



Sistemas Informáticos

Curso 04-05

Servidor de Fusión de Volúmenes

Beatriz Herrezuelo Sánchez
Eva Cristina Sánchez San José
Juan Antonio Martín Pedro

Dirigido por:
Prof. Luis Hernández Yáñez
Departamento de Sistemas Informáticos y Programación

Facultad de Informática
Universidad Complutense de Madrid




1.	LISTA DE PALABRAS CLAVE.....	5
2.	INTRODUCCIÓN.....	6
3.	LAS IMÁGENES MÉDICAS Y SU IMPORTANCIA .	7
	3.1. Imágenes médicas	8
4.	OBJETIVOS	10
	4.1. Objetivo inicial.....	10
	4.2. Fusión 2D.....	10
	4.3. Fusión 3D.....	11
	4.4. Proceso completo	14
	4.5. Objetivos cumplidos	14
5.	PLAN DE TRABAJO	15
6.	SEGUIMIENTO	17
7.	PROBLEMAS SURGIDOS.....	18
8.	REQUISITOS INICIALES	19
9.	DISEÑO DE LA APLICACIÓN	20
	9.1. Tipos de usuarios y su porqué. Funcionalidad de cada usuario.....	20
	9.1.1. Usuario Particular.....	20
	9.1.2. Usuario Entidad / Institución	20
	9.1.3. Usuario Demo	21
	9.1.4. Usuario Administrador.....	21
	9.1.5. Usuario General.....	21
	9.2. Diseño de las clases, organización y jerarquía.....	22
	9.2.1. BD.....	22
	9.2.2. Logica.....	23
	9.2.3. Sesion	23
	9.2.4. Servlets.....	23
	9.3. Seguridad.....	24
	9.4. Autonomía	24
	9.5. Requisitos hardware.....	24
	9.6. Facilidad de uso	25
	9.7. Escalabilidad	25
10.	DESARROLLO E IMPLEMENTACIÓN	26
	10.1.Arquitectura del sistema.....	26
	10.2.Lenguaje utilizado	26
	10.3. ¿Por qué servlets?.....	27
	10.4. ¿Por qué no ampliar las cgi's?	28
	10.5. ¿Por qué MySql?	28
	10.6. ¿Por qué Tomcat?.....	28



10.7.	Seguridad.....	29
10.8.	Auto-mantenimiento.....	30
10.9.	Requisitos hardware.....	30
10.10.	Facilidad de uso.....	30
10.11.	Escalabilidad	31
11.	DIAGRAMA ENTIDAD / RELACIÓN.....	32
12.	PAQUETES DE LA APLICACIÓN	33
13.	DIAGRAMAS DE JERARQUÍA DE CLASES	34
13.1.	Clases pertenecientes al tipo de usuario Administrador .	45
13.2.	Clases pertenecientes al tipo de usuario Entidad/Institucion	46
13.3.	Clases Pertenecientes al tipo de usuario Particular	46
13.4.	Clases pertenecientes al tipo de usuario General y clases comunes.....	47
14.	CASOS DE USO	48
15.	INTERFAZ DE USUARIO	50
15.1.	¿Por qué se diseño así, la interfaz de usuario?	50
15.2.	Apariencia de la interfaz de usuario.....	51
15.2.1.	Usuario administrador.....	51
15.2.2.	Página de visualización de los usuarios demo .	53
15.2.3.	Página de visualización de las entidades dadas de alta	55
15.2.4.	Página de visualización de los usuarios particulares.....	59
15.2.5.	Usuario Entidades/Instituciones	60
15.2.6.	Pantalla de modificación de datos	61
15.2.7.	Pantalla de visualización de usuarios dados de alta	62
15.2.8.	Pantalla de alta de Usuario.....	64
15.2.9.	Baja del Servicio	65
15.2.10.	Usuarios particulares	66
15.2.11.	Pantalla de modificación de datos	68
15.2.12.	Baja del Servicio	68
15.2.13.	Usuarios demo.....	69
16.	TRABAJO ADICIONAL INVESTIGADO.....	70
16.1.	¿Cómo funciona la fusión 3D?.....	70
16.2.	Transformación de los volúmenes a formato AIR	75
16.3.	Ajuste de volúmenes	76
16.4.	Explicación de los módulos que componen el código implementado:	77
16.5.	Formato imágenes DICOM	78



16.5.1. Características de DICOM.....	80
16.6. Servidor Dicom	81
16.7. Software para visualización de imágenes Dicom 	81
16.8. Arquitectura cliente-servidor y sockets.....	82
17. TRABAJO FUTURO	83
18. APÉNDICES	84
18.1. Supercomputador abaco.sim.ucm.es	84
18.2. Tuning del Air 3.08 (aplicado al estudio de alignlinear)	86
19. AGRADECIMIENTOS	91
20. BIBLIOGRAFÍA	92
21. AUTORIZACIÓN.....	94



1. LISTA DE PALABRAS CLAVE

Portal web

Servlet

Telemedicina

Fusión

DICOM

Imágenes/volúmenes

3D

AIR



2. INTRODUCCIÓN

Este proyecto está realizado con el fin de colaborar con el Instituto de Tecnología de la Universidad Complutense de Madrid. La aplicación cuyo desarrollo se ha llevado a cabo durante este año, forma parte de un gran Proyecto de Investigación sobre la fusión de imágenes médicas que comenzó hace varios años y en el que han colaborado médicos, psicólogos, físicos, informáticos y demás personal relacionado con la medicina.

Nuestro trabajo ha consistido en implementar una aplicación más moderna y con más funcionalidades que la que ya existía. De este modo, médicos y entidades relacionadas con este mundo pueden registrarse y hacer uso de dichas funcionalidades para elegir distintas imágenes de RM(Resonancia Magnética), PET(Tomografía por emisión de Positrones) o TAC(Tomografía Computerizada), en formato DICOM, y a través del envío de las mismas al supercomputador (ABACO) que se encuentra en el Centro de Proceso de Datos de la UCM obtener su fusión. Con la imagen resultante se pueden realizar estudios más exhaustivos sobre tumores o enfermedades. Todo este gran proyecto lleva implícito mucho trabajo de investigación de los resultados de la fusión, y también toda una serie de comunicaciones entre distintos ordenadores individuales con el servidor que aloja la aplicación, y de dicho servidor con el supercomputador que realiza la fusión de las imágenes y después las devuelve.

INTRODUCTION

This project has been done to collaborate with the Institute of Technology of the Complutensian University of Madrid (Spain). The application developed during this year, is a part of a big Investigation Project, which started many years ago, about the Fusion of Medical Images, and in which develop have collaborated doctors, psychologists, physicists, computer sciences, and people related to medicine.

Our project has consisted of creating a modern and more functional application than the actual one. In this manner, doctors and Institutions related to medicine can be registered in the application and use its functionalities to select MR (Magnetic Resonances), PET (Positron Emission Tomography) or CAT (Computer Assisted Tomography) images, that will be in DICOM format, send them to the Supercomputer (ABACO), located in Data Processing Centre of UCM, and get the fusion of them. With the resulting image, doctors can make more exhaustive studies about tumours or diseases. All this big project implies a lot of investigation about the fusion results, and about the communication between pc's and server too. Moreover, the project implies investigation about the communication between server and Supercomputer, and how to make the fusion of the images.



3. LAS IMÁGENES MÉDICAS Y SU IMPORTANCIA

En los últimos años se ha producido un gran crecimiento en las tecnologías para la producción de imágenes médicas y el coste del procesamiento digital de alta capacidad se ha reducido considerablemente. Esto ha promocionado el uso de la visualización científica en muchas áreas donde los datos son abundantes, complejos y ricos en calidad. Cada vez es mayor el número de imágenes que se obtienen para caracterizar la anatomía y las funciones del cuerpo humano y por ello es imprescindible la familiarización del médico con los métodos y sistemas que le permitirán analizar y manejar esta gran cantidad de información de una forma rápida y eficiente.

Tradicionalmente, la práctica de la medicina ha requerido el manejo de información de diverso tipo: tacto, sonido, apariencia, olor, etc. de una manera comparable a un arte. Este arte de la medicina es aún más evidente cuando la terapia depende de la coordinación ojo-mano del médico y es el paso final que decide el resultado de un largo y costoso proceso de investigación.

De esta forma, la visualización de la información que describe la naturaleza de la enfermedad es vital en medicina, y aun más en cirugía. A pesar de que la información basada en imágenes ha guiado siempre el proceso terapéutico, en los últimos años se ha producido un crecimiento explosivo en la calidad y cantidad de las mismas, produciéndose dificultades para el procesamiento e integración de ellas. Todo parece indicar que esta tendencia lejos de detenerse, se irá haciendo cada vez más fuerte.

Hace algunos años un Tomógrafo Computerizado(TAC) tardaba casi dos minutos en producir un corte anatómico de un centímetro de grosor. Un estudio de cerebro estaba constituido por aproximadamente 12 cortes. Hoy, una TAC helicoidal multicorte puede producir imágenes a razón de 32 por segundo, y dependiendo de la resolución con que se esté trabajando, pueden producirse 100 a 200 cortes por cada volumen cerebral, con un grosor que puede llegar a 0.5 mm. Si consideramos que el estudio de algunas patologías requiere de varias modalidades de examen para la misma región anatómica, no es raro que algunos conjuntos de datos puedan estar constituidos por varios miles de imágenes.

Hoy en día, es muy común ver en la consulta pacientes a los cuales se les ha practicado TAC, angio TAC, resonancia magnética (RM), angio resonancia, angiografía convencional digital, SPECT, ECO Duplex, etc... Todas estas imágenes suelen pesar varios kilos, en placas difíciles de transportar, y cuya revisión es larga y tediosa. La revisión de los antecedentes puede tardar mucho, y está sujeta de esta manera a frecuentes errores y omisiones. La integración de la información es muy influyente en el resultado de todo este proceso terapéutico, por lo que las herramientas que faciliten esta integración tienen un resultado inmediato en la calidad de la atención. Es por esto que la comprensión y la utilización de las herramientas adecuadas por parte de la comunidad médica, son fundamentales para



aprovechar el desarrollo que ha significado la incorporación de las diversas tecnologías de imágenes.

Así como la visualización de las imágenes es un aspecto del problema, no menos importante es la interacción entre el médico y la información contenida en ellas. Es aquí donde el desarrollo de la tecnología de la información gráfica ha tenido y tendrá el mayor impacto. La colaboración estrecha entre médicos e ingenieros ha creado un sinergismo que ha permitido desarrollar sistemas de interacción virtual y de feedback táctil que tienen y tendrán un gran impacto en la medicina.

3.1. Imágenes médicas

Hace poco más de cien años que se descubrió cómo hacer una imagen del cuerpo humano combinando la fotografía con los rayos X. El premio Nóbel de Física se le otorgó a W. Roentgen en 1901 por este descubrimiento. La combinación de los rayos X y la tecnología computacional producen en 1971 la primera imagen de TAC (Fig. 1), lo que le valió a su inventor, Godfrey Hounsfield, el Premio Nóbel de Medicina en 1979. Así mismo, Felix Block & Edward Purcell descubrieron en 1946 que los núcleos atómicos absorben y reemiten energía de radiofrecuencia, fundando las bases de la resonancia magnética. En 1952 ambos comparten el Premio Nóbel de Física. La formación de imágenes por resonancia magnética es muy reciente, y su origen es controvertido ya que por ser un método más complejo, fueron varias las contribuciones. Raymond Damadian fue quien demostró que los tejidos tumorales tienen una señal más alta que los tejidos normales en RM, y fue quien introdujo la RM en el ámbito de la medicina. En la actualidad, además de estos, encontramos todos los tipos de imágenes que se han nombrado en el apartado anterior.

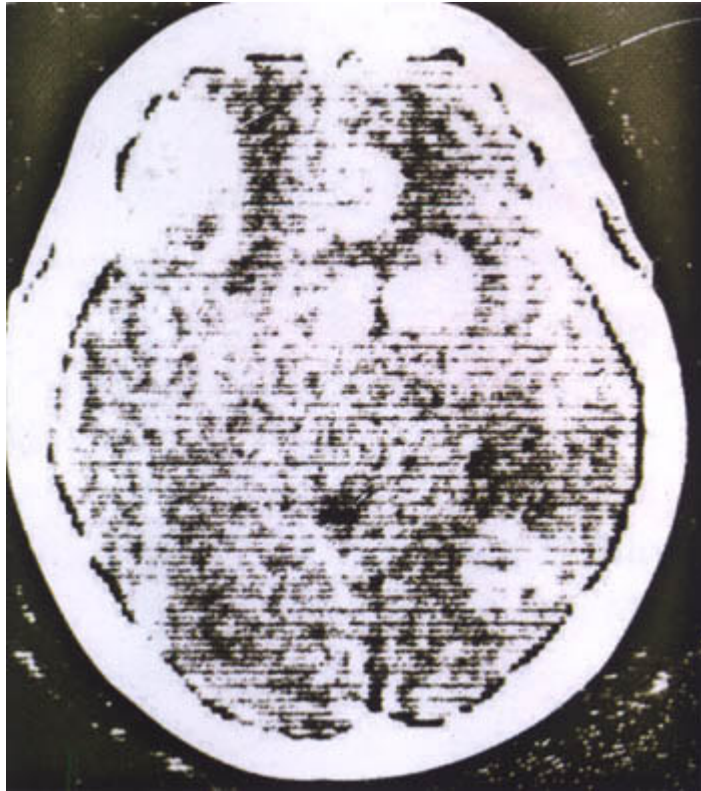


Figura 1: Imagen de la primera tomografía computarizada (TAC) obtenida con el EMI scanner en 1971, demostrándose varias metástasis cerebrales. Su aplicación clínica comenzó en 1973. En Chile el primer tomógrafo se instaló en 1978 en el Centro Scanner.

Esta globalización ha hecho absolutamente necesaria la incorporación de un estándar de comunicación que permita el libre tráfico de imágenes entre los diversos sistemas que requieren de la utilización directa de la información digital. El desarrollo de Internet ha permitido además que muchos ingenieros, físicos y médicos reúnan la información y publiquen en la red los elementos esenciales de cada formato propietario, para poder decodificar y utilizar las imágenes (ver <http://www.dclunie.com>). Al tener esta urgente necesidad de comunicación surgió DICOM, un estándar creado por la NEMA (National Electrical Manufacturers Association) y mantenido en colaboración con la RSNA (Radiological Society of North America). Sobre el estándar de imágenes DICOM, que es con el que trabaja la fusión 3D, hablaremos más detenidamente en el apartado de Investigación, en la última parte de la memoria.



4. OBJETIVOS

4.1. Objetivo inicial

El objetivo principal del proyecto era crear un nuevo entorno de trabajo para el servicio de telemedicina que actualmente ofrece el ITC, que sirviera de renovación al anterior y que además se realizara con vistas a permitir futuras ampliaciones del mismo, como podría ser incluir nuevos servicios telemédicos.

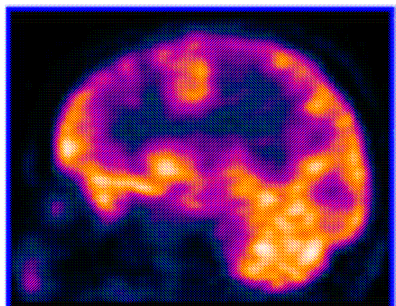
Desde el ITC se pretendía hacer un servicio a través de una página web, en la que un usuario, ya fuese de forma individual, o través de una entidad médica pudiese realizar operaciones de creación y tratamiento de imágenes, tales como fusión, modelización, coloreado... a partir de otras imágenes que el mismo usuario posea.

En principio nuestra aplicación debía consistir en renovar el servicio de la fusión 3D, haciendo uso de tecnologías, e implementando el sistema de forma que permitiese introducir nuevas funcionalidades en el futuro. Este servicio de fusión 3D se centra en el tratamiento de imágenes médicas, tales como radiografías, tomografías computerizadas, escáneres, etc., en formato DICOM, que por la gran potencia de cálculo que necesitan serán enviadas al Supercomputador(ABACO) del CPD de la UCM que será el que devuelva el volumen resultante. Debido al tipo de información que se trata (las imágenes contienen una cabecera con información personal de los pacientes), tiene como uno de los requisitos principales la seguridad, ya que las imágenes viajan a través de Internet.

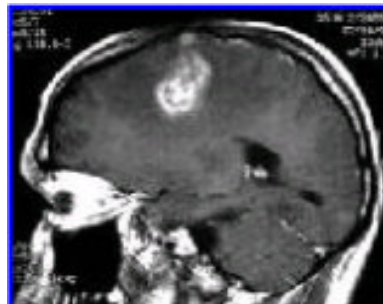
A continuación se muestra el resultado de una fusión 2D, su diferencia con la fusión 3D, y el proceso que siguen las imágenes desde que son elegidas hasta que se obtiene el volumen resultante.

4.2. Fusión 2D

Imágen de PET

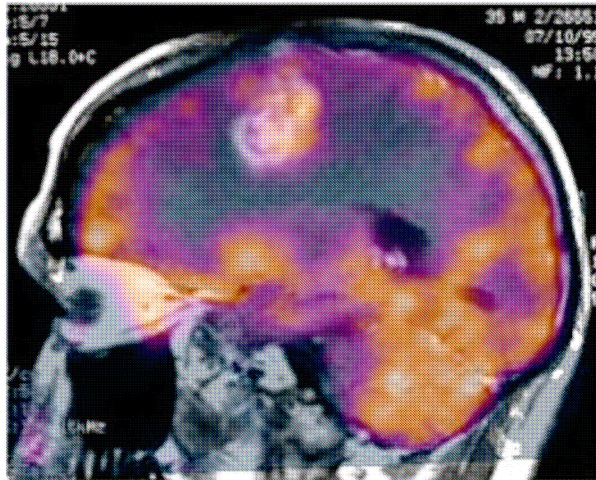


Imágen de RM





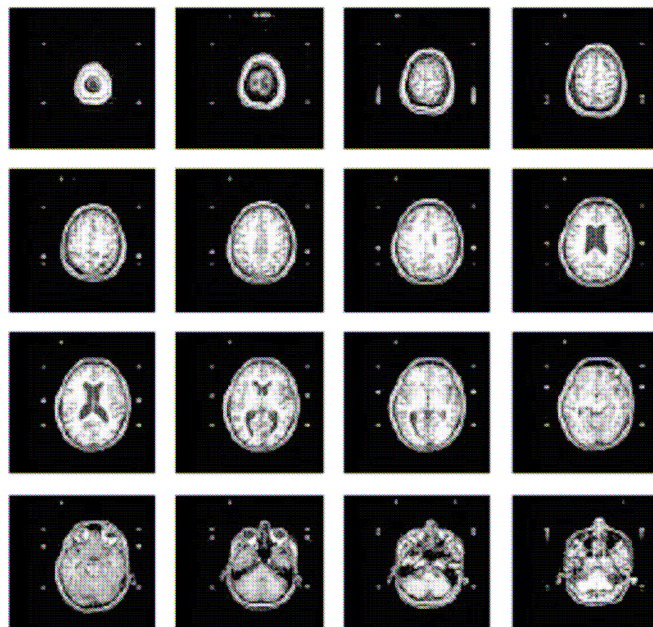
Imágen resultante



Con la fusión 2D obtenemos una única imagen de un corte del cerebro.

