

## 5691 - UABPROFITEM / E-NCANTS

Memoria del Proyecto de Final de Carrera correspondiente a los estudios de Ingeniería Superior en Informática presentado por Daniel Valverde Saavedra i dirigido por Ramon Martí Escalé.

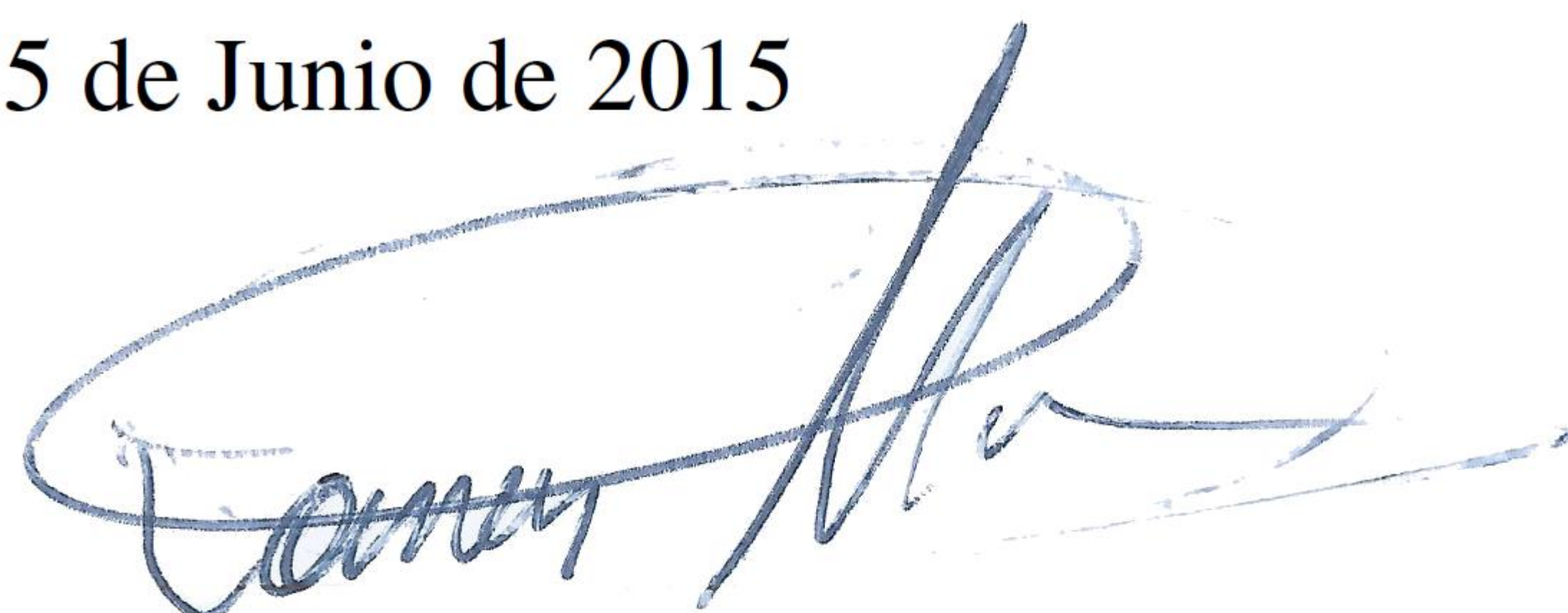
Bellaterra, 15 de Junio de 2015

El firmante, Ramon Martí Escalé, profesor del Departamento de Ingeniería de la Información y de las Comunicaciones de la Universidad Autónoma de Barcelona

CERTIFICA:

Que la presente memoria ha sido realizada bajo su dirección por Daniel Valverde Saavedra

Bellaterra, 15 de Junio de 2015



---

Firmado: Ramon Martí Escalé



*A todas las personas que han influido en mi vida.*



# Agradecimientos

Me gustaría agradecer el apoyo que he tenido por parte de mi familia, especialmente de mis padres, que siempre me han ayudado incondicionalmente. También quiero agradecer y dedicar una parte de este proyecto a Nadia, por animarme y estar siempre que la necesito.

Finalmente quisiera dar las gracias a mi director de proyecto, Ramon Martí Escalé, por darme la oportunidad de realizar este proyecto; también a los profesores de la carrera que me han marcado por su forma de enseñar y sus ganas.



# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Objetivos del proyecto . . . . .	1
1.2. Estructura de la memoria . . . . .	2
<b>2. Estado del arte</b>	<b>3</b>
2.1. Introducción . . . . .	3
2.1.1. Frameworks . . . . .	3
2.1.2. Sistemas de gestión de contenidos . . . . .	4
<b>3. Análisis</b>	<b>7</b>
3.1. Requisitos funcionales . . . . .	7
3.2. Requisitos no funcionales . . . . .	8
3.3. Viabilidad del proyecto . . . . .	8
3.3.1. Viabilidad técnica . . . . .	8
3.3.2. Viabilidad económica . . . . .	9
3.3.3. Viabilidad legal . . . . .	9
3.3.4. Conclusiones . . . . .	9
3.4. Planificación temporal . . . . .	10
3.4.1. Planificación inicial . . . . .	10
3.5. Especificación de los casos de uso . . . . .	11
3.5.1. Caso de uso: Registrarse . . . . .	13
3.5.2. Caso de uso: Iniciar sesión . . . . .	13
3.5.3. Caso de uso: Cerrar sesión . . . . .	14
3.5.4. Caso de uso: Administrar cuenta . . . . .	14
3.5.5. Caso de uso: Editar perfil de usuario . . . . .	14
3.5.6. Caso de uso: Consultar material publicado . . . . .	15
3.5.7. Caso de uso: Crear material . . . . .	15
3.5.8. Caso de uso: Ver material . . . . .	16
3.5.9. Caso de uso: Solicitar material . . . . .	16
3.5.10. Caso de uso: Buscar material . . . . .	17
3.5.11. Caso de uso: Administrar el sistema . . . . .	17

<b>4. Diseño e implementación</b>	<b>19</b>
4.1. Diseño . . . . .	19
4.1.1. Base de datos . . . . .	20
4.1.2. Nodos . . . . .	21
4.1.3. Módulos . . . . .	21
4.1.4. Hooks . . . . .	21
4.1.5. Bloques . . . . .	22
4.1.6. Menús . . . . .	22
4.1.7. Usuarios . . . . .	22
4.1.8. Temas . . . . .	22
4.2. Implementación . . . . .	24
4.2.1. Creación de contenido . . . . .	24
4.2.2. Solicitud de material . . . . .	25
4.2.3. Interfaz de usuario . . . . .	26
4.2.4. Gestión de usuarios . . . . .	29
4.2.5. Estilo . . . . .	29
<b>5. Pruebas</b>	<b>31</b>
5.1. Usuarios . . . . .	31
5.2. Creación de contenido . . . . .	33
5.3. Solicitud de material . . . . .	36
<b>6. Conclusiones</b>	<b>39</b>
6.1. Objetivos . . . . .	39
6.2. Planificación final . . . . .	40
6.3. Observaciones . . . . .	41
6.4. Futuras ampliaciones . . . . .	41
<b>Bibliografía</b>	<b>43</b>

# Índice de figuras

3.1. Planificación temporal (Octubre 2014 - Febrero 2015) . . . . .	10
3.2. Diagrama de casos de uso . . . . .	12
4.1. Plantillas tpl.php . . . . .	23
4.2. Implementación de tipo de contenido Material . . . . .	25
4.3. Página de materiales de informática para usuario identificado . . . . .	27
4.4. Panel de creación de una consulta . . . . .	28
5.1. Introducción de usuario y correo . . . . .	31
5.2. Mensaje de correo con enlace . . . . .	32
5.3. Introducción de contraseña . . . . .	32
5.4. Campos nombre y descripción . . . . .	34
5.5. Campos categoría e imágenes . . . . .	35
5.6. Material publicado . . . . .	35
5.7. Solicitud de material . . . . .	36
5.8. Confirmación de solicitud . . . . .	36
5.9. Mensaje de correo . . . . .	37
6.1. Planificación temporal (Octubre 2014 - Febrero 2015) . . . . .	41
6.2. Planificación temporal (Febrero 2015 - Julio 2015) . . . . .	41



# Índice de tablas

3.1. Planificación de tareas . . . . .	10
3.2. Registrarse . . . . .	13
3.3. Iniciar sesión . . . . .	13
3.4. Cerrar sesión . . . . .	14
3.5. Administrar cuenta . . . . .	14
3.6. Editar perfil de usuario . . . . .	14
3.7. Consultar material publicado . . . . .	15
3.8. Crear material . . . . .	15
3.9. Ver material . . . . .	16
3.10. Solicitar material . . . . .	16
3.11. Buscar material . . . . .	17
3.12. Administrar el sistema . . . . .	17
6.1. Planificación final de tareas . . . . .	40



# Capítulo 1

## Introducción

La UAB, como otras entidades públicas, dispone de un procedimiento para deshacerse de materiales que ya no se necesitan o bien han llegado al final de su vida útil. Entre los pasos a seguir, hay que solicitar a la administración de cada centro la retirada del material (el centro debe hacer una valoración de los objetos que se quieren retirar), darlo de baja del inventario de activos de la universidad y enviarlo al punto de recogida selectiva (Deixalleria de Cerdanyola Campus).

Existe la posibilidad de entregar directamente el material a cualquier otro departamento, con el único requisito de hacerlo constar en el inventario. El problema con esta opción es que no hay ningún lugar donde poder anunciar o buscar materiales descartados. Teniendo en cuenta esta situación, este proyecto tiene como objetivo la creación de una aplicación web para el intercambio de material en desuso entre departamentos de la UAB. El sistema debe incluir elementos como por ejemplo la creación y gestión de usuarios o la publicación de los materiales a intercambiar (formularios para introducir datos, tanto texto como imágenes).

