



PARAMETRIZACIÓN DE CMS PARA ANÁLISIS SANGUÍNEOS

Memòria del projecte
d'Enginyeria Tècnica en
Informàtica de Sistemes
realitzat per
Sergi Vicente Llenas
i dirigit per
Òscar Cubillo Alonso

Escola d'Enginyeria

Sabadell, Setembre de 2012

FULL DE RESUM – PROJECTE FI DE CARRERA DE L'ESCOLA D'ENGINYERIA

Títol del projecte:

PARAMETRIZACIÓN DE CMS PARA ANÁLISIS SANGUÍNEOS

Autor[a]: Sergi Vicente Llenas

Data: 09 - 2012

Tutor[a]/s[es]: Òscar Cubillo Alonso

Titulació: Enginyeria Tècnica en Informàtica de Sistemes

Paraules clau (mínim 3)

- Català: CMS, Drupal, Parametrització
- Castellà: CMS, Drupal, Parametrización
- Anglès: CMS, Drupal, Parameteritzation

Resum del projecte (extensió màxima 100 paraules)

• Català:

El projecte ha consistit en la implantació d'una web per a DBSS, amb doble finalitat: una pública d'informació general sobre la mateixa i altre privada per l'automatització de l'entrada de dades d'anàlisis de sang i la seva posterior consulta per part del pacient. S'han seleccionat les tecnologies adequades tant pel disseny de la web com de les bases de dades associades. S'han programant la web i les bases de dades amb èxit i s'ha realitzat la implementació del sistema, realitzant-se proves de funcionament internes i externes amb èxit, comprovant-ne la bondat de l'aplicació.

• Castellà:

El proyecto ha consistido en la implantación de una web para DBSS, con doble finalidad: una pública de información general sobre la misma y otra privada para la automatización de la entrada de datos de análisis de sangre y su posterior consulta por parte del paciente. Se han seleccionado las tecnologías adecuadas tanto para el diseño de la web como de las bases de datos asociadas. Se han programado la web y las bases de datos con éxito y se ha realizado la implementación del sistema, realizándose pruebas de funcionamiento internas y externas con éxito, comprobándose la bondad de la aplicación.

• Anglès:

The project has involved the establishment of a website for DBSS, with a dual purpose: one public of general information about itself and other private for automating data entry of blood analysis and its later consulting by the patient. Appropriated technologies have been selected for web design and associated databases. The website and the databases have been programmed successfully and performed system implementation, performing tests of internal and external functioning successfully, checking the goodness of the application.

AGRADECIMIENTOS

Quiero expresar todo mi agradecimiento a Òscar Cubillo, director de este proyecto, por toda su ayuda, los sabios consejos y a la confianza que ha depositado en mí, ya que sin él no hubiera podido recorrer el camino.

Desde mi llegada a Dry Blood Spot Screening, S.L. ha pasado prácticamente medio año en el cual he tenido la oportunidad de conocer a personas de una gran calidad humana. Gracias a su apoyo han hecho que haya podido llevar a cabo este proyecto. A todos ellos quiero manifestarles mis más sinceros agradecimientos. Y en especial a Anabel, que ha vivido el proyecto conmigo siempre dispuesta a ayudar en cualquier situación.

A todas aquellas personas que de una u otra forma, colaboraron o participaron en la realización de este proyecto, hago extensivo mi más sincero agradecimiento, sin olvidar a mis amigos y familiares que siempre fueron un respaldo muy importante.

Por último y los más importantes a mis padres, Manel y Rosa María, porque siempre han estado respaldándome y sobre todo esforzándose en darme educación y formación, y a mi hermano Albert que siempre me ha apoyado cuando lo he necesitado.

A todos, muchas gracias por vuestra dedicación y preocupación.

ÍNDICE

1. INTRODUCCIÓN	2
1.1. Diagnóstico precoz de enfermedades	2
1.1.1. Tecnología DBSS	5
1.2. Justificación del proyecto de informatización	8
2. OBJETIVOS, ALCANCE Y PLANIFICACIÓN	12
2.1. Objetivos	12
2.2. Alcance	13
2.3. Planificación	14
2.4. Estimación de costes.....	17
3. ANÁLISIS FUNCIONAL	20
3.1. Análisis de requerimientos	20
3.1.1. Descripción de la aplicación.....	20
3.2. Requisitos funcionales	21
3.2.1. Roles/Perfiles	21
3.2.2. Contenido.....	23
3.2.3. Gestión de grupos	25
3.2.4. Vistas (Views)	27
3.2.5. Secciones/Estructuras de la web	27
3.2.6. Gestión de ficheros	30
3.2.7. Administración	30
3.2.8. RSS	33
3.2.9. Logs del sistema	34
3.2.10. Recuperabilidad	34
3.3. Requisitos no funcionales	35
3.3.1. Seguridad	35
3.3.2. Fiabilidad	35
3.3.3. Usabilidad.....	35
3.3.4. Eficiencia	36
3.3.5. Mantenimiento	36
3.3.6. Portabilidad	36

3.4. Diagrama casos de uso.....	37
3.4.1. Anonymous User	37
3.4.2. Authenticated User	37
3.4.3. Organizer	38
3.4.4. Administrator	39
3.5. Descripción casos de uso.....	40
4. DISEÑO Y PARAMETRIZACIÓN DE DRUPAL.....	48
4.1. Justificación del uso e implementación de un CMS y su elección	48
4.1.1. Elección del CMS	50
4.2. El gestor de contenidos Drupal	50
4.2.1. Módulos en Drupal.....	51
4.2.2. Arquitectura	52
4.2.3. Características generales	53
4.2.4. Bases de datos y seguridad	56
4.3. Gestión de usuarios	56
4.4. Herramientas de desarrollo y entorno de pruebas	57
4.5. Esquema de red	58
4.6. Instalación Drupal	59
4.6.1. Requerimientos del servidor.....	59
4.6.2. Configuración del servidor	59
4.6.3. Creación de las BBDD's	60
4.6.4. Proceso de instalación de Drupal.....	60
4.6.5. Configuración inicial del sitio web	60
4.7. Módulos utilizados.....	61
4.8. Requisitos no funcionales	66
4.8.1. Usuarios y permisos	66
4.8.2. Configuración del aspecto visual	67
4.8.3. Creación de bloques y menús	68
4.8.4. Tipos de contenido.....	70
4.8.5. Vistas	71
4.8.6. Logs del sistema.....	73
4.9. Otras funcionales	73

4.9.1. Gestión de ficheros	73
4.9.2. Mirroring	77
4.9.3. Backup	77
4.9.4. Cron	79
4.9.5. Actualización	79
5. CONCLUSIONES.....	84
5.1. Cumplimiento de objetivos.....	84
5.2. Retrasos en planificación	85
5.3. Líneas de ampliación.....	87
5.4. Valoración final	88
6. GLOSARIO	90
7. ÍNDICE DE TABLAS Y FIGURAS.....	92
7.1. Índice de tablas	92
7.2. Índice de figuras.....	93
8. BIBLIOGRAFÍA	96

1. INTRODUCCIÓN

1. INTRODUCCIÓN

Este trabajo final de carrera se ha realizado en Dry Blood Spot Screening, S.L. (DBSS). La empresa, de nueva creación, está ubicada en Sant Cugat del Vallès (Barcelona) y centra sus esfuerzos en facilitar un diagnóstico sanguíneo preventivo tanto de enfermedades infecciosas como de enfermedades crónicas o no transmisibles para comunidades desfavorecidas y con sistemas de salud con pocos recursos.

Desarrollando su propia tecnología y basándose en un modelo de subsidio mixto, su intención no se limita a ampliar el rango de parámetros bioquímicos sencillos a analizar, que permiten hacer el seguimiento sobre factores de riesgo para determinadas enfermedades crónicas, sino también obtener una gran cantidad de datos, que tras su tratamiento y análisis, se conviertan en información relevante para la optimización de las políticas y estrategias a realizar por las autoridades sanitarias.

1.1. Diagnóstico precoz de enfermedades

El diagnóstico precoz y la prevención en salud están ampliamente promovidos por cada uno de los sistemas de salud pública, organizaciones internacionales y ONG's que trabajan en el terreno. Pero a pesar que la prevención y el diagnóstico precoz sean de gran importancia para el propio paciente y representen un importante ahorro para los sistemas de salud pública, la implementación de estas medidas quedan limitadas por la falta de la tecnología adecuada y por la indefinición de prioridades sanitarias.

Las razones de esta situación son conocidas. Por una parte, se sabe que en los países en vías de desarrollo mantener la cadena de frío en todo momento para que las muestras de sangre no pierdan sus propiedades no es posible. También es conocido que, en países como el nuestro y para determinados colectivos, por ejemplo la población infantil, el hecho de tomar muestras de sangre por punción venosa (flebotomía) resulta muy difícil. Estos dos hándicaps, junto con el coste actual que representa el proceso de toma de muestras sanguíneas explican el porqué no se realizan campañas preventivas.

Desde el punto de vista tecnológico, DBSS da solución a estos 3 problemas al desarrollar un diagnóstico sanguíneo por punción capilar (punción en un dedo) y depositando la gota de sangre en papel de filtro, facilitando de ese modo, la prevención masiva, de forma individualizada, de enfermedades transmisibles (infecciosas) y no transmisibles (crónicas), tanto en países desarrollados como en comunidades desfavorecidas.

La tecnología desarrollada es sencilla, efectiva, indolora y económica. No precisa ni condiciones especiales de almacenaje, ni cadena de frío durante el transporte, a la vez que la inversión en formación al personal que debe manipular las muestras es mínima. Su sencillez, la hace fácilmente replicable y el coste que presenta se sitúa entre el 40-50% por debajo que utilizando técnicas tradicionales de extracción.

Para hacer frente a la fragmentación y al bajo volumen de pruebas diagnósticas realizadas en países en vías de desarrollo, DBSS ha establecido un novedoso modelo de negocio que permite la aplicación a gran escala de su nueva tecnología (Figura 1.1.).

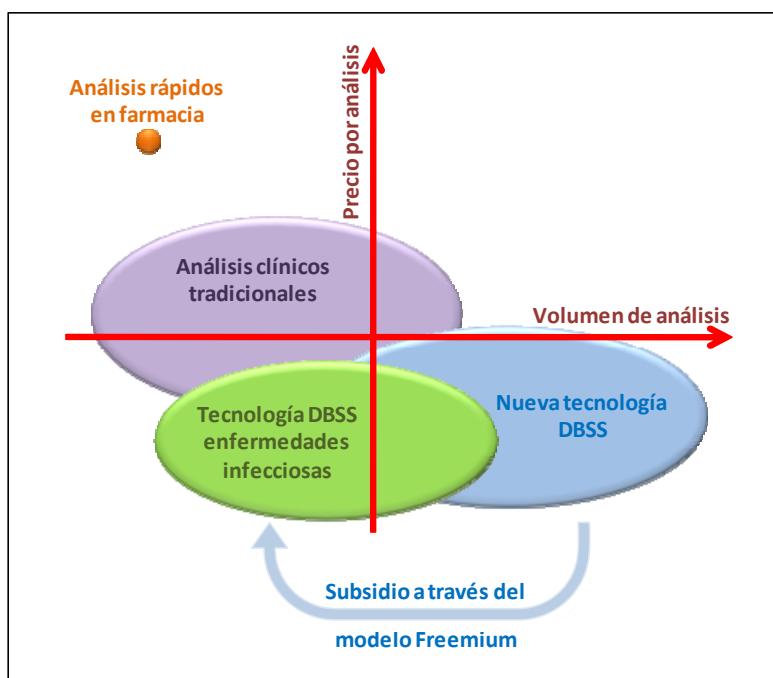


Figura 1.1. Posicionamiento de DBSS y modelo de negocio

Este modelo está basado en un innovador modelo de franquicias, con productos diferenciados por perfiles diagnósticos (que responden a las necesidades sanitarias tanto de los países desarrollados como de los países en vías de desarrollo) y a la vez una segmentación de los precios en cada uno de estos países, para subsidiar los costes.

DBSS actúa a nivel global. Nace con la intención de replicar su modelo en cuantos países y zonas lo soliciten. El ámbito de actuación y razón de ser de DBSS, punto innegociable, es el mantener la dualidad que lo hace estar presente en el primer mundo y en países en vías de desarrollo, de forma que uno retroalimente al otro.

La clave del éxito de DBSS en un nuevo mercado, radica en saber mantener también la dualidad local-global. DBSS nace global, pero actúa de forma local, dando respuesta a las necesidades locales y contratando y formando personal de la zona en la que se van a llevar a cabo las actuaciones.

Sin embargo, el desarrollo de nuevas necesidades específicas se llevarán a cabo en un único laboratorio central de I+D+i, situado en España, desde el que se realizará la transferencia de tecnología, el soporte técnico y la formación del personal nativo (ver Tabla 1.1.).

Mercado	Tipo de economía	Modelo de negocio	Tipo de enfermedad	Tecnología
Nacional	Desarrollada	Propio	Crónicas Infecciosas	Centralizada en España.
Internacional	Desarrollada	Franquicia	Crónicas Infecciosas	Un único laboratorio de I+D+i
	En vías de desarrollo	Franquicia	Crónicas Infecciosas	

Tabla 1.1. Estrategia de crecimiento

Por otra parte, existe una diferencia evidente entre las enfermedades a prevenir dependiendo del país y de su ubicación geográfica. De este modo, el sustituir la toma de muestras sanguíneas líquidas por una toma sobre papel de filtro permitirá:

- En países desarrollados: una importante mejora del diseño de las campañas de prevención, al poder acceder a un número mayor de la población y en especial a aquella que presenta cierta dificultad, focalizándose en la población infantil y adolescente.

Se han priorizado como enfermedades crónicas a tratar aquéllas que según la OMS y los expertos tienen gran transcendencia, como son: los problemas cardiovasculares (una gran parte provienen del problema de la obesidad infantil), la insuficiencia renal crónica (IRC), la diabetes y las disfunciones hepáticas, todas ellas conocidas, también, como las “epidemias del siglo XXI”.

- En países en vías de desarrollo: un diagnóstico preventivo tanto de enfermedades infecciosas como enfermedades no transmisibles crónicas. Se han priorizado como objetivos las enfermedades infecciosas y en especial a mujeres embarazadas, para obtener una reducción en la transmisión vertical madre-feto de enfermedades como el VIH, la Hepatitis, la Sífilis, etc.), sin dejar de lado el diagnóstico precoz de la diabetes.

1.1.1. Tecnología DBSS

La tecnología desarrollada por DBBS se basa, como se ha comentado, en el diagnóstico sanguíneo por punción capilar (punción en un dedo) y depositando la gota de sangre en papel de filtro. Una vez seca la sangre se transporta al laboratorio de análisis donde se lleva a cabo su reconstitución y el análisis cuantitativo de los parámetros bioquímicos e inmunológicos a determinar.

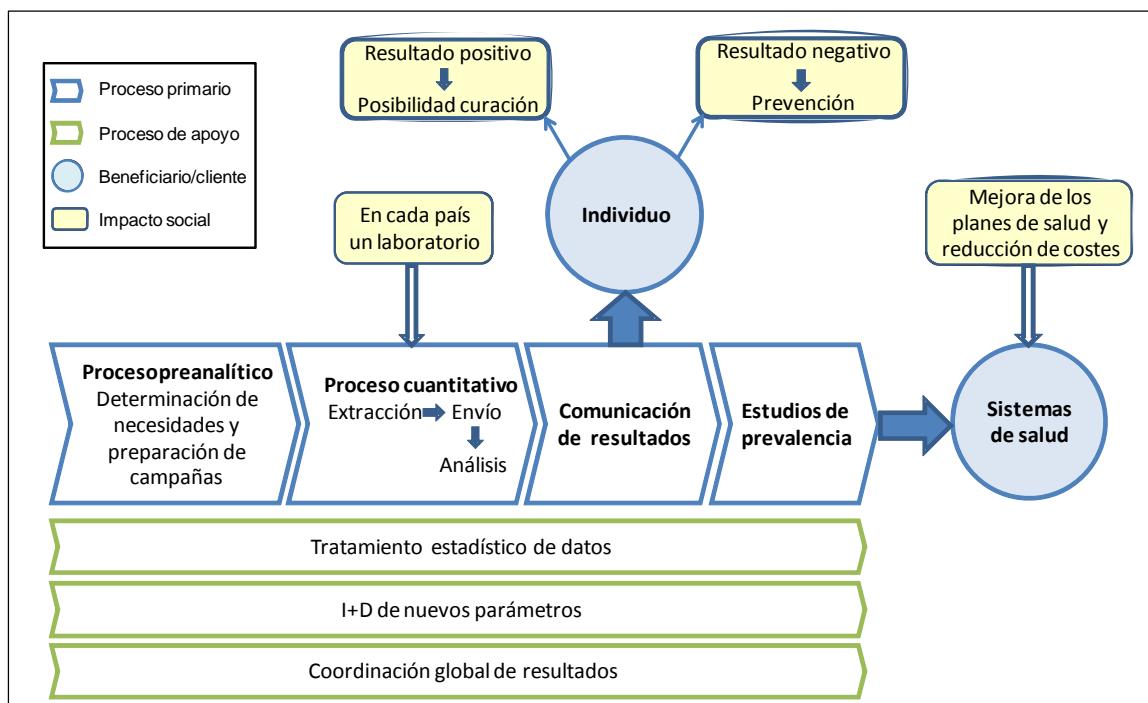


Figura 1.2. Ventajas de la tecnología DBSS frente a la tradicional

Como se puede observar en la Figura 1.2., la tecnología desarrollada es sencilla, efectiva, indolora y económica. No precisa ni condiciones especiales de almacenaje, ni cadena de frío durante el transporte, a la vez que la inversión en formación al personal que debe manipular las muestras es mínima.

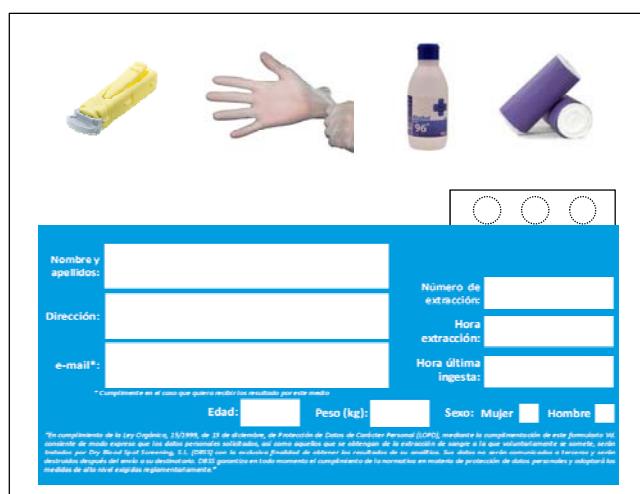
Su sencillez, la hace fácilmente replicable y el coste que presenta se sitúa aproximadamente en un 50% por debajo del que presentan las técnicas tradicionales de extracción.

En la Figura 1.3. se muestra esquemáticamente la cadena de valor, en la que se distinguen los procesos primarios y los de apoyo, así como los beneficiarios y los puntos con impacto social.

**Figura 1.3. Cadena de valor**

Cabe señalar que el proceso cuantitativo de determinación de los parámetros se ha dividido en tres etapas: extracción, envío de las muestras secas y el propio análisis de las mismas.

En la etapa de extracción cabe remarcar que el material necesario es la lanceta (para llevar a cabo la extracción capilar en un dedo de la mano), guantes, alcohol y algodón (como material auxiliar) y el propio papel de filtro para la toma de muestra incluido en la misma tarjeta para la recogida de los datos personales (tal como se muestra en la Figura 1.4. adjunta).

**Figura 1.4. Material necesario para la extracción**

Por otra parte en la Figura 1.5., se muestra el proceso de análisis de forma esquemática.

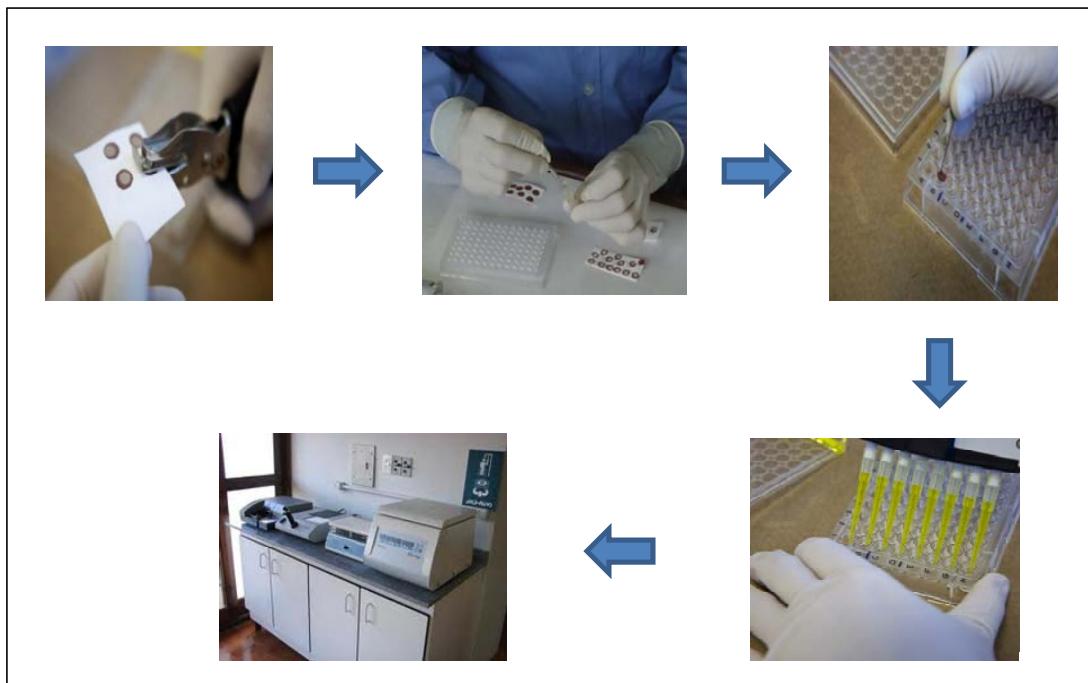


Figura 1.5. Etapas del proceso de análisis cuantitativo en el laboratorio

Este proceso cuantitativo de determinación de las concentraciones de los parámetros bioquímicos incluye las siguientes etapas:

- Reconstitución de la muestra de sangre seca. Para ello se usa un reconstituyente específico para cada parámetro bioquímico. Se utiliza un área determinada y siempre la misma del papel de filtro.
- Reacción colorimétrica. Para cada parámetro se utilizan reactivos específicos para la formación del complejo que presente color a una determinada longitud de onda. La reacción se lleva a cabo en una placa tipo Elisa.
- Determinación cuantitativa. Se mide la absorbancia de la solución mediante un espectrofotómetro. A partir de ese dato se calcula la concentración del parámetro en la sangre.

