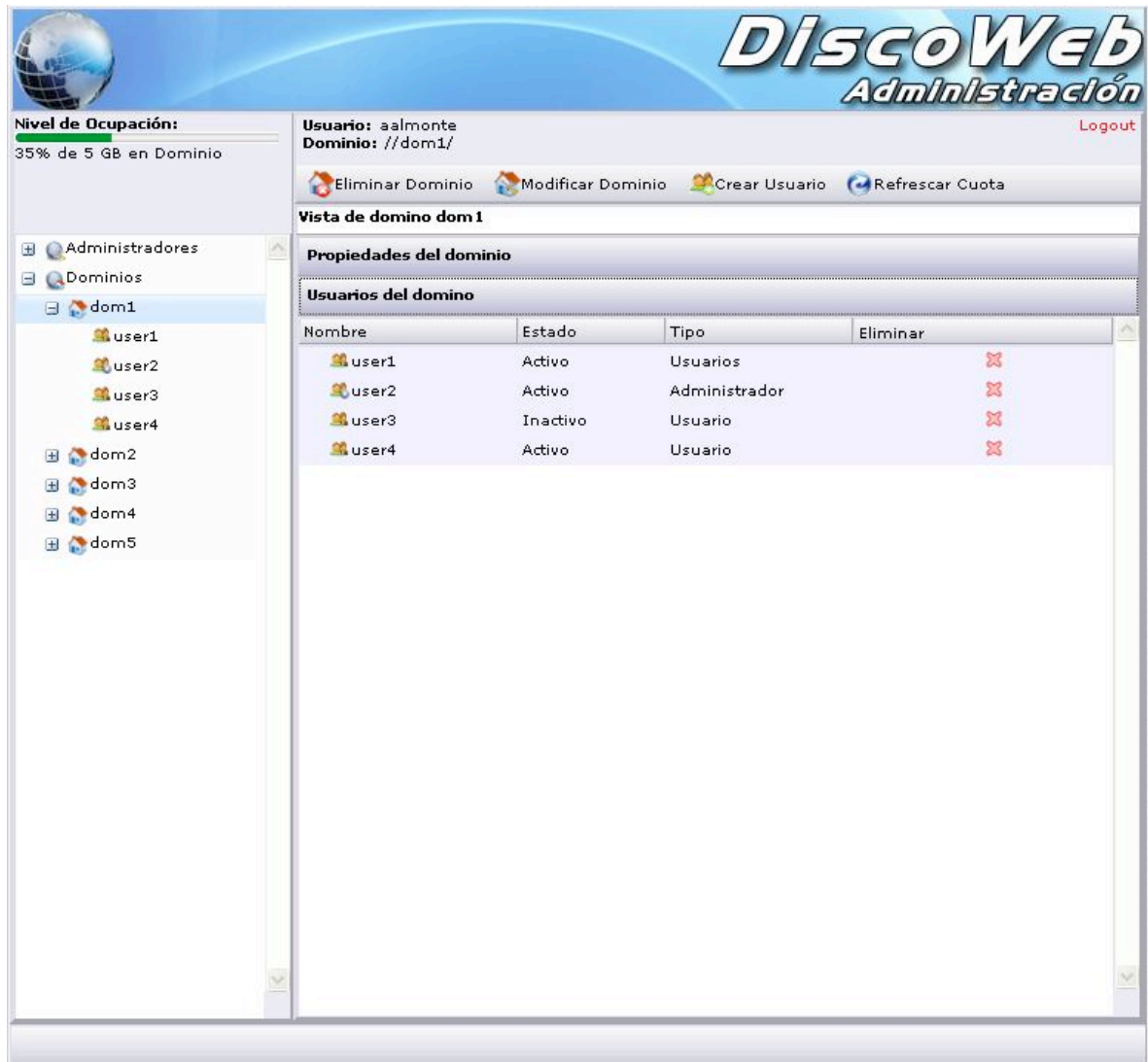
**Figura 23:** Interfaz de Administración

Como se ve en la figura 23 la interfaz de administración posee un árbol de navegación para recorrer los distintos niveles de la estructura. Además, existe un apartado especial en el árbol de navegación para los administradores de plataforma y su mantención.



**Figura 24.1:** Vista de dominio en interfaz de administración.

La figura 24.1 muestra la vista de detalle de un dominio en particular en la interfaz de administración. Esta vista se compone de dos secciones, una en que se informan las propiedades y una segunda sección en donde se dan a conocer los usuarios pertenecientes al dominio consultado. La figura 24.2 muestra la segunda sección mencionada.