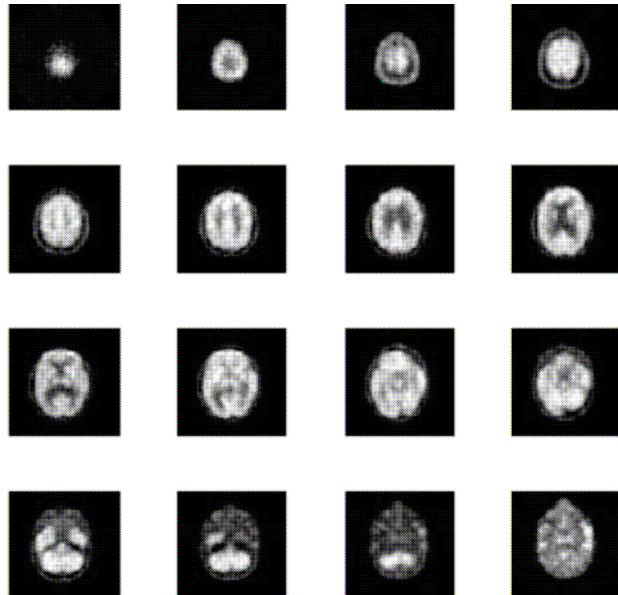
4.3.Fusión 3D

Imágenes de RM de distintos cortes del cerebro

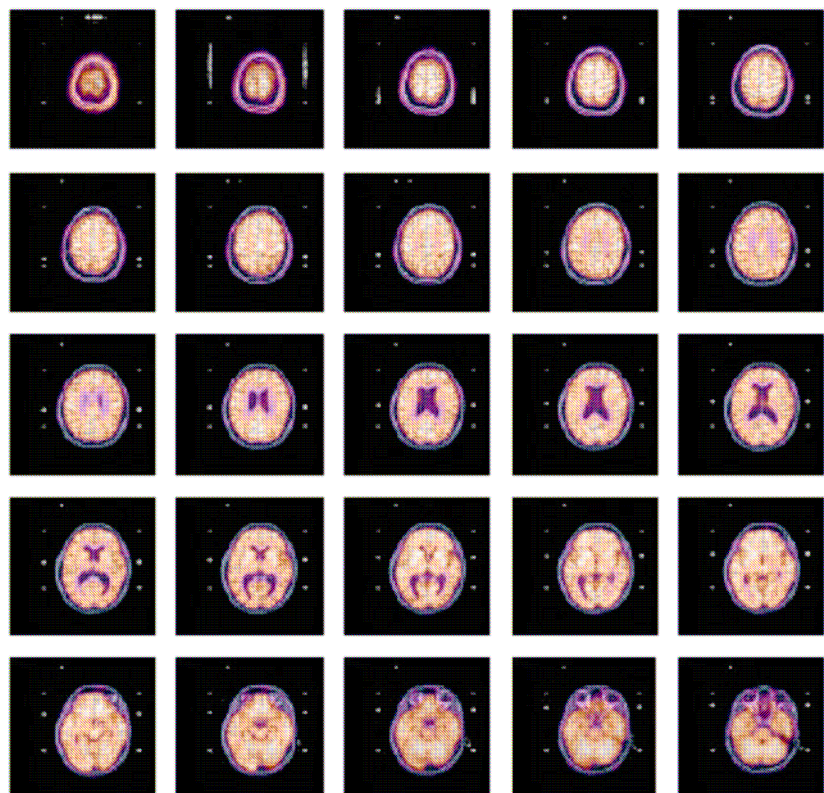




Imágenes de PET de los cortes del cerebro anteriores

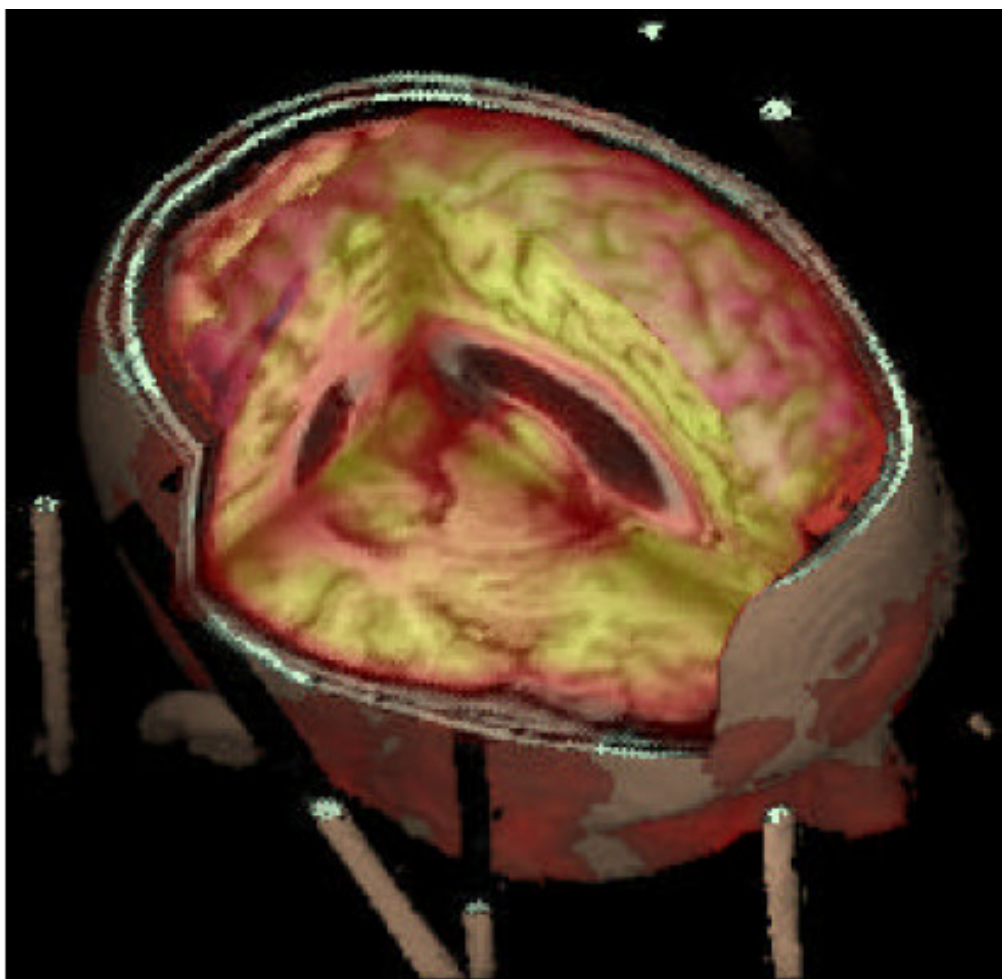


Imágenes fusionadas de cada corte



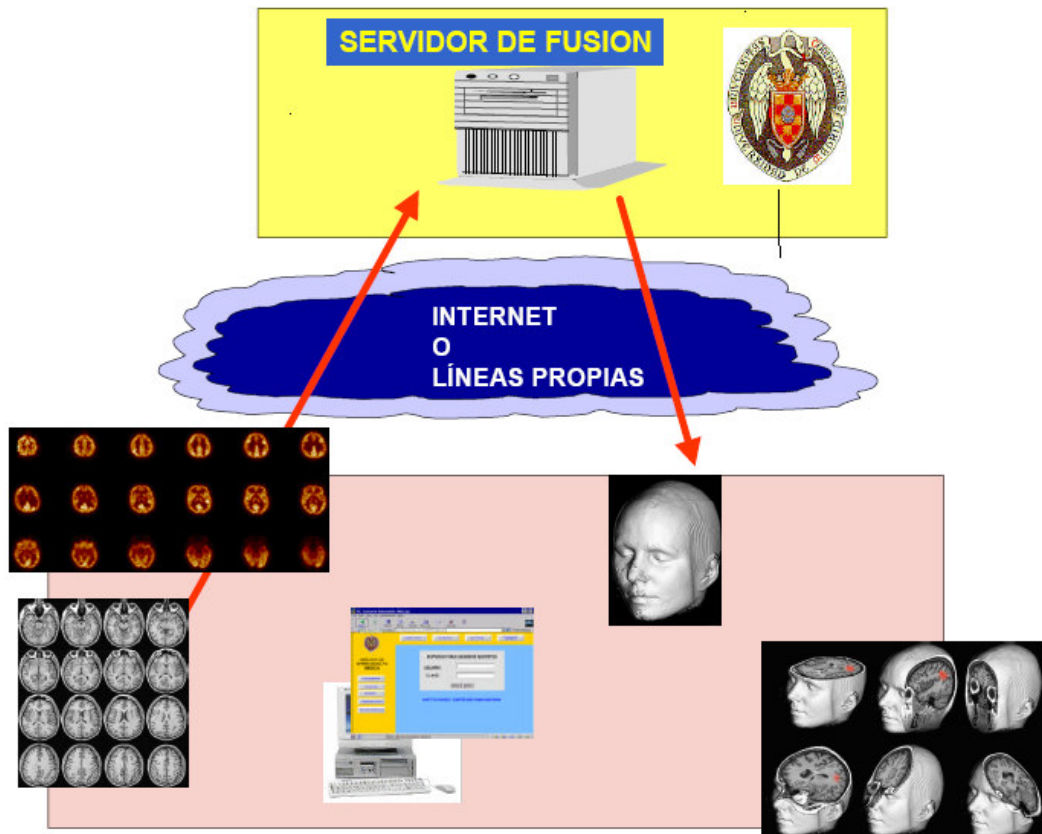


Volumen del cerebro reconstruido





4.4. Proceso completo



Las imágenes son seleccionadas desde una base datos que contiene objetos DICOM, son enviadas a través de Internet al Supercomputador y éste, después de unos minutos, devuelve el volumen resultante.

Por las características de nuestra aplicación, hemos tenido que realizar un estudio de la fusión más detallado que lo explicado en este apartado, pero el resto queda mostrado en la parte de Investigación de la última sección de la memoria.

4.5. Objetivos cumplidos

De la idea inicial, lo que hemos conseguido terminar, es la renovación de la aplicación para la fusión 3D. Se ha desarrollado con nuevas tecnologías, mayor seguridad, nuevas funcionalidades y posibilidad para otras nuevas. El portal web se ha implementado en Java, pero la justificación se explica en el apartado del Diseño e Implementación.



5. PLAN DE TRABAJO

Al comienzo de este proyecto realizamos una serie de documentos internos al grupo que nos servirían durante el resto del curso para organizarnos a la hora de trabajar, investigar y poner en común todo lo que hubiésemos hecho por separado. Es importante puntualizar que al tener un cliente real que iba a usar esta aplicación y a cuyo gusto debía ser diseñada e implementada, la planificación y definición del alcance de este proyecto ha sido mucho más complicada, aunque este tema lo trataremos más detenidamente en apartados posteriores donde veamos los problemas que nos han surgido a lo largo de este desarrollo.

Volviendo a la organización, tenemos que separar dos grandes fases en el proyecto, en las cuales el modo de trabajo y la organización del grupo ha sido distinta según las necesidades de cada momento. Estas dos fases coinciden con los dos cuatrimestres naturales que tiene el curso académico. Y entre estas dos fases ha habido un pequeño período más relajado durante la segunda parte del mes de enero y la primera de febrero para dejar lugar a los exámenes, etapa que no ha tenido lugar para los exámenes de junio.

En la primera fase, nuestro principal cometido fue la investigación, que estuvo encaminada por varias ramas. Puesto que el proyecto se ha realizado en colaboración con el ITC, una parte primordial era conocer todo lo que ya se había hecho con anterioridad, todo lo que había que modificar y todo lo que quedaba por hacer. Por otro lado, y con el consentimiento definitivo de nuestro director de proyecto, teníamos que investigar las tecnologías que íbamos a utilizar para llevar a cabo nuestro trabajo, de modo que estuviesen relacionadas con los conocimientos adquiridos durante nuestra estancia en la Facultad, y con dos objetivos finales, uno, el de que llegase a ser lo que hoy es, el proyecto de fin de carrera, y otro, que fuese la aplicación que cumpliera las necesidades de nuestro cliente.

En base a esto, las reuniones con dicho cliente, fueron más frecuentes durante esta primera fase, ya que era necesario definir con claridad cuál iba a ser nuestro trabajo y cuáles las necesidades de la aplicación, que además tenía proyección para futuras ampliaciones. No había un día predefinido para estas reuniones, sino que teniendo en cuenta la disponibilidad de todos, al final encontrábamos la fecha y hora adecuados.

Nosotros, entre una reunión y otra repartíamos las ramas o temas de investigación, para trabajar por separado, y teníamos nuestros encuentros para poner conocimientos, resultados y dudas en común, que después llevaríamos a la reunión definitiva con nuestro cliente, nuestro director y demás colaboradores. Las reuniones, se han realizado de igual forma en esta fase y en la segunda, los temas y dudas se han expuesto y debatido hasta llegar a un acuerdo final.



Pasados los primeros meses, y tomadas ya las principales decisiones, redactamos los documentos nombrados arriba, para que todo estuviese por escrito para el cliente y para nosotros. Gracias a esto, obtuvimos el primer prototipo, en lo que a implementación se refiere al final de esta fase, que finalmente tuvo éxito con el cliente y después se ha convertido en el proyecto que es ahora.

En la segunda fase, ya coincidiendo con el segundo cuatrimestre, la principal tarea, ha sido la de establecer el diseño definitivo de la aplicación e irlo implementando.

Para llevar a cabo esta tarea , los tres componentes del proyecto nos hemos reunido al menos un día durante todas las semanas desde febrero hasta junio, para poner en común el trabajo individual que pudiésemos haber realizado, para integrar código y para comentar los avances en la investigación. Además nos hemos reunido con el cliente, aunque bastante menos que en la primera fase, y también con nuestro director, pero podríamos decir, que nosotros hemos sido el motor que ha hecho posible que lleguemos hasta aquí, por nuestra perseverancia y trabajo.

Nos parece interesante aclarar que en esta parte del proyecto la comunicación no siempre ha sido bis a bis. En muchas ocasiones no ha habido tiempo para planificar una reunión con todos los miembros, y por necesidad de aclarar ciertos puntos y de proporcionar determinada información, se ha llevado a cabo mediante correo electrónico. Esto se ha llevado a cabo internamente en el grupo y también externamente con el cliente, el director y demás colaboradores.

No nos olvidamos de que, aunque en esta fase la implementación ha tenido un gran peso, hemos continuado con la investigación, como nombramos un poco más arriba. De manera que, puesto nuestras expectativas iniciales para el proyecto eran demasiado ambiciosas y no todas ellas se han podido cumplir, dejaremos constancia en esta memoria, en apartados sucesivos de todo lo que hemos descubierto, para que en posibles proyectos futuros se hagan mejoras y ampliaciones.



6. SEGUIMIENTO

Nuestro director de proyecto, Luis Hernández Yáñez, ha realizado un seguimiento frecuente del proyecto, nos ha ayudado con dudas y nos ha aconsejado siempre que ha sido necesario. Lógicamente, en la segunda parte del proyecto el seguimiento ha sido más exhaustivo puesto que era la fase más importante, y la definitiva.



7. PROBLEMAS SURGIDOS

Como hemos explicado ya en varias ocasiones, esta aplicación forma parte de un proyecto mucho más grande, que se está llevando a cabo desde hace varios años en el ITC de la UCM. Aunque para nosotros ha sido una gran satisfacción poder colaborar con la investigación en el campo de la Telemedicina, nos hemos dado cuenta de que nuestras expectativas iniciales pretendían acaparar campos a los que finalmente no hemos podido llegar, porque realizar una aplicación para un cliente determinado no siempre es fácil.

Han sido dos los principales problemas encontrados a lo largo del desarrollo de este proyecto:

El primero de ellos ha sido el de que ya existieran partes implementadas que había que conocer, y en algún caso modificar. Mientras nosotros hemos estado implementando, hemos tenido la necesidad de conocer partes de código y partes de la comunicación entre el supercomputador y el servidor de imágenes por un lado, y entre el usuario y el servidor de la aplicación por otro, y no siempre nos ha sido fácil que nos proporcionasen esa información. A veces, porque en el ITC no sabían lo que les pedíamos y otras porque había que preservar la seguridad y privacidad de datos que aparecían en las imágenes y en zonas del código.

En segundo lugar, ha sido un problema importante tener que tomar una decisión sobre las tecnologías utilizadas, porque se pretendía hacer uso de conocimientos adquiridos durante la carrera, a la vez se ha querido hacer un sistema más moderno, más seguro, y más fácilmente ampliable que el que había. Pero cuando hemos conseguido todo esto, nos ha supuesto un trabajo fatigoso (que finalmente no hemos podido concluir) investigar cómo unir toda nuestra aplicación con las partes del proyecto del ITC que no se iban a renovar, ni modificar, al menos a corto plazo.

Desde nuestro punto de vista, todo hubiera sido más fácil si el conocimiento que había en el ITC del proyecto que ya estaba hecho, hubiese sido a otros niveles más bajos que el del usuario, porque hemos empleado demasiado tiempo en investigar, y porque a veces hemos dado “palos de ciego” y no hemos ido bien encaminados.

Algo que también ha influido en nuestro proyecto, es que los tres integrantes del grupo hemos estado y estamos trabajando, con lo que hemos tenido que realizar un esfuerzo mayor para trabajar en grupo y para reunirnos.

Por otro lado, el hecho de que nuestro cliente y colaboradores del ITC estuviesen en Somosaguas, ha supuesto una mayor dificultad a la hora de adquirir o intercambiar información.



8. REQUISITOS INICIALES

Tras una primera reunión con el cliente, nos comunicaron los siguientes requisitos para la aplicación, en base a los cuales comenzamos nuestro diseño:

- En la aplicación se podrán registrar tanto usuarios particulares, como empresas o instituciones del campo de la medicina.
- En un futuro podría llegar a cobrarse por el servicio prestado.
- No debe solicitarse demasiada información a los usuarios para su registro, sólo lo esencial.
- Debe tener una interfaz intuitiva y fácil de usar, puesto que la mayoría de usuarios no tendrán altos conocimientos informáticos.
- Debe ser una aplicación austera, no debe necesitar muchos recursos, ya que la mayoría de ordenadores del personal médico poseen pocos recursos, además, podría investigarse el incluir una conexión directa con la aplicación desde las máquinas médicas, dado que éstas poseen un ordenador en su interior.
- Debe ser modular, fácilmente ampliable y modificable.
- Dado que intercambia información confidencial (historiales médicos) debe estar preparada para soportar la mayor seguridad posible.
- Debe ser una aplicación lo más autónoma posible, que no requiera administración exhaustiva.
- Debe administrarse fácilmente.
- Debe permitir a los usuarios tener un periodo de prueba de la aplicación.
- Deberán guardarse datos de seguimiento de los usuarios, que permitan a las Entidades / Instituciones un mayor control sobre las altas y bajas de sus usuarios, así como sobre el uso de éstos de la aplicación

En posteriores reuniones con el cliente, fuimos presentando diversos bocetos del diseño de la aplicación, de ellos se eligió uno sobre el que trabajar, y que hemos ido modificando y ampliando.

Presentamos a continuación el diseño final de la aplicación.



9. DISEÑO DE LA APLICACIÓN

Para el diseño de la aplicación tuvimos que tener en cuenta los requisitos iniciales impuestos por el cliente, en base a los cuales se fueron definiendo las distintas partes de la aplicación.

En los siguientes puntos se informa de cómo se llevó a cabo el diseño de las diferentes partes de la aplicación teniendo en cuenta dichos requisitos, o de cómo influye dichos diseños sobre los requisitos iniciales.

Con esto no se pretende mostrar todo el proceso seguido en la elaboración del diseño, sino solamente los puntos más importantes del mismo.

9.1. Tipos de usuarios y su porqué. Funcionalidad de cada usuario.

Atendiendo a los requisitos iniciales, dividimos las funcionalidades de la aplicación según el tipo de usuario que se conectara a ella, debiendo definir para ello los tipos de usuario que accederán a la aplicación. Mostramos a continuación los tipos de usuario definidos.

(Para más información véase también los casos de uso)

9.1.1. Usuario Particular

Es el tipo de usuario básico, puede pertenecer a una Entidad / Institución si dicha entidad lo registró. Tiene asociado los datos particulares con los que se registró en la aplicación, entre ellos el e-mail, utilizado por nuestra aplicación para la comunicación con el usuario.

Entre sus funcionalidades están la de modificación de sus datos de registro o la baja en el servicio.

En un futuro, este usuario permitirá el acceso a la fusión de volúmenes.

9.1.2. Usuario Entidad / Institución

Este tipo de usuario permitirá la administración de usuarios particulares que pertenezcan a la Entidad / Institución.

Todas sus funcionalidades son relativas a la administración de los usuarios particulares asociados a la Entidad / Institución, como por ejemplo, alta de usuarios, baja de usuarios, modificación de datos de los usuarios, ...

Una Entidad / Institución dará de alta a un usuario particular (que estará asociado a la Entidad / Institución) de esta forma, el nuevo usuario podrá acceder a los servicios como un usuario particular.



Se ha elegido este diseño, pensando en la futura ampliación que permita el cobro de los servicios, dado que podría cobrarse a una Entidad / Institución por el número de usuarios que hayan usado los servicios, dejando a la Entidad / Institución la administración de dichos usuarios.

Este diseño mejora además la autonomía de la aplicación, puesto que no es necesario un administrador que realice las altas y bajas de usuarios, sino que será la Entidad / Institución quien lo haga, permitiendo además, mayor libertad a la Entidad / Institución en la gestión de recursos.

9.1.3. Usuario Demo

Este tipo de usuario mantiene las mismas funcionalidades que el usuario particular. Su diferencia es que no tiene asociado en sus datos de registro más que un e-mail y la fecha de alta.

Se definió para permitir la prueba de la aplicación durante un período de 30 días, tiempo que permanecerá activa su cuenta y tras el cuál no podrá acceder a la aplicación.

9.1.4. Usuario Administrador

El usuario Administrador será el encargado de mantener la aplicación, posee funcionalidades para el alta, baja, modificación, y eliminación de usuarios, tanto usuarios Entidades / Instituciones como usuarios Particulares o usuarios Demo. Así mismo, posee funciones que muestran listados de los distintos tipos de usuarios.

El usuario Administrador se definió con la idea de que mantuviera la aplicación remotamente y pudiera llevar a cabo acciones de limpieza de las bases de datos, así como posteriores análisis de los datos de los usuarios, con fines estadísticos y de estudio de recursos.

9.1.5. Usuario General

Aunque no está definido como usuario en ninguna parte de la aplicación, entenderemos como usuario general a todo aquel que accede a la aplicación, antes de iniciar sesión como alguno de los usuarios mencionados anteriormente.

Las funcionalidades que posee este tipo de usuario serían el registro como nuevo usuario de la aplicación, el recordatorio de contraseña, y el inicio de sesión.



9.2. Diseño de las clases, organización y jerarquía

El diseño de las clases que conforman la aplicación se realizó buscando la mayor modularidad posible, permitiendo además la ampliación de los servicios de la aplicación sin muchas dificultades.

Por estas razones, creamos 3 paquetes de clases separados, cada uno independiente, que proporcionaran las funcionalidades extras, necesarias para las clases de la aplicación.

Dichos paquetes son:

9.2.1. BD

Este paquete está compuesto por las clases BaseDatos y DAOUsuario, y son una abstracción que proporciona métodos de acceso y modificación de tablas en la base de datos de la aplicación.

BaseDatos contiene los métodos que permiten manejar la Base de Datos (en nuestro caso una base de datos MySql). Se encarga de la conexión con el servidor de base de datos y gestiona las peticiones SQL. Contiene por tanto, los datos de conexión así como información del usuario y contraseña de la base de datos.

Esto en un principio pudiera parecer un inconveniente, ya que en caso de querer cambiar cualquier dato del servidor, como el usuario o la contraseña o el puerto de conexión, necesitaría realizarse sobre el código de la clase y volver a compilar esta clase para que la aplicación utilizara los nuevos datos. Sin embargo esta solución nos garantiza más seguridad, ya que los datos de conexión no son accesibles, al estar compilados en un archivo .class.

En caso de querer sustituir la base de datos por otra que no sea MySql, no habría más que modificar esta clase, incluyendo los conectores de la nueva base de datos y modificando las llamadas SQL para adaptarlas al nuevo entorno. Esto nos permite una escalabilidad muy positiva, al no relegarnos a un solo tipo de base de datos.

La clase DAOUsuario (**Data Acces Object Usuario**) contiene la transformación de los datos devueltos por la clase BaseDatos (dicha clase devuelve conjuntos de resultados con los datos almacenados en las tablas de la base de datos en datos) en datos de tipos conocidos por java y manejables por la aplicación.

Posee métodos para la extracción de datos según el tipo de usuario del que se van a obtener los datos.



Se creó esta clase por que, aunque la clase BaseDatos ya obtiene los mismos resultados, éstos sólo son válidos mientras la conexión con la base de datos permanece abierta; por tanto, y ya que no nos interesaba mantener una conexión abierta si no íbamos a realizar más peticiones, creamos la clase DAOUsuario, que almacena los datos obtenidos para su uso posterior, permitiendo procesarlos y operar con ellos tras cerrar la conexión con la base de datos.

9.2.2. Logica

Este paquete contiene las clases que gestionan la lógica de comunicación con el usuario, que en nuestro caso es mediante correo electrónico.

Se basa en el desarrollo de 2 clases, Mail y Logica.

Logica contiene métodos para el envío de correos electrónicos. Estos métodos se utilizarán para informar al usuario de los distintos eventos relacionados con su cuenta en el servicio (nuevas altas, bajas,...) así como otros eventos (recordatorio de contraseñas, avisos del administrador,...).

En la clase Mail, se implementa la funcionalidad del envío de mails a desde java, a través del api javaMail.

9.2.3. Sesion

Esta clase contiene los datos necesarios a lo largo de la sesión http, así como los métodos de acceso y modificación de estos datos.

Se pensó en crear una clase que contuviera dichos datos, en lugar de almacenarlos directamente en el objeto HttpSession (proporcionado por los servlets) porque mejorabamos la modularidad, ya que de esta manera, si queremos incluir nuevos datos o eliminar alguno de los utilizados en la sesión, no tenemos más que incluirlos (o eliminarlos) en esta clase, sin embargo, si hubiésemos adoptado la segunda solución, sería necesario modificar todos los servlets en los que se hiciera referencia a los datos.

Así, todos los servlets que necesitan datos de la sesión, mantienen objetos Sesión que guardan en sus HttpSession.

9.2.4. Servlets

El resto de la aplicación se basa en llamadas a los diferentes servlets que proporcionan la funcionalidad necesaria según el tipo de usuario.

Es por eso, que los servlets tienen una división (imaginaria) según el tipo de usuario al que proporcionen funciones exclusivas. De esta forma,



todos los servlets cuyo nombre de clase comience por 'Ad' proporciona métodos exclusivos para el usuario administrador, y sólo podrá ser utilizado en un contexto en el que la petición la realice un usuario que haya iniciado sesión como administrador. De la misma manera, 'Us' hace referencia a los servlets de los usuarios particulares y por 'En' comienzan las clases de los usuarios Entidad / Institución.

Todos los servlets comprueban el tipo de usuario antes de realizar ninguna otra tarea, mostrando un error en caso de que el tipo de usuario no corresponda con el servlet.

Las clases que no comienzan por ninguno de los prefijos anteriores, como por ejemplo ErrorInicio, pertenecen a servlets válidos para cualquier tipo de usuario, como podrá comprobarse, suelen ser clases que reflejan errores en la aplicación o funcionalidades pertenecientes al tipo de usuario general (tales como inicio de sesión o registro como nuevo usuario).

9.3. Seguridad

Otro de los requisitos iniciales de la aplicación, era mantener la máxima seguridad posible, dado que albergaríamos datos particulares de los usuarios.

Este requisito incidió en ciertos puntos del diseño. Como por ejemplo la creación de la clase DAOUsuario, ya que al almacenar todos los datos del usuario en un objeto aumentamos la seguridad y aunque obtengan los datos de una comunicación, no tendrían ningún interfaz para la transformación del objeto y sería más difícil extraer los datos.

No obstante, este punto tuvo más repercusión en la implementación de la aplicación, donde se comentará con más detalle.

9.4. Autonomía

Para poder optimizar la autonomía de la aplicación, hemos tenido que diseñar a los usuarios y sus procesos de forma que no necesitaran de un administrador, por ello no es necesario nada para registrarse como nuevo usuario (salvo datos de registro), y por ello también el usuario Entidad / Institución está diseñado para que gestione sus usuarios. De esta forma nos liberamos de estas tareas, típicas de un administrador.

Así, sólo será necesario la intervención inmediata de un administrador, en caso de un fallo, ya sea en el servidor de bases de datos o el servidor web.