Durante todo el proceso se llevan a cabo los controles de calidad necesarios para asegurar la validez, la repetitividad y la reproducibilidad de los resultados analíticos obtenidos.

1.2. Justificación del proyecto de informatización

Como se expone en el apartado anterior, la empresa necesita comunicar por una parte los servicios que presta, y por otra, los resultados de los análisis de sangre realizados a los usuarios y/o a los prescriptores o a los sistemas de salud implicados en las campañas de detección/prevención.

Debido a que el proceso de introducción de contenido al repositorio es dificultoso, y que el diseño de la web ha quedado obsoleto, ha surgido la necesidad de actualizar la página web y el sistema de gestión de datos con un gestor de contenidos eficiente y cómodo, que permita organizar todo el contenido según sus necesidades.

Para alcanzar este objetivo se decidió crear un proyecto, denominado “informatización DBSS”, que pretende dar soporte a los usuarios públicos, que deseen disponer de la información y contenido asociados a la web de Dry Blood Spot Screening, y a los usuarios registrados, el acceso a su aplicación. En la Figura 1.6. se esquematiza las etapas de este proyecto y su distribución en el tiempo de forma condensada.

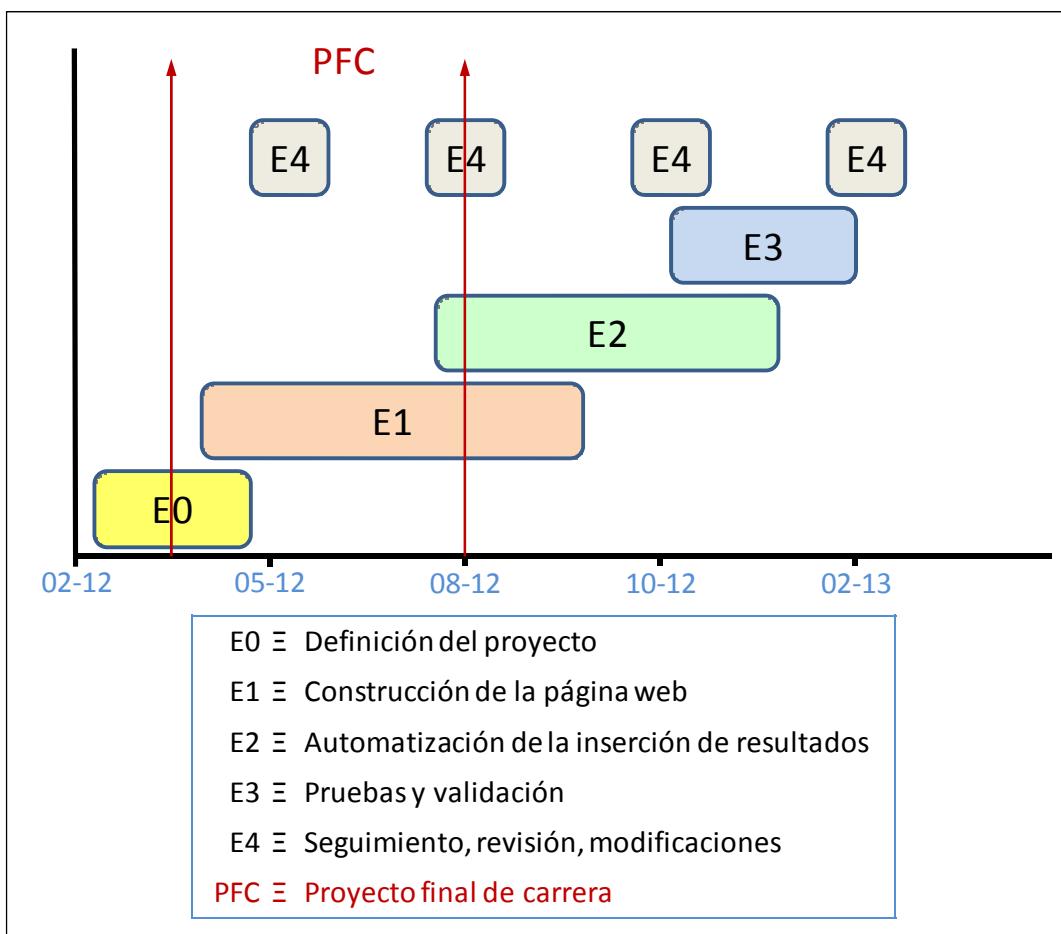


Figura 1.6. Etapas y evolución del proyecto “informatización DBSS”

Dada la importancia del proyecto, en el equipo tienen representación activa todos los departamentos implicados de la misma.

La primera etapa es la propia definición del proyecto. En ella se han determinado las necesidades y se plantearon las distintas etapas, definiéndose, para cada una de ellas, los objetivos específicos, las tareas, los tiempos de ejecución y los responsables. Asimismo, al tratarse de un proyecto con múltiples interrelaciones se plantearon los hitos y las reuniones de intercambio de información y coordinación necesarias. La responsabilidad del proyecto la tomó el departamento de informática.

Cabe destacar, que la etapa E2 incluye un programa de transferencia de los resultados obtenidos a partir del espectrofotómetro utilizado para determinar las concentraciones de los parámetros bioquímicos. Asimismo ese programa incorpora todas las funcionalidades de la web.

Por último, debe tenerse en cuenta que dada la naturaleza de los datos con los que se trabaja se deben respetar todas las Leyes de Protección de Datos Personales existentes.

2. OBJETIVOS, ALCANCE Y PLANIFICACIÓN

2. OBJETIVOS, ALCANCE Y PLANIFICACIÓN

Como ya se ha mencionado en la introducción de esta memoria, resulta imprescindible para conseguir la sostenibilidad de la propia empresa, la implantación de una página web con una doble finalidad: de marketing de sus servicios por una parte y de comunicación de los resultados de los análisis de sangre a los beneficiarios o prescriptores por otra. Es por ello, por lo que se inició el proyecto “Informatización DBSS”, que se ha esquematizado en la Figura 1.6.

Es dentro de este contexto en el que se enmarca el presente proyecto final de carrera (PFC) y más específicamente en la etapa E1 denominada “construcción de la página web”. Como se puede comprobar, engloba parte de la etapa de definición del proyecto así como el principio de la etapa de “automatización de la inserción de resultados”.

2.1. Objetivos

El objetivo general del presente PFC es el estudio, el diseño y la implementación de la página web de la compañía, dando el soporte necesario a los usuarios públicos que deseen disponer de la información y contenido asociados a la web y a los usuarios registrados el acceso a su aplicación o a los resultados de las pruebas analíticas realizadas. Para ello se han planteado los siguientes objetivos específicos:

- Diseño de la propia página web y de las bases de datos asociadas, así como su implantación.
- Tener un sistema de autogestión de contenido intuitivo, fácil de usar y eficiente que permita a cualquier usuario utilizarlo sin problemas.
- La web debe ser intuitiva para los usuarios, que en todo momento sepan donde se encuentran, qué pueden hacer, donde pueden ir; eso, de manera fácil sencilla y eficaz.
- Minimizar tareas repetitivas y mecánicas, como por ejemplo, la elaboración de la carta con los resultados que se envía al paciente y que en la actualidad se lleva a cabo introduciendo manualmente los datos en una plantilla de Word.

Los beneficios esperados son:

- Una mejora en la introducción de los datos personales de los pacientes, así como una navegación por la web cómoda e intuitiva con la información claramente estructurada, agrupada y clasificada.
- Una forma sencilla y segura de comunicación de los resultados de los análisis sanguíneos a las personas y/o a sus prescriptores, asegurando en todo momento tanto la trazabilidad como la protección de los datos personales.

Desde el punto de vista personal, los objetivos planteados son:

- Poner en práctica los conceptos aprendidos durante la carrera.
- Aprendizaje y uso de herramientas de desarrollo de software.
- Colaborar con una empresa de elevado contenido social, ayudando indirectamente a la mejora de la salud a nivel global.

2.2. Alcance

El presente PFC se inició en Marzo de 2012 con la determinación de las necesidades planteadas por la empresa que se han concretizado en la determinación de los requisitos necesarios. En la Figura 2.1. se muestra un esquema condensado de la planificación del proyecto, en el que se pueden ver los bloques de actividades a realizar para alcanzar los objetivos planteados.

A partir de estos requisitos se han de seleccionar las tecnologías más adecuadas a utilizar en el diseño de la web y de las bases de datos asociadas.

Una vez realizada la programación de la web y de las bases de datos, se ha llevar a cabo su implementación. Se deberán realizar las pruebas de funcionamiento (también con casos reales) para su validación.

Cabe destacar que se ha de verificar que el sistema funciona según lo establecido poniendo especial atención a la seguridad en la protección de datos (recuérdese que se está tratando con datos personales de salud).

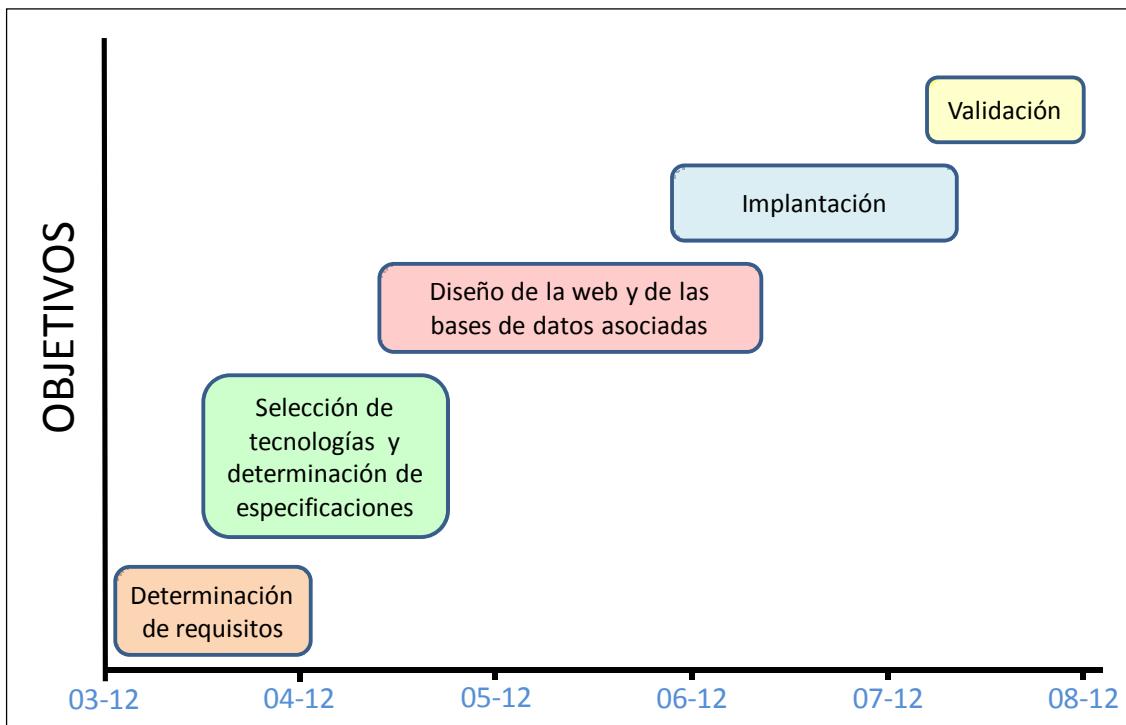


Figura 2.1. Etapas y alcance del PFC

2.3. Planificación

Para poder cumplir con los objetivos específicos planteados para el presente proyecto de parametrización del CMS, se han establecido los hitos, las actividades y las tareas necesarias para su consecución.

De este modo, en la Tabla 2.1. se muestran las tareas planificadas a realizar, así como el tiempo estimado de trabajo personal (se ha utilizado el programa MS Project para su cálculo). En ella se puede observar que se han considerado seis actividades principales, cada una de ellas con las tareas y subtareas necesarias para lograr su consecución.

Cabe señalar, que esta planificación se ha determinado teniendo en cuenta una sucesión óptima de todas las tareas, es decir, sin considerar posibles desviaciones, imprevistos o incidencias que pudieran ocurrir.

	Duration	Task Name
1	238,5 hrs	□ Proyecto Final de Carrera
2	17 hrs	□ Planificación
3	7 hrs	Identificar tareas
4	10 hrs	Establecer metodología de trabajo
5	40 hrs	□ Estudio previo
6	15 hrs	Estudio del escenario
7	25 hrs	Estudio de la tecnología y entorno de programación
8	20 hrs	□ Objetivos y Alcance
9	20 hrs	Definición de objetivos
10	20 hrs	Establecer alcance del proyecto
11	30 hrs	□ Análisis funcional
12	20 hrs	Requisitos funcionales
13	20 hrs	Requisitos no funcionales
14	10 hrs	Casos de uso
15	125 hrs	□ Diseño y parametrización de Drupal
16	110 hrs	□ Drupal
17	10 hrs	Instalación Drupal
18	100 hrs	Modulos
19	100 hrs	Gestión de usuarios
20	115 hrs	□ Diseño
21	50 hrs	Diseño del modelo de datos
22	15 hrs	Diseño del cuerpo de la aplicación
23	50 hrs	Diseño de procesos
24	50 hrs	Configuración web
25	50 hrs	Recuperación
26	50 hrs	□ Documentación
27	50 hrs	Memoria
28	4 hrs	Revisión

Tabla 2.1. Planificación de las tareas

Los puntos críticos desde el punto de vista del proyecto son: la selección del CMS y la estructura de los datos.

Por otra parte, en la Figura 2.2. se esquematiza el diagrama de Gantt del proyecto, en el que muestra el cronograma de las actividades y tareas a realizar, explicitándose la duración de cada una de ellas.

Cabe señalar que el tiempo total estimado para la realización del proyecto es de 4,5 meses.

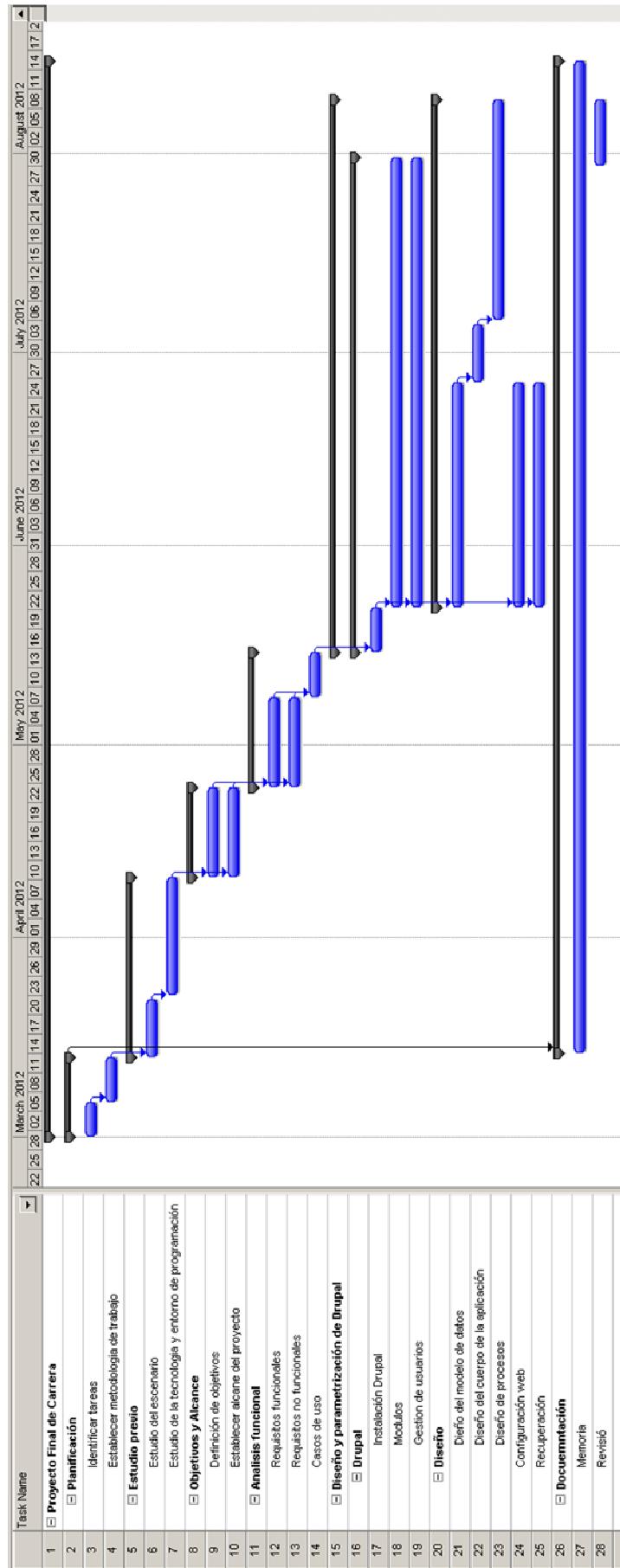


Figura 2.2. Diagrama de Gantt del PFC

2.4. Estimación de costes

A continuación, se hace una valoración económica del proyecto. Para ello se ha llevado a cabo una estimación de los costes en que se ha incurrido para la realización del PFC, separándolos entre costes materiales y costes de personal.

Para el cálculo de los costes materiales se han tenido en cuenta los costes de amortización del PC y los costes de las licencias del software utilizado. En la Tabla 2.2. se detallan estos costes.

RECURSO	CANTIDAD (unidades)	COSTE
PC	1	40,00€
Licencias		
Windows XP Professional	1	150,00€
Ubuntu	1	
Microsoft Office	1	
TOTAL		190,00 €

Tabla 2.2. Estimación de los costes materiales

Para el cálculo de los costes de personal, se han estimado las horas de trabajo de un técnico programador para llevar a cabo cada una de las actividades que conforman la planificación. Para el cálculo de este coste se ha supuesto que el precio promedio por hora de trabajo ha sido de 18 €/h. Los resultados obtenidos se muestran en la Tabla 2.3.

RECURSO	CANTIDAD (h)	COSTE
Planificación	12,0 h	216,00 €
Estudio previo	25,0 h	450,00 €
Objetivos y Alcance	15,0 h	270,00 €
Análisis funcional	31,5 h	567,00 €
Diseño y Parametrización	115,0 h	2.070,00 €
Documentación	40,0 h	720,00 €
TOTAL	238,5 h	4.311,00 €

Tabla 2.3. Estimación de los costes de personal

A partir de estos valores se puede estimar el coste total del PFC. En la Tabla 2.4. se resumen los costes mencionados.

COSTES ESTIMADOS DEL PROYECTO	
Materiales	190, 00 €
Personal	4.311,00 €
TOTAL	4.501,00 €

Tabla 2.4. Estimación de costes del PFC

3. ANÁLISIS FUNCIONAL

3. ANÁLISIS FUNCIONAL

3.1. Análisis de requerimientos

El análisis de requerimientos consiste en la determinación de una serie de condiciones a satisfacer por el software que se debe desarrollar. En este apartado se hará una descripción del proyecto de carácter general, con sus funcionalidades principales y los usuarios que la conformarán.

3.1.1. Descripción de la aplicación

La aplicación a desarrollar se basa en un repositorio de eventos pasados y anuncio de eventos futuros, algunos de ellos con material asociado, como por ejemplo: documentos de apoyo, diapositivas, vídeo, etc. La web dispondrá también de una sección de noticias, suscripción de RSS, así como diferentes vistas para filtrar los eventos.

Se requiere también que la web sea compatible con la Base de Datos que se ha desarrollado para la siguiente etapa del proyecto y que, al mismo tiempo, sea capaz de extraer información de ella para mostrarla en la página web.

Para poder controlar de una forma estructurada y coherente todas las herramientas que ofrece la aplicación, se han definido diferentes roles/perfils para los usuarios de ésta, que determinará el grado de acceso para realizar acciones sobre la aplicación.

Los roles/perfils que se han definido para dicho propósito son:

- usuario no registrado
- usuario registrado
- organizador
- administrador

Los usuarios no registrados sólo podrán acceder al contenido público de la web. El contenido privado estará asociado a diferentes grupos (previamente creados por los organizadores), que determinará el grado de privacidad del contenido adjunto. El tipo de grupo y sus restricciones determinarán si el usuario puede acceder al contenido o no.

El usuario registrado podrá solicitar la afiliación al grupo que le interese para ver los eventos y material adjunto a dicho grupo.