**Figura 24.2:** Vista de dominio en interfaz de administración.

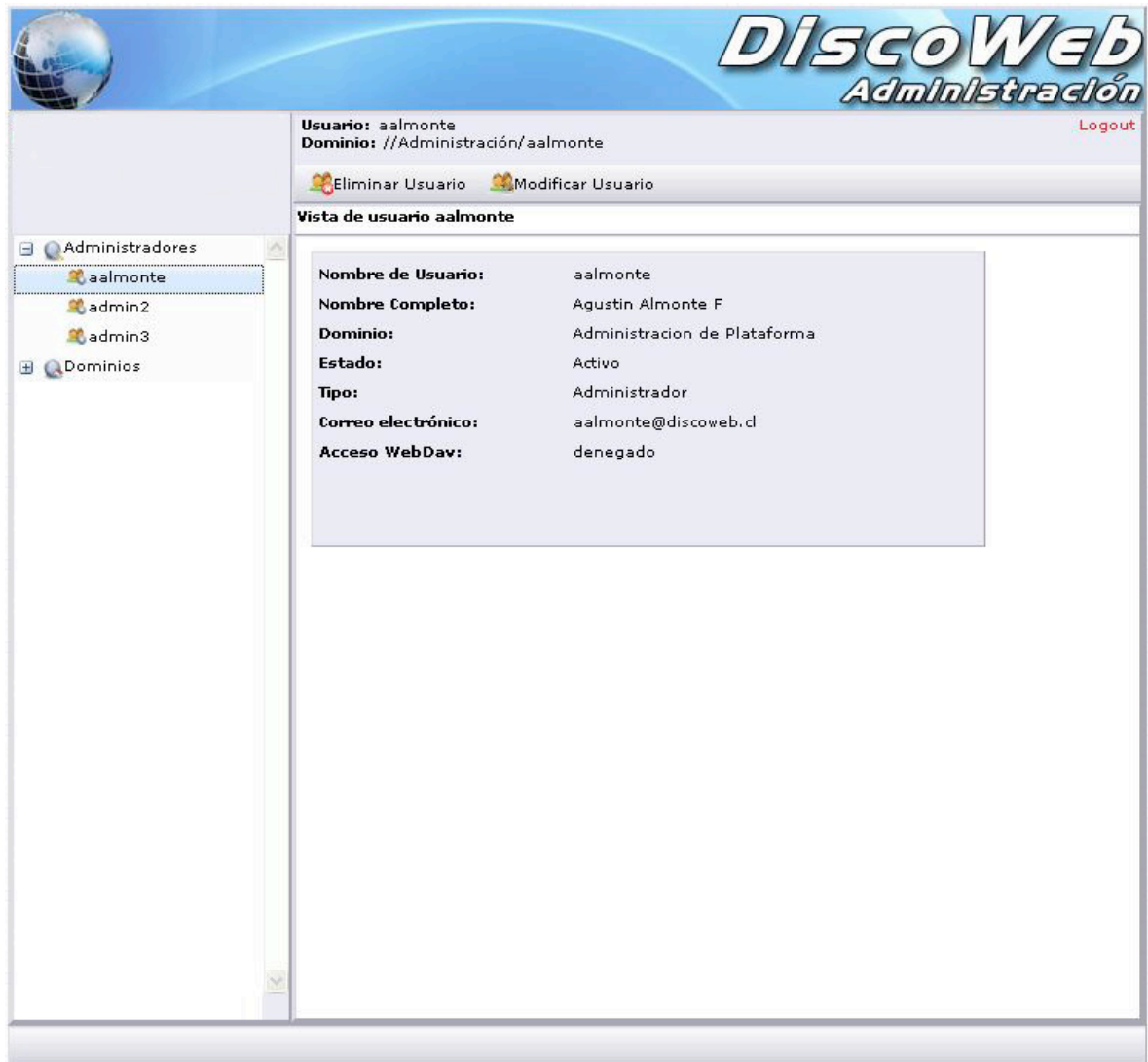
Dentro de la interfaz de administración, existen tres tipos de usuarios. Los usuarios administrados, los administradores de dominio y los administradores de plataforma. Si bien, estos usuarios cumplen roles diferentes en el sistema, el tratamiento implementado para su mantención es igual para todos. Las figuras 25.1 y 25.2 muestran la vista para usuarios utilizada en la interfaz de administración.

