



ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN

Titulación :

INGENIERO SUPERIOR EN INFORMÁTICA

Título del proyecto:

“DESARROLLO DE UN SITIO WEB PARA CLUBS DE
LECTURA”

Hilton Trelles Labairu
Alfredo Pina Calafi
Pamplona, a 1 de Julio de 2016

Índice

1	Proyecto Club de lectura de la UPNA	7
2	Estado de la cuestión	8
2.1	Introducción	8
2.2	Evolución de los Sistemas Gestores de Contenido	8
2.3	Sistemas gestores de contenido.....	10
2.3.1	Definición	10
2.3.2	Creación de contenido	11
2.3.3	Gestión de contenido	11
2.3.4	Publicación	11
2.3.5	Presentación.....	12
2.4	Necesidad de los gestores de contenido.....	12
2.4.1	Criterios de selección de gestores de contenidos.....	13
2.4.2	Ejemplo de gestores de contenido	15
3	Selección del gestor de contenidos.....	17
3.1	Wordpress	18
3.2	Drupal	18
3.3	Joomla.....	19
3.4	Evaluación	20
3.4.1	WordPress.....	20
3.4.2	Drupal	21
3.4.3	Joomla.....	21
3.4.4	Elección	22
4	Especificaciones del alojamiento necesario para la web.	23
4.1	Descripción de la necesidad.....	23
4.2	Posibles soluciones de infraestructura	24
4.2.1	Hospedaje gratuito	24
4.2.2	Alojamiento compartido	25
4.2.3	Servidor Virtual Privado (VPS).....	26
4.2.4	Servidor Dedicado	27
4.2.5	Cloud Hosting.....	27
4.2.6	Hospedaje especializado/gestionado	28
4.3	Infraestructura seleccionada para el proyecto.....	29

5	Fase de desarrollo	30
5.1	Herramientas utilizadas.....	30
5.2	Estructura de un componente Joomla	31
5.2.1	Patrón Modelo – Vista – Controlador.....	31
5.2.2	Estructura de ficheros básica de un componente.....	33
5.3	Diseño del componente com_lectura	34
5.3.1	Tecnologías utilizadas	34
5.3.2	Estructura de ficheros	39
5.3.3	Diagrama de componentes.....	42
5.3.4	Modelo entidad-relación.....	46
5.3.5	Diseño de la interfaz	47
6	Fase de Implantación	52
6.1	Arquitectura de Joomla.....	52
6.1.1	Organización de contenidos: categoría y artículo.....	53
6.1.2	Presentación de contenidos: menús	53
6.1.3	Apariencia del portal web: plantillas	53
6.1.4	Extensiones: componentes, módulos y plugins	54
6.2	Proceso de Implantación	55
6.2.1	Instalación de Joomla	55
6.2.2	Configuración del portal.....	60
6.2.3	Instalación y configuración de la plantilla.....	62
6.2.4	Instalación de extensiones: módulos, componentes y plugins	64
6.2.5	Categorías y artículos	67
6.2.6	Gestión de menús	68
6.2.7	Gestión de imágenes en la galería.	70
7	Testing	74
7.1	Estrategia de pruebas	74
7.1.1	Alcance de las pruebas	74
7.1.2	Enfoque de las pruebas	75
8	Aspectos SEO	90
8.1	Introducción al SEO.....	90
8.2	Acciones de mejora de SEO	92
8.2.1	Introducir Meta-datos	92

1 de julio de 2016

8.2.2	URL's amigables	94
8.2.3	Google Webmasters Tools	95
8.2.4	Mapa del sitio con OSMap	96
8.2.5	Informes SEO con Google Analytics.	98
8.2.6	Eliminar las URL's duplicadas.....	100
8.2.7	Contenido - Blog o sección de Noticias	100
8.2.8	Etiquetado HTML	100
8.2.9	Peso de las imágenes de tus artículos	101
8.2.10	HTML5 y nuevas plantillas	101
9	Procesos de mantenimiento del sistema	103
9.1	Copia Automática	103
9.2	Copia Manual.....	104
9.3	Restaurar copias de seguridad.....	104
9.4	Actualización del gestor de contenido	104
10	Conclusiones.....	106
11	Glosario de términos.....	107
12	Bibliografía.....	107
13	Webs de consulta.....	108

Ilustraciones

Ilustración 1.	Estadísticas de uso de CMS - 06/2016	17
Ilustración 2 .	Esquema de funcionamiento del MVC.....	32
Ilustración 3.	Instalación estilos. Definición CSS	37
Ilustración 4.	Instalación estilos. Gestión de estilos. Prioridad CSS.....	38
Ilustración 5.	Modelo de comunicación entre capas.....	43
Ilustración 6.	Diagrama de comunicación Joomla Framework	45
Ilustración 7 .	Diagrama de comunicación de un componente Joomla	45
Ilustración 8.	Modelo entidad relación com_lectura	46
Ilustración 9 .	Distribución áreas de contenido. PC y Tableta horizontal.	48
Ilustración 10	Distribución áreas de contenido. Movil y tableta vertical	48
Ilustración 11.	Interfaz componente com_lectura	49
Ilustración 12 .	Detalle del libro seleccionado	50
Ilustración 13.	Elementos que componen la arquitectura de Joomla!	52
Ilustración 14.	Plantillas. Especificación de la estructura de la plantilla	54
Ilustración 15.	Cliente FTP y carpeta raíz del servidor web	56
Ilustración 16 .	Instalación de Joomla. Panel de control del espacio web.....	56
Ilustración 17.	Instalación de Joomla. Bases de datos MySQL.....	57

1 de julio de 2016

Ilustración 18. Instalación de Joomla. Lista de bases de datos activas	57
Ilustración 19. Instalación de Joomla. Configuración principal	58
Ilustración 20. Instalación de Joomla. Configuración de la base de datos	59
Ilustración 21 .Instalación de Joomla. Finalización de la instalación	60
Ilustración 22. Instalación de Joomla. Confirmación de la instalación	60
Ilustración 23. Configuración de Joomla. Panel de control.....	61
Ilustración 24 . Configuración Joomla. Configuración del correo	61
Ilustración 25 .Instalación estilos. Gestión de estilos. Estilos por defecto	62
Ilustración 26. Instalación estilos. Gestión de estilos. Configuración de estilos.....	62
Ilustración 27. Instalación estilos. Gestión de estilos. Personalización de estilos.....	63
Ilustración 28 .Instalación estilos. Gestión de estilos. Validación W3C.....	63
Ilustración 29 . Galería. Gestión multimedia.....	72
Ilustración 30 Galería. Patrón Lazy Load o carga diferida.....	73
Ilustración 31 . Estartegia de pruebas. Modelo en V.....	74
Ilustración 32 .Ciclo del desarrollo de pruebas con TDD	78
Ilustración 33 . Resultados de ejecución de pruebas en PHPUnit.....	80
Ilustración 34 . QUnit. Resultados de pruebas de componente automaticas	85
Ilustración 35 . Creación de la maquina virtual. SO	87
Ilustración 36 . Creación de laa maquina virtual. Añadir disco virtual	88
Ilustración 37 . SEO. Distribución de motores de búsqueda por paises.....	91
Ilustración 38 . SEO. Resultado de búsqueda en Google. Meta-descripción.	92
Ilustración 39. SEO. Configuración Global de Joomla 3	93
Ilustración 40 . SEO. Configuración de los metadatos específicos para una página.....	93
Ilustración 41 . SEO. Reescritura URL's	94
Ilustración 42. SEO. Configuración del "Alias"	94
Ilustración 43. SEO. Google Webmasters	95
Ilustración 44. SEO. Contenido Sitemap.....	97
Ilustración 45. SEO. “Añadir o probar sitemap” en Google Webmasters Tools.....	98
Ilustración 46. SEO. Añadir o probar un sitemap en Google Webmasters Tools	98
Ilustración 47. SEO. Google analytics	99
Ilustración 48. SEO. Google Webmaster Tools. Dominio preferido.	100
Ilustración 49. Configuración de Copia de seguridad automática.....	103
Ilustración 50. Configuración de Copia de seguridad automática.....	104
Ilustración 51. Copia de seguridad manual. Akeeba Backup.....	104
Ilustración 52. Actualizaciones en Joomla	105

Tablas

Tabla 1. Ejemplo de gestores de contenido	16
Tabla 2. Requerimientos de Joomla 3	24
Tabla 3. Herramientas utilizadas en el proyecto.....	30
Tabla 4. Estructura de ficheros básica de un componente Joomla	34
Tabla 5 . Ficheros del sitio	40
Tabla 6. Ficheros de la administración.....	41

1 de julio de 2016

Tabla 7. Tabla CLUBPN_LIBROS	46
Tabla 8. Tabla CLUBPN_PAISES	46
Tabla 9 . Tabla CLUBPN_PAISES_LIBROS.....	47
Tabla 10 . Modulos cargados en el proyecto	66
Tabla 11. Plugins activos.....	67
Tabla 12 . Categorías y artículos en el proyecto.....	68
Tabla 13 Menús del proyecto	69
Tabla 14 . Enfoque de pruebas. Tipos de pruebas	76

1 Proyecto Club de lectura

El proyecto contenido en el siguiente documento busca proporcionar al Club de lectura de un canal de comunicación con los alumnos que ayude a potenciar la lectura individual de los libros propuestos, con el fin de confrontar, dialogar e intercambiar experiencias y opiniones que su lectura suscita.

Los objetivos del Club de lectura de son los siguientes:

- Descubrir el valor de la lectura en sí misma, como base para la ampliación cultural y, al mismo tiempo, como entretenimiento.
- Servir de apoyo para leer obras y autores variados, representativos de diversas tendencias artísticas e intelectuales, clásicas y contemporáneas.
- Comprender la obra literaria en el contexto de su creación; es decir, saber qué opinan de ella el autor, el crítico, etc.
- Aprender a tolerar ideas, en muchos casos contrarias a las nuestras.
- El intercambio de ideas y opiniones diversas sobre la obra entre las personas que integran el club como factor para el enriquecimiento mutuo.

Los requerimientos para la web a desarrollar en este proyecto son:

- Ha de permitir añadir contenido por parte de los componentes del Club de lectura de forma fácil.
- Ha de disponer de un foro donde poder llevar a cabo discusiones en torno al club y los libros propuestos.
- Ha de poder permitir consultar los libros ya leídos, consultando por el país donde se lleva a cabo la trama. Se solicita que el filtrado de los países se haga mediante un mapamundi interactivo.
- Ha de poder consultarse tanto desde un ordenador como desde terminales móviles.
- Ha de disponer de una galería de imágenes.

Tal y como se detalla en el presente documento, la estrategia de implementación del proyecto es usar un Sistema Gestor de Contenido, ya que proporciona prácticamente la totalidad de la funcionalidad requerida de entrada, y desarrollar a medida únicamente la gestión de los libros ya que la funcionalidad no es estándar.

El uso del Sistema Gestor de Contenido permitirá ahorrar tiempo de pruebas del sistema ya que hay que desarrollar menos código y reducirá el mantenimiento, ya que las actualizaciones las proporciona el creador del sistema gestor.

2 Estado de la cuestión

2.1 Introducción

Para cubrir los objetivos del proyecto se hará uso de un Sistema Gestor de Contenido ya que proporciona muchos de los requerimientos, junto a las extensiones de las que disponen, sin necesidad de desarrollar código.

Gracias al uso del Sistema Gestor de Contenido se reduce el tiempo de implementación del proyecto y se aumenta la calidad del mismo, ya que los Sistemas Gestores de Contenido más populares actualmente tienen un alto grado de madurez.

A lo largo de este capítulo se explicará la evolución de estos sistemas, que características tienen y cuáles son los Sistemas Gestores de Contenido de referencia actualmente.

2.2 Evolución de los Sistemas Gestores de Contenido

Los Sistemas Gestores de Contenido han evolucionado mucho desde sus inicios, ganando en funcionalidad, robustez, extensibilidad y capacidad de integración con otros sistemas.

Los Sistemas Gestores de Contenido se crearon inicialmente para su uso en las intranets de bibliotecas, diarios u otros sitios enfocados a la información. Actualmente el término de Sistema Gestor de Contenido se usa exclusivamente para referirse a sistemas web.

1992-95

En los primeros tiempos de la era web, los Sistemas Gestores de Contenido eran utilizados únicamente por los sites de gran presupuesto como Amazon.com. La mayoría estaban programados en C++ y funcionaban sobre bases de datos propietarias como por ejemplo Oracle. Los costes de puesta en marcha eran millonarios.

1995

En 1995 aparece el primer Sistema Gestor de Contenido de código libre. Fue creado para la Portland Pattern Repository. El desarrollador puso como nombre al sistema "Wiki Wiki," expresión Hawaiana para "rápido y fácil". El concepto del wiki se le ocurrió a su creador al final de los años 1980, y lo implantó inicialmente como una pila de Apple Hypercard. El sitio de noticias tecnológicas CNET sacó su sistema de administración de documentos y publicación, creando la compañía llamada Vignette, que se convirtió a la postre en el pionero de los sistemas de administración de contenidos comerciales. Luego ha venido una sostenida cadena de nuevos productos, actualizaciones y adelantos extraordinarios.

1 de julio de 2016

1997

PHP, el motor HTML creado en 1995, fue reestructurado y lanzado con un éxito inmediato por su versatilidad, estabilidad y facilidad de uso. El acrónimo “PHP” originalmente significaba “Personal Home Page”, pero fue rebautizado a “Procesador de hipertexto PHP”.

1995-1999

Aparecen miles de sitios Wiki basados en Perl alrededor de Internet, cada uno dedicado a intereses específicos y con multitud de colaboradores. En esta época aparecen también una serie de Sistemas Gestores de Contenido basados en PHP, como PhpNuke y PostNuke, y una combinación de los dos llamado PhpWiki. Sin embargo, su difusión se limitaba a la comunidad tecnológica. La mayoría de los Sistemas Gestores de Contenido de este período se financiaron por compañías y utilizaban software propietario para su creación como ColdFusion de Allaire, "Active Server Pages", de Microsoft ASP y bases de datos Oracle. Durante estos años, otros tres importantes elementos de código abierto maduraron: el programa de base de datos MySQL, el sistema operativo Linux para servidores y el servidor web Apache. Todas ellas se hicieron más útiles y estables lo que propició implantación generalizada por parte de los proveedores de alojamiento, permitiendo que el servicio llegara a los usuarios finales a bajo coste.

1999

IntraNet Solutions lanza Xpedio, el primer Sistema Gestor de Contenido de extremo a extremo.

2000

Aparece PHP 4.0, que incorporaba una gran cantidad de mejoras respecto a versiones anteriores. La combinación de un sistema operativo libre y estable (Linux), un servidor web libre (Apache), una base de datos libre (MySQL), y un generador libre de código HTML fácil de aprender (PHP), permitido a los desarrolladores web construir y gestionar grandes y complejos sitios a bajo coste sin necesidad de años de formación técnica. Las cuatro herramientas conjuntamente forman lo que se conocería como LAMP (no todos los Sistemas Gestores de Contenido de código libre utilizan LAMP, hay diversas variantes p.e, utilizar Unix en vez de Linux o codificando en Perl en vez de PHP – pero la combinación LAMP es la más común).

2001

Aparece Drupal, inicialmente orientado como un gestor de noticias.

2003

1 de julio de 2016

La foto de internet a cambiado radicalmente respecto a los 90: cualquiera puede realizar de forma gratuita y por sí mismo algo equivalente al sistema por el cual Amazon.com invirtió millones de dólares y miles de personas tan solo una década antes. En los últimos tres años el número de sitios que utilizaban LAMP creció de unas pocas docenas de miles a millones. Los entusiastas de LAMP afirmaban que las ventajas con respecto a los sistemas propietarios como Microsoft y Oracle no era sólo el precio, sino también la facilidad de uso, la estabilidad, la seguridad y la ayuda de las comunidades de usuarios. Los sistemas propietarios a menudo no parecían valer lo que costaban, podían ser más difíciles de usar y ser menos seguros.

2004

Aparece en enero WordPress.

2005

Aparece Joomla como resultado de una bifurcación de Mambo Open Source.

En la actualidad

Actualmente hay en internet unos 24,5 millones (Fuente: trends.builtwith.com/cms; datos del 13 de Junio de 2016) de webs que se basan en algún Sistema Gestor de Contenido. Algunos Sistemas Gestores de Contenido como por ejemplo Wordpress, con 12,2 millones de webs, tienen una gran aceptación y entorno a estos Sistemas Gestores de Contenido que tienen tanta difusión se han creado comunidades que colaborar desarrollando extensiones para los mismos o dando soporte a las dudas que puedan aparecer en su utilización.

2.3 Sistemas gestores de contenido

2.3.1 Definición

Los sistemas de gestión de contenidos son un programa que tiene el fin de facilitar la gestión de webs, ya sea en Internet o en una intranet. Por este motivo también son conocidos como gestores de contenido web (Web Content Management o WCM). Actualmente estos sistemas proveen multitud de funcionales, en su instalación base y mediante extensiones, pero como mínimo proveen las siguientes funcionalidades:

- Creación de contenido
- Gestión de contenido
- Publicación
- Presentación.

1 de julio de 2016

2.3.2 Creación de contenido

Los Sistemas Gestores de Contenido facilitan la creación de contenidos habilitando su uso para usuarios sin experiencia técnica, habitualmente disponen de un editor de texto WYSIWYG, en el que el usuario ve el resultado final mientras escribe, para la creación y edición de contenidos, al igual que haría en un procesador de textos. El editor suele estar limitado a las funciones básicas a la hora de aplicar estilos. Adicionalmente, los Sistemas Gestores de Contenido permiten mayor nivel de personalización a usuarios con conocimientos técnicos, pudiendo incluir código HTML con hojas de estilo.

Para la creación de la estructura del sitio web los Sistemas Gestores de Contenido disponen de herramientas para definir la estructura y formato del contenido, modificar el aspecto visual o añadir funcionalidad mediante el uso de extensiones.

Los contenidos que se pueden generar no se limitan a texto, ya que actualmente todos los Sistemas Gestores de Contenido permiten añadir imágenes o videos para la generación de contenidos multimedia.

2.3.3 Gestión de contenido

La estructura de la web se puede crear/modificar mediante las herramientas proporcionadas por el Sistema Gestor de Contenido, estas herramientas normalmente presentan el contenido de forma jerárquica.

Los gestores de contenido aplican a esta estructura unos roles, con ciertos permisos asociados, que permiten realizar la gestión del ciclo de vida del contenido (responsables, editores, autores y usuarios). La gestión de los usuarios se puede realizar mediante el propio Sistema Gestor de Contenido.

El contenido creado en el gestor de contenido(texto, imágenes (ruta), versión, autor, fecha de publicación, etc.), así como la información de configuración del Sistema Gestor de Contenido (datos de los usuarios, la estructura de la web, las extensiones instaladas, etc.), se guardan en una base de datos.

2.3.4 Publicación

Una página aprobada se publica de forma automática al llegar la fecha de publicación, y se archiva al llegar la de caducidad, quedando archivada.

Se puede gestionar la publicación también de forma manual, permitiendo al editor generar los contenidos y esperar al momento adecuado para publicarlos y despublicarlos posteriormente cuando ya no se quieren mostrar.

2.3.5 Presentación

El Sistema Gestor de Contenido se encarga de gestionar los menús de navegación, la jerarquía de la página actual dentro del web o los módulos, internos o externos, que componen el sistema.

Al publicar el contenido, este seguirá el diseño definido para toda la web, o para una parte concreta, de forma que el resultado es un sitio web con un aspecto uniforme. Esta separación entre contenido y forma, permite desacoplar la capa de presentación y que se pueda modificar el aspecto visual del sitio sin afectar al contenido ya creado. En los Sistemas Gestores de Contenido más populares existen multitud de plantillas, tanto gratuitas como de pago, que permiten cambiar el aspecto de la página de forma fácil y sin tener que re-codificar la página.

Un Sistema Gestor de Contenido puede gestionar automáticamente la accesibilidad del web, con soporte de normas internacionales de accesibilidad, y adaptarse a las preferencias o necesidades de cada usuario, ya sea de forma nativa o mediante extensiones. Los Sistemas Gestores de Contenido soportan de forma nativa el multi-país y permiten la internacionalización del contenido permitiendo adaptarse al idioma, sistema de medidas y cultura del visitante.

2.4 Necesidad de los gestores de contenido

La flexibilidad y escalabilidad que permiten los sistemas gestores de contenido justifican su utilización en prácticamente cualquier tipo de web; ya que permiten, incluso en páginas sencillas, reducir el tiempo de puesta en funcionamiento.

Esta idea queda reforzada si se tiene en cuenta que hay soluciones libres muy completas que permiten elaborar y gestionar sus webs personales a usuarios finales, obteniendo webs dinámicos llenos de funcionalidades que tendrían un alto coste si se realizaran desde cero.

A continuación se detallan los puntos más importantes que ponen de manifiesto la utilidad de un Sistema Gestor de Contenido son:

Orientación a la extensión

Ampliar la funcionalidad de un sitio tradicional puede implicar la revisión/modificación de multitud de páginas y el desarrollo y pruebas del código que implementa las funcionalidades. En los Sistemas Gestores de Contenido, ampliar el funcionamiento puede implicar incluir un módulo realizado por terceros, sin que eso suponga muchos cambios en la web existente. El sistema permite crecer y adaptarse a las necesidades futuras a bajo coste.

Mantenimiento del contenido de las páginas

1 de julio de 2016

Un Sistema Gestor de Contenido facilita la distribución de los trabajos de creación, edición y mantenimiento del contenido del sitio, aplicando permisos de acceso a las diferentes partes.

- **Reutilización de objetos o componentes.** Un Sistema Gestor de Contenido permite la reutilización de páginas, documentos, y en general de cualquier objeto publicado o almacenado. Permite reutilizar componentes generados por terceros o la comunidad de usuarios.
- **Páginas interactivas.** Los Sistemas Gestores de Contenido trabajan de forma nativa orientados al contenido dinámico, disponen de una base de datos que hace de repositorio central de todos los datos de la web. En la actualidad con el auge de las redes sociales, los Sistemas Gestores de Contenido más populares disponen de extensiones que permiten mostrar en las páginas el contenido de las redes sociales o permiten publicar los contenidos en ellas.
- **Cambios del aspecto de la web.** En las páginas web tradicionales, la separación del contenido y la capa de presentación son responsabilidad del desarrollador, provocando en muchos casos que un cambio de diseño implique la revisión de muchas páginas para su adaptación. En los Sistemas Gestores de Contenido la capa de presentación está separada del contenido permitiendo cambiar el aspecto visual de forma ágil. Muchos de los Sistemas Gestores de Contenido actuales disponen de plantillas que permiten cambiar el aspecto visual, instalándolas como un componente más.
- **Consistencia de la web.** Los Sistemas Gestores de Contenido permiten aplicar a toda o parte de la web una misma estructura mediante plantillas, ayudando a la consistencia de la parte visual y unificando así como se visualizan las páginas de un sitio web.
- **Control de acceso.** Los Sistemas Gestores de Contenido disponen de mecanismos para controlar el acceso a la web y las acciones que se pueden realizar sobre la misma, permitiendo gestionar además los permisos de los usuarios sobre el contenido de la página. Los gestores de contenido permiten asignar roles a los diferentes usuarios del sitio web.

2.4.1 Criterios de selección de gestores de contenidos

Actualmente existen multitud de Sistemas Gestores de Contenido, tanto libres como propietarios, con características diferentes que permiten cubrir una tipología de webs u otra.

1 de julio de 2016

En el proceso de selección de un Sistema Gestor de Contenido, el primer paso es tener clara la necesidad a cubrir, teniendo claros:

- Los objetivos de la web que se quiere contruir.
- El público destinatario, incluyendo el canal de acceso mayoritario esperado a la página (web/móvil) según el público objetivo.
- Los requerimientos que tendría que poder satisfacer el Sistema Gestor de Contenido.

A continuación se exponen los criterios a valorar a la hora de seleccionar un Sistema Gestor de Contenido para un nuevo proyecto.

- **Código abierto.** Los Sistemas Gestores de Contenido han de poder ser ampliables, por tanto la selección tendría que ser de código fuente abierto (o libre). Esto facilita la generación de nuevas extensiones por parte de terceros, la comunidad de usuarios o incluso por parte de los creadores de la página, si estos tienen conocimientos técnicos.
- **Arquitectura.** Tiene que ser fiable y permitir la escalabilidad del sistema para adecuarse a futuras necesidades con módulos. Ha de implementar arquitecturas que separen el contenido, presentación y estructura, permitiendo modificar cualquier de ellos independientemente sin impactar en los otros.
- **Grado de madurez.** Es importante que el Sistema Gestor de Contenido escogido tenga la madurez suficiente y disponga de un amplio catálogo de módulos que le permitan añadir nuevas funcionalidades. Es importante ver cada cuanto tiempo aparece una versión nueva y cuánto tiempo hace desde que apareció la última versión a la hora de valorar la madurez de un sistema. Es importante también revisar los comentarios de las comunidades de usuarios para conocer si en la versión evaluada del sistema gestor se han detectado muchos fallos.
- **Soporte.** El sistema gestor de contenidos ha de disponer de un grupo de soporte, ya sea por parte del creador del sistema o de la comunidad de desarrolladores del mismo, de cara a poder encontrar respuestas ante posibles problemas en la implantación o uso o ayudar en la corrección de defectos. Un soporte de calidad garantiza que habrán futuras mejoras para el Sistema Gestor de Contenido.
- **Posicionamiento en el mercado.** Una herramienta con mucha difusión tiene mayor probabilidad de continuidad y de que sigan haciéndose nuevas extensiones y correcciones para ella.