Los administradores de los grupos serán quienes decidan a qué categoría pertenecen los grupos, es decir, que restricciones tendrán éstos, lo que determinará los permisos que tendrán los usuarios para acceder al contenido de dichos grupos.

Los administradores de los grupos son los usuarios con el rol de organizador. Es el usuario que se encarga de crear el contenido de la web (eventos, noticias, grupos) y asignar los permisos de acceso al evento y a su contenido. Este acceso se puede definir como público o privado. Si es privado, el organizador deberá asignar el evento a uno de los grupos ya existentes.

Como administrador de la web se utilizará el rol de administrador, el cual podrá gestionar todos los usuarios, todo el contenido y ficheros asociados, podrá modificar los aspectos visuales y estructurales de la web, etc.

En los apartados que vienen a continuación se establecen de forma detallada todas las características y funcionalidades que tendrá la aplicación. Se especifica el formato que se le dará a la misma, así como la distribución de espacio, estructuración de usuarios y las funcionalidades con las que se les dotará.

3.2. Requisitos funcionales

Teniendo en cuenta la descripción del apartado anterior se obtienen las necesidades que se describen a continuación, clasificadas según su naturaleza.

3.2.1. Roles/Perfiles

Para poder gestionar todo el contenido de la web y controlar el acceso al mismo, ha sido necesario determinar qué usuarios podrán crear contenido y quiénes lo podrán ver.

De ese modo, se han definido 4 roles para determinar los diferentes grados de acceso e interacción a la aplicación. En la Tabla 3.1. se muestran esos roles conjuntamente con el grado de acceso y de interacción para cada uno de ellos.

Rol	Grado de acceso e interacción
Anonymous user	Este rol de usuario ya viene definido por Drupal, que representa el usuario que navega por la aplicación sin haberse registrado o identificado. Podrá ver todo el contenido de la web que haya sido definido como público.
Authenticated user	Para adquirir este rol, el usuario anónimo deberá registrarse en la aplicación. Una vez registrado, el usuario podrá acceder a su contenido.
Organizer	Es el usuario que se encarga de crear y gestionar los eventos y las noticias que aparecen en la web. También puede crear grupos, asignar contenido a esos grupos (eventos y material asociado) y determinar qué usuarios podrán acceder a ese contenido.
Administrator	<p>Como administrador de la web se ha definido este rol para poder gestionar todo el contenido de ésta, la alta, la baja o la modificación de eventos, noticias y grupos.</p> <ul style="list-style-type: none"> • Usuarios: alta, baja, modificación de perfiles y asignación de roles. • Aspecto visual: Podrá modificar los estilos de la aplicación y controlar diferentes aspectos estructurales de la web (disposición de bloques, enlaces, primary links, etc.).

Tabla 3.1. Roles, grado de acceso e interacción con la aplicación

Un usuario *Anonymous User* no puede registrarse en la aplicación, ya que el apartado privado de la misma es información a la que sólo los empleados de la empresa pueden acceder.

Cuando un usuario accede a la aplicación le aparecerá, en función de su rol, la información de su perfil. En el caso del *Authenticated User*, le aparecerán los datos básicos de su registro. Para ver los grupos a los que está afiliado deberá ir a la sección “*My Subscriptions*” en el bloque inferior izquierdo de la web.

En el caso de los *Organizers* y *Administrators*, tendrán la opción de visualizar, además de su información personal, los grupos de los cuales son administradores, para posteriormente modificarlos, añadirles contenido o bien gestionar sus afiliados.

Cualquier usuario registrado en la aplicación tiene permisos para cambiar los datos básicos de su perfil.

3.2.2. Contenido

Casi todo el contenido de Drupal se almacena como un nodo. Un nodo se puede definir como un objeto de contenido, al que se puede acceder siempre mediante una url fija (permalink).

Drupal permite crear y definir tipos de contenido propios. Por defecto, la instalación básica de Drupal, sólo permite crear 3 tipos de contenido básico: Page, Story y Poll. Para el caso de esta aplicación se reutilizará el objeto Page para definirlo como noticia (new).

Mediante la instalación de un módulo específico, se puede crear nuevos tipos de contenido con los atributos que se quiera definir para ellos. Los usuarios autorizados a publicar, modificar o eliminar cualquier tipo de contenido son el *Administrator* y *Organizer*.

A continuación se detallan los tipos de contenido definidos para esta aplicación:

– Evento (Event)

Un evento es uno de los tipos de contenido más utilizados en esta web. En el formulario de alta de un evento deben aparecer los siguientes campos:

- *Start Date*: Fecha de inicio del evento.
- *End Date*: Fecha de finalización del evento.
- *Repeat Event options*: Opciones de repetición del evento.
- *Topic*: Título representativo del evento.
- *Tags*: Son palabras clave asociadas al evento. Son una ayuda para clasificar o filtrar el contenido de un mismo tipo de objeto, en este caso, los eventos.
- *Abstract*: Es el cuerpo del evento. Contiene la descripción completa del mismo; se expone la temática de la que se hablará en el evento.
- *Sponsor*: Entidad que esponsoriza el evento.
- *Location*: Ubicación física del evento.

– Noticia (New)

Se ha utilizado el tipo de contenido page (predefinido por Drupal) para crear este objeto. Las noticias serán accesibles por todos los usuarios de la aplicación, ya

que se ha definido el tipo de contenido como público. Los atributos del objeto son:

- *Title*: Título de la noticia
- *Body*: Descripción de la noticia.

– **Grupo (Group)**

Un grupo no es exactamente un tipo de contenido, sino que más bien se puede definir como un contenedor de uno o más objetos, tales como eventos, noticias u otro tipo de contenido. En la aplicación se ha definido el evento como único tipo de contenido para los grupos. Los atributos del grupo son:

- *Group Name*: Nombre del grupo.
- *Description*: Descripción del grupo.
- *Membership Request*: Proceso de afiliación al grupo (detallado en el siguiente apartado).
- *Registration form (checkbox)*: se deberá marcar la casilla si se quiere que se muestre la opción de solicitud de afiliación al grupo en el formulario de alta de usuario a la aplicación.
- *List in groups directory (checkbox)*: Marcando la casilla se habilita la visibilidad del grupo en el directorio de grupos (Group Directory).
- *Private Group (checkbox)*: Esta opción permite mantener el grupo totalmente privado e inaccesible por parte del usuario. El administrador del grupo deberá especificar los usuarios afiliados al grupo.

– **Tag**

Es un atributo que se le puede añadir a un objeto o tipo de contenido, que permitirá, mediante la activación del módulo Taxonomy, clasificar o catalogar el contenido mediante palabras clave.

Esta descripción se encuentra en este apartado porque tanto el *Organizer* como el *Administrator* podrán crear tags independientemente de los eventos, como si de un tipo de contenido se tratara.

En la mayoría de los campos de cada objeto se puede definir el formato de entrada del texto, para poder determinar si el texto introducido es: HTML filtrado, HTML puro o PHP. Ello permitirá obtener una mayor flexibilidad y versatilidad cuando se desee darle formato al texto.

En cada tipo de contenido es posible asociar ficheros adjuntos al nodo, tal como presentaciones, pdf's, vídeos, etc.

En el apartado 4.9.1. se detallará cómo funciona el sistema o proceso de upload y la asociación de ficheros a los nodos.

Existe un campo común para los tipos evento y noticia, que se denomina *published*. Al marcar la casilla se definirá el nodo como publicado y será visible para todos los usuarios que tengan acceso a él. Si por lo contrario, se pretende mantener el contenido oculto, no se marcará la casilla mencionada. En este caso, únicamente los administradores de la aplicación y el creador del nodo podrán acceder a él.

3.2.3. Gestión de Grupos

Como se ha mencionado en el apartado anterior, un grupo es un contenedor de objetos de tipo evento. Sólo los administradores y organizadores de la aplicación tendrán permisos para crearlos.

Se puede definir como grupo a un conjunto de eventos con un elemento en común, que podría ser, una temática, una campaña de prevención, unos resultados analíticos de un parámetro bioquímico, los datos del paciente, un centro de salud, una escuela, una entidad pública, una empresa, etc.

Del mismo modo que para los eventos también se puede adjuntar material al grupo, sin necesidad de crear un evento para dicho propósito. Un ejemplo sería información adicional del grupo, documentación común para todos los eventos del grupo, etc.

Para poder hacer la solicitud a un grupo, el usuario deberá estar registrado previamente en la aplicación.

Cuando un usuario (administrador u organizador) da de alta un grupo, deberá definir (además del nombre, descripción, etc.) qué tipo de restricción de acceso va a asignar al grupo. En función de la opción seleccionada se determinará el proceso de afiliación que deberá seguir el usuario para poder acceder al grupo y a su contenido adjunto. En la Tabla 3.2. se detallan los distintos tipos de proceso de afiliación conjuntamente con sus restricciones.

Tipo	Restricciones de afiliación
Open	La solicitud de afiliación al grupo de un usuario será aceptada automáticamente. El usuario podrá ver todo el contenido asociado al grupo. Se puede definir esta restricción como pública.
Moderated	Cuando un usuario hace la petición de afiliación deberá esperarse a que el administrador del grupo acepte la solicitud. El administrador del grupo podrá decir si decide aceptar la solicitud o rechazarla, que en el segundo caso, prohibiría al usuario el acceso al contenido del grupo.
Invited only	Los miembros deben ser introducidos por el administrador del grupo.
Closed	El grupo está totalmente cerrado. Ningún usuario de la aplicación excepto el administrador del grupo podrá conocer su existencia, e interactuar con él.

Tabla 3.2. Tipos de restricción de afiliación

El administrador del grupo será el usuario que lo haya creado, teniendo posteriormente la posibilidad de asignar nuevos usuarios como administradores adicionales del grupo. Si se quiere definir un usuario como nuevo administrador adicional, éste deberá estar afiliado al grupo previamente. Una vez cumplido el requerimiento, el administrador principal podrá (a través de la aplicación) asignar este rol al usuario en cuestión.

– [Grupos y Eventos](#)

Los grupos permiten agrupar o clasificar el contenido de la web, además de controlar el acceso a él. Cuando un evento es dado de alta, éste se define por defecto como público, lo que permitirá a todos los usuarios de la aplicación ver la información del nodo. Además de público, el evento puede ser asociado a un grupo.

En el caso de que el evento sea público y también esté afiliado a un grupo, aunque el nivel de restricción de afiliación de dicho grupo fuera muy alto, cualquier usuario de la aplicación podría ver el contenido del evento. Si el evento no está definido como público, la aplicación exige que se asigne el evento a un grupo. En este momento el evento será únicamente visible por los afiliados a dicho grupo.

Para terminar, cabe señalar que cuando se accede al contenido de un grupo, la aplicación muestra todos los eventos asociados a este grupo, ordenados por fecha, empezando por los más recientes hasta los más antiguos.

3.2.4. Vistas (Views)

Drupal permite definir, mediante la instalación y aplicación de algunos módulos, qué contenido se quiere mostrar y cómo se quiere mostrar, pudiéndolo estructurar y organizar según las necesidades.

Cada una de las vistas será personalizada para cada rol de la aplicación. Según los permisos del usuario que está solicitando la vista, se podrá interactuar con mayor o menor medida con la aplicación. Por ejemplo, según el usuario que haga la petición para ver la lista de grupos le aparecerán unas opciones u otras. En el caso del organizador le aparecerán botones de creación de nuevos grupos y modificación de existentes, y en cambio, al usuario autenticado le aparecerán los botones de afiliación a los grupos entre otras opciones. En el apartado 4.8.5. se describe con detalle cuál es el proceso de creación de las vistas.

3.2.5. Secciones/Estructura de la web

Para satisfacer las necesidades del cliente, se ha diseñado una estructura genérica en la que se basará la vista principal de la aplicación para todos los usuarios. En el momento que un usuario se registre, se le mostrarán los bloques específicos de las funcionalidades asignadas para su rol. La Figura 3.1. muestra la estructura visual que se ha definido para la aplicación.

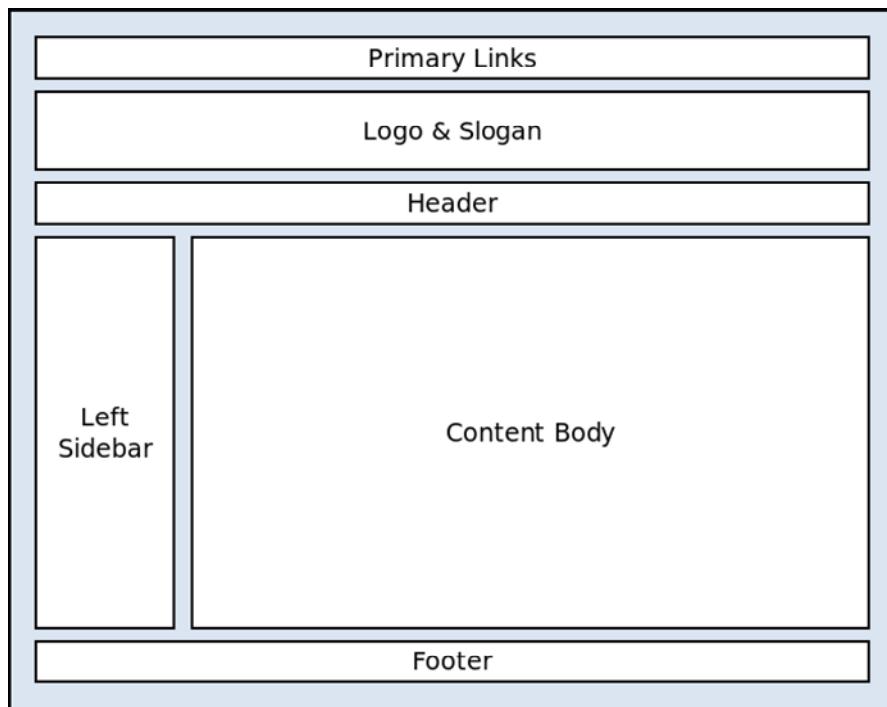


Figura 3.1. Esquema de la estructura de la página web

A continuación se definirán las diferentes partes que componen la estructura de la web y las funcionalidades específicas para cada uno de los roles.

– **Primary Links**

En esta parte se muestra un grupo de links considerados de los más importantes/funcionales de la web.

- *Usuario no registrado:* Los links que aparecen son:
 - *HOME:* Muestra la página principal de la web.
 - *News:* Muestra las noticias de la web.
 - *LogIn:* Formulario de registro a la aplicación.
- *Demás usuarios:* Los 3 roles restantes verán una pequeña variación respecto al usuario no autenticado. Los links son:
 - *HOME:* Página principal.
 - *News:* Lista de noticias.
 - *My Account:* Muestra los datos de perfil del usuario, tales como los grupos afiliados del usuario, los grupos de los cuales se es administrador e historial del usuario, entre otros. También se le permite al usuario cambiar el correo electrónico definido inicialmente y la contraseña.
 - *LogOut:* Cierra la cookie de nuestra sesión.

– **Logo & Eslogan:**

Esta parte de la web es fija para todas las páginas. Siempre se muestra el mismo contenido, en este caso el logo de la web y su eslogan.

– **Header:**

Sección fija para todas las páginas. Se puede definir un texto explicativo, aunque en este caso se ha dejado en blanco.

– **Left Sidebar:**

Esta parte está formada por bloques. Los bloques son contenedores que permiten mostrar una determinada información que puede ser fija (por ejemplo links a otras páginas), o bien puede ser variable, es decir, información que se genera dinámicamente cuando se renderiza el contenido del bloque en cuestión.

En la parte izquierda de la web se han definido 4 bloques diferentes:

- *Upcoming Campaign*: Bloque dinámico visible para todos los usuarios. Se mostrarán los eventos futuros (título y días restantes para la campaña) y los eventos activos con su título y el tag “now”, indicando que el evento se está impartiendo en ese preciso momento.
Si el evento es público todos los usuarios lo verán expuesto en el bloque, pero si el evento está asociado a un grupo privado, sólo los usuarios registrados en la aplicación y afiliados al grupo lo verán en el bloque.
- *User*: Este bloque está dedicado únicamente a los usuarios con el rol *Organizer* y *Administrator*. Aquí aparecen las opciones que dispone el usuario para crear contenido y administrarlo. En el caso del *Administrator* dispondrá también de opciones de administración de la web.
- *Site navigation*: Bloque de navegación visible para todos los usuarios. Si el usuario está registrado le aparecerá algún elemento más en el bloque. Permite buscar cualquier tipo de contenido de cualquier parte de la web. Está compuesto por diferentes links, que son: lista de grupos, diferentes vistas para filtrar los eventos, grupos a los que se está afiliado y búsqueda de tags.
- *Related Links*: Este bloque es visible e idéntico para todos los usuarios, es totalmente estático. Contiene links externos a páginas “amigas” o relacionadas con la web.

– **Content Body:**

En esta sección se muestra el contenido que el usuario ha solicitado mediante su petición. La generación de este contenido es totalmente dinámica. Cuando el usuario accede a cualquier página de la web, el navegador hace una petición a la aplicación, lo que implica que ésta haga la consulta a la base de datos para poder ofrecer el contenido solicitado. Previamente se habrá comprobado que el usuario tenga acceso a dicho contenido.

– **Footer:**

Sección fija visible para todos los usuarios. El contenido de esta sección será común en todas las páginas.

Por otra parte cabe remarcar que el “Right Sidebar” ha sido inhabilitado para engrandecer el espacio para el *content body*. En este proyecto se había propuesto inicialmente mostrar un calendario de eventos en la parte derecha, pero debido a exigencias del cliente se ha desestimado dicha opción y se ha dejado como oculta.

3.2.6. Gestión de ficheros

La gestión de ficheros es muy importante en esta aplicación, ya que se requiere poder controlar el acceso a todo el contenido adjunto a los eventos. Además, se pretende hacer el upload de ficheros con un tamaño superior a 500Mb, por lo que es necesario buscar una alternativa al upload por http.

Por un lado existe la limitación del servicio PHP respecto al tamaño de los ficheros que se quiere subir, y por otra parte el upload de ficheros grandes por http sería muy lento.

Como solución rápida y sólida se ha decidido que el usuario realizará el upload de ficheros por ftp, y posteriormente pueda anadir los ficheros al nodo que desee.

Para realizar esta acción se necesitará instalar un módulo específico que ofrecerá una API (Application Programming Interface) implementada en AJAX (*Asynchronous JavaScript And XML*), que permitirá insertar en la base de datos los ficheros subidos al servidor, y posteriormente adjuntarlos al nodo que se desee, definiendo el grado de restricción de acceso al mismo por parte del usuario.

Estas acciones se pueden realizar mediante el procedimiento descrito en el apartado 4.9.1. de la presente memoria.

3.2.7. Administración

El área de administración será únicamente accesible por los usuarios con el rol *adminnistrator* o bien, por el *super-administrador* de la aplicación.

Los servicios que ofrece esta sección son: la gestión total de los usuarios, la gestión del contenido, la modificación de los aspectos funcionales y visuales de la web, entre otros.

A continuación se describen las funcionalidades que se han requerido para esta sección:

– Gestión del Contenido

- *Creación de contenido*: Posibilidad de crear eventos, noticias o grupos.
- *Gestión Categorías*: Permitirá dar de alta categorías (tags) y modificar las existentes.
- *Content Templates*: Permite crear un template personalizado para el RSS.
- *Publicación RSS*: Configuración general del RSS.
- *Contenido*: Ver, editar y eliminar contenido de la web.

– Construcción de la web

- *Bloques*: Configuración de los bloques de contenido que aparecen en los laterales de la web.
- *Menús*: Creación, edición, eliminación y reorganización de los primary links y menús.
- *Módulos*: Habilitación o inhabilitación de los módulos de la aplicación.
- *Temas*: Edición del aspecto visual general de la web.
- *Vistas*: Creación, edición y eliminación de vistas.

– Grupos Orgánicos

- *Configuración de grupos*.
- *Configuración del acceso a los grupos*.

– Configuración de la web

- *Tema de administración*: Modificación del tema de la aplicación de administración.
- *Clean URL's*: Configuración de url's amigables.
- *Preferencias Content Templates*: Configuración específica del módulo Content Templates.
- *Fecha y hora*: Configuración de la zona horaria, hora y fecha, y formato de la misma.

- *Error reporting*: Configuración del redireccionamiento en caso de errores 403 (access denied) y 404 (page not found).
- *Configuración eventos*: Configuración específica para del módulo que gestiona los eventos.
- *File System*: Configuración del sistema de ficheros. Las dos opciones que ofrece Drupal son Public o Private. En este caso se ha marcado la segunda opción.
- *File Uploads*: Restricción de ficheros según la extensión, tamaño y usuario.
- *Google Analytics*: Configuración de los parámetros específicos para el tracking de páginas mediante la aplicación Google Analytics.
- *Kit de herramientas para las imágenes*: Selección de la calidad de las imágenes. Un valor porcentual definirá este campo.
- *Formatos de entrada*: Configuración del formato de entrada del contenido introducido por el usuario.
- *Caching*: Activación o desactivación de la caché de la web.
- *Configuración repetición de eventos*: Configuración específica del módulo Repeat Events.
- *Información de la web*: Configuración del nombre de la web, configuración de la dirección de correo utilizada como remitente de los correos electrónicos enviados por la aplicación, edición del eslogan y footer de la web y configuración de la página principal por defecto.
- *Site Maintenance*: Configuración del estado de la web. Existen dos modos posibles, “Online” (la web está funcionando),y “OffLine”, que desactiva la web mostrando un único mensaje al usuario.
- *Lista de Taxonomías*: Configuración específica del módulo Taxonomy.

