



Universitat
Autònoma
de Barcelona



2755-2: Confecció del portal de l'investigador universitari

Memòria del Projecte Fi de Carrera
d'Enginyeria en Informàtica
realitzat per
David Garcia Machado
i dirigit per
Ramon Grau Sala
Bellaterra, 22 de Juny de 2011

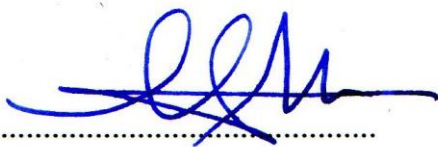


El sotasignat Ramon Grau Sala Professor/a de l'Escola Tècnica Superior d'Enginyeria de la UAB,

CERTIFICA:

Que el treball a què correspon aquesta memòria ha estat realitzat sota la seva direcció per en **DAVID GARCIA NACHADO**

I per tal que consti firma la present.

Signat: 

Bellaterra, 20 de juny de 2011

Quiero agradecer a mis padres y hermanas el apoyo que me han dado para seguir mis ideales, y a mi novia por decir que soy capaz de hacer lo irreal en real.

Gracias.

Índice de los contenidos

1.	Presentación.....	7
1.1.	Introducción	7
1.2.	Estado del arte.....	7
1.3.	Objetivos	8
2.	Análisis de requisitos	9
2.1.	Introducción	9
2.2.	Usuarios (roles).....	9
2.3.	Administración	10
3.	Diseño	11
3.1.	Introducción	11
3.2.	Escritorio Web (Web Desktop)	12
3.3.	Justificación del uso e implementación de un Web Desktop	14
3.4.	Web Desktop Existentes	16
3.5.	Motivo de la nueva implementación	16
3.6.	El sistema UAB_OS	17
3.7.	Diseño de las diferentes interfaces.....	18
4.	Implementación	21
4.1.	Introducción	21
4.2.	Módulo del núcleo.....	21
4.3.	Módulos de gestión	22
4.3.1.	Módulo Gestor de Ventanas.....	22
4.3.2.	Módulo Gestor de Ficheros	22
4.4.	PHP, SQL, JavaScript y AJAX.....	23
4.4.1.	PHP	23

4.4.1.1.	Acceso a Base de Datos	23
4.4.1.2.	Acceso a los ficheros NFS.....	23
4.4.2.	SQL.....	24
4.4.2.1.	Tabla USERS	24
4.4.2.2.	Tabla GROUPS.....	24
4.4.3.	JavaScript.....	24
4.4.4.	Ajax.....	24
4.5.	Aspectos Visuales	25
4.6.	Gestión de los ficheros	29
4.6.1.	Desktop.....	30
4.6.2.	Programs	32
4.7.	Entorno del Servidor.....	33
5.	Flujo de datos	34
6.	Presupuesto	35
6.1.	Valoración de Gastos	35
6.2.	Planificación temporal	35
7.	Conclusiones y Ampliaciones	36
7.1.	Conclusiones.....	36
7.2.	Ampliaciones	38
8.	Bibliografía	39
8.1.	Enlaces Web	39
8.2.	Libros de consulta.....	39
8.3.	Software utilizado.....	40
9.	ANEXOS	41
9.1.	ANEXO I: Requisitos	41
9.1.1.	Requisitos Funcionales	41

9.1.1.1.	Requisitos funcionales de usuario	41
9.1.1.2.	Requisitos funcionales de servicios.....	41
9.1.2.	Requisitos No Funcionales	42
9.1.2.1.	Requisitos plataforma.....	42
9.1.2.2.	Requisitos de seguridad.....	42
9.1.2.3.	Requisitos de recursos	42
9.1.2.4.	Requisitos de disponibilidad	42
9.1.2.5.	Requisitos de rendimiento.....	42
9.1.3.	Casos de uso	43
9.1.3.1.	Escenario 1: Identificación del usuario	43
9.1.3.2.	Escenario 2: Uso del sistema de ficheros	43
9.1.3.3.	Escenario 3: Uso de las aplicaciones	44
9.1.3.4.	Escenario 4: Uso de la interacción de las aplicaciones con el núcleo.....	45
9.2.	ANEXO II: Sistemas Desktop Web Actuales	46

1. Presentación

1.1. Introducción

Este proyecto está enfocado en la mejora del nivel de comunicación y uso de funcionalidades en el entorno de los investigadores universitarios. Para ello deberemos ver que software usan y que medios de comunicación utilizan.

El trabajo de investigación requiere tener unas bases solidas de información y test coherentes. Los investigadores requieren de un conjunto de herramientas para llevar de forma organizada y coherente sus investigaciones.

En este entorno, los usuarios dependen de varios programas informáticos para contener de forma eficaz y coherente la información. Los aplicativos usados son de lo más variado posible, desde sistemas que gestionan versiones hasta software con un alto nivel de computo.

Dado que los investigadores, normalmente, son a su vez profesores de materias universitarias, tiene una cierta necesidad de gestionar los temarios junto con los temas de investigación. Esto puede llevar a una cierta confusión si no lo tienen todo bien organizado. Por eso este proyecto busca una forma de coordinar la información de las investigaciones, junto con la parte de docencia, aportando todas aquellas necesidades no cubiertas por el software existente.

1.2. Estado del arte

En la actualidad los investigadores usan aplicaciones de ofimática, como podrían ser los paquetes Windows Office o el Open Office. Dado que las investigaciones requieren de un formato concreto para su publicación en congresos y/o revistas, sale más eficaz el uso de herramientas como pueden ser la programación en Latex.

Este sistema hace que los investigadores se centren en el estudio que están realizando, sin necesidad de perder tiempo en dejar los informes en un formato concreto. El formato, normalmente, lo publican las revistas y/o congresos para que puedan tener el mismo estilo todos los documentos que se les envían.

Una vez creado un documento en Latex, lo guardan como una versión del documento de la propia investigación. Para ello, disponen de forma centralizada o de forma local un servicio de versiones, para mantener un historial de lo que se ha ido realizando durante el periodo en el que se está investigando.

A parte de todas estas herramientas de versionado para los documentos, requieren de algún canal de comunicación. Estos canales pueden ser servicios de mensajería instantánea, correo electrónico, llamadas por voz IP (VOIP), etc. Bastante difícil de centrar todos los contactos en un único medio de comunicación.

1.3. Objectivos

Este proyecto está enfocado en la creación de un sistema Web, que permita una interacción fluida y a demás, una usabilidad elevada. De esta forma se conseguirá que los investigadores no contemplen posibles pérdidas de información y una mejor accesibilidad a los recursos. A parte de contener ciertas funcionalidades que actualmente no disponen.

Para ello, se tendrán en cuenta que se utilice todo el potencial existente en la red. La mejor forma de hacerlo, será utilizando las siguientes tecnologías: Cloud Computing, Computación Distribuida y Modelo Básico de Cliente.

2. Análisis de requisitos

2.1. Introducción

Se han encontrado ciertos requisitos no cubiertos en su totalidad. Para ello, lo que se va hacer será un completo estudio de las herramientas usadas por los investigadores para el mantenimiento y compartición de los archivos propios y/o comunes, y como se comportaría nuestro sistema.

Para hacerlo de forma efectiva seguiremos el siguiente orden:

- ¿Qué amplitud comprenderá el Proyecto?

Aquí determinaremos, que aspectos son los que se tendrán en cuenta en nuestro proyecto y cuáles no son incluidos en él.

- ¿Qué requisitos funcionales se incluirán?

En este momento especificaremos el comportamiento que tendrá nuestro sistema al enfrentarse con las diferentes entradas de los usuarios.

- ¿Qué requisitos no funcionales se incluirán?

Para ello, se tendrán en cuenta ciertas políticas de la UAB, donde susodichos aspectos no implicarán una modificación directa en los requisitos funcionales.

- Casos de uso.

En este apartado, contemplaremos las diferentes interacciones que pueden tener los diferentes usuarios existentes.

Para ver todo el análisis de requisitos, en el ANEXO I está definido.

2.2. Usuarios (roles)

Desde un principio se pensó en la opción de separar por áreas de investigación, donde se determinaría que tipo de aplicaciones son las que tendrían asociados. Pero, para hacer viable el proyecto, se optó por crear un único rol, donde los usuarios dispondrán de todas las aplicaciones existentes en el entorno.

De esta forma disminuimos el volumen de validaciones, diferencias, etc., que nos podrían suponer un tiempo mucho mayor de lo disponible.

2.3. Administración

Para la administración del entorno Web Desktop, se ha optado por gestionarlo de forma directa en el servidor. Estaría correcta, la gestión mediante cuenta administrador, pero no se requiere de forma imprescindible.

Como parte importante, pero no obligada para este proyecto, sería recomendable realizar un pequeño “backoffice” que nos aportaría las siguientes funcionalidades:

Usuarios

- Agregar nuevos usuarios al entorno.
- Eliminar usuarios del entorno.
- Modificar datos de los usuarios.

Aplicaciones

- Agregar una nueva aplicación web.
- Vincular una aplicación.
- Desvincular una aplicación.
- Eliminar una aplicación.

Para poder decir que el entorno permite un canal para el administrador, se debería hacer una entrada única para él.

3. Diseño

3.1. Introducción

En este proyecto, se ha optado por utilizar una arquitectura basada en Cliente-Servidor. Donde el cliente, es decir el navegador Web en cuestión, genera solicitudes al servidor. Una vez el servidor recibe dichas solicitudes, las gestionará y generará las respuestas oportunas a cada una de ellas.

Los motivos que me han llevado a escoger esta arquitectura están basados principalmente en la mínima dependencia de los usuarios. Estos, no requieren disponer de unos recursos concretos. Únicamente requerirán disponer de un navegador Web.

Además, quitamos la dependencia de la instalación de software externo. Esto hace que sea más viable un acceso a los datos desde cualquier ordenador y/o dispositivo móvil, sin tener instalaciones en diferentes arquitecturas o sistemas.

Desde un principio, se han determinado que tecnologías serán las usadas en la implementación del proyecto, y son los siguientes lenguajes de programación: PHP, JavaScript y Ajax. Esta elección viene dada por uno de los requisitos estipulados anteriormente, donde se especifica la no utilización de lenguajes de programación que requieran de licencias. Esto no significa quedarse exento de incluir ciertos componentes en dicha tecnología.

En el mercado, existen muchos productos que nos satisfacen parte de los requisitos deseados. Hay aplicativos web desarrollados para la gestión de contenido. Este caso, no se ajusta del todo a lo que realmente se requiere, pero sería una buena opción. A pesar de ello, lo que mejor se adecua a nuestras expectativas son las aplicaciones Web denominadas Web Desktop, donde podemos dar todas las necesidades por cubiertas.

A continuación se detallará la viabilidad de aplicar dicha tecnología en el ambiente donde se centra este proyecto.

3.2. Escritorio Web (Web Desktop)

Un escritorio web (Web Desktop) es un entorno de escritorio en un navegador web, donde se integran aplicaciones web, servicios web, aplicaciones clientes-servidor, etc. Estos se comportan como en entorno real de escritorio. Disponiendo a su vez de una interfaz parecida a los sistemas operativos conocidos, como pueden ser Windows, Mac y Linux.