1 de julio de 2016

- **Usabilidad.** La herramienta tiene que ser fácil de utilizar y aprender. En ocasiones los usuarios no serán técnicos, por lo tanto hace falta asegurar que la curva de aprendizaje no sea muy pronunciada para permitir utilizar la herramienta sin muchos esfuerzos y sacarle el máximo partido.
- **Accesibilidad.** Para asegurar la accesibilidad de una web, el Sistema Gestor de Contenido tendría que cumplir un estándar de accesibilidad. El más extendido es WAI (Web Accessibility Initiative) del World Wide Web Consortium.
- **Rendimiento.** El Sistema Gestor de Contenido ha de estar suficientemente optimizado para permitir el acceso a su contenido desde cualquier máquina, ya que en Internet no se puede controlar quien accede a la página.
- **Funcionalidades.** Un punto importante a valorar es las funcionalidades que provee el Sistema Gestor de Contenido, tanto de forma nativa como mediante extensiones, y ver si estas cubren todas las necesidades de la web. En términos generales un Sistema Gestor de Contenido actual, como mínimo, debería tener las siguientes funcionalidades:
 - Editor de texto WYSIWYG a través del navegador para la generación/modificación del contenido.
 - Herramienta de búsqueda en el contenido.
 - Comunicación entre los usuarios (foros, correo electrónico, chat).
 - Publicación de noticias.
 - Artículos. Incluyendo la gestión de la vigencia del artículo.
 - Gestión de ficheros multimedia.
 - Gestión de usuarios y adaptación del flujo de trabajo a los diferentes perfiles de usuarios y grupos de trabajo.
 - Sistema de avisos, tanto por correo electrónico como en la propia web.
 - Envío de páginas por correo electrónico o redes sociales.
 - Páginas en versión imprimible.
 - Disponibilidad o posibilidad de traducción a las lenguas del público al que se destina la web.
 - Soporte multinavegador, tanto para navegadores de ordenador como de móvil o tableta.
 - Soporte de sindicación (RSS, listas de email, etc.).
 - Estadísticas de uso e informes.

2.4.2 Ejemplo de gestores de contenido

A continuación se muestran los 10 gestores de contenido más usados de Internet y el número de webs que hacían uso de ellos a fecha de Junio de 2016. Los sistemas que aparecen en este listado proveen características diferentes que los hacen más adecuados

1 de julio de 2016

para un uso u otro. En esta lista aparecen sistemas gestores de contenido tanto de código abierto como propietario y construidos con diferentes tecnologías.

Fuente: <http://trends.builtwith.com/cms>, datos de 06/2016

Gestor de Contenido	# Webs '16	Código abierto	Tecnología
WordPress	12.233.915	Si	PHP, Javascript, HTML
Joomla!	2.018.386	Si	PHP, Javascript, HTML
Drupal	576.399	Si	PHP, Javascript, HTML
Blogger	532.694	No	jQuery, HTML5 y CSS3.
Google Search Appliance	135.108	No	No disponible
ExpressionEngine	41.904	No	No disponible
vBulletin	21.804	No	PHP, Javascript, HTML
Adobe CQ	15.165	No	Java Javascript, HTML
Liferay	10.335	Si	Java Javascript, HTML
Sitecore CMS	8.020	No	Microsoft. NET 3.5/4.0
Adobe Experience Manager	7.129	No	No disponible
dotCMS	2.980	Si	Java Javascript, HTML
Adobe Scene7	2.673	No	No disponible
HP TeamSite	1.358	No	No disponible
WordPress VIP	1.168	No	No disponible
Salesforce Desk	912	No	No disponible
Thomson Reuters Corporate Solutions	714	No	No disponible
Endeca	656	No	No disponible
Oracle RightNow	501	No	No disponible
MoovWeb	115	No	No disponible

Tabla 1. Ejemplo de gestores de contenido

1 de julio de 2016

3 Selección del gestor de contenidos.

De acuerdo con las estadísticas de uso de la fuente citada, los Sistemas Gestores de Contenido más usados para la generación de páginas web son WordPress, Joomla!, Drupal! Y Blogger, con mucha diferencia de popularidad respecto al resto de sistemas.

Fuente: <http://trends.builtwith.com/cms>, datos de 06/2016

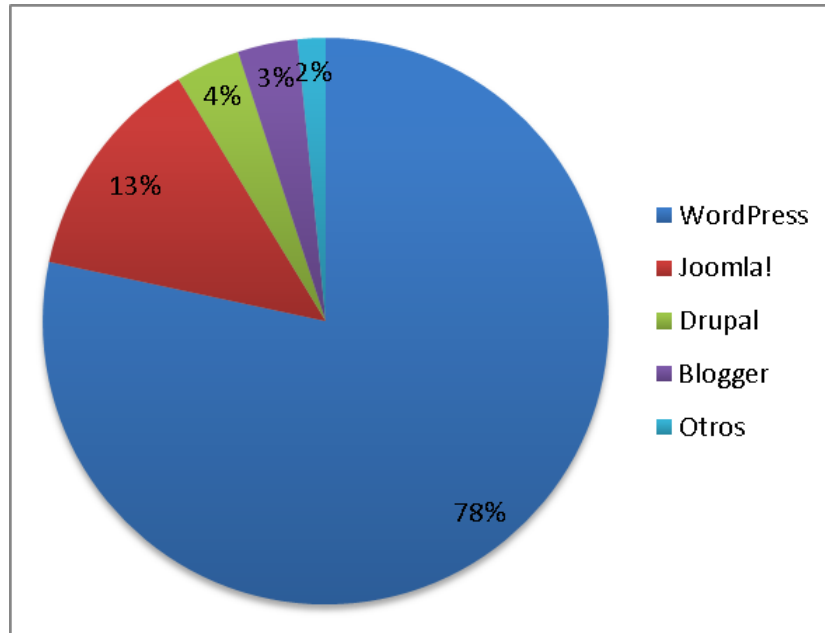


Ilustración 1. Estadísticas de uso de CMS - 06/2016

El de mayor difusión es WordPress gracias al boom de los blogs en los últimos años, ya que este sistema está especializado en la generación de este tipo de páginas. Blogger que ocupa la cuarta posición ofrece características similares.

En la segunda y tercera posición se encuentra Joomla! y Drupal, que son sistemas gestores de contenido más genérico y que pueden cubrir un abanico de necesidades mayor gracias a su flexibilidad y multitud de extensiones que permiten generar desde un sitio de noticias hasta un sitio de comercio electrónico.

En el marco del proyecto fin de carrera se ha focalizado el análisis sobre los tres Sistemas Gestores de Contenido más extendidos en la actualidad: WordPress, Joomla! y Drupal.

Los tres Sistemas Gestores de Contenido analizados tienen características particulares que los hacen más adecuados para un uso u otro, aunque comparten algunas de sus características:

- Son software libre y gratuito.
- Están desarrollados en PHP.
- Pueden correr en un servidor Apache o IIS, lo que permite multitud de opciones de hosting para alojar el sitio al ser las opciones más extendidas en la actualidad.

1 de julio de 2016

- Son Sistemas Gestores de Contenido muy maduros y estables (todos aparecieron hace más de 10 años)
- Tienen detrás una gran comunidad que da soporte, prueba las nuevas versiones y genera componentes para extender su funcionamiento.

3.1 Wordpress

Es el Sistema Gestor de Contenido ideal, y más difundido, para la creación de sitios web simples, como blogs o sitios de noticias ya que su facilidad de uso lo hace ideal para aquellos usuarios que quieren una web fácil de gestionar.

WordPress se inició en 2003, desde entonces, ha crecido hasta convertirse en la mayor herramienta de blog auto-alojado del mundo, que es utilizado por millones de sitios y visto por decenas de millones de personas cada día. WordPress comenzó como un simple sistema de blogs, pero ha evolucionado para ser utilizado como sistema de gestión de contenido completo a través de los miles de plugins, widgets y temas disponibles. La gran cantidad de extensiones disponibles permiten ampliar en gran medida su funcionalidad que por defecto está muy focalizada en la generación de blogs y sitios de noticias.

WordPress nació con la intención de crear un sistema de publicación personal elegante, con un sistema de publicación bien estructurado construido en PHP y MySQL y licenciado bajo la GPLv2 (o posterior). WordPress es un software moderno, pero sus orígenes se remontan al 2001, sucediendo a otro sistema gestor de contenidos existente b2/cafelog. Se trata de un producto maduro y estable, centrado en la experiencia del usuario y sigue los estándares actuales

WordPress es un proyecto Open Source, y tiene a cientos de personas en todo el mundo que trabajan en él. Es de uso libre sin tener que pagar un derecho de licencia y permite ser modificado. WordPress es completamente personalizable y se puede utilizar para casi cualquier cosa.

WordPress se puede descargar e instalar en un proveedor de alojamiento web que cumpla con los requisitos mínimos o se puede usar mediante el servicio web WordPress.com que permite empezar a trabajar con un nuevo blog libre basado en WordPress en cuestión de segundos, aunque esta opción es menos flexible que la versión descargable.

Para generar una web con WordPress no es necesario experiencia técnica, es un sistema intuitivo, sencillo y resulta sencillo poner en marcha un sitio rápidamente.

3.2 Drupal

En 2000, dos estudiantes de la Universidad de Amberes, Dries Buytaert y Hans Snijder, se dieron cuenta que necesitaban una herramienta para hablar con otros compañeros de la universidad. Dries comenzó un sitio de noticias con un pequeño tablón web, donde él y sus amigos podían publicar notas sobre el estado de su conexión red, hablar de donde estaban cenando, compartir noticias interesantes, etc.

1 de julio de 2016

A continuación se indican los valores y principios suscritos por Drupal y que condicionan tanto la arquitectura del sistema como de las extensiones que se desarrollen para él.

Valores

- Flexibilidad , simplicidad y utilidad en el producto
- El trabajo en equipo , la innovación , y la apertura a la comunidad
- Modularidad, extensibilidad y capacidad de mantenimiento del código

Principios

- Modular y extensible: Proporcionar un núcleo sencillo pero de gran alcance que se puede extender fácilmente a través de módulos personalizados
- Calidad de codificación: Alta calidad en el código, con código elegante y bien documentado.
- Basados en estándares: Soporte para los estándares consolidados y emergentes.
- Bajo consumo de recursos: Luchar por un excelente rendimiento mediante el cuidado en la codificación de las funciones de bajo nivel, p.e. las consultas a base de datos.
- Código abierto: comprometidos con el desarrollo de software libre y de consolidar y apoyar otros proyectos de código abierto
- Facilidad de uso: Hacer una plataforma fácil de usar para los desarrolladores, administradores, diseñadores y productores de contenidos.
- Colaboración: apoyo a los sistemas cooperativos abiertos de intercambio de información.

Drupal, y todos los archivos descargables en Drupal.org, están licenciados bajo la Licencia Pública General de GNU, versión 2 o posterior. Eso significa que se pueden descargar, reutilizar, modificar y distribuir los archivos con libertad.

Drupal es la opción más potente de las tres analizadas y está destinado a sitios avanzados, complejos y versátiles que requieran de una gestión de los datos compleja. Es adecuado también para plataformas comunitarias con múltiples usuarios o tiendas online, ya que es el más seguro de las tres opciones analizadas.

En el plano técnico es el más complejo de los tres sistemas analizadas y requiere de desarrolladores con más experiencia para explotar sus características. Se ha de remarcar que el sistema ha evolucionado en este sentido y que en las ultimas se ha simplificado su uso.

3.3 Joomla

Joomla nace el 17 de agosto de 2005 de una bifurcación de Mambo Open Source de la corporación Miro of Australia. Miro of Australia mantenía la marca Mambo y al principal grupo de desarrolladores en esa época.

1 de julio de 2016

El 22 de enero de 2008 se lanzó la versión 1.5 estable de Joomla, que incorporaba notables mejoras en el área de seguridad, administración y cumplimiento con estándares W3C, y fue la versión que diferenció a Joomla claramente de Mambo.

Joomla permite construir sitios web con más contenido y flexibilidad que WordPress, pero manteniendo en gran parte la sencillez de uso. La instalación y puesta en funcionamiento es relativamente sencilla y permite ser instalado por personas con pocos conocimientos técnicos.

Invirtiendo cierto tiempo en entender el Framework de Joomla, permitirá desarrollar nuevas extensiones o modificar el sistema o extensiones existentes para realizar sitios relativamente complejos.

Algunos usos para los que está pensado el Sistema Gestor de contenidos Joomla son los siguientes:

- Webs corporativas o portales
- Intranets corporativas y extranets
- Revistas, diarios y publicaciones online
- Comercio electrónico y reservas online
- Aplicaciones gubernamentales
- Sitios web de pequeñas empresas
- Web sites de organizaciones sin ánimo de lucro
- Portales comunitarios
- Webs de escuelas e iglesias
- Webs personales o familiares

3.4 Evaluación

3.4.1 WordPress

PROS

- ✓ **Amigable.** No es necesaria experiencia técnica para usarlo; es intuitivo y fácil de crear un sitio sencillo rápidamente. P.E. permite copiar/pegar texto de un documento de Microsoft Word manteniendo los estilos.
- ✓ **SEO.** Es una plataforma que promueve el uso de las mejores prácticas en SEO y tiene multitud de extensiones SEO de terceros para optimizar las publicaciones y el análisis de ciertas paginas focalizado en las palabras clave.
- ✓ **Adaptabilidad web.** Las plantillas de WordPress están muy optimizadas en su versión móvil. Esto asegura el acceso al sitio desde cualquier dispositivo.
- ✓ **Soporte.** Tiene una comunidad inmensa de usuarios que garantiza la respuesta a casi cualquier pregunta relacionada con WordPress.

1 de julio de 2016

CONTRAS

- ✗ **Seguridad.** Los sitios Wordpress han recibido bastantes ataques, y aunque en las últimas versiones ha mejorado en este aspecto siguen habiendo vulnerabilidades, especialmente en las extensiones de 3ros.
- ✗ **Actualizaciones.** El sistema de actualizaciones automáticas de WordPress puede desconfigurar el sitio.
- ✗ **Velocidad.** El tiempo de carga de las páginas web es lento ya que los sitios WordPress tienen gran cantidad de código genérico que muchas páginas no usan.

3.4.2 Drupal

PROS

- ✓ **Versátil.** Drupal puede dar respuesta a casi cualquier necesidad ya que permite extender su funcionalidad fácilmente y es muy ajustable.
- ✓ **Multi-site.** Drupal soporta varios sitios web desde un mismo código, aunque es bastante complejo de usar.
- ✓ **Escalabilidad.** Drupal permite ser escalado de forma sencilla.
- ✓ **Velocidad.** Dispone de un sistema de cache's avanzado gracias al cual se mejora la velocidad y rendimiento.

CONTRAS

- ✗ **Desarrolladores.** En caso de encontrar problemas durante el desarrollo, es más complicado encontrar ayuda de desarrolladores que en los otros dos Sistemas Gestores de Contenido .
- ✗ **Curva de aprendizaje.** Desarrollar en Drupal es difícil y requiere de una larga curva de aprendizaje para usarlo correctamente.

3.4.3 Joomla

PROS

- ✓ **Administración.** Joomla es el Sistema Gestor de Contenido con el panel de Administración más amigables. Tiene la mejor interface para la generación de contenido.
- ✓ **E-Commerce.** Tiene multitud de extensiones para transformar el gestor de contenidos en una tienda virtual.
- ✓ **Comunidad.** Joomla tiene una comunidad de desarrolladores muy extensa que continuamente están creando extensiones de código libre y gratuitos.
- ✓ **Extensiones.** Joomla dispone de una gran variedad de extensiones de todas las categorías.

CONTRAS

- ✗ **Mercado de extensiones reducido.** El mercado de extensiones de Joomla es más reducido que en los otros dos Sistemas Gestores de Contenido .

1 de julio de 2016

✖ **Compatibilidad de extensiones.** Es posible que aparezcan problemas de compatibilidad en las extensiones gratuitas que requieran de modificar el código PHP.

3.4.4 Elección

Para la implementación de la web objeto de este proyecto, se ha escogido el sistema gestor de contenido Joomla ya que es la opción que proporciona la relación ideal entre versatilidad y sencillez de desarrollo.

Joomla cubre de base o mediante extensiones la mayoría de las funcionalidades requeridas, lo que acota bastante aquellas que se han de desarrollar desde cero. El framework de Joomla tiene una curva de aprendizaje no muy pronunciada que permite empezar a desarrollar extensiones para el sistema en poco tiempo.

Finalmente, otro de los motivos por lo que Joomla se considera la opción más adecuada es que es la opción que tiene el panel de administración más sencillo y amigable, facilitando la generación de nuevos contenidos a usuarios sin conocimientos técnicos.

4 Especificaciones del alojamiento necesario para la web.

4.1 Descripción de la necesidad

Un alojamiento web es un servicio donde un proveedor te alquila un servidor o parte de él conectado a Internet y en el que puedes alojar todo tipo de ficheros que para que sean accesibles vía Internet.

El uso más típico de un alojamiento web es para ubicar en el los ficheros que componen un sitio web, aunque también puede ser usado para permitir la descarga de cualquier otra cosa: documentos, ficheros de audio, vídeo, etc.

Además de los servicios básicos de alojamiento de fichero, un servicio de hospedaje incluye en la mayoría de los casos otros servicios de valor añadido que permiten realizar sitios web más complejos y con mayor funcionalidad.

Los servicios de valor añadido más comunes son:

- Un servidor de correo electrónico que permite tener cuentas de correo con el nombre del dominio contratado.
- Alojamiento de aplicaciones web basadas en PHP y bases de datos que permiten hospedar webs con contenido dinámico.
- Acceso vía FTP para almacenar y descargar ficheros.
- Crear discos virtuales, es decir, crear almacenamiento en la nube con tu propio servicio de hosting al que accedes como si lo tuvieras en tu ordenador.

Los requisitos de hospedaje para Joomla han de contemplar para su funcionamiento de un servidor web y una base de datos, a continuación se detallan los pre-requisitos que ha de cumplir el servicio de alojamiento a contratar para la instalación de Joomla 3:

Software	Recomendado	Mínimo	Consideraciones
PHP	5.6 o 7.0 +	5.3.10	En la configuración de PHP ha de estar la opción de comillas mágicas deshabilitada. Las comillas mágicas son un procedimiento que automáticamente limpian los datos de entrada de un script PHP.
Bases de datos soportadas			
MySQL	5.5.3 +	5.1	Se requiere soporte para el formato InnoDB
SQL Server	10.50.1600.1 +	10.50.1600.1	

PostgreSQL	9.1 +	8.3.18	
Servidores Web soportados			
Apache	2.4 +	2.0	Es necesario tener instalados los módulos de Apache: mod_mysql, mod_xml, y mod_zlib. Para utilizar las URL SEO es necesario tener instalada la extensión de Apache mod_rewrite.
Nginx	1.8 +	1.0	
Microsoft IIS	7	7	Para Microsoft IIS, dependiendo de la configuración puede ser necesaria la siguiente configuración: <ul style="list-style-type: none"> • PHP 5.2 • MySQL 5.1 • Módulo Microsoft URL Rewrite, requerido solamente en caso de URLs SEO • FastCGI

Tabla 2. Requerimientos de Joomla 3

4.2 Posibles soluciones de infraestructura

A continuación se describen los diferentes tipos de soluciones de infraestructura que puede proporcionar un proveedor web para alojar el gestor de contenido Joomla, indicando para cada una de las soluciones sus características principales.

4.2.1 Hospedaje gratuito

Los alojamientos web gratuitos suelen incluir las prestaciones más importantes de los hospedajes de pago:

- Alojamiento de ficheros
- Acceso vía FTP
- Correo electrónico
- Soporte para aplicaciones PHP

Muchos de estos hospedajes proveen un dominio gratuito, basado en un subdominio del proveedor.

Ventajas

- ✓ Es una opción interesante para páginas de prueba o para aprender.

1 de julio de 2016

- ✓ Permite mantener un entorno de desarrollo para un entorno real a coste cero.

Desventajas

- ✗ La capacidad de proceso suele ser muy pequeña, disponen de poca memoria, y suelen tener severas limitaciones de algunos recursos básicos como espacio en disco o la transferencia mensual.
- ✗ El nivel de servicio proporcionado suele ser de muy baja calidad, con muchas caídas y paradas por mantenimiento. Es normal ya que suelen ser reclamos, con capacidades limitadas, para contratar un hospedaje de pago con el proveedor que proporciona el alojamiento gratuito.

Debido a las importantes desventajas no son una opción válida para un proyecto real y su uso se debería limitar a entornos de demostración o desarrollo.

4.2.2 Alojamiento compartido

En el alojamiento compartido varios clientes comparten una misma máquina aunque no tienen visibilidad entre ellos, es decir cada uno tiene su espacio privado y exclusivo, pero los recursos y la carga que generan sus webs en el servidor es compartida y por tanto una carga excesiva de una o varias de las webs puede afectar a otras.

Este tipo de alojamiento está orientado a proyectos web que no tengan que soportar una gran carga ni complejidad técnica ni de configuración y no necesitan, por tanto, toda la potencia de un servidor exclusivo. El alojamiento compartido permite en planes de distinta capacidad técnica (espacio web, cuentas de correo electrónico...) que permiten adaptar el servicio a la necesidad del cliente, estos planes se gestionan fácilmente desde un Panel de Control.

No es preciso conocimientos de administración de servidores, ya que el proveedor asume la gestión técnica de las máquinas, pero sí es necesario tener ciertos conocimientos informáticos, que dependerán del tipo de proyecto.

Ventajas

- ✓ Este tipo de hospedaje se adapta muy bien a las necesidades de uso personal y profesional y para pequeños negocios.
- ✓ Es una opción con una buena relación calidad/precio si se escoge el proveedor adecuado.
- ✓ Una cuenta de un proveedor de calidad aguanta mucho tráfico, más que suficiente para la gran mayoría de los casos.
- ✓ Un buen servicio de hosting suele ofrecer una disponibilidad muy razonable, es normal que se garantice una disponibilidad del 99,9% , es decir que el servidor se encuentra caído como máximo el 0,1% del tiempo (8,76 horas al año).

Desventajas

1 de julio de 2016

- ✗ El principal problema del hosting compartido es que abundan proveedores que explotan las capacidades de los servidores al máximo, tanto en número de clientes como de recursos asignados, proporcionando el servicio con servidores completamente infradimensionados.
- ✗ Debido a lo anterior, con algunos proveedores es frecuente encontrarse con problemas de lentitud, incapacidad para absorber picos de tráfico altos, inestabilidad (caídas frecuentes y de larga duración), etc.

Es la solución ideal para proyectos de pequeña envergadura ya que los costes son bajos, es la opción más adecuada para la enorme mayoría de los usuarios. Es muy importante escoger un proveedor que tenga cierto prestigio y que provea de un buen servicio.

4.2.3 Servidor Virtual Privado (VPS)

Un servidor virtual privado (Virtual Private Server o VPS) es un método de particionar un servidor físico en varios servidores de tal forma que todo funcione como si se estuviese ejecutando en una única máquina. Cada servidor virtual es capaz de funcionar bajo su propio sistema operativo y cada servidor puede ser reiniciado de forma independiente.

Ventajas

- ✓ Es un esquema muy flexible, capaz de adaptarse rápido a la demanda ya que ante mayores necesidades de recursos el proveedor puede aumentar los recursos asignados, aumentando también el coste.
- ✓ Al no estar compartido el servidor, no se puede ver comprometido por picos de actividad de otros clientes.
- ✓ La solución provee de un mayor control del servidor de hospedaje y permite crear varios espacios independientes de alojamiento que funcionan como varias cuentas de hospedaje compartido dentro del servidor VPS.
- ✓ Con un único servicio de hosting es posible alojar diferentes sitios web como cuentas de alojamiento diferenciadas y adaptadas a diferentes necesidades.

Desventajas

- ✗ El precio aumenta sensiblemente, mayor cuanto mayor son los recursos asignados.
- ✗ Las prestaciones de los VPS básicos suelen ser inferiores a las prestaciones de un hosting compartido de calidad. Esto es debido a que un hosting compartido de calidad ya ofrece unas prestaciones considerables en cuanto a recursos, simplemente no garantiza que éstas estén disponibles en todo momento, cosa que sí garantiza un hosting VPS.
- ✗ La complejidad técnica de administración del servidor aumenta y normalmente sólo está al alcance de personas con buenos conocimientos técnicos. El objetivo de un servicio VPS no consiste únicamente en mayores prestaciones y garantía de

1 de julio de 2016

servicios, sino que una de las características más importantes es el mayor control sobre el servidor.

Es recomendable en proyectos que requieren mayor flexibilidad y capacidad que un servicio de hosting compartido, pero no necesitan todos los recursos técnicos de un servidor. Los VPS requieren conocimientos básicos de administración de servidores, pero también se manejan desde una plataforma online con las principales opciones que se pueden encontrar en un servidor físico, aunque no son recomendables para usuarios sin conocimientos técnicos

4.2.4 Servidor Dedicado

En el servidor dedicado, el proyecto utiliza en exclusiva todos los recursos de un único servidor o de un número concreto de servidores, así que está destinado a proyectos cuyo alojamiento requieran una capacidad técnica destacada. Son una buena opción para empresas que realizan un uso intensivo de recursos o agencias de diseño web que agrupan sus desarrollos en un único servidor para mayor comodidad o re-vender espacio web.