– **Gestión de usuarios**

- *Control de acceso*: Esta sección es muy importante ya en ella se determinará el acceso a las funcionalidades de la aplicación para cada uno de los roles determinados.
- *Reglas de acceso*: Se podrá deshabilitar usernames, direcciones de correo y direcciones IP.

- *Roles*: Creación, modificación o eliminación de roles.
- *Usuarios*: Creación, modificación o eliminación de usuarios. Permitirá también asignar roles a los usuarios.
- *Preferencias de usuario*: Permite configurar el comportamiento por defecto de los usuarios, incluyendo los requerimientos de registro, correo electrónico y foto de usuario.

– [Logs del sistema](#)

Drupal dispone de sus propios gestores de logs, y aunque se utilice Google Analytics para recoger esta información, cabe destacar que puede dar datos interesantes.

- *Recent hits*: Últimas páginas visitadas.
- *Recent log entries*: Log de las entradas más recientes
- *Top 'access denied' errors*: Muestra los últimos errores de acceso denegado.
- *Top 'page not found' errors*: Muestra los últimos errores de página no encontrada.
- *Top pages*: Páginas con más visitas.
- *Top visitors*: Usuarios con más páginas visitadas.
- *Access log settings*: Configuración de logs.
- *Status report*: Reporte de errores detectados en la web.

3.2.8. RSS

En este proyecto se requiere, como una de las funcionalidades, la posibilidad de poder suscribirse a la web mediante RSS. Si un usuario está suscrito a esta aplicación, recibirá una notificación cada vez que se cree un nuevo evento.

De esta forma, el usuario podrá estar informado en todo momento de los nuevos eventos publicados en la web.

3.2.9. Logs del sistema

El sistema de logs de la aplicación sólo podrá ser consultado por el usuario con el rol de *administrator*. Se podrá consultar toda la información referente a número de accesos por página, accesos totales, estado de los servicios del servidor, etc., tal y como se ha descrito en el apartado 3.2.7.

Para complementar esta información y poder hacer un análisis más exhaustivo del usuario que accede a la aplicación, se podrán consultar (externamente a la aplicación) los logs del sistema a través de *Google Analytics*.

Se pueden obtener informes como el seguimiento de usuarios exclusivos, el rendimiento del segmento de usuarios, el rendimiento del contenido, el análisis de navegación, los objetivos y proceso de redireccionamiento o los parámetros de diseño web. Esto permitirá saber, entre otras cosas, qué sistemas operativos, navegadores, versión de navegador, etc., utilizan los usuarios de la web.

Se debe integrar un módulo al sistema para generar el script que va incrustado en cada una de las páginas de la web.

3.2.10. Recuperabilidad

En una aplicación donde se almacena una gran cantidad de información, es de vital importancia disponer de un sistema de recuperación de datos que permita restablecer la web en caso de fallida.

Los dos procedimientos que se utilizarán para disponer de este sistema serán el mirroring y el backup.

La aplicación deberá replicarse en otro servidor con el fin de poder redireccionar las peticiones en caso de caída del servidor principal, para normalizar el estado de la web. Se realizará además un backup diario en un tercer servidor, separado físicamente de los dos anteriores, para garantizar una mayor seguridad.

En los apartados 4.9.2., 4.9.3. y 4.9.4. se detalla el proceso de implementación de este sistema.

3.3. Requisitos no funcionales

A diferencia de los requerimientos funcionales, los requerimientos no funcionales definen las calidades que debe cumplir la aplicación al llevar a cabo su función. Para su evaluación en profundidad, se toma como base el estándar ISO/IEC9126-1, el cual permite evaluar la calidad del software. Se utiliza la extensión NT-ISO/IEC 9126 del estándar mencionado para poder evaluar aspectos no técnicos, como por ejemplo, la vertiente económica del software usado.

A continuación, se describen los requerimientos no funcionales de la aplicación agrupados según las características que define el estándar mencionado.

3.3.1. Seguridad

Capacidad del producto software para proteger la información y los datos de forma que las personas o sistemas no autorizados no las puedan leer ni modificar, al mismo tiempo que no se deniega el acceso a personas ni sistemas autorizados.

- La aplicación debe utilizar contraseñas para identificar y autenticar los usuarios.
- La aplicación debe almacenar cifradas las contraseñas de todos los usuarios de la aplicación.

3.3.2. Fiabilidad

Capacidad del producto software de mantener el nivel de prestación bajo condiciones establecidas durante un periodo de tiempo establecido (madurez, recuperabilidad).

- La aplicación debe estar disponible en todo momento en el dominio que DBSS ha decidido hospedar la aplicación.
- Se debe realizar una copia de seguridad de la aplicación diariamente, de forma que en caso de baje del sistema se pueda restaurar la del día anterior.

3.3.3. Usabilidad

Capacidad del producto software para ser entendido, aprendido, usado y ser atractivo para los usuarios finales, cuando se utiliza bajo condiciones determinadas.

- La interfaz gráfica de usuario debe ser desarrollada en inglés, puesto que es el idioma que se ha definido para la aplicación.

- La aplicación debe ser fácil de utilizar y lo más intuitiva posible.
- La aplicación debe disponer de herramientas que permitan administrar los contenidos, los usuarios y la apariencia de la forma más sencilla posible.
- La aplicación deberá respetar, siempre que sea posible, los estándares web establecidos por el W3C respecto a accesibilidad, internacionalización e independencia de dispositivos.

3.3.4. Eficiencia

Capacidad del producto software para proporcionar prestaciones adecuadas, relativas a la cantidad de recursos usados, bajo condiciones determinadas. El tiempo de respuesta de la aplicación debe ser aceptable, se espera una respuesta de un máximo de 2 segundos.

3.3.5. Mantenimiento

La aplicación debe trabajar siempre con versiones estables del software. Las versiones no estables quedan restringidas a aplicaciones en pruebas. La aplicación debe disponer de un procedimiento establecido de actualización.

3.3.6. Portabilidad

Capacidad de un producto software para ser transferido de una plataforma a otra (instalación, reemplazo, adaptabilidad).

- La aplicación debe visualizarse correctamente, como mínimo, en los siguientes navegadores: Internet Explorer (versiones 6 y 7), Mozilla Firefox (versiones 2 y 3) y GoogleChrome, puesto que son los más utilizados entre los usuarios.
- La aplicación debe desarrollarse sobre un servidor Apache, se debe programar en PHP5 y debe utilizar MySQL como sistema gestor de bases de datos.
- La base de datos de la aplicación debe poder administrarse con herramientas específicas para bases de datos, como por ejemplo PHPMyAdmin.
- Sólo se permitirá subir documentos a la aplicación con los siguientes formatos: jpg, jpeg, gif, png, txt, doc, xls, pdf, ppt, pps, odt, ods, odp, mp4, avi, mpeg, mpg y mp3.

3.4. Diagrama casos de uso

Con la finalidad de analizar los posibles comportamientos que podrán llevar a cabo los distintos usuarios que accedan a la aplicación y empezar a definir así qué funcionalidades deberá incorporar la aplicación en sí, se presentan los diagramas de caso de uso pertenecientes a cada uno de los 4 roles de usuario definidos.

3.4.1. Anonymous User

Como se puede apreciar en la Figura 3.2. el comportamiento de este usuario dentro de la aplicación se reducirá a acceder a las diferentes estructuras de contenidos de la aplicación (eventos futuros, pasados y actuales, a las noticias, la lista de grupos y lista de tags).

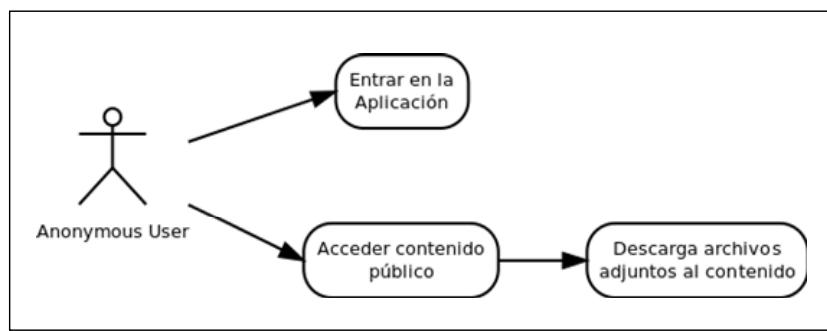


Figura 3.2. Casos de uso Anonymous User

3.4.2. Authenticated User

Este usuario tendrá algún privilegio más que el usuario no autenticado, como por ejemplo, la posibilidad de afiliarse a un grupo y el acceso al contenido de los grupos a los que está afiliado y tendrá la posibilidad de modificar su perfil (ver Figura 3.3.).

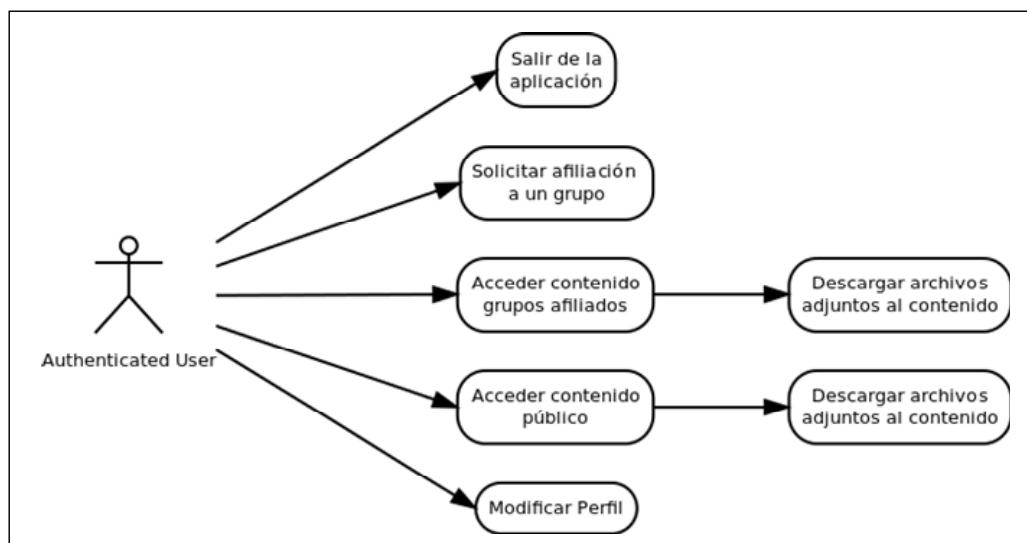


Figura 3.3. Casos de uso Authenticated User

3.4.3. Organizer

El usuario con rol de *organizer* dispone de una mayor cantidad de competencias dentro de la aplicación, pues es el encargado de crear, modificar y eliminar cualquier tipo de contenido y gestionar los grupos que haya creado. Podrá invitar a miembros al grupo, aceptar o rechazar la solicitud de afiliación de los usuarios, entre otras funcionalidades (ver Figura 3.4.).

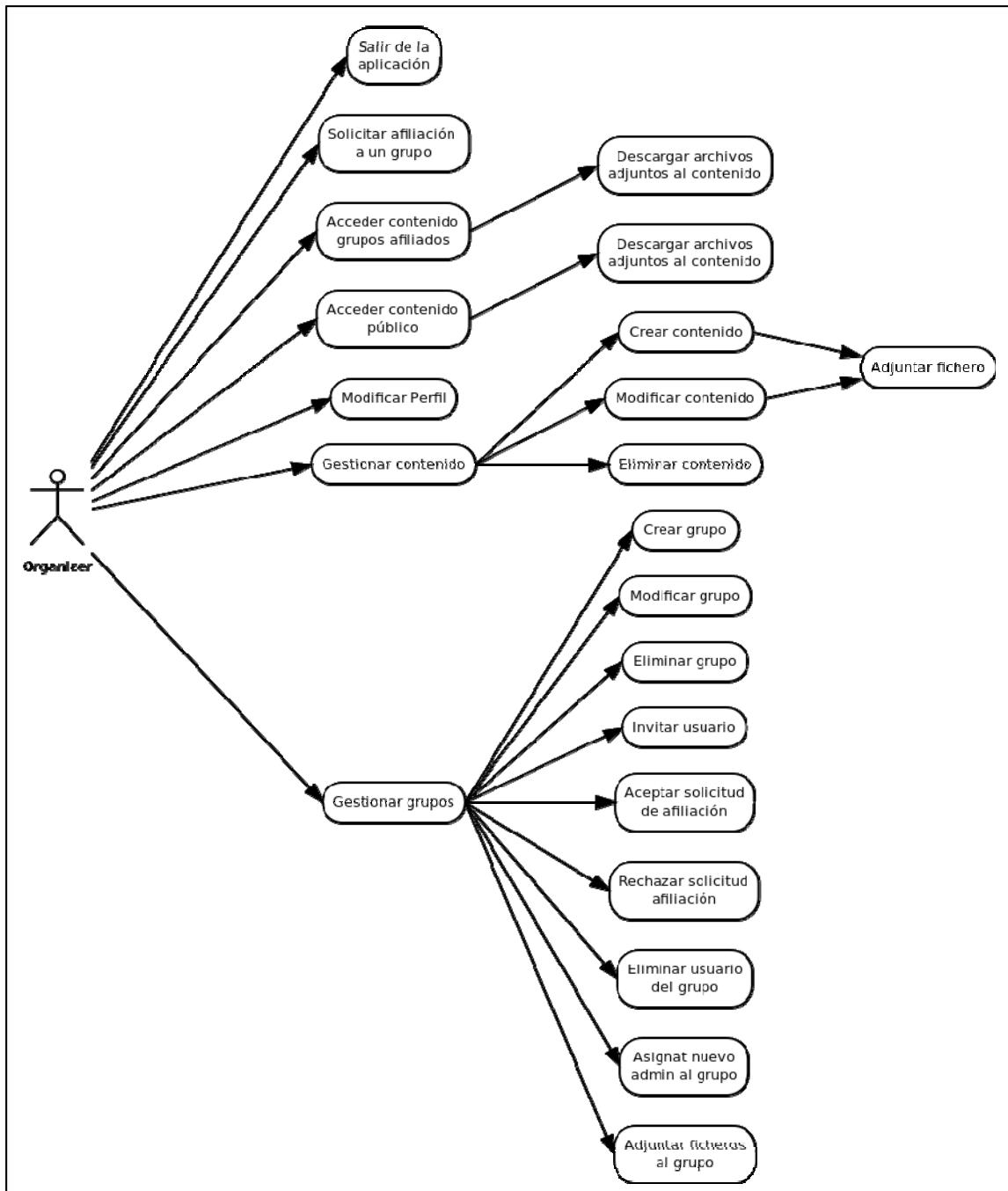


Figura 3.4. Casos de uso Organizer

3.4.4. Administrator

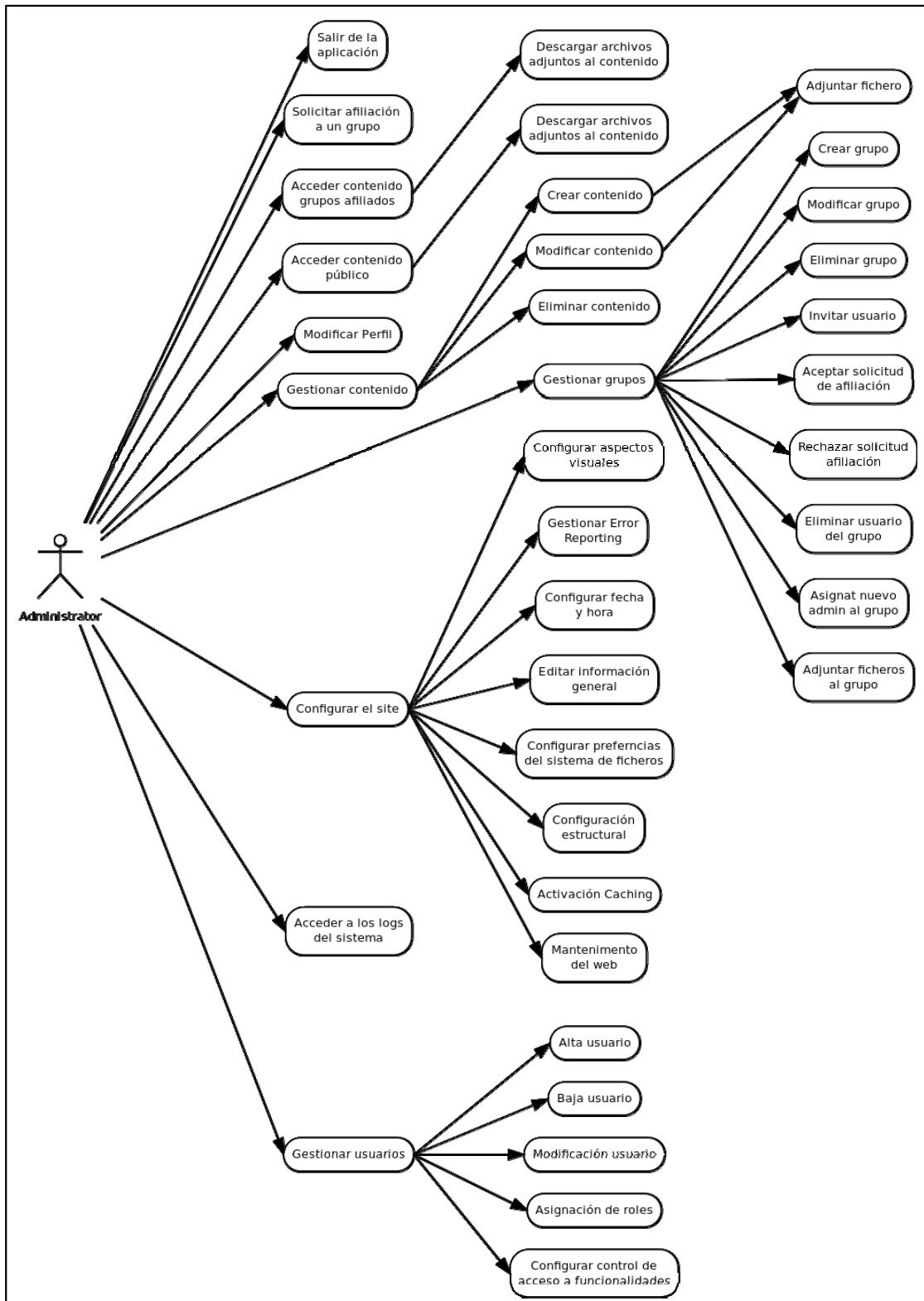


Figura 3.5. Casos de uso Administrator

El administrador tiene las mismas competencias que el *organizer*, pero además tiene la opción de gestionar usuarios y la configuración de la web y revisar los logs del sistema.

3.5. Descripción casos de uso

A continuación se describen los casos de uso que se requieren en la especificación de la aplicación:

– Generales

Caso de uso: Entrar en la aplicación

Perfil: Anónimo

Descripción: Se muestra el formulario de login de la aplicación.

Caso de uso: Acceder al contenido público

Perfil: Todos

Descripción: Se puede acceder al contenido público desde cualquier parte de la web, tanto eventos pasados, futuros y actuales, noticias, lista de grupos, lista de tags y eventos clasificados por tags, links externos, etc.

Caso de uso: Descargar ficheros

Perfil: Todos

Descripción: Se pueden descargar todos los ficheros adjuntos al contenido al cual se tenga acceso.

Caso de uso: Solicitar afiliación a un grupo

Perfil: Autenticado, organizador y administrador

Descripción: Se puede solicitar la afiliación a un grupo. Si el grupo es de libre afiliación se pertenecerá a miembros del grupo automáticamente. Si se necesita supervisión del administrador del grupo, se deberá esperar a que éste acepte.

Caso de uso: Modificar perfil

Perfil: Autenticado, organizador y administrador

Descripción: Cualquier usuario registrado podrá modificar su perfil (username, mail, afiliaciones a los grupos).

Caso de uso: Acceder al contenido de grupo afiliado
Perfil: Autenticado, organizador y administrador
Descripción: Una vez afiliado a un grupo, el usuario podrá acceder al contenido asociado al grupo.

– Gestión contenido

Caso de uso: Crear evento
Perfil: Organizador, administrador
Descripción: El usuario podrá dar de alta un evento mediante el formulario correspondiente.

Caso de uso: Modificar evento
Perfil: Organizador, administrador
Descripción: El usuario creador del evento podrá modificar el evento en cuestión o bien podrá ser modificado por un administrador.

Caso de uso: Eliminar evento
Perfil: Organizador, administrador
Descripción: El usuario creador del evento podrá eliminar el evento en cuestión o bien podrá ser eliminado por un administrador.

Caso de uso: Crear noticia
Perfil: Organizador, administrador
Descripción: El usuario podrá dar de alta una noticia mediante el formulario apropiado.

Caso de uso: Modificar noticia
Perfil: Organizador, administrador
Descripción: El usuario creador de la noticia podrá modificar el nodo en cuestión. También podrá ser editado por un administrador.

Caso de uso: Eliminar noticia
Perfil: Organizador, administrador
Descripción: El usuario creador de la noticia podrá eliminarla. También podrá ser eliminada por un administrador.

Caso de uso: Crear nuevo tipo de contenido
Perfil: Administrador
Descripción: El usuario podrá crear nuevos tipos de contenido, definiendo los campos y atributos del mismo mediante diferentes formularios.

Caso de uso: Modificar tipo de contenido
Perfil: Administrador
Descripción: El usuario podrá modificar los tipos de contenido creados por él, pero no los creados por los demás.

Caso de uso: Eliminar tipo de contenido
Perfil: Administrador
Descripción: Podrá eliminar los tipos de contenido que el mismo usuario ha creado.