The screenshot displays the DiscoWeb Administration interface. At the top right, the logo "DiscoWeb Administración" is visible. The interface is divided into several sections:

- Top Left:** A globe icon and a "Nivel de Ocupación:" section showing "35% de 5 GB en Dominio".
- Top Center:** User information: "Usuario: aalmonte" and "Dominio: //dom1/user2". A "Logout" link is on the right.
- Top Right:** Action buttons: "Eliminar Usuario" and "Modificar Usuario".
- Left Panel:** A tree view showing a hierarchy: "Administradores" > "Dominios" > "dom1" > "user1", "user2", "user3", "user4". Below "dom1" are "dom2", "dom3", "dom4", and "dom5". "user2" is selected.
- Main Content Area:** Titled "Vista de usuario user2", it displays the following user details:

<b>Nombre de Usuario:</b>	user2
<b>Nombre Completo:</b>	Usuario dos
<b>Dominio:</b>	dom1
<b>Estado:</b>	Activo
<b>Tipo:</b>	Administrador
<b>Correo electrónico:</b>	user2@discoweb.d
<b>Acceso WebDav:</b>	permitido

**Figura 25.1:** Vista de usuarios en interfaz de administración

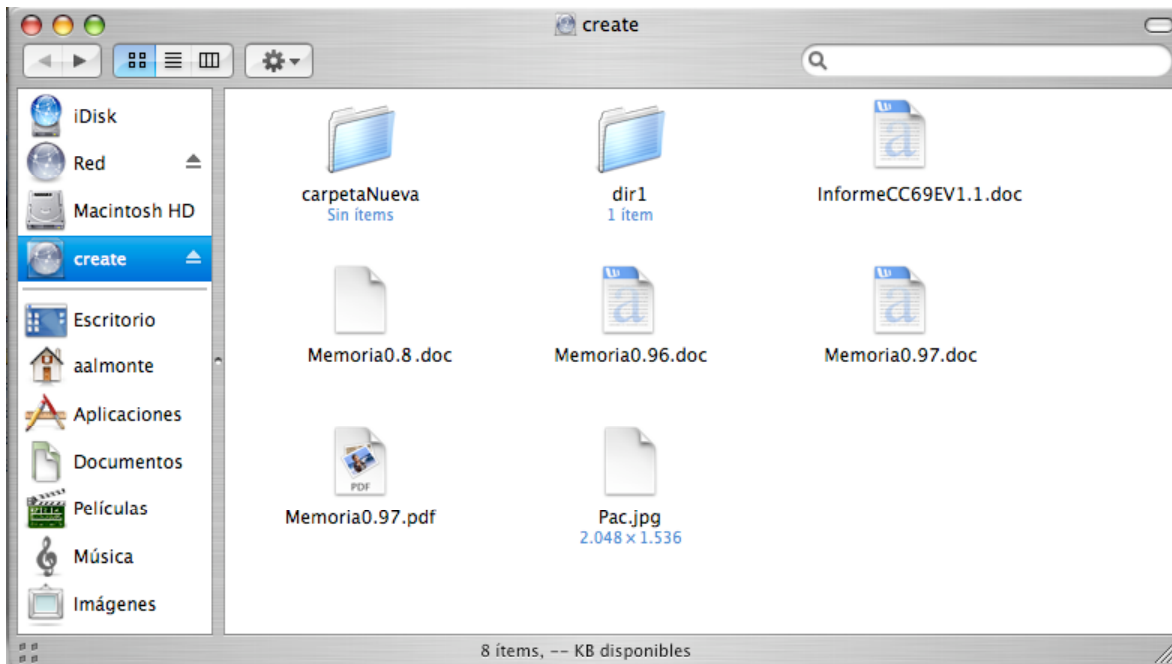


**Figura 25.2:** Vista de administradores en interfaz de administración

### 8.1.5 Vista del disco remoto como unidad local

Utilizando la implementación del servidor webDav incluido en el sistema, se puede acceder a los contenidos de las carpetas remotas, obteniendo una vista local en el sistema operativo, tal como si se tratara de un disco mas conectado a nuestro computador. La figura 26 muestra como se visualiza el contenido de la carpeta remota en el explorador de archivos.