### 1.1. Objetivos del proyecto

El objetivo principal de la aplicación es proveer de un entorno donde el personal de los departamentos de la universidad puedan dar a conocer material que ya no necesitan, y quienes estén interesados puedan solicitarlos. Para ello debemos conseguir los siguientes objetivos:

- Análisis y diseño de la base de datos de usuarios y materiales.
- Análisis y diseño de la gestión de usuarios.
- Análisis y diseño de la introducción de datos de materiales y subida de imágenes.
- Análisis y diseño de la resolución de peticiones de material.

- Implementación de la aplicación web.
- Testeo del sistema.

## 1.2. Estructura de la memoria

Los capítulos en los que se divide la memoria de este proyecto son:

- **Capítulo 1:** En este capítulo exponemos los objetivos del proyecto y la estructura de la memoria.
- **Capítulo 2:** En este capítulo hacemos una exposición de las tecnologías de desarrollo web.
- **Capítulo 3:** En este capítulo hacemos un análisis de los requisitos de nuestro proyecto, la viabilidad y la planificación.
- **Capítulo 4:** En este capítulo vemos el diseño y la implementación del sitio web.
- **Capítulo 5:** En este capítulo explicamos las pruebas que hemos realizado para verificar el correcto funcionamiento de nuestro sitio.
- **Capítulo 6:** en este último capítulo exponemos las conclusiones a las que hemos llegado al finalizar nuestro proyecto y sugerimos ampliaciones.

# Capítulo 2

## Estado del arte

### 2.1. Introducción

Existen muchos lenguajes que permiten el desarrollo de aplicaciones web como por ejemplo ASP, Java, Python, Perl, Ruby o PHP. Este último es actualmente uno de los más usados en sitios web. Es de código abierto, y destaca por ser fácil de aprender, por su eficiencia (puede manejar grandes cantidades de usuarios y peticiones simultáneas.) y flexibilidad (puede correr en la mayoría de sistemas operativos y servidores web).

Todo proyecto web puede implementarse desde cero usando una combinación de PHP, HTML, CSS, Javascript y alguna base de datos como por ejemplo MySQL. Pero cuando una web crece en tamaño y complejidad es recomendable usar algún *framework* o algún sistema de gestión de contenidos (*CMS*) que nos facilite el trabajo.

#### 2.1.1. Frameworks

Un framework es una estructura de librerías y funciones que facilita el desarrollo de aplicaciones web aportando funcionalidades para actividades comunes como el acceso a bases de datos, plantillas, gestión de sesiones, etc. Mejoran la calidad del desarrollo, pero es necesario un conocimiento profundo de las APIs de desarrollo para llevar a cabo cualquier proyecto. Son recomendables cuando queremos llevar a cabo un desarrollo muy a medida.

Existen multitud de frameworks para diferentes lenguajes de programación. En el caso de PHP tenemos, por ejemplo, KumbiaPhp, CakePhp, Symfony o Laravel.

### 2.1.2. Sistemas de gestión de contenidos

Un sistema de gestión de contenidos es una plataforma para la creación y administración de contenidos de páginas web [4]. Proporciona un entorno que permite una gestión uniforme y sencilla de una aplicación web, su mantenimiento y actualización. Contiene herramientas para la administración de bases de datos y de usuarios, la gestión de menús, el diseño personalizable de páginas o la administración del sistema en general. Existen multitud de CMS en lenguajes como Java, ASP.NET, Ruby o PHP, entre otros. Según su uso y funcionalidad, existen diferentes tipos de gestores de contenidos:

- Blogs
- Foros
- Wikis
- E-Learning
- Comercio electrónico
- Propósito general

Utilizar uno de estos sistemas de gestión de contenidos en la implementación de una aplicación web tiene varias ventajas:

- El uso de un CMS permite centrarse en el contenido y no en implementar soluciones a problemas ampliamente desarrollados en aplicaciones web como por ejemplo la administración de usuarios o la gestión de la conexión con una base de datos, por ejemplo.
- Se puede extender fácilmente las funcionalidades del sitio web instalando módulos, que son pedazos de código que añaden nuevas funciones. En un CMS esto es tan sencillo como instalar uno de estos módulos y aprender a configurarlo correctamente.
- Incorporan sistemas de control de acceso, que permiten la identificación de usuarios y otorgar permisos a estos con diferentes grados de acceso al sitio.
- Soportan la creación de contenidos, proporcionando herramientas genéricas que permiten configurar la edición de textos o la subida de imágenes, por ejemplo.
- Soportan multitud de sistemas de bases de datos, tanto de código abierto (como MySQL o PostgreSQL) como código propietario (Microsoft SQL Server).
- Algunos son auténticos frameworks, con APIs de desarrollo propias que permiten desarrollar y añadir *plugins*, componentes, módulos, etc. que extienden su funcionalidad.

Los tres CMS más populares son WordPress, Joomla y Drupal (todos ellos open source y basados en PHP). WordPress está enfocado a la creación de blogs, pero también permite crear webs sencillas haciendo uso de plantillas. Es muy intuitivo y fácil de utilizar para usuarios sin conocimientos técnicos.

Joomla es más potente que WordPress, y requiere más conocimientos técnicos. La curva de aprendizaje es mayor, y cuenta con un conjunto de plugins que pueden extender su funcionalidad permitiendo crear páginas webs de baja-media complejidad.

Drupal es un gestor de contenidos que pone a disposición del programador un framework de desarrollo que permite afrontar desarrollos complejos. Es el que tiene la curva de aprendizaje más pronunciada, pero a cambio es mucho más potente. Dispone de una gran cantidad de módulos adicionales creados por una extensa comunidad de usuarios que lo convierten en un CMS muy flexible y adaptable.



# Capítulo 3

## Análisis

En este capítulo nos centraremos en el análisis del sistema mediante la especificación de requerimientos y el modelado del comportamiento mediante casos de uso. En el primer apartado estableceremos los requisitos funcionales, definiendo las funciones que debe tener nuestra aplicación (comportamiento del sistema). En el segundo apartado explicaremos los requisitos no funcionales. En el tercer apartado estudiaremos la viabilidad del proyecto. En el cuarto apartado presentaremos la planificación temporal y finalmente, en el quinto y último apartado describiremos los casos de uso de nuestra aplicación.

### 3.1. Requisitos funcionales

Los requisitos funcionales son características que definen de manera concreta el funcionamiento que debe tener el sistema. Describe las funcionalidades que el sistema debe proporcionar a los usuarios. Normalmente estos requisitos surgen de entrevistas realizadas al cliente, que expone sus necesidades. En nuestro caso, estas necesidades se reflejan en los requerimientos del propio proyecto académico, así que nos basaremos en los objetivos expuestos en el primer capítulo. Las funcionalidades que podemos extraer de esos requerimientos son las siguientes:

- Registrarse en el sistema.
- Acceder al sistema / Cerrar sesión.
- Editar el perfil de usuario.
- Añadir material nuevo en la web.
- Acceder al material publicado en la web.
- Solicitar material publicado.
- Administrar el funcionamiento de la web

## 3.2. Requisitos no funcionales

Los requisitos no funcionales son aquellos que especifican criterios de carácter técnico y de calidad que se tienen que tener en cuenta en el desarrollo de un proyecto. En el caso de nuestro proyecto, estos requisitos no están definidos en cuanto que no hay restricciones impuestas al desarrollo del mismo. De manera general, podemos establecer los requisitos siguientes:

- La implementación de la web debería usar herramientas de software libre.
- El sistema debe tener una interfaz gráfica intuitiva y amigable que permita a los usuarios un fácil uso.
- Al tratarse de una aplicación web, esta debe ser compatible con los principales navegadores web.

## 3.3. Viabilidad del proyecto

En este apartado estudiaremos la viabilidad del proyecto teniendo en cuenta los aspectos técnicos, económicos y legales.

### 3.3.1. Viabilidad técnica

Sin restricciones previas en la implementación de nuestra aplicación web, nos hemos inclinado a utilizar un sistema de gestión de contenidos (CMS), que proporciona todas las funcionalidades básicas que se requieren en este proyecto, como la creación de usuarios o la publicación de contenidos por parte de estos. De entre la multitud de CMS existentes, nos hemos decidido por usar Drupal, basado en el lenguaje de programación PHP. Como base de datos usaremos MySQL. Los motivos para usar estas soluciones son:

- Tanto PHP como MySQL son tecnologías muy extendidas en la web.
- Drupal es uno de los tres CMS más populares basados en PHP. Cuenta con una gran comunidad de desarrollo detrás y con mucha información disponible.
- A nivel personal tengo experiencia previa en el uso de Drupal.

En cuanto al equipo necesario para realizar el proyecto, utilizaremos un ordenador personal equipado con el sistema operativo Linux, el software de servidor Apache y el sistema gestor de base de datos MySQL.

#### 3.3.2. Viabilidad económica

El equipamiento necesario para realizar el desarrollo es un ordenador personal aportado por el estudiante. Todo el software utilizado en el desarrollo de la aplicación web es software libre y de código abierto, por lo que no se requiere ningún desembolso económico en la adquisición o uso de licencias.

El proyecto es una asignatura de 15 créditos, lo que representa unas 300 horas. Si tenemos en cuenta una remuneración de unos 20€/hora, que se corresponde con el convenio laboral de la UAB para un ingeniero técnico superior de soporte a la investigación, y consideramos que hay una única persona desarrollando el proyecto, tendríamos un coste de unos 6000 €.

#### 3.3.3. Viabilidad legal

El software utilizado, como ya se ha comentado anteriormente, está basado en licencias de software libre, en sus diferentes versiones (licencias GPL, Apache, PHP), que permiten su libre uso.

Los datos de carácter personal, como en todo servicio, deben estar sujetos al cumplimiento de la ley de Protección de Datos de Carácter Personal (LOPD). El responsable de la web debe comprometerse, entre otras cosas, a tratar confidencialmente los datos de los usuarios y a poner los medios técnicos que impidan el uso o acceso no autorizado a esos datos.

#### 3.3.4. Conclusiones

Como hemos visto en los apartados anteriores, no hay aspectos en el desarrollo de este proyecto que impidan su buen desarrollo y funcionamiento. Por una parte, las tecnologías incluidas son ampliamente conocidas y utilizadas en el área del desarrollo web. Además, el uso de estas soluciones no implica un impacto económico, y el equipamiento necesario para el desarrollo es fácilmente accesible. Por último, los aspectos legales están cubiertos por licencias abiertas y el uso de medidas de protección adecuadas en el trato de la información personal de los usuarios. Podemos afirmar entonces que el desarrollo de este proyecto es viable.