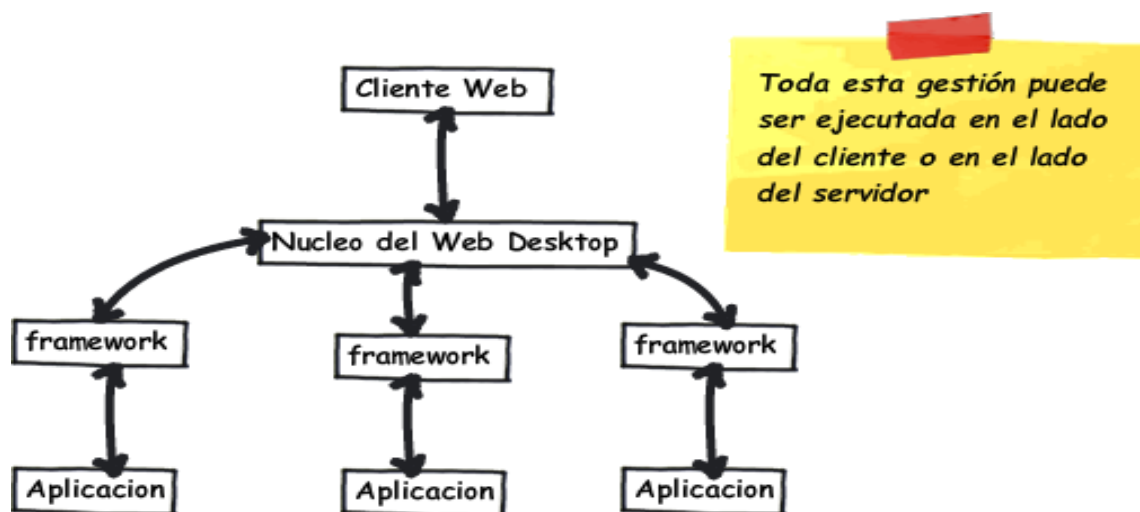
Estos sistemas se basan en un conjunto de herramientas, donde las más características son:

- Una selección de aplicaciones web que aportan funcionalidad al entorno.
- Una Base de Datos (BBDD) donde guardar la información relevante de las conexiones, mensajería interna, tareas, datos de los usuarios, etc.
- Una carpeta (normalmente en la misma máquina o en su defecto en un servicio web) donde los usuarios guardan su contenido.

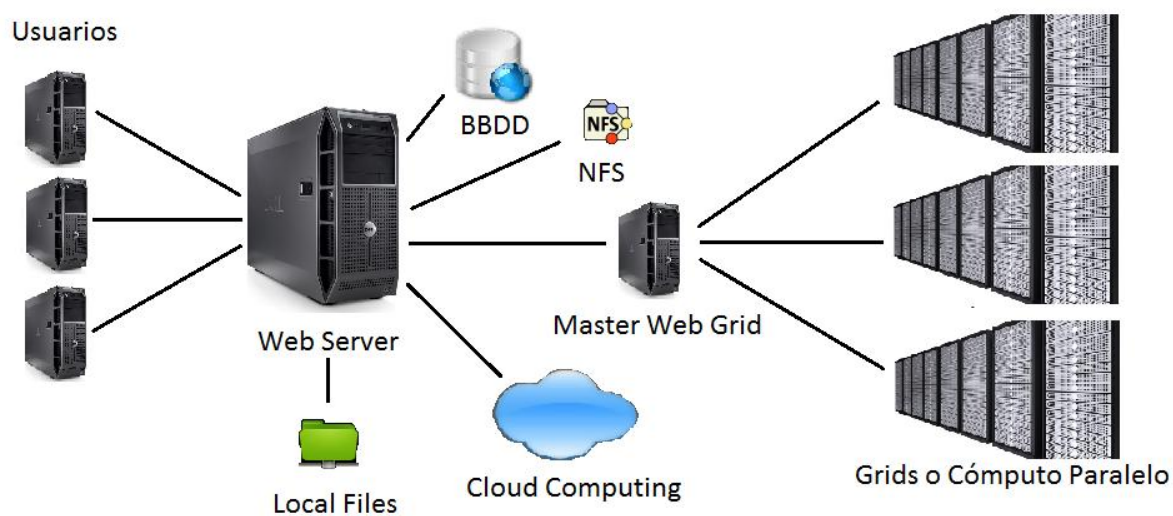
Para una buena gestión de los Web Desktop, requerimos tener un servicio FTP al servidor para poder subir los cambios realizados. También existe la opción de tener acceso directo con el servidor o conectarse mediante Terminal Server o SSH.

Para crear nuevas aplicaciones en los entornos Web Desktop, requerimos de la descarga del framework. Este aportará todas las funcionalidades del entorno a nuestra aplicación, y nos facilitará la comunicación del entorno con la nueva aplicación.

A continuación se muestra un ejemplo de la comunicación que se realiza en un Escritorio Web.



Para poder ver con mayor ampliación como se comportaría un entorno con las tecnologías Cloud Computing y Computación Distribuida, dado que el modelo básico ha sido explicado anteriormente, visualizamos la gestión que realizaría.



3.3. Justificación del uso e implementación de un Web Desktop

Existen muchos sistemas web que nos aportan la usabilidad deseada, donde pueden estar los Gestores de contenido (CMS).

Si comprobamos las diferentes versiones existentes de Gestores de contenido, podemos darnos cuenta que en ninguno de ellos puede darse la opción de incrementar la funcionalidad. Si que se pueden ampliar módulos, pero aplicaciones completas con interacción entre el CMS y dicha aplicación no se contempla. Por ello hace inviable la implantación de un sistema CMS para la mejora.

La siguiente opción nos reporta toda la funcionalidad deseada, junto con las especificaciones que se han detallado anteriormente. El uso de Web Desktops.

Estos sistemas conllevan una serie de ventajas a tener en cuenta.

Ventajas:

- Comodidad: Permite tener en cualquier ordenador con internet acceso al escritorio web.
- Hardware: No requiere un ordenador potente para ejecutar y/o desarrollar aplicaciones.
- Actualizaciones: No nos tenemos que preocupar de actualizar nuestro sistema, dado que de ello se encargan los administradores de la web.
- Software: No se requiere que los clientes instalen aplicaciones.
- Virus: Menos propensos a los ataques típicos, como virus, gusanos, etc.
- Información: Se guardan en sistemas robustos para una mayor seguridad.

A pesar de tener unas ventajas muy remarcadas, también contienen ciertos inconvenientes.

Inconvenientes:

- Software: No se puede instalar el software que deseemos como clientes, dado que de ello se encarga el administrador de la web. Y no siempre las aplicaciones contienen toda la funcionalidad deseada.
- Seguridad: Por motivos lógicos, se requiere pasar los datos por internet. Por ello existe un riesgo de robo de información, pero hay medidas de encriptación para evitarlo. Como por ejemplo, el uso del protocolo HTTPS.
- Velocidad: A tener en cuenta si se dispone de una línea de internet con una tasa de transferencia baja, dado que nos genera un retraso en las solicitudes que hagamos al Escritorio Web.

Dado que este proyecto se basa en un entorno donde el problema de la velocidad de conexión no es realmente inquietante, este inconveniente no nos afecta.

Lo que realmente podría llevarnos algún problema sería el aspecto de querer una aplicación con una funcionalidad elevada, donde elevada significa un volumen considerable de datos en el menor tiempo posible. Pero existen soluciones para estos sistemas, como pueden ser el uso de sistema de cómputo en paralelo, mejorando considerablemente el tiempo de respuesta. Por lo tanto no nos genera problemas asociados.

En resumen, se podría decir que nos aporta una serie de ventajas que realmente hace que sea una buena opción. Partiendo de un sistema base que al conectarse dispondría de unas herramientas, y que con la unión de las tecnologías Cloud Computing y Cómputo Paralelo, podrían ser herramientas realmente potentes y con unas capacidades realmente elevadas.

3.4. Web Desktop Existentes

En la actualidad existen varios sistemas “Web Desktop”, que se detallan en el [anexo II](#) y a continuación los listamos:

Tribalos, Iveneo, Widows4All, Browser OS, Desktop On Demand, Desktop Two, EyeOs, G.ho.st, ISDesktop, NetDesktop, Netvibes, Purefect, Virtual-OS, YouOS, Cloudo, iCloud.

Dichos sistemas se pueden separar en dos grupos: los de acceso gratuito y los de acceso privativo. Donde la mayoría de ellos se engloban dentro de los privativos o con previo registro, y con periodos de prueba. También podemos englobarlos entre los “opensource” y exclusivos, donde podremos ver que escasean los que además de ser acceso gratuito son “opensource”. Pero en la actualidad no hay ninguno que de un soporte para el sistema NFS, cosa que es uno de los requisitos que tenemos en cuenta.

3.5. Motivo de la nueva implementación

La mayoría de los sistemas Web Desktop usan políticas de gestor de ficheros de forma local. Dado que nuestro sistema tendrá una gestión de ficheros basado en el protocolo NFS (Network File System), no podemos implementarlos como realmente deseamos.

Si descartamos los sistemas que no son “opensource”, y luego quitamos los que no permiten la descarga del sistema para su implantación, nos quedamos con un único sistema disponible. El sistema EyeOs, el cual no contiene el funcionamiento adecuado de la gestión de los ficheros tal y como la deseamos. La modificación del núcleo podría suponer, la inversión de mucho tiempo dada la dependencia de ciertos programas web que contiene dicho Web Desktop.

Por ello se optó por la creación de un nuevo Web Desktop, que nos aportara el plus de los sistemas NFS y que la inclusión de las aplicaciones fuera de lo más intuitivo posibles. En definitiva, deseamos un sistema muy modular.

3.6. El sistema UAB_OS

He bautizado el aplicativo generado como “UAB_OS”, dado que se ha pensado para los investigadores de la Universidad Autónoma de Barcelona.

Este Web Desktop, tendrá las siguientes cualidades:

- Sistema de ficheros: El sistema de ficheros se gestiona de forma completamente transparente para los usuarios. Donde a demás de tener su propio espacio de almacenaje, dispondrán de todos aquellos grupos donde estén dados de alta.

Este sistema de ficheros, de forma interna, se gestionará mediante el protocolo NFS. Aportando todas las propiedades dicho protocolo. Este tipo de gestión nos facilitará la creación de nuevos grupos en servidores externos o maquinas virtuales, donde estas últimas no requieren de disponer de una interfaz gráfica.

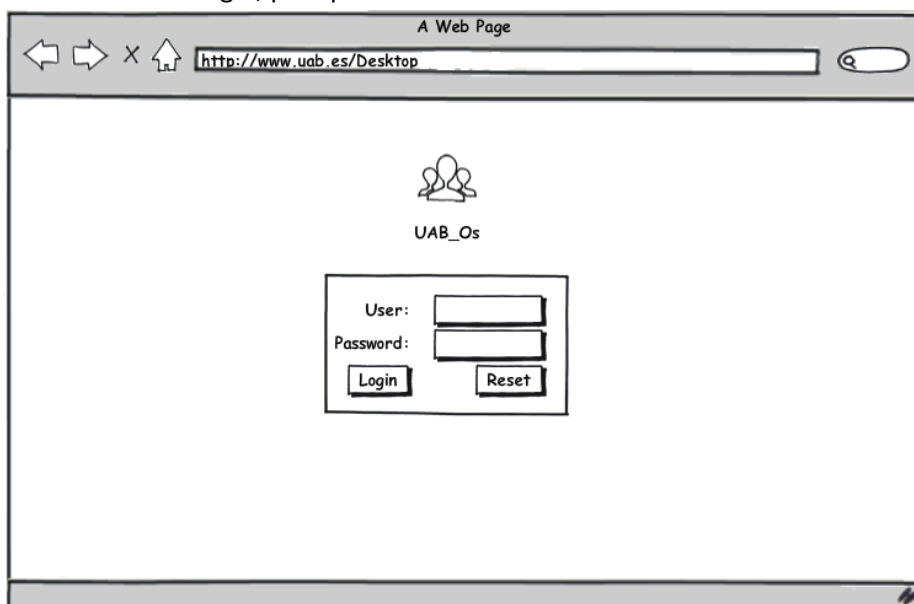
- Sistema Gráfico: El entorno tendrá la característica que los navegadores web donde se esté ejecutando el Web Desktop, tendrá toda la parte de gestión de la parte visual ejecutándose en el cliente remoto.
- Aplicaciones externas: Se dispondrá de un pequeño fichero que se deberá introducir en la cabecera de las aplicaciones web desarrolladas, para permitir la comunicación entre la aplicación y el núcleo propio del sistema.