**Figura 26:** Vista de carpeta remota como local.

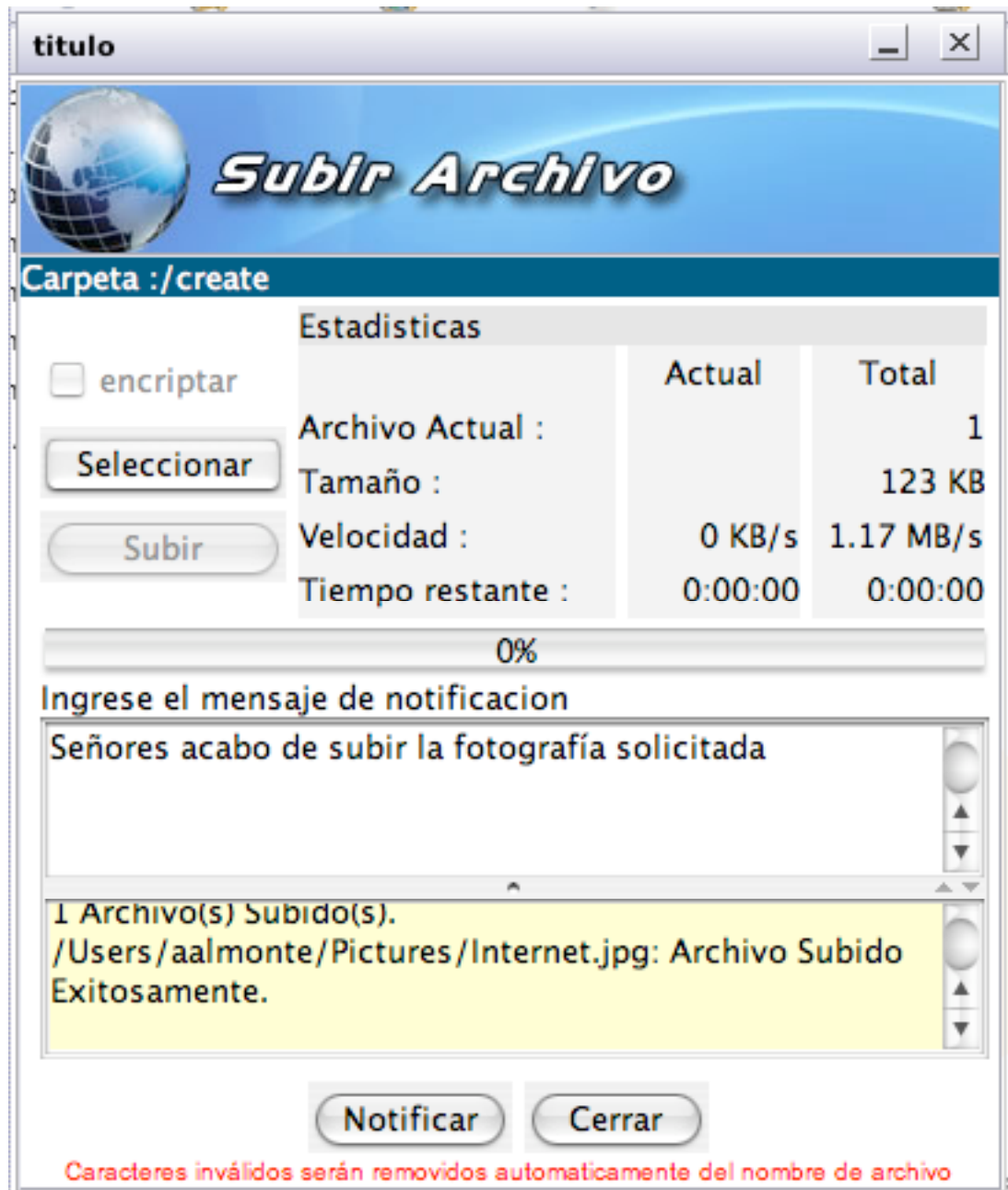
## 8.2 Funcionalidades destacadas

A continuación se detallan algunas de las funcionalidades destacadas incluidas en el sistema.

### 8.2.1 Envío de notificaciones por actualización

El envío de notificaciones es una de las operaciones destacadas en el sistema debido a que da la posibilidad de establecer comunicación entre los usuarios. En el sistema existen dos formas de enviar notificaciones: al momento de subir un archivo nuevo utilizando la interfaz de subida de archivos, y mediante la opción de notificación presente en la barra de operaciones de la interfaz.