## 3.4. Planificación temporal

### 3.4.1. Planificación inicial

En esta sección se expone la planificación temporal al inicio del proyecto, mostrando las tareas con las fechas previstas para su realización en la tabla 3.1. En la figura 3.1 mostramos el diagrama de Gantt representando el período de tiempo de Octubre de 2014 a Febrero de 2015.

Tareas	Fecha inicio	Fecha finalización
Análisis de los requerimientos	01/10/2014	15/10/2014
Análisis de las opciones de desarrollo	01/10/2014	15/10/2014
Realización del informe previo	15/11/2014	08/12/2014
Entrega del informe previo	12/12/2014	12/12/2014
Diseño de la base de datos	15/10/2014	31/10/2014
Diseño de la aplicación	15/10/2014	31/10/2014
Implementación de la aplicación	01/11/2014	31/12/2014
Testeo de la aplicación	01/12/2014	31/12/2014
Realización de la memoria	15/12/2014	15/01/2015
Preparación de la exposición oral	16/01/2015	31/01/2015
Entrega de la memoria	21/01/2015	27/01/2015
Exposición oral	02/02/2015	10/02/2015

Tabla 3.1: Planificación de tareas

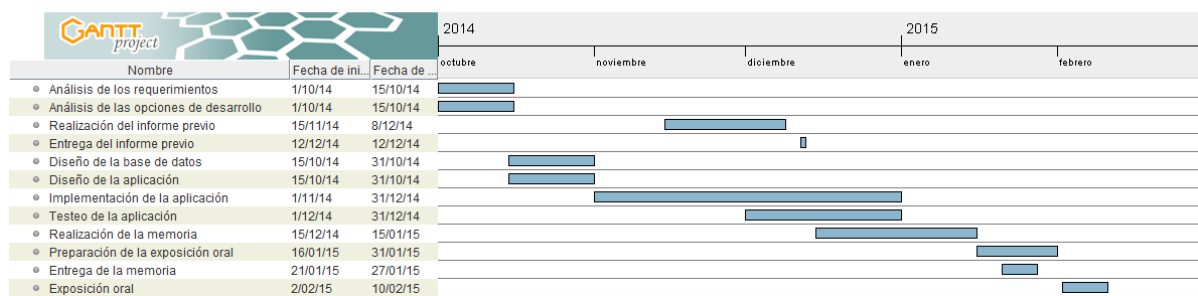


Figura 3.1: Planificación temporal (Octubre 2014 - Febrero 2015)

### 3.5. Especificación de los casos de uso

Una vez conocidos los requerimientos del proyecto, podemos realizar un modelo del sistema basado en la definición de casos de uso. Un caso de uso es una descripción de los pasos a seguir para poder llevar a cabo una función del sistema en su interacción con un agente externo, ya sea un usuario u otro sistema. Mediante los casos de uso se muestran los requerimientos de comportamiento para los requisitos funcionales.

La descripción de un caso de uso consta de los siguientes elementos:

- Descripción.
- Actores
- Precondición
- Flujo principal
- Flujos alternativos
- Postcondición

Para poder describir correctamente los casos de uso, debemos determinar quienes son los actores que participan en ellos. Un actor es un agente externo al sistema que se comunica con este. Nuestro sistema está formado por los siguientes actores:

- Usuario: se trata de una persona que desea interactuar con el sistema. Dependiendo de si está registrado o no, podrá realizar diferentes acciones:
  - Usuario anónimo: un usuario no registrado puede acceder a las funciones básicas del sistema, como visualizar los anuncios de material publicados o registrarse.
  - Usuario registrado: un usuario registrado puede acceder a todas las funciones que el sistema proporciona a su perfil, como publicar anuncios de material o hacer solicitudes de material.
- Administrador: se trata de la persona responsable de la aplicación y gestiona el sistema.

Con los actores definidos, pasamos a hacer una descripción de los diferentes casos de uso del sistema, que deben cubrir los requisitos funcionales que surgieron del análisis previo. En la figura 3.2 se puede ver el diagrama de casos de uso. A continuación, se hace una descripción detallada de cada uno de esos casos de uso.

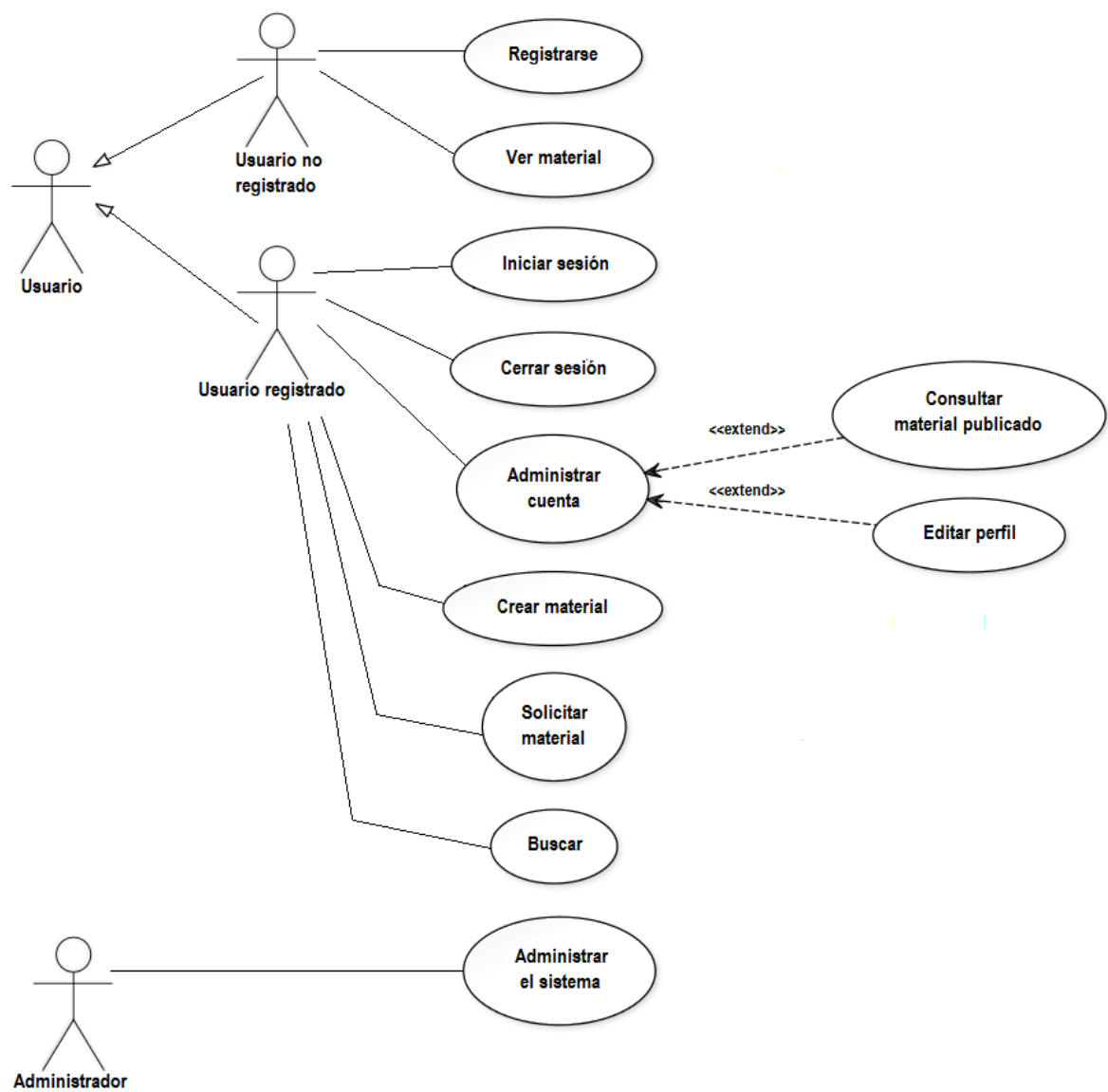


Figura 3.2: Diagrama de casos de uso

### 3.5.1. Caso de uso: Registrarse

Caso de Uso	Registrarse
Descripción	Registrar a un nuevo usuario en el sistema.
Actores	Usuario no registrado
Precondición	El usuario no esta registrado en el sistema.
Flujo principal	El nuevo usuario introduce un nombre de usuario y una dirección de correo electrónico. El sistema envía un mensaje a la dirección de correo del usuario con un enlace para completar el registro y una contraseña aleatoria.
Flujos alternativos	En el caso que ya exista el nombre de usuario o la dirección de correo, el sistema indicará el problema y permitirá introducir un nuevo valor.
Postcondición	El usuario queda registrado en el sistema.

Tabla 3.2: Registrarse

### 3.5.2. Caso de uso: Iniciar sesión

Caso de Uso	Iniciar sesión
Descripción	Ingresa al sistema identificando al usuario.
Actores	Usuario registrado
Precondición	La cuenta de usuario debe existir en el sistema.
Flujo principal	El usuario introduce su nombre de usuario y su contraseña, identificándose en el sistema.
Flujos alternativos	En el caso de ser incorrectos el nombre de usuario y/o la contraseña, el sistema informa del error y vuelve a solicitar los datos.
Postcondición	El usuario una vez identificado ingresa en el sistema.

Tabla 3.3: Iniciar sesión

### 3.5.3. Caso de uso: Cerrar sesión

Caso de Uso	Cerrar sesión
Descripción	El usuario quiere cerrar la sesión en el sistema.
Actores	Usuario registrado
Precondición	El usuario está identificado en el sistema.
Flujo principal	El usuario termina la sesión en el sistema. El sistema vuelve a la configuración de usuario no registrado.
Flujos alternativos	
Postcondición	La sesión del usuario en el sistema finaliza.