3.7. Diseño de las diferentes interfaces

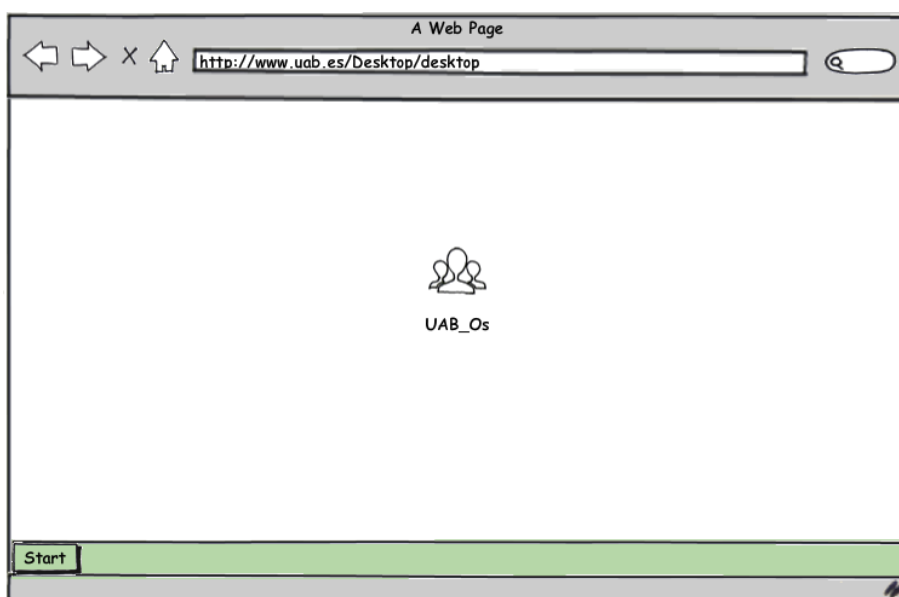
Para la parte visual, se ha tenido en cuenta que la mayoría de los usuarios desean una interfaz parecida a las ya conocidas, por ello se ha intentado mantener un aspecto parecido a los sistemas actualmente conocidos.

Se han definido las siguientes interfaces, que podrían darse en el nuevo Desktop Web que se ha desarrollado, sin tener en cuenta las aplicaciones web desarrolladas.

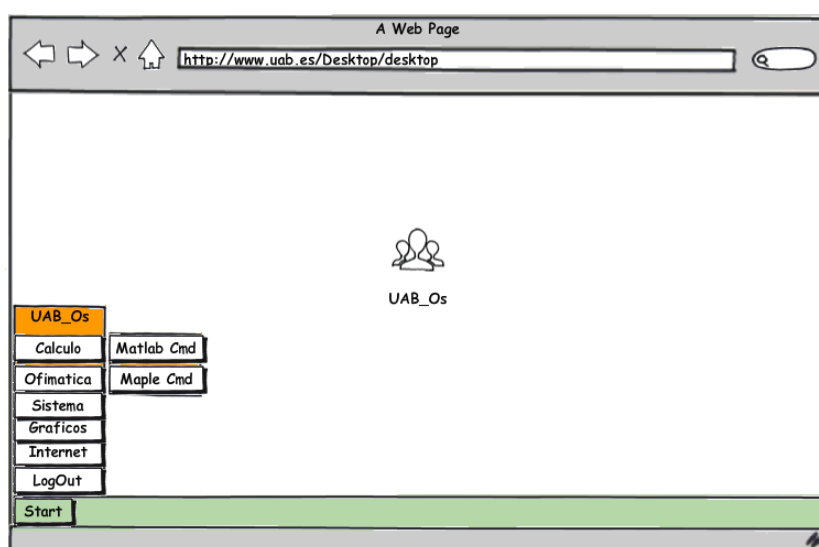
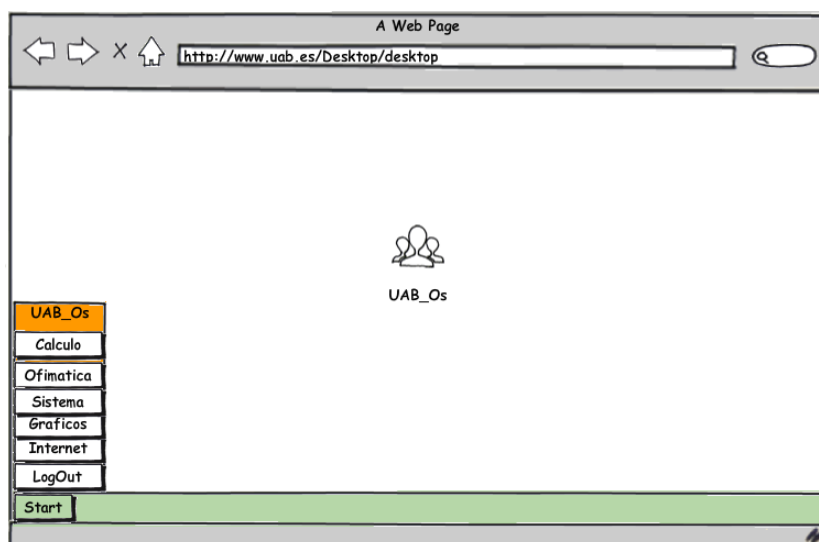
- Una interfaz de Login, para poder entrar en el sistema.



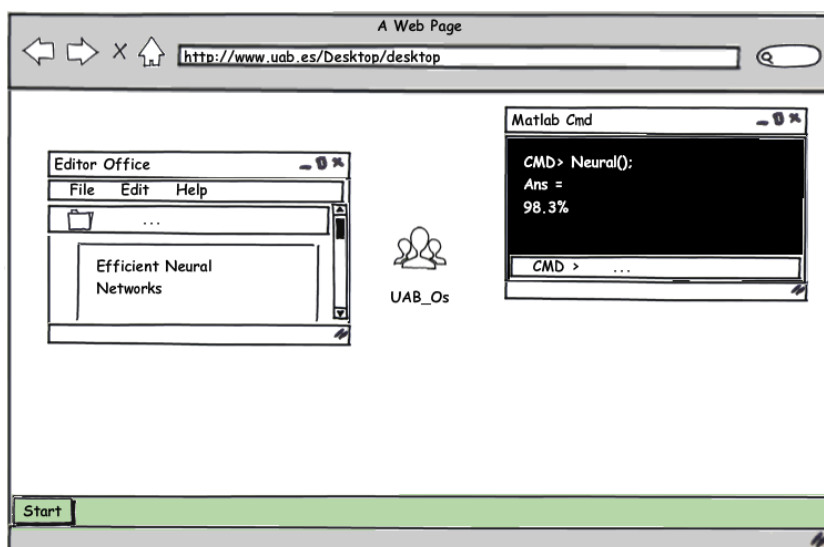
- Una vez se haya entrado en el sistema, requeriremos de un entorno donde trabajar.



- Funcionalidades accesibles desde el entorno, con sus sub-categorías.



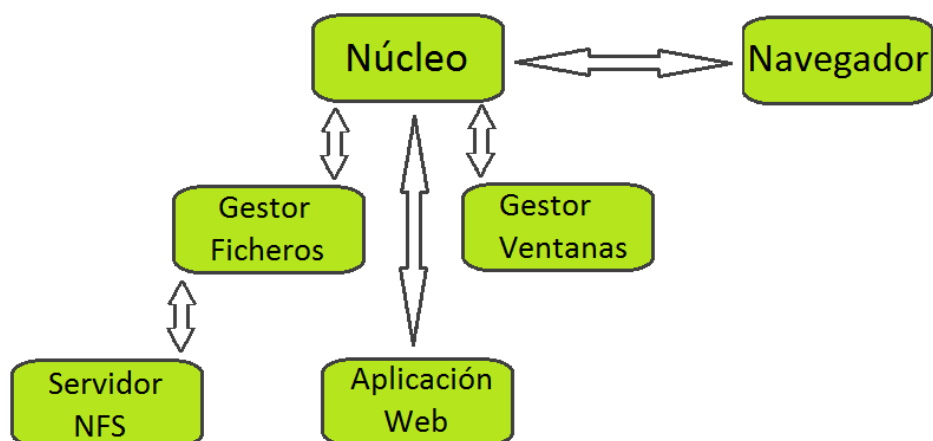
- Se comportarà com un sistema operatiu real, donde tindrem les aplicacions executades en varies "ventanas".



4. Implementación

4.1. Introducción

El sistema sigue la siguiente estructura, donde lo más importante es el núcleo que lo gestiona.



Para que todo funcione de forma coordinada, se ha utilizado la librería JQuery de Javascript, que han facilitado el desarrollo del aplicativo, sobre todo para la parte visual.

4.2. Módulo del núcleo

Este módulo, es el encargado de levantar los demás módulos para hacer que el sistema funcione.

Para ello, este módulo incluye, un conjunto de funcionalidades pre-establecidas:

- Add_Peticion (acción,efecto) : esta función sirve para añadir en una cola interna las acciones a realizar por el núcleo.
- Value_Peticion(): Función que trata las acciones solicitadas, y que se encuentran en la cola interna de solicitudes.
- Init(): Función de inicialización de los demás módulos. Esta es llamada en la carga inicial del escritorio web.

4.3. Módulos de gestión

4.3.1. Módulo Gestor de Ventanas

Este módulo incluye todas las funcionalidades para controlar la gestión de las “ventanas” que se solicitan o las ya existentes.

Incluye:

- Solicitud de página: Cuando se requiere de otra aplicación o herramienta exterior.
- Vincular un nuevo contenido: Nos conecta la página solicitada con el entorno donde ha de mostrarse.
- Mover ventanas: Para mover ventanas.
- Generación de una nueva ventana: Crea una nueva “ventana” donde incluir contenido.
- Destruir ventana: Elimina la ventana seleccionada.
- Swap Windows: Intercambia la ventana seleccionada, según el orden de visualización o profundidad.

4.3.2. Módulo Gestor de Ficheros

Este módulo se encarga de gestión de los ficheros accesibles por el usuario. Esto facilita la comunicación de las solicitudes de las aplicaciones respecto a la lectura o escritura de los ficheros. Por ello tenemos las siguientes funciones definidas.

- DownloadFile: Genera la descarga del archivo escogido, incluyendo la versión deseada.
- InitNFS: Inicialización del servicio NFS cliente, donde recoge el sistema de ficheros del usuario en cuestión.
- FinderContent: Buscar el contenido del Folder escogido por el usuario o aplicativo.
- SaveFile: Guarda el contenido del fichero modificado como una nueva versión.
- OpenFile: Extrae el contenido del fichero seleccionado para insertarlo en algún contenedor especificado.
- NewFolder: Crea una nueva carpeta dentro de la carpeta seleccionada.
- DeleteFolder: Elimina una carpeta de un directorio determinado.

4.4. PHP, SQL, JavaScript y AJAX

4.4.1. PHP

El PHP posee muchas características muy buenas, pero las siguientes características son las más importantes para nuestra aplicación:

- Lenguaje multiplataforma.
- Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en un base de datos. También permite la aplicación de programación orientada a objetos.
- El código PHP es un lenguaje interpretado por el servidor que nos genera como resultado un HTML que es el enviado al cliente. Cosa que nos aporta que la programación sea segura y confiable.
- Capacidad de conexión a bases de datos que se usan en la actualidad, como pueden ser MySQL o PostgreSQL.
- Es “opensource” por lo que es una buena alternativa para tener en cuenta.
- Ampliación del potencial mediante módulos o extensiones.

Ahora veremos que partes, las más relevantes, han sido desarrolladas con esta tecnología.

4.4.1.1. Acceso a Base de Datos

Para asegurarnos de la correcta funcionalidad de la web junto con la base de datos, se ha utilizado la tecnología PHP. Este fichero, que nos permite la conexión a base de datos, se ha definido como una clase. Al instanciarla, se inicializa una conexión a la base de datos y se pueden generar las consultas que queramos, pero al finalizar las consultas, automáticamente se cierra la conexión.

Esta es una forma de gestionarla para no tener varias conexiones abiertas en la base de datos, y reducir posibles fallos de conexiones posteriores.

4.4.1.2. Acceso a los ficheros NFS

Para poder definir seguridad en el sistema de ficheros, se ha escogido hacer una clase que controle el acceso al sistema de ficheros y lo haga según la sesión que se haya generado. En esta clase se filtran los directorios privados de los comunes o de grupo. Para ello, están definidas las mismas funciones de dos formas distintas, de tal forma asegurar la privacidad de lo exclusivo.

4.4.2. SQL

Esta tecnología es la que nos va a servir para llevar a cabo la validación de los usuarios y sus correspondientes grupos. A continuación detallamos las tablas de la base de datos.

4.4.2.1. Tabla USERS

En esta tabla, encontraremos las siguientes columnas y con las siguientes características:

- ◆ NIU: este es el número identificativo de cada usuario. Esta es una llave primaria.
- ◆ PASSWORD: esta es, tal como dice el nombre, la contraseña usada por el usuario.

