

UNIVERSIDAD DE ALCALÁ

Escuela Técnica Superior de Ingeniería Informática

INGENIERÍA INFORMÁTICA



Trabajo Fin de Carrera

Implantación de un Museo Virtual en el Real Observatorio de Madrid

Héctor Navarro Martín
2010

UNIVERSIDAD DE ALCALÁ

Escuela Técnica Superior de Ingeniería Informática

INGENIERÍA INFORMÁTICA

Trabajo Fin de Carrera

**Implantación de un Museo Virtual en el Real Observatorio de
Madrid**

Alumno: Héctor Navarro Martín

Director: José Javier Martínez Herráiz

Tribunal:

Presidente:

Vocal 1º:

Vocal 2º:

Calificación:

Fecha:

Agradecimientos

A HBO, AMC y ABC, por hacerme más ameno el trabajo.

Índice general

1. Resumen	13
1.1. Resumen	15
2. Memoria	17
2.1. Introducción	19
2.2. Base teórica	19
2.2.1. Estado del Arte	19
2.2.1.1. ¿Qué puede ser un museo virtual?	19
2.2.1.2. Museos	20
2.2.1.2.1. Tate Modern (Londres, Reino Unido)	20
2.2.1.2.2. MoMA (New York, EEUU)	21
2.2.1.2.3. Actual museo del OAN	21
2.2.1.3. Enciclopedias	21
2.2.1.3.1. Wikipedia	22
2.2.1.3.2. Indianopedia, Frikipedia, *pedia	22
2.2.1.4. Sistemas gestores de contenidos vs Wikis	22
2.3. Descripción del diseño	22
2.3.1. Baremación de alternativas	22
2.3.1.1. Relación de características	23
2.3.1.2. Comentarios	23
2.3.2. Comparativa	24
2.3.2.1. Premisas iniciales	24
2.3.2.2. Candidatos pre-seleccionados	24
2.3.2.3. Comparativa con CMSMatrix	25
2.3.2.3.1. System Requeriments (Requisitos de sistema)	25
2.3.2.3.2. Security (Seguridad)	26
2.3.2.3.3. Support (Asistencia)	26
2.3.2.3.4. Ease of Use (Facilidad de uso)	26
2.3.2.3.5. Performance (Rendimiento)	26
2.3.2.3.6. Management (Administración)	26
2.3.2.3.7. interoperability (Interoperatividad)	28
2.3.2.4. Propuesta preliminar	31
2.3.2.4.1. Liferay Portal (Community Edition)	31
2.3.2.4.2. Drupal	32
2.3.2.4.3. Moodle	32
2.3.2.4.4. Plone	32
2.3.3. Anexo a comparativa	32
2.3.3.1. Breve análisis de Alfresco	32
2.3.4. Modelo de datos	33

2.3.4.1.	Catálogo	34
2.3.4.1.1.	OBJETO	34
2.3.4.1.2.	ÁREA	34
2.3.4.1.3.	TIPO_OBJETO	34
2.3.4.1.4.	ARTÍCULOS	35
2.3.4.1.5.	AREA_ARTÍCULO	35
2.3.4.1.6.	INVENTOR	35
2.3.5.	Drupal	35
2.4.	Conclusiones y trabajo futuro	36
3.	Planos	37
3.1.	Esquemáticos	39
4.	Pliego de condiciones	41
4.1.	Requisitos de Hardware	43
4.2.	Requisitos de Software	43
5.	Presupuesto	45
5.1.	Presupuesto	47
6.	Manual de Usuario	49
6.1.	Manual	51
7.	Soporte informático	53
8.	Bibliografía	55

Índice de figuras

2.1. System Requeriments desde CMSMatrix.org	25
2.2. System Requeriments desde CMSMatrix.org	27
2.3. System Requeriments desde CMSMatrix.org	28
2.4. System Requeriments desde CMSMatrix.org	29
2.5. System Requeriments desde CMSMatrix.org	29
2.6. System Requeriments desde CMSMatrix.org	30
2.7. System Requeriments desde CMSMatrix.org	31

Índice de tablas

Capítulo 1

Resumen

1.1. Resumen

El objetivo principal de este proyecto es la implantación de una solución software que desempeñe funciones de museo virtual para el Real Observatorio de Madrid. Además, comprenderá una plataforma de gestión de contenidos operable por personal no informático. Para ello, la solución propuesta es la búsqueda comparada, elección e integración de una plataforma de gestión de contenidos (CMS), añadiendo, si se da el caso, las funcionalidades no incluidas en el paquete escogido pero sí en los requisitos recogidos de la empresa. Se hace especial hincapié en la reducción del mantenimiento posterior, la accesibilidad móvil, la facilidad de uso para personal no técnico y la posibilidad de modificar y/o añadir funcionalidades en un futuro.

Palabras clave: cms, museo virtual, móvil, wiki

Capítulo 2

Memoria

2.1. Introducción

El trabajo expuesto en este documento tiene como fin la implantación de un museo virtual en el Observatorio de Madrid. La parte más importante en este tipo de proyectos (y, por extensión, de todos) es la recogida de requisitos y el análisis (en este caso, elección principalmente). Para poder entender mejor lo que se solicita por parte del cliente es necesario conocer el concepto de “museo virtual” al que tratamos de acercarnos. Se trataría de una plataforma accesible físicamente desde puestos instalados en una sala de exposiciones dentro del propio observatorio, accesible en línea desde Internet y que comprenda una base de datos posiblemente reutilizable en el futuro. Con estas pinceladas he deducido que la solución existente (no hay que olvidar que esto es ingeniería) más cercana a lo que se nos pide pasaría, en mi opinión, por un gestor de contenidos o CMS (Content Management System). Dada la amplísima oferta que hay de estos productos en el mercado, creo conveniente hacer una selección a partir de criterios, unos surgidos y otros sugeridos, analizados mano a mano con el cliente. A partir de ahí, se escogerá el CMS candidato con mejores cualidades en cuanto a facilidad de uso, bajo mantenimiento y características incluidas por defecto. Las características no comprendidas en esta selección (pero necesarias igualmente) serán incorporadas junto con las diferentes modificaciones previstas para darle forma final a la plataforma. Posteriormente, se integrará y se pondrá en servicio con los plazos que el cliente disponga para ello.

2.2. Base teórica

Al tratarse de un proyecto tan práctico, la base teórica del mismo no tiene un gran misterio. Se trata de realizar el clásico enfoque de presentar distintas aplicaciones al cliente cumpliendo unos requerimientos mínimos y adaptando e integrando la solución escogida para que encaje con el sistema solicitado.