Tabla 3.4: Cerrar sesión

### 3.5.4. Caso de uso: Administrar cuenta

Caso de Uso	Administrar cuenta
Descripción	El usuario accede a los datos de su cuenta.
Actores	Usuario registrado
Precondición	El usuario está identificado en el sistema.
Flujo principal	El usuario selecciona la opción que le permite ver los datos asociados a su cuenta y el sistema muestra la información.
Flujos alternativos	El usuario puede editar su perfil o consultar el material que haya publicado.
Postcondición	El sistema muestra los datos asociados al usuario.

Tabla 3.5: Administrar cuenta

### 3.5.5. Caso de uso: Editar perfil de usuario

Caso de Uso	Editar perfil de usuario
Descripción	El usuario edita los datos de su perfil en el sistema.
Actores	Usuario registrado
Precondición	El usuario está identificado en el sistema y está en el apartado de administración de su cuenta.
Flujo principal	El usuario selecciona la opción de editar su perfil. El sistema muestra los datos asociados y el usuario realiza modificaciones.
Flujos alternativos	
Postcondición	Las modificaciones del perfil de usuario quedan registradas en el sistema.

Tabla 3.6: Editar perfil de usuario

### 3.5.6. Caso de uso: Consultar material publicado

Caso de Uso	Consultar material publicado
Descripción	El usuario consulta los materiales que ha publicado en el sistema.
Actores	Usuario registrado
Precondición	El usuario está identificado en el sistema y está en el apartado de administración de su cuenta.
Flujo principal	El usuario selecciona la opción de consultar el material publicado. El sistema muestra los datos asociados.
Flujos alternativos	
Postcondición	

Tabla 3.7: Consultar material publicado

### 3.5.7. Caso de uso: Crear material

Caso de Uso	Crear material
Descripción	El usuario crea un anuncio de material en el sistema.
Actores	Usuario registrado
Precondición	El usuario está identificado en el sistema.
Flujo principal	El usuario selecciona la opción de añadir nuevo material. El sistema muestra un formulario que el usuario debe rellenar con nombre, descripción, categoría a la que pertenece el material e imágenes. Una vez rellenado, se guardan los cambios y el anuncio queda publicado en el sistema.
Flujos alternativos	El usuario decide no publicar el material y cancela la acción.
Postcondición	Se crea una nueva entrada de material en el sistema.

Tabla 3.8: Crear material

### 3.5.8. Caso de uso: Ver material

Caso de Uso	Ver material
Descripción	El usuario consulta un anuncio de material.
Actores	Usuario registrado, Usuario no registrado
Precondición	
Flujo principal	El usuario (sea registrado o no) puede consultar una descripción detallada de cualquier material publicado en el sistema.
Flujos alternativos	
Postcondición	El sistema muestra una descripción detallada del material seleccionado.

Tabla 3.9: Ver material

### 3.5.9. Caso de uso: Solicitar material

Caso de Uso	Solicitar material
Descripción	El usuario solicita el material publicado en un anuncio.
Actores	Usuario registrado
Precondición	El usuario está identificado en el sistema y está en un anuncio de material.
Flujo principal	El usuario selecciona la opción de solicitar el material anunciado. El sistema se asegura que el usuario quiera solicitar el material con una pregunta de confirmación. Al confirmar la solicitud, el sistema envía un correo electrónico al autor de la publicación del material comunicándole la solicitud del usuario, así como su dirección de correo para que pueda ponerse en contacto con él.
Flujos alternativos	El usuario, al mostrarle la pregunta de confirmación, cancela la solicitud.
Postcondición	

Tabla 3.10: Solicitar material

**3.5.10. Caso de uso: Buscar material**

<b>Caso de Uso</b>	<b>Buscar material</b>
<b>Descripción</b>	El usuario busca una palabra o término en el sistema .
<b>Actores</b>	Usuario registrado, Usuario no registrado
<b>Precondición</b>	
<b>Flujo principal</b>	El usuario introduce una palabra en la caja de búsqueda y el sistema muestra los resultados que coinciden.
<b>Flujos alternativos</b>	El sistema no encuentra ninguna coincidencia con la palabra buscada y muestra un mensaje al usuario conforme su búsqueda no ha producido resultados.
<b>Postcondición</b>	El sistema muestra los resultados de la búsqueda.

Tabla 3.11: Buscar material

**3.5.11. Caso de uso: Administrar el sistema**

<b>Caso de Uso</b>	<b>Administrar el sistema</b>
<b>Descripción</b>	El administrador administra la configuración y funcionamiento del sistema.
<b>Actores</b>	Administrador
<b>Precondición</b>	El administrador está identificado en el sistema.
<b>Flujo principal</b>	El administrador selecciona el apartado de administración del sistema, pudiendo elegir entre un conjunto de operaciones a realizar.
<b>Flujos alternativos</b>	
<b>Postcondición</b>	El administrador gestiona la configuración del sistema.

Tabla 3.12: Administrar el sistema



# Capítulo 4

## Diseño e implementación

El diseño e implementación de un sitio web mediante un gestor de contenidos está más orientado a la configuración de las funciones que debe ofrecer que al clásico desarrollo de software que debe crearse desde cero. Estas funciones pueden conseguirse bien porque estén incluidas en los módulos del núcleo del CMS o bien añadiendo módulos que añadan nuevas funcionalidades que se adapten a nuestros requisitos. Una vez que el sitio incorpora todos los módulos necesarios, el siguiente paso es configurar adecuadamente las funciones que puede ofrecer cada módulo a nuestras necesidades.

En el primer apartado de este capítulo veremos el diseño del gestor de contenidos, y en el segundo apartado veremos como se ha implementado el sitio web a partir de las herramientas que proporciona Drupal.

### 4.1. Diseño

Drupal es un sistema de gestión de contenidos que incorpora una serie de funcionalidades básicas que permiten crear diferentes tipos de sitios, como webs comerciales de unas pocas páginas, blogs personales, forums o comunidades donde los usuarios generan el contenido. Estas funcionalidades básicas se conocen como núcleo de Drupal, y proporcionan la capacidad de crear contenido, crear y administrar usuarios, menús, y en general administrar el sitio web a través de una interfaz administrativa a la que se puede acceder a través de un navegador web.

El sistema está diseñado para poder extender estas funcionalidades básicas mediante el uso de módulos contribuidos por terceras partes, de manera que la creación y personalización de un sitio web se basa más en la configuración de dichos módulos que en la modificación del código. De todas maneras existe la posibilidad de desarrollar tus propios módulos; para ello Drupal pone a disposición de los usuarios una API de desarrollo [6].

En los siguientes subapartados vamos a explicar los elementos que forman parte del diseño de gestor de contenidos en el que basamos nuestra aplicación.

#### 4.1.1. Base de datos

El diseño de la base de datos en este proyecto viene asociado al hecho de usar un sistema de gestión de contenidos (CMS), que define su propio modelo de gestión de la información. Es en el proceso de instalación del sistema cuando se generan las tablas correspondientes al modelo que sigue el CMS. En el caso concreto de Drupal, estamos hablando de un conjunto de algo más de setenta tablas, que representan las diferentes entidades que componen la estructura del sistema [8].

Para implementar este diseño el CMS incluye una capa de manejo de base de datos basada en la librería *PHP Data Objects* (PDO) de PHP, que aporta una capa de abstracción para el acceso a los datos, de manera que acepta a diferentes gestores de bases de datos como MySQL, PostgreSQL o Microsoft SQL Server, entre otros.

Las tablas de esta base de datos gestionan los diferentes aspectos que componen el sistema. A grandes rasgos pueden agruparse en las siguientes categorías:

- Gestión de Usuarios: estas tablas gestionan toda la información referente a los usuarios (datos de perfil, sesiones, etc.)
- Gestión de Caches: estas tablas se encargan de gestionar el manejo de las diferentes caches que construye el gestor.
- Gestión del Sistema: estas tablas están relacionadas con la administración interna del sistema.
- Gestión de Nodos: estas tablas están relacionadas con la gestión de los nodos, que representan las piezas de contenido del sitio.
- Gestión de Bloques: estas tablas gestionan la disposición de piezas de información en la estructura gráfica del sitio.
- Gestión de Campos: estas tablas gestionan el almacenamiento de unidades de información, como imágenes o texto.
- Gestión de Taxonomías: estas tablas gestionan el sistema de clasificación de contenido.

### 4.1.2. Nodos

Los nodos representan el tipo de contenido básico en Drupal. Son piezas de información estructurada que puede representar cualquier tipo de contenido que se quiera crear en el sitio (entradas de blog, noticias, anuncios, etc.). Todos los nodos comparten un conjunto de campos en común, que son el autor, la fecha de creación, el título y el contenido. Además de estos campos, se pueden añadir otros que permitan definir tipos específicos de contenido. Esta manera de almacenar la información permite estandarizar la forma en que los módulos de Drupal manejan los datos.

### 4.1.3. Módulos

Como hemos comentado anteriormente, los módulos son los encargados de proporcionar funcionalidad al sitio web. La estructura de Drupal se basa en una librería de funciones comunes que proporcionan una base sobre la que funcionan los módulos, que son archivos de código PHP que suelen implementar una única funcionalidad.

Existen dos clases de módulos: los del núcleo y los contribuidos. Los módulos del núcleo aportan funcionalidades básicas, como la creación de contenido o la gestión de usuarios; los módulos contribuidos añaden funcionalidades adicionales, como la gestión gráfica de consultas a la base de datos, por ejemplo.

Salvo los módulos del núcleo requeridos para el funcionamiento básico de Drupal, el resto de módulos pueden ser activados o desactivados según las necesidades del sitio. Por ejemplo, el módulo de creación de blogs está incluido en la distribución básica, pero por defecto no viene activado.

Drupal hace uso del patrón de inversión de control; cuando se produce un suceso el sistema llama a los módulos que tienen interés en participar en la gestión de dicho suceso. Esto se implementa mediante el uso de *hooks*.