Caso de uso: Asignar ficheros al contenido
Perfil: Organizador, administrador
Descripción: Los usuarios podrán subir ficheros y añadirlos al contenido que deseen (evento, noticia, grupo).

– Gestión de grupos

Caso de uso: Crear grupo
Perfil: Organizador, administrador
Descripción: Mediante un formulario el usuario podrá dar de alta un grupo en la aplicación, donde definirá todos sus atributos relacionados.

Caso de uso: Modificar grupo
Perfil: Organizador, administrador
Descripción: El usuario podrá modificar el grupo que haya creado y el contenido asociado a él.

- Caso de uso: Eliminar grupo
Perfil: Organizador, administrador
Descripción: El creador del grupo podrá eliminarlo. Tendrá que decidir si desea mantener el contenido asociado al grupo y asociarlo a otro grupo o bien eliminarlo también.
- Caso de uso: Invitar usuarios
Perfil: Organizador, administrador
Descripción: El creador del grupo podrá invitar directamente usuarios al grupo sin la necesidad de que soliciten la afiliación.
- Caso de uso: Aceptar solicitud de afiliación
Perfil: Organizador, administrador
Descripción: Cuando un usuario requiere la afiliación al grupo, el administrador de éste, podrá aceptar la solicitud o rechazarla.
- Caso de uso: Rechazar solicitud de afiliación
Perfil: Organizador, administrador
Descripción: Si el administrador del grupo decide no admitir a un usuario, puede rechazar su solicitud de afiliación.
- Caso de uso: Eliminar usuario del grupo
Perfil: Organizador, administrador
Descripción: El administrador del grupo puede desafiliar a un usuario de un grupo. Entonces éste no podrá acceder al contenido hasta que haga la solicitud de nuevo y se acepte.
- Caso de uso: Asignar ficheros al grupo
Perfil: Organizador, administrador
Descripción: El grupo puede contener ficheros adjuntos que pueden ser reglas del grupo, descripción del mismo o documentación necesaria para los usuarios del grupo. El administrador del grupo podrá asociar ficheros al grupo para que sus afiliados puedan descargarlo.

– Configuración de la web

Caso de uso: Configurar aspectos visuales

Perfil: Administrador

Descripción: La configuración de aspectos visuales es un conjunto de posibilidades que tiene el administrador para modificar los estilos de la web.

Caso de uso: Gestionar error reporting

Perfil: Administrador

Descripción: El usuario podrá configurar los redireccionamientos de páginas en caso de que se le muestre una página de error al usuario.

Caso de uso: Configurar fecha y hora

Perfil: Administrador

Descripción: El usuario podrá modificar los parámetros hora, fecha y zona horaria.

Caso de uso: Configurar información general de la web

Perfil: Administrador

Descripción: El usuario dispondrá de un formulario para editar esta información.

Caso de uso: Configurar preferencias sistema de ficheros

Perfil: Administrador

Descripción: El usuario podrá definir qué tipo de gestión de ficheros se realiza en la aplicación, definición de directorios de almacenamiento de ficheros, acceso general a los mismos, etc.

Caso de uso: Configurar estructura de la web

Perfil: Administrador

Descripción: El usuario podrá definir la estructura de los bloques de la web, así como de los primary links y vistas disponibles para los usuarios.

Caso de uso: Control de caching
Perfil: Administrador
Descripción: Se podrá activar o desactivar la gestión de caché para la web.

Caso de uso: Mantenimiento de la web
Perfil: Administrador
Descripción: El usuario podrá desactivar la aplicación mediante un formulario que le permitirá mostrar una página principal en la que se muestre el mensaje “Site under maintenance”.

– Gestión de usuarios

Caso de uso: Alta usuario
Perfil: Administrador
Descripción: El administrador podrá dar de alta un usuario, editando sus datos de perfil.

Caso de uso: Baja usuario
Perfil: Administrador
Descripción: Eliminación del usuario de la aplicación.

Caso de uso: Modificación usuario
Perfil: Administrador
Descripción: El administrador podrá modificar los datos de perfil de cualquier usuario.

Caso de uso: Asignación de roles
Perfil: Administrador
Descripción: El usuario podrá asignar los diferentes roles a los usuarios de la aplicación.

Caso de uso: Configuración control de acceso a funcionalidades de la web
Perfil: Administrador
Descripción: Funcionalidad de la aplicación que permitirá al administrador determinar qué funcionalidades se atribuyen a cada uno de los roles de la aplicación.

- Logs

Caso de uso: Acceder a los logs del sistema

Perfil: Administrador

Descripción: Conjunto de funcionalidades que permitirán al administrador consultar todos los logs que se generan en el sistema.

4. DISEÑO Y PARAMETRIZACIÓN DE DRUPAL

4. DISEÑO Y PARAMETRIZACIÓN DE DRUPAL

Un factor importante de la fase de diseño, es escoger la tecnología con la que se implementará el entorno visual. Esta elección condiciona en gran medida el éxito del proyecto y es necesario tomarla con detenimiento.

Una de las opciones es utilizar un editor web. Los editores web facilitan la tarea de creación de páginas y formularios en HTML. También se pueden encontrar algunos que interpretan ASP, PHP, JSP, Ruby o Pittón, lenguajes web con los que se pueda crear las funcionalidades del entorno visual. Esta elección sólo se ve limitada por el conocimiento y la habilidad del programador al desarrollar el código que dará forma al proyecto.

Existen también otras herramientas de construcción de espacios virtuales. Herramientas con las que se puede construir rápidamente una página web, personalizarla y adaptarla a las propias necesidades, y todo esto con unas garantías de seguridad, son los llamados gestores de contenidos. En este capítulo se explica qué es un gestor de contenidos y por qué puede ser útil.

4.1. Justificación del uso e implementación de un CMS y su elección

Los sistemas gestores de contenidos, del inglés Content Management System (CMS), son aplicaciones que facilitan una interfaz para la creación, edición, administración y publicación de contenidos de forma rápida y sencilla. En el caso de los gestores de contenidos web (WCMS), son aplicaciones web diseñadas para el tratamiento del contenido web, de manera que se simplifican todas las tareas (creación, edición, administración, publicación y mantenimiento) hasta el punto que, una persona sin conocimientos de HTML (HyperText Markup Language) ni PHP podría ser capaz de gestionar un sitio web. Sin embargo, el hecho de tener conocimientos de programación ayuda a que los CMS aún sean más potentes y versátiles.

El contenido que generan es almacenado en las bases de datos asociadas a la aplicación, el cual es frecuentemente guardado como XML (eXtensible Markup Language) para facilitar la reusabilidad y permitir opciones de presentación flexibles.

La presentación se basa en un conjunto de plantillas concretas y personalizables en las cuales se define cómo se debe mostrar cualquier contenido almacenado en la base de datos.

Por otra parte, los CMS definen diferentes roles de usuario (tal y como se ha definido en el apartado anterior) a los cuales se les otorgan diferentes partes del contenido y la ejecución de las tareas concretas. Además, los CMS acostumbran a disponer de un área de administración integrada en la aplicación a través de la cual se pueden gestionar los contenidos, la presentación, los usuarios, los permisos... es decir, casi la totalidad de la aplicación.

El hecho de utilizar un CMS en el desarrollo y el mantenimiento de esta aplicación aporta muchos beneficios si se compara con el desarrollo basado en una metodología tradicional de una aplicación (programada también en PHP y que utilice bases de datos). A continuación se exponen los más significativos:

– **Facilidad de uso**

Cualquier usuario que conozca el uso de Internet es capaz de navegar e interactuar con la aplicación. Además, por parte del administrador tan sólo se requiere el aprendizaje de funcionamiento del propio CMS.

– **Fácil de poner en marcha y configurar**

La instalación del CMS se acostumbra a basar en una serie de pasos automatizados. Una vez realizada, la gestión del CMS se realiza a través de menús y formularios. Éstos se encuentran el apartado de administración que configuran el contenido, su organización y disposición al navegador, gestionar los usuarios, sus permisos, las características del sitio web y las herramientas que permiten llevar a cabo estas acciones.

– **Aplicación más estable, más segura y con herramientas útiles incorporadas (o con posibilidad de incorporarse)**

El CMS que se instala ya ha sido probado y se ha garantizado su seguridad. A parte, el hecho de ser utilizado por muchos usuarios finales facilita el proceso de mejora continua por parte de los desarrolladores. Por otra parte, los CMS incorporan muchas herramientas como la generación automática de menús de navegación o buscadores) y dan la oportunidad de instalar nuevas.

– **Bajo coste económico**

Si la aplicación es de software libre, el coste se basará en el coste de instalación y el coste de aprendizaje por parte del desarrollador. Respecto al coste de mantenimiento de un CMS, se ahorra mucho más tiempo al crear o editar el contenido así como instalar nuevas funcionalidades.

– **Cumplimiento de estándares de Internet**

Casi todos los CMS cumplen (o intentan cumplir) los estándares de Internet establecidos por el W3C para garantizar la accesibilidad del lugar. De esta manera la aplicación será correctamente visualizada en cualquier navegador y sistema operativo.

– **Más atractivo para los usuarios finales**

La adaptación a una imagen moderna, de acuerdo con otras aplicaciones, hará que sea más atractiva y permitirá una mayor interacción con los usuarios finales.

En definitiva, estas razones han llevado a la decisión de realizar el proyecto de la aplicación mediante un CMS.

4.1.1. Elección del CMS

Dado que la página web actual de DBSS utiliza el gestor de contenidos Drupal, se ha decidido mantener este gestor para el desarrollo de este proyecto. De este modo, tanto la parte pública como la parte privada utilizarán el mismo gestor de contenidos.

Respecto a la versión de Drupal a utilizar, se ha seleccionado la versión 6, porque los módulos que se necesitaban instalar para satisfacer las funcionalidades que se requerían en la aplicación sólo estaban implementados para esa versión.

Cuando estén disponibles todos los módulos utilizados en la aplicación para la versión 7 de Drupal, se podrá proceder a la actualización de la web.

4.2. El gestor de contenidos Drupal

En este apartado se va a hacer un resumen de las características más importantes del gestor seleccionado.

Tal como se ha mencionado, Drupal es un CMS y como tal, proporciona un entorno de desarrollo orientado a gestionar contenidos. Dentro del amplio abanico de posibilidades, Drupal ofrece una API (Application Programming Interface), que gracias a esta capa intermedia de programación se eliminan muchos riesgos de seguridad programando y se gana mucho tiempo de desarrollo. Estas funciones son soportadas por los desarrolladores encargados del proyecto y mejoradas por la comunidad y en ocasiones son actualizadas para eliminar errores, ya sean funcionales o de seguridad.

En este sentido Drupal de nuevo es una buena herramienta, ya que permite la actualización programada, manual y al más mínimo detalle.

A grandes rasgos Drupal se caracteriza por tener un buen diseño y un código de buena calidad (documentado para ser aceptado y con una gran comunidad). Es muy extensible ya que está basado en un sistema de módulos (hay miles disponibles en la propia página del proyecto) que se pueden añadir o quitar independientemente del resto. Entre ellos destaca la posibilidad de internacionalización y todo tipo de gestores de tipo de datos.

La API de Drupal también ofrece toda una capa de acceso a bases de datos con funciones que prestan especial atención a la seguridad. En la presentación de datos Drupal cuenta con una potente gestión de temas gráficos, extensiones para usar AJAX (*Asynchronous JavaScript And XML*), Java Script y de manera transparente añadir una gran cantidad de efectos gráficos a las aplicaciones web. El aspecto visual, aunque no imprescindible funcionalmente, no deja de ser un punto importante de cara al usuario final.

Básicamente Drupal se basa en:

- Sistema modular basado en eventos (hooks).
- Capa de abstracción de base de datos.
- Sistema de menús.
- Generación de formularios.
- Sistema de control de acceso.
- Sistema de temas gráficos (templates).

4.2.1. Módulos en Drupal

Los módulos de Drupal son extensiones de sus capacidades básicas que aportan funcionalidades tanto genéricas como específicas, si se ha implementado un módulo bajo demanda.

El sistema de módulos de Drupal es uno de sus puntos fuertes y para manejarlos se basa en los llamados Hooks. Los Hooks son un conjunto de eventos definidos por los diseñadores de Drupal que permiten que cualquier módulo interactúe con el sistema gestor.

La idea de los Hooks es que al haber un evento, todas las funciones mencionadas con el nombre de este evento son ejecutadas, sencillas y potentes. Por ejemplo, el

Hook menú, muestra los datos en el menú de Drupal. Hay más de 70 Hooks implementados al núcleo de Drupal 8. Todo esto permite desarrollar módulos que no interfieran en la codificación del resto, y así poder añadir y eliminar módulos de forma transparente a los demás.

4.2.2. Arquitectura

Para empezar, el hecho de utilizar Drupal hace que sea este sistema el que defina la arquitectura del software. Drupal propone una arquitectura por capas que guarda mucha relación con un MVC (Modelo-Vista-Controlador) aunque muchos usuarios le atribuyen más aspectos de una arquitectura PAC (Presentación-Abstracción-Control).

No se va a entrar en detalles de la arquitectura software que utiliza Drupal, pero se puede resumir en la idea de que, a grandes rasgos, utiliza una arquitectura por capas con distinción entre presentación, lógica de la aplicación y gestión de la persistencia de datos. Drupal tiene también un diseño que utiliza algunos patrones muy conocidos en la ingeniería del software, como podrían ser Singleton o Decorator,

Llegados a este punto, vale la pena mencionar que Drupal, pese a utilizar PHP para su codificación, no está encasillado dentro del paradigma de orientación a objetos. PHP, a partir de la versión 5, soporta orientación a objetos y Drupal trabaja en algunos aspectos con éstos, pero su núcleo está basado en código procedural. La notación UML funciona y encaja muy bien en sistemas que se piensan en orientación a objetos.

Igual que sucede con cualquier sistema de notación, UML es capaz de representar unos sistemas de forma más concisa o eficiente que otros. Así pues, los desarrolladores tienden a ir hacia soluciones que resultan de la intersección entre las capacidades de UML y el lenguaje de implementación utilizado. Este problema es especialmente importante si dicho lenguaje de implementación no se adhiere a la doctrina de orientación a objetos más ortodoxa y se agrava, en la medida de las diferencias existentes, entre el lenguaje de implementación y la capacidad expresiva de UML.

Por otra parte, también hay que tener en cuenta que Drupal en sí mismo está constituido por un sistema central, que se va extendiendo y adquiriendo funcionalidades nuevas mediante los módulos. Al tratarse de Open Source Software, Drupal tiene detrás una comunidad muy activa de usuarios que mejoran y amplían los módulos constantemente, y también producen módulos nuevos para suplir funcionalidades que la versión por defecto no cubre.

En esta situación, muchas veces, cuando es necesario añadir una nueva funcionalidad a la web, es probable que otro usuario lo haya necesitado antes y haya diseñado e implementado un módulo para cubrir esa necesidad, por lo que se puede adoptar este módulo y reutilizarlo para nuestro sistema. De este modo se agiliza el desarrollo del sistema y no “se reinventa la rueda”.

Sin embargo, al hacer esto, se trae al sistema piezas que ya han sido diseñadas de una forma concreta y sirven para un propósito determinado. Si esto no acaba de encajar en el desarrollo, se tiene que tomar la decisión de, o bien adaptar el diseño al del nuevo módulo para que éste encaje de forma consistente en el sistema, o bien renunciar a este módulo que es “casi” lo que se prefiere, pero no lo suficiente, y hacer un desarrollo propio más a medida.

4.2.3. Características generales

A continuación se mencionan las características más relevantes de Drupal.

- Ayuda on-line: un robusto sistema de ayuda on-line y páginas de ayuda para módulos del “núcleo” tanto para usuarios como por los administradores.
- Búsqueda: todo el contenido en Drupal está totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- Código abierto: el código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario que otros sistemas de “blogs” o de gestión de contenido propietarios es posible extender o adaptar Drupal según las necesidades.
- Módulos: la comunidad de Drupal ha contribuido con muchos módulos que proporcionan funcionalidades como “página de categorías”, autenticación mediante LDAP, mensajes privados, bookmarks, etc.
- Personalización: un robusto entorno de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo a las preferencias definidas por el usuario.
- URL's amigables: Drupal utiliza el modo rewrite de Apache para crear URLs que es manejable por los usuarios y los motores de búsqueda.

- Gestión y validación de usuarios:
 - o Los usuarios se pueden registrar e iniciar sesión de forma local o utilizando un sistema de autentificación externo como Jabber, Blogger, Live Journal u otro sitio Drupal.
 - o Para su uso en una intranet, Drupal se puede integrar con un servidor LDAP.
 - o En Drupal, los permisos están basados en roles. De este modo, los administradores de Drupal no tienen que establecer permisos para cada usuario y pueden asignar permisos a un rol y agrupar los usuarios por roles.
- Gestión de contenidos: para la gestión de contenidos Drupal utiliza el control de versiones. El sistema de control de versiones de Drupal permite seguir y auditar totalmente las sucesivas actualizaciones del contenido: qué se ha cambiado, la hora y la fecha, quién lo ha cambiado, etc.
- Enlaces permanentes (Permalinks): todo el contenido creado en Drupal tiene un enlace permanente asociado a él para que pueda enlazar externamente sin temor a que el enlace falle en el futuro.
- Objetos de contenido (Nodes): el contenido creado en Drupal es, funcionalmente, un objeto (Node). Esto permite un tratamiento uniforme de la información como una misma cola de moderación para los envíos de diferentes tipos, promocionar cualquiera de estos objetos a la página principal o permitir comentarios –o no– sobre cada objeto.
- Plantillas (template): el sistema de temas de Drupal separa el contenido de la presentación permitiendo controlar o cambiar fácilmente el aspecto del sitio web. Por otra parte, se pueden crear plantillas con HTML y/o con PHP.
- Sindicación del contenido: Drupal exporta el contenido en formato RDF/RSS (Resource Description Framework/Really Simple Syndication) para ser utilizado por otros sitios web. Esto permite que cualquiera con un navegador agregador de noticias visualice el contenido publicado en la web desde el escritorio.
- Blogging: Drupal incluye un potente agregador de noticias para leer y publicar enlaces a noticias de otros sites. Incorpora un sistema de caché en la base de datos con temporización configurable.

- *Soporte de Blogger API*: el API de Blogger permite que un sitio Drupal sea actualizado utilizando diversas herramientas que pueden ser herramientas web o herramientas de escritorio que proporcionen un entorno de edición más manejable.
- *Plataforma*: independencia de la base de datos. Aunque la mayor parte de las instalaciones de Drupal utilicen MySQL, existen otras opciones. Drupal incorpora una capa de abstracción de base de datos, que actualmente está implementada para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos.
- *Multiplataforma*: Drupal ha sido diseñado desde el principio para ser multiplataforma. Puede funcionar con Apache o Microsoft IIS como servidor web y en sistemas como Linux, BSD, Solaris, Windows y Mac OS X. Por otra parte, al estar implementado en PHP, es totalmente portable.
- *Múltiples idiomas y localización*: Drupal está pensado para una audiencia internacional y proporciona opciones para crear un portal multilingüe. Todo el texto puede ser fácilmente traducido utilizando una interficie web, importando traducciones existentes o integrando otras herramientas de traducción GNU Gettext.
- *Administración y análisis vía web*: la administración y configuración de Drupal se puede realizar completamente con un navegador y no necesita ningún programario adicional.
- *Análisis, seguimiento y estadísticas*: Drupal puede mostrar a las páginas de administración informes sobre referals (enlaces entrantes), popularidad del contenido o de cómo los usuarios navegan por el sitio.
- *Registros e informes*: toda la actividad y los acontecimientos del sistema son capturados en un registro de eventos que puede ser visualizado por un administrador.
- *Características de comunidad, comentarios enlazados*: Drupal proporciona un potente modelo de comentarios enlazados que hace posible seguir y participar fácilmente en la discusión sobre el comentario publicado. Los comentarios son jerárquicos, como en un grupo de noticias o en un fórum.
- *Encuestas*: Drupal incluye un módulo que permite a los administradores y/o usuarios crear encuestas on-line totalmente configurables.

- Fórmulas de discusión: Drupal incorpora fórmulas de discusión para crear sitios comunitarios vivos y dinámicos.
- Libro colaborativo: esta característica es única de Drupal y permite crear un proyecto o “libro” a ser escrito y que otros usuarios contribuyan con el contenido. El contenido se organiza en páginas cómodamente navegables.
- Rendimiento y escalabilidad: Drupal incorpora un mecanismo de control de congestión que permite habilitar y deshabilitar determinados módulos o bloques dependiendo de la carga del servidor. Este mecanismo es totalmente configurable y ajustable.
- Sistema de Caché: el mecanismo de caché elimina consultas en la base de datos incrementando el rendimiento y reduciendo la carga del servidor.

Aunque el desarrollo de Drupal se centraliza en drupal.org, existen, además de Drupal Hispano (comunidad hispánica de Drupal), diferentes comunidades o grupos de usuarios sobre Drupal en castellano y catalán.

4.2.4. Bases de datos y seguridad

La capa de abstracción de bases de datos es muy importante por todo lo que se pretende ofrecer, ya que el uso que se hará mayoritariamente será recoger datos de formularios, guardarlos en la BBDD y mostrarlo posteriormente.

Aunque PHP ya ofrece muchas funciones de acceso a la BBDD, la API de Drupal aporta una funcionalidad extra; un alto grado de seguridad. Mientras que en PHP deben implementarse las propias funciones de seguridad para evitar inyecciones SQL o datos no saneados, gracias a la capa de abstracción de acceso a la BBDD ya no habrá que preocuparse por esto.