Las ventajas y desventajas son similares a las de un VPS, ya que la diferencia es que la maquina contratada no es virtual sino física.

Los servidores dedicados ofrecen más potencia pero requieren de conocimientos de administración de sistemas, aunque se puede delegar esta función en el proveedor, no es aconsejable para usuarios sin conocimientos técnicos. Es importante remarcar Según los requerimientos de la maquina contratada pueden tener un coste elevado.

4.2.5 Cloud Hosting

Es el hospedaje con los mayores ratios de disponibilidad, capacidad técnica y escalabilidad del mercado, ya que los proyectos no dependen de una única máquina, con una u otra capacidad concreta, sino que están en una nube que pone todos sus recursos a disposición de los clientes y lo hace como si fuera un servicio. De este modo, un proyecto utiliza los recursos que necesita exactamente en cada momento. Aportan también otros beneficios como una elevada redundancia. Todas las opciones se administran desde un Panel de Control.

Ventajas

- ✓ Las ventajas propias del concepto de “Cloud hosting”: sobre todo, seguridad, disponibilidad por redundancia y una gran flexibilidad para crecer fácil y rápidamente ante un incremento de la demanda (escalabilidad).
- ✓ En la actualidad existe una oferta variada y con muy buena relación calidad/precio.
- ✓ Debido a su gran flexibilidad es un alojamiento web capaz de adaptarse de manera muy rápida y sencilla a las necesidades cambiantes del usuario.

1 de julio de 2016

Desventajas

- ✘ Las opciones de entrada suelen tener un precio mayor que las opciones de hosting compartido con similares características.

El hospedaje en la nube está altamente recomendado para nuevos proyectos tecnológicos o empresas que necesitan mucha capacidad en momentos muy concretos o prevén necesidades de escalabilidad altas. Permite un servicio de mucha potencia a un coste más reducido que en un servidor dedicado.

4.2.6 Hospedaje especializado/gestionado

Debido al aumento de popularidad entre usuarios no técnicos de ciertas aplicaciones web como WordPress, Joomla, Prestashop o Magento ha ido aumentando el uso de la opción de alojamiento especializado. Estos servicios van más allá de proveer únicamente el alojamiento y ofrecen un soporte integral que no se limita a los servidores en sí, sino que incluye también el soporte técnico para garantizar el correcto funcionamiento de estas aplicaciones.

En el alojamiento gestionado se realiza una administración proactiva de la aplicación sobre la que se presta el servicio. Es decir, el proveedor administra el hospedaje y la aplicación (sus actualizaciones, parches, configuración de seguridad, solucionar problemas, etc.). Estos servicios se conocen también como servicios “llave en mano” el cual pretende que el cliente solo se deba centrar en los contenidos del sitio.

Ventajas

- ✓ Estos servicios están optimizados para la aplicación sobre la que proveen el servicio.
- ✓ Eliminan el esfuerzo de las tareas más pesadas y complejas para alguien no técnico como la instalación inicial, mantenimiento y seguridad de tu aplicación (en diferentes grados, según el nivel de servicio contratado).
- ✓ Al ser especialistas en la aplicación conocen los problemas frecuentes y la mejor manera de optimizarlos.
- ✓ Mejor funcionamiento a nivel de velocidad, estabilidad, seguridad ... de la aplicación web y reducción del trabajo para el dueño del sitio web.

Desventajas

- ✘ La principal desventaja de este tipo de hospedajes es que en el caso de los alojamientos gestionados aumenta el precio sensiblemente.

Esta solución es ideal para usuarios sin conocimientos técnicos que quieran olvidarse del hospedaje y el mantenimiento de la aplicación y estén dispuestos a pagar el sobre coste que ello conlleva.

1 de julio de 2016

4.3 Infraestructura seleccionada para el proyecto

Después de analizar las distintas opciones, se ha determinado que para el proyecto de la web de la página del club de lectura el alojamiento más adecuado es el alojamiento compartido ya que se trata de un proyecto pequeño y en el que no se prevén unas necesidades a nivel de potencia elevadas, gracias a los costes altamente competitivos de este tipo de hospedajes los costes recurrentes del proyecto serán mínimos.

En caso de crecer el número de accesos y por tanto crecer la demanda de potencia y para mantener los tiempos de respuesta dentro de unos rangos aceptables (menos de 5 segundos), se podría aumentar las prestaciones del hospedaje ya que el proveedor permite cambiar de tipo de alojamiento compartido a VPS o dedicado de forma transparente.

1 de julio de 2016

5 Fase de desarrollo

5.1 Herramientas utilizadas

A continuación se detallan las herramientas utilizadas en el desarrollo de este proyecto, todas ellas son de libre uso, aunque no todas son de código abierto.

Aplicación	Versión	Descripción	Licencia
Notepad++	6.8.6	Es un editor de texto y de código fuente libre con soporte para varios lenguajes de programación	GPL
Paint.net	4.0.9	Es un editor de imágenes para Windows, desarrollado en el marco de trabajo .NET. Permite el uso de capas y transparencias.	Freeware
WinScp	5.7.7	WinSCP es un cliente SFTP gráfico para Windows que emplea SSH, también puede usarse con SCP. Su función principal es facilitar la transferencia segura de archivos entre dos sistemas informáticos, el local y uno remoto que ofrezca servicios SSH.	GPL
PhpMyAdmin	3.5.2.2	Es una herramienta para manejar la administración de MySQL a través de páginas web, utilizando Internet. Permite crear y eliminar Bases de Datos, crear, eliminar y modificar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios y exportar datos en varios formatos	GPL Versión 2
Oracle VirtualBox	5.0.22	Oracle VM VirtualBox es un software de virtualización para arquitecturas x86/amd64, por medio de esta aplicación es posible instalar sistemas operativos adicionales, conocidos como «sistemas invitados», dentro de otro sistema operativo «anfitrión», cada uno con su propio ambiente virtual.	Privativa / GPL (Opcionalmente CDDL para la mayoría de archivos en la fuente)

Tabla 3. Herramientas utilizadas en el proyecto

5.2 Estructura de un componente Joomla

Los componentes del gestor de contenido Joomla siguen el patrón de Arquitectura de Modelo Vista Controlador (MVC) y están desarrollados con los lenguajes PHP, Javascript y HTML.

A lo largo de este bloque se explicará el funcionamiento del patrón MVC y como se implementa en Joomla para el desarrollo de nuevos componentes.

5.2.1 Patrón Modelo - Vista - Controlador

El MVC o Modelo-Vista-Controlador es un patrón de arquitectura de software que, utilizando tres componentes separa la lógica de la aplicación de la capa de presentación en una aplicación.

- Modelos
- Vistas
- Controladores

Es una arquitectura importante ya que es utilizado en multitud de situaciones desde componentes gráficos básicos hasta aplicaciones empresariales complejas; actualmente la mayoría de los frameworks modernos utilizan MVC o alguna adaptación del MVC para la implementación de su arquitectura.

El patrón MVC fue inventado en el contexto de Smalltalk para realizar una separación entre la interfaz gráfica y el código del funcionamiento de una aplicación. Este patrón impacto en gran medida en el código de SmallTalk.

5.2.1.1 Modelo

El modelo representa la parte de la aplicación que implementa la lógica de negocio, es responsable de recuperar los datos y convertirlos en conceptos significativos para la aplicación, así como su procesamiento, validación, asociación y cualquier otra tarea relativa a la manipulación de dichos datos.

Los objetos del modelo en general representan los principales conceptos, ya que modelan las entidades del dominio del sistema en torno a las cuales se desea implementar un programa. Es la capa que interacciona, en caso de existir, con la base de datos utilizada por la aplicación.

5.2.1.2 Vista

La vista hace una presentación de los datos del modelo, pero estando separada de los objetos del modelo. Es responsable del uso de la información disponible para producir cualquier interfaz de presentación de cualquier petición recibida.

1 de julio de 2016

Puede haber más de una vista para procesar un mismo conjunto de datos. La capa de la Vista no se limita únicamente a la presentación en formato HTML o texto que represente los datos, sino que puede ser utilizada para ofrecer una amplia variedad de formatos en función de sus necesidades: videos, música, documentos, etc.

5.2.1.3 Controlador

La capa controlador gestiona las peticiones realizadas por los usuarios y es la capa responsable de responder con la información solicitada, apoyándose tanto en la capa del modelo como la capa de la vista.

La capa controlador espera las peticiones de los clientes, comprueba su validez de acuerdo a las normas de autenticación o autorización, delega la búsqueda de datos al modelo y selecciona el tipo de respuesta más adecuado según las preferencias del cliente. Finalmente una vez se ha preparado la información de respuesta delega el proceso de presentación a la capa de la Vista.

5.2.1.4 Funcionamiento básico

En el patrón MVC el ciclo de una petición comienza cuando un usuario solicita una página o un recurso de la aplicación, esta solicitud es procesada por un **despachador** que selecciona el **controlador** correcto para procesar la petición.

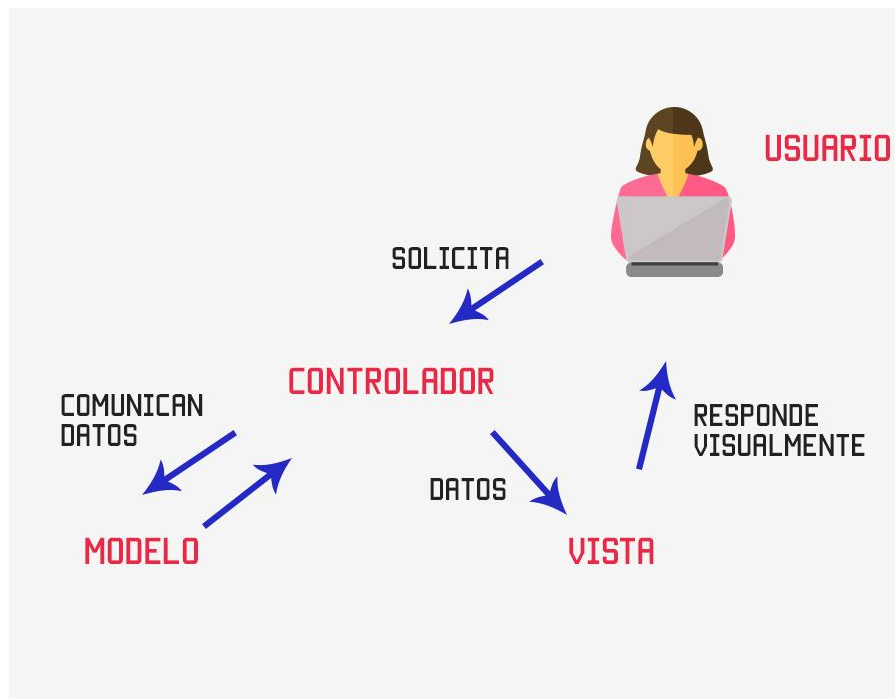


Ilustración 2 . Esquema de funcionamiento del MVC

Cuando la solicitud llega al **controlador**, éste se comunicará con la capa del **modelo** para cualquier proceso de recuperación de datos o el guardado de los mismos según se sea necesario.

1 de julio de 2016

Una vez finalizada esta comunicación el **controlador** procederá a delegar en el objeto de **vista** correcto la tarea de generar una presentación con los datos proporcionados por el modelo. Finalmente, cuando esta presentación se genera, se envía de inmediato al usuario.

5.2.1.5 Beneficios

- ✓ Es un patrón de diseño de software muy extendido y probado.
- ✓ Convierte una aplicación en un paquete modular fácil de mantener. Las nuevas características se pueden añadir fácilmente, añadir nuevas funcionalidades a código existente es más sencillo.
- ✓ Mejora la rapidez del desarrollo. El diseño modular permite a los desarrolladores y los diseñadores trabajar simultáneamente, incluyendo la capacidad de hacer prototipos rápidos.
- ✓ La separación del código de la aplicación en modelos, vistas y controladores hace que la estructura de la aplicación sea más fáciles de entender.
- ✓ La separación también permite a los desarrolladores hacer cambios en una parte de la aplicación sin afectar al resto.

5.2.2 Estructura de ficheros básica de un componente

A continuación se describe como se ha de implementar el patrón MVC en Joomla a la hora de crear un nuevo componente, identificando los ficheros que se han de crear en cada caso.

Para crear el componente que se instalará en Joomla se deberá crear una carpeta con el nombre "*com_<nombre-componente>*", dentro de esta se generará la siguiente estructura de ficheros:

Fichero	Descripción
<i><nombre-componente>.xml</i>	Este es un archivo XML (manifiesto) que le indica a Joomla! cómo instalar nuestro componente.
<i>site/<nombre-componente>.php</i>	Este es el punto de entrada del componente que se está creando, en la parte del sitio web donde se ubica el contenido.
<i>site/index.html</i>	Evita que se liste el contenido del directorio del servidor web.
<i>admin/index.html</i>	Evita que se liste el contenido del directorio del servidor web.

1 de julio de 2016

<i>admin/<nombre-componente>.php</i>	Este es el punto de entrada del componente que se está creando, en la parte de administración del sitio web.
<i>admin/sql/index.html</i>	Evita que se liste el contenido del directorio del servidor web
<i>admin/sql/updates/index.html</i>	Evita que se liste el contenido del directorio del servidor web
<i>admin/sql/updates/mysql/index.html</i>	Evita que se liste el contenido del directorio del servidor web
<i>admin/sql/updates/mysql/0.0.1.sql</i>	Es un archivo que permite inicializar el esquema de versiones del componente que se está desarrollando.

Tabla 4. Estructura de ficheros básica de un componente Joomla

El componente desarrollado se instalará siguiendo el procedimiento standard de instalación de componentes en Joomla.

5.3 Diseño del componente com_lectura

A continuación se detallan los temas relativos al diseño del componente *com_lectura*, encargado de proveedor la búsqueda, personalizada para este proyecto, de libros por país.

5.3.1 Tecnologías utilizadas

Las tecnologías utilizadas para el desarrollo de este componente de Joomla son:

5.3.1.1 PHP5

PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

Las páginas de PHP contienen HTML con código incrustado que implementa alguna funcionalidad. El código de PHP está encerrado entre las etiquetas de comienzo y final `<?php` y `?>` que permiten entrar y salir del contexto PHP.

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Ejemplo</title>
  </head>
  <body>
    <?php
      echo "¡Hola, soy un script de PHP!";
```

1 de julio de 2016

```
?>  
</body>  
</html>
```

En PHP el código se ejecuta en el servidor, generando el código HTML y enviándolo al cliente. El cliente recibirá el resultado de ejecutar el script sin tener visibilidad sobre el código PHP que lo generó.

PHP5 incorporó un nuevo motor llamado Zend II con un nuevo modelo de objetos enfocado a mejorar el rendimiento de PHP en las aplicaciones complejas al mismo tiempo que permite una mayor modularidad.

5.3.1.2 HTML5

HTML5 (HyperText Markup Language, versión 5) es la quinta revisión importante del lenguaje básico de la World Wide Web, HTML, la versión definitiva de la quinta revisión del estándar se publicó en octubre de 2014.

HTML5 establece una serie de nuevos elementos y atributos que reflejan el uso típico de los sitios web modernos. Algunos de ellos son técnicamente similares a las etiquetas `<div>` y ``, pero tienen un significado semántico, como por ejemplo `<nav>` (bloque de navegación del sitio web) y `<footer>`.

Una de las características más importantes es Canvas (para dibujar gráficos en la web), ya que se integra en HTML5 si necesidad de tener que añadir otras aplicaciones como Flash.

5.3.1.3 Javascript

JavaScript (abreviado JS) es un lenguaje ligero e interpretado, orientado a objetos con funciones de primera clase, es sobretodo conocido como el lenguaje de script para páginas web, pero también usado en muchos entornos sin navegador, tales como node.js o Apache CouchDB.

Javascript es un lenguaje script con las siguientes características:

- Multi-paradigma
- Basado en prototipos
- Dinámico
- Soporta estilos de programación
 - Funcional
 - Orientada a objetos
 - Imperativa

1 de julio de 2016

El estándar de JavaScript es ECMAScript, soportado desde el 2012 por todos los navegadores modernos (ECMAScript 5.1). Los navegadores más antiguos soportan por lo menos ECMAScript 3.

Joomla además de utilizar Javascript estándar utiliza una librería Javascript llamada JQuery, jQuery es una biblioteca de funciones de JavaScript que incluye algunas de las funciones más usadas. Es una librería muy versátil y ampliable.

Algunas de las funciones más populares del amplio catálogo que ofrece esta librería son:

- Recorrer y manipular el documento HTML
- El manejo de los eventos.
- Realizar animaciones con los objetos de la página (forma en la que se abren las ventanas emergentes, como se despliegan los menús, etc...)
- Gestión de las peticiones Ajax con una API, sencilla de usar y que funciona a través de una multitud de navegadores.

5.3.1.4 CCS3

CSS (Cascading Style Sheets) es un lenguaje para definir el estilo o la apariencia de las páginas web, escritas con HTML o de los documentos XML. CSS se creó para separar el contenido de la forma, a la vez que permite a los diseñadores mantener un control mucho más preciso sobre la apariencia de las páginas.

A grandes rasgos las mejoras que incorpora CCS3 respecto a anteriores versiones con el objetivo de que sea más sencillo posicionar los controles dentro de la página y que cuenten con otras características como desplazamiento.

- **Nuevas propiedades**
 - Bordes: border-color, border-image, border-radius, box-shadow.
 - Backgrounds: background-origin, background-clip, background-size, layering multiple background images.
 - Color: HSL colors, HSLA colors, RGBA colors opacity.
 - Texto: text-shadow, text-overflow.
 - Interface: box-sizing, resize.
 - Selectores: attribute selectors.
 - Formatos: media queries, multiple column layout, speech.
- **Fondos y Bordes:** Permite agregar múltiples fondos a un objeto, los bordes podrán ser redondos con posibilidad de indicar el radio de curvatura.
- **Diseño multi-columna:** Hace más sencillo crear diseños con múltiples columnas sin etiquetas adicionales. Esta nueva característica permite organizar

1 de julio de 2016

los elementos en pantalla mejor y combinarlos de diferentes maneras sin etiquetas adicionales.

- **Posicionamiento en red:** Permite agregar reglas con cuadrículas invisibles para maquetar páginas con un aspecto de libro o revista y jugar con el tamaño y las propiedades de los objetos para hacer que los elementos “floten” entre las posiciones.
- **Selectores:** Aparecen nuevos selectores:
 - E:only-of-type: un elemento que es único en su tipo.
 - E:not(s): un elemento que no coincide con los selector simples.
 - E ~ F: un elemento F precedido de un elemento E.
 - E:nth-child: un objeto que es el enésimo hijo del nodo padre.
 - E:nth-last-child: un objeto que es el último hijo del nodo padre.
 - E:nth-of-type: un elemento que es el enésimo nodo hijo de otro elemento.
 - E.first-of-type: el primer elemento de un tipo.

A continuación se explican los conceptos básicos, aplicados en el desarrollo de este proyecto. La nomenclatura estándar de CSS que consiste en un selector y un bloque de declaración:

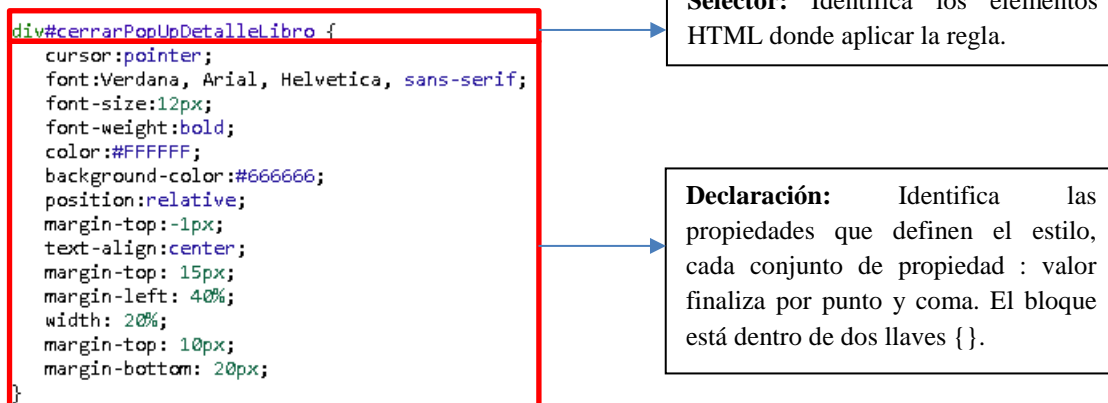


Ilustración 3. Instalación estilos. Definición CSS

En los estilos definidos en este proyecto se han usado los siguientes selectores, el selector marca a que elementos se aplicará el estilo:

- **Selección por id:** Se aplica la regla a los elementos que tienen el identificador HTML indicado, normalmente un único elemento de la página.

```
#bottom-d{
  width: 300px;
  float: right;
  display:block;
}
```

1 de julio de 2016

- **Selección por clase:** Se aplica la regla a todos los elementos que tienen en la etiqueta HTML la clase indicada, pueden ser diferentes tipos de elementos.

```
.uk-container{
  max-width: inherit;
  box-sizing: initial;
  text-align: right;
}
```

- **Selección de grupo:** Se aplica la regla a los elementos del grupo indicado en el selector.

```
h2{
  font-family: Rockwell;
  font-size: 24px;
}
```

En la definición de los estilos se han utilizado también selectores combinados que permiten restringir los elementos HTML donde se aplica la regla. Tienen especial importancia a la hora de sobrescribir los estilos existentes en la plantilla.