### 4.1.4. Hooks

Los hooks son funciones que permiten a los módulos participar en eventos del sistema. Cuando se produce un evento, Drupal lanza uno o varios hooks relacionados con ese evento, y todas las funciones concretas de los módulos que siguen el patrón de dichos nombres responden. Por ejemplo, cuando un usuario entra en el sistema mediante el login, Drupal llama a las funciones que sigan el patrón *hook\_user\_login*. Entonces, los módulos que implementen este hook responderán, como por ejemplo la función *comment\_user\_login* del módulo de comentarios, la función *node\_user\_login* del módulo de nodos, o, en general, cualquier función que siga el mismo patrón.

### 4.1.5. Bloques

Un bloque es información que puede ser situada en una región específica de la plantilla de nuestro sitio web, y permite personalizar la manera en que se muestra el contenido. Por ejemplo, un bloque puede mostrar el número de usuarios logueados en el sistema, y puede situarse en las regiones laterales. Puede configurarse para mostrarse únicamente en cierto tipo de páginas, o solo a determinado tipo de usuarios. Muchos módulos incluyen sus propios bloques; estos pueden ser activados y colocados en diferentes zonas a través de la página de administración de bloques del menú de administración.

### 4.1.6. Menús

Los menús son conjuntos de enlaces que permiten la navegación por el sitio. Drupal proporciona tres mecanismos básicos que permiten la navegación por el sitio web:

- Enlaces de texto embebidos en el contenido que dirigen al usuario a una página nueva.
- Imágene y botones que dirigen al usuario a una página nueva cuando se pulsan.
- Menús propiamente dichos, que son listas de enlaces dispuestas de manera horizontal o vertical. Drupal permite definir diferentes menús y colocarlos en diferentes regiones de la plantilla del sitio. Además, se pueden configurar estos menús para que aparezcan según los permisos que tengan sobre ellos los diferentes tipos de usuario.

### 4.1.7. Usuarios

La gestión de usuarios permite administrar quién hace qué en nuestro sitio web. Drupal provee la capacidad de crear usuarios, definir roles y gestionar permisos. Los roles son un mecanismo que permite definir categorías de usuarios. Drupal viene por defecto con tres tipos de usuario: anónimo, identificado y administrador. Los roles son una forma de agrupar tipos de usuario a los que se les quiere dar en conjunto una serie de permisos. Los permisos permiten asignar qué puede hacer un usuario y qué no puede hacer. En este aspecto, Drupal permite una gran granularidad en la concesión de permisos. Por ejemplo, se puede otorgar permiso a los usuarios identificados para que puedan crear nuevas entradas de contenido en el sistema, pero no permitirlo a los usuarios anónimos.

### 4.1.8. Temas

Los temas son colecciones de archivos que definen la capa de presentación del sitio. El sistema de temas de Drupal está compuesto de hojas de estilo, imágenes, ficheros Javascript y plantillas PHP que pueden modificar la salida por defecto del contenido. El objetivo de un tema es separar la lógica de negocio del sistema de los elementos de diseño.

El sistema de temas de Drupal utiliza un sistema de plantillas escritas en PHP que están formadas por un esqueleto HTML y sentencias en PHP que se encargan de los datos dinámicos. Las plantillas siguen el formato \*.tpl.php y se encargan de construir diferentes regiones del sitio, como puede verse en la figura 4.1.

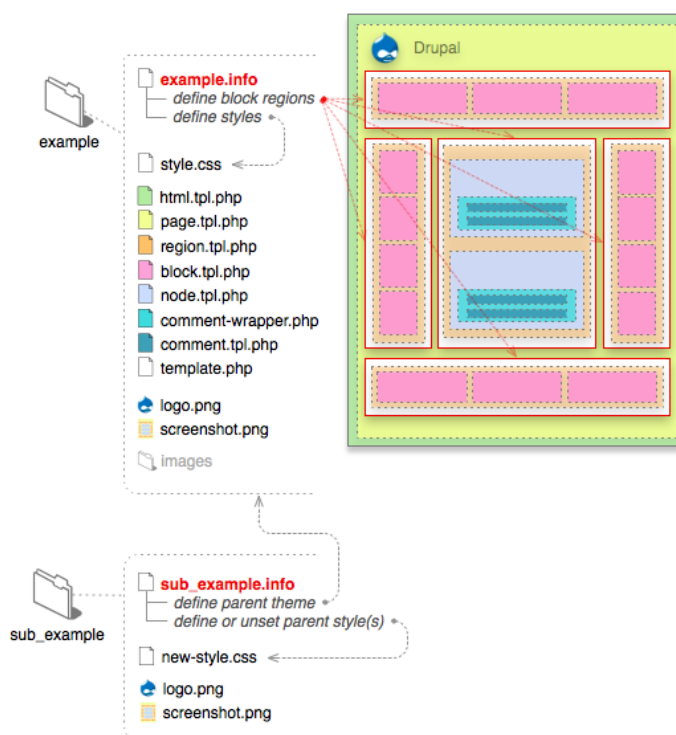


Figura 4.1: Plantillas tpl.php

Las partes básicas de cualquier tema son:

- Archivo .info: Este fichero incluye las rutas a los archivos CSS y define las regiones que forman parte del tema.
- Archivos de plantilla: Definen como se organiza el contenido de cualquier región del sitio. El nombre de cada plantilla indica el elemento que controla. Por ejemplo, la plantilla html.tpl.php define la estructura básica HTML de la página, mientras que la plantilla node.tpl.php controla la salida de los campos que forman un nodo, como puede ser campos de texto, imágenes, etc.
- Archivo template.php: Este archivo define una serie de funciones que se utilizan en las plantillas para procesar las variables que contienen información sobre el sitio.

- Archivos CSS: están formados por un conjunto de reglas que modifican el aspecto visual del sitio.
- Archivos Javascript: añaden funcionalidad a la interfaz de usuario.

## 4.2. Implementación

En los siguientes apartados veremos como se han implementado las funcionalidades de la aplicación, mediante la configuración del sistema, la creación de estructuras de contenidos nuevas y el uso de módulos contribuidos.

### 4.2.1. Creación de contenido

La instalación básica del gestor de contenidos proporciona dos tipos de contenidos básicos, páginas y artículos, que disponen de unos cuantos campos básicos. En el caso de nuestro sitio web, hemos creado un tipo de contenido nuevo, llamado Material, que se utiliza para publicar información sobre el material que se quiere anunciar. Los campos de los que consta Material son los siguientes:

- Título: descripción básica del material.
- Descripción: permite añadir cualquier información sobre el material que se anuncia, o en general, cualquier información que el usuario considere adecuada.
- Categoría: permite clasificar el material de entre una lista de términos. Este campo está basado en el concepto de taxonomía, que es la forma que Drupal utiliza para la clasificación del contenido que se crea en el sistema. En el caso de Categoría, se ha creado la siguiente lista de términos: Material de oficina, Mobiliario, Electrónica, Electrodomésticos, Informática, Herramientas y Otros.
- Imágenes: permite subir un número determinado de imágenes. Puede configurarse la cantidad, las dimensiones máximas y mínimas y el tamaño de los ficheros.

Para implementar un nuevo tipo de contenido el gestor proporciona una interfaz gráfica en la que se puede ir añadiendo los campos que necesitamos para representar las propiedades del contenido, como se puede ver en la figura 4.2

ETIQUETA	NOM-MÀQUINA	TIPUS DE CAMP	GINY	OPERACIONS	
✚ Nom	title	Element del mòdul Node			
✚ Descripció	body	Text llarg i resum	Àrea de text amb un resum	edita	suprimeix
✚ Categoria	field_categoria	Referència de terme	Llista de selecció	edita	suprimeix
✚ Imatges	field_imatges_material	Imatge	Imatge	edita	suprimeix
✚ Afegeix un nou camp					
<input type="text"/>		- Selecciona un tipus de camp -	- Selecciona un giny -		
Etiqueta		Tipus de dades a emmagatzemar.	Element de formulari per a editar-hi les dades.		
✚ Afegeix un camp existent					
<input type="text"/>		- Selecciona un camp existent -	- Selecciona un giny -		
Etiqueta	Camp a compartir		Element de formulari per a editar-hi les dades.		

Figura 4.2: Implementación de tipo de contenido Material

### 4.2.2. Solicitud de material

El mecanismo para solicitar un material publicado es el siguiente:

- Un usuario previamente registrado accede a un anuncio de material.
- Si está interesado en el material, lo solicita.
- El sistema le muestra un mensaje de confirmación preguntándole si quiere continuar con la solicitud.
- Si continua, el sistema envía un mensaje de correo al dueño comunicándole que el usuario está interesado en su material, así como su correo electrónico para que pueda ponerse en contacto con el.

Para implementar la solicitud de material publicado hemos utilizado dos módulos contribuidos, *Rules* y *Rules Link*. *Rules* es un módulo que nos permite configurar acciones condicionadas a la ejecución de eventos. *Rules Link* nos permite crear enlaces y configurar las acciones a ejecutar cuando se accede a estos con la colaboración del módulo *Rules*. En el caso de la solicitud de un material, hemos creado un enlace que se adjunta únicamente al tipo de contenido Material que hemos implementado anteriormente, y hemos añadido un paso de confirmación previo para asegurarnos que el usuario quiera realmente continuar con la solicitud. Para poder asegurarnos que solo los usuarios registrados puedan acceder a este enlace, hemos configurado el permiso correspondiente en la lista de gestión de permisos.

Una vez creado el enlace, hemos creado una regla de ejecución que consta de una condición previa y una acción. La condición previa es que el enlace solo puede mostrarse a usuarios que no son el propio autor del material. En el caso que se cumpla esta condición, y siempre que el usuario este registrado, podrá solicitar el material. En cuanto a la acción, el modulo *Rules* nos permite elegir entre un conjunto de acciones predeterminadas. En este caso, la acción será

que el sistema envíe un correo al autor del material. Para ello podemos configurar mediante el panel de administración el mensaje de correo, utilizando selectores de datos que nos permiten acceder a información del sistema como la dirección de correo del autor o del usuario.