4.3. Gestión de usuarios

Drupal posee un sistema de gestión de usuarios que permite el registro, la conexión, la desconexión y el mantenimiento de sus perfiles actualizados, llevándose a cabo todo ello de una forma segura y dirigida por parte de los administradores.

Los usuarios pueden crear sus cuentas y autenticarse con o sin permiso del administrador, según éste crea conveniente. Un correo es enviado cuando la nueva

cuenta se crea, y si necesita permiso se envía cuando la cuenta es activada por el administrador.

Una vez creados los usuarios, Drupal permite agruparlos por roles que permiten ajustar fácilmente la seguridad y administración del sitio. Un rol define un grupo de usuarios que tienen ciertos privilegios que deben definirse en la sección permisos del usuario. Drupal incluye de forma predefinida dos tipos de roles, usuario anónimo y usuario autenticado:

- Usuario anónimo: se emplea para los usuarios que visitan el sitio web pero no se han creado una cuenta o para los usuarios que no se han autenticado.
- Usuario autenticado: este rol se asigna de forma automática a cualquier usuario que ingresa en la web. Si con este tipo de administración no es suficiente se pueden crear tantos roles como sean necesarios y darle los permisos adecuados.

Con los roles ya establecidos es necesario asignarles los permisos que van a disponer, ya que la creación de un tipo de rol en Drupal, éste no tiene asignado ningún permiso.

4.4. Herramientas de desarrollo y entorno de pruebas

Durante el proceso de implementación se ha tenido que instalar un entorno de pruebas para el proceso de aprendizaje del propio CMS, su configuración, instalación de módulos, etc. Para no realizar todas estas pruebas en el servidor de explotación, se ha utilizado un software que ha permitido ir construyendo la web de forma local.

Se ha decidido, por tanto, el uso de un programa que incluya todo el software necesario para el funcionamiento de Drupal. EasypHP y XAMPP son dos de los programas que incluyen las funcionalidades necesarias para programar entornos web en PHP más conocidos. Como ya se poseía experiencia con XAMPP, se decidió utilizar ese programa.

XAMPP es un programa liberado bajo licencia GNU/GPL que actúa como servidor web libre. Es fácil de usar y permite desarrollar aplicaciones web dinámicas. Está compuesto principalmente por la base de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y Perl.

En la actualidad, está disponible para las plataformas GNU/Linux, Windows, Mac OS X y Solaris y únicamente requiere descargar y ejecutar un archivo y realizar unas pequeñas configuraciones en algunos de sus componentes.

4.5. Esquema de red

En la Figura 4.1. se muestra el esquema de red aplicado a la web.

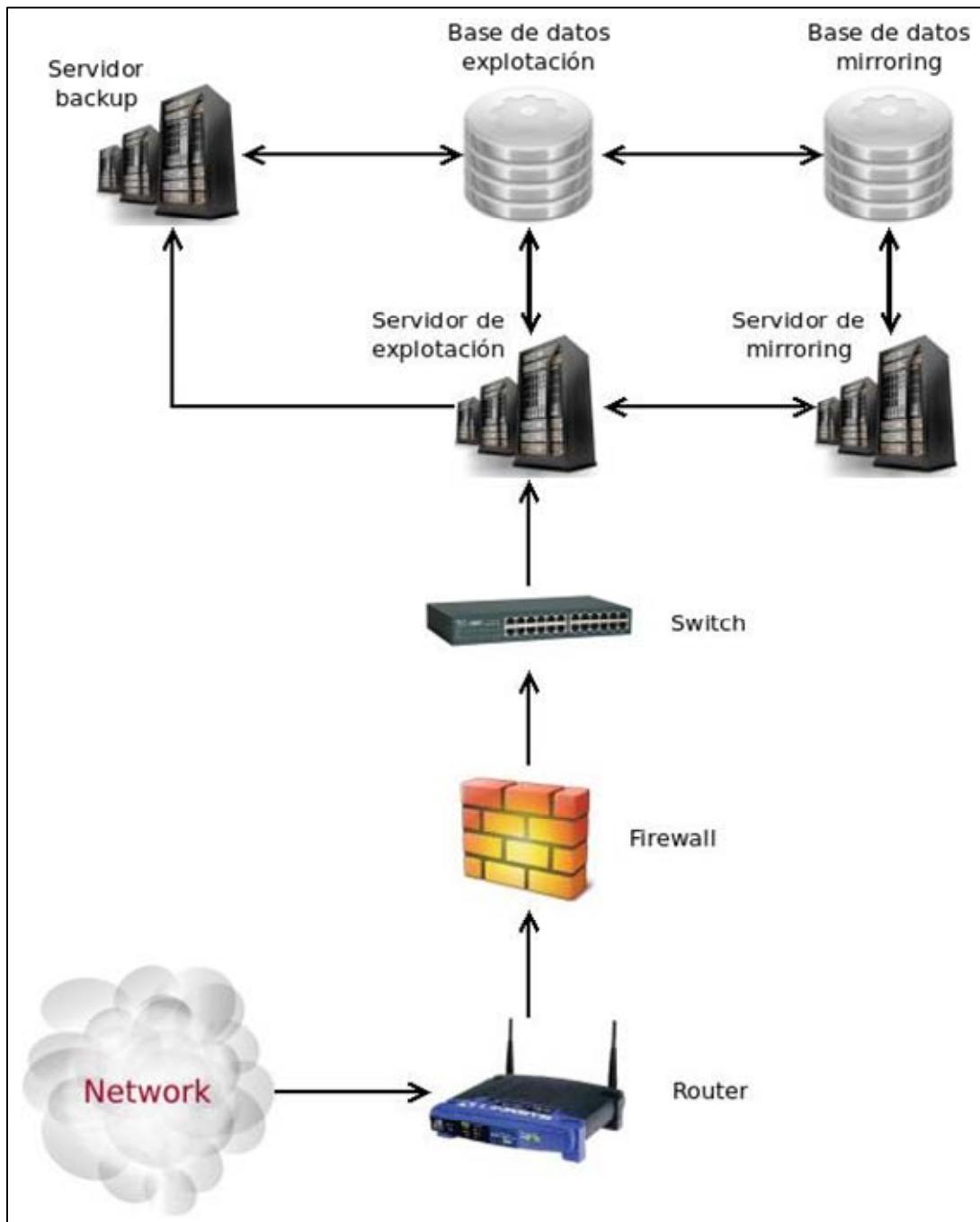


Figura 4.1. Esquema de red

Cuando un usuario accede a la aplicación, será dirigido al servidor de explotación. Si este servidor hubiera caído o hubiera sufrido algún daño, las peticiones serían redirigidas al servidor secundario (*mirror*), de una forma totalmente transparente al usuario.

Como se puede observar, se guarda una réplica del servidor primario y de la BBDD en el servidor de backup's.

4.6. Instalación Drupal

Drupal requiere para su instalación un servidor que pueda ejecutar scripts PHP, un motor de bases de datos soportado por PHP, y PHP.

El servidor recomendado es Apache pero, opcionalmente, puede funcionar con otros servidores como IIS (servidor desarrollado por Microsoft para Windows).

El gestor de la base de datos suelen ser MySQL o PostgreSQL, aunque los usuarios suelen usar en su mayoría la primera.

Las últimas versiones de Drupal funcionan correctamente con PHP 5.

4.6.1. Requerimientos del servidor

- Servidor web PHP: Apache, Nginx o Microsoft IIS.
- Intérprete PHP: PHP 4.4.0 o superior (5.2 recomendada).
- Servidores de BBDD: MySQL 4.1 o superior, PostgreSQL 7.1

Los requerimientos para todas las versiones de Drupal se pueden encontrar en:
<http://drupal.org/requirements/>

4.6.2. Configuración del servidor

En primer lugar deben instalarse los servicios PHP y MySQL. Mediante sencillos comandos se podrá disponer de ellos en el servidor.

Posteriormente se deberá instalar un gestor de BBDD, en este caso se instalará un gestor grafico de BBDD que permitirá la creación y edición de las tablas necesarias para el correcto funcionamiento de Drupal de una forma sencilla y rápida. En este caso se ha instalado PhpMyAdmin.

4.6.3. Creación de las BBDD's

Se entra en el administrador de base de datos para crear la base de datos que utilizará Drupal para almacenar toda la información necesaria. Para crearla, simplemente bastará con introducir un nombre para la misma y seleccionar el cotejamiento, que es la codificación para la base de datos.

Una vez establecida la base de datos ya se puede iniciar la instalación de Drupal.

4.6.4. Proceso de instalación de Drupal

En las primeras versiones que aparecieron de Drupal su instalación no resultaba una tarea sencilla puesto que había que insertar todo manualmente y crear las tablas y sus relaciones en la base de datos, pero a partir de las versiones 5.x y 6.x el proceso de instalación ha mejorado y la instalación resulta mucho más sencilla.

Después de descargar la versión que interesa, se descomprimen los ficheros en la carpeta definida para dicha instalación. La instalación se realiza desde el navegador web. Para ello se debe acceder a la dirección definida para la web y aparecerá la pantalla de instalación de Drupal.

La instalación se realiza por defecto en inglés, pero posteriormente se puede traducir Drupal a otros idiomas, aunque en este caso no interesa.

Drupal pedirá el nombre de la base de datos que se ha creado para la instalación, y el usuario y contraseña del propietario de dicha base de datos. Ya finalizada la instalación, se mostrará un link a la home de la web.

4.6.5. Configuración inicial del sitio web

Una vez instalado Drupal ya se puede iniciar el desarrollo, pero primero deben realizarse algunas configuraciones básicas antes de empezar.

Se debe crear el primer usuario de la aplicación, que será el superusuario que dispondrá de control total de la aplicación. Este usuario será el encargado de realizar las configuraciones iniciales de la web, y posteriormente podrá asignar otros usuarios como administradores de ciertos aspectos funcionales de la web.

La aplicación pide que se cree el primer usuario, que será el superadministrador de la web. Se crea el nuevo usuario introduciendo un username y una dirección de correo electrónico válida en la que se recibirá el link de confirmación de la cuenta.

Clicando en el link se entrará en la aplicación como superusuario y se podrá acceder al menú de administración y a nuestro perfil para modificar la contraseña o el username.

En la información del sitio puede cambiarse el nombre del sitio web, así como el eslogan, que será mostrado a menudo junto al título del sitio. Se podrá cambiar también el mensaje de pie de página, texto que se muestra en la zona inferior de cada página y que es útil para añadir información, condiciones de uso y distribución.

Drupal trae en su núcleo una serie de módulos para realizar las tareas más importantes, aunque se pueden instalar muchos más, los cuales otorgan nuevas funcionalidades, tal y como se ha explicado en apartados anteriores.

Si se desea el idioma del sitio web es necesario el paquete de traducción, en este caso no hará falta porque se dejará en inglés.

La web advierte de que se debe dirigir al reporte de estado (status report) y ejecutar el cron del servidor manualmente, módulo que sirve para la ejecución regular de acciones. Posteriormente, se deberá definir el directorio donde se almacenarán los ficheros subidos al portal.

Realizadas las configuraciones básicas, el sitio web queda un poco más personalizado y puede iniciarse el desarrollo del sistema.

A continuación, se define por una parte el proceso de construcción de la web mediante la instalación y configuración de los módulos utilizados, y por otra, la configuración general de la web.

4.7. Módulos utilizados

A continuación se detallan los módulos que se han utilizado para la implementación de la web. La instalación básica de Drupal ofrece los módulos del núcleo necesarios para el correcto funcionamiento de la aplicación. Posteriormente se le han añadido más módulos para darle las funcionalidades que se han descrito en la especificación.

En el apartado anterior ya se ha introducido el concepto de los módulos, y a continuación, se verá cuáles se han utilizado para ofrecer las funcionalidades necesarias para satisfacer los requerimientos del sistema.

- Módulos del core

Por módulos del core, se refiere a aquellos que ya vienen integrados con la versión básica de Drupal. Dentro del core, hay dos tipos de módulos: los requeridos y los opcionales.

En la Tabla 4.1. se muestran los módulos del core requeridos que se han utilizado en el presente proyecto.

Módulo	Funcionalidad
Block	Controla las cajas (o bloques) que se muestran alrededor del contenido principal.
Filter	Maneja el filtrado de contenido para mostrarlo.
Node	Permite que se pueda publicar contenido en el sitio web y que se muestre en páginas
System	Gestión de la configuración general del sitio por administradores.
User	Maneja el sistema de registro y acceso de usuarios.
WatchDog	Genera y almacena los logs del sistema.

Tabla 4.1. Módulos del core requeridos utilizados en el proyecto

Estos módulos son una parte vital del sistema y siempre deben estar habilitados. Sobre ellos se carga la responsabilidad de que funcionen las partes básicas del sistema.

Los módulos opcionales del core son módulos que también van incluidos por defecto en la instalación básica de Drupal, pero la mayor parte de ellos vienen desactivados inicialmente.

Para el sistema que se ha desarrollado, se han activado los que se muestran en la siguiente tabla (Tabla 4.2.).

Módulo	Funcionalidad
Aggregator	Permite agregar contenido RSS a nuestra web. Se ha tenido que activar para poder posteriormente definir el contenido RSS que se quiere mostrar.
Color	Permite cambiar el color de determinados temas de la web.
Help	Administra el despliegue de ayuda en línea. El uso que se hace de este módulo es principalmente interno, la mayor parte de ayuda en línea viene dada en forma de mensajes que guían, sobre todo, a los administradores cuando interactúan con la web.
Menú	Permite a los administradores personalizar los menús de navegación de la web. Se trata de un módulo muy importante que gestiona el sistema de menús de Drupal, la navegabilidad, los permisos y las conexiones de las diferentes partes de la web.
Statistics	Anota estadísticas de acceso a su sitio. Recopila estadísticas sobre la web que pueden ser consultadas posteriormente en los informes correspondientes.
Upload	Permite a los usuarios subir archivos y anexarlos a contenido. Con este módulo se permite que los autores puedan adjuntar ficheros a la web.

Tabla 4.2. Módulos del core opcionales utilizados en el proyecto

– Módulos contribuidos

Los módulos contribuidos son aquellos que alguien ha desarrollado por su cuenta y ha decidido compartir con toda la comunidad, con el propósito de mejorar algún aspecto de Drupal u ofrecer alguna funcionalidad que no se puede obtener solamente con los módulos del core. Todos ellos se pueden descargar libremente de la web oficial de Drupal.

En ningún caso se ha necesitado modificar el código de los módulos contribuidos que se ha utilizado para este proyecto, solamente ha sido necesario instalarlos y configurarlos, por lo que tampoco se extenderá aquí en sus detalles, puesto que existen muchas fuentes de documentación procedentes de la comunidad de Drupal.

En la Tabla 4.3. se listan los módulos contribuidos utilizados en el proyecto y a continuación se comentan y se relacionan los submódulos de cada uno de ellos.

Módulo
Content Construction Kit (CCK)
Event
Organic Groups (OG)
Google Analytics
Views
Web File Manager

Tabla 4.3. Módulos contribuidos utilizados en el proyecto

- **Content Construction Kit (CCK)**: es un módulo muy popular que consiste en un conjunto de módulos que permiten crear tipos de contenido adicionales respecto a los que ya existen por defecto en Drupal, asignándoles los atributos que se deseen, tales como, campos de texto, radio-buttons, campos de imagen, etc.

Los módulos que se han utilizado pertenecientes a CCK han sido:

- Content: Permite a los administradores definir nuevos tipos de contenido.
- Content templates: Permitirá definir la vista que se generará en el RSS feed y también se podrá configurar el código XML que se introducirá en el RSS.
- ImageField: Este módulo no venía por defecto en el CCK, se ha instalado posteriormente. Permite asignar un campo de upload de imagen para un tipo de contenido.
- Option Widgets: Permite asignar campos de radio-button, campos de texto (text-fields) y campos numéricos (numeric fields).
- Text: Permite crear campos de texto simple.

- Date: Permite crear campos de tipo fecha para luego ordenar el contenido en un calendario.
- **Event**: Conjunto de módulos que permite la gestión del tipo de contenido evento. Permitirá hacer un seguimiento de los eventos y mostrarlo o clasificarlos en un calendario. Los submódulos que se han activado son:
 - Basic event: Permite la creación y gestión de eventos.
 - Event: Se podrá utilizar la API calendar para mostrar los eventos.
 - Event repeat: Añade la funcionalidad de hacer una secuencia de repetición de evento.
 - Event views: Permite la creación de tipo evento y su posterior visualización.
- **Organic Groups (OG)**: Conjunto de módulos que permite la creación y gestión de módulos. Los submódulos utilizados son:
 - Organic Groups: Habilita a los usuarios crear y gestionar sus propios grupos.
 - Organic groups access control: Habilita el control de acceso para posts y grupos privados.
- **Google Analytics**: Añade el código que se debe incrustar en todas las páginas para hacer el tracking.
- **Views**: Views es un módulo muy popular que ofrece una interfaz muy potente mediante la cual los usuarios pueden construir vistas sobre muchos aspectos de la web. Típicamente se usa para ofrecer listados de algún tema en concreto.

Views permite concretar muchos aspectos de estos listados, como por ejemplo qué mostrar exactamente para cada ítem del listado que genere, condiciones que deben cumplir los elementos que se listan (mediante filtros y parámetros) y de qué forma se visualizarán los resultados de la vista generada.

Otro aspecto interesante, es que también gestiona la generación de feeds RSS para las vistas. De esta forma se puede ofrecer varios canales de este

estilo por toda la web. En uno de los próximos apartados se especifica las vistas definidas para la aplicación.

Los submódulos que se han activado han sido:

- Views: Permite crear vistas personalizadas de listas de contenido.
- Views RSS: Permite crear vistas con contenido RSS.
- Views UI: Permite editar las vistas creadas.

- **Web File Manager**: Módulo que permite añadir ficheros (previamente subidos al servidor por ftp) a la base de datos y adjuntarlos a los nodos o a los grupos, adoptando las doctrinas de restricción de acceso del nodo o grupo en cuestión.

4.8. Configuración de la web

En este apartado se especificarán las opciones de configuración que ofrecen Drupal y la aplicación.

4.8.1. Usuarios y permisos

La definición de los diferentes roles permite que cada grupo de usuarios tenga unas funciones específicas en la aplicación. El mismo Drupal permite definir como interactuará cada rol con el portal, pudiendo especificar a qué partes o contenido de la web podrá acceder el usuario.

Drupal permite controlar qué es lo que pueden hacer los usuarios en la web. Cada rol de usuario tiene su conjunto de permisos, que aparecen listados en su totalidad y pueden ser incluidos o excluidos según sea necesario.

Los permisos también permiten compartir con usuarios la carga de administración en caso que sea necesario, puesto que a cualquier rol se le puede asignar cualquier permiso.

En el siguiente esquema se muestra gráficamente los niveles de acceso de los roles de la aplicación:

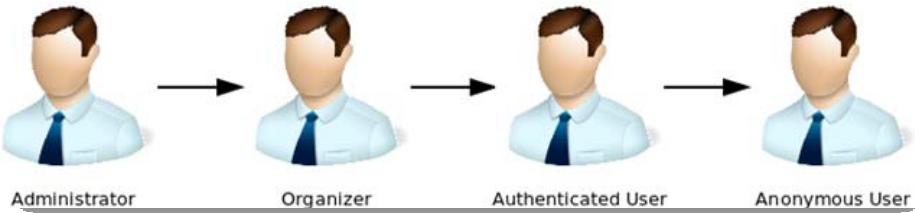


Figura 4.2. Niveles de acceso

El administrador será el rol de usuario con los permisos más elevados en la web. Seguidamente le proceden el organizador, que puede añadir contenido a la web, y le siguen el usuario autenticado y no autenticado.

4.8.2. Configuración del aspecto visual

Drupal es un sistema de gestión de contenidos altamente configurable, tanto en las funcionalidades que proporciona como en el aspecto visual, configurable mediante los temas que incluye y los que pueden ser instalados y posteriormente modificados.

Los temas en Drupal definen el aspecto visual del sitio web, incluyendo entre otras cosas, el tipo de letra, los colores, los menús y la posición de los bloques. Reciben el contenido que se pretende mostrar en una página y le dan formato.

Los temas que no incluye Drupal en la instalación pueden encontrarse y descargarse de muchos sitios en internet, especialmente la página oficial de Drupal (www.drupal.org), donde pueden encontrarse prácticamente todos los temas disponibles.

La configuración de los temas puede encontrarse en el apartado temas, dentro de la sección de administración, donde se listan todos los temas instalados y se ofrecen opciones para configurarlos.

Las opciones de configuración que se muestran ofrecen diferentes ajustes, entre los que se incluyen ajustes globales, que se aplican a todos los temas, y ajustes particulares para cada tema en concreto.

Las opciones de configuración permiten configurar muchos parámetros, como indicar que se muestre el logotipo que viene con el tema o uno propio, configurar las URL's que aparecerán en los enlaces e indicar que se muestren o no, diversos parámetros como fecha de publicación del contenido, avatar de los usuarios, nombre

del sitio o enlaces primarios y secundarios. Drupal usa por defecto el tema Minnelli e incluye otros cuatro temas en su núcleo.

Se ha creado un nuevo tema, que le proporcionará a la web los rasgos básicos que va a tener en cuanto a aspecto visual, pero posteriormente se irá mejorando y modificando con el uso de menús, secciones, bloques y vistas, que se podrá ir añadiendo gracias a los módulos que se han explicado en el apartado anterior.

Debe destacarse que, una vez instalado el tema en cuestión, se ha realizado una serie de cambios en los ficheros de estilo CSS 10, que han permitido ajustar algunos parámetros de la configuración de la estructura de bloques y vistas.