En esta ocasión, presentaremos varios subapartados según las distintas fases del proyecto.

2.2.1. Estado del Arte

Para elaborar este proyecto pensé que lo más sensato era, una vez acotado el problema, tratar de echar un vistazo a cómo estaban las cosas en la actualidad en otros museos. Considero importante saber en qué estado se encuentran tanto el círculo de los CMS, wikis, museos y catálogos informatizados. En este periplo, he topado con varias fuentes de información desconocidas para mí, algunas ideas sacadas de museos visitados en persona y una exhaustiva (aunque difícil de discernir) cantidad de datos sobre gestores de contenido.

2.2.1.1. ¿Qué puede ser un museo virtual?

Al hablar de museos virtuales, a uno se le pueden venir a la mente distintos conceptos sobre lo que representan esas palabras:

- Visita virtual al museo (modo “Second Life”).
- Galería de imágenes.
- Sitio web del museo (meramente informativo).

- Catálogo de objetos.
- Ampliación de lo expuesto en el museo.

En el primer caso, nos referimos a esa idea de recrear a través de gráficos en 3D la experiencia de una visita real al museo. Esta solución es muy poco usada debido a lo costoso de la misma. Además de modelar infográficamente la arquitectura del museo y construir una aplicación que utilice un potente motor gráfico para sostenerla, hay que tener en cuenta el gran tráfico generado para que esto funcione adecuadamente. En mi opinión, dado que difícilmente se puede reemplazar la experiencia de visitar el museo en persona, creo que no compensa tanto despliegue de medios ya que no contribuye (al menos en el caso expuesto) sensiblemente al objetivo marcado.

El segundo caso, una mera galería de imágenes, aunque pobre, podría vale como museo virtual para un museo de arte en según qué circunstancias. Obviamente, y aunque la presentación de la información en este formato sea algo a tener en cuenta, si limitáramos nuestro museo virtual a este tipo de sitio, estaríamos, creo yo, limitando muchísimo la información ofrecida, cosa que no cuadra con el objetivo de complementar el museo.

El tercer caso lo he incluido como posibilidad de que alguien lo identificase al hablar de “museos virtuales”, dado que crear una página con información corporativa, de horarios y accesos es sólo una parte de lo que pretendería este proyecto.

El cuarto caso será aquel en el que básicamente tenemos una somera base de datos con una sencilla interfaz de búsqueda y acceso a la colección, sin tratar de explicar o poner en contexto los objetos que tenemos delante.

El quinto caso es el que nos ocupa, siendo el cometido del mismo complementar la exposición física, permitiendo a los visitantes (y no visitantes) conseguir un trasfondo y una explicación más completa de los objetos y su contexto para contribuir a la divulgación de un conocimiento relacionado con lo expuesto. Obviamente incluirá parte de galería de imágenes en cuanto a interfaz, información de típico sitio web como caso anecdótico de contenido, componentes que mejoren la interactividad sin sacrificar el desempeño y el coste y datos concretos sobre los objetos como manera de acceder a la base de datos.

2.2.1.2. Museos

Aprendiendo de museos conocidos, podemos poner ejemplos prácticos de los que sacar ideas o matices que el cliente pueda identificar como deseables. Está claro que sería imposible abarcar todos o siquiera la mayoría de los museos del mundo, ni siquiera los más importantes (¿quién decide esa lista?), pero sin embargo sí se pueden incluir ideas que hayamos rescatado de otras implementaciones satisfactorias. Algunos ejemplos son:

2.2.1.2.1. Tate Modern (Londres, Reino Unido) Pude comprobar en directo algunas de las ideas puestas en práctica por esta entidad gracias a un viaje reciente. Allí pude ver buenas ideas como el acercamiento de los contenidos de las salas al gran público (en especial, a niños) a base de actividades interactivas que introducían con naturalidad y entretenimiento conceptos difíciles de asimilar de otra manera. Sin duda es una buena idea, aunque no se encuadra en este proyecto, podría ser considerada para futuros trabajos.

Más concretamente, en materia de museos virtuales, vi la adopción de terminales de mano que complementan la experiencia del visitante físico. Estamos acostumbrados a ver las típicas guías locutadas en diversos idiomas que nos amplían el contenido de las exposiciones introduciendo

un número del índice de grabaciones. Suelen ser aparatos con usos muy restringidos (especiales para museos) con lo que eso supone para el coste final para el museo. En cambio, en este caso, se trataba de dispositivos móviles que están en el mercado, en nuestro caso, una especie de PDA de marca Toshiba o Dell (según renovaciones). Son dispositivos básicos que permiten mostrar contenidos web y multimedia sin muchas complicaciones. Esto es: pantalla en color, interfaz intuitiva, salida de auriculares y conectividad inalámbrica. Este último punto permite renovar sin coste temporal importante (y de manera transparente al visitante) los contenidos presentados. El hecho de que sea un dispositivo común abarataría los costes, permitiendo, incluso, trabajar con tecnología no necesariamente de última generación, dada la baja exigencia de los contenidos a presentar. Parece evidente que será más fácil conseguir un buen precio por una buena cantidad de dispositivos a punto de ser retirados del mercado o simplemente de stock sobrante. Debajo de todo esto subyace la idea de utilizar unos estándares mínimos, aunque sean estándares de facto. Esto nos permitiría (como en el caso del Tate Modern) reemplazar los dispositivos por otros nuevos sin hacer cambios significativos en la plataforma y haciendo que coexistiesen ambos modelos. Y lo que es, quizá, más interesante, dar la opción a usuarios que posean dispositivos con capacidades similares a los ofrecidos de conectarse al museo virtual por su cuenta y en directo, ahorrando bastante en costes.

2.2.1.2.2. MoMA (New York, EEUU) Este apartado, desgraciadamente, se refiere a las ideas y modelos observados directamente en su página web. Nos encontramos con ideas, sobre todo en cuanto a presentación visual de contenidos. Digamos que la idea de presentar los contenidos en una matriz de imágenes con pies identificativos que, al pulsar sobre ellas, nos lleven a la información específica de ese objeto podría ser cercana a lo buscado. También sería una idea a tener en cuenta la categorización de las obras por área temática, época o tipo de objeto, aunque no lo firmamos como algo seguro en nuestra plataforma. Por último, reseñar que ellos parece que hacen (o al menos tienen la posibilidad de hacer) uso de la vista “por perspectivas”. Esto sería como tener una lista desplegable en la pantalla que nos permitiera fácilmente cambiar entre los distintos puntos de vista (como si de idiomas se tratara) pudiendo adaptar la plataforma a una presentación para , por ejemplo, niños, estudiantes o expertos en el tema. El sistema finalmente escogido para implementar esa “granularidad” de contenidos se tomó de otro sitio web que veremos más adelante.