4.4.2.2. Tabla GROUPS

En la tabla de los grupos, encontraremos las siguientes columnas y con las siguientes características:

- ◆ NIU: este es el número identificativo de cada usuario. Esta es una llave externa.
- ◆ NAME: aquí podremos definir el nombre del grupo que vera cada usuario.
- ◆ GROUP: en este campo, determinaremos el identificador único para cada grupo.

4.4.3. JavaScript

Esta tecnología es uno de los puntos importantes de la aplicación, dado que recibe la mayoría de solicitudes que se generan.

Con esta tecnología, se han desarrollado las partes del núcleo y de los gestores, excluyendo dos partes concretas que se encuentran en el lado del servidor. El motivo de hacerlo con esta tecnología es la de optimizar los recursos del servidor sin requerir consultar siempre a él. Esto implica que el navegador del cliente, junto con esta tecnología, se ha de gestionar aquellas características que no implican nada al servidor. Como pueden ser el movimiento de las ventanas o el cierre de una ventana.

4.4.4. Ajax

El motivo por el que se usa la tecnología Ajax, es para poder refrescar partes del entorno del escritorio, sin necesidad de recargar todo el escritorio, que podría ser más engorroso trabajar con él.

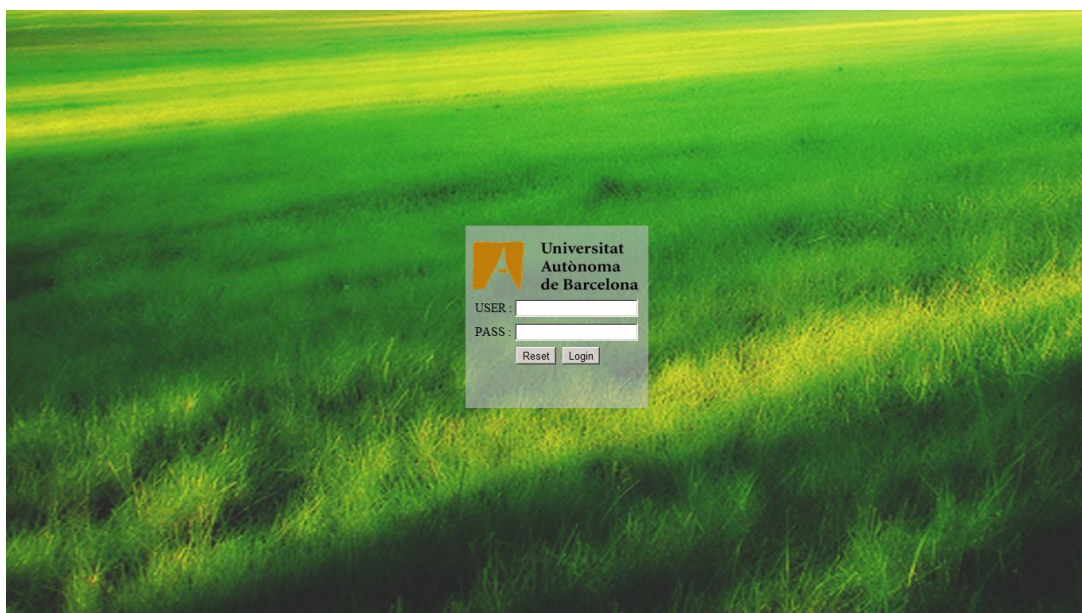
Esta tecnología se aplica en los momentos en los que se generan peticiones de nuevas ventanas o de cambio del contenido de una ventana en concreto.

4.5. Aspectos Visuales

Se ha tenido en cuenta el plan de diseño para hacer que el aspecto visual sea de lo más parecido a lo que se dijo principalmente.

- Página de Login

Podemos ver como se contempla los dos campos requeridos para hacer la validación.



- Entono del Escritorio Web

Dentro del entorno, podemos ver como existe la “carpeta” Dekstop, donde contiene nuestros documentos personales. También, podemos visualizar las carpetas de grupos a los que está vinculado el usuario.



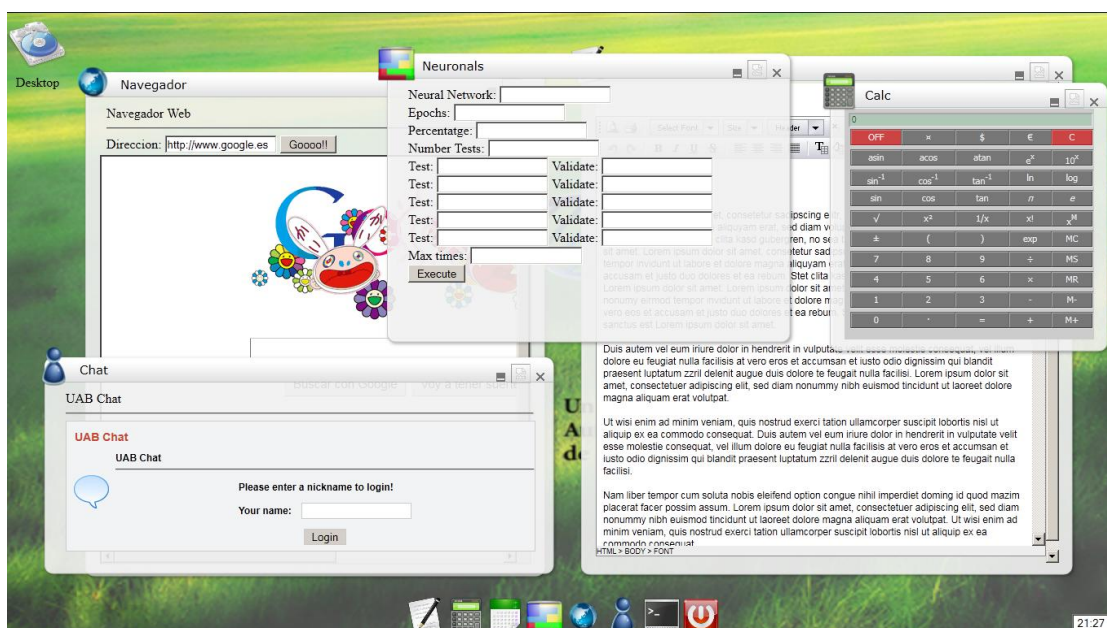
- Menú de Aplicaciones

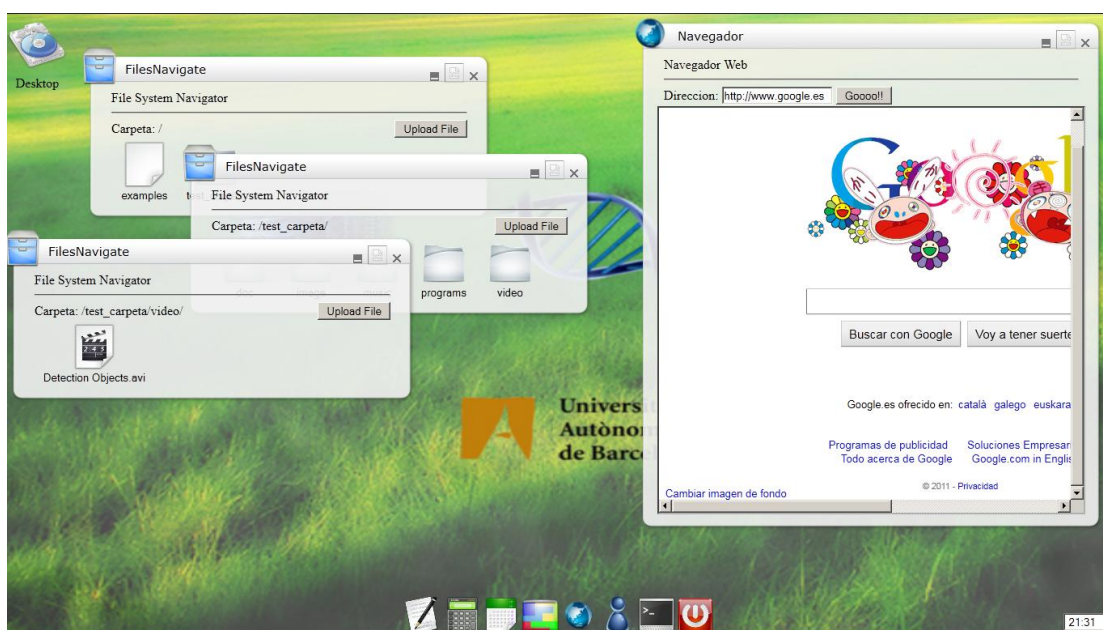
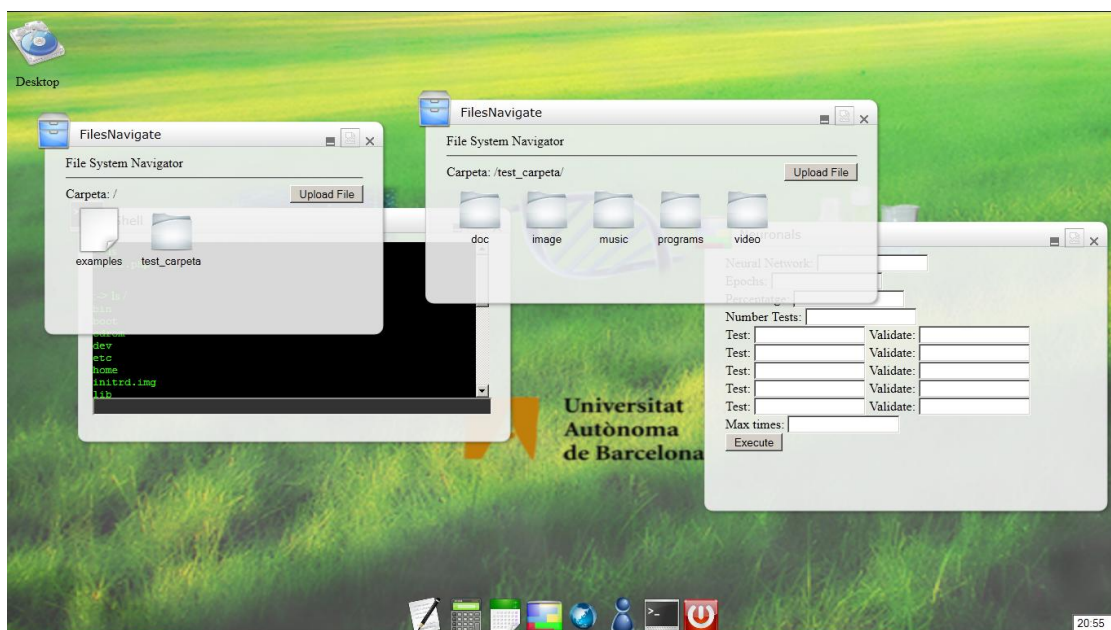
En el entorno podemos ver como se ha generado un menú que nos da acceso a las diferentes aplicaciones disponibles. Tal como se puede visualizar, tiene un cierto parecido con el sistema dock de “Mac”.