### 4.2.3. Interfaz de usuario

La navegación a través de la aplicación se realiza mediante menús, que podemos definir mediante el panel de configuración gráfica que Drupal proporciona al administrador del sitio. En el caso de nuestro sitio web, tenemos un menú primario y un menú de navegación por categorías. El menú primario se encuentra en la parte superior de la aplicación y, en función de que tipo de usuario se conecte al sistema, da acceso a diferentes funcionalidades. Todos los usuarios tienen acceso al menú primario y al menú lateral de navegación. En el menú primario hemos definido un enlace a la página principal, que muestra el material publicado en el sistema sin distinción de categoría, y un enlace a una página que explica la motivación que hay detrás de la aplicación (al estilo de *Sobre nosotros*).

Además de estos enlaces básicos, si eres usuario registrado, se muestran tres enlaces más: uno para gestionar la propia cuenta, otro para poder añadir material nuevo y el último para salir del sistema. En cambio, si eres un usuario no registrado solo se muestran enlaces que permiten identificarse en el sistema o registrarse.

El menú de navegación por categorías se encuentra en el lateral izquierdo de la aplicación y permite navegar por las diferentes categorías de materiales publicados. Para poder implementarlo hemos añadido un módulo contribuido a la aplicación llamado *Taxonomy Menu*. Este módulo permite crear de manera automática enlaces a cada uno de los términos que forman parte de una taxonomía. La ventaja de este método es que si más adelante se quieren añadir nuevas categorías, el módulo genera automáticamente los enlaces a estos nuevos términos.

En la figura 4.3 puede verse una página de nuestro sitio donde aparecen las zonas de los menús y la zona de contenido principal.

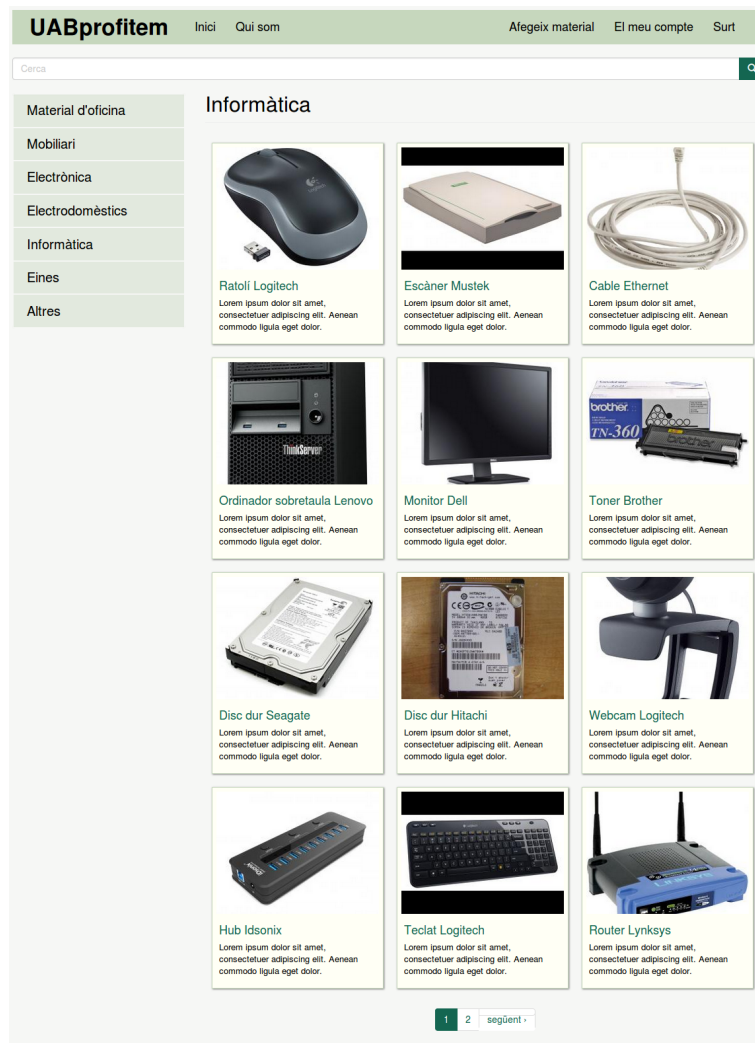


Figura 4.3: Pàgina de materials de informàtica per usuari identificat

El contenido tanto de la página inicial como de las páginas de categorías se crea a partir de la información de materiales almacenados en la base de datos. Para poder realizar consultas hemos utilizado un módulo llamado *Views*. Este módulo es un constructor gráfico de consultas a la base de datos, y es una alternativa a escribir consultas manualmente. En nuestra aplicación hemos creado consultas para mostrar el contenido de la página inicial y las páginas de categorías.

<b>TÍTOL</b> Títol: <a href="#">Cap</a>	<b>CONFIGURACIÓ DE PÀGINA</b> Camí: <a href="#">/taxonomy/term/%</a> Menú: <a href="#">Sense menú</a> Accés: <a href="#">Permís</a>   <a href="#">Accedeix al contingut publicat</a>
<b>FORMAT</b> Format: <a href="#">Llista sense format</a>   <a href="#">Paràmetres</a> Mostra: <a href="#">Camps</a>   <a href="#">Paràmetres</a>	<b>CAPÇALERA</b> <a href="#">Afegeix</a>
<b>CAMPS</b> <a href="#">Afegeix</a> ▼ Contingut: <a href="#">Imatges</a> Contingut: <a href="#">Títol</a> Contingut: <a href="#">Body</a>	<b>PEU</b> <a href="#">Afegeix</a>
<b>CRITERIS DE FILTRATGE</b> <a href="#">Afegeix</a> ▼ Contingut: <a href="#">Publicat (Sí)</a>	<b>PAGINADOR</b> Usa paginador: <a href="#">Complet</a>   <a href="#">Paged, 12 items</a> Enllaç "més": <a href="#">No</a>
<b>CRITERIS D'ORDENACIÓ</b> <a href="#">Afegeix</a> ▼ Contingut: <a href="#">Estàtic (desc)</a> Contingut: <a href="#">Data de publicació (desc)</a>	

Figura 4.4: Panel de creación de una consulta

Como puede verse en la figura 4.4, la interfaz de configuración de consultas se divide en varios apartados:

- **Título:** permite dar un nombre a la consulta que se mostrará con la vista.
- **Formato:** permite seleccionar diferentes formatos de salida para los campos de la consulta.
- **Campos:** en este apartado podemos seleccionar los campos del tipo de contenido que queremos que se muestren.
- **Criterios de filtrado:** este apartado nos permite realizar un filtrado del contenido en función de si se cumplen ciertas condiciones.
- **Criterios de ordenación:** en este apartado podemos seleccionar los criterios por los que se ordenará la salida de los datos, como por ejemplo la fecha de publicación de los materiales.
- **Configuración de la página:** en este apartado podemos configurar diferentes opciones, como la ruta de salida de la consulta o el tipo de acceso a la vista, en función de los permisos configurados para cada tipo de usuario.
- **Cabecera y pie:** en este apartado podemos añadir información relacionada con la vista en las zonas de cabecera y pie de la región donde se visualiza el resultado de la consulta, como por ejemplo el número de elementos devueltos por la consulta.

- Paginador: permite seleccionar el número de elementos de la consulta que se pueden ver por página.

#### 4.2.4. Gestión de usuarios

La gestión de usuarios es una funcionalidad que implementa por defecto Drupal. Mediante el panel de administración podemos configurar todos los aspectos de relacionados con la gestión de usuarios, como el proceso de registro o otorgar permisos de uso en función del tipo de usuario.

En nuestra aplicación tenemos tres tipos de usuario diferentes: usuario anónimo, usuario registrado y administrador. Para que un usuario pueda crearse una cuenta, hemos configurado mediante el panel de administración un tipo de registro no supervisado. En el proceso de registro, se pide al usuario un nombre de usuario y una dirección de correo electrónico. Una vez el usuario solicita registrarse, el sistema envía un correo a su dirección con un enlace de un único uso que permite al usuario establecer una contraseña para su cuenta.

Mediante el sistema de gestión de permisos hemos configurado qué opciones están disponibles para cada tipo de usuario. Este sistema despliega una lista en la que aparecen todas las acciones que pueden realizarse en el sitio web. Los permisos que hemos dado a cada rol son los siguientes:

- Usuario anónimo: puede ver el contenido del sitio y tiene acceso a la función de búsqueda de contenido, pero no puede solicitar materiales.
- Usuario registrado: puede publicar anuncios de material. En concreto puede crear, editar y suprimir contenido propio. Además, puede solicitar material de otros anunciantes.
- Administrador: puede administrar todos los aspectos del sitio web.

#### 4.2.5. Estilo

La instalación por defecto de Drupal proporciona varios temas que pueden administrarse desde el panel de administración. En el caso de nuestra aplicación, hemos utilizado un tema contribuido basado en el framework Bootstrap [7], que proporciona elementos de diseño basados en HTML, CSS y Javascript y permite que el diseño de la página se adapte al tamaño de la pantalla. Este tema adapta la distribución de Bootstrap al sistema de temas de Drupal, añadiendo plantillas PHP personalizadas que sobrescriben las plantillas básicas de Drupal, como por ejemplo las plantillas `html.tpl.php` o `page.tpl.php`.

Para modificar el aspecto de nuestro sitio, hemos creado un archivo CSS llamado `style.css` donde hemos creado nuestras propias reglas de estilo para modificar el aspecto por defecto

que proporciona Bootstrap, como por ejemplo la paleta de colores del sitio, la tipografía o las vistas previas de los materiales publicados.

También hemos utilizado un módulo llamado *Colorbox* que permite a los usuarios ver una versión ampliada de las imágenes que hay en una entrada de material publicado sin la necesidad de ir a otra página. Estas imágenes se muestran en una ventana que flota sobre el resto de la página, que queda oscurecida.

Tanto el tema basado en Bootstrap como la funcionalidad del módulo Colorbox se basan en el uso de jQuery, una librería de Javascript que simplifica la manera de interactuar con los documentos HTML.