2.2.1.2.3. Actual museo del OAN Existe una versión antigua del museo virtual del Observatorio Astronómico Nacional, cuya información será contenida presumiblemente cuando se complete el contenido de la plataforma objeto de este proyecto. Si entramos al sitio web que lo hospeda dentro del sitio del Instituto Geográfico Nacional (Ministerio de fomento) descubrimos una estructura de navegación con algunas deficiencias para movernos entre objetos, para dar un vistazo rápido y desde luego una descontextualización importante. Podríamos catalogarlo como uno de esos museos virtuales del caso cuarto, un catálogo de objetos. No se puede negar su utilidad pero son las funciones de las que carece las que hacen necesaria la creación del proyecto actual.

2.2.1.3. Enciclopedias

Como parte de la gestión del conocimiento, además de las webs de los propios museos, he indagado en otras webs que se dedican a funciones parecidas (no olvidemos que el museo al que se refiere este trabajo dista de ser un museo de arte) a lo que queremos conseguir en cuanto a organización del conocimiento.

2.2.1.3.1. Wikipedia Digamos que este sería el paradigma de enciclopedia dinámica y colaborativa hoy en día. Es más o menos familiar para cualquier persona que use con asiduidad internet (sobre todo para buscar información) y aunque sus fuentes, consistencia y fiabilidad puedan ser cuestionables, su manera de disponer la información no puede tomarse en absoluto como mala. De aquí podríamos tomar en consideración el aspecto colaborativo del medio, pero en reuniones con el cliente se convino que al ser una institución de referencia, los contenidos (aunque aportados por diferentes personas) debían ser acotados a unos pocos usuarios (generalmente trabajadores) y moderados en última instancia por la figura de un editor que revisaría y daría el visto bueno a los mismos. La disposición de portal inicial ofrecida por este sitio es una idea que podría funcionar en nuestro caso, incluyendo esos pequeños “ganchos” que cambian diaria o aleatoriamente para que determinados artículos puedan ser descubiertos por un navegante no asiduo. Además, mantiene una división temática necesaria para el sitio y tiene un sistema de control de versiones válido para documentos escritos, así como la relación semántica de entradas y la inclusión de contenido multimedia fácilmente. Referido al punto anterior, hay que señalar que, en un principio, se ha optado por la manera de Wikipedia de ampliar la profundidad de un artículo: en vez de diseñar distintas capas para distintos individuos, enlazar los artículos que profundizan sobre un tema para que sea el usuario el que decida cuándo parar.

2.2.1.3.2. Indianopedia, Frikipedia, *pedia Esta entrada engloba varios sitios por una razón: existe un fenómeno creciente que consiste en que cada “ecosistema” de conocimiento puede generar su propio repositorio de información. Esto contribuye a diversificar las fuentes evitando fraudes y manipulaciones, a la vez que descentraliza y permite pluralidad de visiones. Igual que existen el diccionario de la RAE, María Moliner, Espasa Calpe, etc., el concepto de la propia enciclopedia, en nuestro caso, puede servir para contener los artículos a los que no llegan otras enciclopedias más generales o de los que tenemos un conocimiento certero y fuentes propias.

2.2.1.4. Sistemas gestores de contenidos vs Wikis

Llegados a este punto parece evidente el camino que estamos sugiriendo. Para crear nuestra plataforma y que tenga algunos factores básicos como diferentes presentaciones al usuario (al menos internet y móviles), incluya una base de datos, interfaz amigable para creadores de contenido no informáticos y demás funciones de administrador facilitadas al máximo.

En un principio se tuvo la tentación de basar toda la plataforma en un mero wiki (La base de datos más simple del mundo) pero se vio que las características más destacables de un wiki nos sobran salvo la facilidad de edición y de uso, que se puede apreciar también en los CMS más comunes. Además, claro está, si nos ciñéramos a un simple wiki estaríamos renunciando a una serie de herramientas útiles que ya se integran en los CMS con naturalidad.

De la toma de decisiones sobre la elección de plataforma base hablaré en la sección de diseño.

2.3. Descripción del diseño

2.3.1. Baremación de alternativas

Llegados a este punto, nuestro proyecto tiene que empezar a definirse y descartar unas alternativas de otras. Hasta la fecha, recapitulando, hemos puesto encima de la mesa la necesidad de un gestor de contenidos con o sin componente wiki. A este esqueleto inicial habría que añadirle una serie de características y su grado de importancia (donde 0 es prescindible y 5 es impres-

cindible). Este grado de importancia ha sido recogido en entrevista directa con responsables de la supervisión del proyecto.

2.3.1.1. Relación de características

Característica	Nota
Interfaz de creador en español	4
Interfaz de usuario en múltiples idiomas	5
Varias perspectivas diferenciadas	1
Web semántica	0
Uso de estándares	0
Editor WYSIWYG	5
Soporte comercial	1
Comunidad de desarrolladores / soporte	4
Actualizaciones recientes	1
Entorno colaborativo	3
Buzón de sugerencia / Preguntas frecuentes	0
Página de control de cambios / CVS	5
Licencia libre	5
Base de datos	3
Sistema de ficheros (en contraposición a BD)	0
Posibilidad de exportar a pdf/LaTeX	1
Versión para plataformas móviles	4
Blog	0

2.3.1.2. Comentarios

La cuestión sobre las perspectivas diferenciadas se ha decidido plantear a la manera de Wikipedia, es decir, a base de enlaces con contenido más exhaustivo en vez de un selector de perspectiva que nos dejase “ponernos las gafas” de “estudiante”, “visitante” o “investigador” con un botón, al menos, en esta fase del proyecto.

Aun no habiendo resuelto temas importantes como si la figura del “responsable de mantenimiento” de la aplicación existirá y quién se encargará de ella, en la misma reunión se esbozó la idea de que existiría una fase inicial de carga con un principio y un final y una fase de mantenimiento en la que se harán cambios pero a menor ritmo. Todas ellas irán recubiertas de la figura del supervisor, que se encargará de aprobar o rechazar los cambios u añadidos propuestos por los generadores de contenido antes de que éstos se reflejen en la producción final.

En contrapartida a las ideas de blog y buzón de sugerencias, se propondrá (con mayor o menor grado de insistencia) al usuario realizar una encuesta evaluación con su propio espacio para sugerencias.