La figura 27 muestra el momento en que se envía una notificación de actualización de contenidos luego de subir un archivo nuevo.



**Figura 27:** Envío de notificación

La notificación lanzada por esta interfaz será entregada a todos los usuarios inscritos en la lista de notificaciones para esta carpeta. La forma de inscripción en la lista de notificación se muestra mas adelante.

## 8.2.2 Envío de notificaciones arbitrarias

Una segunda opción de notificación es dada por la operación de notificación dispuesta en la barra de operaciones de la interfaz. La figura 28 muestra el uso de esta interfaz para enviar un aviso a ciertos destinatarios sobre cambios en la carpeta visualizada.



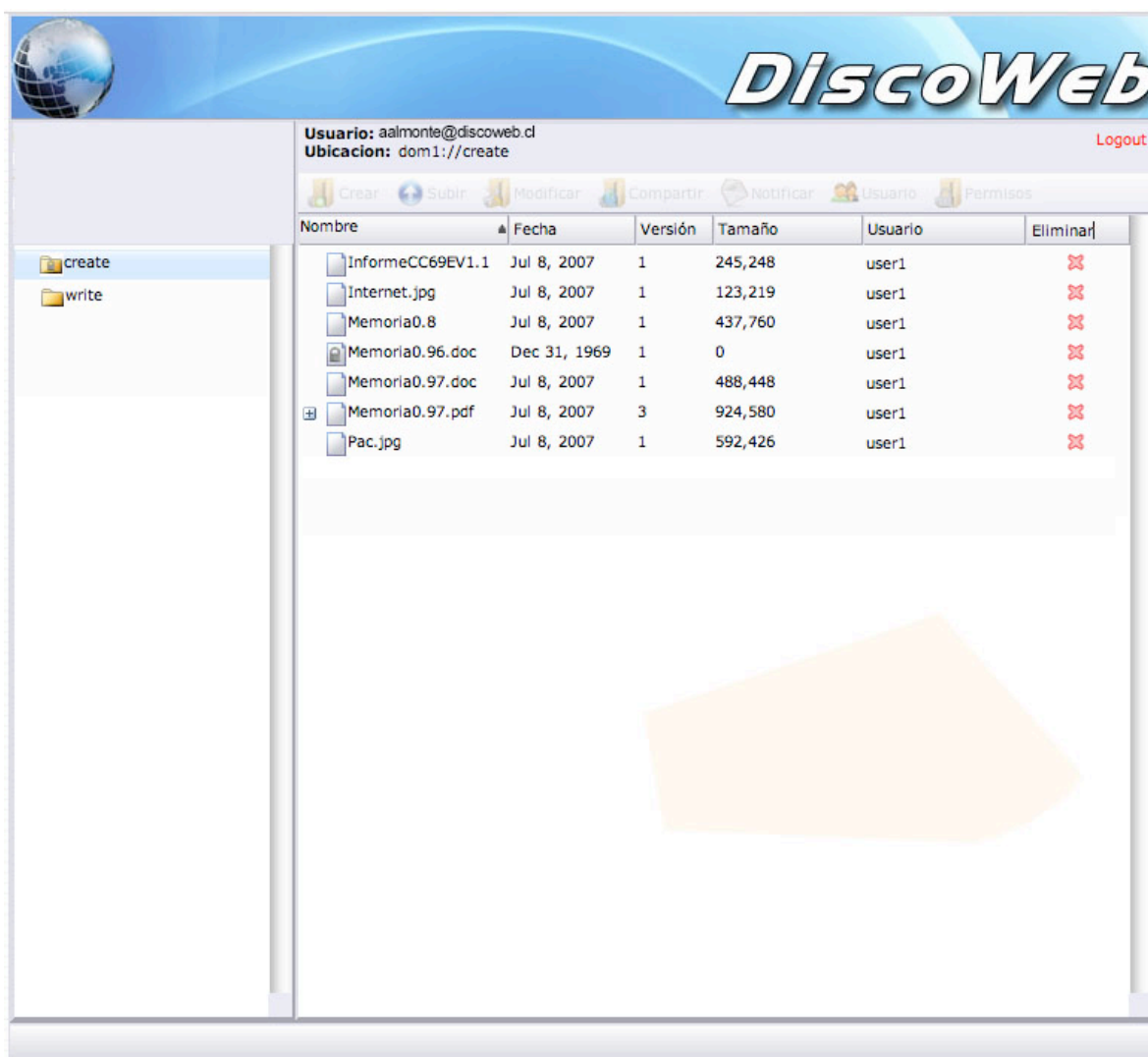
**Figura 28:** Envío de invitación

La idea de estas notificaciones es acelerar y facilitar la interacción entre los usuarios del sistema, brindando una herramienta de contacto. Las notificaciones enviadas consisten en correos electrónicos enviados con links directos hacia el contenido del notificado. Los links enviados son del estilo:

<http://www.discoweb.cl/visitor.jsp?email=aalmonate@gmail.com&token=gNuLm4bcCNGzxW9mna3cMw%3D%3D&uri=entel://create>

En donde se indica el correo electrónico, la carpeta a visitar y un token, generado a partir de los parámetros anteriores, para verificar la validez de la invitación a la carpeta.

La figura 29 muestra la vista generada por el sistema ante el ingreso de un usuario invitado por un link.



**Figura 29:** Vista de invitación

En esta vista el usuario solo ve los contenidos que le fueron compartidos y no puede realizar operaciones sobre la carpeta remota, solo descargar los archivos.

### 8.2.3 Compartir Carpeta

Además del envío de links en forma de invitaciones, la interfaz provee la capacidad de habilitar un acceso directo, mediante una url, a los contenidos de una carpeta. La figura 30 muestra esta posibilidad en la interfaz de habilitación.



**Figura 30:** Compartir carpeta

Al habilitar la opción mostrada en la figura 30, se da la posibilidad de utilizar una url fija por donde los personas externas al sistema podrán ingresar y ver los contenidos almacenados en la carpeta remota. Este ingreso puede limitarse o no con el uso de una contraseña especificada en la misma interfaz.

La figura 31 muestra la interfaz de configuración de la lista de correos electrónicos para notificación ante cambios mencionada en el punto 8.2.1.



**Figura 31:** Configuración de lista de notificación

## **CAPITULO 9. Conclusiones y discusión**

El sistema logrado cumple con los requisitos planteados convirtiéndose en una real competencia para los sistemas actuales al reunir las funcionalidades que estos ofrecen, agregando nuevas, dando una orientación al trabajo grupal, y utilizando los últimos avances en interfaces Web para crear una interfaz con una vista y funcionamiento atractivo para los usuarios.

La orientación grupal lograda en el sistema facilita la interacción entre los usuarios al agregar operaciones agregadas al proceso de almacenamiento de información que permiten notificar e invitar a visualizar los contenidos a otros usuarios, reduciendo el tiempo de transferencia de conocimientos entre los integrantes de la organización.

Por otro lado, la generación de listas de notificación, encriptación de archivos y determinación de permisos sobre carpetas permite la creación de grupos de interés seguros sobre sectores de la información almacenada, brindando confianza sobre el almacenamiento aun cuando la información se encuentre fuera de los dominios de la empresa.

### **9.1 Alcances de la Solución Lograda**

La solución desarrollada al tener un enfoque grupal y una estructura jerárquica de usuarios y dominios esta pensada para ser aplicada en organizaciones. Sin embargo, se podría decir que no importando el numero de usuarios puede aplicarse la solución a cualquier organización, incluso a una unipersonal en donde no se hará uso de la orientación al trabajo grupal pero la labor de respaldo seguirá siendo útil.

Además del mencionado uso como plataforma de almacenamiento y respaldo, al sistema desarrollado también se le puede dar uso como un sistema para compartir contenidos. Esto gracias a la posibilidad de extender invitaciones a los contenidos a personas no usuarios del sistema, de esta forma alguien podría subir sus archivos al disco

remoto y extender invitaciones a otras, no pertenecientes al sistema, para que los descarguen.

Si bien, las interfaces implementadas para el sistema son específicas para solucionar el problema de respaldo de información, la estructura definida, el conjunto de funciones, el soporte WebDav y servicios remotos implementados dan la posibilidad para crear aplicaciones complementarias al servicio de respaldos. Aprovechando el ambiente generado en el servidor se pueden implementar nuevas interfaces, por ejemplo una interfaz para el almacenamiento de contenidos multimedia como fotografías o videos que serian compartidos y vistos dentro de una comunidad.

## 9.2 Tareas a seguir

Una característica en las nuevas aplicaciones Web disponibles en la red es la constante actualización que poseen, de ahí que se diga que estas nuevas aplicaciones siempre estarán en estado beta. Las aplicaciones aprovechan el control sobre la distribución que da Internet para realizar cambios de tecnología o de funcionalidades incluidas.