9.5. Requisitos hardware



Este requisito no incide principalmente en el diseño de la aplicación, si no en su implementación.

Se tratará, por tanto, en dicho apartado.

9.6. Facilidad de uso

Hemos intentado cumplir este requisito en nuestro diseño de las funciones de los diferentes tipos de usuarios. Ya que serán ellos los que utilicen la aplicación.

Los usuarios particulares poseen métodos simples, de forma que no sea nada difícil utilizar la aplicación.

La gestión de usuarios por parte de los usuarios Entidad / Institución, es la más básica posible, de esta forma, permitimos una administración elemental, que cumple los requisitos necesarios además de ser fácil e intuitiva.

Igualmente, la administración general de la aplicación no esta llena de innumerables funcionalidades, que a veces no hacen sino dificultar su gestión. Si no que posee métodos de administración básicos e idénticos para los distintos tipos de usuario.

Otro punto a favor, es que la administración se puede realizar de forma remota, facilitando la labor al administrador al no tener que desplazarse para realizarla.

9.7. Escalabilidad

La creación de los paquetes y la división de los servlets según al tipo de usuario al que pertenecen son los puntos fuertes de la escalabilidad de nuestra aplicación.

Con los paquetes logramos proporcionar funcionalidades al resto de clases, cualquier cambio o ampliación podrá seguir utilizando dichos paquetes.

Si el cambio fuera en alguno de los paquetes, será igualmente visible al resto de clases sin necesidad de realizar grandes cambios, por ejemplo, si cambiásemos los datos personales de los usuarios para almacenar su ciudad de nacimiento, habría que modificar levemente el paquete de base de datos y los servlets que vayan a utilizar este nuevo dato.

La división (imaginaria) de los servlets permite controlar por separado los servicios que se proporcionan a los diferentes tipos de usuario. De esta forma, ampliaciones de funcionalidad para un tipo de usuario no repercutirán al resto, y si se quiere que sí repercuta, los cambios a realizar son muy similares, facilitando esta tarea.



10. DESARROLLO E IMPLEMENTACIÓN

Siguiendo el diseño realizado, pasamos a implementar la aplicación. Para ello también tuvimos que tomar decisiones, que afectarían de forma importante al resultado final.

Dichas decisiones estuvieron guiadas por los requisitos iniciales dados por el cliente, y en muchos casos refuerzan y complementan al diseño elegido.

Por ello mostramos a continuación las decisiones más relevantes tomadas en la implementación, muchas de las cuales ya aparecieron en el diseño.

Comenzaremos indicando la arquitectura general del sistema, para luego pasar a detallar cada parte de la misma.

10.1. Arquitectura del sistema

Nuestra aplicación está realizada principalmente en Java. Utilizamos Java tanto en los paquetes que manejan funcionalidades externas a la comunicación web, como en los servlets sobre los que se basa el portal. Utilizamos también código HTML en la generación de páginas web, así como JavaScript.

La aplicación está alojada en un servidor Tomcat.

Hemos optado por una base de datos MySQL.

Para el envío del mail desde la aplicación es necesario un servidor SMTP. Existen servidores SMTP gratuitos para la mayoría de sistemas operativos, con lo cual, no indicaremos ninguno en particular aquí.

10.2. Lenguaje utilizado

Para la implementación se ha utilizado java, tanto en la creación de los paquetes de conexión con la base de datos, como en la creación de servlets.

Se eligió java, porque es un lenguaje muy utilizado, potente, con muchas funcionalidades e independiente de la plataforma.

Al principio del proyecto no sabíamos en qué tipo de sistema operativo íbamos a trabajar. Java es un lenguaje que posee entornos de desarrollo en todos los sistemas operativos, lo cual le daba puntos frente a otros.

Además, la aplicación era un portal, luego eso restringía los lenguajes a aquellos que permiten creación de aplicaciones web.

Otro punto más fue que a lo largo de la carrera ya habíamos manejado java, sin embargo con otros lenguajes necesitaríamos un periodo inicial de investigación.



Java es ampliamente utilizado, y hay una gran cantidad de APIs y librerías desarrollados para Java, esto podría sernos muy útil en la implementación por tanto seguía ganando Java como lenguaje a utilizar.

Por último, Java nos permitía una división muy clara en clases y paquetes, lo cual facilitaría su mantenimiento e incrementaría la modularidad.

Tras consultarlo con el cliente, llegamos a la conclusión de utilizar Java como lenguaje principal del proyecto.

Hemos de decir, que en la implementación del proyecto también se han utilizado otros lenguajes, como HTML para la creación de las páginas web, y javascript para el manejo de eventos en dichas páginas.

10.3. ¿Por qué servlets?

A la hora de implementar el proyecto, existían muchas alternativas para crear el portal. Desde utilizar php, o asp, hasta continuar utilizando las cgi's existentes.

Php era una buena alternativa, ampliamente utilizado y fácil de programar, sin embargo, nuestro conocimiento de java era mayor.

ASP requería un servidor IIS de Microsoft y nos cerraba la opción de utilizar un servidor que corriera en Linux (o Unix). Aunque existen servidores compatibles con ASP para Linux, no creímos que fueran una buena alternativa. Sobre todo, cuando eligiendo otro lenguaje, podríamos utilizar Apache o Tomcat, que, además de ser gratuitos, cuentan con un historial que les hace ser servidores estables y seguros. También debemos decir que existen soluciones Apache y Tomcat para la mayoría de sistemas operativos, por lo que ante una eventual migración del servidor no tendríamos mayores problemas.

Nosotros optamos por utilizar servlets. También podríamos haber optado por basarnos en una arquitectura de struts. Sin embargo, nuestro desconocimiento inicial de struts nos impidió elegirlo como posible alternativa.

Por otro lado, y como vimos después, struts necesitaría utilizar jsp's para mostrar los resultados, sin embargo el rendimiento de los jsp's es menor, ya que deben compilarse y ejecutarse en el servidor antes de devolver una respuesta al cliente. Como además, las respuestas del servidor no serían muy elaboradas (recordemos que las páginas deberían ser lo más austeras posible para minimizar los recursos hardware necesitados por el cliente) optamos por encapsular todo el código que genera la respuesta dentro del propio servlet.

No obstante, en cualquier ampliación posterior, podríamos pasar del proyecto actual a struts sin tener que re-implementar todo. Bastaría con devolver



el código adecuado en el servlet (código que nos indicará que jsp se va a mostrar a continuación) en lugar de generar la página html. Ya que los servlets seguirán manteniendo el flujo de la aplicación.

10.4. ¿Por qué no ampliar las cgi's?

El portal a renovar estaba realizado en cgi. Una posible opción hubiera sido actualizar y ampliar las cgi's para permitir las nuevas funcionalidades, creando una base fuera escalable.

Aunque existen formas de actualizar las cgi's obteniendo los mismos resultados, nosotros creímos que era mejor cambiar hacia una tecnología menos árida y más utilizada. Dando un repaso a las tecnologías y lenguajes utilizados actualmente para la programación de aplicaciones web, vimos que cgi era cada vez menos utilizado, y que no podía competir en facilidad de uso, modificación o ampliación con otros lenguajes.

10.5. ¿Por qué MySql?

Ante los diferentes servidores y clientes de bases de datos existentes en el mercado, nosotros elegimos MySql.

Aunque MySql no sea un servidor muy potente (no podemos definir claves externas, ni gestionar tantas conexiones simultáneas como con otros gestores de bases de datos) dada la estructura de las tablas para nuestra aplicación, vimos que MySql era suficiente para nuestros propósitos.

Además, MySql poseía la ventaja de la gratuidad.

Como en los otros casos, también investigamos la repercusión que tendría un cambio en el gestor de bases de datos. Podemos afirmar que el cambio afectaría únicamente a un par de clases (las del paquete de base de datos) que deberían modificarse para acceder a este nuevo entorno.

10.6. ¿Por qué Tomcat?

Actualmente, Apache y Tomcat son los servidores más utilizados. Son gratuitos, existen versiones para casi cualquier sistema operativo, permiten incluir seguridad fácilmente, tienen un gran rendimiento, y su grupo de desarrollo continúa trabajando para mejorarlos día a día.

Ante estas razones no cabía discutir qué servidor utilizar. Aunque existen buenas alternativas en el mercado (como por ejemplo Weblogic). La simpleza de Tomcat le otorgaba un puesto superior.

Tomcat es el proyecto de Apache para servir servlets y jsp's. Por lo cual elegimos utilizar Tomcat solamente, ya que nos ofrece las funcionalidades que necesitamos para nuestra aplicación.



10.7. Seguridad

La seguridad de la aplicación se implementó mediante el uso de SSL en las transacciones de información sensible.

SSL nos permitía afirmar que la información viajaría de forma segura a través de la red, ya que es el protocolo que actualmente se usa incluso en transacciones bancarias.

También la configuración del servidor Tomcat se realizó de forma que aumentara la seguridad del portal, impidiendo el acceso a directorios en modo index.

La separación entre la parte pública y la administración del sistema nos garantiza que sólo los administradores conocen la dirección de acceso. Además estas páginas no son visibles ni accesibles sin la dirección de las mismas.

Otro punto que aumenta la seguridad es el uso de contraseñas de usuario para el acceso a los servicios de la aplicación. De esta manera, sólo una persona que tenga usuario y contraseña podrá acceder.

También se revisa que el tipo de usuario que solicita una operación tiene permisos para realizarla. Por ejemplo, antes de dar de alta un nuevo usuario de una Entidad / Institución se comprueba que el tipo de usuario que solicita dicha operación es una Entidad / Institución. El tipo de usuario sólo se asigna al iniciar sesión, y se mantiene durante la sesión, por tanto, sólo se atenderán peticiones desde ése navegador que es el que posee la sesión.

En las partes del portal que requieren SSL se verifica que el acceso se realiza a través de SSL para evitar solicitudes desde otros puertos y que no utilicen esta seguridad. Si se intenta realizar un acceso a una de estas partes sin SSL se informará que no es posible continuar.

Otra forma de seguridad, es que la mayoría de cambios del estado de la cuenta de un usuario, son notificados al mismo mediante e-mail. De esta forma, si alguien utilizara una cuenta ajena para dar de baja a usuarios, éstos se enterarían de ello, al comunicárseles la baja en su correo. Además, tanto el alta como baja de usuarios de una Entidad / Institución adjunta información sobre la persona que realiza dicha operación, si la persona no es conocida o los datos son falsos, sabremos que alguien ha accedido a nuestra cuenta y podremos poner remedio a ese agujero de seguridad.

Aunque al final no se realizó, también se pensó en encriptar los datos almacenados en la base de datos. No obstante, se vió que, al menos de momento, esto no sería necesario.



La base de datos MySQL se configuró para que sólo el usuario determinado tuviera acceso a las tablas de la aplicación, negando el acceso incluso al usuario root. Esto, aunque parezca extremo, garantiza que sólo un usuario que conozca la aplicación pueda modificar la base de datos, aunque esta contenga tablas de otras aplicaciones.

10.8. Auto-mantenimiento

La autonomía de la aplicación se implementó creando procesos de registro automáticos, que no requieran de un administrador que los valide.

La administración del sistema se puede realizar de forma remota, aumentando la facilidad de gestión de la aplicación, y permitiendo incluso que se compruebe con mayor frecuencia que si no se realizara de forma remota, ya que el administrador no tendrá necesidad de desplazarse hasta el servidor, podrá comprobar la aplicación en momentos libres del día.

10.9. Requisitos hardware

Uno de los objetivos iniciales era que la aplicación fuera lo más compatible que se pudiera con cualquier sistema, independientemente del hardware o software que utilizara el cliente para acceder al portal.

Considerando las características de los ordenadores de los usuarios, que en término medio, son:

Memoria: 128 Mb

Tarjeta gráfica de 32 Mb

Frecuencia del reloj: 750 Mhz

Por ello se pensó en un portal austero, sin contenidos multimedia, ni flash, ni shockwave, y que ocupara poco, para evitar esperar demasiado en conexiones lentas.

Las páginas creadas son así. Quizás parezcan simples, pero permiten que un número mayor de personas puedan acceder a ellas. Además se han hecho de forma que sean compatibles con los principales navegadores (tantos como hemos podido probar).

10.10. Facilidad de uso

Sólo implementar operaciones básicas, sin complejos menús de elección, incluyendo información sobre el funcionamiento de cada servicio. De esta forma hemos intentado potenciar la facilidad de uso de la aplicación.

También incluyendo múltiples menús de acceso a las operaciones, e informando de cualquier error o anomalía al usuario para que pueda conocer porqué ha ocurrido.



No obstante, hemos incluido manuales de uso, para que todo el mundo pueda llegar a dominar la aplicación.

10.11. Escalabilidad

Dividir las clases en grupos de usuarios, crear servlets para cada posible acción del usuario, incluso dividir las acciones en 2 o más pasos (que se reflejan en tantos servlets como pasos), son decisiones de implementación que permitirán ampliar la aplicación fácilmente.

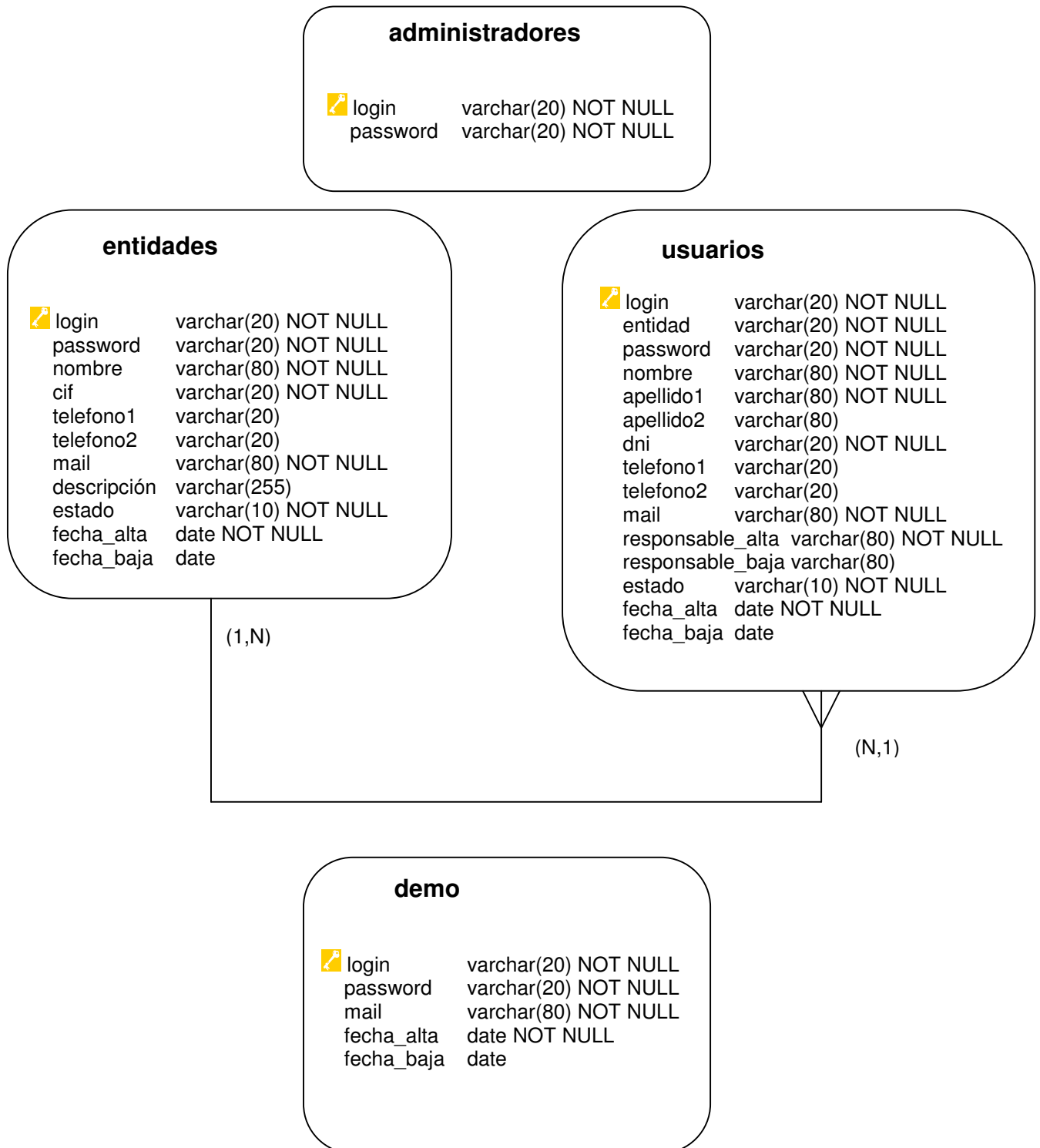
La abstracción de funcionalidades externas a la comunicación web en paquetes de clases es otro punto que mejora la escalabilidad del sistema.

También hemos intentado también comentar al máximo el código creado, para que sea analizado más legible. Y hemos creado documentación de las clases y paquetes creados (javadoc). Que permitirán un análisis general del funcionamiento del sistema sin tener que acceder al código fuente.



11. DIAGRAMA ENTIDAD / RELACIÓN

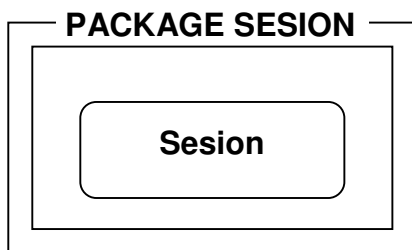
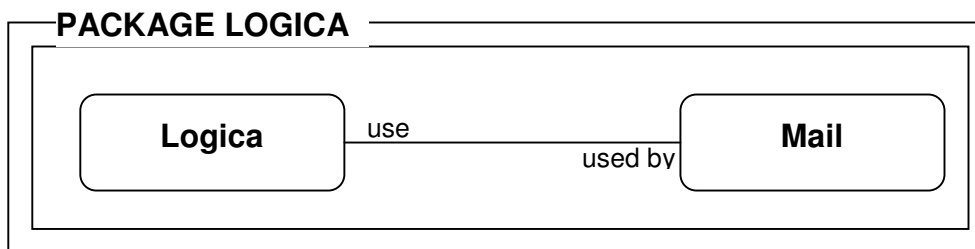
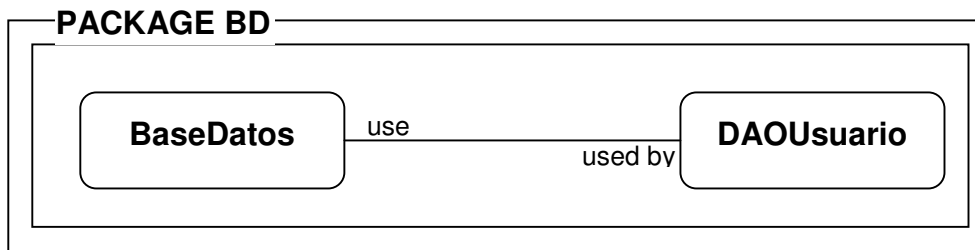
Adjuntamos a continuación el diagrama Entidad / Relación que muestra el diseño de nuestra base de datos. Podemos comprobar aquí la división entre tipos de usuarios y la relación entre usuarios Entidad / Institución y usuarios Particulares (en el diagrama bajo el nombre 'usuarios')





12. PAQUETES DE LA APLICACIÓN

Mostramos ahora un esquema de la composición de los diferentes paquetes utilizados en la aplicación. Dichos paquetes, que ya se analizaron anteriormente, se exponen aquí para facilitar la comprensión de la arquitectura interna de la aplicación.



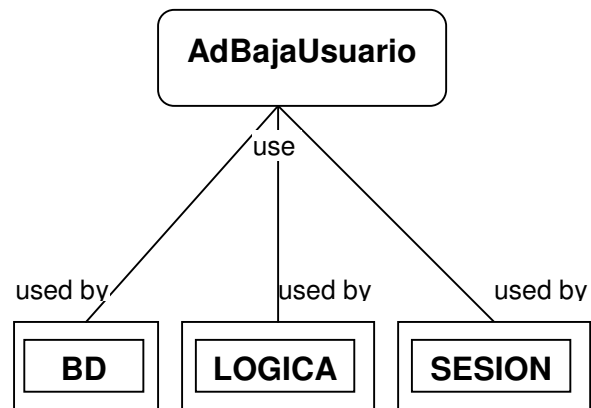
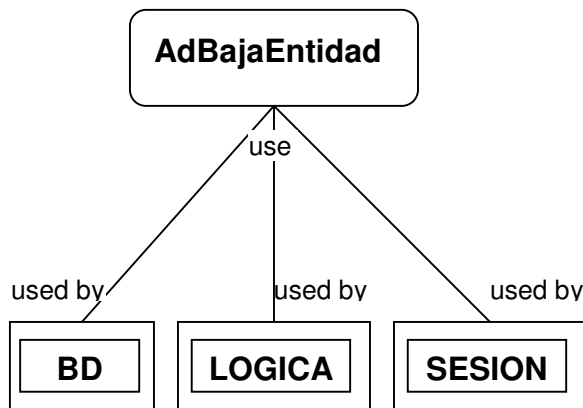
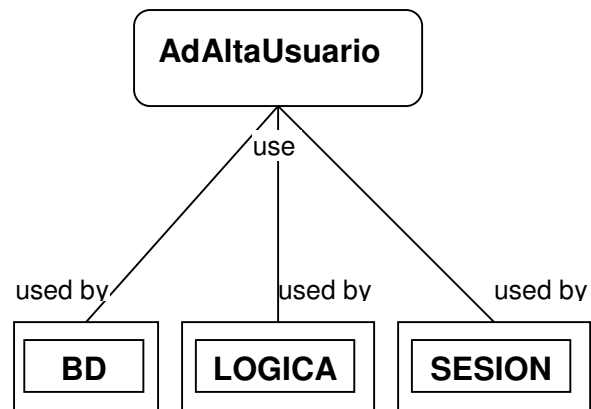
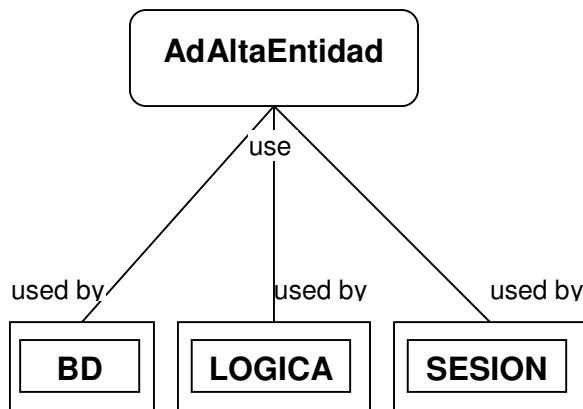


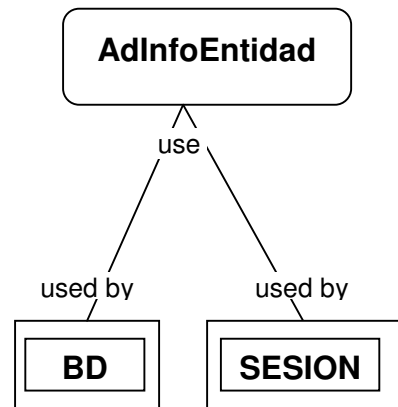
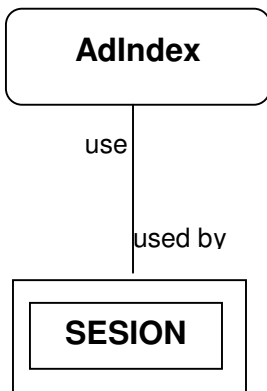
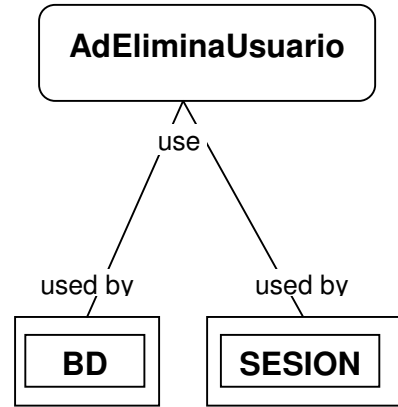
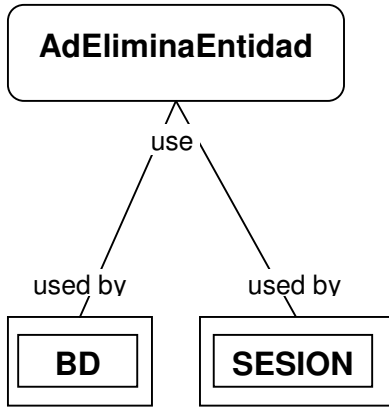
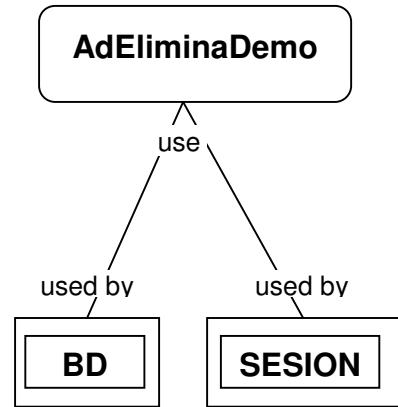
13. DIAGRAMAS DE JERARQUÍA DE CLASES