El tema de uso de estándares y web semántica, si bien es un objetivo deseable, no debe obstaculizar el resto de ellos, se asume de la última reunión.

Por último, y para que la base de datos creada pueda ser fácilmente usada por otros proyectos, se ha optado por buscar una solución orientada a bases de datos y no a sistemas de ficheros como modo de almacenamiento. Esto descartará unas cuantas opciones.

2.3.2. Comparativa

Habiendo descartado o relativizado la importancia de algunos de los requerimientos iniciales para nuestro gestor de contenidos, vamos a tratar de hacer una propuesta inicial. Para ello, vamos a utilizar herramientas gratuitas en línea que presentan los datos como una matriz de características (CMSMatrix.org y CMSMatch.com).

2.3.2.1. Premisas iniciales

Las premisas iniciales para los CMS (y las que más descartan) son:

- Más populares y mejor puntuados. De esta manera potencialmente hayaremos estabilidad y una comunidad de soporte activa. Para ello, echaremos un vistazo a las estadísticas disponibles en la red sobre CMS más utilizados, mejor puntuados y más premiados, tratando de sopesar adecuadamente cada fuente.
- Editor visual (WYSIWYG Editor / Drag'n'Drop interface).
- Con capacidad para gestionar la plataforma en varios idiomas (Internationalization = Multilingual Content / Multilingual Content Integration / Web-based Translation Management).
- Página de cambios (Versioning / Asset Management=Media Library)
- Coste cero.
- Licencia OpenSource.
- Comunidad de desarrolladores (Developer Community).
- Interfaz en español (Interface Localization).
- Versión para móviles (Mobile Internet).

2.3.2.2. Candidatos pre-seleccionados

Primeros elegidos por su demostrada utilización, premios, etc:

1. dotCMS
2. Drupal
3. Joomla!
4. LifeRay Portal
5. MODx
6. mojoPortal
7. Moodle
8. Plone

System Requirements	dotCMS 1.9	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	MODx 1.0.2	mojoPortal 2.3.1.3	Moodle 1.9	Plone 3.0	SilverStripe CMS 2.x Open Source 2.3.1	WordPress 2.2.1
Application Server	J2EE	Apache	CGI	J2EE	Apache	IIS/Net	Apache	Zope	Other	Apache
Approximate Cost	Depends on license model	Free	Free	Also available in an Enterprise Edition with professional support.	Free	Free	0	Free	No Cost	Free
Database	Other	MySQL	MySQL	DB2	MySQL	MSSQL	MySQL	Other	MySQL	MySQL
License	Open Source	Open Source	Open Source	Closed Source	Open Source	Open Source	Open Source	Open Source	Open Source	Open Source
Operating System	Platform Independent	Platform Independent	Platform Independent	Platform Independent	Platform Independent	Platform Independent	Platform Independent	Platform Independent	Platform Independent	Platform Independent
Programming Language	Java	PHP	PHP	Java	PHP	C#	PHP	Python	PHP	PHP
Root Access	No	No	No	Yes	Yes	No	No	No	No	No
Shell Access	Yes	No	No	Yes	No	No	No	Yes	No	No
Web Server	Other	Apache	Apache	Other	Apache	Apache	Any	Apache	Apache	Apache

Figura 2.1: System Requiriments desde CMSMatrix.org

9. SilverStripe

10. WordPress

El primer problema que encontramos es que estos sistemas que hemos elegido para comparar los diferentes CMS no nos dejan saber qué idiomas tiene la interfaz de usuario, así que ha que hacer una pequeña investigación entrando en las páginas web de cada uno de los sitios. Tras esto, sabremos que, al menos en cuanto a la interfaz de administración, todas las plataformas preseleccionadas pueden ejecutarse en español.

A continuación, pasamos a desgranar las dos comparativas hechas con las diferentes herramientas encontradas:

2.3.2.3. Comparativa con CMSMatrix

En esta ocasión, vamos a ir comparando las características más reseñables de cada producto, por bloques, según está organizado en el servicio consultado.

2.3.2.3.1. System Requiriments (Requisitos de sistema) Vamos a ir analizando lo que se ve en la figura:

En este caso, debemos fijarnos en la parte “License” principalmente, comprobando que todos menos Liferay Portal ostentan una licencia libre. Posteriormente he podido comprobar que la licencia de Liferay Portal cambió de licencia MIT a LGPL a partir de la versión 6 [1], así que, a falta de argumentación en contra sobre los pormenores de cada licencia, todos entrarían dentro del requisito principal.

Hay que comentar que una de las guías sugeridas por la empresa era que, de ser posible, la plataforma fuese familiar con lo que ya se conoce en la casa por parte de los trabajadores fijos. En este caso, las cosas irían si no ya por la plataforma en sí, por el lenguaje de programación el que están programados, siendo estos Java, PHP, Python y C Sharp los que ocupan nuestra comparativa.

2.3.2.3.2. Security (Seguridad) Según el gráfico de debajo, los datos que más nos interesan son “Content Approval”(aprobar contenidos antes de publicación), “Granular Privileges”(posibilidad de tener distintos privilegios para cada usuario o grupo de usuarios), “Versioning”(control de versiones).

La única opción con inconvenientes en este campo sería Joomla!, que según cmsmatrix no permite la asignación de privilegios personalizada, señalando que MODx en supervisión de contenidos y control de versiones necesitaría un añadido gratuito, al igual que Joomla! para este último caso.

2.3.2.3.3. Support (Asistencia) Esta es una de las partes menos importantes de la comparativa, pues es incierto el futuro en cuestión de mantenimiento de la plataforma. En este caso nos estaríamos fijando más en “Commercial Support”(asistencia de pago), “Developer Community”(comunidad de desarrolladores) y “Public Forum”(foros públicos). Destacar únicamente que WordPress no dispone de ese servicio, aunque el servicio gratuito sí que exista.

2.3.2.3.4. Ease of Use (Facilidad de uso) Este es uno de los puntos más importantes de la comparativa. Si observamos:

Encontraremos que, a destacar, podemos seleccionar como las más relevantes las opciones de “Drag-N-Drop Content”(arrastrar y soltar contenido), “Image Resizing”(redimensionado de imágenes), “Mass Upload”(subida de archivos masiva) y “WYSIWYG Editor”(editor visual).