En base a lo anterior, el desarrollo de un sistema como el logrado debería estar constantemente sujeto a cambios para mantenerse competitivo. A continuación se presentan ideas de tareas en las que se podría continuar para mejorar el sistema:

**Sincronización automática:** La construcción de una aplicación de respaldo automática de archivos es bastante útil al momento de brindar un servicio de respaldos.

**Nuevas funcionalidades:** El sistema se diseño para poder brindar un buen servicio de respaldos y dar una base para compartir la información almacenada entre los usuarios. Ya dispuesta la base del sistema, se puede trabajar en el desarrollo de nuevas funcionalidades que busquen agregar mayor valor al uso del sistema. Por ejemplo, podría se trabajar sobre las siguientes ideas:



- **Reuniones en línea:** Dar soporte para realizar reuniones entre los usuarios utilizando un sistema de Chat en base a invitaciones. La reunión podría quedar archivada como un archivo mas dentro del sistema para su posterior revisión.
- **Mensajería entre usuarios:** Permitir la comunicación entre los usuarios del sistema. Los mensajes serían entregados al instante en el caso de usuarios logeados o en el próximo inicio de sesión para los no conectados.
- **Edición compartida de documentos:** Dar soporte para la edición de contenidos almacenados en el disco remoto por varios usuarios a la vez. Esto iría como una funcionalidad extra al soporte de versiones.

## Bibliografía y referencias

- [1] <http://www.gemelostorage.com> Gemelo Storage Solutions
  
- [2] <http://engdemo.internetdisk.com> Novell InternetDisk
  
- [3] <http://www.webdav.org/> Centro de documentación WebDAV
  
- [4] RFC 2518: <http://www.ietf.org/rfc/rfc2518.txt>
  
- [5] RFC 3253: <http://www.ietf.org/rfc/rfc3253.txt>
  
- [6] RFC 3648: <http://www.ietf.org/rfc/rfc3648.txt>
  
- [7] RFC 3744: <http://www.ietf.org/rfc/rfc3744.txt>
  
- [8] RFC 4316: <http://www.ietf.org/rfc/rfc4316.txt>
  
- [9] RFC 2518: HTTP Extensions for Distributed Authoring—WEBDAV. Y. Goland, E. Whitehead, A. Faizi, S.R. Carter, D. Jensen.
  
- [10] <http://www.openldap.org>
  
- [11] OpenLDAP Quick Start Guide.

- [12] Java.Web Services. Davis A. Chappel & Tyler Jewell . O'Reilly
  
- [13] Java.And Soap. Robert Englander. O'Reilly
  
- [14] Enterprise JavaBeans 3<sup>rd</sup> Edition. Richard Monson-Haefel. O'Reilly
  
- [15] AXIS, Next Generation Java SOAP. Romin Irani, S. Jeelani Basha. Wrox
  
- [16] Java & XML 2<sup>nd</sup> Edition. Brett McLaughlin. O'Reilly
  
- [17] Java Cryptografy. Jonathan Knudsen. O'Reilly
  
- [18] Java Security 2<sup>nd</sup> Edition. Brett McLaughlin. O'Reilly
  
- [19] Java Server Pages 2<sup>nd</sup> Edition. Hans Bergsten. O'Reilly
  
- [20] Java Servlet Programing. Jason Hunter, William Crawford. O'Reilly
  
- [21] Knowledge Management Systems:Issues, Challenges, And Benefits. Maryam Alavi.  
Communications of the Association for Information Systems, Volume1 Article 7.  
February 1999.
  
- [22] Ajax: Creating Web Pages with Asynchronous JavaScript and XML.  
Edmond Woychowsky
  
- [23] QooxDoo AJAX Framework  
[www.qooxdoo.org](http://www.qooxdoo.org)

[24] TIBCO General Interface

[http://www.tibco.com/software/rich\\_internet\\_application/general\\_interface/](http://www.tibco.com/software/rich_internet_application/general_interface/)

[24] Web Services - AXIS

<http://ws.apache.org/axis>

[25] Knowledge management and knowledge management systems: conceptual foundations and research issues

Alavi, M., Leidner, D.E. (2001), MIS Quarterly Review, Vol. 25 No.1, pp.107-36.

[26] What Is Web 2.0. Design Patterns and Business Models for the Next Generation of Software.

Tim O'Reilly. 2005

<http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>