- Aplicaciones ejecutándose

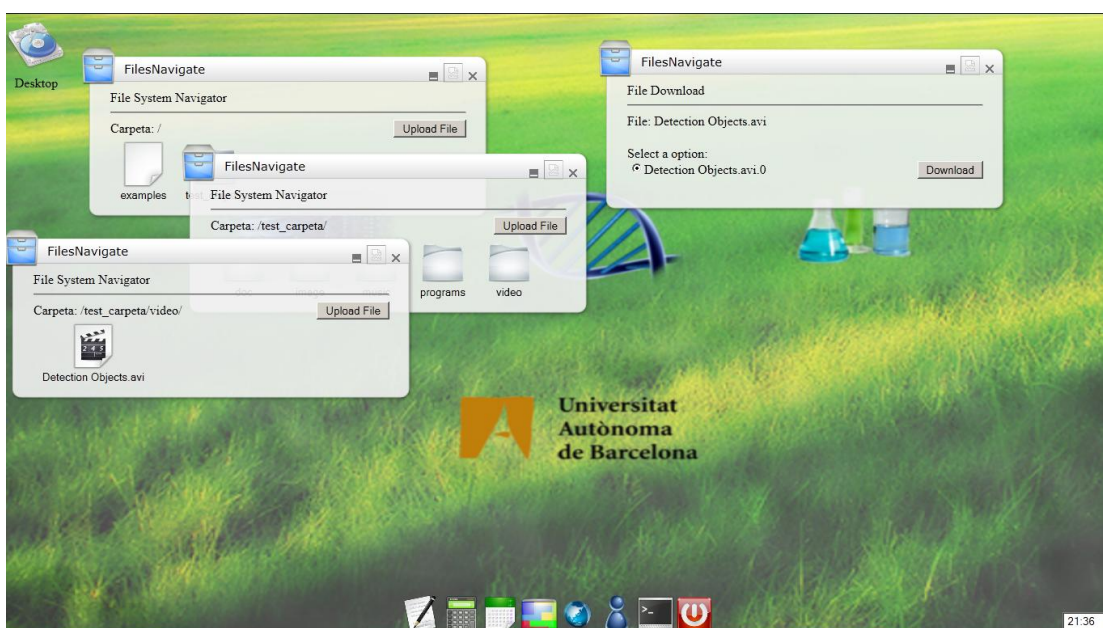
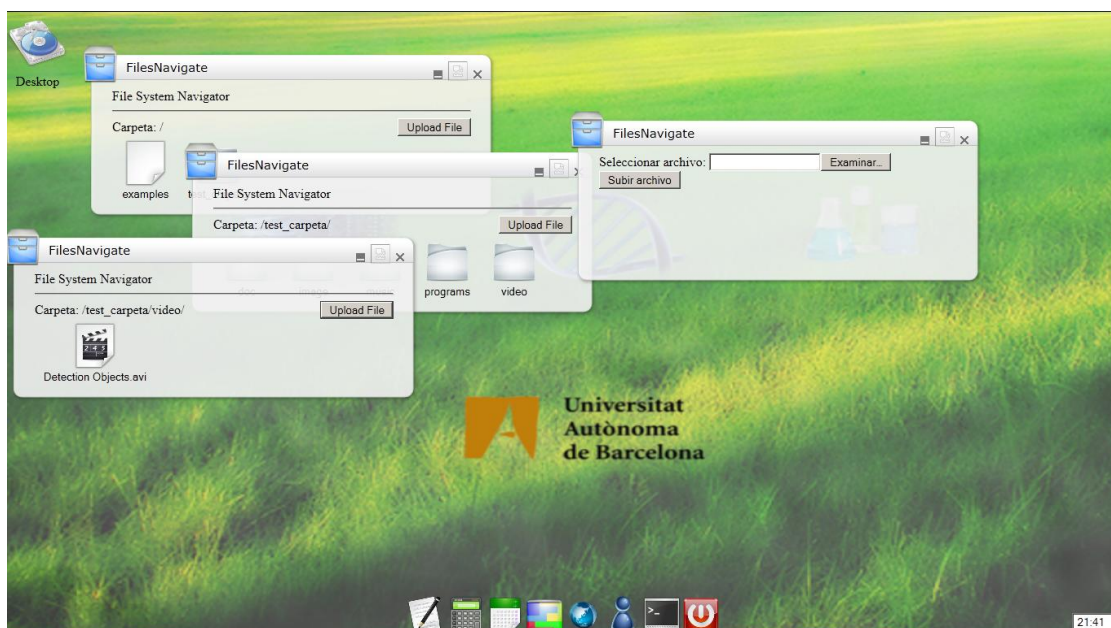
En este caso, visualizamos a diferentes aplicaciones ejecutándose, y comportándose el sistema como un entorno equivalente a un sistema operativo.





- Acciones como descargar una versión, o subir una nueva versión





Dentro del entorno, visualizamos el funcionamiento de la gestión de versiones









4.6. Gestión de los ficheros

Para llevar un control de cómo está gestionado el sistema de ficheros en el servidor y su ha implementado de la siguiente forma.





Raíz

 Desktop	06/06/2011 2:02	Carpeta de archivos
 img	23/05/2011 9:10	Carpeta de archivos
 JS	23/05/2011 9:10	Carpeta de archivos
 default	23/05/2011 9:10	PHP Script

- Desktop: En este directorio encontraremos todo lo relacionado con el entorno del Web Desktop.

 img	06/06/2011 9:53	Carpeta de archivos
 Jscript	06/06/2011 19:09	Carpeta de archivos
 Programs	05/06/2011 18:15	Carpeta de archivos
 Server	06/06/2011 20:10	Carpeta de archivos
 style	06/06/2011 0:46	Carpeta de archivos
 desktop	06/06/2011 2:02	PHP Script

- Img: En esta carpeta encontraremos las imágenes requeridas por el entorno de Login, y de esta forma poder entrar autenticado en el sistema.

 fondo	07/05/2011 19:04	Imagen PNG
 fondo.png_old	03/04/2011 23:56	Archivo PNG_OLD
 icon	03/04/2011 23:56	Imagen PNG
 UAB	12/05/2011 0:10	Imagen PNG







- JS: en esta carpeta contempla los ficheros Javascript requeridos por la página de Login.

 Common	23/05/2011 9:10	Archivo de secuenci...
--	-----------------	------------------------















- Default.php: esta es la página principal, donde nos dará la opción de logearnos y acceder al entorno Web Desktop.

4.6.1. Desktop











En este directorio encontraremos todo lo relacionado con el entorno del Web Desktop.

 img	06/06/2011 9:53	Carpeta de archivos
 Jscript	06/06/2011 19:09	Carpeta de archivos
 Programs	05/06/2011 18:15	Carpeta de archivos
 Server	06/06/2011 20:10	Carpeta de archivos
 style	06/06/2011 0:46	Carpeta de archivos
 desktop	06/06/2011 2:02	PHP Script





- Img: En esta carpeta encontraremos las imágenes requeridas por el entorno de Desktop.

 Files	23/05/2011 9:10	Carpeta de archivos
 Group_Files	03/04/2011 23:56	Imagen PNG
 User_Files	03/04/2011 23:56	Imagen PNG
 Windows	23/05/2011 9:10	Carpeta de archivos
 alert	05/06/2011 19:26	Carpeta de archivos
 bgnds	05/06/2011 19:26	Carpeta de archivos
 black	05/06/2011 19:26	Carpeta de archivos
 default	05/06/2011 19:26	Carpeta de archivos
 icons	05/06/2011 19:26	Carpeta de archivos
 stiky	05/06/2011 19:26	Carpeta de archivos
 white	05/06/2011 19:26	Carpeta de archivos
 blank	05/06/2011 16:25	Imagen PNG
 proxy	05/06/2011 16:25	Imagen PNG
 fondo2	07/05/2011 19:04	Imagen PNG
 fondo2_old	07/05/2011 19:04	Imagen PNG

- Jscript: en esta carpeta contemplamos los ficheros Javascript requeridos por la página del escritorio, donde está en parte, la mayoría de la funcionalidad del núcleo.

 GestorFicheros	05/06/2011 18:56	Archivo de secuenci...
 GestorVentanas	06/06/2011 17:29	Archivo de secuenci...
 jquery.cookie	05/06/2011 16:25	Archivo de secuenci...
 jquery.jqDock.min	31/05/2011 2:48	Archivo de secuenci...
 jquery.metadata	05/06/2011 16:25	Archivo de secuenci...
 jquery-1.6.1.min	28/05/2011 21:41	Archivo de secuenci...
 jquery-ui-1.8.13.min	28/05/2011 21:48	Archivo de secuenci...
 Kernel	05/06/2011 18:28	Archivo de secuenci...
 mbContainer	06/06/2011 19:09	Archivo de secuenci...
 Start	23/05/2011 9:10	Archivo de secuenci...



- Programs: en esta carpeta encontraremos todas aquellas aplicaciones a nivel local de que dispone el entorno del Web Desktop.

 img	03/06/2011 1:08	Carpeta de archivos
 Jscript	03/04/2011 23:56	Carpeta de archivos
 Programs	30/05/2011 3:11	Carpeta de archivos
 List	05/06/2011 18:15	Opera Web Document

- Server: Esta carpeta contiene, únicamente, las partes del núcleo que no pueden ser interpretadas en el entorno del cliente.




 BaseDatos	06/06/2011 20:10	PHP Script
 FileSystem	05/06/2011 18:14	PHP Script

- Style: En este contenedor tendremos las hojas de estilo del proyecto Web Desktop.









 Desktop	05/06/2011 18:55	Documento de hoja ...
 mbContainer	06/06/2011 0:46	Documento de hoja ...

4.6.2. Programs









Aquí es donde residen todas las aplicaciones web que existen dentro del entorno del Escritorio Web.

 img	22/06/2011 18:06	Carpeta de archivos
 Programs	30/05/2011 3:11	Carpeta de archivos
 List	05/06/2011 18:15	Opera Web Document

- **Img:** Imágenes del menú del sistema, deben ser las imágenes que representen al programa.

 Calc_256	21/10/2010 13:12	Imagen PNG
 Calendar_256	06/02/2010 17:54	Imagen PNG
 Chat_256	06/02/2010 17:54	Imagen PNG
 LogOut_256	06/02/2010 17:54	Imagen PNG
 Neuronals_256	02/06/2011 19:47	Imagen PNG
 Shell_256	06/02/2010 17:54	Imagen PNG
 Web_256	02/06/2011 19:46	Imagen PNG
 Writer_256	06/02/2010 17:54	Imagen PNG

- **Programs:** Aquí se centran, todas las aplicaciones disponibles. No por encontrarse en esta carpeta será accesible desde el entorno. Primero se deberá dar de alta en el fichero "List.htm". En caso de disponer de otras aplicaciones externas, solo haría falta crear la referencia en la lista. No hace falta que se encuentre en el directorio.

 Calc	30/05/2011 1:39	Carpeta de archivos
 Calendar	30/05/2011 1:20	Carpeta de archivos
 Chat	23/05/2011 9:10	Carpeta de archivos
 FilesNavigate	30/05/2011 10:57	Carpeta de archivos
 Neurals	30/05/2011 3:03	Carpeta de archivos
 Shell	23/05/2011 9:10	Carpeta de archivos
 WebNavigate	06/06/2011 17:24	Carpeta de archivos
 Writer	23/05/2011 9:10	Carpeta de archivos

4.7. Entorno del Servidor

Para que todo funcione de forma correcta se han tenido que realizar una serie de cambios en el servidor. A continuación, están detallados:

- PHP User

Para poder ejecutar y hacer que funcione el sistema de ficheros NFS, se ha tenido que dar privilegios en al usuario de PHP. Con ello conseguimos que pueda montar únicamente aquellos NFS que se deseen en el momento que se solicite.

- NFS

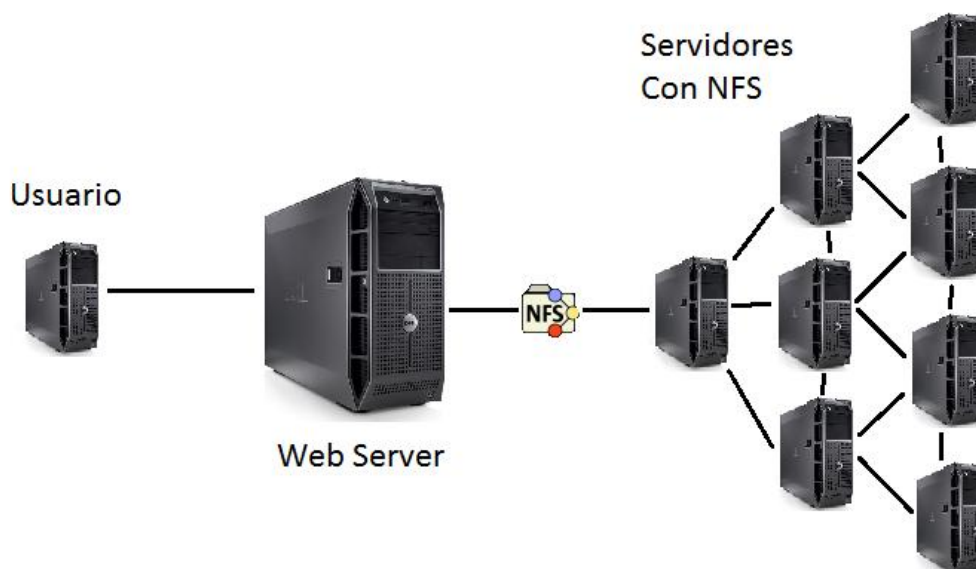
El sistema NFS requiere tener definidos que usuarios son los que tiene accesibilidad a ellos. Para ello, se ha abierto en todos los ordenadores que tengan el sistema NFS una conexión dentro del subdominio. Donde estos tienen la característica que no requieren autenticación para el usuario de PHP.