De la primera característica, tenemos Joomla! que no la soporta, MODx con una implementación limitada y “Drupalcon un añadido para esa función. De la segunda, tendríamos que tanto “mojoPortalcomo “WordPress”presentan esta función limitada y “Drupal”necesita, otra vez, añadidos. En cuanto a la subida de ficheros masiva, mojoPortal sería incapaz mientras que Drupal y WordPress necesitarían un add-on. El editor visual está presente en todas, aunque se nos avisa que Drupal necesita de un añadido también. Lo que nos hace pensar que Drupal está construido de forma modular ya que muchas de sus características se consiguen a través de módulos suplementarios. Esto puede tener consecuencias positivas en cuanto a agilidad, tamaño y complejidad y negativas en cuanto que el proceso inicial de puesta en marcha puede (aún no lo sabemos) ser más complicado que otros.

2.3.2.3.5. Performance (Rendimiento) Este apartado no contiene aspectos demasiado relevantes para el proyecto que tenemos entre manos, pues no se espera que la carga de trabajo del servidor sea elevada como para preocuparse por estos detalles, aunque llegado el momento del desempate, podría declinar la balanza entre una opción y otra. Siendo tanto “WordPresscomo “Joomla!”los más perjudicados.

2.3.2.3.6. Management (Administración) La figura presenta aspectos importantes para el administrador de la plataforma como: “Online Administration”(gestión en línea), “Themes/Skins”(temas gráficos), “Web-based Style/Template Management”(gestión de estilos y plantillas vía web) y “Web-based Translation Management”(gestión de traducciones vía web).

Las características que he citado primera y cuarta son, a mi juicio, fundamentales, ya que permiten la gestión de la plataforma virtualmente desde cualquier dispositivo que pueda mostrar páginas web. Las fallas más significativas son las de “MODxz “mojoPortal”, que gracias a la falta de gestor de traducciones vía web quedarían, por ahora, descalificadas por completo. Hay que

Security	dotCMS 1.9	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	MODx 1.0.2	mojoPortal 2.3.1.3	Moodle 1.9	Plone 3.0	SilverStripe CMS 2.x Open Source 2.3.1	WordPress 2.2.1
audit Trail	Yes	Yes	No	Yes	Yes	No	Yes	Yes	Yes	Limited
captcha	Yes	Free Add On	Free Add On	Yes	Yes	Yes	No	Free Add On	Yes	No
Content Approval	Yes	Yes	Yes	Yes	Free Add On	Yes	Yes	Yes	Yes	Yes
Email Verification	Yes	Yes	Yes	Free Add On	Yes	Yes	Yes	Yes	Yes	Free Add On
Granular Privileges	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Members Authentication	Costs Extra	No	No	Yes	No	Yes	Yes	Free Add On	No	No
SAP Authentication	Yes	Free Add On	Yes	Yes	Free Add On	Yes	Yes	Yes	Free Add On	No
Login History	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Free Add On	No	Free Add On
SS Authentication	Costs Extra	No	No	No	No	Yes	No	Free Add On	No	No
TLM Authentication	Costs Extra	Free Add On	No	Yes	No	Yes	Yes	Free Add On	No	No
Suggestable Authentication	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Problem Notification	Yes	No	No	No	No	Yes	Yes	Free Add On	Yes	Free Add On
Mailbox	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Limited
Session Management	Yes	Yes	Yes	Yes	Yes	Limited	Yes	Free Add On	Limited	Free Add On
MB Authentication	Yes	No	No	No	No	No	No	Free Add On	No	No
SSL Compatible	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SSL Logins	Yes	No	Yes	Yes	Yes	Yes	Yes	Free Add On	Yes	Free Add On
SSL Pages	Yes	No	Yes	Yes	Yes	Yes	Yes	No	Limited	Limited
Versioning	Yes	Yes	Free Add On	Yes	Free Add On	Yes	Yes	Yes	Yes	Free Add On

Figura 2.2: System Requeriments desde CMSMatrix.org

Support	dotCMS 1.9	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	MODx 1.0.2	mojoPortal 2.3.1.3	Moodle 1.9	Plone 3.0	SilverStripe CMS 2.x Open Source 2.3.1	WordPress 2.2.1
Verification Program	Yes	No	No	Yes	No	No	Yes	No	No	No
Code Skeletons	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	No
Commercial Manuals	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	No
Commercial Support	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Commercial Training	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	No
Developer Community	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Online Help	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Draggable API	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Professional Hosting	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Professional Services	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Public Forum	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Public Mailing List	Yes	Yes	No	Yes	No	Yes	Yes	Yes	Yes	Yes
Test Framework	Yes	Free Add On	Yes	Limited	No	No	Yes	Yes	Yes	Yes
Third-Party Developers	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes
Users Conference	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	No	Yes

Figura 2.3: System Requeriments desde CMSMatrix.org

destacar que “WordPress” tiene limitaciones en esto y acumularía ya unas cuantas pegadas que no lo hacen muy aconsejable.

2.3.2.3.7. interoperability (Interoperatividad) Toda las características mostradas en la tabla siguiente tienen una importancia técnica alta, aunque su repercusión en el resultado visual final no sea grande. Quiero, en este apartado, analizarlas una por una para que se sepa de qué trata cada una.

Content Syndication (RSS) (sindicación de contenidos): Se trata de un sistema que permite a los usuarios estar al tanto de las nuevas publicaciones de una plataforma. Como en nuestro caso el contenido será mayoritariamente estático no es algo relevante.

FTP Support (soporte para transferencia de ficheros por FTP): FTP es un protocolo especialmente diseñado para la transferencia de ficheros, siendo, aunque ahora menos utilizado, un valor más. Su importancia es media.

UTF-8 Support (soporte para codificación UTF-8): Esta codificación es la que pretende ser universal para la inmensa mayoría de idiomas. Utilizarla es un paso adelante para la interoperatividad de aplicaciones, pero no es altamente relevante para el proyecto.

WAI Compliant (cumple con WAI): Web Accessibility Initiative (Iniciativa para la Accesibilidad Web) es una serie de reglas que buscan la accesibilidad de los contenidos para usuarios impedidos. Propone unos mínimos y puede ser interesante implementarlo, aunque no vital.

WebDAV Support (soporte para WebDAV): “Edición y versionado distribuidos sobre la web.”^{es}, en esencia, un protocolo que nos permite cambiar a distancia los contenidos de un servidor con cualquier aplicación que lo implemente (sin necesidad de una herramienta propia). No es imprescindible ni es el único modo.