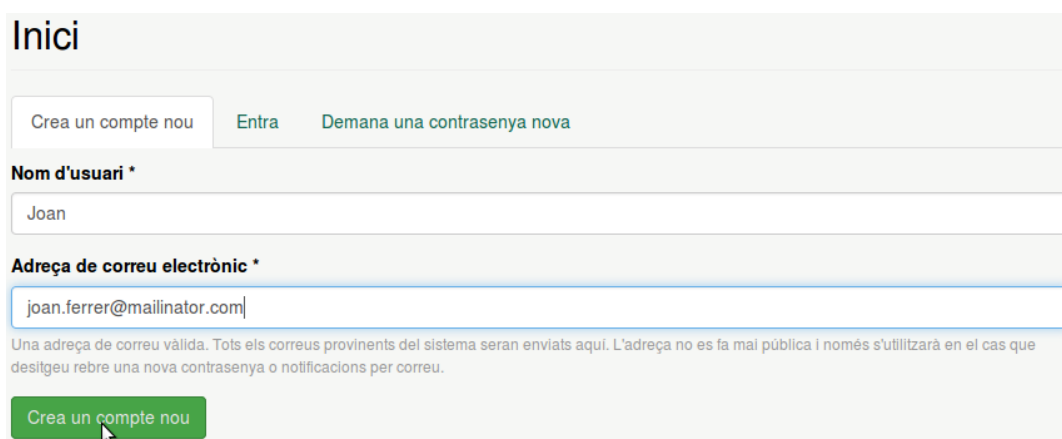
# Capítulu 5

## Pruebas

A continuación veremos las diferentes pruebas que hemos realizado para comprobar que el funcionamiento de la aplicación sea correcto.

### 5.1. Usuarios

Una de las pruebas realizadas ha sido comprobar que funcione correctamente el proceso de creación de nuevos usuarios. Para crear un nuevo usuario se pide un nombre de usuario y una dirección de correo electrónico. Si alguno de los datos que introduce el usuario ya existen en el sistema, se muestra un mensaje de aviso para que se cambie el valor. En cambio, si los datos son nuevos, el sistema envía un correo a la dirección proporcionada en el que facilita un enlace a nuestro sitio donde puede continuar con el proceso. Al seguir el enlace, el usuario puede crear una contraseña para su cuenta. En las figuras 5.1, 5.2 y 5.3, podemos ver como se desarrolla el proceso de creación de un nuevo usuario.



The screenshot shows a web form titled "Inici" (Start) for user registration. At the top, there are three links: "Crea un compte nou" (Create a new account), "Entra" (Log in), and "Demana una contrasenya nova" (Request a new password). Below these, the form has two main sections. The first is labeled "Nom d'usuari \*" (Username) and contains a text input field with the value "Joan". The second is labeled "Adreça de correu electrònic \*" (Email address) and contains a text input field with the value "joan.ferrer@mailinator.com". Below the email field, there is a small note in Catalan: "Una adreça de correu vàlida. Tots els correus provinents del sistema seran enviats aquí. L'adreça no es fa mai pública i només s'utilitzarà en el cas que desitgeu rebre una nova contrasenya o notificacions per correu." (A valid email address. All emails from the system will be sent here. The address is never made public and will only be used in the case that you wish to receive a new password or notifications by email). At the bottom of the form, there is a green button labeled "Crea un compte nou" (Create a new account).

Figura 5.1: Introducción de usuario y correo

To: [joan.ferrer](#) March Original Forward Delete

From: [uabprofitem@gmail.com](#)

Subject: **Joan, has creat un nou compte a UABprofitem**

Received: Tue March 31 2015 16:07:06 GMT+0200 (CEST)

---

Joan,

Gràcies per registrar-te a UABprofitem. Ara pots iniciar sessió fent clic a l'enllaç següent o copiant-lo i enganxant-lo al teu navegador:

<http://localhost/uabprofitem/user/reset/49/1432591619/aPTTrluU9tg1gxRMVD2Z0EwM2QMDD50UeshWoSnrpaE>

Aquest enllaç només es pot utilitzar un sol cop per a iniciar sessió i et dirigirà a una pàgina on podràs establir la teva contrasenya.

Un cop hakis establert la teva contrasenya, podràs iniciar sessió a <http://localhost/uabprofitem/user> utilitzant:

nom d'usuari: Joan  
contrasenya: La vostra contrasenya

-- L'equip de UABprofitem

Figura 5.2: Mensaje de correo con enlace

**Joan**

Acabeu d'utilitzar el vostre enllaç d'inici de sessió d'un sol cop. Ja no és necessari utilitzar aquest enllaç per iniciar sessió. Canvieu la vostra contrasenya. ✕

MostraEditaMaterial publicat

**Adreça de correu electrònic \***

joan.ferrer@mailinator.com

Una adreça de correu vàlida. Tots els correus provinents del sistema seran enviats aquí. L'adreça no es fa mai pública i només s'utilitzarà en el cas que desitgeu rebre una nova contrasenya o notificacions per correu.

**Contrasenya**

\*\*\*\*\*

**Confirma la contrasenya**

\*\*\*\*\*

Complexitat de la contrasenya: **Feble**

Les contrasenyes concorden: sí

Figura 5.3: Introducción de contraseña

## 5.2. Creación de contenido

Hemos realizado pruebas para comprobar que la creación de contenido funciona correctamente. Para poder publicar materiales, el usuario debe rellenar una serie de campos de un formulario:

- Nombre: este campo solo admite texto plano.
- Descripción: este campo admite tanto texto plano como HTML filtrado, que permite utilizar algunas etiquetas HTML. Aquí hemos realizado pruebas para comprobar que solo interpreta las etiquetas HTML predeterminadas en caso que este seleccionada esta opción. En caso de estar seleccionada la entrada de texto plano, hemos comprobado que no interpreta ninguna etiqueta HTML.
- Categoría: este campo es una lista desplegable, así que solo se admiten los valores de dicha lista.
- Imágenes: en este campo hemos realizado pruebas para comprobar que se cumplen las restricciones que hemos configurado. Hemos comprobado que se admiten un máximo de tres imágenes. También hemos comprobado que el tamaño máximo de una imagen no puede sobrepasar 1 MB. Por último, hemos comprobado que solo se admiten los formatos de imágenes png, gif, jpg y jpeg. Cuando no se cumplen alguna de estas restricciones el sistema muestra un mensaje informando del error concreto.

En las figuras 5.4, 5.5 y 5.6, podemos ver el proceso introducción de datos y la publicación del contenido.

## Crea Material

**Nom \***

**Descripció (Edita el resum)**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus.

Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.

Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus.

Més informació sobre els formats de text

**Format del text**

Filtered HTML

- Les adreces de pàgines web i de correu electrònic es tornen automàticament en enllaços.
- Etiquetes HTML permeses: <a> <em> <strong> <cite> <blockquote> <code> <ul> <ol> <li> <dl> <dt> <dd>

- Les línies i paràgrafs es trenquen automàticament.

Figura 5.4: Campos nombre y descripción




**Categoria \***

Material d'oficina

**Imatges**

Pots afegir fins a tres imatges del material que vols anunciar (com a mínim és obligatori afegir una imatge). S'accepten imatges de com a màxim 1 MB.

[Mostra el pes de les files](#)

Informació del fitxer	Operacions
   silla.jpg (29.36 KB)	<a href="#">Suprimeix</a>

**Afegeix un nou fitxer**

No se ha seleccionad ningú archivo.

Els fitxers han de ser de menys de 1 MB.  
Tipus de fitxers permesos: png gif jpg jpeg.

Figura 5.5: Campos categoría e imágenes

## Cadira amb rodes

S'ha creat Material *Cadira amb rodes*.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus.

Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo.

Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus.

04/05/2015 - 13:56  
Dani




Figura 5.6: Material publicado

### 5.3. Solicitud de material

Otra de las pruebas realizadas ha sido comprobar que el proceso de solicitud de material funciona correctamente. Hemos comprobado que solo un usuario registrado puede solicitar un material. Gracias a la configuración de permisos, a un usuario no registrado no le aparecerá el botón de solicitud en el anuncio. De la misma manera, tampoco aparece esta opción al autor del contenido, ya que no es aplicable.

Al solicitar un material publicado, el sistema comprueba mediante un mensaje si el usuario confirma que quiere continuar con la solicitud. En caso positivo, el sistema envía un correo electrónico al autor del anuncio indicando el nombre del usuario interesado, su dirección de correo electrónico y el material en el que está interesado. En caso contrario se aborta el proceso.

En las figuras 5.7, 5.8 y 5.9, podemos ver el proceso de solicitud de un material publicado.

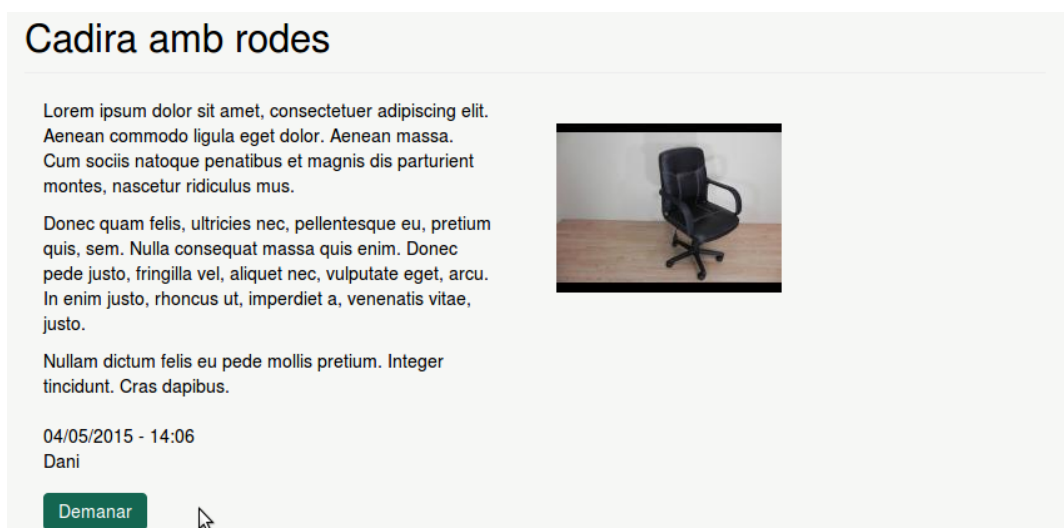


Figura 5.7: Solicitud de material

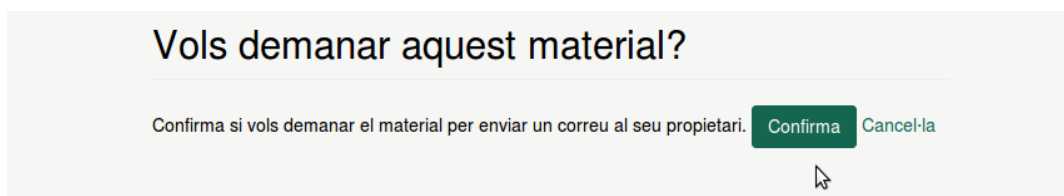


Figura 5.8: Confirmación de solicitud



Figura 5.9: Mensaje de correo



# Capítulo 6

## Conclusiones

### 6.1. Objetivos

El objetivo de este proyecto ha sido la creación de una aplicación web con un conjunto de funcionalidades básicas que permitiesen al personal de los departamentos de la UAB gestionar las donaciones de material no utilizado. Las funcionalidades básicas que se requieren son la gestión de usuarios, el soporte a la publicación de material y la solicitud del mismo. Estos requerimientos se han alcanzado haciendo uso de un sistema de gestión de contenidos, que permite ahorrar tiempo de desarrollo y facilita herramientas que de otra manera deberían ser programadas desde cero.