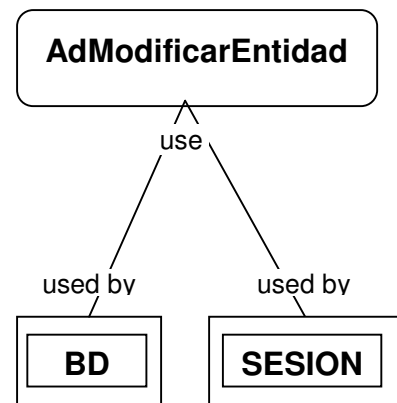
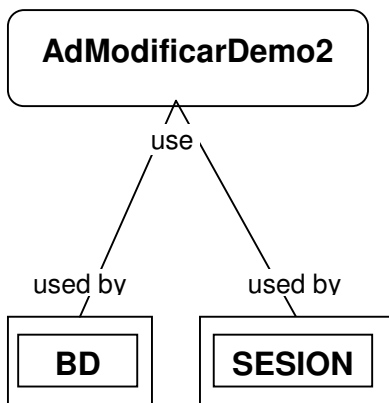
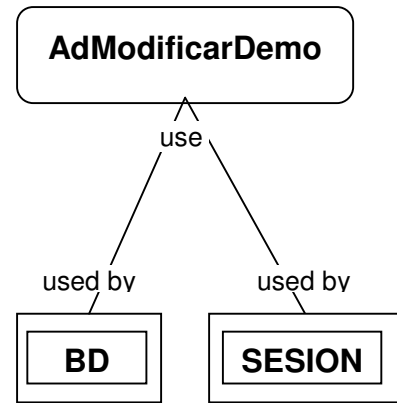
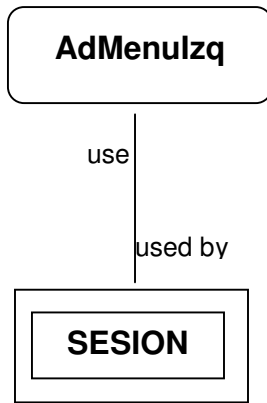
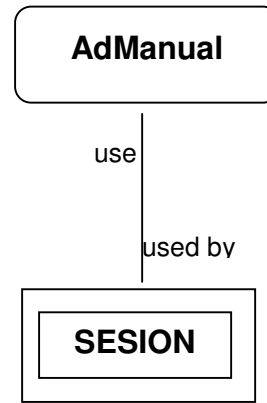
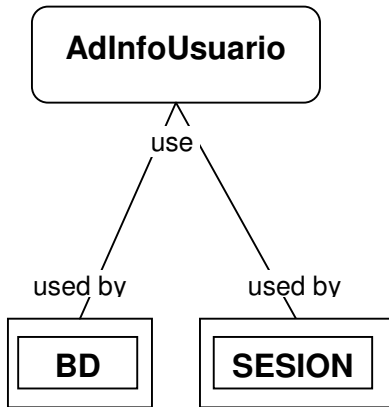
Los siguientes diagramas quieren mostrar cómo se establece la jerarquía de clases de la aplicación, indicando para cada clase, qué paquetes o clases de la propia aplicación utiliza.

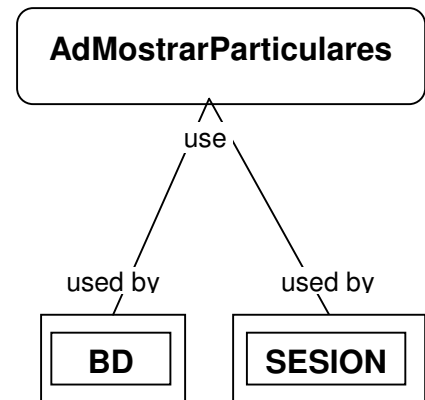
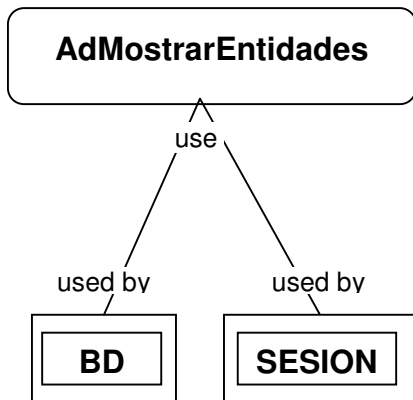
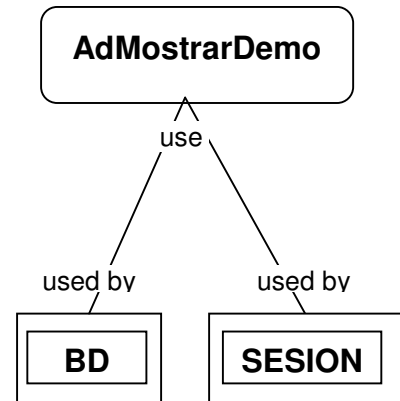
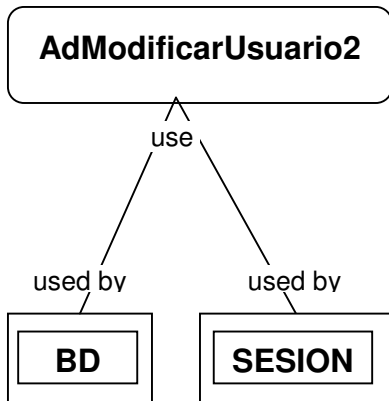
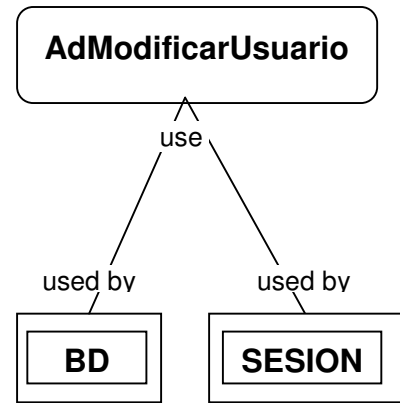
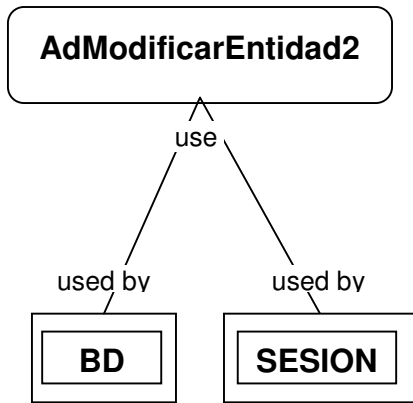
Posteriormente se incluye una tabla-resumen por cada tipo de usuario con la misma información.

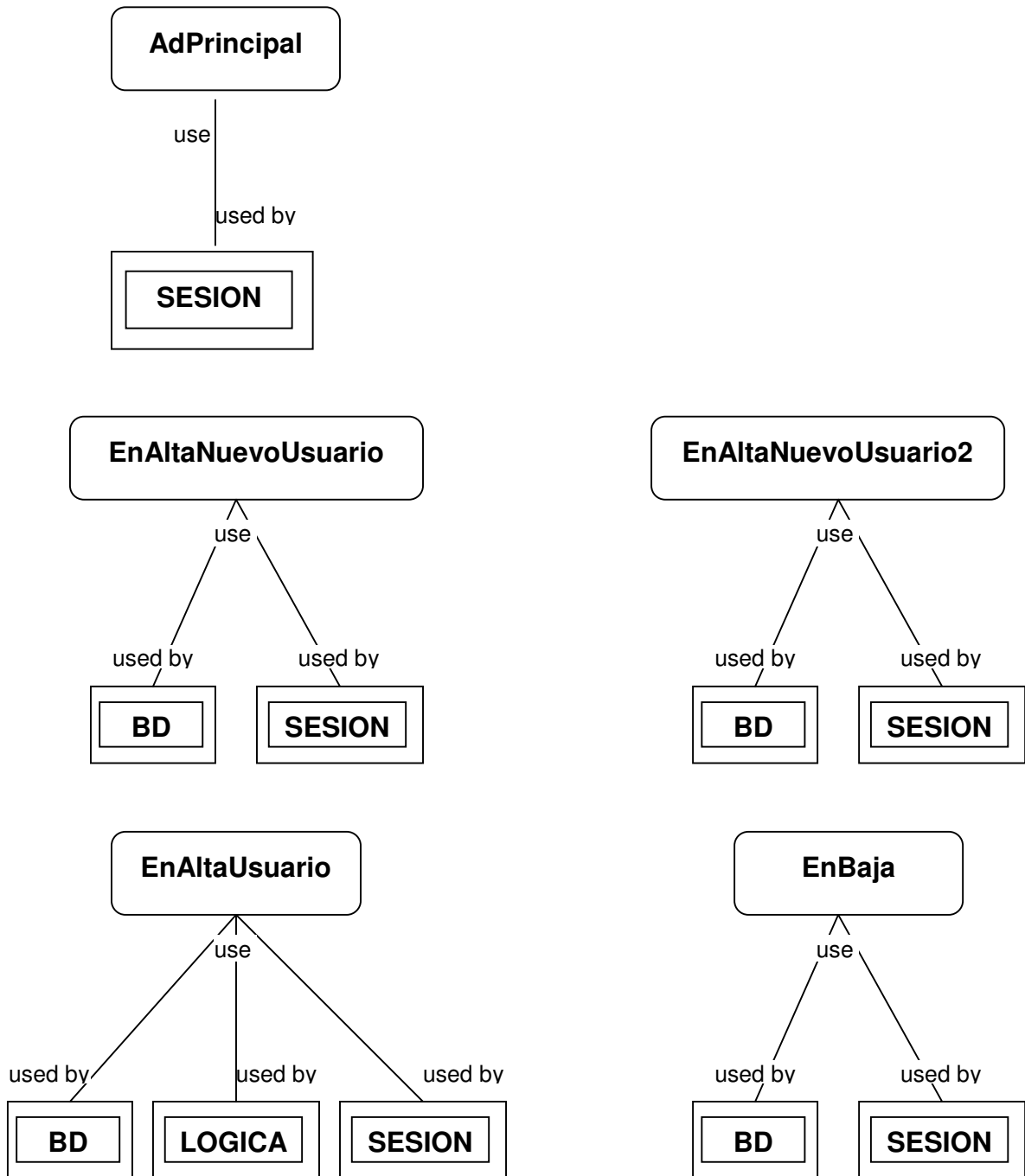
Esta información es útil a la hora de ampliar o modificar la aplicación ya que nos indica qué clases podrían necesitar modificaciones tras un cambio en una de ellas.

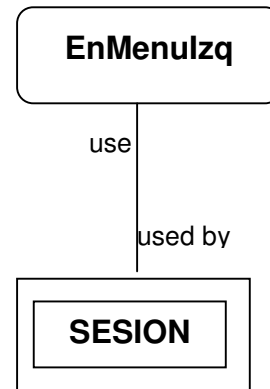
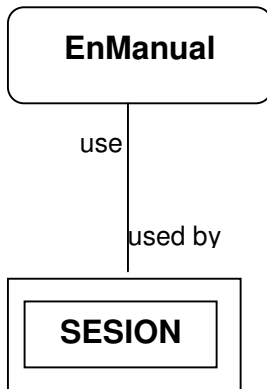
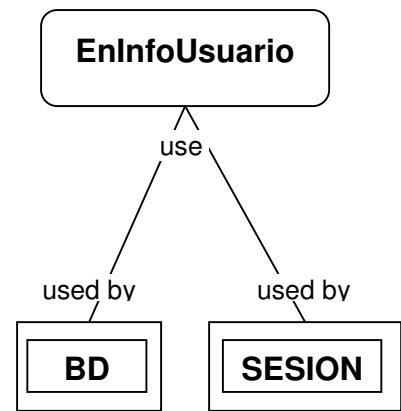
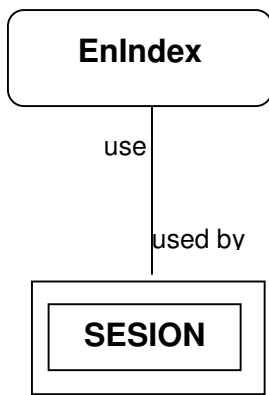
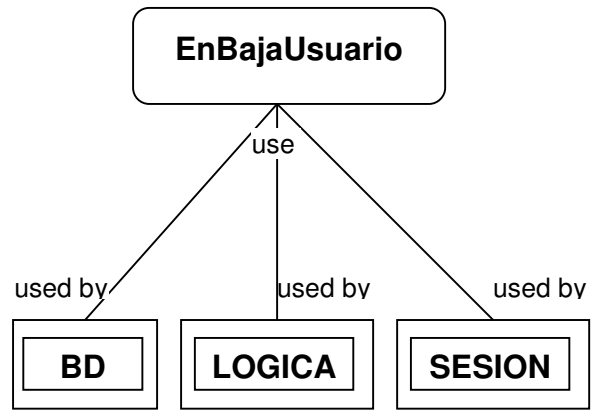
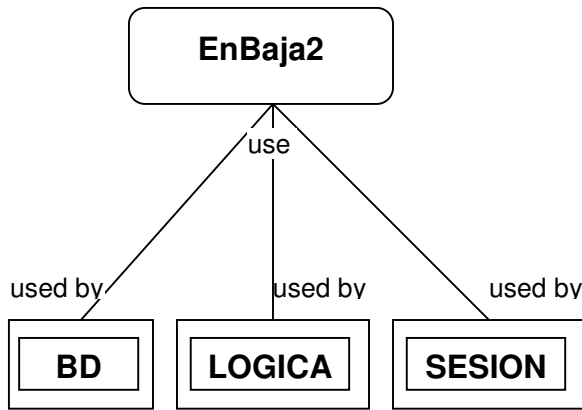


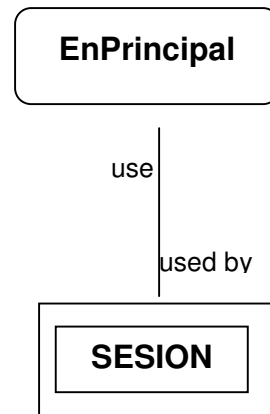
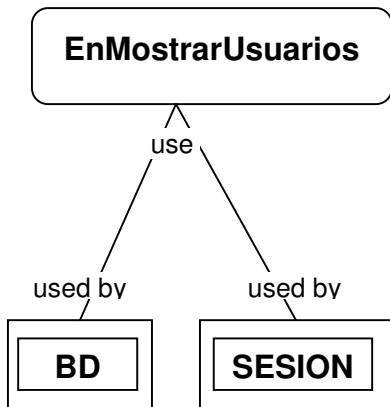
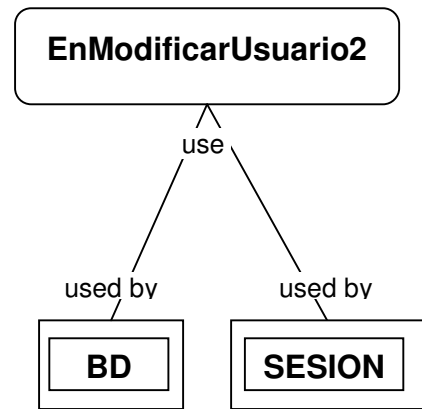
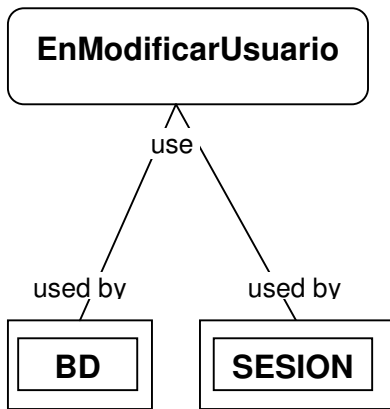
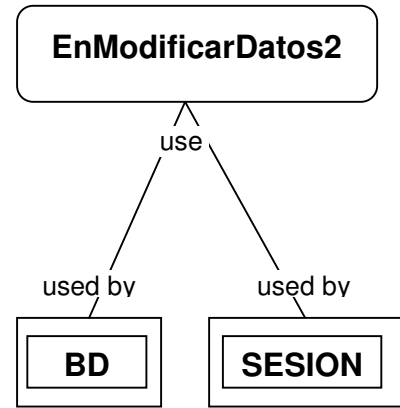
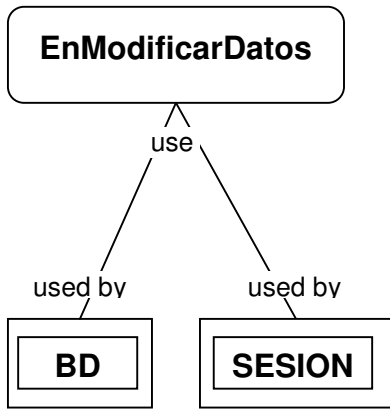


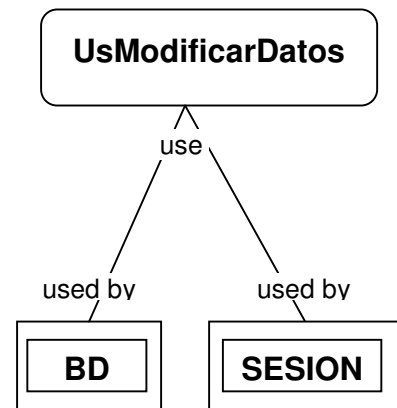
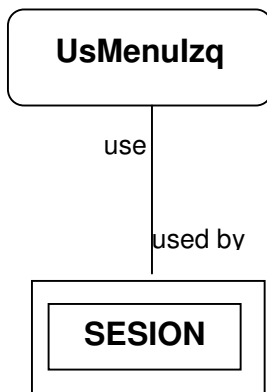
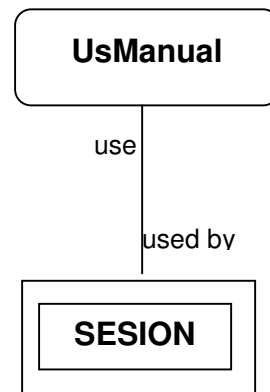
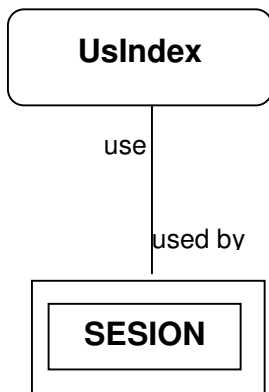
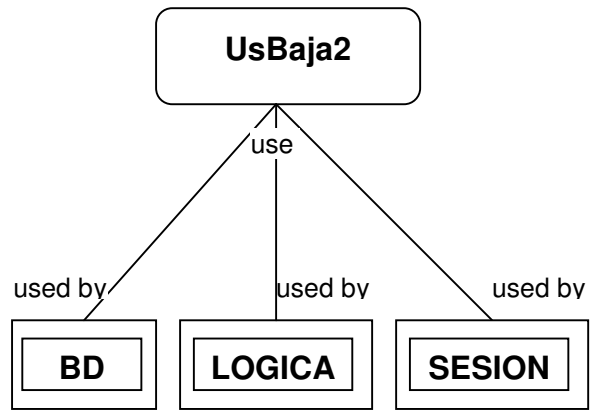
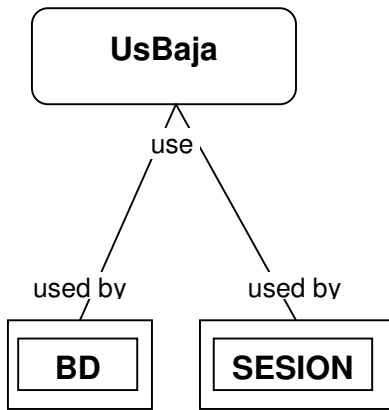


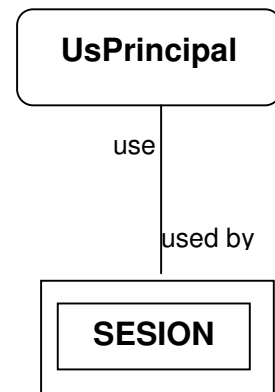
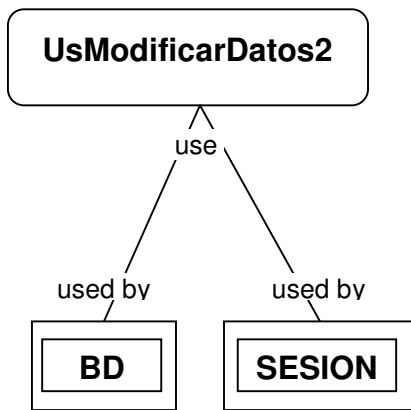












CerrarSesion

ConfirmacionPassword

ErrorBaja

ErrorEstado

ErrorFinDemo

ErrorInesperado

ErrorInicioIn

ErrorInicioLogin

ErrorInicioPass

ErrorInNoRedir

ErrorRegistro

ErrorRegLogEx

ErrorSesNoIn

RegistroCompletado

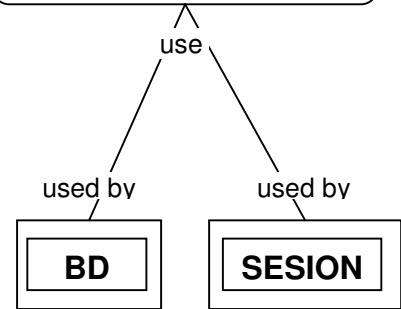
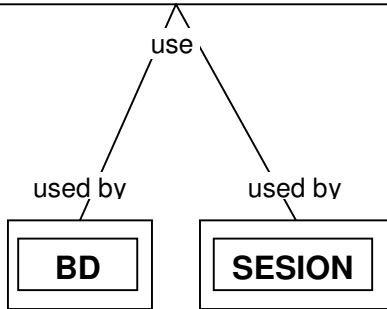


RegistroDemoCompletado

UploadFichero

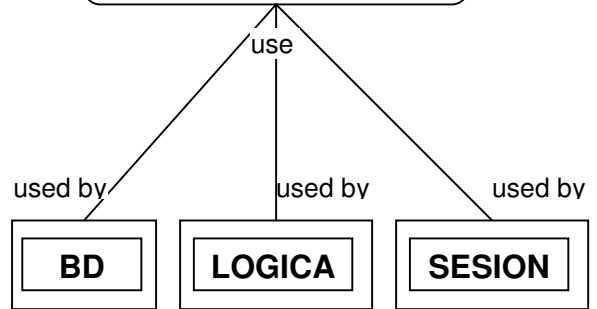
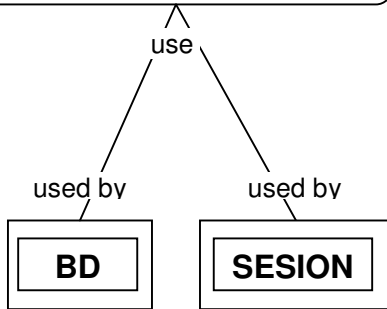
InicioSesionAdministrador