XHTML Compliant (cumple con XHTML): XHTML es una especificación que trata de dar

Case of Use	dotCMS 1.9	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	MODx 1.0.2	mojoPortal 2.3.1.3	Moodle 1.9	Plone 3.0	SilverStripe CMS 2.x Open Source 2.3.1	WordPress 2.2.1
Drag-N-Drop Content	Yes	Free Add On	No	Yes	Limited	Yes	Yes	Yes	Yes	Yes
Email To Discussion	Costs Extra	Free Add On	Free Add On	Yes	No	No	Yes	Free Add On	No	Limited
Friendly URLs	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Image Resizing	Yes	Free Add On	Yes	Yes	Yes	Limited	Yes	Yes	Yes	Limited
Macro Language	Yes	Free Add On	Yes	Yes	Yes	No	No	Yes	No	Free Add On
Mass Upload	Yes	Free Add On	Yes	Yes	Yes	No	Yes	Yes	Yes	Free Add On
Prototyping	Yes	Limited	Yes	Yes	Yes	No	No	Yes	Yes	No
Server Page Language	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Spell Checker	Yes	Free Add On	No	Yes	Limited	Yes	Free Add On	Free Add On	Limited	Free Add On
Style Wizard	Costs Extra	Limited	No	Limited	No	No	Yes	Free Add On	No	No
Subscriptions	Yes	Free Add On	Costs Extra	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Template Language	Yes	Limited	Yes	Yes	Yes	Yes	No	Yes	Yes	No
UI Levels	Yes	No	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes
Undo	Yes	Limited	No	Limited	Limited	No	Yes	Yes	Yes	Free Add On
WYSIWYG Editor	Yes	Free Add On	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Zip Archives	Yes	No	No	Yes	Limited	No	Yes	Free Add On	No	Free Add On

Figura 2.4: System Requeriments desde CMSMatrix.org

Performance	dotCMS 1.9	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	MODx 1.0.2	mojoPortal 2.3.1.3	Moodle 1.9	Plone 3.0	SilverStripe CMS 2.x Open Source 2.3.1	WordPress 2.2.1
Advanced Caching	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Limited	Free Add On
Database Replication	Yes	Limited	No	Yes	No	Yes	Yes	Yes	No	No
Load Balancing	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Limited	No
Page Caching	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Free Add On
Static Content Export	Yes	No	No	Yes	Yes	No	No	Free Add On	Yes	Limited

Figura 2.5: System Requeriments desde CMSMatrix.org

Management	dotCMS 1.9	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	MODx 1.0.2	mojoPortal 2.3.1.3	Moodle 1.9	Plone 3.0	SilverStripe CMS 2.x Open Source 2.3.1	WordPress 2.2.1
Advertising Management	Yes	Free Add On	Yes	Yes	Free Add On	Limited	No	Free Add On	Limited	No
Asset Management	Yes	Yes	Yes	Yes	Yes	Limited	Yes	Yes	Yes	Limited
Dashboard	Yes	No	No	No	Yes	No	Yes	Yes	Yes	No
Content Scheduling	Yes	Free Add On	Yes	Yes	Yes	Yes	No	Yes	Yes	Free Add On
Content Staging	Yes	Free Add On	No	Yes	Yes	No	Yes	Free Add On	Yes	No
Online Administration	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Limited	No
Online Administration	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Limited	Yes
Package Deployment	Yes	No	No	Yes	Free Add On	Yes	Yes	Yes	No	No
Sub-sites / Roots	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Themes / Skins	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Trash	Yes	No	Yes	No	Yes	No	No	Free Add On	No	No
Web Statistics	Yes	Yes	Yes	Limited	Yes	Limited	Yes	Free Add On	Limited	Free Add On
Web-based Style/Template Management	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	No	Yes
Web-based Translation Management	Yes	Yes	Free Add On	Yes	No	No	Yes	Yes	Yes	Limited
Workflow Engine	Yes	Limited	No	Limited	No	No	Yes	Yes	Yes	No

Figura 2.6: System Requeriments desde CMSMatrix.org

Performance	dotCMS 1.9	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	MODx 1.0.2	mojoPortal 2.3.1.3	Moodle 1.9	Plone 3.0	SilverStripe CMS 2.x Open Source 2.3.1	WordPress 2.2.1
Advanced Caching	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Limited	Free Add On
Database Replication	Yes	Limited	No	Yes	No	Yes	Yes	Yes	No	No
Load Balancing	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Limited	No
Page Caching	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Free Add On
Static Content Export	Yes	No	No	Yes	Yes	No	No	Free Add On	Yes	Limited

Figura 2.7: System Requeriments desde CMSMatrix.org

coherencia y acercar el lenguaje HTML a los estándares XML. Busca la ausencia de errores (se puede validar con facilidad) y la fiel reproducción de un contenido en distintos navegadores sin demasiadas diferencias. Es decir, busca establecer un estándar. Yo lo consideraría un mínimo exigible para cualquier plataforma web aunque en el futuro se de la posibilidad de cambiar a otros estándares como HTML5 (que aún no están implantados masivamente).

Dicho todo esto, considero que Joomla! también debería ser apartado de los candidatos dada la falla en cuanto a XHTML. Los costes extra de dotCMS para algunas características me hacen pensar que la filosofía de ese proyecto no es tan cercana a nuestros objetivos como podría parecer, así que también los desaconsejaría.

2.3.2.4. Propuesta preliminar

Los candidatos serán cuatro. Todos ellos son multiplataforma, corren en cualquier servidor y se componen de código abierto. Todos, además, disponen de interfaz en idioma español e inglés (como mínimo), galería de fotos y sistema de control de versiones (más o menos completo). Todos funcionan alimentándose de una base de datos con API estándar.

Como particularidades reseñables, podemos decir que, en cuanto a servidor web, Plone y Drupal están muy bien integrados con Apache, mientras que Liferay Portal se integra con otros servidores más orientados a Java (Tomcat, Glassfish, etc.) y Moodle que se integra con cualquier plataforma que tenga soporte para PHP.

2.3.2.4.1. Liferay Portal (Community Edition) Esta herramienta no presenta inconvenientes importantes ahora que su código es libre [1], pues antes no era así. Más a su favor, tiene una empresa como IBM detrás lo que supone que su código viene siendo mantenido y desarrollado por profesionales. Es una solución seria, empresarial y con una curva de aprendizaje media. Se usa en sitios como los de CISCO, Honda, T-Mobile, Harvard Bussiness School, China Mobile, etc. La rumorología parece indicar que sería la solución que el Instituto Geográfico Nacional estaría adoptando para su gestión de contenidos hipertextuales, con lo que, de ser cierto, ganaría puntos como solución definitiva, dado que habría personal cualificado para mantener la aplicación si fuese necesario. Además de todo lo anterior, posee soporte comercial consolidado y está bastante probado.