```
h1.tm-title{
  display:none;
  font-family: 'Enriqueta', arial, serif;
  line-height: 1.25;
  margin: 0 0 10px;
  font-size: 40px;
  font-weight: bold;
  background-color: #c91032;
  padding-right: 30px;
  text-shadow: 1px 2px 5px #000;
  color: white;
  padding-left: 10px;
}
```

A la hora de crear los estilos de este proyecto se ha tenido especial cuidado en la prioridad de estilos CSS, ya que es lo que permite modificar reglas existentes sin tener que modificar las hojas de estilos de la plantilla, únicamente trabajando con la hoja *custom.css* y usando selectores más restrictivos que los usados en las propiedades a sobrescribir.



Ilustración 4. Instalación estilos. Gestión de estilos. Prioridad CSS

Si a una declaración CSS se le añade la palabra reservada *!important*, se aumenta su prioridad.

1 de julio de 2016

5.3.1.5 Desarrollo de las paginas

A continuación se indican las tecnologías utilizadas, tanto en el Sistema Gestor de Contenidos como en el componente desarrollado, para desarrollar los ficheros que componen la capa visual.

- PHP 5
- HTML
- Javascript
- CCS3

5.3.1.6 Desarrollo de la lógica de negocio

A continuación se indican las tecnologías utilizadas, tanto en el Sistema Gestor de Contenidos como en el componente desarrollado, para desarrollar los ficheros que implementan la lógica de negocio.

- PHP5

5.3.1.7 Desarrollo de la capa de persistencia

A continuación se indican las tecnologías utilizadas, tanto en el Sistema Gestor de Contenidos como en el componente desarrollado, para desarrollar los ficheros la capa de persistencia, es decir, la capa encargada de gestionar el acceso a la base de dato.

- PHP5 (uso de framework de Joomla!)
- SQL

5.3.2 Estructura de ficheros

Joomla implementa el patrón Vista-Controlador, por tanto sus componentes también lo hacen. En el caso del componente creado:

- **Vista:** la capa de presentación está formada por los ficheros y subcarpetas que cuelgan de la carpeta *views*.
- **Controlador:** está formado por el fichero *controller.php*
- **Modelo:** está formado por el fichero *lectura.php*.

5.3.2.1.1 Ficheros del Sitio

A continuación se listan los ficheros que implementan la parte publica del componente, la que ven los usuarios de la página, indicando para cada uno de ellos si corresponden a la Vista, Controlador o Modelo.

Capa	Fichero	Descripción
Vista	views/lectura/tmpl/dist/css/jqvmap.css	Contiene los estilos utilizados por el componente <i>jquery.vmap</i> .

1 de julio de 2016

Vista	views/lectura/tmpl/dist/js/custom.js	Contiene las funciones sobre escritas para modelar las acciones a realizar al interactuar con el mapa.
Vista	views/lectura/tmpl/dist/js/jquery.vmap.min.js	Fichero Javascript con la librería <i>jquery.vmap</i> comprimida, es decir se han eliminado los espacios y los saltos de línea.
Vista	views/lectura/tmpl/dist/js/jquery.vmap.js	Fichero Javascript con la librería <i>jquery.vmap</i> sin comprimir.
Vista	views/lectura/tmpl/dist/maps/jquery.vmap.world.js	Contiene los datos que modelan el mapamundi y los datos a cargar sobre el mapamundi, nombre del país, coordenadas de la área que ocupa en la imagen y acrónimo.
Vista	views/lectura/tmpl/default.php	Plantilla PHP que modela el aspecto del componente.
Vista	views/lectura/view.raw.php	Vista utilizada en las llamadas Ajax, en la que no se añaden etiquetas HTML.
Vista	views/lectura/view.html.php	Vista utilizada en la carga de las pantallas del componente, añade los etiquetas HTML correspondientes al contenedor de Joomla.
Controlador	controller.php	En el controlador se implementan los servicios que se encargan de interactuar con los modelos para dar los datos a la vista. Los servicios llamados en la vista se implementan en este fichero.
Controlador	lectura.php	Punto de entrada al componente, se encarga de la creación del objeto controlador.
Modelo	models/lectura.php	Contiene los servicios que se encargan de la lectura/ escritura/ modificación a la base de datos.

Tabla 5 . Ficheros del sitio

Todas las carpetas tienen un fichero *index.html* vacío para evitar que se puedan listar los directorios accediendo a ellos desde el navegador.

1 de julio de 2016

El componente, en su capa de presentación, utiliza el componente *jquery.vmap*, encargado de proveer el mapa interactivo que al ser clicado uno de sus países invoca una función encargada de recuperar los datos. La función llamada mediante una petición Ajax solicitara al servicio “getLibrosPais” del controlador los datos y después procesara los resultados para mostrarlos en la vista.

5.3.2.1.2 Ficheros de la administración

A continuación se listan los ficheros que implementan la parte de Administración del componente, la que ven los administradores de la página, indicando para cada uno de ellos si corresponden a la Vista, Controlador o Modelo.

Capa	Fichero	Descripción
Vista	<i>views/lectura/tmpl/default.php</i>	Plantilla PHP que modela el aspecto visual del componente.
Vista	<i>views/lectura/view.html.php</i>	Vista utilizada en la carga de las pantallas del componente, añade los etiquetas HTML correspondientes al contenedor de Joomla.
Controlador	<i>controller.php</i>	En el controlador se definen los servicios que podrán ser llamados desde la página. Para realizar el listado, alta, baja y modificación de nuevas entradas.
Controlador	<i>lectura.php</i>	Punto de entrada al componente, se encarga de la creación del objeto controlador.
Modelo	<i>models/lectura.php</i>	Contiene los servicios que se encargan de la lectura/ escritura/ modificación a la base de datos.
N/A	<i>sql/updates/mysql/0.0.1.sql</i>	Fichero con el script SQL para la preparación de la base de datos. Añade las tablas necesarias para el correcto funcionamiento del sistema.
N/A	<i>lectura.xml</i>	Descriptor del componente.

Tabla 6. Ficheros de la administración

Todas las carpetas tienen un fichero *index.html* vacío para evitar que se puedan listar los directorios accediendo a ellos desde el navegador.

5.3.2.2 Instalable

A continuación se indica la estructura de carpetas que ha de tener el fichero usado para realizar la instalación del componente en Joomla.

1 de julio de 2016

Para poder instalar el componente mediante el gestor de extensiones de Joomla, se generará un fichero comprimido (.zip) que contendrá la siguiente estructura de ficheros

- **lectura.xml** –Fichero XML que contiene la descripción de como se ha de instalar el componente indicando que ficheros se han de ubicar en la ruta de ficheros de la parte publica y que ficheros van en la parte de administración. Se indica ademas que scripts SQL se han de ejecutar para crear los elementos de base de datos necesarios.
- **site** – Es la carpeta que contiene la estructura de ficheros de la parte visible por los usuarios comunes. El listado de ficheros se puede encontrar en el apartado “Ficheros del Sitio”
- **admin** – Es la carpeta que contiene la estructura de ficheros de la parte visible por los usuarios administradores. El listado de ficheros se puede encontrar en el apartado “Ficheros de la administración”.

Una vez instalado el componente en el sistema Joomla los ficheros desarrolladores se ubicaran en las siguientes estructuras de carpetas:

- La estructura de ficheros de “Ficheros del Sitio” se ubicará en */public_html/components/com_lectura*.
- La estructura de ficheros de “Ficheros de la administración” se ubicará en */public_html/administrator/components/com_lectura*.

5.3.3 Diagrama de componentes

A continuación se muestra la relación conceptual entre los diferentes bloques del patrón modelo-vista controlador utilizado tanto en el Sistema Gestor de Contenidos Joomla como en el componente *com_lectura*, en el caso de los ficheros correspondientes a la extensión desarrollada se puede ver que ficheros del componente corresponden a cada bloque en el apartado “5.3.2 Estructura de ficheros”:

1 de julio de 2016

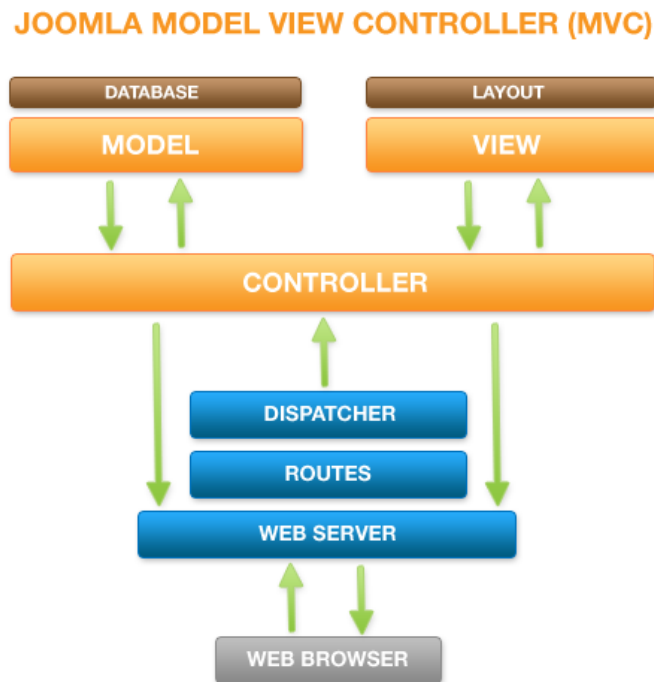


Ilustración 5. Modelo de comunicación entre capas

El punto de entrada al componente es el fichero:

/components/com_lectura/lectura.php

Este fichero se encarga de crear el objeto controlador del componente. Como se puede apreciar en la Ilustración “5. Modelo de comunicación entre capas” el controlador se encarga de coordinar la interacción con el resto de capas, modelo y vista, para generar la respuesta que devolverá después al servidor web. El controlador del componente creado se corresponde con el fichero:

/components/com_lectura/controller.php

Este fichero contiene la clase `LecturaController` que extiende de la clase del framework de Joomla `JControllerLegacy`, esta clase es el controlador de la extensión `com_lectura`.

La capa de modelo, que se encargará de la interacción con la base de datos, se encuentra implementada en el fichero:

com_lectura/models/lectura.php

1 de julio de 2016

Este fichero contiene la clase `LecturaModelLectura` que extiende de la clase del framework Joomla `JModelItem`. Esta clase contiene las funciones que acceden a la base de datos para recuperar la información.

La vista, que se encargará de generar la capa de presentación, esta implementada principalmente en los ficheros:

```
com_lectura/views/lectura/view.raw.php
com_lectura/views/lectura/view.html.php
com_lectura/views/lectura/tmpl/default.php
```

Los ficheros `view.*.php` contienen la clase `LecturaViewLectura` que es la encargada de preguntar al modelo mediante el método “get” de la clase `JView`, la información a cargar en la página, definida en el fichero `default.php`.

La diferencia entre los dos ficheros `view.*.php` es que el fichero `view.raw.php` tiene la función `display` vacía, con el objetivo de no cargar la plantilla `default.php` ni el código HTML del gestor de contenido ya que es usado para las llamadas Ajax. En el fichero `view.html.php` la función `display` invoca a la función `display` de la clase padre `JViewLegacy` pasando como parámetro la plantilla.

A continuación se muestra un diagrama en el que se puede ver como el componente `com_lectura`, en verde en el gráfico, se comunica con los componentes del framework Joomla a la hora de generar el contenido de las páginas.

1 de julio de 2016

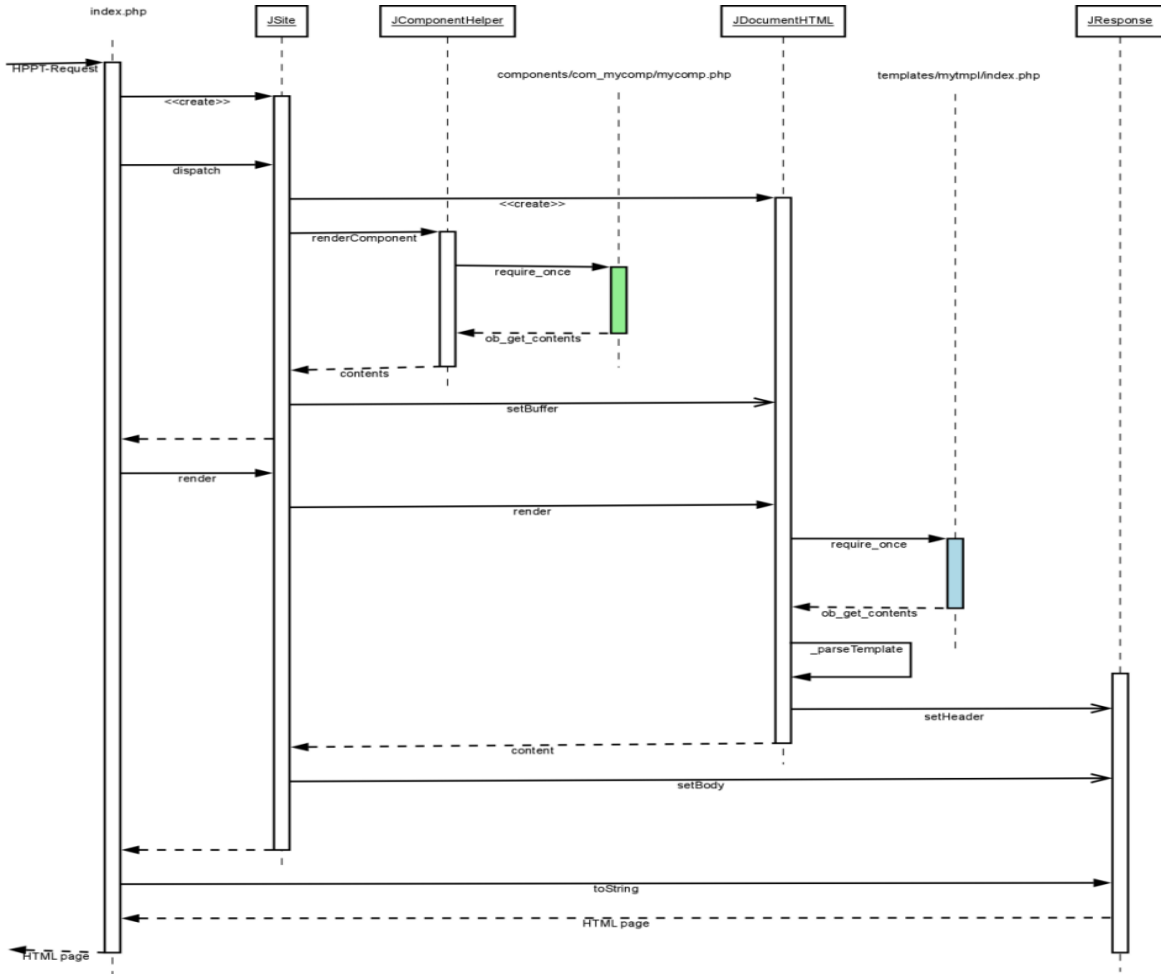


Ilustración 6. Diagrama de comunicación Joomla Framework

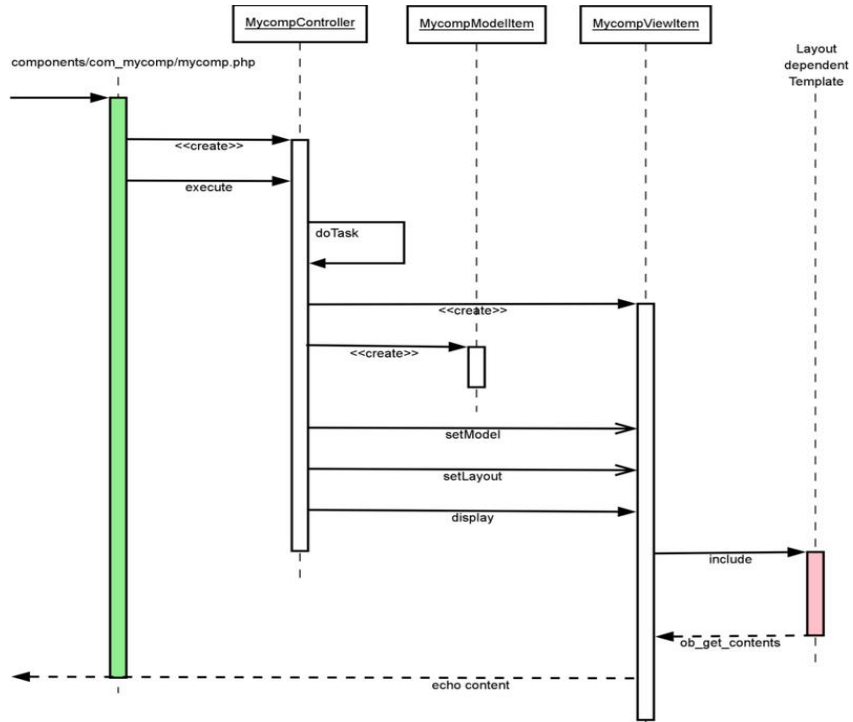


Ilustración 7 . Diagrama de comunicación de un componente Joomla

5.3.4 Modelo entidad-relación

A nivel de base de datos para el uso del componente se han creado tres entidades específicas, en el siguiente diagrama no se tienen en cuenta las tablas de Joomla necesarias para el normal funcionamiento del Sistema Gestor de Contenido:

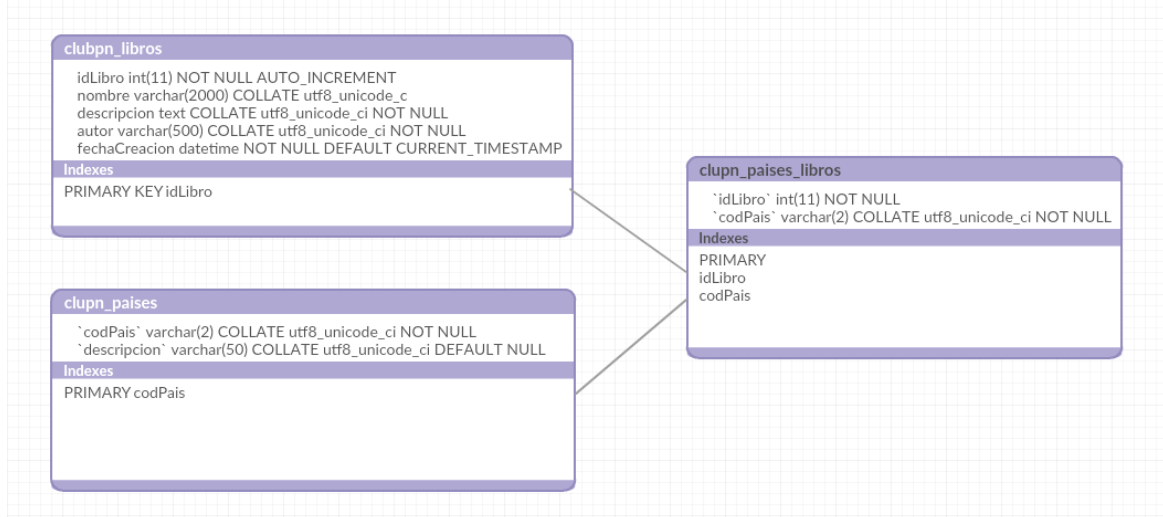


Ilustración 8. Modelo entidad relación con_lectura

5.3.4.1 Descripción de las tablas

A continuación se describen las tablas creadas en el desarrollo del componente para la gestión de los libros leídos en el club de lectura.

CLUBPN_LIBROS	Tabla que contiene los libros dados de alta en el sistema.
idLibro	Clave primaria. Identifica los libros de forma única en el sistema, es un valor auto numérico.
nombre	Nombre del libro
descripcion	Descripción del libro
autor	Autor del libro
fechaCreacion	Fecha en la que se da de alta el libro en el sistema.
Esta tabla se relaciona con la tabla clubpn_paises_libros con una relación 1-N	

Tabla 7. Tabla CLUBPN_LIBROS

CLUBPN_PAISES	Tabla que contiene el catálogo de países
codPais	Clave primaria. Identifica los países de forma única en el sistema, es un código de 5 dígitos alfanuméricos.
descripcion	Nombre del país
Esta tabla se relaciona con la tabla clubpn_paises_libros con una relación 1-N	

Tabla 8. Tabla CLUBPN_PAISES

1 de julio de 2016

CLUBPN_PAISES_LIBROS	Tabla que relaciona los libros con los países que aparecen en la historia
idLibro	Clave primaria. Identifica los libros de forma única en el sistema, es un valor auto numérico.
codPais	Clave primaria. Identifica los países de forma única en el sistema, es un código de 5 dígitos alfanuméricos.

Tabla 9 . Tabla CLUBPN_PAISES_LIBROS

5.3.5 Diseño de la interfaz

La imagen general de la página viene marcada por la plantilla seleccionada, Joomla Template Rooms, la cual se ha cogido como base para la estructura de la página a pesar que se ha modificado para dar el aspecto final. Los cambios realizados en la plantilla, con afectación global, son:

- El cambio del fondo
- El cambio de los tipos de letra utilizados en la pagina
- El cambio del logotipo por el logotipo del club de lectura de la UPNA.

La plantilla utilizada esta está basada a su vez en el framework WARP7, WARP es un Framework desarrollado por Yootheme que tiene como objetivo facilitar la creación de temas para los sistemas gestores de contenido más populares. WARP en su versión 7 incorpora un Framework de desarrollo llamado UIKIT. Este framework está desarrollado usando las mejores prácticas con el objetivo de hacer temas que funcionen correctamente en todos los navegadores, incluso los de móvil, que sean fáciles de configurar y que estén desacoplados respecto al framework del Sistema Gestor de contenidos.

A continuación se analiza la distribución del contenido en la pantalla, tanto para la visualización en ordenador como en móvil y tableta.

Las áreas de contenido utilizadas en la web para la visualización desde un ordenador o una tableta puesta en horizontal son las siguientes:

1 de julio de 2016

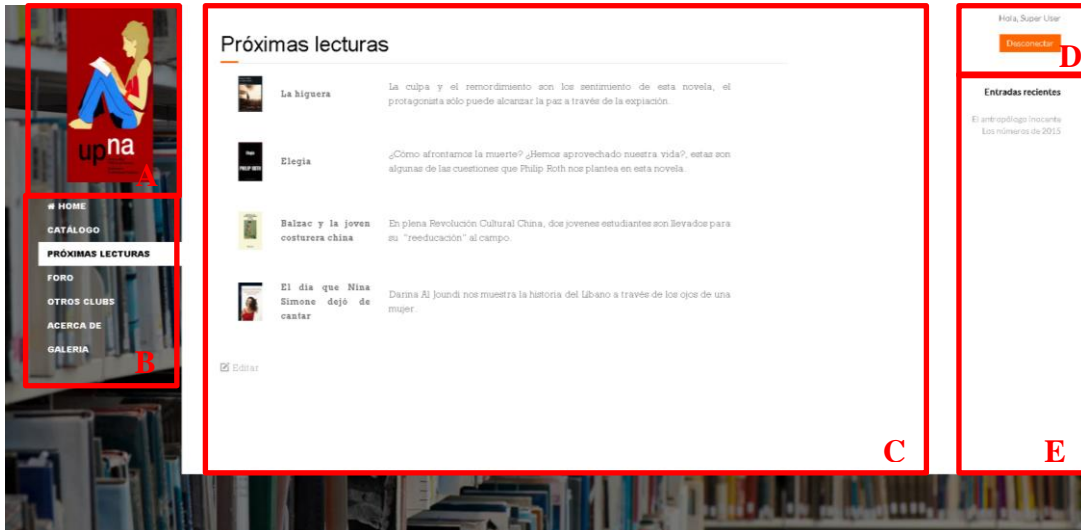


Ilustración 9 . Distribución áreas de contenido. PC y Tableta horizontal.

A continuación se muestra la estructura visual correspondiente para el caso en que la página se visualizase desde un móvil o una tableta en vertical. La página al ser de diseño adaptable se adapta para poder ser visualizada en estos dispositivos, moviendo la ubicación de las áreas de contenido en una disposición más vertical.

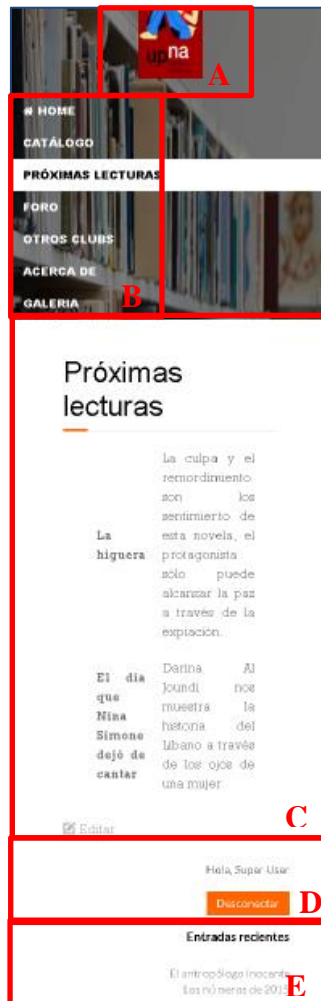


Ilustración 10 Distribución áreas de contenido. Movil y tableta vertical

1 de julio de 2016

- A. **Logo.** Esta región contendrá el logo del club de lectura, este bloque aparece en todas las páginas.
- B. **Menú.** Esta región contendrá el menú de la página, configurado desde el panel de control de Joomla. Este bloque aparece en todas las páginas.
- C. **Contenedor.** En este bloque es donde se va cargando el contenido correspondiente a la selección en el menú.

En caso de seleccionar la opción de menú “Catalogo” se cargará la parte del componente *com_lectura*, componente desarrollado en el marco de este proyecto, correspondiente a la parte del Sitio.

A continuación se muestra el aspecto del bloque contenedor cuando se ha seleccionado la opción “Catalogo” y se ha cargado esta opción de menú.

ID	TÍTULO	AUTOR
447	Título 1	Autor 1
448	Título 2	Autor 2
449	Título 3	Autor 3
493	Título 4	Autor 4

Ilustración 11. Interfaz componente *com_lectura*

La región del contenedor se divide en 3 zonas:

- **Título:** que contiene el literal “Lectura”.
- **Mapa interactivo:** Contiene el mapa del mundo interactivo que permite, clicando en los países mostrar los libros que cuentan una historia, o parte de ella, ocurrida en ese país.

Para la implementación del mapa se utiliza JQVMap, que es un Plugin de la librería Javascript jQuery que renderiza mapas de vectores. JQVMap se apoya

1 de julio de 2016

en Scalable Vector Graphics (SVG) para su implementación, lo que hace este plugin compatible con la mayoría de navegadores web.

- **Resultados de la búsqueda.** En este bloque en la carga inicial de la pantalla se cargan todos los libros leídos en el club de lectura, independientemente del país donde ocurren. Al clicar, sobre un país se carga la lista de libros en los que ese país aparece en la trama.

Por cada uno de los libros se muestra:

- El identificador interno del libro.
- El título del libro
- El autor del libro.

Al clicar sobre uno de los registros del listado, aparecerá una ventana emergente con más información del libro en cuestión, en concreto se muestran los campos:

- El identificador interno del libro.
- El título del libro
- El autor del libro.
- Descripción del libro.

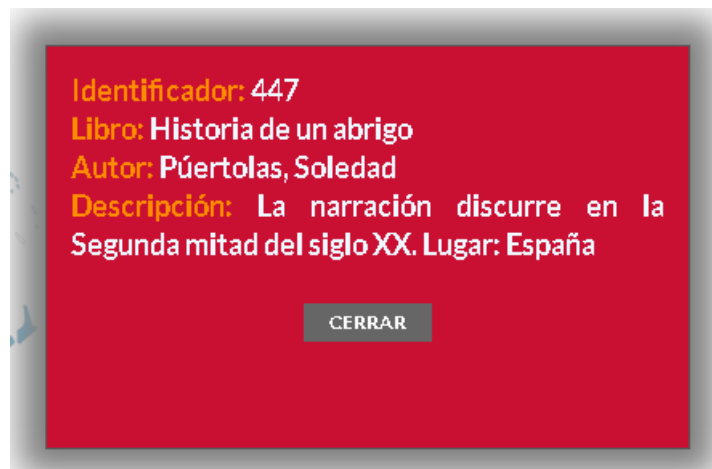


Ilustración 12 . Detalle del libro seleccionado

Pulsando en el botón cerrar se cerrará la ventana emergente.

D. Login/Logout. Este bloque corresponde al Plugin de gestión de usuarios por defecto de Joomla. El contenido de este bloque varía según se haya validado un usuario en la página o no. Aparece en todas las páginas.

- **Sin usuario**
 - Aparecen los cuadros de texto para introducir el usuario y la contraseña.
 - Aparece el botón "Identificarse", que envía el formulario con el usuario y la contraseña.

1 de julio de 2016

- Aparece el enlace “¿Recordar contraseña?”, que carga el formulario para recuperar la contraseña.
 - Aparece el enlace “¿Recordar usuario?” , que carga el formulario para recuperar el usuario.
 - **Usuario validado**
 - Aparecerá el saludo al usuario validado.
 - Aparecerá el botón “Desconectar” que cerrará la sesión del usuario validado. Una vez desconectado, en este bloque se cargará listado de componentes indicados en “Sin usuario”.
- E. Entradas recientes:** Contiene un listado de los artículos que tienen indicada la categoría “Artículo”, si se clica encima de un elemento de la lista se cargará. Aparece en todas las páginas.

6 Fase de Implantación

6.1 Arquitectura de Joomla

Joomla es un gestor de contenidos modular, esta característica le permite crecer en funcionalidad para poder dar soporte a soluciones muy diferentes. Joomla es de código abierto y se distribuye con licencia GPL.

Joomla está desarrollado utilizando los lenguajes PHP, HTML y Javascript, además requiere para su funcionamiento una base de datos creada con un gestor de bases de datos (MySQL es lo más habitual), así como de un servidor HTTP Apache.

Joomla es un Sistema Gestor de contenido modular que permite ser extendido con facilidad, a continuación se muestra una ilustración en la que se pueden ver las diferentes capas y como se comunican entre ellas.

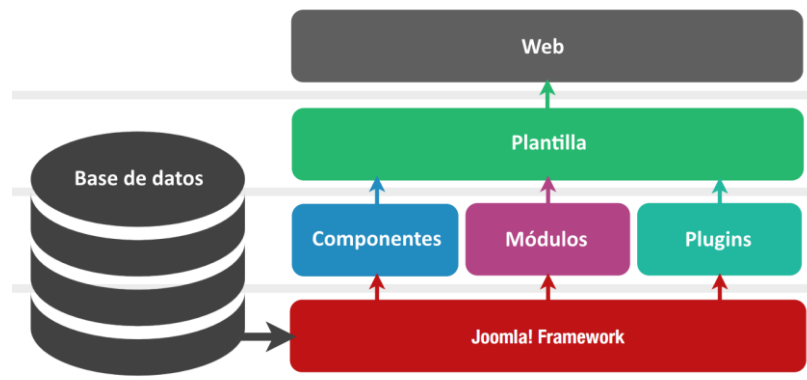


Ilustración 13. Elementos que componen la arquitectura de Joomla!

- **Plantilla:** Permite cambiar la imagen de Joomla.
- **Componentes:** Extensiones que permiten ampliar la funcionalidad original de Joomla.
- **Módulos:** Permiten ubicar los componentes en la pantalla, referenciando las zonas correspondientes de la plantilla.
- **Plugins:** Código que realiza acciones específicas del sistema, son más sencillos que los componentes y realizan acciones concretas.
- **Joomla! Framework:** Es el núcleo del sistema gestor y provee los servicios que utilizan las capas superiores para la comunicación entre módulos y las tareas de más bajo nivel, p.e. la gestión de la comunicación con la base de datos.

A continuación se describen los conceptos básicos a conocer a la hora de trabajar con el gestor de contenidos Joomla y que permiten entender cómo funciona y como se puede personalizar para adaptarlos a la necesidad a cubrir.

6.1.1 Organización de contenidos: categoría y artículo

La estructura del contenido en un sitio web Joomla está organizada en un orden jerárquico en base a categorías y artículos de contenido. Esta organización permite independizar la organización del contenido de cómo se esté presentando en el sitio, la presentación se gestiona mediante el sistema de menús del que dispone el gestor.

Las categorías permiten agrupar artículos que tratan de temas relacionados, permitiendo trabajar con ellos después como grupos de artículos. Una categoría puede tener uno o más artículos de contenido, mientras que los artículos sólo pueden incluirse en una categoría (característica que difiere de otros sistemas gestores de contenido).

El artículo es la unidad básica de contenido, cualquier tipo de contenido que se añada al sistema se encuentra forzosamente en un artículo. Los artículos pueden ser:

- Sin clasificar, es decir, no se incluyen en la jerarquía de categorías.
- Clasificados, pertenecen a una categoría.

6.1.2 Presentación de contenidos: menús

Los menús permiten definir la navegación por todo el portal web y permiten acceder a los contenidos publicados. Cada elemento del menú se gestiona independientemente y pueden enlazarse con categorías, artículos sin clasificar, etc.

El gestor de contenidos permite crear uno o varios menús con distintos niveles de menú, anidando ítems de menú. Cada menú se deberá posicionar en alguna de las regiones de contenido definidas dentro de la plantilla.

6.1.3 Apariencia del portal web: plantillas

Los gestores de contenido permiten una separación del contenido y del diseño. Esto permite modificar el diseño del sitio sin tener que modificar los contenidos ni su organización.

Las plantillas son el conjunto de archivos destinados a configurar la apariencia del portal web. El contenido del portal se estructura mediante uno o varios ficheros PHP que contienen el código que conforma la distribución de los elementos en pantalla. Para colocar los módulos definidos en el gestor en el lugar donde se deberán ver, se añadirán estos referenciando las etiquetas PHP marcadas en la estructura.

La plantilla utiliza hojas de estilo en cascada (CSS) para aplicar estilo. La instalación básica de Joomla incorpora las plantillas por defecto, pudiendo instalar otras o crear nuevas.

1 de julio de 2016

6.1.4 Extensiones: componentes, módulos y plugins

Las extensiones son un conjunto de componentes que aportan al gestor de contenidos funcionalidades extra. Estos pueden estar desarrollados por la comunidad de usuarios o por empresas y están escritos en lenguaje PHP. La comunidad Joomla dispone de un buscador de extensiones donde se pueden encontrar la mayoría de ellas: <http://extensions.joomla.org>

Los tipos de extensiones existentes en Joomla son plantillas, componentes, módulos y plugins.



Ilustración 14. Plantillas. Especificación de la estructura de la plantilla

- Las **plantillas** permiten configurar la presentación del sitio web. Se explican en detalle en *Apariencia del portal web: plantillas*.
- Los **componentes** son las extensiones más grandes y complejas de todas; que proporcionan funcionalidades complejas. La mayoría de ellos tienen dos partes: una para la parte pública y otra para la parte administrativa. Por defecto, el sistema gestor de contenidos contiene componentes con las funcionalidades típicas de un portal web: banners, buscadores, gestión de usuarios, noticias externas y enlaces web.
- Los **módulos** permiten añadir componentes en distintas posiciones de la pantalla, asociándolos a los anclajes de la plantilla. Los menús, encargados de facilitar la navegabilidad por el sitio, son los módulos más usados (*mod_mainmenu*).
- Los **plugins** son las extensiones más avanzadas, y son en esencia controladores de eventos. En la ejecución de cualquier parte de Joomla, ya sea en el núcleo, un módulo o un componente, un evento puede ser disparado. Cuando esto ocurre, los plugins que están registrados con la aplicación para controlar ese evento se

1 de julio de 2016

ejecutan. Un ejemplo muy común de plugins son los editores WYSIWYG, que facilitan la generación de artículos.

6.2 Proceso de Implantación

El proceso seguido para la implantación de la web <http://clubdelecturaupna.es/> ha sido el siguiente:

1. Instalación de Joomla
2. Configuración de Joomla
3. Instalación y configuración de la plantilla
4. Instalación de extensiones: módulos, componentes y plugins
5. Creación de las Categorías y artículos
6. Creación del menú
7. Gestión de imágenes en la galería.

6.2.1 Instalación de Joomla

Para la instalación de Joomla se hará uso del instalador que lo acompaña, a continuación se indica el proceso a realizar, paso a paso, para la instalación base del gestor de contenido. El instalable del Sistema Gestor de Contenido Joomla se puede descargar de la web oficial de Joomla, <https://www.joomla.org/3/es/>, la versión utilizada del gestor de contenidos en este proyecto es Joomla 3.4.8.

A posteriori a la instalación se actualizó la versión de Joomla a la versión 3.5.1, esta versión respecto a la versión 3.4.8 incorpora las siguientes mejoras:

- Incorpora soporte para PHP 7, aunque en este proyecto se utilice todavía PHP5
- Permite recibir las notificaciones de actualizaciones por correo electrónico.
- Permite exportar información de sistema y de entorno.
- Permite utilizar la tecnología de arrastrar y soltar imágenes (drag&drop).
- Los datos del sistema desde la versión 3.5 están anonimizados.
- Facilita la tarea de insertar módulos en artículos.

Pasos de la instalación

1. Se ha de descomprimir el paquete Joomla 3.4.8., por ejemplo en la carpeta Joomla_3.4.8-Stable-Full_Package.
2. Se deberá subir el contenido de la carpeta Joomla_3.4.8-Stable-Full_Package de Joomla mediante un cliente FTP (Ref.: *Herramientas utilizadas*) al servidor web donde se quiere desplegar, concretamente a la carpeta *public_html* del servidor web, esta carpeta contiene los ficheros que el servidor web publicara a Internet. La ubicación de esta carpeta puede variar según la configuración del servidor, en nuestro caso */public_html*.

1 de julio de 2016

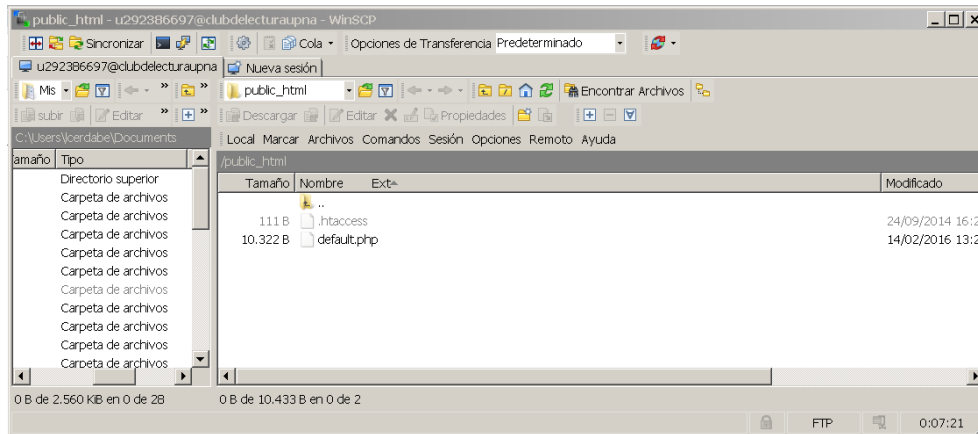


Ilustración 15. Cliente FTP y carpeta raíz del servidor web

Para subir los ficheros si se utiliza la herramienta WinSCP, u otros clientes FTP similares, se puede arrastrar el contenido a la pantalla de la aplicación con la conexión al FTP abierta y situados en la carpeta donde se quiere subir el contenido, al soltar el contenido se iniciará la subida de ficheros.

3. A continuación se ha de crear la base de datos usada por Joomla para almacenar los datos. En el caso de este proyecto el alojamiento web utilizado ya provee el servicio para ubicar la base de datos. Por tanto, se deberá abrir en una nueva pestaña del navegador el panel de control del espacio web, concretamente se deberá ir a la parte de Bases de datos del Panel de Control del alojamiento web, y crear desde allí la base de datos.

En el caso del panel de control del proveedor del espacio la opción es Bases de datos > Bases de datos MySQL.

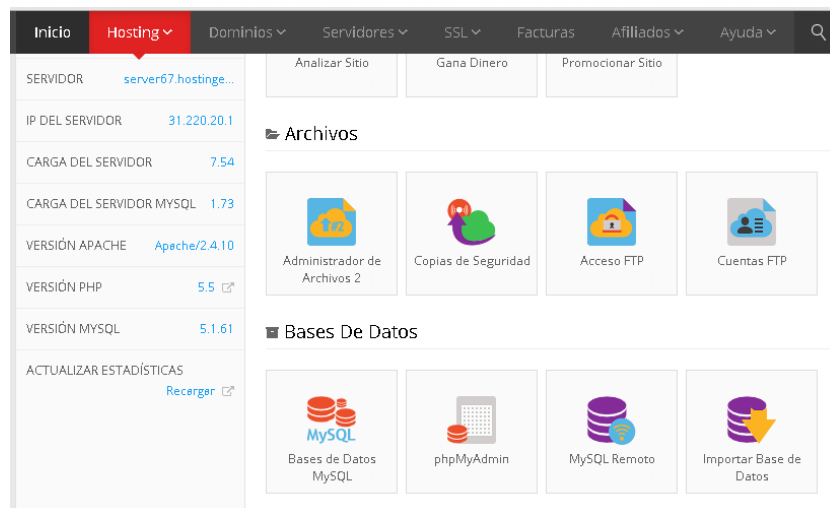


Ilustración 16 . Instalación de Joomla. Panel de control del espacio web

En la pantalla Bases de Datos MySQL se deberán rellenar los datos “Nombre de base de datos MySQL”, “Usuario MySQL”, “Contraseña” y “Contraseña de

1 de julio de 2016

nuevo” y hacer clic en el botón “Crear”. Con la creación de la base de datos se creará el usuario indicado, con perfil Administrador y acceso a ella.

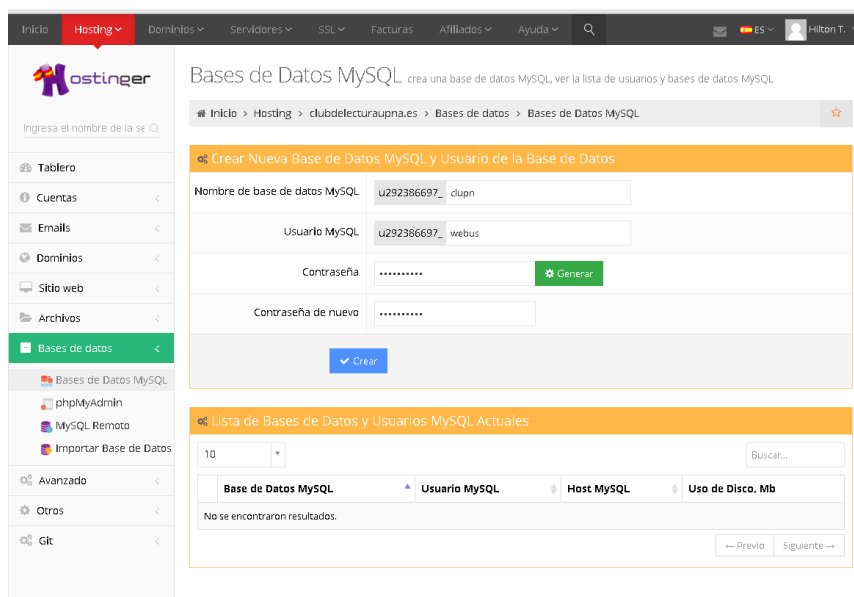


Ilustración 17. Instalación de Joomla. Bases de datos MySQL

Si se ha realizado la creación de la base de datos correctamente aparecerá en la Lista de Bases de Datos y Usuarios MySQL Actuales.

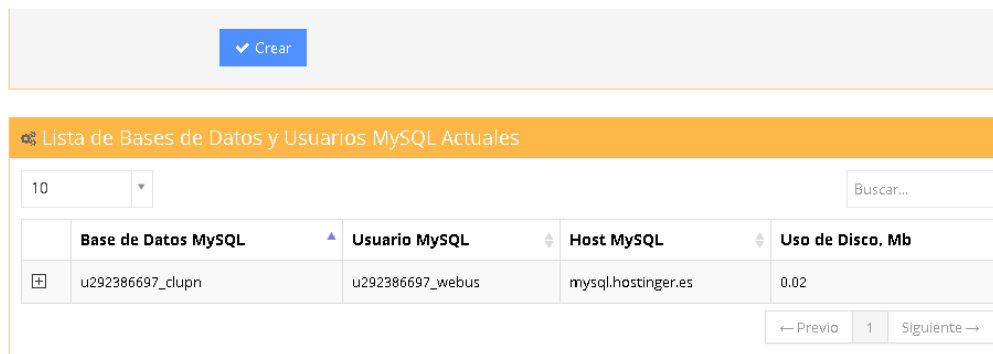


Ilustración 18. Instalación de Joomla. Lista de bases de datos activas

4. A continuación para proseguir con la instalación se deberá ir a la URL del sitio web, asociado al FTP donde se han subido los ficheros de Joomla, en el caso de este proyecto la URL correspondiente es <http://clubdelecturaupna.es/>. Una vez se ha accedido a la URL aparecerá la pantalla de instalación de Joomla donde se deberá rellenar la información principal del sitio, una vez rellenados los campos se deberá pulsar en “Siguiente”:

- **Nombre del sitio:** Nombre que se le quiere dar al sitio web.
- **Descripción:** Descripción del sitio web, usado por los motores de búsqueda en el proceso de indexación.

1 de julio de 2016

- **El correo electrónico del administrador:** Correo electrónico de la persona o equipo que gestiona el sitio web.
- Nombre de usuario del administrador, Contraseña del administrador y Confirmar la contraseña del administrador: Datos para la creación del usuario Administrador del gestor de contenidos.
- **Sitio fuera de línea:** Permite poner el gestor de contenidos en modo fuera de línea, lo que hará que al entrar al portal este muestre un mensaje indicándolo. Si marcamos “Sí”, a posteriori se deberá desde la parte de Administración, volver a ponerlo en línea.

The screenshot shows the Joomla! installation configuration interface. At the top, there are three tabs: '1 Configuración', '2 Base de datos', and '3 Visión general'. Below the tabs, there is a language selection dropdown set to 'Spanish (Español)' and a blue 'Siguiente' button. The main section is titled 'Configuración principal' and contains several form fields:

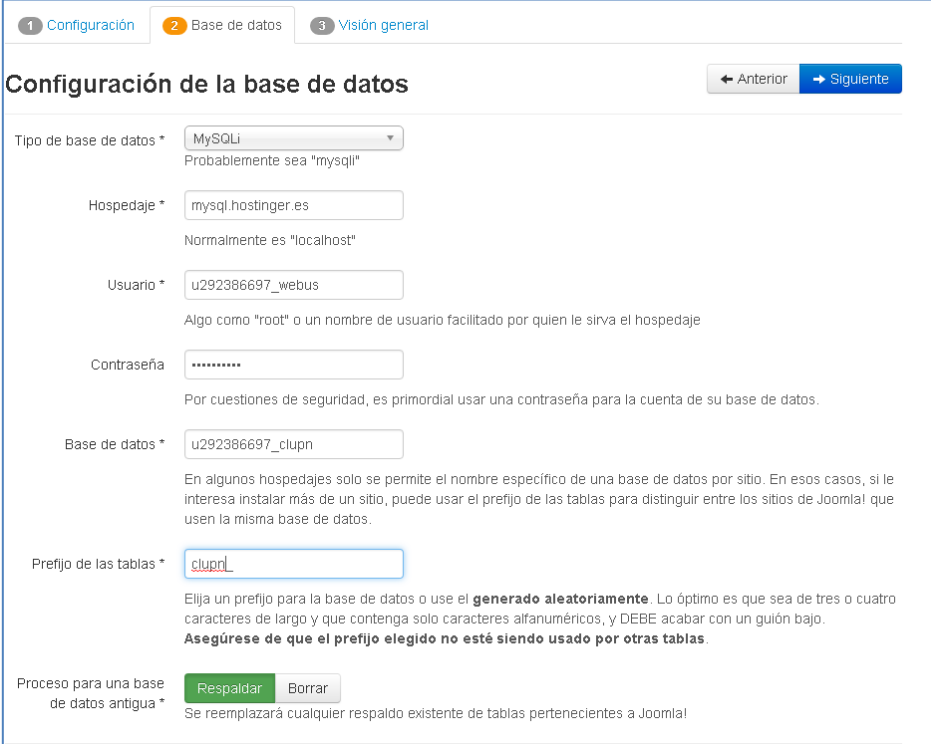
- Nombre del sitio ***: A text input field with the instruction 'Introduzca el nombre de su sitio Joomla!'. Below it is a 'Descripción' text area with the instruction 'Introduzca la descripción general de todo el sitio, la cual será usada por los motores de búsqueda. Generalmente, un máximo de 20 palabras suele ser lo óptimo.'
- El correo electrónico del administrador ***: A text input field with the instruction 'Introduzca una dirección de correo electrónico. Debe ser la dirección de correo electrónico del súper administrador del sitio.'
- Nombre de usuario del administrador ***: A text input field with the instruction 'Asigna el nombre de usuario para su cuenta de súper administrador.'
- Contraseña del administrador ***: A text input field with the instruction 'Asigne la contraseña de la cuenta del súper administrador y confírmela en el campo de más abajo.'
- Confirmar la contraseña del administrador ***: A text input field.
- Sitio fuera de línea**: Radio buttons for 'Sí' and 'No'. Below them is the instruction: 'Poner fuera de línea el acceso a la zona pública del sitio cuando se complete la instalación. Si ahora no es necesario, recuerde que siempre que lo desee podrá poner el sitio fuera de línea desde la configuración global.'

Ilustración 19. Instalación de Joomla. Configuración principal

5. A continuación se deberá configurar la base de datos, creada previamente desde el Panel de Control del alojamiento web, en el gestor de contenidos. Se deberán rellenar los campos indicados y pulsar en “Siguiente”.
 - **Tipo de base de datos:** Tipo de la base de datos que queremos configurar, en el caso de este proyecto MySQLi.
 - **Hospedaje:** URL de acceso a la base de datos, en el caso de este proyecto la marca el proveedor del servicio web, el mismo que provee la base de datos.
 - **Usuario y Contraseña:** Datos del usuario de base de datos, en el caso de este proyecto el creado anteriormente. Ha de tener permisos de lectura/escritura/Creación de tablas/Modificación de tablas.
 - **Base de datos:** Nombre de la base de datos a configurar, en el caso de este proyecto la creada anteriormente.
 - **Prefijo de las tablas:** Prefijo a añadir en los nombre de las tablas de Joomla que permitirá identificarlas al revisar el modelo de datos en el Sistema Gestor de Base de Datos.

1 de julio de 2016

- **Proceso para una base de datos antigua:** Permite indicar si hacer copia de seguridad (Respaldar) o borrar la base de datos existente en el sistema con el mismo nombre que la que se está configurando. En el caso de este proyecto no aplica al ser una instalación nueva.



The screenshot shows the Joomla! installation configuration page for the database. It is titled "Configuración de la base de datos" and has three tabs: "1 Configuración", "2 Base de datos", and "3 Visión general". The "Base de datos" tab is active. The form includes the following fields and options:

- Tipo de base de datos:** A dropdown menu set to "MySQLi". Below it, a note says "Probablemente sea 'mysql'".
- Hospedaje:** A text input field containing "mysql.hostinger.es". Below it, a note says "Normalmente es 'localhost'".
- Usuario:** A text input field containing "u292386697_webus". Below it, a note says "Algo como 'root' o un nombre de usuario facilitado por quien le sirva el hospedaje".
- Contraseña:** A password input field with masked characters. Below it, a note says "Por cuestiones de seguridad, es primordial usar una contraseña para la cuenta de su base de datos."
- Base de datos:** A text input field containing "u292386697_clupn". Below it, a note says "En algunos hospedajes solo se permite el nombre específico de una base de datos por sitio. En esos casos, si le interesa instalar más de un sitio, puede usar el prefijo de las tablas para distinguir entre los sitios de Joomla! que usen la misma base de datos."
- Prefijo de las tablas:** A text input field containing "clupn_". Below it, a note says "Elija un prefijo para la base de datos o use el **generado aleatoriamente**. Lo óptimo es que sea de tres o cuatro caracteres de largo y que contenga solo caracteres alfanuméricos, y DEBE acabar con un guión bajo. **Asegúrese de que el prefijo elegido no esté siendo usado por otras tablas.**"
- Proceso para una base de datos antigua:** Two buttons: "Respaldar" (highlighted in green) and "Borrar". Below them, a note says "Se reemplazará cualquier respaldo existente de tablas pertenecientes a Joomla!"

Navigation buttons "Anterior" and "Siguiente" are located at the top right of the form.

Ilustración 20. Instalación de Joomla. Configuración de la base de datos

6. Una vez configurada la base de datos el proceso de instalación preguntará si se quiere cargar alguno de los juegos de datos de ejemplo, Si se quieren enviar las credenciales de Administrador del gestor de contenidos por correo y mostrará un resumen de los datos informados durante la instalación para configurar el sitio. En este proyecto no se cargaran datos de prueba ni se enviará la configuración por correo. Al clicar en "Instalar" se iniciará la instalación del gestor de contenidos en el servidor.

1 de julio de 2016

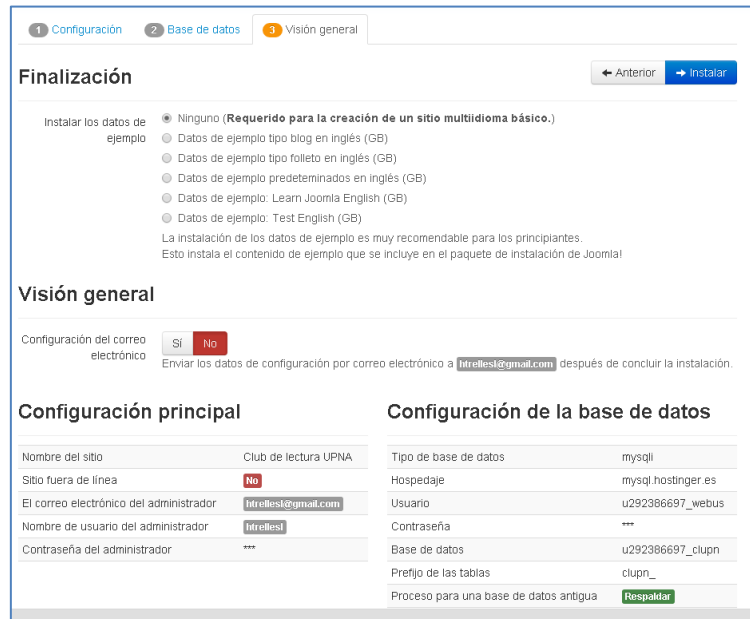


Ilustración 21. Instalación de Joomla. Finalización de la instalación

- Al finalizar el proceso, se mostrará un mensaje de confirmación y se deberá eliminar la carpeta de instalación.



Ilustración 22. Instalación de Joomla. Confirmación de la instalación

Una vez instalado el gestor de contenido se podrá acceder a cada una de las partes del gestor mediante las siguientes URL:

- Parte de Administración: <http://clubdelecturaupna.es/administrator/>
- Parte del Sitio: <http://clubdelecturaupna.es>

6.2.2 Configuración del portal

A continuación se detallan las configuraciones a realizar post-instalación antes de comenzar a dotar a la página de contenido.

1 de julio de 2016

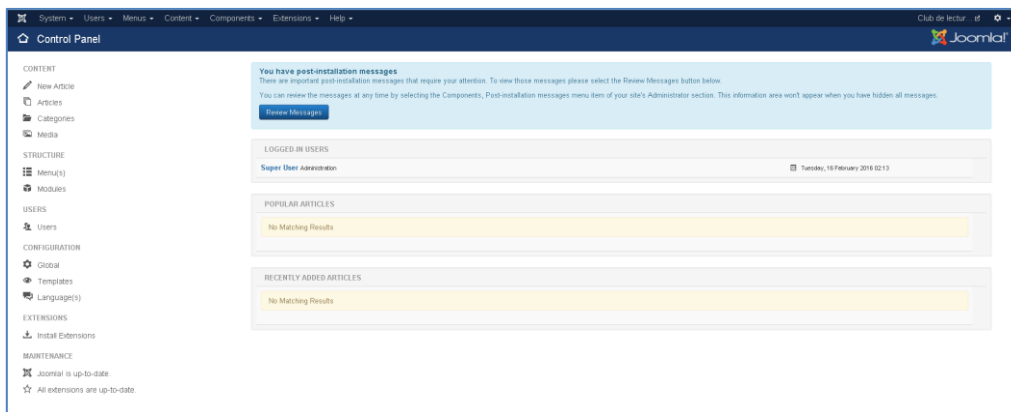


Ilustración 23. Configuración de Joomla. Panel de control

Todas las configuraciones se realizarán desde el Panel de control,

6.2.2.1 Cookies

En Sistema>Panel de Control > Sitio > Configuración Global se deberá indicar en “Configuración de las 'cookies'” el dominio de la cookie con el valor “clubdelecturaupna.es”.

6.2.2.2 Caché

En Sistema>Panel de Control > Sistema > Configuración de la caché se deberá activar la Caché con la opción Activado Caché progresiva.

6.2.2.3 Correo electrónico

En Sistema>Panel de Control > Servidor > Configuración del correo electrónico se revisará la configuración de correo electrónico para dejarla tal y como ilustra la siguiente imagen.

 The image shows the Joomla! configuration page for email settings. The title is 'Configuración del correo electrónico'. It contains several settings: 'Enviar correo' with a green 'Sí' button and a grey 'No' button; 'Gestor de correo *' with a dropdown menu set to 'PHP Mail'; 'Correo electrónico del sitio' with a text input field containing 'htrellesl@gmail.com'; 'Remitente' with a text input field containing 'Club de lectura UPNA'; and 'Deshabilitar el correo masivo' with a grey 'Sí' button and a red 'No' button.

Ilustración 24 . Configuración Joomla. Configuración del correo

A nivel de perfiles de usuario se utilizarán los que provee Joomla por defecto.

1 de julio de 2016

6.2.2.4 SEO

Se realizarán las acciones para mejorar el SEO indicadas en los puntos 8.2.1 *Introducir Meta-datos* y 8.2.2 *URL's amigables*.

6.2.3 Instalación y configuración de la plantilla

Para instalar una nueva plantilla que aplique un estilo diferente al gestor de contenido se deberá de descargar el fichero comprimido de instalación de la web y seguir el procedimiento descrito en el apartado: 6.2.4 *Instalación de extensiones: módulos, componentes y plugins*.

Una vez instalado se podrá configurar, tanto el estilo del Sitio como de la Administración, desde el panel de control en el menú Extensiones > Plantillas. Para este proyecto se ha instalado el estilo `jp_rooms` de JoomlaPlates para el Sitio y se ha mantenido el estilo por defecto para la parte de Administración. Para ello se marcaron como predeterminados en la pantalla de gestión de plantillas.

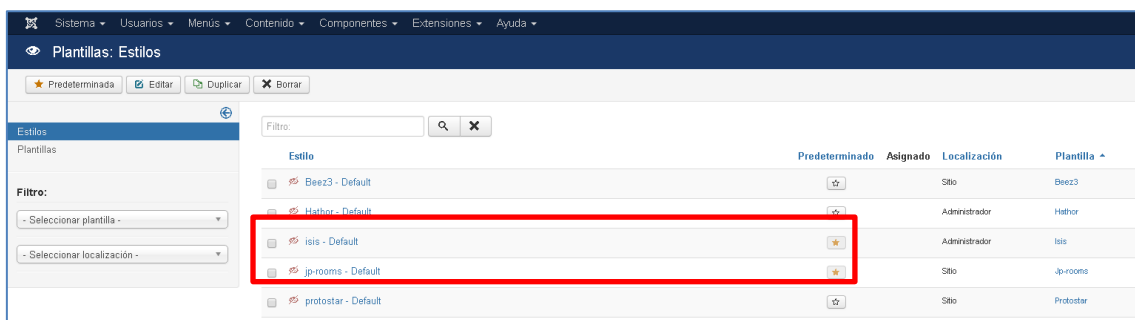


Ilustración 25 .Instalación estilos. Gestión de estilos. Estilos por defecto

Para acceder a la configuración del estilo se deberá hacer clic encima de su nombre en el listado de estilos. La configuración de la plantilla de un estilo depende del creador del mismo que es quien marca que elementos se pueden modificar mediante configuración y cuáles no. Si se desea hacer modificaciones no contempladas dentro de las opciones de configuración se debe modificar el código de la plantilla o extenderlo mediante estilos CSS.

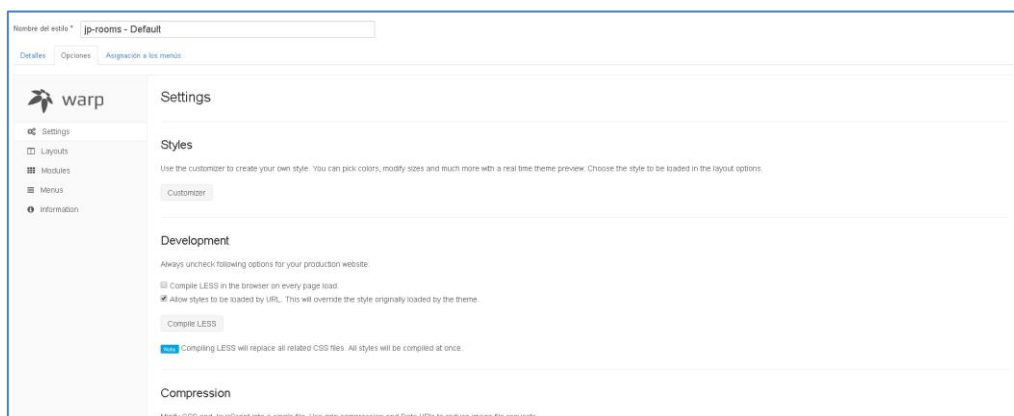


Ilustración 26. Instalación estilos. Gestión de estilos. Configuración de estilos

1 de julio de 2016

Desde la configuración se puede gestionar también donde se aplican los estilos en el Sitio, desde el menú Asignación a los menús, en el caso de este proyecto el estilo instalado aplica a todos los menús del sitio.

Para acceder al editor de los ficheros que componen el estilo se deberá hacer clic en el nombre del tema en la columna plantillas y se cargará la jerarquía de carpetas y ficheros que componen la plantilla. En este proyecto se han añadido o modificado las reglas que definen los estilos del portal, utilizando CSS3, añadiéndolos al fichero `css/custom.css`, de esta manera protegemos los cambios realizados ante actualizaciones del tema, en caso contrario se perderían al sobrescribirse los ficheros modificados.

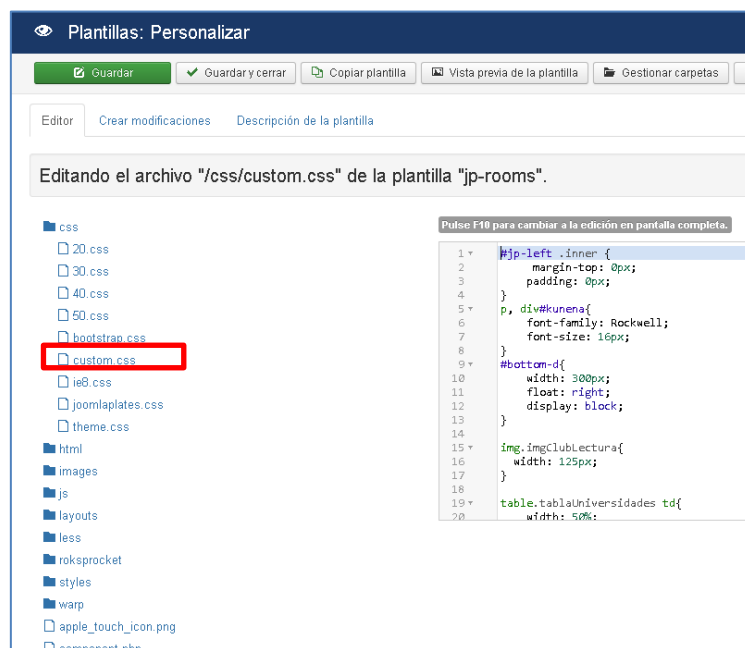


Ilustración 27. Instalación estilos. Gestión de estilos. Personalización de estilos

Finalmente, mediante el servicio de validación de hojas de estilo del W3C (<http://jigsaw.w3.org/css-validator/validator>) se ha validado que la hoja de estilo desarrollada en el marco de este proyecto, `custom.css`, cumple con los estándares.

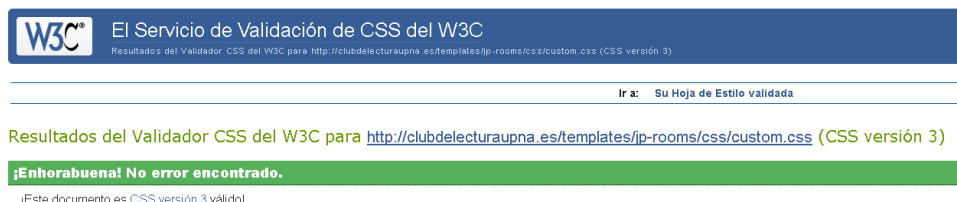


Ilustración 28 .Instalación estilos. Gestión de estilos. Validación W3C

6.2.4 Instalación de extensiones: módulos, componentes y plugins

El proceso de instalación de extensiones en Joomla es común tanto si se trata de una extensión desarrollada a medida o una extensión de terceros publicada en <http://extensions.joomla.org/>, en el caso de ser una extensión de terceros se recomienda leer la documentación asociada con la extensión. La mayoría de las extensiones tienen páginas web y foros y es una buena idea para buscar en ellos primero. Si hay un archivo README incluido con la extensión, se debería leer.

En Joomla hay dos procedimientos para instalar una extensión:

- **Mediante el Gestor de extensiones (Recomendado):** Para realizar la instalación de un componente mediante el Gestor de extensiones se han de seguir los siguientes pasos:
 1. Descargar la extensión en la máquina local como un archivo comprimido zip. A veces es necesario descomprimir el archivo de forma local antes de la instalación, suele indicarse en el nombre del fichero y las notas de la extensión.
 2. Desde el Lado Servidor del sitio Joomla (administración), seleccionar Extensiones → Gestionar.
 3. Clicar en el botón Examinar y seleccionar el paquete de extensión en el equipo local.
 4. Hacer clic en el botón Subir e instalar. Si obtiene un error diciendo que el archivo no está en el formato correcto, revisar si es necesario descomprimirlo e instalar los elementos individualmente, que deberían ser también ficheros comprimidos.

Consideraciones

- Algunas extensiones pueden ofrecer más instrucciones sobre la instalación, el proceso aquí indicado es el estándar.
- Se ha de tener en cuenta que los módulos y plugins deben estar habilitados antes de usarlos, en muchos casos estos se instalan deshabilitados.
- A veces no se puede utilizar el instalador automático. Por ejemplo, extensiones muy grandes que pueden exceder el tamaño máximo de carga permitido por el alojamiento web, en estos casos se seguiría el procedimiento de instalación manual.
- Puede existir una restricción del alojamiento web que hace que Joomla! trate de comprobar si el directorio raíz existe. En estos casos se debería realizar la instalación manual. En estos casos aparecería el siguiente mensaje de error:

Warning: is_dir() [function.is-dir]: open_basedir restriction in effect. File(/) is not within the allowed path(s): ...

1 de julio de 2016

- **Instalación manual:** Para realizar la instalación de un componente de forma manual se han de seguir los siguientes pasos:
 1. Primero, se han de descomprimir todos los archivos en un directorio local (por ejemplo *com_lectura*).
 2. Se han de transferir de directorio (usando FTP) a una carpeta en el directorio de instalación de Joomla (por ejemplo *administrator/components*), adecuado para el tipo de extensión que se está instalando (se puede consultar en el archivo XML, una línea como `<extension type="component" ...>`).
 3. A continuación, Desde el Lado Servidor del sitio Joomla (administración), seleccionar Extensiones → Gestionar, pero se debe seleccionar "Instalar desde directorio", indicando el nombre de la carpeta donde se han descomprimido los ficheros. El nombre de esta carpeta debe ser una ruta absoluta a partir de la raíz del sistema de ficheros.

6.2.4.1 Extensiones necesarias: Publicadas y Despublicadas

6.2.4.1.1 Extensiones necesarias en el portal web (publicadas)

A continuación se muestran, separados por tipología, las extensiones publicadas en Joomla.

Módulos

Módulo	Posición	Tipo	Páginas	Acceso	Idioma	Descripción
Logo	logo	Personalizado	Todas	Publico	Todos	Módulo que muestra el logo de la UPNA.
Main Menu	menu	Menú	Todas	Publico	Todos	Módulo que contiene el menú principal del sitio web.
Breadcrumbs	position-2	Ruta de navegación	Todas	Publico	Todos	Módulo que contiene la ruta has, el artículo que se está visualizando.
Login Form	sidebar-b	Datos de acceso	Todas	Publico	Todos	Módulo que contiene el formulario de acceso a la parte privada de la web.

1 de julio de 2016

Entradas recientes	sidebar-b	Artículos de la categoría	Todas	Publico	Todos	Módulo que lista todos aquellos artículos de la categoría "Artículo"
--------------------	-----------	---------------------------	-------	---------	-------	--

Tabla 10 . Modulos cargados en el proyecto

Plugins

A continuación se listan los plugins activos en el gestor de contenidos Joomla, resaltando en gris aquellos que forman parte de las extensiones que se han instalado en el marco del proyecto. La misión de cada una de las extensiones se explican en los apartados donde se referencian.

Public	Nombre del Plugin	Elementos
cookie	Autenticación – Cookie	authentication
joomla	Autenticación – Joomla	authentication
recaptcha	Captcha – ReCaptcha	captcha
contact	Contenido – Contactar	content
emailcloak	Contenido - Enmascarar correo	content
joomla	Contenido - Joomla	content
loadmodule	Contenido - Cargar módulos	content
pagebreak	Contenido - Salto de página	content
pagenavigation	Contenido - Navegación entre páginas	content
sige	Content - Simple Image Gallery Extended	content
vote	Contenido - Votar	content
codemirror	Editor - CodeMirror	editors
none	Editor - Sin editor	editors
tinymce	Editor - TinyMCE	editors
article	Botón - Artículo	editors-xtd
image	Botón - Imagen	editors-xtd
pagebreak	Botón - Salto de página	editors-xtd
readmore	Botón - Leer más	editors-xtd
joomla	Extensión - Joomla	extension
categories	Búsquedas inteligentes - Categorías	finder
contacts	Búsquedas inteligentes - Contactos	finder
content	Búsquedas inteligentes - Contenido	finder
newsfeeds	Búsquedas inteligentes - Canales electrónicos	finder
tags	Búsqueda inteligente - Etiquetas	finder
joomla	Kunena - Integración Joomla	kunena
kunena	Kunena - Integración Kunena	kunena
com_content	OSMap - Content Plugin	osmap
com_k2	OSMap - K2 Plugin	osmap
com_kunena	OSMap - Kunena Plugin	osmap
com_mtree	OSMap - Mosets Tree Plugin	osmap
com_sobipro	OSMap - SobiPro Plugin	osmap
com_virtuemart	OSMap - Virtuemart Plugin	osmap

1 de julio de 2016

com_weblinks	OSMap - WebLinks Plugin	osmap
akeebabackup	Quick Icon - Akeeba Backup Notification	quickicon
extensionupdate	Enlaces rápidos - Notificación de actualizaciones de extensiones para Joomla!	quickicon
joomlaupdate	Enlaces rápidos - Notificación de actualizaciones para Joomla!	quickicon
kunena	Iconos Rápidos - Foro Kunena	quickicon
categories	Buscar - Categorías	search
contacts	Buscar - Contactos	search
content	Buscar - Contenido	search
newsfeeds	Buscar - Canales electrónicos	search
tags	Buscar - Etiquetas	search
cache	Sistema - Caché	system
debug	Sistema - Depurador	system
highlight	Sistema - Resaltador	system
kunena	Sistema - Foros Kunena	system
lazyloadforjoomla	System - LLFJ - Lazy Load for Joomla!	system
log	Sistema - Acceso de usuario	system
logout	Sistema - Desconectar	system
osssystem	System - OSSystem	system
redirect	Sistema - Redirección	system
remember	Sistema - Recuérdeme	system
sef	Sistema - SEF	system
joomla	Usuario - Joomla!	user
installer	Instalador - Instalar desde la web	webinstaller
creativecontactform	Creative Contact Form	system

Tabla 11. Plugins activos

Plantillas

Joomla tiene de base varios juegos de plantillas, de la parte del sitio y de la parte de la Administración. Para este proyecto se ha instalado una nueva plantilla para maquetar la parte correspondiente al sitio público. Por tanto, en la configuración de plantillas del Panel de Control de Joomla, tiene marcadas dos plantillas como predeterminadas, las únicas usadas en este proyecto aunque se podría seleccionar alguna de las no predeterminadas al crear un artículo:

- isis: Plantilla por defecto de la pantallas de Administración.
- jp-rooms: Plantilla por defecto de las pantallas del sitio web.

6.2.5 Categorías y artículos

En el gestor de contenidos del proyecto se han definido las siguientes categoría y artículos:

Categoría		Artículo	
Por defecto	Agrupar los artículos que conforman la	Galería (Alias: galeria)	Contiene la galería de imágenes de la web.

1 de julio de 2016

	estructura de la web.	Próximas lecturas (Alias: proximas-lecturas)	Contiene el listado de próximas lecturas a realizar en el club.
		Acerca de (Alias: acerca-de)	Página de presentación del club de lectura.
		Otros clubs de lectura (Alias: otros-clubs-de-lectura)	Listado de otros clubs de lectura.
Artículos	Incluye los artículos que hablan de temas específicos y que aportan contenido. Agrupa los contenidos de tipo blog.	Los números de 2015 (Alias: los-numeros-de-2015)	Entrada tipo blog donde se explican los números de 2015.
		El antropólogo inocente (Alias: el-antropologo-inocente)	Entrada tipo blog con la crítica del libro “El antropólogo inocente”.

Tabla 12 . Categorías y artículos en el proyecto

6.2.6 Gestión de menús

A continuación se indican la estructura de menús que definen la estructura de la página web.

- **Main menu.** Es el menú principal del sitio web, permite navegar por las diferentes partes de la web. El número de opciones del menú varía según si hay un usuario registrado viendo la página o no.
- **Menú Kunena.** Menú del foro que permite acceder a las funciones proporcionadas por el componente Kunena.

Menú	Punto de menú	Tipo	Acceso	Descripción
Main menu	Home (Alias: home)	Artículos » Mostrar una categoría en formato blog	Publico	Página principal del sitio.
	Catálogo	Crear una URL	Usuarios registrados	Catálogo de libros leídos en el club de lectura.
	Próximas lecturas (Alias: proximas-lecturas)	Artículos » Mostrar un solo artículo	Publico	Acceso al artículo Próximas lecturas.
	Foro (Alias: kunena-2016-05-09)	Crear un alias a otro elemento del menú	Usuarios registrados	Acceso al foro del sitio, proporcionado por el componente Kunena.

1 de julio de 2016

	Otros clubs (Alias: otros-clubs)	Artículos » Mostrar un solo artículo	Publico	Acceso al artículo Otros clubs.
	Acerca de (Alias: acerca-de)	Artículos » Mostrar un solo artículo	Publico	Acceso al artículo Acerca de.
	Galería (Alias: galeria)	Artículos » Mostrar un solo artículo	Publico	Acceso al artículo Galería.
	Contacto (Alias: contacto)	Creative Contact Form » Creative Contact Form	Publico	Da acceso al formulario para contactar con el club de lectura.
Menú Kunena	Foro (Alias: foro)	Foro Kunena » home	Publico	Acceso al foro del sitio, proporcionado por el componente Kunena.
	Categorías (Alias: categorias)	Foro Kunena » category	Publico	Acceso a las categorías del foro.
	Mensajes Recientes (Alias: recientes)	Foro Kunena » topics	Publico	Acceso a los mensajes recientes del foro.
	Nuevo Tema (Alias: nuevo-tema)	Foro Kunena » topic	Usuarios registrados	Acceso a la creación de un nuevo tema en el foro.
	Sin Respuesta (Alias: sin-respuestas)	Foro Kunena » topics	Usuarios registrados	Acceso a la lista de mensajes pendientes de respuesta del usuario registrado.
	Mis Temas (Alias: mis-temas)	Foro Kunena » topics	Usuarios registrados	Acceso a la lista con los temas abiertos por el usuario que ha accedido.
	Perfil (Alias: perfil)	Foro Kunena » user	Usuarios registrados	Permite acceder al perfil del foro y configurarlo.
	Buscar (Alias: buscar)	Foro Kunena » search	Publico	Permite acceder al buscador de contenidos del foro.

Tabla 13 Menús del proyecto

1 de julio de 2016

6.2.7 Gestión de imágenes en la galería.

La galería de imágenes se implementa sobre un artículo estándar de Joomla, sobre el que se diseña la estructura de álbumes.

Para la implementación de cada uno de los álbumes de imágenes se utiliza el Plugin de Joomla “SIGE - Simple Image Gallery Extended”, no forma parte de la instalación base de Joomla y se ha instalado desde el repositorio de extensiones de Joomla.

Este Plugin permite crear galerías con todas las imágenes contenidas en una carpeta. Para añadir una nueva galería en un artículo se deberá introducir una línea con la configuración de la galería, entre las etiquetas `{gallery}``{/gallery}` se deberá indicar las características de la galería y la carpeta, con su ruta relativa a la carpeta raíz ficheros multimedia de Joomla, con las imágenes (“galeria” en el siguiente ejemplo).