InicioSesionEntidad



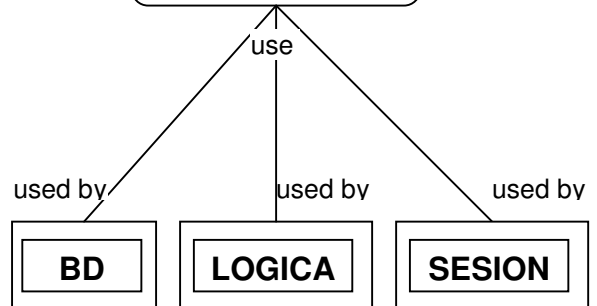
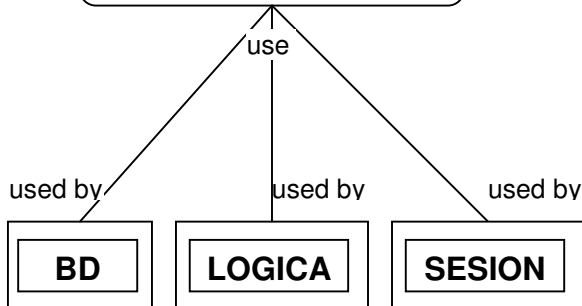
InicioSesionUsuario

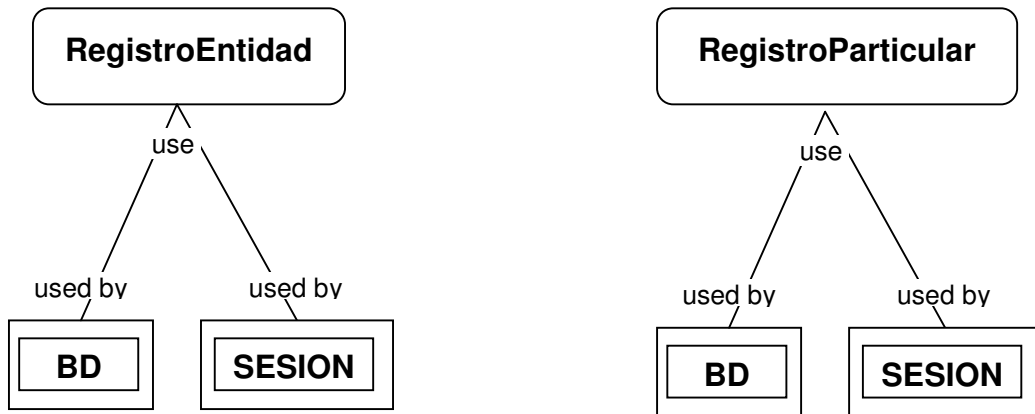
RecuperarPasswordE



RecuperarPasswordU

RegistroDemo







13.1. Clases pertenecientes al tipo de usuario Administrador

CLASES	PAQUETES		
	BD	SESION	LOGICA
AdAltaEntidad	x	x	X
AdAltaUsuario	x	x	X
AdBajaEntidad	x	x	X
AdBajaUsuario	x	x	X
AdCerrarSesion			
AdEliminaDemo	x	x	
AdEliminaEntidad	x	x	
AdEliminaUsuario	x	x	
AdIndex		x	
AdInfoEntidad	x	x	
AdInfoUsuario	x	x	
AdManual		x	
AdMenuIzq		x	
AdModificarDemo	x	x	
AdModificarDemo2	x	x	
AdModificarEntidad	x	x	
AdModificarEntidad2	x	x	
AdModificarUsuario	x	x	
AdModificarUsuario2	x	x	
AdMostrarDemo	x	x	
AdMostrarEntidades	x	x	
AdMostrarParticulares	x	x	
AdPrincipal		x	



13.2. Clases pertenecientes al tipo de usuario Entidad/Institucion

CLASES	PAQUETES		
	BD	SESION	LOGICA
EnAltaNuevoUsuario	x	x	
EnAltaNuevoUsuario2	x	x	
EnAltaUsuario	x	x	x
EnBaja	x	x	
EnBaja2	x	x	x
EnBajaUsuario	x	x	x
EnIndex		x	
EnInfoUsuario	x	x	
EnManual		x	
EnMenuIzq		x	
EnModificarDatos	x	x	
EnModificarDatos2	x	x	
EnModificarUsuario	x	x	
EnModificarUsuario2	x	x	
EnMostrarUsuarios	x	x	
EnPrincipal		x	

13.3. Clases Pertenecientes al tipo de usuario Particular

CLASES	PAQUETES		
	BD	SESION	LOGICA
UsBaja	x	x	
UsBaja2	x	x	x
UsIndex		x	
UsManual		x	
UsMenuIzq		x	
UsModificarDatos	x	x	
UsModificarDatos2	x	x	
UsPrincipal		x	



13.4. Clases pertenecientes al tipo de usuario General y clases comunes

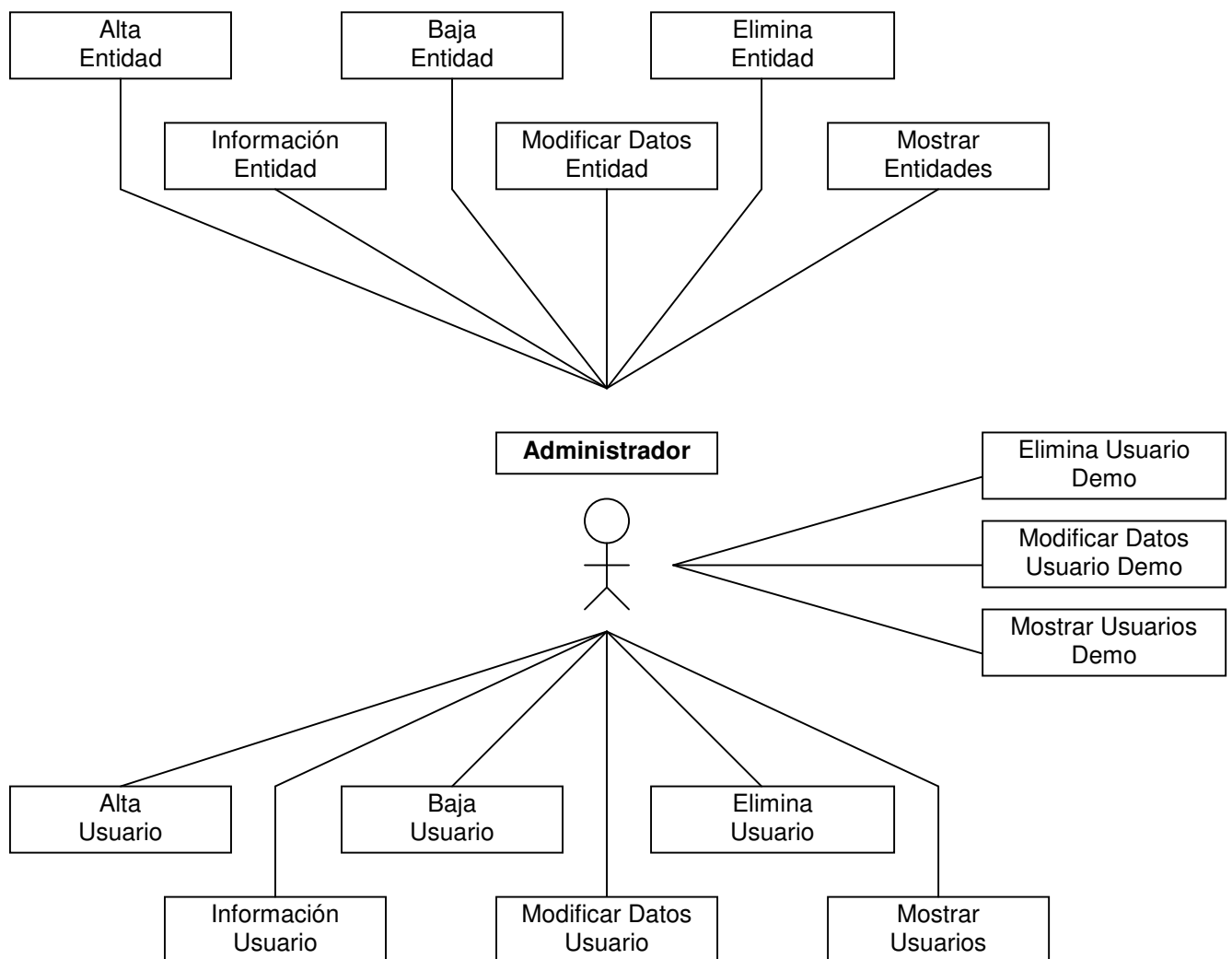
CLASES	PAQUETES		
	BD	SESION	LOGICA
CerrarSesion			
ConfirmacionPassword			
ErrorBaja			
ErrorEstado			
ErrorFinDemo			
ErrorInesperado			
ErrorInicioIn			
ErrorInicioLogin			
ErrorInicioPass			
ErrorInNoRedir			
ErrorRegistro			
ErrorRegLogEx			
ErrorSesNoIn			
InicioSesionAdministrador	x	x	
InicioSesionEntidad	x	x	
InicioSesionUsuario	x	x	
RecuperarPasswordE	x	x	x
RecuperarPasswordU	x	x	x
RegistroCompletado			
RegistroDemo	x	x	x
RegistroDemoCompletado			
RegistroEntidad	x	x	
RegistroParticular	x	x	
UploadFichero			

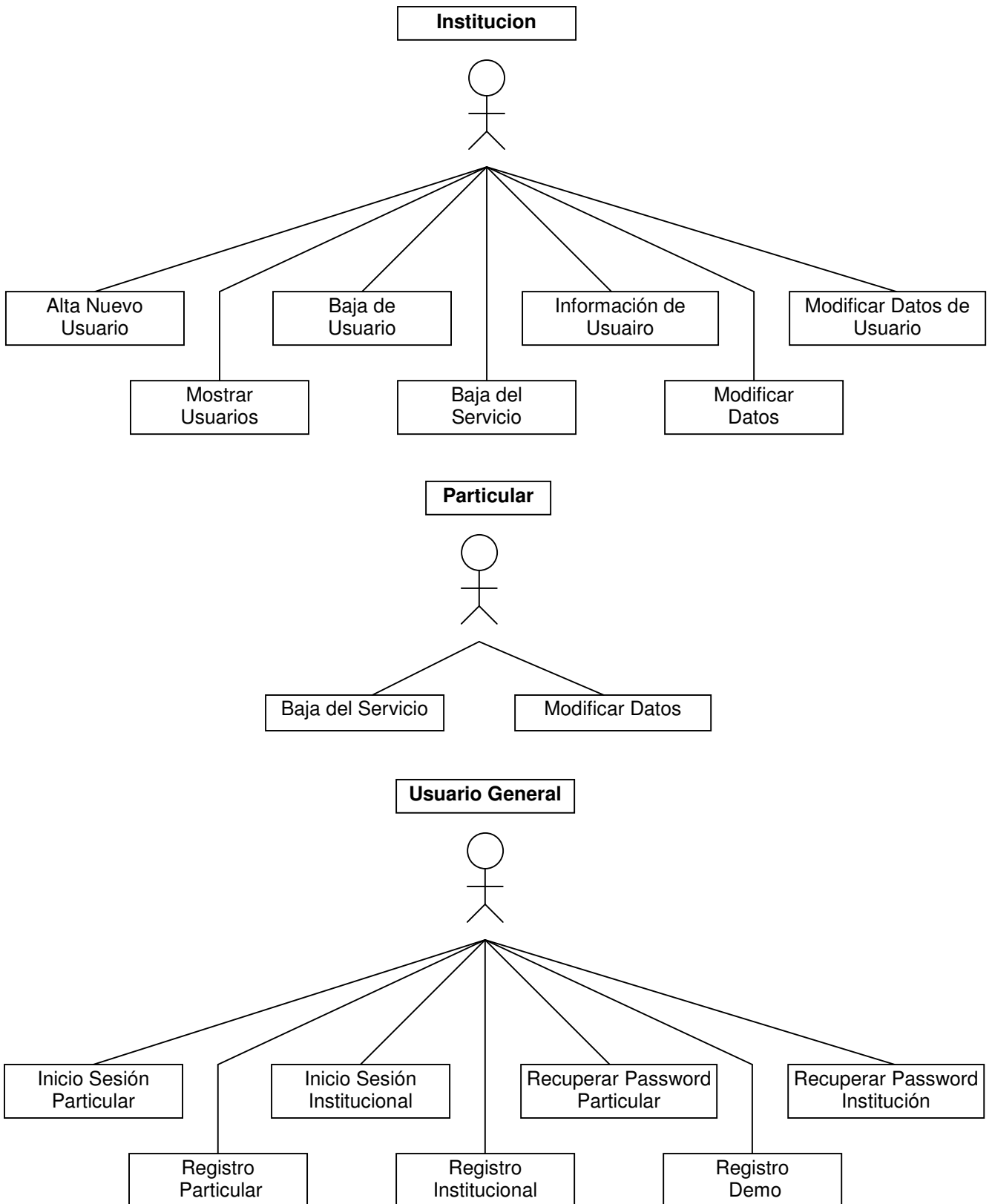


14. CASOS DE USO

Para cada tipo de usuario se indican en los diagramas expuestos a continuación, los casos de uso que pertenecen a ese tipo de usuario. Estos casos de uso hacen referencia a los tipos de servicios o funcionalidades que pueden utilizar los diferentes tipos de usuarios.

Nótese que el usuario demo poseerá los mismos casos de uso que el usuario particular, por tanto no lo hemos adjuntado.







15. INTERFAZ DE USUARIO

15.1. ¿Por qué se diseñó así, la interfaz de usuario?

La interfaz de usuario implementada finalmente, ha sido el fruto de realizar un estudio exhaustivo, de lo fácil que un sistema puede ser aprendido o usado, de la efectividad, eficiencia y satisfacción desde el punto de vista de lo que esperaría el usuario final de la aplicación. No debemos olvidar en ningún momento, que el usuario final pertenece al gremio médico, considerando por tanto que la familiaridad con el mundo informático no es elevada.

Las características de los ordenadores de dichos usuarios nos llevan a limitar los métodos a usar para visualización, es decir, por ejemplo no se deben incluir programas flash, puesto que esto provocaría una ralentización notable en el tiempo de respuesta final de la aplicación. Por tanto, a pesar de la tendencia a hacer uso de las últimas tecnologías, en este caso prima más a la hora de determinar el diseño, considerar los usuarios finales.

Por tanto considerando las características tanto de los usuarios finales como de los Pc's descritas anteriormente, la interfaz de usuario final se ha implementado con el objetivo de realzar los siguientes propósitos:

- Alto grado de **usabilidad**: con la finalidad de conseguir una aplicación fácil de entender, de aprender y de utilizar, independientemente de la experiencia del usuario, consiguiendo una interacción eficiente y efectiva. Por tanto se conseguirá que los nuevos usuarios, comprendan el modo de uso del sistema interactivo llegando a un nivel máximo de conocimiento del mismo.
- Entorno familiar: en el sentido en el que hay correlación entre el conocimiento que tiene el usuario y el conocimiento que se requiere para una interacción efectiva. Se ha creído conveniente, proporcionar al usuario además de varios puntos de acceso a los distintos apartados (a través del frame izquierdo y a través de los accesos directos ubicados en la parte inferior), el sentido de tener un punto de partida, en nuestro caso el enlace "home".

Por tanto, la interfaz, tiene un entorno explorable pero no enrevesado, no incluye múltiples opciones las cuales aumentan el grado de complejidad en el método de aprendizaje. En todo momento el usuario debe sentir que posee un pleno control sobre la aplicación, estando alertado de todo error ocasionado en la misma.

Para aquellos usuarios que sean noveles en el uso de la aplicación, se han incluido los manuales respectivos, ajustados a los distintos tipos de usuario que pueden interaccionar con la aplicación.



15.2. Apariencia de la interfaz de usuario

A continuación mostramos la apariencia e interacciones fundamentales posibles que los distintos usuarios (particular, entidad/institución y demo), dados de alta en la aplicación pueden llevar a cabo. Los usuarios deben previamente haberse registrado como tales, salvo el usuario administrador. Los datos de acceso al servicio, son enviados para todos los tipos de usuario, vía e-mail a la dirección de correo especificada en el formulario de registro.

15.2.1. Usuario administrador

Cabe destacar que la url de acceso a esta sección es privada, no se tiene acceso desde el portal principal de la aplicación, consiguiendo así un mayor grado de seguridad y la transparencia para el resto de usuarios de la existencia de un administrador. La url de acceso mencionada es: https://IP_servidor:8443/telemedicina/web/admin.html

Un usuario que ha iniciado sesión como administrador, comienza su interacción en la aplicación a través de la siguiente página, teniendo acceso a las distintas utilidades de la aplicación a través de los dos menús que tienen habilitadas todas las páginas:



Administración

Administración del Servicio.

Telemedicina Web Administrator

Desde esta página podrá usted administrar la aplicación de una manera sencilla.

Podrá usted consultar los usuarios registrados, eliminarlos, ver sus datos... En definitiva, estar en permanente conocimiento del estado de su aplicación y la posibilidad de administrarla comodamente.

Si es la primera vez que utiliza esta aplicación, le recomendamos que lea el [manual del administrador](#).

No obstante, en caso de alguna duda o sugerencia, consulte con nosotros y le atenderemos en el menor tiempo posible.

Muchas gracias.

El equipo de Telemedicina.

Menús

Home Mostrar Usuarios Demo Mostrar Entidades Mostrar Particulares Manual del Administrador

Home
Mostrar Usuarios Demo
Mostrar Entidades
Mostrar Particulares
Manual del Administrador
Cerrar Sesión

La funcionalidad de los distintos enlaces, es la siguiente:

- **Home:** redirige a la página principal.
- **Modificar Usuarios Demo:** permite modificar los datos de los usuarios que tienen acceso por un periodo de validez limitado.
- **Mostrar Entidades:** Muestra las entidades dadas de alta en el servicio, junto con la información que tienen asociada.
- **Mostrar particulares:** Muestra los usuarios particulares dados de alta en el servicio, junto con la información que tienen asociada.
- **Manual del Administrador:** redirige a la página que permite la descarga del manual de ayuda para el administrador.
- **Cerrar Sesión:** Cierra su sesión del administrador.



15.2.2. Página de visualización de los usuarios demo

Se muestran los usuarios 'demo' dados de alta en la base de datos:

Usuarios Demo

Usuarios Demo registrados en el servicio.

A continuación se muestra una lista con los usuarios DEMO dados de alta en el servicio
Si tiene alguna duda sobre el uso consulte el [Manual del Administrador](#).

Login	Password	e-mail	Fecha Alta	Fecha Baja		
izKwOb	izKwOb	alberto.garcia.botero@telemedicina.com	2005-05-22	2005-06-21	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>

Home [Mostrar Usuarios Demo](#) [Mostrar Entidades](#) [Mostrar Particulares](#) [Manual del Administrador](#)

En la lista de usuarios se muestran los datos del usuario junto con las operaciones posibles:

Login	Password	e-mail	Fecha Alta	Fecha Baja		
izKwOb	izKwOb	alberto.garcia.botero@telemedicina.com	2005-05-22	2005-06-21	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>

Mediante el botón Modificar, se podrán cambiar los datos del usuario. Para ello, una vez pulsado el botón se abrirá una ventana con un formulario mostrando los datos actuales del usuario. Una vez realizados los cambios, deberá validarlos mediante el botón enviar. Tras ello una ventana emergente le informará si el cambio se realizó con éxito o hubo algún error.

Login	Password	e-mail	Fecha Alta	Fecha Baja		
izKwOb	izKwOb	alberto.garcia.botero@telemedicina.com	2005-05-22	2005-06-21	<input type="button" value="Modificar"/>	<input type="button" value="Eliminar"/>



Modificar Datos - TELEMEDICINA - Microsoft Internet Explorer

Modificar Datos

Formulario para modificación de datos.

IMPORTANTE: Revise la dirección de e-mail, dado que es el método de comunicación con nuestros usuarios.

Datos personales:

Login: izKwOb

Password:* De 4 a 20 letras.
Se permiten símbolos

E-mail:* Hasta 80 caracteres alfanuméricos.

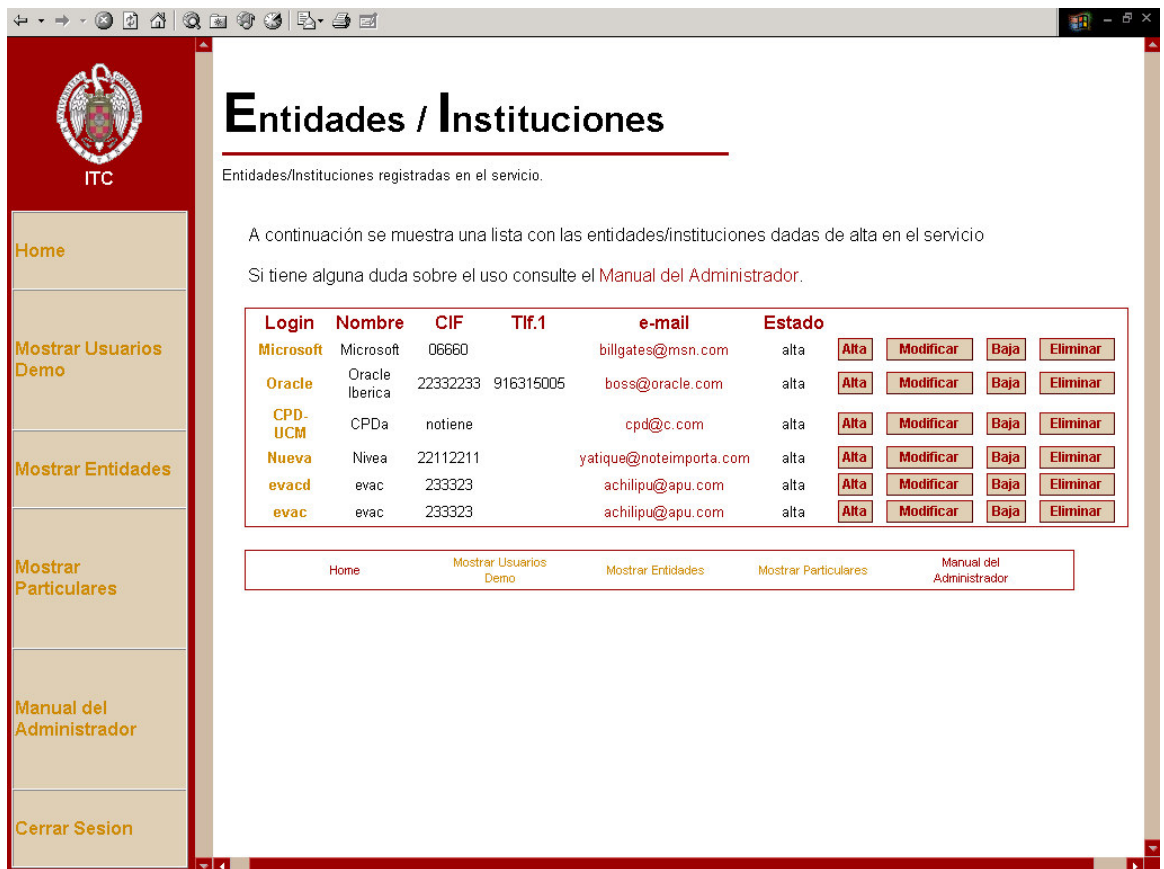
Fecha Alta:* aaaa-mm-dd.

Fecha Baja: aaaa-mm-dd.

* Campos obligatorios



15.2.3. Página de visualización de las entidades dadas de alta



Entidades / Instituciones

Entidades/Instituciones registradas en el servicio.

A continuación se muestra una lista con las entidades/instituciones dadas de alta en el servicio
Si tiene alguna duda sobre el uso consulte el [Manual del Administrador](#).