4.8.3. Creación de bloques y menús

Como todo sitio web, es necesaria la creación de menús y bloques para tener las secciones bien organizadas y disponibles, y ofrecer todas las herramientas necesarias a los usuarios para poder acceder a ellas, crear nuevos contenidos y disponer de información actualizada sobre el sitio.

Se va a crear una sección con links en la cabecera para mostrar los links básicos o primarios (primary links) y 3 ó 4 bloques (dependiendo del rol del usuario) en la parte izquierda, que contendrán diferente información. Uno de ellos será un listado de links en el que se añadirán enlaces a páginas relacionadas con la web; otro bloque será el de navegación, en el que se encontrará redirecciónamientos a diferentes secciones de la web. Los dos bloques restantes son: uno para la creación de contenido y administración y el otro para la gestión de grupos. En función del rol del usuario aparecerán unos bloques u otros.

– Bloques laterales

En la mayoría de sitios web existen enlaces a otras páginas que están relacionadas con el propio sitio o complementan la información que éste contiene. En este caso, para el propósito de este sitio web pueden ser necesarios enlaces a otras páginas, o bien links a diferentes partes de la misma web.

Gracias al módulo menú pueden crearse bloques que contiene links de una forma sencilla, porque al fin y al cabo, los bloques de links no son más que menús donde las secciones son enlaces a otras páginas.

Para crear los bloques de links es necesario ir a la sección menú, que se encuentra en construcción de la web, dentro del menú de administración. Una

vez dentro debe seleccionarse la opción añadir menú, darle nombre y crearlo. Una vez creado, habrá que editarlo para añadir el contenido y definir la ubicación del bloque, en la parte derecha o izquierda. Una vez guardados los cambios, las secciones de links serán creadas. Para finalizar se deberá definir a qué usuarios se les debe mostrar el bloque en cuestión. Para crear los demás bloques se seguirá el mismo procedimiento.

Drupal permite generar diferentes vistas para organizar el contenido, y para acceder a ellas se hará mediante un link para cada una. En nuestro caso, para crear el bloque “Site Navigation”, que enlaza a las diferentes partes de la web, se creará un bloque con diferentes links (igual que en el bloque anterior), pero en este caso los enlaces serán links a las vistas que se habrá generado anteriormente.

Además de personalizar los bloques también se puede utilizar los que vienen definidos por defecto en la instalación de Drupal, o bien los que se añaden al instalar nuevos módulos. Estos bloques generan información dinámica, información que va cambiando constantemente. Algunos de estos bloques son:

- [Calendar to browse events](#): genera un calendario en el que se muestra el mes actual con un icono distintivo en los días en los que se imparte un seminario.
- [Group members](#): Se le muestran al administrador de un grupo todos sus usuarios.
- [Group notifications](#): Información del grupo (usuarios totales, usuarios online, gestión de los usuarios, etc.).
- [New groups](#): Nuevos grupos creados recientemente.
- [Popular content](#): Contenido más visitado.
- [Primary links](#): Links de acceso directo.
- [Taxonomy List for Tags](#): Lista de tags y categorías.
- [User login](#): Formulario de login para entrar a la aplicación.
- [Upcoming events](#): Muestra en forma de lista los 5 eventos futuros más próximos a hoy.

Drupal ofrece más bloques que los anteriores citados, simplemente se ha pretendido dar una idea de sus posibilidades.

– Primary Links

Es uno de los bloques que viene por defecto en la instalación, formado por los “links destacados”. Sin tener en cuenta el tema instalado en la aplicación, estos links siempre destacarán con respecto a los demás links de la aplicación. Aparecerán en todas las páginas generadas, pudiéndose definir cada uno de ellos, para los cuatro tipos de perfiles definidos en la aplicación. En este caso se ha definido los siguientes elementos: home, logIn/logOut.

4.8.4. Tipos de contenido

La finalidad principal de un sistema de gestión de contenido es la creación de contenido y el tratamiento del mismo. Drupal proporciona las herramientas indispensables para crear contenido y organizarlo de forma adecuada.

Todo el contenido que se crea a través de Drupal es tratado internamente como un tipo de entidad genérica conocida como nodo. Un nodo podría considerarse como cualquier tipo de contenido que se le puede asociar de forma individual un conjunto de atributos independientemente que sea una noticia, un tema del foro o un artículo.

Los tipos de contenido que integra inicialmente Drupal son páginas e historias. Una página es un método sencillo de crear y mostrar información que no suele cambiar y es similar en forma y contenido a un artículo. Una historia, que tiene una estructura similar a una página, es ideal para crear y mostrar contenido que informa a los visitantes del sitio, como una entrada informal de un blog, un anuncio del sitio o una nota de prensa.

Sin embargo, en Drupal se pueden crear otros tipos de contenido gracias al módulo CCK (descrito en el apartado anterior). Permite la construcción de tipos de contenido personalizados de manera fácil a través de un panel de control.

CCK permite añadir nuevos campos al contenido básico, tales como imagen, fecha, link, correo electrónico, votaciones o cualquier tipo de texto o campo numérico, etc., para personalizar el tipo de contenido que se está creando.

Para crear un tipo de contenido, en la sección tipos de contenido que se encuentra dentro de gestión de contenido se selecciona la opción añadir tipo de contenido. Se le da un nombre, se define cuál es la función del tipo de contenido y se crea. A partir de este momento se le pueden añadir los campos que sean necesarios mediante el módulo CCK.

Una vez se han creado los tipos de contenido y ya existe contenido publicado, la forma más sencilla de organizarlo es mediante taxonomías. Drupal incluye en su núcleo el módulo taxonomía (definido en el apartado anterior), que permite categorizar el contenido usando etiquetas y términos definidos por un administrador.

Las categorías se implementan mediante vocabularios, términos y las relaciones que se establecen entre ellos, que pueden ser simples o jerárquicas, incluyendo jerarquías múltiples.

Los vocabularios se dan de alta en Drupal en la sección taxonomía que se encuentra en gestión de contenido. Una vez creado el vocabulario se le pueden añadir términos o etiquetas para definir diversas propiedades del contenido o tenerlo más esquematizado. En la aplicación se ha definido una única categoría llamada tags, que bastará para clasificar el contenido en etiquetas.

Crear un nuevo vocabulario es relativamente sencillo en Drupal, basta con ir a la sección taxonomía y seleccionar la opción añadir vocabulario. Se debe llenar el campo nombre, las opciones y elegir los tipos de contenido que formarán la taxonomía.

Una vez guardados los cambios ya se pueden añadir términos y agregar contenido a través de la interfaz de taxonomía. En el propio formulario de alta de evento también se pueden crear diferentes etiquetas sin tener que hacerlo des del menú de administración (al cual sólo tienen acceso los administradores).

4.8.5. Vistas

Si no se diseña de otro modo, todo el contenido publicado aparece en una lista simplemente organizada por orden de publicación, pero cuando el número posts empieza a crecer se necesita otro tipo de organización para que todos los contenidos sean fácilmente accesibles. Por ese motivo, se han creado diferentes vistas para ordenar el contenido publicado.

Para crear una vista se debe seguir el siguiente procedimiento:

- Añadir la información básica (nombre y descripción).
- Definir qué roles podrán acceder a la vista.
- La URL o link de la vista para poder acceder a ella.
- Definir el “Tipo de vista”: El formato de la vista, que puede ser una tabla, un globo con un título y descripción, vista para RSS, etc.

- Definir los “Campos”: Atributos que se deben mostrar en la vista.
- Definir los “Argumentos”: Permiten flexibilidad a la hora de generar vistas con parámetros concretos. Se utilizaría este campo si se quisiera, por ejemplo, que se ordenara el contenido por años y luego por meses.
- Filtros: Se define qué condiciones debe cumplir la selección de contenido, por ejemplo, tipo de contenido que se muestra, si está publicado o no, etc.
- Ordenación: Se puede ordenar el contenido por algún campo en concreto, ascendente o descendente.

Asimismo, se han definido las siguientes vistas para poder estructurar el contenido de la web:

- ***Events_list***: Se muestra una lista de eventos en forma de tabla con la información básica de cada evento:
 - *Date*: Fecha de inicio del evento.
 - *Topic*: Título del evento.
- En esta vista aparecerán todos los eventos públicos que hayan sido publicados en la web, tanto futuros como pasados, hasta un máximo de 20 eventos por página. Se utilizará esta vista como página principal de la web.
- ***Events_repository***: Vista que muestra todos los eventos pasados respecto a la fecha actual.
- ***Future_events***: Se muestran los eventos futuros a la fecha de hoy y también los que se están impartiendo en este momento.
- ***News***: Vista que nos muestra las noticias publicadas en la web.

Además de las vistas anteriormente citadas, existen otras vistas que ya vienen previamente definidas en la instalación básica de Drupal. A continuación se describen las que se han utilizado en este proyecto:

- ***Lista de grupos (og)***: Es el directorio de grupos de la aplicación. Todos los grupos creados como públicos aparecerán en esta vista.
- ***Grupos afiliados del usuario (og_my)***: Esta vista muestra los grupos afiliados al usuario que la solicita.
- ***Perfil***: Vista que muestra y permite modificar los datos de usuario.

- **Login:** Formulario de entrada a la aplicación.
- **Node event:** Muestra el contenido íntegro del nodo tipo evento.
- **Node group:** Muestra el contenido del grupo (descripción y ficheros adjuntos).
- **Node New:** Muestra el contenido completo de una noticia.
- **Tags:** Vista que muestra la lista de todos los tags introducidos en la aplicación.
- **Single Tag:** En esta vista se puede ver todo el contenido asociado a un tag en concreto.

4.8.6. Logs del Sistema

Como se ha descrito anteriormente, Drupal incorpora un sistema de logs que genera el sistema, que en este caso, sólo pueden consultar los usuarios con el rol de administrador.

Para activar el sistema de logs se debe activar el módulo *Statiscs*, definir qué información se desea mostrar y qué roles podrán acceder a ella. Éste módulo generará los logs de carácter más general, como el número de accesos totales, páginas más visitadas, accesos de los usuarios y algunas más que se han definido en el apartado 3.2.7. “Administración”.

Se ha instalado el módulo *Google Analytics*, que incrustará el código necesario, para que *Google Analytics* pueda registrar toda la actividad del site en todas las páginas de la aplicación.

Si se pretende que otros usuarios (por ejemplo los que tienen el rol de administrador) puedan acceder a este sistema de logs, el superadministrador de la aplicación deberá entrar en la aplicación de *Google Analytics* y darles de alta des de allí, agregándolos como otros usuarios de la cuenta.

4.9. Otras funcionalidades

4.9.1. Gestión de ficheros

En este apartado se definirá la estructura de directorios de la instalación de Drupal, así como las directrices para gestionar los ficheros.

En la Figura 4.3. se muestra el directorio principal de la instalación básica de Drupal.

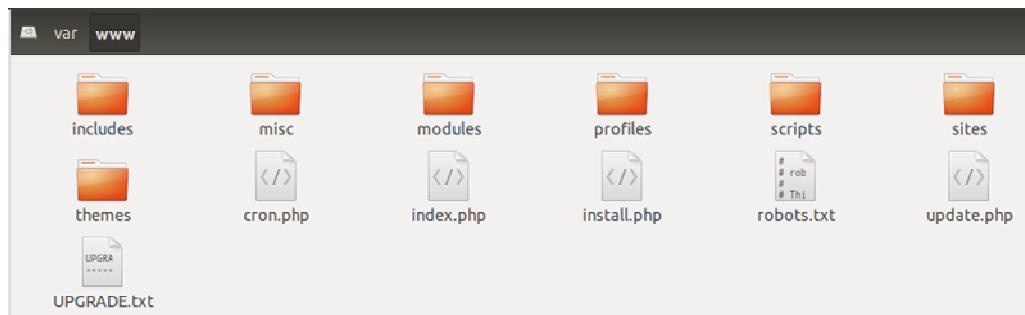


Figura 4.3. Sistema de ficheros

En la Tabla 4.4. se detallan brevemente cada uno de los iconos que aparecen en el directorio principal.

Icono	Funcionalidad
Carpeta includes	Contiene librerías de funciones comunes usadas por Drupal.
Carpeta misc	Contiene ficheros JavaScript e iconos usados en la web, resultantes de una instalación por defecto.
Carpeta modules	Se encuentran los módulos que forman el núcleo de Drupal. Todos los módulos que se añadan posteriormente se deberán introducir en este directorio.
Carpeta profiles	Contiene diferentes perfiles de instalación de Drupal.
Carpeta scripts	Contiene Scripts en Shell y Perl que son utilizados para revisar la sintaxis, limpiar el código e interaccionar con el cron.
Carpeta sites	Se guardan las modificaciones hechas sobre la web por defecto.
Carpeta themes	Contiene ficheros de los temas Drupal predeterminados. Si se instala nuevos temas se deberán copiar dentro de este directorio.
Fichero index.php	El fichero index.php es la página principal del site.
Fichero install.php	Se puede ejecutar manualmente en la instalación inicial del framework. Muestra un asistente de configuración del la web y de la base de datos.
Fichero update.php	Se encarga de realizar la actualización de la base de datos cuando se actualiza la versión de Drupal.
Fichero robots.txt	Controla el indexado de contenidos por parte de robots rastreadores evitando el acceso a determinadas partes de la web.
Fichero cron.php	Maneja peticiones de entrada para activar las tareas programadas regularmente.

Tabla 4.4. Sistema de ficheros

– Upload de ficheros

Drupal dispone de un módulo específico para gestionar el upload de ficheros, que permite subir ficheros mediante http al directorio especificado en la configuración del mismo. Los ficheros subidos tendrán una limitación de tamaño, ya que la configuración del servicio PHP del servidor prohíbe realizar un upload de un fichero superior a 2Mb.

Una de las necesidades de los usuarios de la aplicación que se debe satisfacer es permitir al *organizer* o al *administrator* añadir contenido adjunto (ficheros) a los eventos con un tamaño superior a 2Mb.

Este problema se puede resolver fácilmente dando acceso por ftp al usuario, para que suba los ficheros al servidor directamente de una forma fácil, y por supuesto, más rápida.

Si se quiere dar algún tipo de restricción de acceso al fichero subido, se debe primero añadir o registrar en la base de datos y posteriormente asociarlo al nodo que se quiera (evento o grupo), para que los ficheros adquieran la restricción de acceso que está definida para el nodo. De esa forma, sólo los usuarios con acceso al grupo o evento podrán acceder a los ficheros.

Para realizar las acciones descritas se utilizará el módulo WebFM, que permitirá realizar todo el procedimiento de una forma sencilla y rápida. A continuación se describe el funcionamiento y uso del módulo.

– Gestor WebFileManager (WebFM)

El módulo WebFM presenta un cambio de paradigma en la gestión de archivos para Drupal. Este administrador de archivos se basa en una estructura jerárquica de directorios a diferencia del tradicional sistema de estructuración de archivos utilizando por defecto en Drupal.

WebFM utiliza AJAX para permitir a los usuarios organizar archivos en el servidor de la misma forma que lo hacen en sus sistemas personales. Esta habilidad para organizar jerárquicamente los archivos mejora la capacidad de gestión de grandes colecciones de ficheros.

- Configuración

Se debe instalar este módulo de la misma forma que se ha hecho con los demás. Una vez activado se deberá realizar las configuraciones pertinentes, que consistirán en especificar qué funcionalidades del módulo se le atribuye a cada uno de los roles que interactuarán con la aplicación.

También deberá definirse el directorio raíz de todos los ficheros subidos, ya que posteriormente se podrán crear los propios directorios.

- Funcionamiento

En el formulario de creación del contenido aparece un campo llamado *WebFM Attachments* con una interficie que mostrará el contenido del directorio root que se ha definido en la configuración del módulo.

La interficie permite crear subdirectorios respecto al directorio root, para poder clasificar el contenido bajo un criterio. Estos directorios también pueden ser creados directamente en el servidor.

En primer lugar se debe subir al servidor (por debajo del directorio root definido en el módulo) por ftp los ficheros que se quieren asociar a los nodos que se van crear. Una vez subidos los ficheros se procederá a la creación del contenido.

Se debe especificar qué ficheros se quieren adjuntar al nodo que se está creando. Primero se añade una entrada correspondiente al nombre y propietario del fichero en la BBDD. Se hará clicando con el botón derecho sobre el fichero y aparecerá un menú contextual con diversas opciones, entre ellas la de *Add to Database*. Una vez añadido a la base de datos, se volverá a clicar con el botón derecho y aparecerá la opción *Attach to Node*, con lo que se conseguirá que adopte los permisos de restricción de acceso del nodo.

Si el proceso se ha realizado con éxito, aparecerá el fichero en cuestión como “Attached Files” en el bloque superior.

Este proceso se puede realizar tantas veces como se desee, ya que no existe limitación de ficheros adjuntos para un nodo en concreto.

4.9.2. Mirroring

Este servicio consiste en replicar el contenido de un servidor a otro. Diariamente se actualiza el servidor secundario (mirror) respecto al servidor de explotación, para tener una réplica lo más actualizada posible de todo el contenido de la web.

En el caso de que el servidor principal tuviera una avería, sólo se debería redireccionar las peticiones del servidor principal al secundario, disponiendo nuevamente del site. La información que se perdería en este caso sería la introducida entre la hora de sincronización de los dos servidores y la hora de la fallida del sistema.

El proceso de sincronización se hace mediante el servicio rsync 12, que permite definir, entre muchas otras opciones, el servidor que actuará como mirror, la frecuencia de sincronización, el contenido a sincronizar y el momento en que debe iniciarse la sincronización.

Se ha creado un shellscript que realizará las llamadas a los servicios necesarios para realizar este proceso. El proceso a seguir es el siguiente:

- Sincronizar el contenido de los ficheros y directorios que forman la aplicación, por lo tanto, el directorio principal de la instalación de Drupal.
- Sincronizar la base de datos. Este procedimiento se hará mediante el servicio mysqldump que permite exportar la base de datos de la aplicación.

El shellscript sincronizará los directorios de la instalación de Drupal y todo el contenido de ambos servidores. Posteriormente se realizará un export de la base de datos del servidor principal y se importará en la base de datos del servidor secundario. De esta forma se tendrá los servidores replicados cuando finalice la ejecución del script. Es importante que la base de datos del servidor mirror tenga el mismo nombre que la que se encuentra en el servidor de explotación, sino la aplicación replicada no funcionaría.

4.9.3. Backup

Para asegurar aún más la integridad de la aplicación se utilizará otro servidor dedicado al almacenamiento de un repositorio de backup's de la aplicación, que se irán generando diariamente. Este servidor estará situado en un espacio físico diferente al que se encuentran los servidores de explotación y mirroring. De esta forma, en caso de que se produjera un incendio que afectara a los otros dos servidores, se podría recuperar toda la información.

En el mismo shellscript que se ha utilizado para realizar la sincronización entre el servidor de explotación y el secundario, se ha definido el proceso de creación del backup, que es el siguiente:

- Exportar la base de datos de la aplicación.
- Comprimir el directorio principal de la instalación de Drupal, excepto el directorio /drupal/files.
- Crear un tar que contendrá los ficheros resultantes de los dos pasos anteriores, con el nombre backup-seminars\$date, donde \$date será la fecha de creación.
- Mover el tar generado al servidor de backup's.
- Sincronizar el directorio /drupal/files del servidor de explotación con el de backups. En este caso simplemente se sincronizará el contenido del directorio, debido a que el volumen de ficheros es muy grande y son muy pesados y sería incoherente copiar ficheros como vídeos, pdf's, presentaciones, etc., que no son nunca modificados.

Si se tuviese que recuperar el sistema mediante el backup, el procedimiento a seguir sería el siguiente:

- Preparar el servidor con los servicios necesarios para que la aplicación pueda funcionar, especificados en el apartado 4.6.1 “Requerimientos del servidor”.
- Crear una base de datos con el mismo nombre que tenía la base de datos del servidor de explotación.
- Importar en la base de datos creada el fichero de exportación generado en el backup.
- Copiar el contenido del directorio de instalación de Drupal (que está comprimido en el backup) al servidor destino.
- Copiar el contenido de los ficheros que se encuentran sincronizados en el servidor de backup al nuevo servidor, concretamente en /drupal/files.

Una vez realizado este procedimiento, ya se dispondrá de la web nuevamente en funcionamiento sin problema alguno.

4.9.4. Cron

Para la ejecución periódica o diaria del shellscript, mencionado en los apartados anteriores, se utilizará el servicio cron del servidor. Cron es un administrador regular de procesos en segundo plano (daemon) que ejecuta procesos o guiones a intervalos regulares (por ejemplo, cada minuto, día, semana o mes).

Los procesos que deben ejecutarse y la hora en la que deben hacerlo se especifican en el fichero crontab. En este fichero se ha especificado que se ejecute el script backup-events.sh diariamente a las doce de la noche.

Existe una alternativa en el caso que no se desee editar el fichero crontab, se puede copiar el script al directorio /etc/cron.daily, lo que hará que también se ejecute diariamente.

4.9.5. Actualización

Periódicamente salen actualizaciones de Drupal que mejoran la seguridad y corrigen errores de versiones anteriores. Por ese motivo es importante que la web se vaya actualizando cada cierto tiempo.

La actualización de Drupal debe realizarla el superadministrador de la web, ya que es un proceso delicado. Será necesario realizar copias de seguridad y modificaciones en la base de datos, así que se utilizará el gestor de BBDD phpMyAdmin, tal y como se ha descrito en apartados anteriores, para realizar este procedimiento.