5. Flujo de datos

Este es el la comunicación básica del entorno, entendiendo las siguientes solicitudes:

- Nuevo Login

Para ello, el servidor recibe la petición y genera una petición en cascada para poder recuperar los ficheros de dicho usuario.

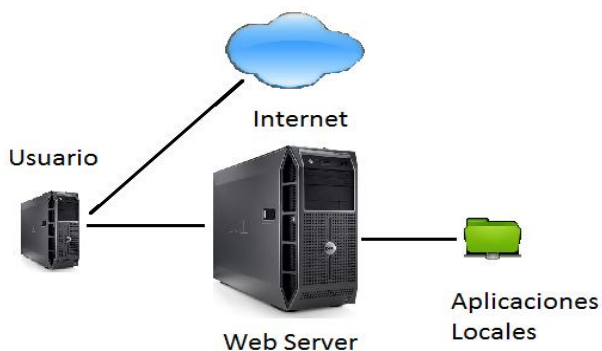


- Solicitud de grupo

En este caso se comporta de la misma forma que en el caso anterior, donde la gran diferencia que existe es un grupo que contiene un usuario. Este usuario nunca hace uso de su cuenta, dado que es un usuario ficticio. En su defecto los que realmente lo usan son usuarios externos a él, donde estos tienen privilegios de lectura y escritura.

- Solicitudes de aplicaciones web internas o externas

En el momento de generar las solicitudes se escoge de forma estricta si ha de pasar por el servidor o no. Para ello, el núcleo del sistema lo gestiona mediante la lista de programas. En esta queda clara si es otra dirección o es la del servidor.



6. Presupuesto

6.1. Valoración de Gastos

A continuación detallo de forma aproximada el coste asociado por el proyecto.

Licencias:

• Base de Datos (MySQL)	0€
• Sistema Operativo Servidor (Ubuntu)	0€
• Tecnologías Web (PHP, SQL, JavaScript, Ajax)	0€
• Servidor (Máquina Virtual VirtualBox)	0€
• Aplicaciones Externas (OpenSource)	0€

Personal:

• (dirección de proyecto) Horas (120) x Precio Hora (50)	6000€
• (diseño gráfico) Horas (300) x Precio Hora (6)	1800€
• (desarrollador) Horas (1200) x Precio Hora (10)	12000€

El total suma 19800€ en gastos generador únicamente con una persona para crear toda la aplicación, un director y un diseñador gráfico.

6.2. Planificación temporal

Realmente ha habido una cierta modificación de la planificación, pero cabe destacar que la modificación a sido por un caso no incluido en el planificación del proyecto. En el proyecto, no se controlo la posibilidad de que me contrataran como empleado en una empresa a seis horas. A parte se ha alargado un poco la planificación, dado que ciertos requisitos que se deseaban en el aplicativo incluyeron una mayor investigación de la estipulada.

Esta ampliación del tiempo implicado esta dentro de un margen de riesgo del 25%(únicamente el desarrollador), por la tanto se podría determinar que el proyecto en parte a cumplido con lo establecido en el informe previo.

7. Conclusiones y Ampliaciones

7.1. Conclusiones

Como resumen general, podría decirse que este proyecto podría llevar a los usuarios a tener una nueva forma de interacción con la Universidad, ya sean investigadores o alumnos.

Este proyecto tiene la ventaja que al trabajar con la tecnología NFS, puede facilitar la creación de nuevas cuentas de usuario de acceso remoto. A parte de que esto podría influenciar en una mejora económica, dado que se pueden montar servidores virtuales con muy pocos requisitos, con el servicio NFS activado, y ser completamente moldeable según las necesidades. Esto implica una mejora energética, e indirectamente una mayor maniobrabilidad en el tratamiento de incidencias.

Estado

Actualmente el proyecto está en una fase alfa. Es operativa pero no contempla realmente todo lo que debería, únicamente se dio importancia en la creación de un sistema completamente modular y usable. Con esto no quiero decir que el proyecto este inacabado, simplemente, que está en fase de mejora.

Este producto se pensó con la finalidad de que en menos de tres años se pueda implementar de forma real en el entorno de la universidad, aportando las mejoras que conlleva.

Teniendo en cuenta que la idea inicial era la de la creación de un entorno con todas las ventajas que nos aportan ciertas tecnologías, podemos decir que está preparado para tener más de una aplicación en los entornos del Cloud Computing y del Computing Paralel.

A nivel de valoración personal.

Este proyecto ha supuesto un gran reto para mí, dado que muy poca gente veía factible la creación de un nuevo sistema Web Desktop desde cero, y encima mezclando diferentes tecnologías. He recibido muchas críticas destructivas sobre mi proyecto, como por ejemplo la no viabilidad, el proyecto es demasiado aparatoso o directamente que era incapaz de finalizarlo en el plazo estipulado. Todas estas críticas, en verdad, las agradezco porque han sido parte de mi afán de superación y de demostrar que no hay nada que no se puede realizar.

He tenido también momentos alegres y no tan alegres. Pasar por en los que encontrar una solución era casi inexistente, como indagar y encontrar, con paciencia y dedicación, caminos que han hecho llevar a buen puerto el proyecto.

En este proyecto he pasado por varias facetas estudiadas durante la carrera, como pueden ser Bases de Datos, Ingeniería de Software, Planificación de sistemas, Redes, y un largo etcétera. He intentado mezclar todos los conocimientos adquiridos durante estos 5 años en un único proyecto, para que se refleje todas las etapas por las que he pasado.

Posiblemente este proyecto refleje lo que un estudiante aprende en esta universidad, y puedo asegurar que, a mi parecer, me han facilitado las herramientas necesarias para poder llevar a cabo nuevas aplicaciones de la ingeniería informática, en ámbitos no tan comunes.

También he de agradecer a mi tutor del proyecto, ya que ha sido un apoyo moral durante el transcurso del proyecto, aportándome ideas y facilitando su disposición en cualquier momento.

7.2. Ampliaciones

Este proyecto puede tener varias vías de desarrollo, a continuación expongo las que destaco como las más importantes:

Investigadores

- Insertar aplicaciones que generen una mayor interacción.
- Insertar aplicaciones que contemplen cómputo en paralelo, para una resolución de problemas de cálculo mayor.

Realmente puede expandirse mucho, dado que al tratarse de un sistema completamente modular se puede crear aplicaciones externas completamente integradas con el sistema.

A parte, hay otro sector que podría verse incluido en este aplicativo si lo extendemos a la comunidad universitaria.

Universitarios

- Insertar aplicaciones que generen una mayor interacción entre universitarios.
- Insertar aplicaciones que contemplen un sistema de aprendizaje “e-learning” para las asignaturas.
- Accesos directos a los dominios existentes en la UAB, como `por ejemplo: Caronte, Deic, CVC, Neptu, etc. Incluyendo una autenticación automática al estar en el escritorio web.
- Aplicaciones para desarrollar en entornos SandBox, para que los alumnos tengan una mayor interacción con la realidad de los sistemas.

Realmente puede expandirse mucho, dado que al tratarse de un sistema completamente modular se puede crear aplicaciones externas completamente integradas con el sistema.

8. Bibliografia

8.1. Enlaces Web

[1] PHP: Hypertext Preprocessor

www.php.net/

[2] JQuery: The WriteLess, Do More, JavaScript Library

www.jquery.com/

[3] JQueryui: User Interface

www.jqueryui.com/

[4] MySQL: The world's most popular open source database

www.mysql.com/

[5] Ubuntu

www.ubuntu.com/

[6] EyeOs

www.eyeos.org/

8.2. Libros de consulta

- La Biblia de PHP 5 (Anaya)
- JavaScript for Dummies
- Computer Networks (A.Tanenbaum)
- Operating Systems: Design and Implementation (A.Tanenbaum)
- Modern Operating Systems (A.Tanenbaum)
- Distributed Systems: Principles and Paradigms (A.Tanenbaum)

8.3. Software utilizado

A continuación detallo todo el software utilizado para llevar a cabo el Web Desktop.

- Sistema Operativo Ubuntu, versión 10.10.
- VirtualBox, Emulador de máquinas.
- Windows 7 con IIS.
- Paquete PHP 5 para Ubuntu y Windows.
- Paquete Portmap para Ubuntu.
- Paquete NFS para Ubuntu.
- DreamWeaver (edición web).
- MySql 5.5 para Ubuntu y Windows.
- MySqlAdmin para Ubuntu.
- MySql WorkBench para Windows.
- Navegador Safari para Windows.
- Navegador Opera para Windows.
- Navegador Firefox para Windows.
- Navegador Chrome para Ubuntu y Windows.
- NotePad++ para Windows.

9. ANEXOS

9.1. ANEXO I: Requisitos

9.1.1. Requisitos Funcionales

9.1.1.1. Requisitos funcionales de usuario

- Los usuarios han disponer un entorno para sus documentos.
- Los usuarios han de poder subir al entorno sus propios documentos, y los documentos de los grupos donde este dado de alta.
- Los usuarios han de poder modificar el contenido de los ficheros a los que se dispone de privilegios.
- Se ha de identificar en la aplicación mediante el identificador NIU y el password asociado a él.
- Acceso a diferentes aplicaciones, dentro del entorno donde se trabaja. Estas aplicaciones, contendrán cada una de ellas otros requisitos.

9.1.1.2. Requisitos funcionales de servicios

- Los usuarios podrán dar uso a todas las aplicaciones existentes en el sistema.
- Los usuarios podrán solicitar un nuevo ingreso a otro grupo.

9.1.2. Requisitos No Funcionales

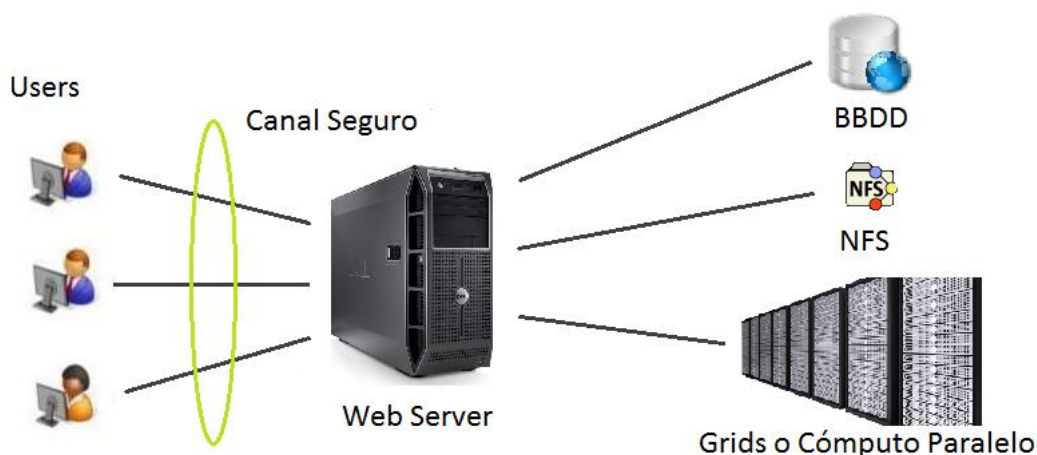
9.1.2.1. Requisitos plataforma

- La aplicación será de tipo web.

9.1.2.2. Requisitos de seguridad

- El sistema debe de contener sistemas de seguridad para asegurar la privacidad de las investigaciones y la documentación privada de cada uno de los usuarios existentes.
- Para ello se requiere un sistema que nos aporte una autenticación única. Para los grupos de investigación, debería existir un sistema de control de versiones. De esta forma nos aseguramos de no perder información por culpa de un error.

Estos requisitos nos llevan a tener canales completamente seguros, tal como se visualizan en el siguiente gráfico.



9.1.2.3. Requisitos de recursos

- Ha de ser compatible, como mínimo, en el navegador Firefox.