Login	Nombre	CIF	Tif.1	e-mail	Estado				
Microsoft	Microsoft	06660		billgates@msn.com	alta	Alta	Modificar	Baja	Eliminar
Oracle	Oracle Iberica	22332233	916315005	boss@oracle.com	alta	Alta	Modificar	Baja	Eliminar
CPD-UCM	CPDa	notiene		cpd@c.com	alta	Alta	Modificar	Baja	Eliminar
Nueva	Nivea	22112211		yatique@noteimporta.com	alta	Alta	Modificar	Baja	Eliminar
evacd	evac	233323		achilipu@apu.com	alta	Alta	Modificar	Baja	Eliminar
evac	evac	233323		achilipu@apu.com	alta	Alta	Modificar	Baja	Eliminar

Home Mostrar Usuarios Demo Mostrar Entidades Mostrar Particulares Manual del Administrador

En la lista de usuarios se muestran los datos personales junto con las operaciones posibles: envío de e-mail, modificación, baja/alta de entidades o eliminar una entidad.

La columna **Estado** indica el estado actual de la entidad / institución. Hay 2 posibles valores: alta y baja. Una entidad cuyo estado es alta podrá acceder normalmente al servicio con su login y password. Sin embargo, una entidad con estado baja no podrá acceder al servicio, ni tampoco ninguno de los usuarios asociados a ella.

Login	Nombre	CIF	Tif.1	e-mail	Estado				
Microsoft	Microsoft	06660		billgates@msn.com	alta	Alta	Modificar	Baja	Eliminar
Oracle	Oracle Iberica	22332233	916315005	boss@oracle.com	alta	Alta	Modificar	Baja	Eliminar
CPD-UCM	CPDa	notiene		cpd@c.com	alta	Alta	Modificar	Baja	Eliminar
evacd	Cofares	233323		achilipu@cofares.com	alta	Alta	Modificar	Baja	Eliminar



Una vez realizada la operación, se mostrará una ventana informando del resultado de la petición.



Los cambios serán visibles en el formulario si cierra esta ventana mediante el botón “Cerrar” mostrado en la ventana o cuando recargue el formulario.

Junto con una entidad / institución se almacenan la fecha de la última alta y baja (si no ha habido baja este campo aparece vacío). Esta y otra información adicional aparece pinchando en el login de la entidad que desee consultar.

Login	Nombre	CIF	Tlf.1	e-mail	Estado				
Microsoft	Microsoft	06660		billgates@msn.com	alta	Alta	Modificar	Baja	Eliminar
Oracle	Oracle Iberica	22332233	916315005	boss@oracle.com	alta	Alta	Modificar	Baja	Eliminar
CPD-UCM	CPDa	notiene		cpd@c.com	alta	Alta	Modificar	Baja	Eliminar
evacd	Cofares	233323		achilipu@cofares.com	alta	Alta	Modificar	Baja	Eliminar



Informacion Entidad - TELEMEDICINA - Microsoft Internet Explorer

Datos personales:

Login: Oracle

Nombre: Oracle Iberica

CIF: 22332233

Teléfono de contacto: 916315005

Teléfono de contacto (2):

E-mail: boss@oracle.com

Fecha de Alta: 2005-04-26

Fecha de Baja: null

Estado: alta

Descripcion: Empresa dedicada a la tecnología de las bases de datos.

Cerrar

Mediante el botón Modificar, se podrá cambiar los datos de la entidad / institución. Para ello, una vez pulsado el botón se abrirá una ventana con un formulario mostrando los datos actuales. Una vez realizados los cambios, deberá validarlos mediante el botón enviar. Tras ello una ventana emergente le informará si el cambio se realizó con éxito o hubo algún error.

Login	Nombre	CIF	Tif.1	e-mail	Estado				
Microsoft	Microsoft	06660		billgates@msn.com	alta	Alta	Modificar	Baja	Eliminar
Oracle	Oracle Iberica	22332233	916315005	boss@oracle.com	alta	Alta	Modificar	Baja	Eliminar
CPD-UCM	CPDa	notiene		cpd@c.com	alta	Alta	Modificar	Baja	Eliminar
evacd	Cofares	233323		achilipu@cofares.com	alta	Alta	Modificar	Baja	Eliminar



Modificar Datos - TELEMEDICINA - Microsoft Internet Explorer

Modificar Entidad

Formulario para modificación de datos.

IMPORTANTE: Revise la dirección de e-mail, dado que es el método de comunicación con nuestros usuarios.

Datos de la Entidad:

Login: Oracle

Password:* De 4 a 20 letras.
Se permiten símbolos

Nombre de la entidad:* De 2 a 20 letras.
Se permiten guiones

CIF:* De 4 a 20 caracteres alfanuméricos.
Sin guiones y con ceros a la izquierda.

Teléfono de contacto: De 6 a 20 números (sin guiones).

Teléfono de contacto (2): De 6 a 20 números (sin guiones).

E-mail:* Hasta 80 caracteres alfanuméricos.

Descripción de la entidad: Máximo 255 caracteres

Estado:*

Fecha Alta:* aaaa-mm-dd.

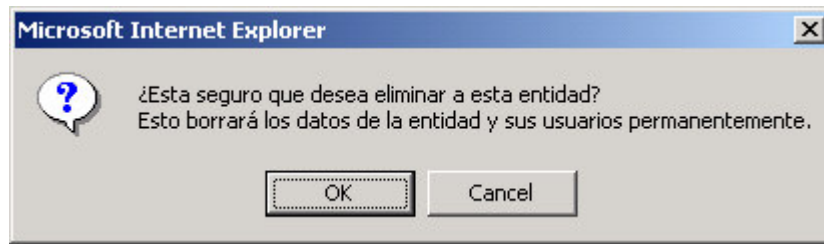
Fecha Baja: aaaa-mm-dd.

* Campos obligatorios

Mediante el botón Eliminar, se podrá dar de baja a una entidad existente.

Login	Nombre	CIF	Tlf.1	e-mail	Estado	Alta	Modificar	Baja	Eliminar
Microsoft	Microsoft	06660		billgates@msn.com	alta	<input type="button" value="Alta"/>	<input type="button" value="Modificar"/>	<input type="button" value="Baja"/>	<input type="button" value="Eliminar"/>
Oracle	Oracle Iberica	22332233	916315005	boss@oracle.com	alta	<input type="button" value="Alta"/>	<input type="button" value="Modificar"/>	<input type="button" value="Baja"/>	<input type="button" value="Eliminar"/>
CPD-UCM	CPDa	notiene		cpd@c.com	alta	<input type="button" value="Alta"/>	<input type="button" value="Modificar"/>	<input type="button" value="Baja"/>	<input type="button" value="Eliminar"/>
evacd	Cofares	233323		achilipu@cofares.com	alta	<input type="button" value="Alta"/>	<input type="button" value="Modificar"/>	<input type="button" value="Baja"/>	<input type="button" value="Eliminar"/>

En el caso en el que esa entidad tenga datos de alta usuarios, en este caso y tras confirmar la acción:



se borrarán todos los usuarios asociados a la entidad, así como la entidad en si. Tras ello una ventana emergente le informará si el cambio se realizó con éxito o hubo algún error.

15.2.4. Página de visualización de los usuarios particulares

Las acciones que se pueden llevar a cabo en esta sección son análogas a las que se pueden realizar en el caso de acceder/manipular la información referente a las entidades/instituciones, por tanto omitimos la explicación al respecto.

Usuarios Particulares

Usuarios registrados en el servicio.

A continuación se muestra una lista con los usuarios dados de alta en el servicio

Si tiene alguna duda sobre el uso consulte el [Manual del Administrador](#).

Login	Nombre	Apellidos	DNI	Tif.1	e-mail	Estado				
Aida	Aida	Coque Hernandez	012345		aida@telemedicina.com	alta	Alta	Modificar	Baja	Eliminar
Mary	Mary	Martin Pedro	543210	915553553	mary@telemedicina.com	alta	Alta	Modificar	Baja	Eliminar
Juan	Juan Antonio	Martin	11850		ja@telemedicina.es	alta	Alta	Modificar	Baja	Eliminar
Beah	Beatriz	Herrezuelo Sánchez	123456		bea@telemedicina.es	alta	Alta	Modificar	Baja	Eliminar
Rigodon	Rigodon	Fog Frog	11888	666555777	ringo@telemedicina.es	baja	Alta	Modificar	Baja	Eliminar

Home Mostrar Usuarios Demo Mostrar Entidades Mostrar Particulares Manual del Administrador

Manual del Administrador

Cerrar Sesión



15.2.5. Usuario Entidades/Instituciones

Una vez que el usuario dado de alta como entidad/institución ha iniciado sesión, la pantalla a la que es redirigido es la siguiente:



En ella, podrá comprobar que existen dos menús que dan acceso a todas las funcionalidades del portal, y que estarán disponibles en todas las páginas del portal. La forma de seguimiento de las interacciones hechas durante la sesión abierta, están identificadas, de forma que el color de los enlaces visitados es beige y los enlaces sin visitar mantienen el color granate original.

La funcionalidad de los distintos enlaces, es la siguiente:

- **Home:** redirige a la página principal
- **Modificar Datos:** permite modificar los datos de su entidad / institución (se mostrarán los campos del formulario rellenos con los datos introducidos en el proceso de registro).



• **Mostrar Usuarios:** muestra una lista con todos los usuarios asociados a la entidad / institución.

• **Alta de Usuario:** permite dar de alta un nuevo usuario asociado a su entidad / institución, introduciendo, para ello, los datos personales del nuevo usuario.

• **Manual de Uso:** redirige a la página que permite la descarga del manual de ayuda, en formato Word o en formato Pdf, de los usuarios dados de alta como entidad/institución.

• **Baja del Servicio:** permite darse de baja en el servicio de telemedicina.

• **Contactos:** muestra una lista con los contactos a los que puede dirigirse en caso de problemas o incidencias relacionadas con nuestro servicio.

• **Cerrar Sesión:** Cierra su sesión como entidad / institución y le redirige a la página principal del portal.

15.2.6. Pantalla de modificación de datos

En esta página se ofrece un formulario mediante el cual, se pueden modificar los datos de la entidad / institución. Al acceder a la página, el formulario mostrará los datos actuales asociados a la entidad / institución. Para realizar cambios, se selecciona el dato a modificar en el cuadro de texto y se introduce el nuevo valor. Una vez realizadas las modificaciones, se validarán haciendo click en el botón enviar.



Modificar Datos

Formulario para modificación de datos.

IMPORTANTE: Revise la dirección de e-mail, dado que es el método de comunicación con nuestros usuarios.

Datos de la Entidad:

Nombre de la entidad:* De 2 a 20 letras. Se permiten guiones

CIF:* De 4 a 20 caracteres alfanuméricos. Sin guiones y con ceros a la izquierda.

Teléfono de contacto: De 6 a 20 números (sin guiones).

Teléfono de contacto (2): De 6 a 20 números (sin guiones).

E-mail:* Hasta 80 caracteres alfanuméricos.

Descripción de la entidad:* Máximo 255 caracteres

* Campos obligatorios

Home Modificar Datos Mostrar Usuarios Alta de Usuario Manual de Uso Contactos Baja del Servicio

15.2.7. Pantalla de visualización de usuarios dados de alta

La columna **Estado** indica el estado actual del usuario. Si un usuario tiene un estado que sea alta, podrá acceder normalmente al servicio con su login y password. Sin embargo, un usuario con estado baja, no podrá acceder al servicio y no se contabilizará como usuario asociado a su entidad / institución. Los usuarios dados de baja, no se pueden eliminar; nos servirán para obtener estadísticas acerca del uso de la aplicación.



Listado de Usuarios

Usuarios dados de alta en la entidad.

A continuación se muestra una lista con los usuarios dados de alta en esta entidad. Desde aquí podrá modificar, dar de baja, o volver a dar de alta a un usuario.

Si tiene alguna duda sobre el funcionamiento consulte el [Manual de uso](#).

Login	Nombre	Apellidos	DNI	Tlf.1	e-mail	Estado			
Aida	Aida	Coque Hernandez	012345		aida@telemedicina.com	alta	Alta	Modificar	Baja
Mary	Mary	Martin Pedro	543210	915553553	mary@telemedicina.com	alta	Alta	Modificar	Baja
Juan	Juan	Antonio Martin	11850		ja@telemedicina.es	alta	Alta	Modificar	Baja
BeaH	Beatriz	Herrezuelo Sánchez	123456		bea@telemedicina.es	alta	Alta	Modificar	Baja
Rigodon	Rigodon	Fog Frog	11888	666555777	ringo@telemedicina.es	baja	Alta	Modificar	Baja

Home [Modificar Datos](#) [Mostrar Usuarios](#) [Alta de Usuario](#) [Manual de Uso](#) [Contactos](#) [Baja del Servicio](#)

El botón Modificar, permite modificar los datos asociados a un usuario. La página a la cual redirigirá al usuario es la mostrada en el apartado de Modificación de datos.

Junto con la información asociada a un usuario, se almacenan también, la fecha de la última alta y baja (si no ha habido baja, el campo asociado, aparecerá vacío), así como el nombre del responsable de dicha alta (baja), esto le permite consultar quién y cuando ha gestionado un alta o baja. Esta información junto con otra adicional, se puede consultar, pinchando en el login del usuario a consultar. A continuación se muestra un ejemplo:



Informacion Usuario - TELEMEDICINA - Microsoft Internet Explorer

Datos personales:	
Login:	Rigodon
Nombre:	Rigodon
Primer apellido:	Fog
Segundo apellido:	Frog
DNI:	11888
Teléfono de contacto:	666555777
Teléfono de contacto (2):	987654321
E-mail:	ringo@telemedicina.es
Estado:	baja
Fecha de Alta:	2005-04-26
Responsable Alta:	Boss
Fecha de Baja:	2005-05-13
Responsable Baja:	el boss

Cerrar

15.2.8. Pantalla de alta de Usuario

Página que permite dar de alta a nuevos usuarios asociados a su entidad / institución. Para ello se deberá rellenar el formulario con los datos del usuario y validarlos mediante el botón de enviar. Tras enviar los datos se mostrará una ventana emergente indicando si se realizó la operación con éxito o informando del error.





Registro de Usuario

Formulario de registro para nuevos usuarios de la institución.

Va a realizar el registro de un nuevo usuario asociado a esta entidad. Se enviará un mail al usuario para informarle del alta. Los campos marcados con * son obligatorios.

IMPORTANTE: Revise la dirección de e-mail, dado que la confirmación de acceso se enviará a dicha dirección

Datos personales:

Nombre de usuario (login):*	<input type="text"/>	De 4 a 20 caracteres alfanuméricos. Se permite - y _
Contraseña:*	<input type="password"/>	De 4 a 20 caracteres. Se permiten símbolos
Confirmar contraseña:*	<input type="password"/>	
Nombre:*	<input type="text"/>	De 2 a 20 letras. Se permiten guiones y espacios blancos
Primer apellido:*	<input type="text"/>	De 2 a 20 letras. Se permiten guiones y espacios blancos
Segundo apellido:	<input type="text"/>	De 2 a 20 letras. Se permiten guiones y espacios blancos
DNI:*	<input type="text"/>	De 4 a 20 caracteres alfanuméricos. Sin guiones y con ceros a la izquierda.
Teléfono de contacto:	<input type="text"/>	De 6 a 20 números (sin guiones).
Teléfono de contacto (2):	<input type="text"/>	De 6 a 20 números (sin guiones).
E-mail:*	<input type="text"/>	Hasta 80 caracteres alfanuméricos.
Responsable:*	<input type="text"/>	Nombre del responsable del alta. De 2 a 20 letras.

* Campos obligatorios

15.2.9. Baja del Servicio

Esta página permite darse de baja en el servicio. En ella también se le informa de las consecuencias de realizar la baja. Para confirmar la baja deberá pulsar el botón Baja del Servicio.



Baja del Servicio

Si realiza la baja en nuestro servicio no podrá volver a iniciar sesión.

Además, la baja de la entidad conllevará automáticamente a la baja de todos los usuarios de su entidad, los cuales tampoco podrán volver a iniciar sesión.

Una vez cursada su petición se enviará un mail de información a su dirección de correo, así como a todos los usuarios que pertenecieran a su entidad / institución.

No obstante, hemos de informarle que por posteriores labores de estudio y mantenimiento, la baja en el servicio no implica la eliminación inmediata de sus datos de nuestras bases de datos, por lo que, en caso de error, podría recuperar toda su funcionalidad poniéndose en contacto con nosotros (no se garantiza la recuperación de sus datos).

Baja del Servicio

Home Modificar Datos Mostrar Usuarios Alta de Usuario Manual de Uso Contactos Baja del Servicio

15.2.10. Usuarios particulares

Un usuario particular que inicie sesión en el servicio de fusión de imágenes 3D, es redirigido inicialmente a la siguiente pantalla.



Particular

Página Principal del Servicio.

Desde esta página podrá usted utilizar la aplicación de una manera sencilla.

Podrá usted modificar sus datos personales, así como utilizar las herramientas que le proporciona nuestros servicios.

Si es la primera vez que utiliza esta aplicación, le recomendamos que lea el [manual de uso](#).

No obstante, en caso de alguna duda o sugerencia, consulte con nosotros y le atenderemos en el menor tiempo posible.

Muchas gracias.

El equipo de Telemedicina.

Home Modificar Datos Fusión Manual de Uso Baja del Servicio Contactos

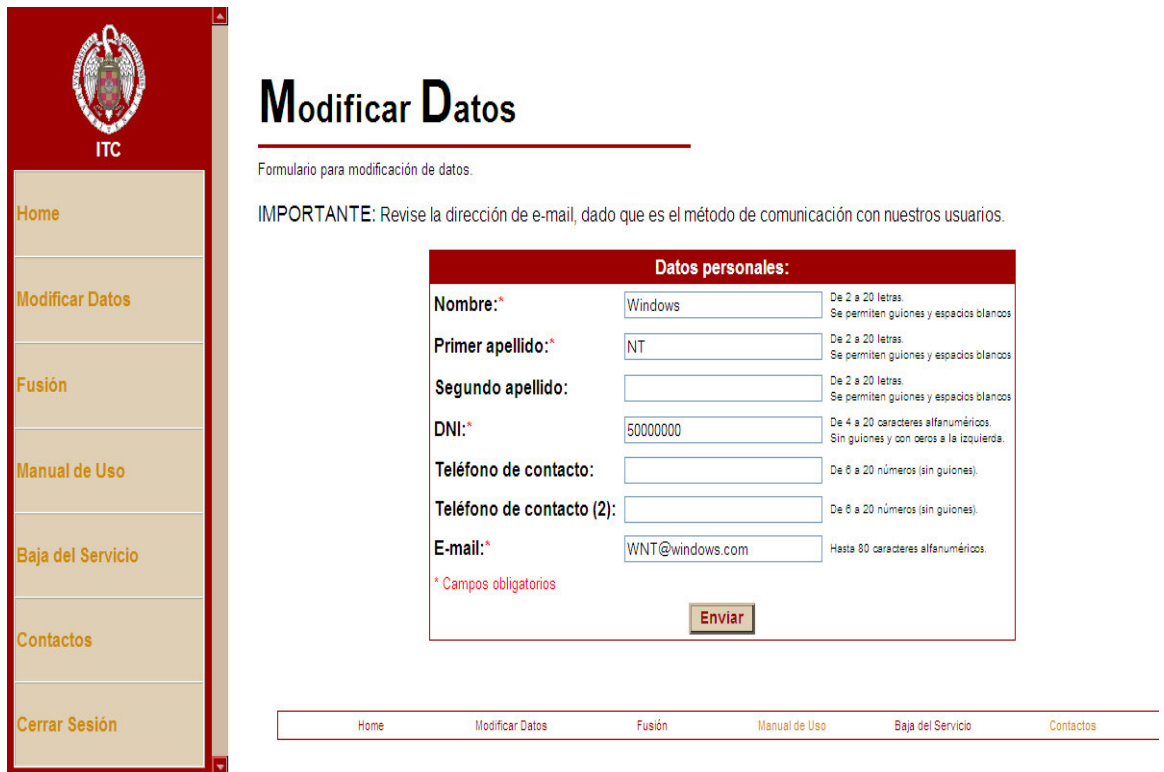
La funcionalidad de los distintos enlaces, es la siguiente:

- **Home:** redirige a la página principal
- **Modificar Datos:** Le permite modificar los datos de su usuario (se mostrarán los campos rellenos con los datos introducidos en el proceso de registro).
- **Fusión:** le permitirá acceder al servicio de fusión de imágenes 3D.
- **Manual de Uso:** redirige a la página que permite la descarga del manual de ayuda, en formato Word o en formato Pdf, de los usuarios dados de alta como entidad/institución.
- **Baja del Servicio:** permite darse de baja en el servicio de telemedicina.
- **Contactos:** muestra una lista con los contactos a los que puede dirigirse en caso de problemas o incidencias relacionadas con nuestro servicio.



• **Cerrar Sesión:** Cierra su sesión privada de usuario particular, redirigiéndole a la página principal del portal.

15.2.11. Pantalla de modificación de datos



Modificar Datos

Formulario para modificación de datos.

IMPORTANTE: Revise la dirección de e-mail, dado que es el método de comunicación con nuestros usuarios.

Datos personales:

Nombre:* De 2 a 20 letras.
Se permiten guiones y espacios blancos

Primer apellido:* De 2 a 20 letras.
Se permiten guiones y espacios blancos

Segundo apellido: De 2 a 20 letras.
Se permiten guiones y espacios blancos

DNI:* De 4 a 20 caracteres alfanuméricos.
Sin guiones y con ceros a la izquierda.

Teléfono de contacto: De 6 a 20 números (sin guiones).

Teléfono de contacto (2): De 6 a 20 números (sin guiones).

E-mail:* Hasta 80 caracteres alfanuméricos.

* Campos obligatorios

Home Modificar Datos Fusión Manual de Uso Baja del Servicio Contactos

15.2.12. Baja del Servicio

Esta página permite darse de baja en el servicio. En ella también se le informa de las consecuencias de realizar la baja. Para confirmar la baja deberá pulsar el botón Baja del Servicio.



Baja del Servicio

Si realiza la baja en nuestro servicio no podrá volver a iniciar sesión.

Una vez cursada su petición se enviará un mail de información a su dirección de correo.

No obstante, hemos de informarle que por posteriores labores de estudio y mantenimiento, la baja en el servicio no implica la eliminación inmediata de sus datos de nuestras bases de datos, por lo que, en caso de error, podría recuperar toda su funcionalidad poniéndose en contacto con nosotros (no se garantiza la recuperación de sus datos).

Baja del Servicio

[Home](#)

[Modificar Datos](#)

[Fusión](#)

[Manual de Uso](#)

[Baja del Servicio](#)

[Contactos](#)

15.2.13. Usuarios demo

Se comporta a todos los efectos, como un usuario particular, con la particularidad de tener acceso al servicio por un tiempo limitado, fijado en 30 días. Al cabo de este período, si el usuario desea poder continuar utilizando el servicio, se tendrá que registrar como usuario particular o como entidad/institución.



16. TRABAJO ADICIONAL INVESTIGADO

La base de este proyecto consiste en crear una interface de usuario, para reducir la complejidad de poder llevar a cabo un tratamiento de imágenes médicas en 3D, haciendo operaciones básicas sobre las informaciones volumétricas: rotación, ajustes, cortes transversales, etc.

El software sobre el que se apoya el servidor de fusión de volúmenes, es el AIR versión 3.08, está instalado en el supercomputador del Centro de Cálculo. El código fuente que lo compone, es libre y está implementado en lenguaje C. Este software puede compilarse bajo plataformas UNIX, PC o Macintosh. La funcionalidad que incorpora, consiste en permitir la puesta en correspondencia automatizada de imágenes, basándose en el registro de volúmenes así como de funciones matemáticas adecuadas a cada aplicación: modelos rígidos o flexibles. Estas herramientas de tratamientos de imágenes proporcionadas por el paquete AIR, no están disponibles bajo un interface de usuario. Por lo tanto, se integró en el servidor WEB del Instituto de tecnología del conocimiento (ITC), la lista de las cadenas de tratamientos de imágenes 3D, para tenerlas accesibles fácilmente por el usuario final, perteneciente al gremio médico, en el que los conocimientos informáticos generalmente, se consideran a nivel de usuario.