En primer lugar se debe comprobar en la comunidad de Drupal que los módulos contribuidos (los que no vienen por defecto en la instalación básica de Drupal) estén actualizados y disponibles para la nueva versión a la que se quiera actualizar. Si alguno de ellos aún no está disponible no se podrá actualizar la web.

El primer paso que se debe dar es realizar una copia de seguridad de la información almacenada en el servidor. Se debe copiar todo el contenido existente en el directorio de instalación de Drupal en adelante y, posteriormente, realizar una exportación de todos los datos almacenados en la base de datos que utiliza la aplicación. De esta forma se asegura la restauración de la web, en el caso de que el proceso de actualización fallara en algún momento.

El segundo paso es cambiar el estado de la web a modo “fuera de línea” o “modo mantenimiento”. Para ello, se entrará en la aplicación como superadministrador y en la sección mantenimiento del sitio, dentro de configuración del sitio, se activará la opción off-line, como se observa en la Figura 4.4. que se muestra en la página siguiente.

En el mismo formulario se podrá configurar el mensaje que verán los usuarios cuando accedan a la web. Cuando se guarde la configuración la web pasará a estar fuera de línea.

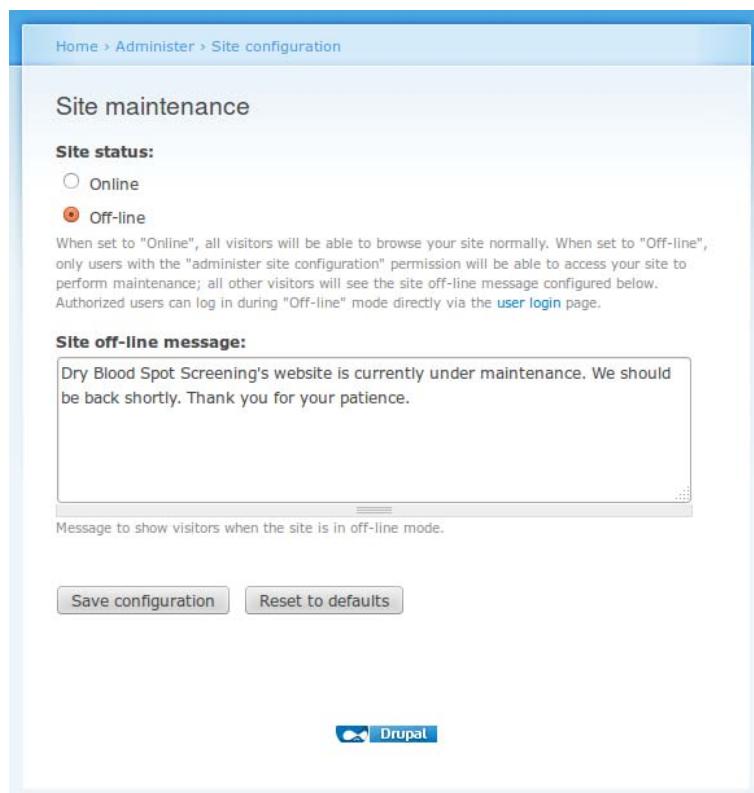


Figura 4.4. Mantenimiento de la web

Se recomienda modificar el tema de visualización de la web a uno por defecto (Garland or Bluemarine) para reducir al máximo el riesgo de no compatibilidad de la nueva versión. También se deberá desactivar los módulos contribuidos durante la instalación, debido a que se substituirán los existentes por los actualizados.

Una vez que la web está fuera de línea, se configurarán los permisos de escritura del archivo `settings.php` que se encuentra dentro del directorio `Drupal/sites/default`. Se le otorgarán sólo permisos de lectura tanto a la carpeta `default` como al archivo `settings.php`.

Posteriormente se deberá editar el archivo `settings.php`, donde se encuentran los parámetros generales de configuración de la web y cambiar la línea 106 del siguiente modo:

Se cambiará la variable

- `$update_free_access = FALSE;`

por:

- `$update_free_access = TRUE;`

Deben realizarse estos pasos para poder acceder y ejecutar el script `update.php` para actualizar la web.

El siguiente paso es reemplazar la instalación de Drupal. Se descargará la versión que se quiera instalar y los módulos contribuidos actualizados. Se copiarán todos los ficheros al servidor reemplazando la versión anterior por la nueva. Posteriormente se ejecutará el script `update.php` con el navegador web.

Se debe tener en cuenta que si simplemente se reemplazan los archivos de una versión a otra y no se realiza todo el procedimiento descrito anteriormente, cuando se intente acceder de nuevo al sitio aparecerá el error:

- "Fatal error: Call to undefined function `user_uid_optional_to_arg()` in `includes/menu.inc` on line 594"

Entonces no se podrá cambiar el sitio web del modo fuera de línea hasta que no se consiga ejecutar el script `update.php` correctamente.

Si el script `update.php` se ha ejecutado con éxito se muestra una página de actualización de la base de datos y ya podrá iniciarse el proceso de actualización del código fuente.

La versión de Drupal que se está actualizando se detecta automáticamente y después de pulsar el botón `update`, el proceso de instalación se inicia. Si se ha actualizado la web sin muestra de errores significará que se habrá realizado con éxito y ya será posible acceder a la página de administración de nuevo.

Una vez terminado el proceso, se debe restablecer el valor de la variable `$update_free_access` del archivo `settings.php`, y restaurar los permisos de escritura del directorio `Drupal/sites/default` y del archivo `settings.php`, que se encuentra dentro de éste.

Ahora debe activarse el sitio y quitar el modo fuera de línea mediante la aplicación de administración del superusuario.

A partir de este momento el sitio web se encontrará nuevamente activo y actualizado.

5. CONCLUSIONES

5. CONCLUSIONES

En este apartado se pretende realizar una valoración del grado de cumplimiento tanto de los objetivos planteados por la empresa como los de carácter personal, hacer una referencia a los retrasos sufridos en planificación, comentar posibles vías de ampliación y una valoración final.

5.1. Cumplimiento de objetivos

Para poder evaluar los resultados obtenidos en el proyecto es necesario tomar como referencia los objetivos, el alcance del proyecto y los beneficios esperados establecidos al principio de esta memoria.

Así pues, a partir de las necesidades de la empresa, se han seleccionado unas tecnologías que han resultado adecuadas tanto para el diseño de la web como de las bases de datos asociadas. Se han programado la web y las bases de datos con éxito y se ha llevado a cabo la implementación del sistema.

Se han realizado las pruebas de funcionamiento internas con éxito y también se han llevado a cabo tres estudios pilotos reales (en una empresa, en una escuela y actualmente en la selva de Alta Verapaz en Guatemala). En todos los casos los resultados han sido completamente satisfactorios. Por ello, se puede concluir que para un número relativamente pequeño de usuarios, la aplicación es robusta y no ha presentado ningún problema.

En cuanto a los beneficios esperados por DBSS, cabe mencionar que la aplicación ha mejorado ostensiblemente el grado de satisfacción de los usuarios internos (empleados):

- se ha mejorado la entrada de datos personales de los pacientes mediante una interfaz mucho más amigable
- y sobre todo, se ha solucionado el problema de la generación de los informes de resultados de los análisis de sangre, que con la aplicación ya se realiza de forma automática, evitándose las entradas manuales de estos valores tal y como se venía realizando.

De este modo, se puede concluir que se ha conseguido una aplicación fiable que cubre las necesidades y los beneficios esperados tanto técnicos y empresariales como de comodidad y facilidad de uso para el usuario.

5.2. Retrasos en planificación

Como suele ser frecuente, y este caso no ha sido una excepción, se han sufrido retrasos con respecto a planificación planteada. Sin embargo, en la mayoría de ellas se han ido cumpliendo los plazos planificados.

En la Figura 5.1. se muestra las diferencias entre las horas reales y las horas programadas. Como se puede observar se ha invertido un 11% más de tiempo que el planificado.

El principal motivo de retraso ha sido el gran número de procesos distintos a diseñar para un mismo objetivo (complejidad del problema) que conllevó a un mayor esfuerzo en la etapa de diseño que el planeado.

Inicial			Final		
	Duration	Task Name		Duration	Task Name
1	238,5 hrs	■ Proyecto Final de Carrera	1	266,25 hrs	■ Proyecto Final de Carrera
2	17 hrs	■ Planificación	2	18 hrs	■ Planificación
3	7 hrs	Identificar tareas	3	7 hrs	Identificar tareas
4	10 hrs	Establecer metodología de trabajo	4	10 hrs	Establecer metodología de trabajo
5	40 hrs	■ Estudio previo	5	40 hrs	■ Estudio previo
6	15 hrs	Estudio del escenario	6	15 hrs	Estudio del escenario
7	25 hrs	Estudio de la tecnología y entorno de programación	7	25 hrs	Estudio de la tecnología y entorno de programación
8	20 hrs	■ Objetivos y Alcance	8	20 hrs	■ Objetivos y Alcance
9	20 hrs	Definición de objetivos	9	20 hrs	Definición de objetivos
10	20 hrs	Establecer alcance del proyecto	10	20 hrs	Establecer alcance del proyecto
11	30 hrs	■ Analisis funcional	11	30 hrs	■ Analisis funcional
12	20 hrs	Requisitos funcionales	12	20 hrs	Requisitos funcionales
13	20 hrs	Requisitos no funcionales	13	20 hrs	Requisitos no funcionales
14	10 hrs	Casos de uso	14	10 hrs	Casos de uso
15	125 hrs	■ Diseño y parametrización de Drupal	15	142 hrs	■ Diseño y parametrización de Drupal
16	110 hrs	■ Drupal	16	118 hrs	■ Drupal
17	10 hrs	Instalación Drupal	17	10 hrs	Instalación Drupal
18	100 hrs	Modulos	18	100 hrs	Modulos
19	100 hrs	Gestion de usuarios	19	100 hrs	Gestion de usuarios
20	115 hrs	■ Diseño	20	120 hrs	■ Diseño
21	50 hrs	Diseño del modelo de datos	21	50 hrs	Diseño del modelo de datos
22	15 hrs	Diseño del cuerpo de la aplicación	22	20 hrs	Diseño del cuerpo de la aplicación
23	50 hrs	Diseño de procesos	23	50 hrs	Diseño de procesos
24	50 hrs	Configuración web	24	50 hrs	Configuración web
25	50 hrs	Recuperación	25	50 hrs	Recuperación
26	50 hrs	■ Docuemntación	26	55 hrs	■ Docuemntación
27	50 hrs	Memoria	27	50 hrs	Memoria
28	4 hrs	Revisió	28	4 hrs	Revisió

Figura 5.1. Retrasos de planificación

Por otra parte, esta mayor inversión en trabajo, también produjo un retraso en el tiempo total del proyecto. En la Figura 5.2. se muestra el diagrama de Gantt real en comparación con el planificado. Se puede observar que se ha sufrido un retraso de 2 semanas: se ha pasado de una duración de 4,5 meses a 5 meses.

5. CONCLUSIONES

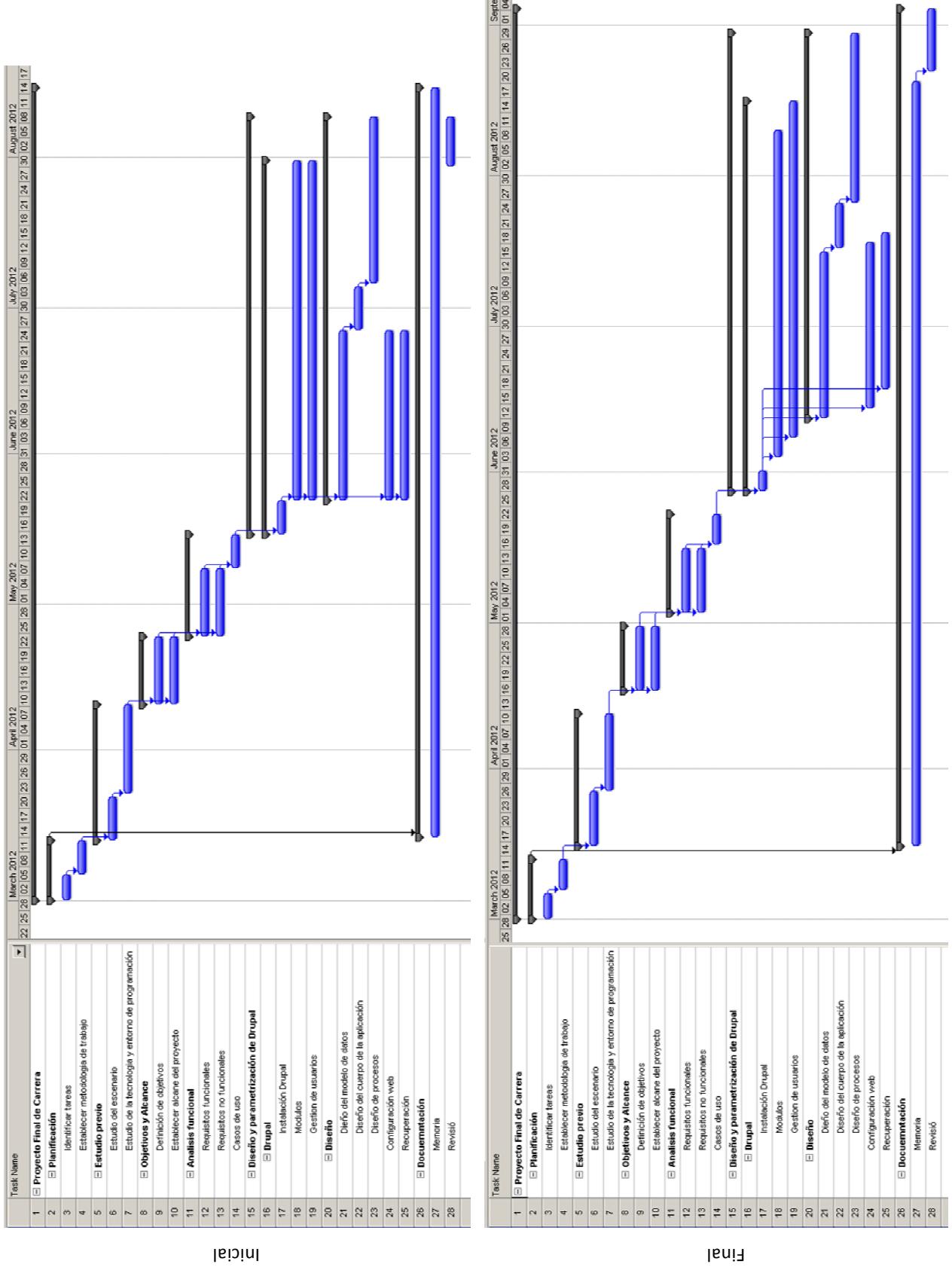


Figura 5.2. Comparación Diagramas de Gantt real vs. planificado

5.3. Líneas de ampliación

Aunque la aplicación ha funcionado y los resultados obtenidos son buenos, siempre existen líneas de ampliación o de mejora.

Así, a medida que se iba avanzando en el proyecto, se encontraban nuevas ideas o funcionalidades que se podrían implementar y representar una mejora en la aplicación como pueden ser una personalización del diseño gráfico de la web y su aplicación a la empresa, o todavía, seguir mejorando la interacción con el usuario con la web.

Sin embargo, hay dos grandes líneas de ampliación que deben tenerse en cuenta:

- Ampliación de la web a un apartado privado al que puedan entrar las personas, para poder consultar directamente los resultados de sus análisis o el histórico de los mismos.

Esto puede resultar de gran valor no sólo para las personas, sino también para sus médicos, que pueden tener on-line la evolución de la salud de sus pacientes.

Por otra parte, y dado que la empresa colabora con entidades públicas y privadas, para hacer campañas masivas de análisis de sangre, el tratamiento estadístico de estos datos y su conocimiento a través de la web pueden ser de gran valor como ayuda en la toma de decisiones y de priorización en la mejora de los sistemas de salud a cualquier nivel.

- Automatización de la entrada de los resultados de los análisis de sangre directamente del espectrofotómetro a la ficha del paciente en la web. Esto representaría una mejora radical del funcionamiento actual, minimizando, más si cabe, la posibilidad de una equivocación.

5.4. Valoración final

La valoración sobre el proyecto final de carrera que se ha realizado es muy positiva por varios aspectos y circunstancias: por una parte, la introducción en el mundo laboral y el primer reto de demostrar que los conocimientos adquiridos sirven en los casos reales.

Como proyecto final de carrera, ha estado un proyecto que me ha ayudado a comprender las distintas fases que completan un proyecto informático en el mundo real y asimilar la importancia, no sólo de analizar y diseñar previamente las ideas, sino también la necesidad de saber comunicarlas.

Como objetivos personales he cumplido con casi todos los propuestos. He mejorado el uso de las técnicas conocidas y he descubierto nuevas herramientas y con mi pequeño aporte, me he sentido cómplice de salvar alguna que otra vida de un niño en cualquier lugar del mundo.

6. GLOSARIO

6. GLOSARIO

A continuación se indican las abreviaturas y acrónimos que aparecen a lo largo de la memoria y que es conveniente definir para facilitar el seguimiento del mismo.

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
ASP	Active Server Pages
BBDD	Base de datos
CCK	Content Construction Kit
CMS	Content Management System
CSS	Cascading Style Sheets
DBSS	Dry Blood Spot Screening
FTP	File Transfer Protocol
GNU/GPL	General Public License
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
IIS	Internet Information Services
IRC	Insuficiencia Renal Crónica
JSP	Java Server Pages
LDAP	Lightweight Directory Access Protocol
MVC	Modelo-Vista-Controlador
OG	Organic Groups
OG_MY	Grupos afiliados al usuario
OMS	Organización Mundial de la Salud
ONG	Organización No Gubernamental
PAC	Presentación-Abstracción-Control
PFC	Proyecto Final de Carrera
PHP	Hypertext Preprocessor
RDF	Resource Description Framework
RSS	Really Simple Syndication
SQL	Structured Query Language
UML	Unified Modeling Language
URL	Uniform Resource Locator
VIH	Virus de Inmunodeficiencia Humana
W3C	World Wide Web Consortium
WCMS	Web Content Management System
WebFM	WebFileManager
XML	eXtensible Markup Language

7. ÍNDICE DE TABLAS Y FIGURAS

7. ÍNDICE DE TABLAS Y FIGURAS

7.1. Índice de tablas

Tabla 1.1. Estrategia de crecimiento	4
Tabla 2.1. Planificación de las tareas	15
Tabla 2.2. Estimación de los costes materiales	17
Tabla 2.3. Estimación de los costes de personal	17
Tabla 2.4. Estimación de costes del PFC	18
Tabla 3.1. Roles, grado de acceso e interacción con la aplicación	22
Tabla 3.2. Tipos de restricción de afiliación	26
Tabla 4.1. Módulos del core requeridos utilizados en el proyecto	62
Tabla 4.2. Módulos del core opcionales utilizados en el proyecto	63
Tabla 4.3. Módulos contribuidos utilizados en el proyecto	64
Tabla 4.4. Sistema de ficheros	74

7.2. Índice de figuras

Figura 1.1. Posicionamiento de DBSS y modelo de negocio	3
Figura 1.2. Ventajas de la tecnología DBSS frente a la tradicional	5
Figura 1.3. Cadena de valor	6
Figura 1.4. Material necesario para la extracción	6
Figura 1.5. Etapas del proceso de análisis cuantitativo en el laboratorio	7
Figura 1.6. Etapas y evolución del proyecto “informatización DBSS”	8
Figura 2.1. Etapas y alcance del PFC	14
Figura 2.2. Diagrama de Gantt del PFC	16
Figura 3.1. Esquema de la estructura de la página web	27
Figura 3.2. Casos de uso Anonymous User.....	37
Figura 3.3. Casos de uso Authenticated User.....	37
Figura 3.4. Casos de uso Organizer	38
Figura 3.5. Casos de uso Administrator	39
Figura 4.1. Esquema de red	58
Figura 4.2. Niveles de acceso	67
Figura 4.3. Sistema de ficheros	74
Figura 4.4. Mantenimiento de la web	80
Figura 5.1. Retrasos de planificación	85
Figura 5.2. Comparación Diagramas de Gantt real vs. planificado	86

8. BIBLIOGRAFÍA

8. BIBLIOGRAFÍA

Referencias bibliográficas

- [1] Nordin, Dani; *Planning and managing Drupal projects*; Beijing, Sebastopol: O'Reilly, cop. 2011.
- [2] Dubois, Paul; "MySQL", Madrid : Anaya Multimedia, DL 2005.
- [3] Fitzgerald, Bill; *Drupal for education and e-learning: teaching and learning in the classroom using the Drupal CMS*; Birmingham, Packt Publishing, 2008.

Recursos Web

- [4] Dries Buytaert; "Drupal - Open Source CMS | drupal.org" [online]. Disponible en: <http://drupal.org/>; (2012, Abril).
- [5] Dries Buytaert; "Drupal Hispano | Comunidad de usuarios de Drupal" [online]. Disponible en: <http://drupal.org.es/>; (2012, Marzo).
- [6] Oracle; "MySQL: The world's most popular open source database" [online]. Disponible en: <http://www.mysql.com/>; (2012, Abril).
- [7] Free Software Foundation, Inc.; "Free Software Foundation; Free Software Foundation; working together for free software" [online]. Disponible en: <http://www.fsf.org/>; (2012, Marzo).
- [8] SEO; "Foros del Web - Programación web - RSS Feed" [online]. Disponible en: <http://www.forosdelweb.com/>; (2012, Abril-Septiembre).
- [9] Desconocido, "Tutorial Outpost; Photoshop Tutorials, Flash Tutorials, and more!" [online]. Disponible en: <http://tutorialoutpost.com/>; (2012, Mayo).