```
{gallery}galeria,crop=1,crop_factor=20, ratio=0,width=100,height=300,thumbs=0,random=1{/gallery}
```

Para modificar la forma en que se muestra las galerías, el Plugin acepta los siguientes parámetros:

- **width** - Ancho de la miniatura – Ejemplo: 300
- **height** – Alto de la miniatura – Ejemplo.: 300
- **gap_v** - Gap vertical- Ejemplo: 30
- **gap_h** – Gap horizontal - Ejemplo: 30
- **quality** - Calidad (jpg) - Rango: 1-100
- **quality_png** - Calidad (png) - Rango: 1-9 (Nivel de compresión)
- **displaynavtip** – Muestra control de navegación - 1 = Si, 0 = no
- **displayarticle** – Muestra el nombre del artículo - 1 = Si, 0 = no
- **thumbs** – Genera y guarda las miniaturas - 1 = Si, 0 = no
- **limit** – Activar la limitación - 1 = Si, 0 = no
- **limit_quantity** – Mostrar un número de imágenes – Ejemplo.: 10
- **noslim** – Desactivar el efecto de diapositiva - 1 = Si, 0 = no
- **sort** – Orden aleatorio - 1 = Si, 2 = no, ascendente, 3 = no, descendente, 4 = ascendente por fecha de modificación, 5 = descendente por fecha de modificación
- **root** – Ruta de las imágenes empezando por la carpeta raíz - 1 = Si, 0 = no
- **ratio** – Mantener el ratio de las imágenes - 1 = Si, 0 = no
- **caption** – Mostrar los enunciados - 1 = Si, 0 = no
- **iptc** – Título y descripción de los datos IPTC, IPTC establece el estándar de la industria para obtener información administrativa, descriptiva y de los derechos de autor sobre las imágenes. - 1 = Si, 0 = no
- **iptcutf8** – Los metadatos IPTC ya están codificados en UTF-8 - 1 = Si, 0 = no
- **print** – Mostrar la opción de imprimir - 1 = Si, 0 = no
- **count** – Informar la variable contador manualmente – Ejemplo: 5

1 de julio de 2016

- **single** – Muestra una única imagen - Ejemplo: imagename.jpg
- **scaption** – Enunciado en el modo “single”, solo usar acompañado del parámetro “single”. Ejemplo: Este es mi enunciado
- **single_gallery** – Muestra la galería con el parámetro “Single” - 1 = Si, 0 = no
- **salign** – Alinea la imagen en modo “single” - Ejemplo: left(izquierda) / center(centro) / right/(derecha)
- **connect** – Conecta las imágenes con otras sintaxis de llamada - Ejemplo: bildset
- **download** – Muestra el botón de descarga - 1 = Si, 0 = no
- **list** – Muestra las imágenes en una lista - 1 = Si, 0 = no
- **crop** – Activa la función “crop”, que realiza un recorte selectivo de la imagen para alinear tamaños. - 1 = Si, 0 = no
- **crop_factor** – Nivel de Zoom – Ejemplo: 50 para un 50 por ciento de zoom. (no hay que poner el %)
- **thumbdetail** – Escoge la sección de la imagen para la generación de la miniatura - 0 = 1:1, 1 = arriba a la izquierda, 2 = arriba a la derecha, 3 = abajo a la izquierda, 4 = abajo a la derecha
- **watermark** – Activa la marca de agua - 1 = Si, 0 = no
- **watermarkposition** – Posición de la marca de agua - 0 = Centrada, 1 = Arriba a la izquierda, 2 = arriba a la derecha, 3 = abajo a la izquierda, 4 = abajo a la derecha.
- **watermark_trans** – Transparencia de la marca de agua - 0 a 100 - 0 = opaca, 100 – completamente transparente.
- **encrypt** – Método de encriptado - 0 = ROT13 – débil pero rápido. 1= MD5 – robusto, rápido. 2 = SHA1 – muy robusto pero más lento que MD5
- **image_info** – Muestra el nombre de la imagen o los metadatos IPTC - 1 = Si, 0 = no
- **image_link** – Muestra un enlace - Ejemplo: clubdelecturaupna.es
- **image_link_new** – Abre el link en una nueva ventana - 1 = Si, 0 = no
- **column_quantity** – Imágenes por línea - Ejemplo: 3 (para 3 imágenes en una línea)
- **css_image** – Activa la descripción emergente CSS - 1 = Si, 0 = no
- **css_image_half** – Reduce a la mitad el tamaño en la descripción emergente- 1 = Si, 0 = no
- **copyright** – Enlace al autor - 1 = Si, 0 = no
- **word** – Enlaza la galería con una palabra - Ejemplo: Galería
- **watermarkimage** – Escoger una imagen de marca de agua diferente (la imagen ha de estar ubicada en plugins/content/sige/plugin_sige) - Ejemplo: watermark-new.png
- **calcmxthumbsize** – Calcula el tamaño máximo de todas las miniaturas - 1 = Si, 0 = no
- **fileinfo** – Información con origen en un fichero de texto - 1 = Si, 0 = no
- **turbo** – Activar el modo turbo - 1 = Si, 0 = no
- **resize_images** – Cambia de tamaño las imágenes originales - 1 = Si, 0 = no

1 de julio de 2016

- **width_image** – Anchura máxima de las imágenes a las que se le cambie el tamaño - Ejemplo: 800
- **height_image** – Altura máxima de las imágenes a las que se le cambie el tamaño - Ejemplo: 800
- **ratio_image** – Mantener el ratio de las imágenes originales - 1 = Si, 0 = no
- **images_new** – Sobrescribe las imágenes modificadas - 1 = Si, 0 = no

Para la gestión de las imágenes se usará la gestión de ficheros multimedia estándar de Joomla, este gestor no se limita a la carga de imágenes sino que permite realizar la subida al gestor de contenido de cualquier tipo de contenido multimedia (imagen, video o sonido). Los ficheros subidos mediante la gestión multimedia pueden ser referenciados desde los artículos o componentes.

El gestor multimedia permite realizar las tareas básicas de gestión:

- Crear carpetas con el botón “Crear carpeta” situado debajo de la ruta actual.
- Subir un archivo al servidor con vistas a utilizarlo posteriormente en un artículo u otro tipo de contenido.
- Eliminar archivos.
- Conocer la ruta de archivos.
- Previsualizar las imágenes.

Esta operativa es suficiente para los requerimientos del proyecto, pero en caso de necesitar un agestión más compleja, Joomla dispone de extensión para ampliar la funcionalidad del gestor multimedia.

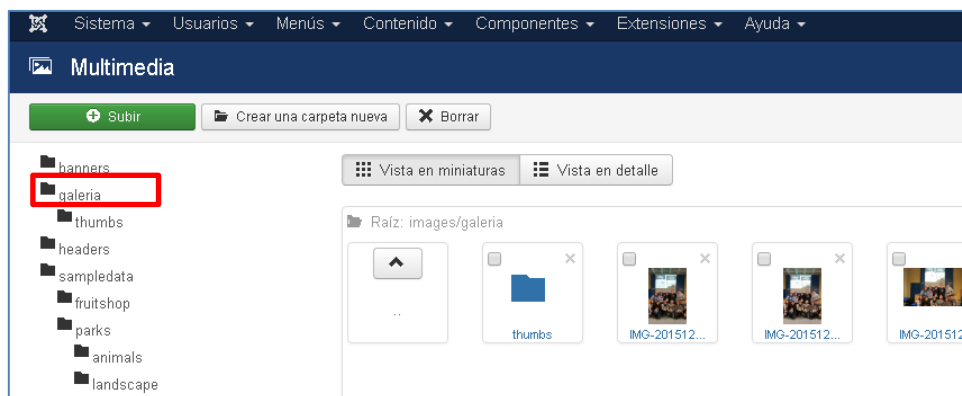


Ilustración 29 . Galería. Gestión multimedia

Para optimizar el rendimiento general de la web, se hará uso del patrón “Lazy Load”, o carga diferida, este patrón consiste en retrasar la carga o inicialización de un objeto hasta el mismo momento de su utilización. Esto contribuye a la eficiencia de los programas, evitando la precarga de objetos que podrían no llegar a utilizarse.

Este patrón se aplicará en las imágenes de la web, solo cargando aquellas que son visibles por el usuario que está navegando por la web.

1 de julio de 2016

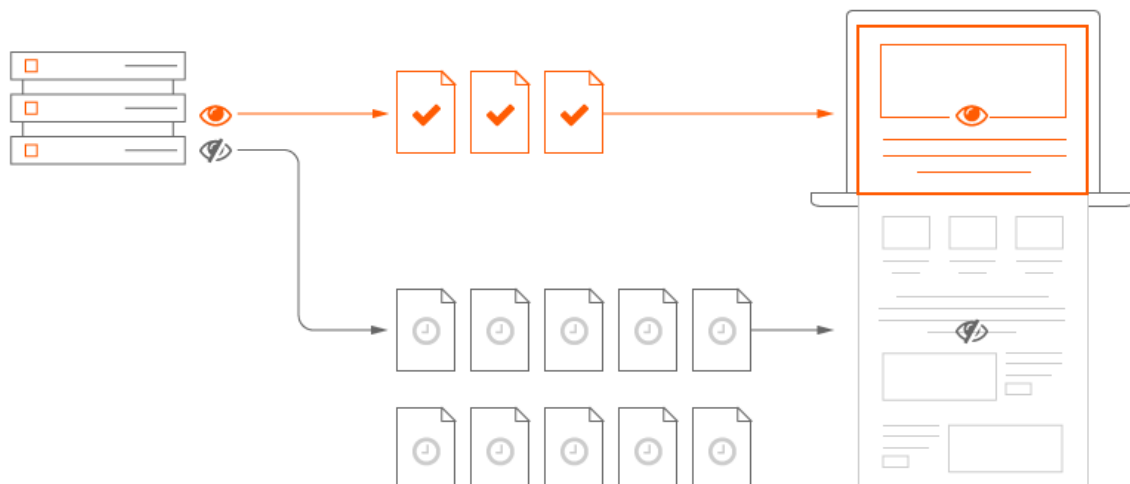


Ilustración 30 Galería. Patrón Lazy Load o carga diferida

Este patrón es especialmente importante en el rendimiento de la galería, ya que a medida que se incremente el número de imágenes de los álbumes el rendimiento se podría ver afectado al tener que cargarse multitud de imágenes, algunas de ellas sin llegar a usarse.

Para implementar este patrón se hará uso del Plugin de Joomla “LLFJ - Lazy Load for Joomla!”. Este Plugin realiza una carga diferida de las imágenes, es decir, al entrar a la galería en vez de cargarse todas las imágenes de la página se cargan solo aquellas que son visible, al mover la barra de desplazamiento hacia abajo se cargaran las imágenes que aparecen al visualizar la nueva parte de la pantalla. Solo se cargan aquellas imágenes que van a ser visualizadas.

7 Testing

Una de las ventajas de reaprovechar componentes existentes para la creación de la web es que se dispone de más tiempo para asegurar la calidad de los componentes desarrollados. Dentro de este proyecto se considera la realización de pruebas de componente, tanto unitarias como integradas, así como pruebas de sistema. Todas las pruebas se han automatizado para poder ser re-ejecutadas en el futuro, reduciendo así el esfuerzo a dedicar en pruebas para asegurar la calidad en futuras versiones.

7.1 Estrategia de pruebas

A continuación se detalla la estrategia de pruebas para cada uno de los niveles de prueba dentro del alcance de esta estrategia. A nivel general el modelo de pruebas escogido es el clásico modelo en V, por el cual cada etapa del ciclo de vida de software tiene su correspondencia en la etapa de pruebas.

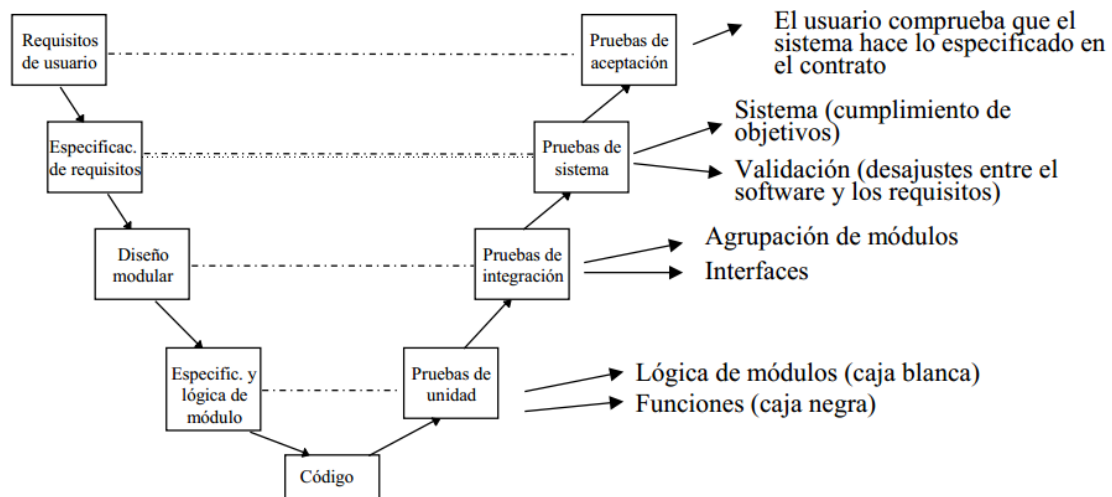


Ilustración 31 . Estrategia de pruebas. Modelo en V

7.1.1 Alcance de las pruebas

A continuación se indican las pruebas que están contempladas en esta estrategia de pruebas.

7.1.1.1 Pruebas dentro del alcance

A continuación se muestra una relación de las pruebas que se contemplan en la estrategia de pruebas de este proyecto, indicando para cada una de ellas el responsable de su correcta ejecución.

Tipo de prueba	Responsable
Componente unitaria	Desarrollo
Componente integrada	Desarrollo
Integrada de sistema	Desarrollo (misma persona que hace el análisis funcional)

1 de julio de 2016

Aceptación de usuario	Usuario de referencia del Club de lectura
-----------------------	---

7.1.1.2 Pruebas fuera del alcance

Quedan fuera del alcance las pruebas:

- **Análisis estático de código.** Descartado del alcance de pruebas.
- **Pruebas de seguridad.** Descartado del alcance de pruebas.
- **Pruebas de rendimiento y carga,** ya que no se dispone de la infraestructura adecuada para hacerlas con garantías de que los resultados sean representativos.
- **Pruebas de migración de datos,** ya que la web es de nueva creación y no hay datos a migrar.

7.1.2 Enfoque de las pruebas

El enfoque de las pruebas de esta estrategia tiene dos pilares, el primero es el uso del enfoque de pruebas basado en riesgos y el otro es la automatización de las pruebas de cara a poder reducir los esfuerzos de prueba en el futuro y poder reducir el tiempo necesario para promocionar un cambio a producción, teniendo en cuenta cambios que no requieran aprobación de usuario.

El enfoque de pruebas basadas en el riesgo (Risk Based Testing, RBT) es una metodología de prueba de software que da prioridad a la hora de probar a los requisitos (características y funciones) en función del riesgo que representan en términos de probabilidad de error y el impacto del mismo.

Este enfoque busca asegurar aquellas funcionalidades que son más críticas para el buen funcionamiento de una empresa, unidad de negocio o área, focalizando los esfuerzos de prueba en ellas en detrimento de otras funcionalidades secundarias. Para poder llevar a cabo este enfoque es necesario catalogar las funcionalidades en prioridad Alta, Media y Baja. Según el tiempo de prueba disponible se intentaran abordar el máximo de pruebas pero siempre cubriendo todas aquellas pruebas tipificadas como prioridad Alta.

7.1.2.1 Tipo de pruebas

A continuación se resumen los tipos de pruebas dentro del alcance de este proyecto:

Tipo de prueba	Estrategia	Responsable	Descripción
Componente unitaria	Prueba automatizada	Desarrollo	Durante el desarrollo se implementaran las pruebas necesarias para garantizar el correcto funcionamiento de los componentes desarrollados.
Componente integrada	Prueba automatizada	Desarrollo	Durante el desarrollo se implementaran las pruebas necesarias para garantizar el

1 de julio de 2016

			correcto funcionamiento de la integración de los componentes desarrollados.
Integrada de sistema	Prueba automatizada	Desarrollo	Una vez finalizado el desarrollo, se desarrollaran las pruebas funcionales destinadas a validar el correcto funcionamiento del sistema.
Aceptación de usuario	Prueba manual	Usuario de referencia del Club de lectura	Una vez finalizadas las pruebas se dará acceso a los usuarios al entorno de test para poder evaluar la funcionalidad y ver que esta se ajusta a los requerimientos.

Tabla 14 . Enfoque de pruebas. Tipos de pruebas

7.1.2.1.1 Pruebas componente

Pruebas de componente puede haber de dos tipos, de componente unitaria e integrada. A lo largo de este bloque se explicaran los dos tipos tanto a nivel teórico como de implementación.

Pruebas componente unitaria

Para comprender que es una prueba unitaria es importante definir qué entendemos por “unidad”: Una unidad es simplemente una porción de código o una funcionalidad que realiza una acción específica, de la cual podemos probar los resultados.

En el caso de PHP o Javascript, la unidad es el método.

Una prueba unitaria es una prueba para asegurarnos que la porción de código o funcionalidad hace lo que debería hacer, independientemente del resto de la aplicación.

En Java, el objetivo de las pruebas unitarias es probar el comportamiento de cada uno de los métodos de la clase de manera aislada.

En PHP o Javascript, el objetivo de las pruebas unitarias es probar el comportamiento de cada uno de los métodos de la clase de manera aislada. Se trata del primer nivel de pruebas que debe superar el código.

Un correcto diseño y ejecución de pruebas unitarias va a repercutir de forma muy positiva en el proyecto, algunas de las ventajas más destacables son:

- **Mejoran la calidad de los productos entregados**
 - **Permiten una revisión exhaustiva y sistemática de los componentes:** Constituye una comprobación sistemática de los componentes, que de otra forma (a través de otros niveles de pruebas) nunca vamos a obtener.

1 de julio de 2016

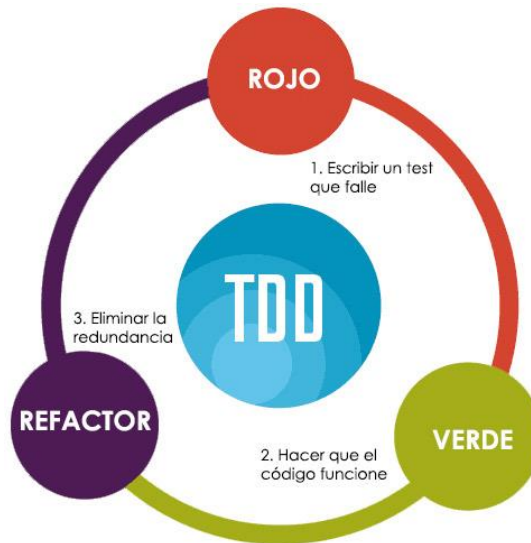
- **Favorece la adopción de buenas prácticas**
 - **Predispone para la creación de código desacoplado y cohesionado:** La programación de este tipo de pruebas permite al programador distinguir de manera más clara, las partes del software que deben trabajar de manera independiente y las que colaboran.
 - **Están orientadas a la aceptación final del producto por parte del usuario.** El objetivo principal de este nivel de pruebas, es dar confianza al usuario de que el sistema funcionará de acuerdo a sus expectativas, cumpliendo los requisitos funcionales y operativos esperados. Normalmente, tras las pruebas de sistema.
 - **Documentan el código:** A través de las pruebas podemos comprender mejor qué hace un módulo y qué se espera de él.
 - **Fomentan el cambio y la refactorización:** Permiten realizar pruebas sobre los cambios de forma rápida, lo cual favorece la mejora del código a través de cambios con mayores garantías de que no introducen errores.
- **Aumentan la productividad en el proyecto**
 - **Evitan el efecto bola de nieve:** La correcta ejecución de pruebas unitarias permite la detección de errores en etapas tempranas, por lo que pueden ser corregidos antes de que se produzcan dependencias.
 - **Los errores que se detectan se pueden corregir más fácilmente que con otro tipo de pruebas:** La detección temprana y localizada de los errores facilita su resolución.
 - **Reducen los problemas y tiempos dedicados a la integración:** Permiten llegar a la fase de integración con un grado alto de seguridad de que el código está funcionando correctamente, incluso las dependencias (se pueden simular).

Las pruebas unitarias pero tienen algunas limitaciones:

- Ayudan a detectar defectos, pero no garantizan la ausencia de éstos. Este hecho, es algo que va a ocurrir siempre en el testing, las pruebas muestran la existencia de defectos, no la ausencia de los mismos.
- Al estar ejercitando trozos de código o funcionalidades independientes de forma aislada, no servirán para descubrir problemas que afecten a todo el sistema en su conjunto. Por eso son más efectivas si se combinan con otros niveles de prueba.

Para la implementación de los tests unitarios se seguirá la metodología TDD, Test Driven Development. TDD Es un proceso de desarrollo de software que implica la creación de test para comprobar funcionalidades antes de su codificación, es decir escribir primero las pruebas unitarias, o sea, escribir el test antes de escribir el programa y refactoring.

1 de julio de 2016

**Ilustración 32 .Ciclo del desarrollo de pruebas con TDD**

A continuación se muestra el ciclo de desarrollo con TDD:

1. Escribir un test e intentar compilar, aunque aún no sea posible porque no se desarrolló la funcionalidad correspondiente. Escribir la mínima cantidad de código posible para que la prueba unitaria compile y ejecutarla. Deberá fallar, ya que no tiene la funcionalidad requerida.
2. Escribir el test de tal manera, que al ser ejecutado, este no falle. La idea es que la función escrita no sea lo más elegante y óptimo posible, sino que simplemente, permita pasar el test unitario exitosamente.
3. Refactorizar el código de la función, para que sea lo más óptimo posible. Es muy posible que existan varias instancias de refactorización del código de la función en distintas etapas del proyecto.
4. Escribir un nuevo test unitario para otra funcionalidad.

Pruebas de componente integradas

Las pruebas de integración tienen como objetivo verificar la correcta integración de todas las unidades probadas en la prueba unitaria.

Herramientas

Para el desarrollo del proyecto y la validación de los desarrollos con las pruebas generadas se hará uso de dos herramientas OpenSource, una por cada tecnología a probar.

- PHPUnit: Para la automatización de las pruebas de componentes sobre código PHP.
- QUnit: Para la automatización de las pruebas de componentes sobre código Javascript.

1 de julio de 2016

Gracias a la automatización de estas pruebas se podrán realizar ciclos de prueba sin tener que dedicar esfuerzo manual, además de reducir el tiempo de validación de los cambios.

PHPUnit

PHPUNIT es un framework basado en xUnit destinado a hacer las pruebas de Componentes, tanto unitarias e integradas, del código desarrollado en Javascript.

PHPUnit es el framework utilizado para el desarrollo del proyecto Joomla y es posible descargar del repositorio de código fuente Git (<https://github.com/joomla/joomla-cms>) las pruebas utilizadas en la validación de las versiones, pudiendo usarlas como base para extender las pruebas que validen personalizaciones en el gestor de contenidos.

PHPUnit se usará conjuntamente con Xdebug, Xdebug es una extensión de PHP para hacer debug con herramientas de depuración tradicionales, desde el editor, tal como se hace en lenguajes de programación clásicos. Xdebug permite analizar el contenido de las variables y también realizar el seguimiento del flujo de ejecución, para saber qué es lo que realmente está ocurriendo cuando algo no funciona como se esperaba. Proporciona información sobre todas las variables creadas en un momento dado de la ejecución de la aplicación y se puede ver su contenido de manera expandida. Ofrece también la posibilidad de marcar puntos de ruptura en el código, donde el flujo del programa se detendrá para poder ver el estado de las variables, o la traza de ejecución.

Para poder utilizar PHPUNIT, una vez instalada esta e instalado Xdebug, se deberá realizar la configuración, para la configuración del framework se deberá crear un fichero *phpunit.xml* que contendrá la configuración del proyecto en el que se quieren ejecutar las pruebas. Para el caso pruebas unitarias del Sistema Gestor de Contenido Joomla el fichero sería de la siguiente forma:

1 de julio de 2016