Todo lo anterior la convierte en la opción más sólida a mi modo de ver.

2.3.2.4.2. Drupal Se trata de una de las plataformas más populares y amigables para la gestión de contenidos. Es usada para todo tipo de proyectos desde poco importantes hasta de envergadura media-alta. Sin embargo, no parece (por lo leído en algunos artículos) que sea la opción más potente. No tiene, tampoco, carencias que señalar, destacando, eso sí, su carácter modular, necesitando así para algunas de las funcionalidades deseables de la instalación de módulos adicionales (gratuitos). Tiene una gran base de usuarios y está mantenido por la comunidad principalmente. Destaca, además, por su buena integración con plataformas móviles. Es rápido, intuitivo y versátil.

2.3.2.4.3. Moodle Este CMS está orientado más al e-learning aunque es cierto que cumple con prácticamente todas las funciones deseables. Se trata de un software maduro y completo. Está programado en PHP, al igual que el anterior.

2.3.2.4.4. Plone Probablemente este CMS sea el más innovador de los presentados. Muy potente, está programado en Python y construido encima del framework Zope, independientemente del servidor que vaya a correr por debajo (que en general será Apache). Al igual que el resto, no presenta deficiencias importantes, si bien, como Drupal, está pensado de manera modular, con añadidos para según qué funciones. Si hay que poner una pega (y no es trivial), es su curva de aprendizaje, dado que llena los huecos dejados por Drupal en cuanto a potencia pero también en cuanto a dificultad de uso haciéndolo una opción no recomendable para no-informáticos, quizá.

2.3.3. Anexo a comparativa

Como resultado de la comunicación con el cliente, se hace una rápida referencia a una plataforma como candidata a ser la destinataria de la migración del CMS de todo el Instituto Geográfico Nacional. Normalmente, todo lo relacionado con la gestión de contenidos y, en general, todo el software generado suele ser subcontratado a otras empresas, dejando los desarrollos propios o artesanales casi exclusiva y paradójicamente al personal especializado (astrónomos, físicos, etc.).

La propuesta es analizar brevemente el candidato ".Alfresco" para ver si sería viable tenerlo en cuenta como candidato para nuestro desarrollo.

2.3.3.1. Breve análisis de Alfresco

Para proceder con este análisis, la principal fuente de información será el sitio web de la compañía dedicado a hablar de la entidad. Alfresco es una empresa fundada por John Newton (Documentum¹) y John Powell (Bussiness Objects²) e integrada por ingenieros de Documentum y Oracle en 2005 con capital de diferentes empresas de inversión como SAP Ventures, Accel Partners y Mayfield Fund. Se autodenominan la ".alternativa de código abierto para la gestión de contenidos empresariales (ECM) [4].

Aquí aparece un concepto desconocido hasta ahora en este documento: gestión de contenidos empresariales. Según la Asociación para la gestión de imágenes e información (AIIM en inglés), ECM se define como [5]: "las estrategias, métodos y herramientas usadas para capturar, gestionar, almacenar, preservar y entregar contenido y documentos relacionados con procesos

¹Empresa americana dedicada a la creación de un gestor de contenidos empresariales con el mismo nombre, hoy mantenido por EMC Corporation. [2]

²Empresa francesa dedicada a Bussiness Intelligence y desde 2007 parte de SAP AG [3]

organizativos. Las estrategias y herramientas ECM permiten la administración de la información no estructurada de una organización donde quiera que ésta exista.”*

Aunque quizá ya era de esperar, podemos ver que este tipo de plataforma abarca mucho más de lo que en principio necesitaríamos y está enfocado de manera diferente a lo que sería nuestro museo virtual. Si nos atenemos a la plataforma, rápidamente vemos que forma parte del modelo de negocio que algunas empresas han encontrado para explotar el código abierto: el soporte. No hay otra manera (salvo una evaluación de 30 días) de utilizar la plataforma Enterprise que la de contratar una suscripción con la empresa atendiendo a diversos paquetes de soporte integrado. Las características incluidas en todas las opciones son:

- Pruebas certificadas de escalabilidad
- Un modelo controlado del lanzamiento
- Mantenimiento y Actualizaciones
- Resolución de Problemas
- Consejos de Configuración, Compatibilidad y Migraciones
- Soporte de Actualización
- Consejos de Puesta a punto
- Tiempos de reacción garantizados

Como todo sitio corporativo, su lenguaje es marcadamente general por lo que conviene entrar un poco más en detalle. Para ello podemos utilizar el mismo recurso antes mencionado de CMSMatrix.org [?]. En este caso, no tenemos muy clara cuál es la plataforma analizada, ya que Alfresco se presenta en distintos sabores [?]: Cloud Trial (prueba online, no incluye gestión de contenidos web, Alfresco Enterprise (probada y certificada para plataformas tanto código abierto como propietarias, con soporte comercial completo) y Alfresco Community (para plataformas de código abierto, soportada por la comunidad).

Debido a la variabilidad de plazos y la diferencia apreciable de alcance en los proyectos del IGN y de este proyecto respectivamente, al final se decidió abandonar la línea de Alfresco sobre Liferay Portal y se optó por Drupal, que reunía los requisitos adecuados y tiene una gran comunidad de usuarios y desarrolladores detrás.

Sin embargo, he creído conveniente adjuntar un anexo donde se indague un poco más sobre Alfresco y sus posibilidades.

2.3.4. Modelo de datos

Hemos dejado bien claro que el propósito de este trabajo no es desarrollar de cero un CMS si no escoger y adaptar uno ya existente a las necesidades que se nos han presentado. Antes de tomar una decisión (que es independiente del autor y tomará tiempo), es conveniente señalar que sí que debería existir una parte de “diseño original” que correspondería al modelado de algunas informaciones.

*Traducción propia.