El módulo desarrollado, está compuesto de clases implementadas en lenguaje C, intercomunicadas con paginas web, implementadas con CGI's, las cuales se encargan de lanzar llamadas a los módulos del AIR.

Buscando una eficiencia máxima, en la ejecución de las operaciones/cálculos, los procesos se ejecutan de forma local en el servidor y paralelamente son enviados al supercomputador ubicado en el Centro de Procesamiento de Datos de la Universidad Complutense de Madrid.

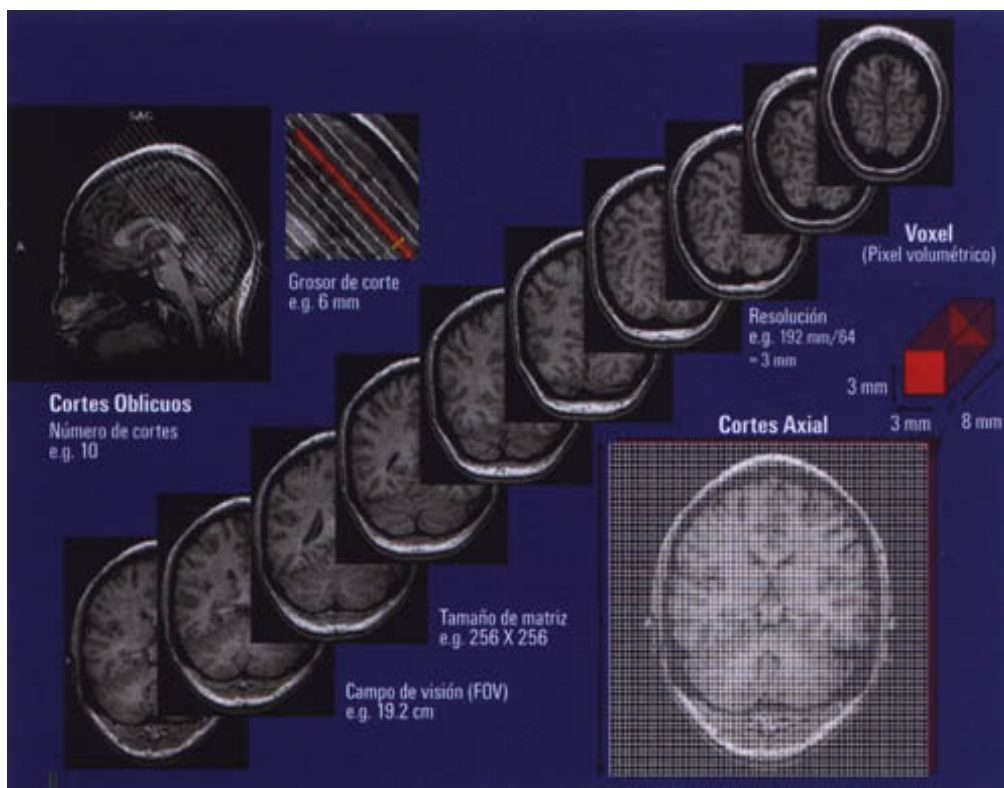
16.1. ¿Cómo funciona la fusión 3D?

El estudio obtenido tras realizar una resonancia magnética (RM), con total seguridad no coincidirá con la orientación y postura con la que se hizo la tomografía por emisión de positrones (PET). Considerando adicionalmente que las imágenes de PET tienen un tamaño de 64x64 o de 128x128, y tras el proceso de slicing, tras el cual se obtienen cortes sucesivos, se obtienen cortes de aproximadamente 3 o 4 mm. Sin embargo, los resultados de RM tienen habitualmente un tamaño de 256x256, obteniéndose por tanto más cortes pero de 1 o 2 mm de grosor que en el caso del PET.

Por lo tanto es necesario ajustar punto a punto lo más posible, ambas imágenes, para poder obtener resultados fiables. Este tuning de imágenes, conlleva el ajuste de parámetros mediante una sólida base de cálculos matemáticos.

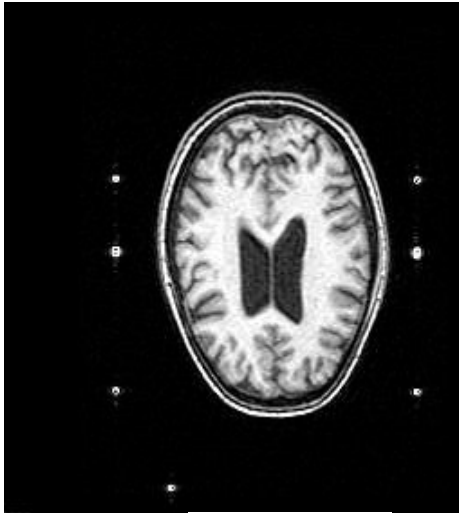


La preparación de las imágenes, se lleva a cabo en el servidor de Somosaguas, el cual prepara los datos que se enviarán al supercomputador. A grandes rasgos, el proceso es el siguiente: a partir de la imagen 3D del PET y de la RM, las cuales tienen tamaños distintos, el primer paso es igualar el tamaño para pasar después a realizar rotaciones de las imágenes para hacer coincidentes todos los puntos de ambas pruebas (es decir, así se conseguirá un resultado análogo al que se obtendría si ambas pruebas se hubiesen hecho simultáneamente, es decir mediante máquinas bimodales). Finalmente, ambos volúmenes son acoplados, realizando sobre ellos nuevos cortes, para obtener los resultados superpuestos y así con ellos identificar puntos estratégicos basándose en la anatomía y en el consumo metabólico plasmado en el volumen resultante.

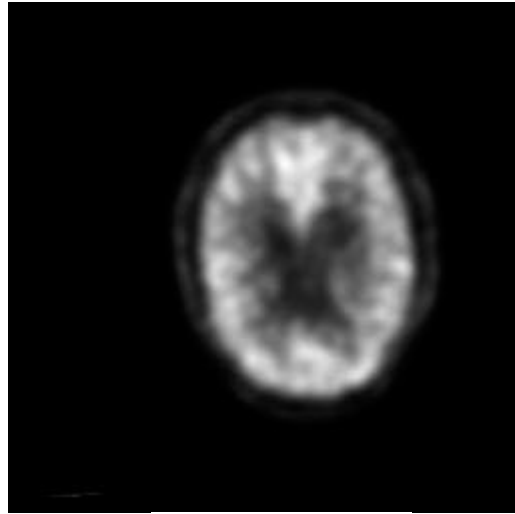




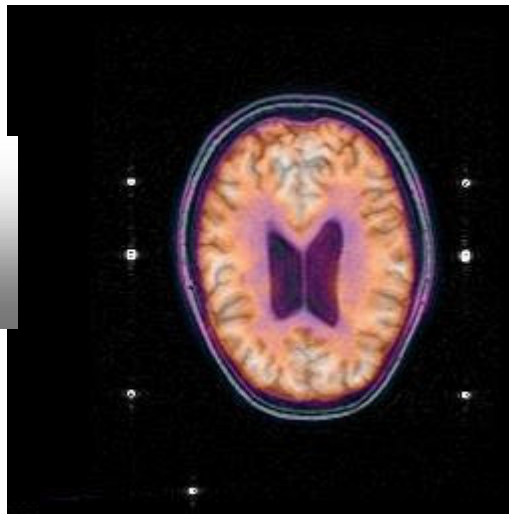
RM



PET



**RESULTADO
DE LA FUSION**



Para un correcto funcionamiento de la aplicación, se requiere un tratamiento previo de las imágenes en formato **DICOM**, es decir, es necesario transformar los volúmenes recibidos por los especialistas, a formatos admitidos por el AIR. En el caso de la fusión de RM y PET, tras el tuning de los parámetros del AIR, el cual se puede consultar en el Apéndice de este documento, es preciso realizar secuencialmente los siguientes pasos:

En el caso de las imágenes *RM*:

- `mogrify -format pnm *.npl`



Convierte las imágenes del formato npl al formato pnm.

- renombrar MR00000 PNM IMG 001 248

Genera las cabeceras AIR para los 248 cortes del volumen de RM:

- mheader MR00000 1 248 0 256 1 1.132812 1.132812 0.8 (valores que coinciden con la mitad del valor especificado en la cabecera del formato DICOM, que se puede ver con el comando 'xmedcon')

- reunitevol prueba MR00000 9 128 (así se crea prueba.hdr, que corresponde con la cabecera, y el fichero prueba.img que contiene la información de la imagen). Este ejecutable, se encarga de combinar una serie de imágenes para obtener único volumen, ya en formato AIR.

En el caso de imágenes de *PET*:

Para ver a que prueba de RM está asociado cada PET, es necesario ver la imagen ejecutando 'adactofni'.

- from3d -input nombre_fichero_PET -prefix PFA
- cambia_PET PFA 126
- renombra PFA pgm img 1 126

Genera las cabeceras AIR para los 126 cortes del volumen de PET:

- mheader PFA 1 126 0 128 128 1 2.000000 2.000000 2.016000 (parámetros que corresponden con lo que pone en la cabecera DICOM)

- reunitevol PFA_pet PFA 1 126 (así se crea PFA_pet.hdr, que corresponde con la cabecera, y el fichero PFA_pet.img que contiene la información de la imagen). Este ejecutable, se encarga de combinar una serie de imágenes para obtener único volumen ya en formato AIR.

Se deben rotar el PET al no estar alineado de la misma forma el PET y la RM:

- mri3dx nombre_mri.img
- mri3dx nombre_pet.img
- reorient PFA_pet PFA_pet_I z

Llamada a la función implementada en el AIR para poder obtener en PFA_pet_I, la imagen espejo de PFA_pet respecto del eje z.



Para ejecutar la fusión de ambas imágenes, tras el proceso de preparación anterior:

Para ajustar el volumen, es decir para preparar la puesta en correspondencia para la interpolación de los voxels de la RM (PFA_mri) con el de PET (PFA_pet_I), se hace uso de la función alignlinear del AIR, obteniendo el resultado en PFA.air:

- alignlinear PFA_mri PFA_pet_I PFA.air -t1 10 -p1 256 -p2 0 -m 6 -v -x 1 donde:
 - PFA_mri: es la imagen que se va a mantener fija, será el patrón con el que se encajará los cortes realizados sobre PFA_pet_I
 - PFA_pet_I: es el volumen sobre el que se ejecutarán una serie de cortes sucesivos con la finalidad de que coincida con el volumen de RM.
 - -t1 10: indica que el valor mínimo de voxel para PFA_mri es de 10, para que sea considerado en el análisis al obtener el valor de la función de coste.
 - -p1 256: indica el número de particiones utilizadas del volumen PFA_mri al computar la desviación estándar del rango de la imagen en dirección no invertida. El valor de 256, se usa normalmente para volúmenes de RM, que se utilizan como patrón de ajuste para registros de modalidad intermedia.
 - -p2 0: indica el número de particiones utilizadas del volumen PFA_mri al computar la desviación estándar del rango de la imagen en dirección invertida. El valor de 0, se usa normalmente cuando el volumen sobre el que se realizarán los cortes, es fruto de un estudio de PET.
 - -m 6: corresponde con el modelo 3D, para cuerpos rígidos.
 - -v: modo en el que imprime por pantalla, los pasos que se están siguiendo, es decir, en modo verbose.
 - -x 1: función de coste que es asocia, al especificar el 1, al rango de la desviación estándar de la imagen.

- reslice PFA.air PFA_pet_I_R -o -k

Toma como entrada la imagen PFA_air, (el cual contiene los volúmenes de PET y de RM acoplados) para utilizar la información que contiene y poder determinar los puntos por donde se realizarán los cortes. El resultado de ejecutar este comando se obtiene en el fichero PFA_pet_I_R. La opción -o es para permitir que se puedan sobrescribir los permisos. La opción -k se utiliza para deshabilitar la interpolación entre volúmenes cúbicos, es decir, se mantendrá las mismas dimensiones que las de los ficheros estándar.



- `separate PFA_pet_I_R PFA_pet_I_R_S y`

Separamos el volumen PET registrado en los nuevos cortes que lo componen, el resultado se obtiene en PFA_pet_I_R_S.

- `totrawtopnm2 PFA_pte_I_R_S 1 120 256 256 1 8`
- `totrawtopnm2 MR00000 129 248 256 256 1 8`

Se transforman los cortes en formato AIR de los volúmenes de PET y de RM al formato PGM.

Fusionamos los cortes obtenidos aplicando la paleta de colores a la imagen resultante:

- `totfusionIHS ./ MR00000 PFA-pet_I_R_S 0 255 50 120 1`

Los cortes fusionados, se pueden visualizar, ejecutando el comando 'gqview'.

Para transformar los cortes, en volúmenes, debemos cambiar el formato para que pase a ser *.PGM (en este caso el color se pierde).

- `mogrify -format pgm FUS*.ppm`
Convierte las imágenes del formato ppm al formato pgm.

- `renombra FUS pgm img 001 120`
- `mheader FUS 1 120 0 256 256 1 1.132812 1.132812 0.8`
- `reunitevol FUSION3D SUS 1 120`

Volumen resultante:

- `mri3dx FUSION3D.img`

A modo de ejemplo, siguiendo la traza del programa ubicado en el servidor, encargado de lanzar las llamadas al supercomputador, los pasos básicos a realizar son:

16.2. Transformación de los volúmenes a formato AIR

Separamos el volumen de RM (V1) con 74 cortes de 256x256, 2 bytes x píxel:

```
expandir V1 74 256 256 2 1
```

Generación de las cabeceras AIR para los 74 cortes del volumen de RM:

```
mheader V1 1 74 0 256 256 1 1.000000 1.000000 1.000000
```



Generación del nuevo volumen de RM en formato AIR:

```
reunitevol VV1 V1 74
```

Separamos el volumen de PET (V2) con 64 cortes de 64x64, 1 byte x píxel:

```
expandir V2 64 64 64 1 1
```

Generación de las cabeceras AIR para los 64 cortes del volumen PET:

```
mheader V2 1 64 0 64 64 1 2.000000 2.000000 2.000000
```

Generación del nuevo volumen de PET en formato AIR:

```
reunitevol VV2 V2 64
```

16.3. Ajuste de volúmenes

Ajuste de los volúmenes en formato AIR:

```
alignlinear VV1 VV2 VV2.air -m 7 -z -q
```

RM: Generación de un nuevo volumen de PET ya interpolado con el volumen

```
reslice VV2.airVV1 RVV2 -o -k
```

Separación del volumen de PET registrado en diversos cortes:

```
separate RVV2 SV2 y
```

PGM: Transformamos los cortes en formato AIR del volumen de RM a formato

```
totrawtopnm2 V1 1 74 256 256 1 8
```

Transformamos los cortes en formato AIR del volumen de PET a formato PGM:

```
totrawtopnm2 SV2 1 74 256 256 1 8
```



Fusión de ambos volúmenes:

totfusionIHS path_volumenes _a_fusionar 0 255 50 74 1

16.4. Explicación de los módulos que componen el código implementado:

Pasamos a desglosar a continuación los módulos básicos implementados hasta ahora en lenguaje C.

Fichero `rutas.h`:

Encontramos en él, las rutas de los path que son necesario crear en el servidor, en los cuales se encontrarán los programas de conversión de formatos, los ficheros ejecutables de las CGI's, la ubicación donde el usuario almacenará su información entre otros.

Incluye también path referentes a rutas creadas en el supercomputador así como alias de uso de comandos remotos (ejecutados en el servidor pero que interaccionan con el supercomputador). Una constante importante es la llamada `remsc`, que corresponde con el nombre del supercomputador. En este fichero está especificado el nombre del supercomputador antiguo "`seymour.csc.ucm.es`", el supercomputador del que se dispone ahora, es:

"**`abaco.sim.ucm.es`**". Para más información consulte el Apéndice del supercomputador de este documento.

Se ha realizado la migración del software y de la configuración, del supercomputador antiguo al nuevo, por lo tanto sería aconsejable comprobar que el puerto de escucha, para establecer comunicaciones entre el servidor del campus de Somosaguas y el supercomputador, está en modo LISTENING, esperando a que lleguen peticiones a atender.

El fichero **`aux_entorno.c`**, es el módulo principal de la implementación. En él, se encuentran implementadas funciones que vuelcan en los ficheros de entorno, los cuales tienen la extensión `*.dat`, la información extraída del contenido de las imágenes 3D, en el momento en el que se reciben, tras la rotación y después del ajuste punto a punto de los cortes.

Estos parámetros son utilizados posteriormente para realizar llamadas a las funciones del AIR ubicado en el supercomputador. Un ejemplo de fichero `fich_entorno.c`, es:

```
antonio 6
VPETA.inf 3 0 128 128 1 2.000000 2.000000 3.100000
VMRA.inf 11 0 256 256 1 1.000000 1.000000 1.000000
```



Más importante aún son, las funciones que generan los ficheros MAKE.com dinámicamente (genera_MAKE.c). Estos ficheros, incluyen las directivas de trabajo remotas (cuyos parámetros de entrada se obtienen en base a la información volcada previamente en los archivos *.dat), necesarias a la hora de ejecutar acciones en el supercomputador, permitiendo por tanto poder subir información desde el servidor para manipularla en el supercomputador, así como poder disponer de los resultados obtenidos en el supercomputador en el servidor. Estos ficheros son copiados remotamente en el supercomputador, manipulados de forma que se le otorguen permisos de ejecución, ejecutados con la orden 'system' y posteriormente borrados. Este proceso consta de diferentes fases, las cuales siguen básicamente, el procedimiento descrito previamente, de fusión de imágenes RM y PET.

En el fichero **aux_ayuda.c**, se encuentran funciones auxiliares destinadas a mostrar en formato html, la información de ayuda, la información propia de las imágenes almacenadas por un usuario concreto, el formulario web en el que se especifican los parámetros de la fusión, etc.

El fichero **visualiza_slice.c**, incluye la posibilidad de visualizar en una página web dinámica, un corte determinado del volumen generado en el supercomputador, en formato jpg.

El fichero **pre_reorient_fus.c**, contiene un tratamiento de las funciones que permiten reorientar los volúmenes ya ajustados, tanto de PET como de RM.

El módulo **entorno_fusion3d.c**, genera dinámicamente el código html que visualizará el usuario final, mostrándole los archivos de imágenes disponibles para trabajar.

El módulo **usuario_fusion3d.c**, realiza un control de la identificación del usuario, junto con los permisos que tiene asociados.

El fichero **aux_globales.c**, contiene funciones globales que usarán otros módulos C, entre ellas cabe destacar las funciones de consulta del formato de la imagen así como funciones de presentación de información vía web.

16.5. Formato imágenes DICOM

El uso de la informática en aplicaciones clínicas es una constante hoy en día, especialmente en el campo del diagnóstico por imagen. El uso de la imagen digital se ha ido imponiendo debido a los avances tecnológicos, ya que suponen una mejor calidad de la misma y la posibilidad de transmitirla a distintos puntos de manera casi inmediata. Además la necesidad de poder comunicar equipos heterogéneos entre sí, induce a la creación y uso de un estándar, siendo el problema fundamental para su empleo, la correcta interpretación de la



información. Es por ello que la American Collage of Radiology y la Nacional Electrical Manufacturers Association, se embarcaron en un proyecto para elaborar un estándar para la transferencia de imágenes y la información asociada a ellas, que tras varias fases, dio origen al formato DICOM 3.0.

Esencialmente es un protocolo de intercambio de datos común y no propietario, diseñado para trabajar en red, que permite definir algunos elementos esenciales del formato de las imágenes y que los fabricantes se comprometen a respetar, al menos al momento de transmitir a través de una red o al grabar en medios físicos. Define a su vez, interfases de conexión entre equipos de diferentes categorías. Se trata de un estándar evolutivo, que se modifica anualmente para permitir el progreso y la incorporación de nuevas tecnologías y modalidades. Es muy importante señalar, que el hecho de que un sistema sea compatible con el estándar DICOM no garantiza la comunicación completa entre equipos. Al estar evolucionando continuamente, las aplicaciones que dependan de la transmisión de imágenes deben actualizarse.

Una imagen DICOM dependiendo del tipo de modalidad, contiene una única imagen o una serie de ellas, adquiridas en un corto espacio de tiempo. Dicha imagen, está formada por un *encabezado* que la identifica como tal y que contiene la información asociada a la imagen, como por ejemplo: información privada del paciente, información del estudio, fecha de realización de la prueba, situación del paciente, etc. En este encabezado aparte de la información contenida en el estándar hay una gran cantidad de espacio reservado para que cada fabricante agregue la información que sea relevante para su propio desarrollo, y es allí donde se producen un buen número de incompatibilidades. Después del encabezado, encontramos la *representación de los datos* (es decir la representación de los voxels de la imagen), codificados como números de 16 bits (2 bytes). Generalmente se ocupan sólo 12 bits, lo que permite la generación de 2048 tonalidades de gris en la imagen final.



DICOMHeader
 Frames: 2
 Rows:109
 Columns: 91
 Bits stored: 8

2x109x91=19838 bytes

MRI.*

La imagen tiene una Resolución de 1 byte

16.5.1. Características de DICOM

- Intercambiabilidad de objetos en redes de comunicación y en medios de almacenamiento a través de protocolos y servicios, manteniendo sin embargo, independencia de la red y del almacenamiento físico. Todo esto a través de comandos definidos por una sintaxis y una semántica, a los que se les asocian datos. Las versiones anteriores sólo ofrecían comunicación punto a punto.

- Especificación de diferentes niveles de compatibilidad. Explícitamente se describe como definir un determinado nivel de compatibilidad, para escoger sólo opciones específicas de DICOM. En las versiones anteriores se especifica un nivel mínimo únicamente.

- Información explícita de Objetos a través de estructuras de datos, que facilitan su manipulación como entidades autocontenidas. Los Objetos no son únicamente imágenes digitales y gráficas, sino también estudios, reportes, etc.

- Identidad de objetos en forma única, como instancias con operaciones permitidas definidas a través de clases.



- Flexibilidad al definir nuevos servicios.

- Interoperabilidad entre servicios y aplicaciones a través de una configuración definida por el estándar, manteniendo una comunicación eficiente entre el usuario de servicios y el proveedor de los mismos

- Sigue las directivas de ISO en la estructura de su documentación multi-partes. De esta forma facilita su evolución, simplificando la adición de nuevas partes.

16.6. Servidor Dicom

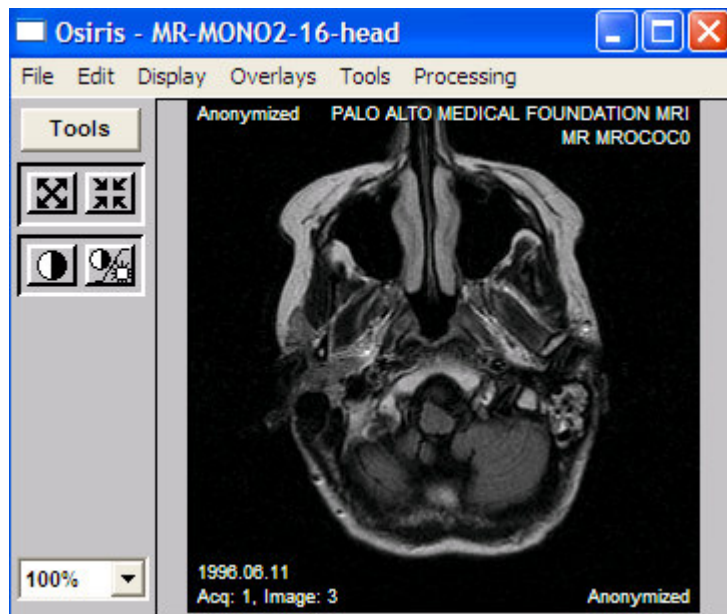
El servidor estudiado es el llamado Cosquest DICOM Server 1.4.8. Este servidor de imágenes, recibiría una query desde el servidor para ubicar una imagen de entre todas las disponibles. Todas las imágenes de un usuario, estarán almacenadas en este servidor y no de forma local.