```

<?xml version="1.0" encoding="UTF-8"?>
<phpunit bootstrap="tests/bootstrap.php" colors="true">
  <php>
    <const name="JTEST_DATABASE_MYSQLI_DSN"
value="host=localhost;dbname=bitnami_joomla;user=bn_joomla;pass=
ca81b2a42d" />
  </php>
  <testsuites>
    <testsuite name="Database">
      <directory>tests/suites/database</directory>
    </testsuite>
    <testsuite name="Unit">
      <directory>tests/suites/unit</directory>
    </testsuite>
  </testsuites>

  <logging>
    <log type="coverage-html" target="build/coverage"
title="Joomla-Platform" charset="UTF-8" yui="true"
highlight="true"
      <lowUpperBound="35" highLowerBound="70" />
    <log type="coverage-clover"
target="build/logs/clover.xml" />
    <log type="junit" target="build/logs/junit.xml"
logIncompleteSkipped="false" />
  </logging>
  <filter>
    <whitelist addUncoveredFilesFromWhitelist="true">
      <directory
suffix=".php">libraries/joomla</directory>
      <file>libraries/import.php</file>
      <file>libraries/loader.php</file>
      <file>libraries/platform.php</file>
    </whitelist>
  </filter>
</phpunit>

```

Se indica tanto el fichero php que realiza la carga de las variables de sesión (bootstrap.php) como que muestre el resultado con colores para facilitar la legibilidad (colors="true").

Configuración de la base de datos.

Configuración de los TestSuite, dos "Database" con las pruebas de BBDD y Unit con otras pruebas.

Define los logs a realizar.

Indica los ficheros que no ha de tener en cuenta en el cálculo de la cobertura.

Las pruebas se han de implementar extendiendo de la clase PHPUnit_Framework_TestCase, del framework PHPUNIT.

```
class JLoaderTest extends PHPUnit_Framework_TestCase
```

Al invocar la ejecución del programa mediante la llamada desde la carpeta raíz de Joomla, p.e.:

```
phpunit -testsuite <nombre_testsuite>
```

PHPUnit ejecutará todos los métodos de las clases, o los que se encuentren en la carpeta indicada en la línea de comandos o en la etiqueta <testSuite> del test suite indicado por línea de comandos, que hereden de PHPUnit_Framework_TestCase y empiecen por el literal "test".

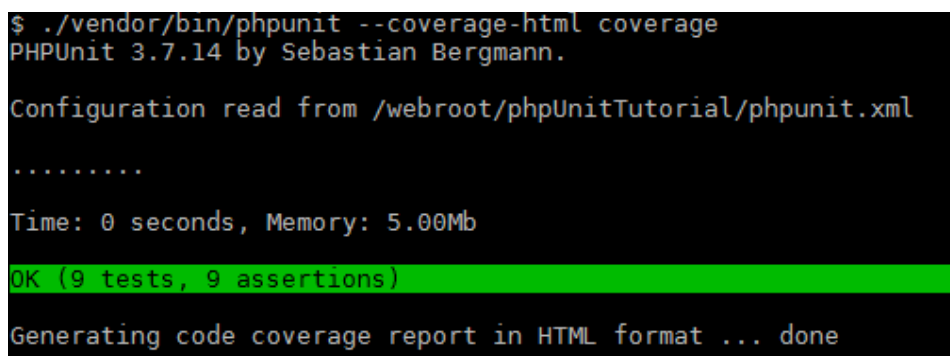


Ilustración 33. Resultados de ejecución de pruebas en PHPUnit

1 de julio de 2016

A continuación se muestra un ejemplo de prueba desarrollada con PHPUnit.