2.3.4.1. Catálogo

Respondiendo a los requisitos enunciados por miembros de la empresa, he elaborado un diseño de modelo de datos que podría utilizarse para abstraer los contenidos de un catálogo de objetos expuestos (subconjunto de ítems informativos del total del museo virtual) y no expuestos. Para ello, me he ayudado de un catálogo editado en papel por el propio Ministerio de Fomento titulado “Instrumentos históricos del Instituto Geográfico Nacional”. Debido a su creación artesanal cada instrumento no sigue la debida canonización que nos piden las bases de datos, y es por eso que he tratado de ordenarlo en un modelo de datos como el que se muestra:

2.3.4.1.1. OBJETO

- id/ref: número entero
- nombre: caracteres(30)
- firmado: (inventor, lugar_origen, fecha_origen) (se eliminaría del original)
- inventor: relación(n:n)
- lugar_origen: relación(n:n)
- fecha_origen: fecha
- dimensiones: caracteres(12)
- características técnicas: texto
- materiales: caracteres(20)
- depósito_habitual: relación(1:n)
- observaciones: texto
- área: relación(1:1)
- tipo de objeto: relación(1:n)
- foto: caracteres(50)
- depósito actual: caracteres(50)

2.3.4.1.2. ÁREA

- id: numerico
- nom_area: caracteres(20) Astronomía esférica, Geodesia geométrica, Topografía planisférica y Fotogrametría terrestre. Topografía altimétrica, Gabinete, Meteorología, Geofísica

2.3.4.1.3. TIPO_OBJETO

- id: numerico
- nom_tipo: caracteres(40)
- id_area: relacion(n:1)

2.3.4.1.4. ARTÍCULOS

- id: numerico
- titulo: caracteres(25)
- contenido: texto

2.3.4.1.5. AREA_ARTÍCULO

- id_area: relación(1:n)
- id_articulo: relacion(n:1)

2.3.4.1.6. INVENTOR

- id_inventor: numerico
- nom_inventor: caracteres(25)

Este modelo será reutilizado (si se aprueba) para crear una interfaz sencilla de entrada de datos para el personal interno.

2.3.5. Drupal

Como dije más arriba, Drupal será inicialmente la plataforma seleccionada para comenzar con la integración y desarrollo del sistema. Se trata de un sistema complejo, del que prefiero describir algunos fundamentos básicos que ayuden a entenderlo todo.

Existen multitud de recursos en Internet sobre Drupal, empezando por su sitio web principal [6], pero el recurso más útil que he encontrado es el libro editado por O'Reilly [7] y creado por el staff del sitio web Lullabot [8]. Aunque la versión traducida al español editada por Anaya [9] facilita el seguimiento general del libro, considero que en algunos aspectos es una traducción, cuanto menos, apresurada.

Para la prueba inicial he utilizado un "software appliance", es decir, una imagen de disco dispuesta a ser arrancada por una máquina virtual (en este caso VirtualBox) y empezar a funcionar. esto facilita mucho las cosas a la hora de realizar configuraciones rápidas y dinámicas dejando para más adelante (cuando tengamos una arquitectura definida) la integración y configuración de un servidor real.

La máquina elegida fue sacada de la web de TurnKey [10] en la que se pueden encontrar multitud de imágenes listas para hacer la función de servidor web con CMS nada más ser arrancadas. Suele tratarse de un montaje LAMP (Linux + Apache + MySQL + PHP) [11] En concreto se utilizó la que integraba la versión 6 de Drupal [12] con un entorno que se describe con los siguientes componentes básicos:

- TurnKey Core [13], compuesto de:
 - Ubuntu 8.04.3 LTS
 - Autoactualizable en el arranque

- Shell en AJAX incorporada (shellinabox [14])
 - Interfaz de administración web (Webmin [15])
 - Apache, MySQL y PHP.
- Soporte para SSH por defecto.
 - MTA integrado para el envío de correo local.
 - Módulos de Webmin para PHP, Apache2 y MySQL.
 - PHPMyAdmin como administrador de la base de datos.
 - Módulos más conocidos de Drupal (CCK, Views, Lightbox2, ImageCache, CKEditor, etc.).

2.4. Conclusiones y trabajo futuro

Capítulo 3

Planos

3.1. Esquemáticos

Aqu tenemos los esquemáticos.

Capítulo 4

Pliego de condiciones

4.1. Requisitos de Hardware

Estos son los requisitos mnimos de Hardware

4.2. Requisitos de Software

Estos son los requisitos mnimos de Software

Capítulo 5

Presupuesto

5.1. Presupuesto

En esta seccin se detalla el presupuesto de ejecucin del proyecto.

Capítulo 6

Manual de Usuario

6.1. Manual

En esta sección se resumirá el manual de usuario de la aplicación en cada caso.

Capítulo 7

Soporte informático

Capítulo 8

Bibliografía

Bibliografía

- [1] Liferay adopting the lgpl license - blog - liferay.com. [Online]. Available: <http://www.liferay.com/web/bryan.cheung/blog/-/blogs/4656695/maximized>
- [2] Documentum - wikipedia, the free encyclopedia. [Online]. Available: <http://en.wikipedia.org/wiki/Documentum>
- [3] Bussiness objects (company) - wikipedia, the free encyclopedia. [Online]. Available: [http://en.wikipedia.org/wiki/Business_Objects_\(company\)](http://en.wikipedia.org/wiki/Business_Objects_(company))
- [4] Sobre alfresco - la alternativa para la gestión de contenidos empresariales en código libre. [Online]. Available: <http://www.alfresco.com/es/about/>
- [5] Aiim - what is ecm? what is enterprise content management. [Online]. Available: <http://www.aiim.org/What-is-ECM-Enterprise-Content-Management.aspx>
- [6] Drupal - open source cms | drupal.org. [Online]. Available: <http://drupal.org/>
- [7] A. Byron, A. Berry, N. Haug, J. Eaton, J. Walker, and J. Robbins, *Using Drupal*, 1st ed. O'Reilly Media, 12 2008.
- [8] Lullabot | drupal videos, training, consulting and guidance. [Online]. Available: <http://www.lullabot.com/>
- [9] A. Byron, A. Berry, N. Haug, J. Eaton, J. Walker, and J. Robbins, *Drupal*, 1st ed. Anaya Multimedia / O'Reilly, 09 2009.
- [10] Turnkey linux virtual appliance library | virtual appliances for virtual machines, cloud computing and bare metal. [Online]. Available: <http://www.turnkeylinux.org/drupal6>
- [11] Lamp (software bundle) - wikipedia, the free encyclopedia. [Online]. Available: [http://en.wikipedia.org/wiki/LAMP_\(software_bundle\)](http://en.wikipedia.org/wiki/LAMP_(software_bundle))
- [12] Drupal 6 appliance - turnkey linux virtual appliance library. [Online]. Available: <http://www.turnkeylinux.org/drupal6>
- [13] Turnkey core - turnkey linux virtual appliance library. [Online]. Available: <http://www.turnkeylinux.org/core>
- [14] shellinabox - project hosting on google code. [Online]. Available: <http://code.google.com/p/shellinabox/>
- [15] Webmin. [Online]. Available: <http://www.webmin.com/>