9.1.2.4. Requisitos de disponibilidad

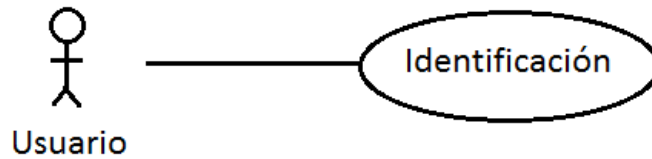
- El entorno debe de ser operativo al 99% del tiempo.

9.1.2.5. Requisitos de rendimiento

- Les pàgines web ha de ser lo más pequeñas posible.
- El tiempo de descarga de las aplicaciones web no han de superar los 15 segundos.
- El sistema deberá ser capaz de soportar como mínimo 100 usuarios de forma concurrente.
- El sistema ha de soportar más de 1000 conexiones diarias y un volumen de transferencia superior a 2 TeraBytes mensuales.

9.1.3. Casos de uso

9.1.3.1. Escenario 1: Identificación del usuario



Actores: Usuario.

Objetivo: Garantizar el acceso único.

Pre-condición: El usuario debe estar dado de alta en el sistema.

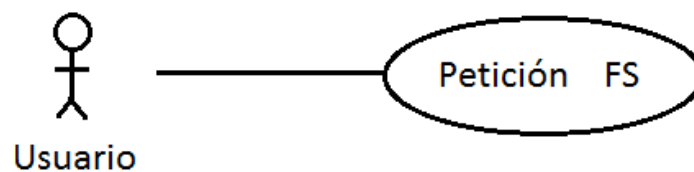
Pos-condición: El usuario es identificado y entra en la aplicación.

Flujo Básico:

- Introducir NIU
- Introducir contraseña
- Darle al botón Login

Errores: Usuario no existente, o si que existe pero no con contraseña errónea.

9.1.3.2. Escenario 2: Uso del sistema de ficheros



Actores: Usuario.

Objetivo: Garantizar el acceso a los documentos.

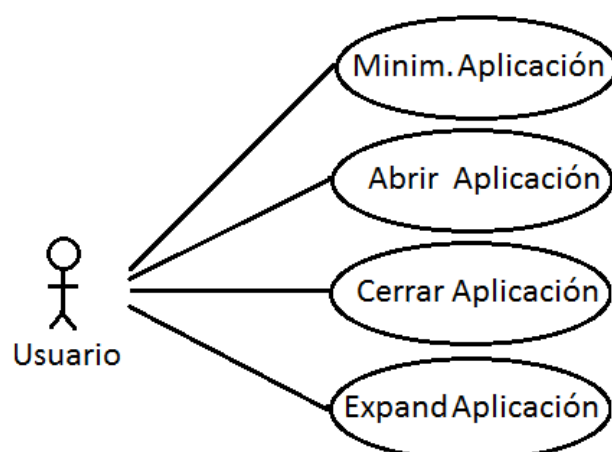
Pre-condición: La petición debe ser una opción válida.

Pos-condición: Se ha realizado la petición.

Flujo Básico:

- Seleccionar opción
- Solicitarla mediante un click.

Errores: Solicitud no válida o temporalmente requerido por otro servicio.

9.1.3.3. Escenario 3: Uso de las aplicaciones

Actores: Usuario.

Objetivo: Ampliar la funcionalidad al usuario.

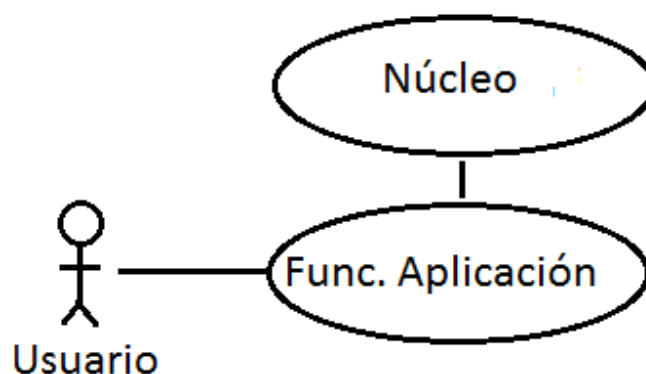
Pre-condición: La aplicación ha de existir en la barra de aplicaciones.

Pos-condición: La aplicación se ha abierto.

Flujo Básico:

- Seleccionar aplicación.
- Solicitarla mediante un click.

Errores: Aplicación no válida o no accesible de forma directa.

9.1.3.4. Escenario 4: Uso de la interacción de las aplicaciones con el núcleo

Actores: Usuario.

Objetivo: Ampliar la interacción del usuario con aplicaciones y núcleo.

Pre-condición: La aplicación ha de existir estar abierta y el comando existente.

Pos-condición: Ejecución de la función.

Flujo Básico:

- Seleccionar aplicación.
- Seleccionar función.
- Solicitarla mediante un click.

Errores: Aplicación no válida, función no existente o no accesible de forma directa.

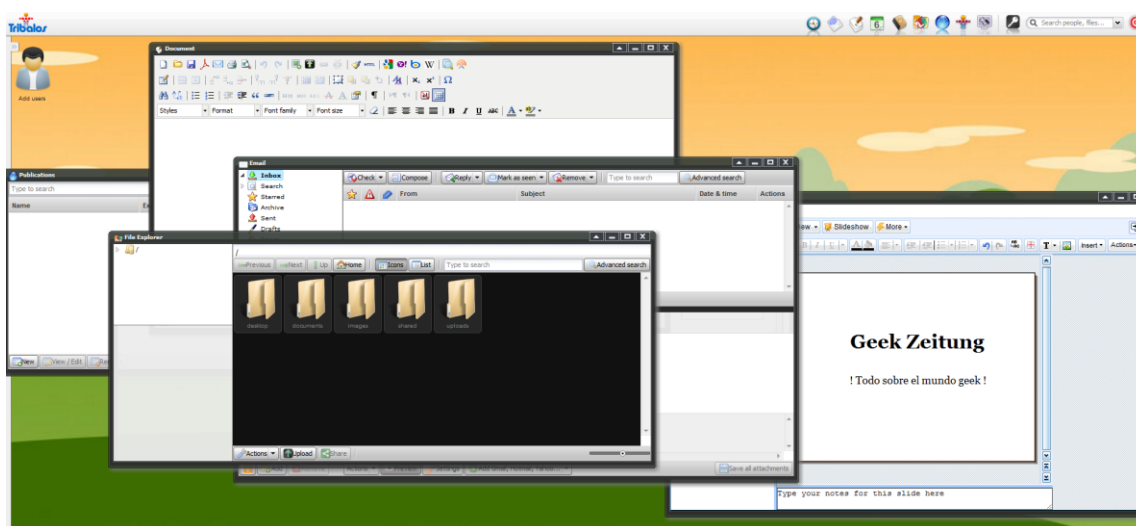
9.2. ANEXO II: Sistemas Desktop Web Actuales

En esta tabla se pueden visualizar los Desktops Web más relevantes actualmente.

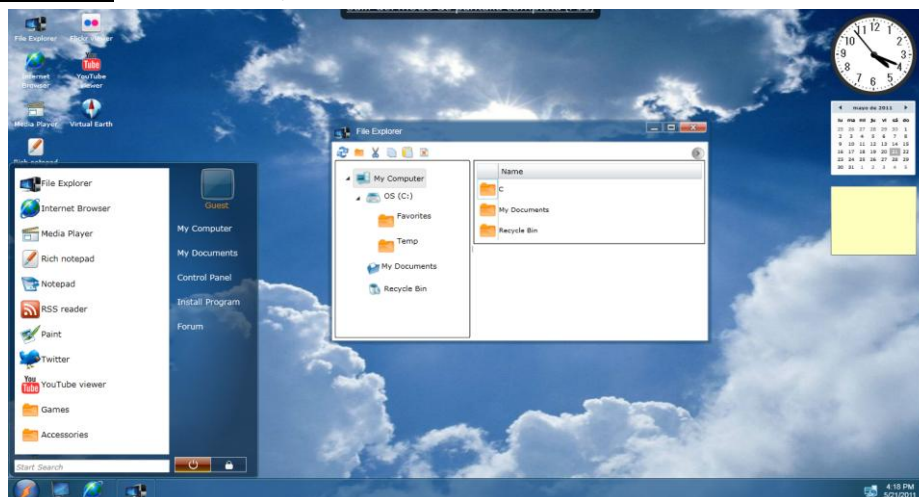
Nombre	Motor	Gratis	Licencia	Aplicaciones Externas	Descargable
Tribalos	Jquery + PHP	No	Propietario	Si	No
Ivaneo	Ajax	Si	Propietario	Si	No
Widows4All	Silverlight	Si	Propietario	Si	Si
Browser OS	JavaScript + Ajax	Si	OpenSource (Api Propietario)	Si	No
Desktop On Demand	Java/NX	Si	Propietario	Si	No
DesktopTwo	Flash	Si	OpenSource	Si	Si
EyeOs	AJAX	Si	OpenSource	Si	Si
G.ho.st	Flash	Si	Propietario	Si	No
ISDesktop	GWT/GWM/Java/J2EE	Si	Propietario	Si	No
netDesktop	JavaScript	Si	Propietario	Si	No
Netvibes	AJAX	Si	Propietario	Si	No
Purefect	PHP + AJAX	Si	OpenSource (GPL)	Si	Si
Virtual-OS	PHP + AJAX	Si	OpenSource (MIT)	Si	No
YouOS	AJAX	Si	Propietario	Si	No
Cloudo	AJAX	Si	Propietario	Si	No
iCloud	XML	Si	Propietario	Si	No

A continuación mostraremos los diferentes Desktop Web existentes, dado que hay ciertos que han dejado de dar soporte.

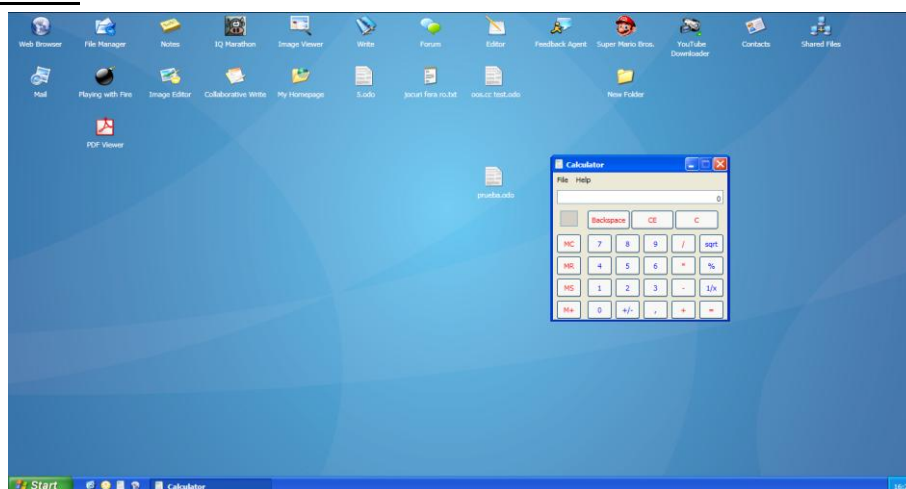
Tribalos:



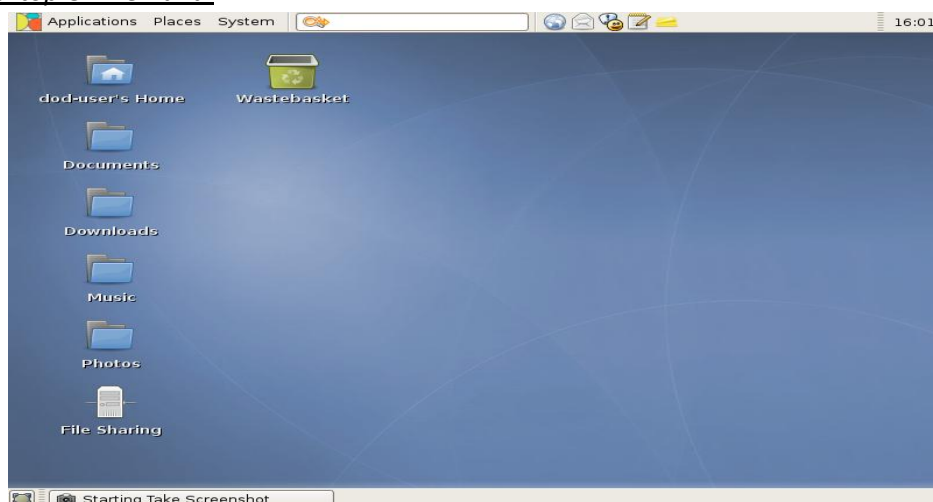
Windows4all: Actualmente, el sistema se llama SilveOS.