En resumen, los objetivos de este proyecto y los logros alcanzados son los siguientes:

- **Análisis y diseño de la base de datos de usuarios y materiales.**  
La base de datos que soporta el contenido de nuestro sitio web es el resultado del modelo interno de gestión de datos aplicado por el gestor de contenidos que hemos usado, en este caso Drupal. En este aspecto, no hemos tenido que diseñar una base de datos propia.
- **Análisis y diseño de la gestión de usuarios.**  
El registro de usuarios y la identificación en el sistema, así como la gestión de permisos de usuario son funcionalidades que nos ofrece el gestor de contenidos y que se ha configurado adaptándolas a las necesidades del proyecto.
- **Análisis y diseño de la introducción de datos de materiales y subida de imágenes.**  
Para este objetivo se ha diseñado un tipo de contenido compuesto por una serie de campos que permiten la subida tanto de información en formato texto como de imágenes.
- **Análisis y diseño de la resolución de peticiones de material.**  
Esta funcionalidad se ha planteado como un sistema de “anuncios clasificados”, en que

el dueño del material publica un anuncio y si una persona está interesada, el sistema envía un correo al dueño que le permite ponerse en contacto con el interesado.

- **Implementación de la aplicación web.**

Se ha implementado una interfaz sencilla e intuitiva, que permite navegar fácilmente por el contenido de la web. La gestión de usuarios, la publicación de materiales y la solicitud de estos se han implementado mediante el uso de las funciones básicas que ofrece del gestor de contenidos y la instalación de módulos que extienden sus capacidades básicas.

- **Testeo del sistema.**

Las pruebas realizadas en el sistema han comprobado el buen funcionamiento de este, cumpliendo con las funcionalidades que nos hemos propuesto en los puntos anteriores. Este objetivo se ha cumplido.

## 6.2. Planificación final

En esta sección se expone la planificación temporal final del proyecto, mostrando las tareas realizadas en la tabla 6.1. En las figuras 6.1 y 6.2 mostramos el diagrama de Gantt representando el período de tiempo de Octubre de 2014 a Julio de 2015.

Tareas	Fecha inicio	Fecha finalización
Análisis de los requerimientos	01/10/2014	15/10/2014
Análisis de las opciones de desarrollo	01/10/2014	15/10/2014
Realización del informe previo	15/11/2014	08/12/2014
Entrega del informe previo	12/12/2014	12/12/2014
Diseño de la base de datos	15/10/2014	31/10/2014
Diseño de la aplicación	15/10/2014	31/10/2014
Implementación de la aplicación	01/11/2014	31/12/2014
Testeo de la aplicación	01/12/2014	31/12/2014
Realización de la memoria	15/12/2014	12/06/2015
Preparación de la exposición oral	15/06/2015	25/06/2015
Entrega de la memoria	15/06/2015	18/06/2015
Exposición oral	26/06/2015	07/07/2015

Tabla 6.1: Planificación final de tareas

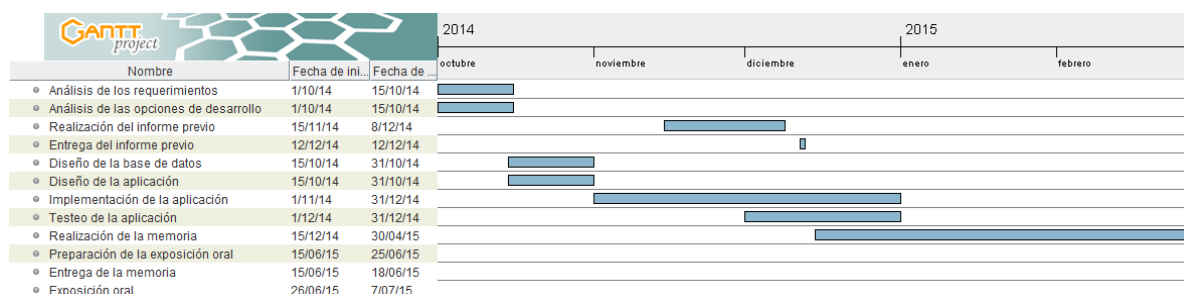


Figura 6.1: Planificación temporal (Octubre 2014 - Febrero 2015)

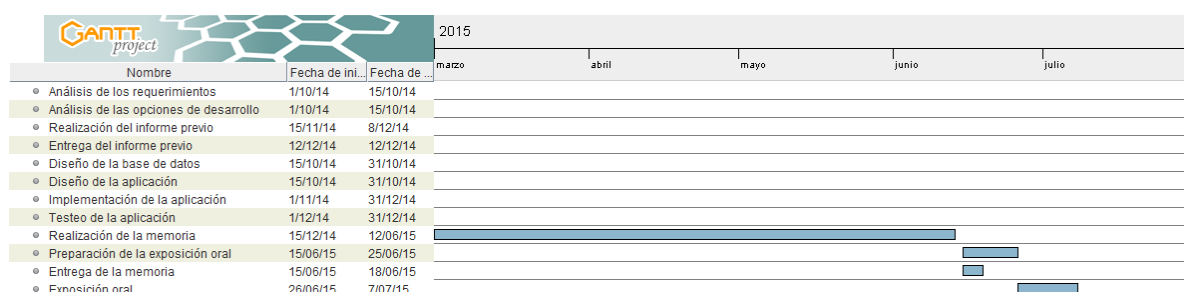


Figura 6.2: Planificación temporal (Febrero 2015 - Julio 2015)

## 6.3. Observaciones

Este proyecto me ha permitido ampliar conocimientos de tecnologías básicas como HTML, PHP, Javascript y CSS, por una parte, y profundizar en el conocimiento del manejo de un gestor de contenidos. También me ha hecho darme cuenta de la complejidad de llevar un proyecto a cabo. La planificación de las diferentes etapas es algo que sin experiencia cuesta abordar, y fácilmente te quedas corto estimando la duración. Pero en general encuentro una experiencia positiva el crear algo desde la primera idea hasta ver el resultado final.

## 6.4. Futuras ampliaciones

Algunas posibles ampliaciones que podemos sugerir son:

- La interfaz de la aplicación está en catalán, pero podría ampliarse a otros idiomas como el castellano o el inglés.
- Se puede ampliar el rango de categorías en las que se pueden clasificar los materiales. Las que hemos incluido en esta aplicación creemos que engloban los tipos de mate-

riales que pueden intercambiarse, pero pueden surgir categorías muy utilizadas que no hayamos previsto.

- Desarrollar una versión para dispositivos móviles. Nuestra aplicación es capaz de adaptar la presentación del sitio a diferentes resoluciones, pero una aplicación específicamente desarrollada para un dispositivo puede obtener mejores resultados.

# Bibliografía

- [1] TOMLINSON, Todd. *Beggining Drupal*. Apress, 2010
- [2] TOMLINSON, Todd. *Pro Drupal 7 Development*. Apress, 2010
- [3] HODGON, Jennifer. *Programmer's guide to Drupal*. O'Reilly, 2012
- [4] Content management system, abril 2015  
<[http://en.wikipedia.org/wiki/Content\\_management\\_system](http://en.wikipedia.org/wiki/Content_management_system)>
- [5] Drupal concepts, abril 2015  
<<https://www.drupal.org/documentation/concepts>>
- [6] API reference, abril 2015  
<<https://api.drupal.org/api/drupal>>
- [7] Bootstrap theme, abril 2015  
<<https://www.drupal.org/project/bootstrap>>
- [8] Drupal 7 database entity relationship diagram, abril 2015  
<[https://www.drupal.org/files/er\\_db\\_schema\\_drupal\\_7.png](https://www.drupal.org/files/er_db_schema_drupal_7.png)>
- [9] Drupal Community Documentation, abril 2015  
<<https://www.drupal.org/documentation>>
- [10] Drupal 7.37, abril 2015. Disponible en:  
<<http://ftp.drupal.org/files/projects/drupal-7.37.tar.gz>>



---

Firmado: Daniel Valverde Saavedra  
Bellaterra, 15 de Junio de 2015

## **Resum**

Aquest projecte s'ha desenvolupat per oferir una eina de publicació de materials que els seus propietaris ja no necessiten i que poden ser reutilitzats per altres persones. Es tracta d'una aplicació web que permet gestionar la creació i identificació d'usuaris, així com donar suport a la publicació del material que es vol anunciar.

## **Resumen**

Este proyecto se ha desarrollado para ofrecer una herramienta de publicación de materiales que sus propietarios ya no necesitan y que pueden ser reutilizados por otras personas. Se trata de una aplicación web que permite gestionar la creación e identificación de usuarios, así como dar soporte a la publicación del material que se quiere anunciar.

## **Abstract**

This project was developed to provide a tool for publishing materials that their owners no longer need and that can be reused by others. It is a web application for managing the creation and identification of users and support the publication of material to be announced.