Para realizar peticiones y comprobar el funcionamiento del software anterior, se hizo uso del cliente dicom k-pacs.

16.7. Software para visualización de imágenes Dicom



El programa Osiris está orientado al tratamiento de imágenes médicas, y permite leer imágenes en el formato DICOM. Este visualizador de imágenes médicas en formato DICOM es gratuito. OsiriX es una aplicación extremadamente especializada y orientada a los profesionales del ámbito médico. Permite la lectura y análisis en 2D y 3D de los datos generados por los escáneres de tipo CT, IRM o PET. El hecho de ser totalmente compatible con redes PACS (Picture Archiving and Communication system, es decir un sistema de archivo y distribución de imágenes médicas digitales), le permite recibir imágenes DICOM.



16.8. Arquitectura cliente-servidor y sockets

Se estudió inicialmente implantar una arquitectura cliente-servidor entre el servidor y el supercomputador, estableciendo las comunicaciones a través de sockets. Se llegaron a implementar módulos de comunicación entre clases locales pero que adquirirían papeles de cliente y servidor. Se pensó que era una forma óptima de no sobresaturar la carga en el supercomputador además de reducir considerablemente el tráfico de red.

Posteriormente, una vez que se pudo chequear los módulos implementados hasta el momento, se comprobó que la comunicación se realizaba a través de la ejecución de comandos de forma remota desde el servidor y mediante la instrucción `system` implementada en C, la cual permite la ejecución de comandos en el sistema.

Otro punto en contra que encontramos durante esta investigación fue el poder determinar si el supercomputador podía o no albergar un proceso servidor que estuviese atendiendo las peticiones recibidas, este punto debería consultarse con la persona encargada del mantenimiento del supercomputador.



17. TRABAJO FUTURO

Llegados a este punto, nos gustaría poder indicar como proseguir en el avance de este proyecto para poder alcanzar los objetivos que nos planteamos en su momento y que desgraciadamente no hemos podido alcanzar. Enumeramos a continuación por orden de prioridad los aspectos que se podrían tratar, así como mejoras futuras:

- Implantar la implementación con J2EE seguida por nosotros, reimplementando el conjunto de CGI's actuales.

- Paralelizar el código del AIR ubicado en el supercomputador, mejorando la eficiencia de los bucles, especificando las opciones óptimas de compilación, etc.

- Incluir ficheros logs en el servicio, en los cuales se almacenará información de monitorización periódica de las acciones llevadas a cabo, como por ejemplo puede ser, accesos de usuario, ubicación desde la que se accede, secuencia de pasos realizados. Sería conveniente así mismo, realizar un script incluido en el crontab del servidor, que permita la rotación y borrado de los logs, para que en casos de un volcado masivo de información no se quede el servidor sin recursos.

- Con el fin de alcanzar un grado de alta disponibilidad, incorporar en las comunicaciones entre el servidor de Somosaguas y el supercomputador, un balanceador que redirija las peticiones en primer lugar al supercomputador del Centro de cálculo, pero en caso en el que éste esté caído, mandar las peticiones a un supercomputador de backup. O bien, estudiar como integrar el servicio dentro de la tecnología Grid, de forma que mediante el acceso remoto a los recursos cedidos por los ordenadores infrautilizados dispersos geográficamente, se puedan procesar aplicaciones, manteniéndose el servicio disponible en todo momento.

- Comprobar si hay disponible, software del AIR, que admita un formato Dicom de las imágenes de entrada, evitando así tener que hacer la transformación inicial de las imágenes recibidas, en imágenes que sigan el formato admitido por el AIR.



18. APÉNDICES

18.1. Supercomputador abaco.sim.ucm.es

Es una máquina del tipo Altix 3700 Bx2. Está ubicado en el Centro de procesamiento de datos de la Universidad Complutense de Madrid.

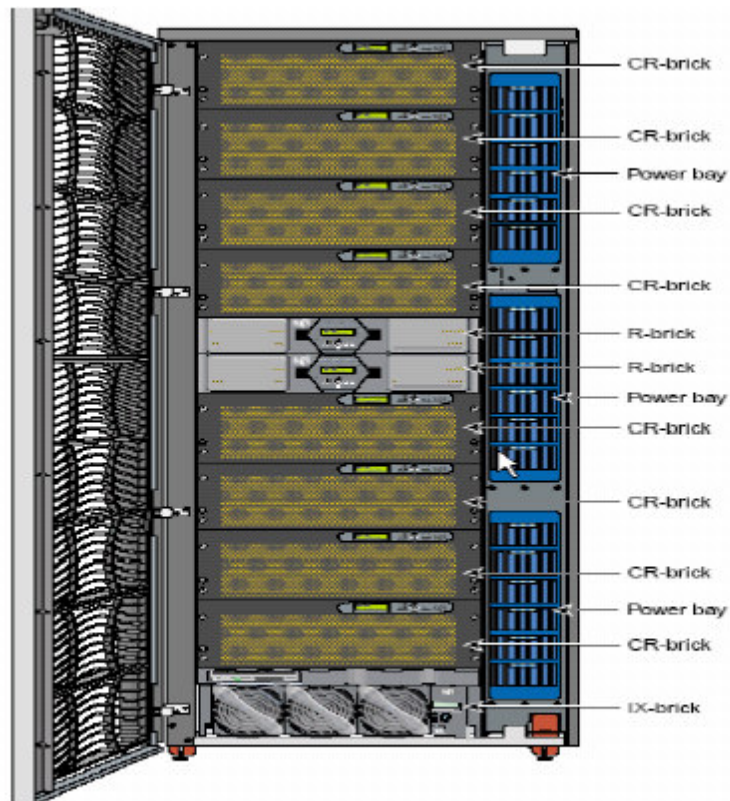


Figure 3-2 SGI Altix 3700 Bx2 Server System (Example Configuration)

Este sistema usa un espacio global de direccionamiento, con dirección ip **147.96.1.107** y dns “abaco.sim.ucm.es”, consta de 64 procesadores 1600 MHz Intel Itanium 2 Rev. 1 Processor y con una memoria 128 GB.

Para acceder a los servicios de ábaco, es necesario realizar una conexión remota a través del protocolo **ssh**.

El sistema operativo instalado es Linux basado en Redhat Enterprise AS version 3.0 con SGI Propack 3 para Linux. La versión del Kernel es la 2.4.21-sgi302r24. Por cada proceso que ejecute un usuario de forma interactiva, tiene una limitación de 20 minutos de CPU (para ver los límites de usuario ejecutar el comando limit). Si se necesita ejecutar algún trabajo que exceda el tiempo de 20 minutos de tiempo de CPU, se tendría que enviar a una de las colas, controladas



por el gestor de colas LSF, que permite ejecutar procesos **batch** a todos los usuarios que poseen cuenta en este sistema.

batch1d:

Limite CPU 24 horas,
1 semana de ejecución,
24 trabajos simultáneos,
8 por usuario.
Ejecución en 1 CPU,
Permite ejecución en interactivo.
Cola por defecto.

batch15d:

Limite CPU 15 días,
1 mes de ejecución,
24 trabajos simultáneos,
8 por usuario.
Ejecución en 1 CPU.

batchduro:

2 meses,
4 meses de ejecución,
64 trabajos simultáneos,
16 por usuario.
Ejecución en 1 CPU.

paralelo:

Limite CPU 2 meses,
4 meses de ejecución,
8 trabajos simultáneos,
8 por usuario.
Ejecución hasta en 8 CPU.

Los comandos que nos permiten trabajar con estas estructuras de datos, son:

Enviar un trabajo a la cola:

bsub [-q nombreCola] [job [argumento]]

Visualizar el estado de las colas:

bqueues [-w | -l [nombreCola]]

Visualizar el estado de un trabajo en concreto:

bjobs [-w | -l] [-q nombreCola] [-u nombre_usuario | all]

Eliminar un trabajo de una cola:

kill id

Es en esta máquina, donde están disponibles los ejecutables del AIR, que utiliza la aplicación. Mostramos a continuación un listado de los ejecutables disponibles:

```
ls /usr/users/dariomv/sbin

alignlinear    cd_warp        fusionC        magnify
mv_warp        r8tohdf.bak   resize        scanheader
totfusionC     align_warp     combine_air
fusionIHS      makeaheader    normalizar
rawtopnm2     reslice        scan_warp
totfusionIHS   binarize       combine_warp
gsmooth        makevolume     palthdf
reconcile_air  reslice_warp  separate
totpnm2oraw   binarymask     definecommon_air
identify       manualreslice  palthdf.bak
reconcile_rigidbody
reunite        setheadermax  totrawtopnm2
```



binarymath	expandir	invert_air	mheader
pnmtoraw	rendervolume3	reunitevol	sizeof
zoomer	cd_air	fixheader	layout
mv_air	r8tohd5	reorient	
scanair	softmean		

18.2. Tuning del Air 3.08 (aplicado al estudio de alinear)

Parámetros de alinear

La descripción de los parámetros analizados es la siguiente:

alignlinear meanpet.emri emri.airmeanpet -m 6 -p1 0 -p2 256 -t2 10

alignlinear	file	meanpet.emri
	ref-frame	emri.airmeanpet
	-m	model_menu_number:6, 7, 9, 12, 15
	[options]	-p1 = 0 , -p2 = 256 ; -t2 = 10

Model Menu: escoger el modelo de puesta en correspondencia

Modelos 3-D:

- 6. parámetro para modelos rígidos.
- 7. parámetro para modelos globales
- 9. parámetro para modelos transicionales
- 12. parámetro para modelos "affine"
- 15. parámetro para modelos en perspectiva

opciones:

[-b1 FWHM_x(4.000000e+00) FWHM_y (4.000000e+00) FWHM_Z(4.000000e+00)]	(standard file) Filtración de la imagen Estándar (=0 no hay filtración)
[-b2 FWHM_x(4.000000e+00) FWHM_y(4.000000e+00) FWHM_Z(4.000000e+00)]	(reslice file) Idem
[-p1 partitions_in_standard_file]	256 si Standard_File es el IRM; 0 si es el PET 1 si es PET/PET o IRM/IRM
[-p2 partitions_in_reslice_file]	Idem
[-f initialization-file]	Poner un nombre de un fichero que cargara las condiciones iniciales de la puesta en correspondencia.



[-g termination-file [overwrite?(y/n)]]	Poner un nombre de un fichero que cargara las condiciones finales de la puesta en correspondencia.
[-s initial_sampling(81) final_sampling(1) sampling_decrement_ratio(3)]	3 parámetros que permiten de acelerar el proceso por initial-sampling una valor alta, final-sampling superior a 0. Si esta demasiado alta, hay biasas.
[-c convergence-threshold (0.000010)]	Criterio de parada.
[-r repeated-iterations (25)]	Repetir a cada paso de energía. No puede estar demasiado pequeño pero se puede reducir.
[-h halt-after-(N)-iterations-without-improvement (5)]	Numero de iteración aceptable sin mejor
[-a alternate-strategy-after-(M)-iterations-without-improvement (6)]	Idem pero con matices. Tiene que sea superior a -r y -h.
[-q] assume noninteraction of spatial parameter derivatives	Si seleccionado, puede acelerar el proceso con un coste <i>standard deviation of ratio image</i> minimo.
[-z] enable pre-alignment interpolation	Si voxels son muy anisotropic y no cubics, esta opción puede mejorar los resultados.
[-x cost-function]	Determina cual es la función que minimiza el proceso.
Cost functions:	
1. standard deviation of ratio image	Independientemente de la intensidad de las Imágenes.
2. least squares	exige buena preparación de los datos. Mas rápido.
3. least squares with intensity scaling	Idem

Los parámetros tienen parámetros por defectos. Estarán todos fijos (-p1 0 -p2 256 -t2 -x 1 -a -h -r -c -s) pero estudiaremos la influencia de **-z -q**. Así, para cada modelo de puesta en correspondencia (X=6, 7, 9, 12, 15), lanzamos 4 AIR con los siguientes parámetros.

1. **Modelo X-1 : valignlinear I1 I2 -m [Modelo] -z -q**
2. **Modelo X-2 : valignlinear I1 I2 -m [Modelo] -z**
3. **Modelo X-3 : valignlinear I1 I2 -m [Modelo] -q**
4. **Modelo X-4: valignlinear I1 I2 -m [Modelo]**

Resultados acerca de los modelos 3D

Las tablas siguientes muestran los resultados de evaluación de la puesta en correspondencia, realizada con diferentes parámetros del AIR y según el tiempo de ejecución y los tres cálculos de errores: standard deviation of ratio image, least squares, least squares with intensity scaling.



I) Tabla de evaluación de puestas en correspondencia con *standard deviation of ratio image*.

Standard Deviation	X-1		X-2		X-3		X-4	
	Cost	Tiempo	Cost	Tiempo	Cost	Tiempo	Cost	Tiempo
Modelo 6	0.567634	0m21.80	0.567634	0m38.02s	0.567634	0m17.48s	0.567634	0m17.48s
Modelo 7	0.566142	0m13.35	0.566141	0m38.45s	0.566142	0m11.04s	0.566141	0m34.17s
Modelo 9	0.565784	0m38.33	0.565787	1m5.36s	0.565784	0m34.05s	0.565787	1m1.03s
Modelo 12	0.570823	0m20.40	0.565375	1m22.69s	0.570823	0m30.61s	0.565375	1m18.33s
Modelo 15	-	0m45.12	-	0m26.35s	-	0m40.66s	-	0m40.66s

II) Tabla de evaluación de puestas en correspondencia con *least squares*.

Least Squares	X-1		X-2		X-3		X-4	
	Cost	Tiempo	Cost	Tiempo	Cost	Tiempo	Cost	Tiempo
Modelo 6	6349.8	3m3.63s	6335.54	3m9.67s	6349.8	1m51.14s	6335.54	3m4.67s
Modelo 7	5447.83	1m51.71s	5427.89	1m46.88s	5447.83	1m47.01s	5427.89	1m42.23s
Modelo 9	5659.01	1m5.24s	5771.85	0m27.48s	5659.01	1m2.59s	5771.85	0m41.27s
Modelo 12	5495.39	1m49.20	5538.02	1m33.03	5495.39	1m44.59s	5538.02	1m28.71s
Modelo 15	6243.78	0m27.89	6243.78	0m27.90	6243.78	0m13.92s	6243.78	0m13.92s

III) Tabla de evaluación de puestas en correspondencia con *least squares with intensity scaling*.

Least Squares with intensity scaling	X-1		X-2		X-3		X-4	
	Cost	Tiempo	Cost	Tiempo	Cost	Tiempo	Cost	Tiempo
Modelo 6	3453.32	0m30.23s	3453.32	0m14.86s	3453.32	0m25.90s	3453.32	0m20.23s
Modelo 7	3490.58	0m29.45s	3510.3	0m31.37s	3490.58	0m44.94s	3510.3	0m28.60s
Modelo 9	3478.95	0m50.64s	3503.62	0m30.89s	3478.95	0m46.16s	3503.62	0m46.92s
Modelo 12	3461.95	0m49.69s	2151.61	3m27.01s	3461.95	0m45.31s	2151.61	3m22.98s
Modelo 15	6243.78	0m16.41s	6243.78	0m27.92s	6243.78	0m13.91s	6243.78	0m23.62s

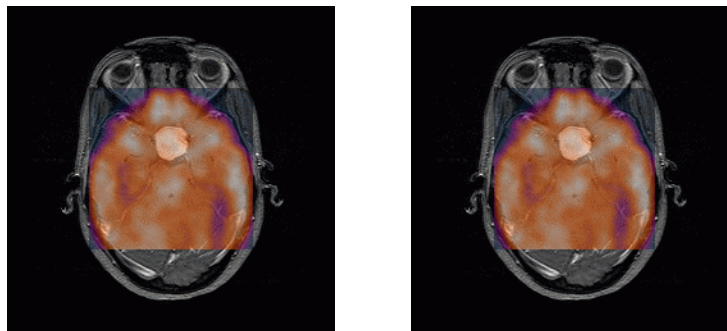


De las tablas precedentes, se puede ver que el tiempo de ejecución aumenta en función del uso de parámetros diferentes de los tomados por defecto.

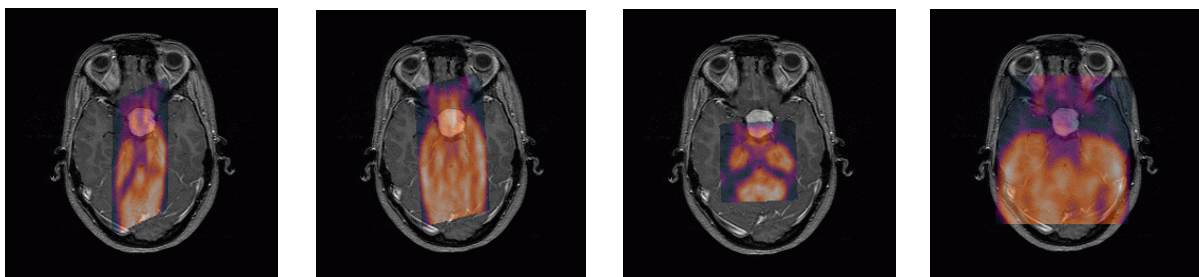
Estos resultados muestran que la opción "[z] enable pre-alignment interpolation no tiene efectos negativos, sino que es recomendable ponerlo siempre, puesto que sirve para indicar que los voxeles son cúbicos y no anisotrópicos en nuestra aplicación. Sin embargo, la opción "[q] asume "noninteraction of spatial parameter derivatives", por lo que es muy importante no especificarla puesto que en nuestro caso, estamos ante interacciones espaciales.

Como conclusión a los datos anteriores, deducimos que los modelos 7 y 12 son mejores que los modelos 6, 9 y 15. En la aplicación, los modelos 7 y 12 son los mejores según los cálculos (sin opción especiales -q y -z), pero visualmente, son los peores. Esta conclusión no puede garantizar que será igual para otras aplicaciones.

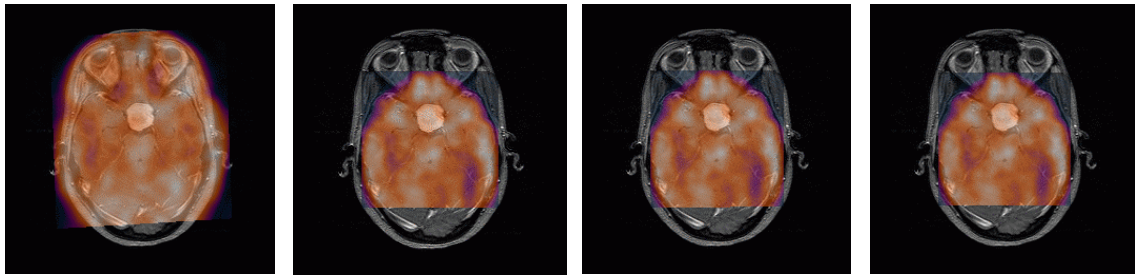
Las siguientes imágenes muestran los resultados de puesta en correspondencia de los dos volúmenes después la fusión de colores (parámetro igualado a 0.74).



I) Standard Deviation: Izquierda X-3 Modelo 12 (Cost = 0.570823), Derecha: X-4 Modelo 12 (Cost = 0.565375). No hay diferencias notables entre modelos 6, 7, 9, 12. Modelo 15 no funciona.



II) Least Square 1: de la izquierda la derecha X-4 con: Modelo 12 (Cost = 5538.02), Modelo 9 (Cost = 5771.85), Modelo 7 (Cost = 5427.89), Modelo 6 (Cost = 6335.54). El criterio de minimización no esta adecuado a nuestra aplicación.



III) Least Square 2: : de la izquierda a la derecha X-4 con: Modelo 12 (Cost = 2151.61), Modelo 9 (Cost = 3503.62), Modelo 7 (Cost = 3510.3), Modelo 6 (Cost = 3453.32). El criterio da una mínima para el resultado visualmente lo peor. El criterio no está adecuado a nuestra experiencia.

El criterio de minimización es el más importante y tiene que ser el standard deviation en la aplicación. La diferencia entre los resultados de los modelos 6, 7, 9, 12 no es apreciable de no ser por la medida de errores.

Para seguir ajustando el software AIR, deberíamos comprobar los efectos resultantes de utilizar los parámetros -b1, -b2, [-c convergence-threshold (0.000010)], [-r repeated-iterations (25)], [-h halt-after-(N)-iterations-without-improvement (5)] y [-a alternate-strategy-after-(M)-iterations-without-improvement (6)]. El aspecto combinatorio de la prueba, no nos permite hacerlo de manera exhaustiva.



19. AGRADECIMIENTOS

Nos gustaría dejar constancia de nuestro agradecimiento a las personas que nombramos a continuación:

A nuestra decana, por hacer posible nuestra ilusión por este proyecto.

A nuestro director, Luis Hernández Yáñez, por aceptar este reto, por su amabilidad, comprensión y buenos consejos.

A nuestro cliente, Luis Jáñez Escalada, por su colaboración y por permitirnos entrar en el conjunto del gran proyecto que se está llevando a cabo.

A los colaboradores de Luis Jáñez, Juan, Iván y Jose, sin relevancia sobre este orden, por acudir a las reuniones y ayudarnos a conocer lo que ya estaba hecho.

Y por supuesto, a nuestros familiares y amigos, por soportarnos en los momentos difíciles y de tensión que ha supuesto el fin de carrera para nosotros.



20. BIBLIOGRAFÍA

Entre la diversa bibliografía consultada, podemos destacar:

J2EE:

- Keogh, Jim , *J2EE: Manual de referencia*. McGraw-Hill (2003).

Información de puesta en correspondencia imágenes médicas:

- P.A.VAN DEN ELSEN, E.D. POL, ET M.A. VIERGEVER, *Medical image matching: A review with classification*, IEEE Engineering in Medicine and Biology, 12 (1993), pp. 26-38.
- O.D. FAUGERAS ET M. HEBERT, *The presentation, recognition and locating of 3-d objects*, Int. J. Robotic Research, 5 (1986), pp. 27-52.
- HUILING LE ET CHRISTOPHER G. SMALL, *Multidimensional Sealing of Simplex Shapes*, Nottingham Statistics Group, University of Nottingham, Department of Mathematics, 1997, pp.1-23.

Información relativa al AIR:

- Jiang AP, Kennedy DN, Baker JR, Weisskoff R, Tootell RBH, Woods RP, Benson RR, Kwong KK, Brady TJ, Rosen BR, Belliveau JW. Motion detection and correction in functional MR imaging. *Human Brain Mapping* 1995;3:224-235

Descripción del AIR 3.0 y validación de datos de RM y de PET:

- Woods RP, Grafton ST, Watson JDG, Sicotte NL, Mazziotta JC. Automated image registration: II. Intersubject validation of linear and nonlinear models. *Journal of Computer Assisted Tomography* 1998;22:155-165.

Información del servidor Dicom:

<http://www.xs4all.nl/~ingenium/dicom.html>

Información de imágenes medicas:

<http://www.barre.nom.fr/medical/samples/>

<http://www.psychology.nottingham.ac.uk/staff/cr1/dicom.html>

Información de comunicación vía sockets:

<http://www.geocities.com/chuidiang/java/sockets/socket.html>



http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/V_2.htm

Información sobre comunicaciones seguras:

<http://www.cl.cam.ac.uk/~fapp2/software/java-ssh/>

Información sobre cvs:

www.cvsnt.com

Información de arquitectura cliente-servidor:

<http://neo.lcc.uma.es/evirtual/cdd/codigos/servidortcp.html>



21. AUTORIZACIÓN

Por la presente, los abajo firmantes Don Juan Antonio Martín Pedro, Doña Eva Cristina Sánchez San José y Doña Beatriz Herrezuelo Sánchez, autorizan a la Universidad Complutense de Madrid a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a sus autores, tanto la propia memoria, como el código, la documentación y/o el prototipo desarrollado.

Y para que así conste, firman:

Juan Antonio Martín Pedro

Eva Cristina Sánchez San José

Beatriz Herrezuelo Sánchez

Madrid, Junio de 2005.