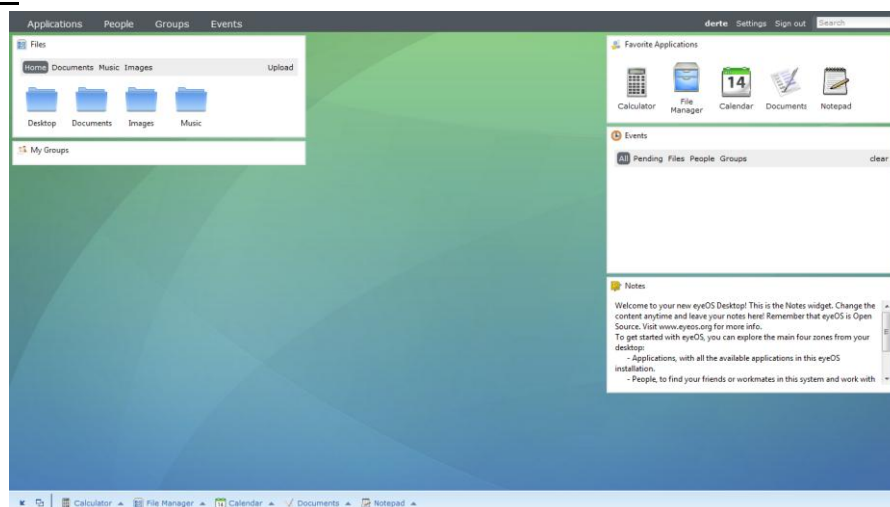
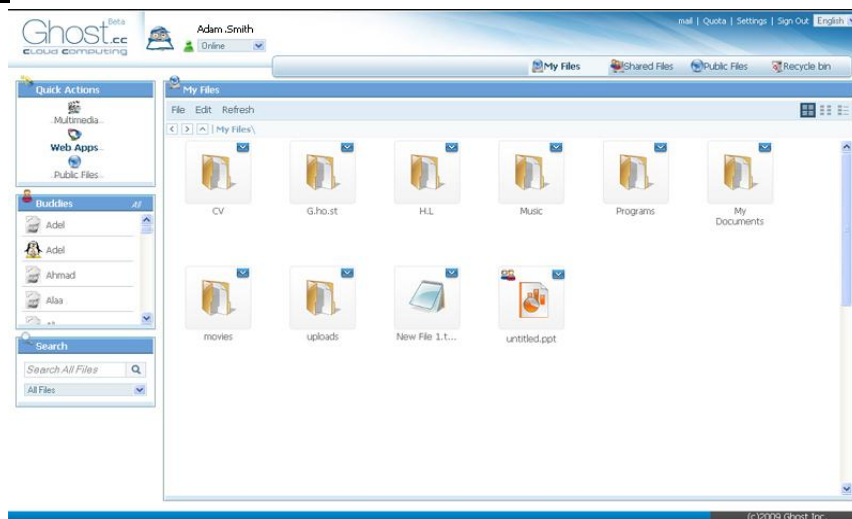


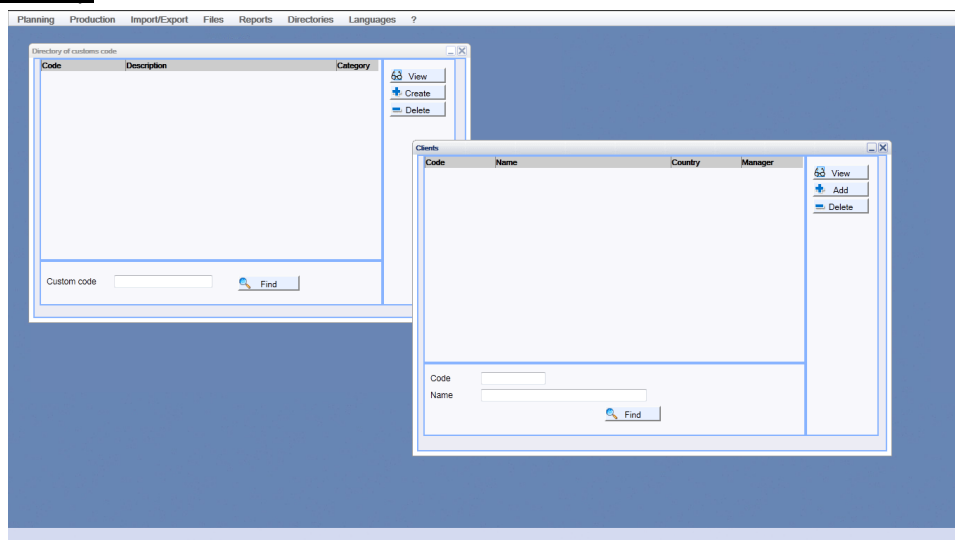
BrowserOs:



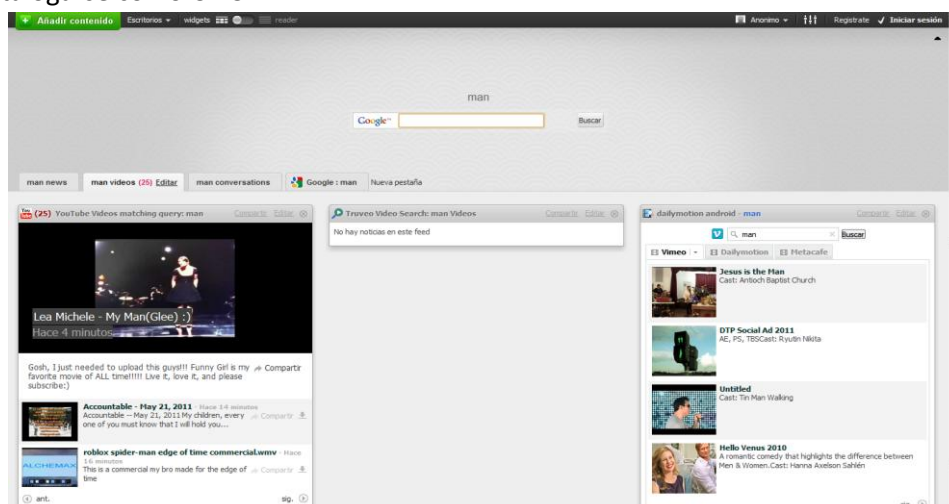
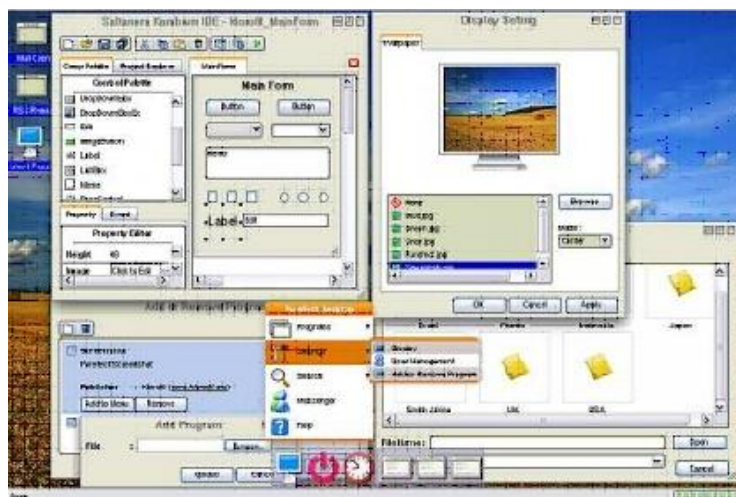
Desktop On Demand:



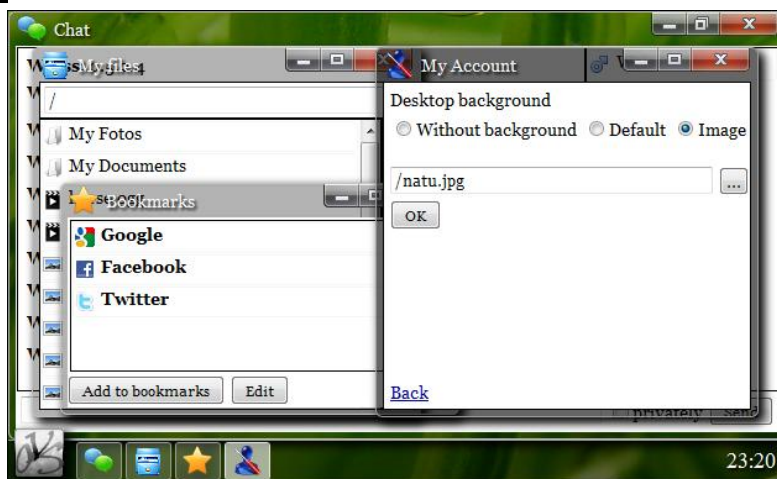
DesktopTwo:EyeOs:G.ho.st:

ISDesktop:

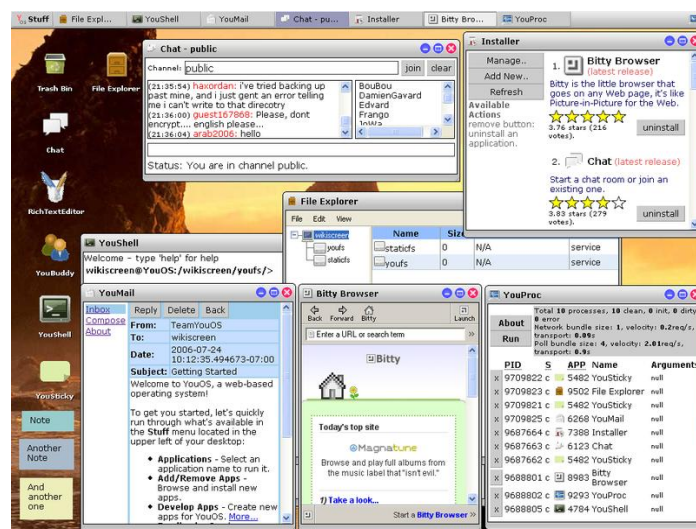
Netvibes: Entra dentro del grupo de Desktop Web por la funcionalidad, pero debería catalogarse como CMS.

Purefect:

Virtual-OS:



YouOS:



Cloudo:



iCloud:



Abstracto

Este proyecto consiste en la elaboración de una propuesta del portal web para permitir la investigación y compartir investigaciones entre los investigadores de la Universidad Autónoma de Barcelona.

Esta memoria explica las necesidades del proyecto, el diseño de la solución, la implementación de dichas solución y los resultados, para finalmente exponer las conclusiones y ampliaciones futuras.

Para llevar a cabo el proyecto se realizo un estudio de los Web Desktop y la gestión interna de ellos, y finalmente se optó por realizar uno nuevo por las grandes diferencias entre las necesidades y lo existente. El producto se llama "UAB OS".

Abstracte

Aquest projecte consisteix en l'elaboració d'una proposta del portal web per permetre la investigació i compartir investigacions entre els investigadors de la Universitat Autònoma de Barcelona.

Aquesta memòria explica les necessitats del projecte, el disseny de la solució, la implementació d'aquestes solució i els resultats, per finalment exposar les conclusions i ampliacions futures.

Per dur a terme el projecte es va realitzar un estudi dels Web Desktop i la seva gestió interna, i finalment es va optar per realitzar un de nou per les grans diferències entre les necessitats i l'existent. El producte es diu "UAB OS".

Abstract

This project is developing a web portal proposal to allow research and share research between researchers from the Autonomous University of Barcelona.

This report explains the needs of the project, solution design, implementation of such solutions and results, to finally present the conclusions and future extensions.

To carry out the project conducted a survey of the Web Desktop and the internal management of them, and finally opted for a new one for the large differences between the needs and existence. The product is called "UAB OS".