```
public function testRegisterNamespaceResetPath()
{
    // Insert a first path.
    $path = dirname(__FILE__) . '/stubs/discover1';
    JLoader::registerNamespace('discover', $path);

    // Reset the path with a new path.
    $path = dirname(__FILE__) . '/stubs/discover2';
    JLoader::registerNamespace('discover', $path, true);

    $namespaces = JLoader::getNamespaces();
    $this->assertCount(1, $namespaces['discover']);
    $this->assertContains($path, $namespaces['discover']);
}
```

En negrita se pueden ver las validaciones realizadas, mediante las funciones assertXX.

A continuación se muestra la lista de aserciones posibles:

- **assertArrayHasKey()**: Devuelve verdadero si el array contiene la clases especificada.
- **assertClassHasAttribute()**: Devuelve verdadero si la clase tiene el atributo indicado como parámetro.
- **assertArraySubset()**: Devuelve verdadero si el array pasado como parámetro es un subconjunto del array a evaluar.
- **assertClassHasStaticAttribute()**. Devuelve verdadero si la clase pasada como paramétrico tiene el atributo indicado estático.
- **assertContains()**. Devuelve verdadero si la colección contiene el elemento pasado como parámetro.
- **assertContainsOnly()**. Devuelve verdadero si el array solamente contiene elementos del tipo indicado como parámetro.
- **assertContainsOnlyInstancesOf()**. Devuelve verdadero si el array solamente contiene elementos de la clase indicada como parámetro.
- **assertCount()**. Devuelve verdadero si el número de elementos de la colección es el pasado como parámetro.
- **assertEmpty()**. Devuelve verdadero si la colección está vacía.
- **assertEqualXMLStructure()**. Compara dos elementos en una estructura DOM para ver si son el mismo.
- **assertEquals()**. Devuelve verdadero si los dos elementos pasados como parámetro son iguales.
- **assertFalse()**. Devuelve verdadero si el parámetro es falso.
- **assertFileEquals()**. Devuelve verdadero si los dos ficheros pasados como parámetro son iguales.
- **assertFileExists()**. Devuelve verdadero si el fichero pasado como parámetro existe.
- **assertGreaterThan()**. Devuelve verdadero si el valor pasado como parámetro es mayor que el que se compara.

1 de julio de 2016

- **assertGreaterThanOrEqual():** Devuelve verdadero si el valor pasado como parámetro es mayor o igual que el que se compara.
- **assertInfinite().** Devuelve verdadero si el parámetro es infinito.
- **assertInstanceOf().** Devuelve verdadero si el parámetro es del mismo tipo que la clase pasada como parámetro.
- **assertInternalType().** Valida si el valor pasado por parámetro es del tipo indicado.
- **assertJsonFileEqualsJsonFile().** Compara dos ficheros Json y devuelve verdadero si son iguales.
- **assertJsonStringEqualsJsonFile().** Compara una cadena Json con un fichero Json y devuelve verdadero si son iguales.
- **assertJsonStringEqualsJsonString().** Compara dos cadenas Json y devuelve verdadero si son iguales.
- **assertLessThan().** Devuelve verdadero si el valor pasado como parámetro es menor que el comparado.
- **assertLessThanOrEqual().** Devuelve verdadero si el valor pasado como parámetro es menor o igual que el comparado.
- **assertNaN().** Devuelve verdadero si el parámetro es NaN(no definido).
- **assertNull().** Devuelve verdadero si el parámetro es nulo.
- **assertObjectHasAttribute().** Devuelve verdadero si el objeto tiene el atributo indicado.
- **assertRegExp().** Devuelve verdadero si el parámetro no cumple la expresión regular.
- **assertStringMatchesFormat().** Devuelve verdadero si el parámetro tiene el formato indicado como parámetro.
- **assertStringMatchesFormatFile().** Devuelve verdadero si el parámetro tiene el formato indicado en el fichero parámetro.
- **assertSame().** Devuelve verdadero si las dos variables tienen el mismo tipo y valor.
- **assertStringEndsWith().** Devuelve verdadero si la cadena de texto acaba con el valor indicado como parámetro.
- **assertStringEqualsFile().** Devuelve verdadero si la cadena de texto es igual al contenido pasado como parámetro.
- **assertStringStartsWith().** Devuelve verdadero si la cadena de texto empieza con el valor indicado como parámetro.
- **assertThat()** Permite hacer validaciones complejas mediante el uso de restricciones.
(<https://phpunit.de/manual/current/en/appendixes.assertions.html#appendixes.assertions.assertThat.tables.constraints>)
- **assertTrue().** Devuelve verdadero si el parámetro es verdadero.
- **assertXmlFileEqualsXmlFile().** Compara dos ficheros XML y devuelve verdadero si son iguales.

1 de julio de 2016

- **assertXmlStringEqualsXmlFile()**. Compara un fichero XML y una cadena XML y devuelve verdadero si son iguales.
- **assertXmlStringEqualsXmlString()**. Compara dos cadenas XML y devuelve verdadero si son iguales.

Para realizar pruebas unitarias en funciones que requieren de otros componentes de los cuales se conoce el funcionamiento pero no se quiere hacer uso, ya que la prueba sería integrada, PHPUnit al igual que otros frameworks de pruebas unitarias permite usar elementos que simular el comportamiento de estos componentes, hay de dos tipos:

- **Stub**

De acuerdo con la definición formal del ISTQB, un Stub es la implantación estructural (esqueleto) o de propósito especial de un componente software, usado para desarrollar o probar un componente al que se llama o es dependiente de él de algún modo. Sustituye a un componente llamado. [Según IEEE 610]. Las posibles respuestas que pueden devolver son predefinidas. P.e.

```
<?php
use PHPUnit\Framework\TestCase;

class StubTest extends TestCase
{
    public function testStub()
    {
        // Se crea el stub de la clase SomeClass.
        $stub = $this->createMock(SomeClass::class);

        // Se configura el stub.
        $stub->method('doSomething')
            ->willReturn('foo');

        // La llamada a $stub->doSomething() ahora devolverá
        'foo'.
        $this->assertEquals('foo', $stub->doSomething());
    }
}
?>
```

- **Mock**

Un Mock sustituye al igual que un Stub a un trozo de código pero las respuestas están programadas de tal forma que cumplen la especificación de lo que se espera que reciban las llamadas.

```
<?php
use PHPUnit\Framework\TestCase;

class SubjectTest extends TestCase
{
    public function testObserversAreUpdated()
    {
        // Se crea un mock para la clase Observer, únicamente
        // se mockea el método update().
        $observer = $this->getMockBuilder(Observer::class)
```

1 de julio de 2016

```

        ->setMethods(['update'])
        ->getMock();

    // Definir la lógica esperada para el método update
    //update() para ser llamado una vez con el parámetro
    // 'something'
    $observer->expects($this->once())
        ->method('update')
        ->with($this->equalTo('something'));

    // Crear un objeto Subject y llamar a su función attach
    //pasándole el objeto observer como parámetro.
    $subject = new Subject('My subject');
    $subject->attach($observer);

    // Llamar al método doSomething() en el objeto $subject
    //que esperamos que llamará al método mockeado update()
    //del objeto $observer con el parámetro 'something'.
    $subject->doSomething();
    }
}
?>

```

QUnit

QUnit es un framework destinado a hacer las pruebas de Componentes, tanto unitarias e integradas, del código desarrollado en Javascript. Este framework, basado en xUnit es el usado en proyectos Javascript tan importantes como jQuery, jQuery UI y jQuery Mobile y es capaz de probar cualquier Javascript genérico.

El framework QUnit está compuesto por dos ficheros, uno que incluye el motor de prueba y generación de resultados y otro con los estilos a aplicar en los resultados:

- qunit-2.0.0.js
- qunit-2.0.0.css

A continuación se muestra un fichero de configuración de pruebas, el cual sigue la siguiente estructura, donde tests.js es el Javascript que contiene las pruebas a ejecutar y los ficheros del framework QUnit están ubicados en la carpeta *qunit* del servidor:

```

<!DOCTYPE html>
<html>
<head>
....
Librerías Javascript de las que depende el código a probar
....
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width">
  <title>QUnit Example</title>
  <link rel="stylesheet" href="qunit/qunit-2.0.0.css">
</head>
<body>
  <div id="qunit"></div>
  <div id="qunit-fixture"></div>

```

1 de julio de 2016

```

<script src="/qunit/qunit-2.0.0.js"></script>
<script src="tests.js"></script>
</body>
</html>

```

El fichero *test.js* contendrá el listado de pruebas a realizar, a continuación se muestra un ejemplo de la prueba de componente integrada que valida el comportamiento de la función cuando no se encuentran datos. En negrita se marcan los elementos estándar de cualquier prueba implementada con QUnit y en azul se especifica el código añadido para la sincronización de la llamada Ajax, ya que esta es asíncrona.

Nombre de la prueba: `GetLibrosPais_NoData_Test`

Función bajo prueba: `getLibrosPais`

```

/*Prueba de no retorno de ningun libro*/
QUnit.test( "GetLibrosPais_NoData_Test", function( assert ) {
  getLibrosPais('la', '');
  var done = assert.async();
  setTimeout(function() {
    assert.ok(ListaLibrosData.data.length == 0, "Passed!" );
    done();
  }, 2000 );
});

```

`assert.async()` es la función encargada de sincronizar la llamada Ajax asíncrona, mediante la instrucción `assert.ok` se explicita que validación se ha de realizar.

Para la ejecución de las pruebas únicamente es necesario entrar a la página web donde se ha inicializado el framework QUnit, en el caso de este proyecto:

URL: <http://clubdelecturaupna.es/tests/qunit.html>

Al acceder a ella se ejecutarán de forma automática las pruebas y mostrará el resultado por pantalla.

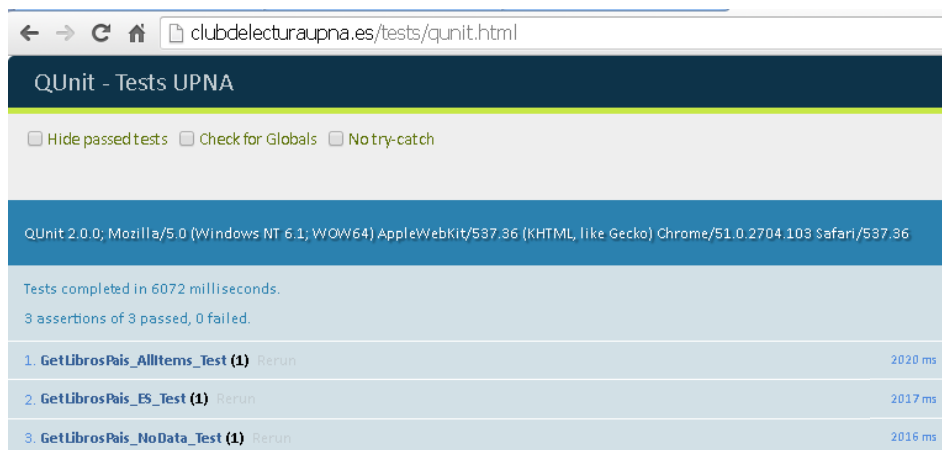


Ilustración 34 . QUnit. Resultados de pruebas de componente automaticas

1 de julio de 2016

A continuación se indican las instrucciones que se pueden utilizar para realizar las validaciones en QUnit.

- **async()** - Instrucción de QUnit para esperar en una operación asíncrona.
- **deepEqual()** - Comparación recursiva profunda, trabajando con tipos primitivos, arrays, objetos, expresiones regulares, fechas y funciones.
- **equal()** - Comparación no estricta, equivalente al `assertEquals` de JUnit.
- **expect()** - Especifica cuantas instrucciones de tipo “assert” se esperan en la ejecución de la prueba.
- **notDeepEqual()** - la inversa a la comparación recursiva profunda, trabajando con tipos primitivos, arrays, objetos, expresiones regulares, fechas y funciones.
- **notEqual()** - Comparación no estricta, comprobando la diferencia.
- **notOk()** - Comprobación booleana, inversa a `OK()`, equivalente `assertFalse()` en JUnit. Pasa si el primer argumento es falso.
- **notPropEqual()** - Comparación estricta de las propiedades de los objetos, comprobando que no son iguales.
- **notStrictEqual()** - Comparación estricta, comprobando que no son iguales.
- **ok()** - Comprobación booleana, equivalente `assertTrue()` en JUnit. Pasa si el primer argumento es verdadero.
- **propEqual()** - Comparación estricta de las propiedades de los objetos, comprobando que son iguales.
- **pushResult()** - Reporta el resultado de una validación a medida.
- **strictEqual()** - Validación estricta de tipo y valor.
- **throws()** - Comprueba si un callback de una llamada devuelve una excepción, opcionalmente compara el error lanzado.

7.1.2.1.2 Pruebas funcionales automatizadas

Para estas pruebas vamos a utilizar la herramienta HP UFT. Es una suite privada de HP, con coste de licencia. Tiene soporte para casi cualquier tipo de aplicación (Web, Escritorio, Móvil, etc...), y permite interactuar con componentes ActiveX y Flex (swf). Una característica importante de UFT es que permite integrar juegos de datos (en formato excel), dentro de una DataTable, lo que permite desarrollar fácilmente los scripts de automatización mediante técnicas de DDT (Data Driven Testing) y KDT (Keyword Driven Testing).

Es posible elaborar las operativas a automatizar mediante el uso de diagramas de flujo que incorporan las Actions necesarias y reutilizables, lo que facilita la lectura y comprensión de los scripts de automatización.

Genera reportes e informes de forma automática.

Permite, tras un cambio ejecutar a coste 0 una regresión parcial de las distintas funcionalidades a la vez que se preserva el conocimiento funcional de la prueba; ya que es “lo que hace” la máquina.

1 de julio de 2016

7.1.2.1.3 Pruebas de aceptación

Las pruebas de aceptación están orientadas a la aceptación final del producto por parte del usuario. El objetivo principal de este nivel de pruebas, es dar confianza al usuario de que el sistema funcionará de acuerdo a sus expectativas, cumpliendo los requisitos funcionales y operativos esperados. Normalmente, se realizan tras las pruebas de sistema.

7.1.2.2 Criterios de entrada/salida

Para dar por finalizado las pruebas en un nivel de pruebas, todas las pruebas del último ciclo de pruebas han de haberse pasado con resultado satisfactorio (100% OK).

Antes de iniciar las pruebas de un nivel superior, se han de cumplir los criterios de salida del actual.

7.1.2.3 Entornos de prueba

Para el desarrollo y la ejecución de las pruebas se ha creado un entorno de desarrollo/test en una máquina virtualizada. Los pasos seguidos para la creación del entorno de desarrollo han sido.

1. Descarga de la imagen de la máquina virtual de la web de Bitnami (<https://bitnami.com/stack/joomla/installer>) esta web permite descargar máquinas virtuales, descargar los instalables o el servicio en la nube de las aplicaciones web más utilizadas, junto con todas sus dependencias, en concreto para el entorno de desarrollo se han utilizado los ficheros correspondientes a la versión de Joomla 3.5.1, bitnami-joomla-3.5.1-1-linux-ubuntu-14.04-x86_64.
2. En caso de no tener instalada la aplicación Oracle Virtual Box, se debería descargar de su web el instalable e instalarlo en la máquina del desarrollador y/o servidor de pruebas (<https://www.virtualbox.org/wiki/Downloads>).
3. Se deberá crear una nueva máquina virtual en Virtual Box con los ficheros descargados de Bitnami, para ello:
 1. se deberá clicar a “Nueva” y en la pantalla que aparece rellenar los datos como indica la imagen y se deberá pulsar en “Siguiente”.

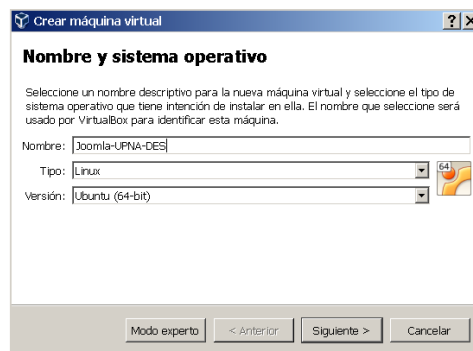


Ilustración 35 . Creación de la maquina virtual. SO

1 de julio de 2016

2. Aparecerá la pantalla para asignar la memoria RAM a la máquina virtual y se deja la opción por defecto. Si se detectara una mayor cantidad de memoria se puede modificar a posteriori.
3. A continuación se seleccionará el disco duro descargado de Bitnami, el cual se ha de descomprimir previamente. Finalmente se deberá pulsar a crear y se creará la máquina virtual.

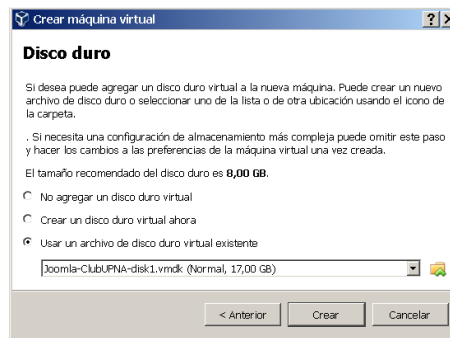


Ilustración 36 . Creación de laa maquina virtual. Añadir disco virtual

4. Se deberá cambiar la configuración del tipo de red utilizado por la máquina virtual. Para ello se deberá ir a “*Configuración > Red*” y modificar el valor de “*Conecto a*” de “*NAT*” a “*Adaptador puente*”.
5. Se deberá iniciar la máquina virtual, seleccionándola del listado y clicando en “*Iniciar*”, se deberá esperar hasta que la máquina virtual se cargue completamente y aparezca la pantalla de registro donde habrá que entrar el usuario y contraseña por defecto.
6. Se deberá cargar el mapa de teclado en español, para poder escribir los comandos correctamente. Para ello se tecleará `sudo loadkeys es`
7. Se deberá modificar la configuración del servidor web Apache para permitir conexiones remotas, ya que por defecto solo permite conectar desde el Local de la máquina virtual.
8. Se deberá modificar la configuración del FTP para permitir el acceso remoto, concretamente se ha de modificar el fichero: `/etc/vsftpd.conf` sustituyendo el valor actual de la propiedad `listen_address` por `0.0.0.0`.
9. Se deberá modificar la configuración de PHPMyAdmin para permitir el acceso remoto, se deberá modificar el fichero: `/opt/bitnami/apps/phpmyadmin/conf/httpd-app.conf`, concretamente los valores remarcados en negrita:


```
Allow from all
Satisfy all
</IfVersion>
<IfVersion >= 2.3>
Require all granted
```
10. A continuación se instalará el framework PHPUnit para la ejecución de las pruebas de componente sobre el código PHP.


```
$ wget https://phar.phpunit.de/phpunit.phar
$ chmod +x phpunit.phar
```

1 de julio de 2016

```
$ sudo mv phpunit.phar /usr/local/bin/phpunit
```

11. Seguidamente se instalara la librería Xdebug:

1. Descargar xdebug-2.3.2.tgz
2. Descomprimir el fichero descargado `tar -xvzf xdebug-2.3.2.tgz`
3. `cd xdebug-2.3.2`
4. `phpize`
5. `sudo ./configure`
6. `sudo make`
7. `sudo cp modules/xdebug.so /usr/lib/php5/xdebug/`
8. `sudo nano /opt/bitnami/php/etc/php.ini` y cambiar la propiedad `zend_extension` por `zend_extension="/usr/lib/php5/xdebug/xdebug.so"`
`memory_limit = 526M`
9. `sudo /opt/bitnami/ctlscript.sh restart apache`

12. Se deberá restaurar la copia de seguridad de la base de datos y de los ficheros de Joomla, en caso de existir alguna instalación previa.

7.1.2.4 Estrategia de datos

Los datos de prueba se recogerán del entorno productivo, mediante una copia de la base de datos de producción al entorno de pruebas. De esta manera, y debido a que el volumen de datos no se prevé elevado, se podrán tener ambos entornos alineados.

7.1.2.5 Gestión de defectos

Defect/Defecto: Imperfección en un componente o sistema que puede causar que el componente o sistema falle en desempeñar las funciones requeridas. Por ejemplo, si se localiza un defecto durante una ejecución puede causar un fallo en el componente o sistema.

8 Aspectos SEO

8.1 Introducción al SEO

SEO(Search Engine Optimization) significa asegurar que los buscadores pueden encontrar, clasificar y valorar el contenido de la página web adecuadamente, donde “Adecuadamente” significa “desde la perspectiva del ser humano que realiza una búsqueda relevante”.

El posicionamiento SEO tiene que ver con todas las acciones hechas en nuestra web con el objetivo de mejorar el posicionamiento de la misma en los diversos motores de búsqueda, acciones de diversa índole encaminadas al aumento de la visibilidad de la web y con el objetivo de aumentar el tráfico de visitantes.

Los principales buscadores de uso general, aquellos que se usan para buscar páginas web en base a los criterios de búsqueda introducidos son:

- Baidu, idioma Chino y Japonés
- Bing, Multilinguaje
- DuckDuckGo, Multilinguaje
- Ecosia, basado en Bing, Multilinguaje
- Exalead, Multilinguaje
- Gigablast, idioma Ingles
- Google, Multilinguaje
- Munax, Multilinguaje
- Qwant, Multilinguaje
- Sogou, idioma Chino
- Soso.com, idioma Chino
- Yahoo! , Multilinguaje
- Yandex, Multilinguaje
- Youdao, idioma chino

Hay que tener en cuenta que, a pesar de la practica hegemonía del buscador Google en el mundo, hay otros motores de búsqueda que tienen más relevancia en ciertos países, los más relevantes serian Yandex en Rusia o Baidu en China. Es importante tener esto en cuenta a la hora de definir la estrategia de SEO de la web ya que ciertos puntos como el idioma de las palabras clave o los metadatos pueden impactar en un mejor o peor posicionamiento en estos motores.

1 de julio de 2016

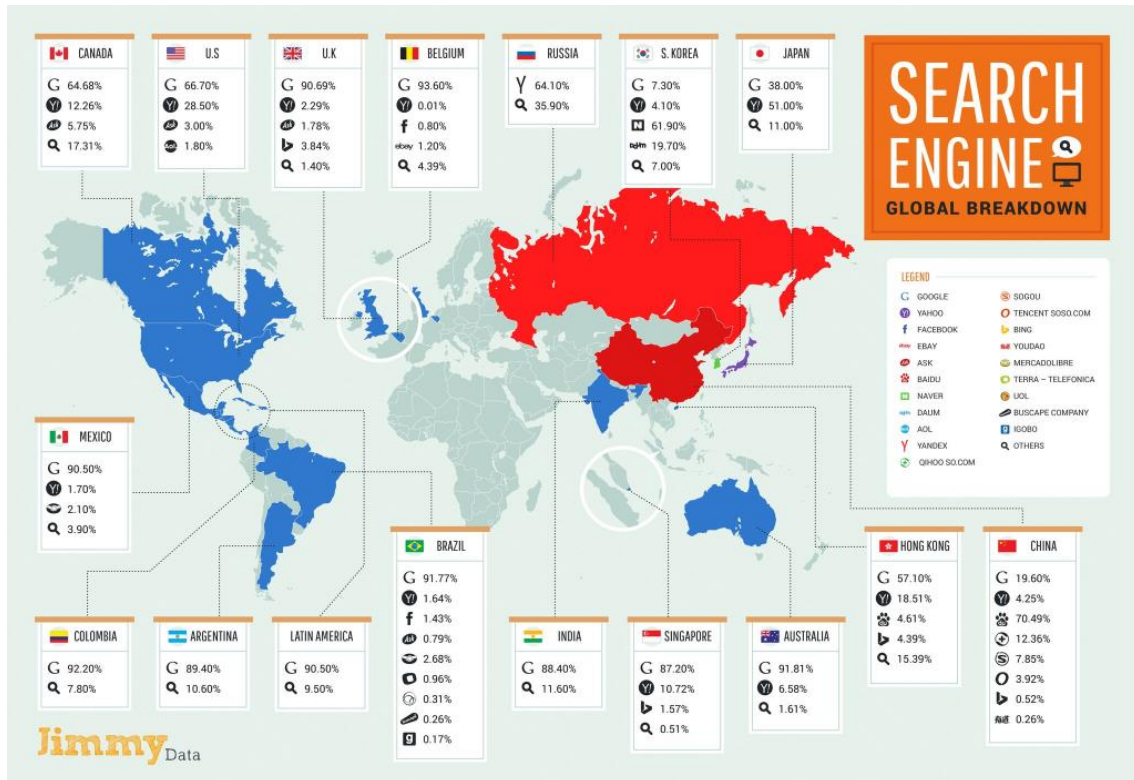


Ilustración 37 . SEO. Distribución de motores de búsqueda por países

A pesar que existen multitud de factores analizados por los motores de búsqueda para posicionar una página u otra se podría decir que hay dos factores básicos: la autoridad y la relevancia:

- La **Autoridad** es la popularidad de una web, los motores de búsqueda dan por supuesto que cuanto más popular es una página más valiosa es la información que contiene. Este factor es al que un motor de búsqueda da más importancia ya que se basa en la propia experiencia del usuario, cuanto más se comparta un contenido es que a más usuarios les ha parecido útil o interesante.
- La **Relevancia** es la relación que tiene una página frente a una búsqueda dada, esto no consiste que una página contenga un montón de veces el termino buscado (tal y como pasaba antes), si no que un motor de búsqueda se basa en cientos de indicadores de la propia página para determinar esto. Estos indicadores se van actualizando con regularidad, lo que obliga a las páginas webs a ir actualizando su estrategia SEO.

A su vez, las técnicas de optimización para motores de búsqueda se pueden dividir en dos grandes grupos:

- **On-site:** El SEO on-site se focaliza en la relevancia de la página, asegurándose de que la web está optimizada para que el motor de búsqueda pueda procesar correctamente el contenido de la misma. Dentro del SEO On-site se incluye la

1 de julio de 2016

optimización de palabras clave, el tiempo de carga de la página, la experiencia del usuario, la optimización del código y el uso de URL amigables.

- **Off-site:** El SEO off-site se centra en factores externos a la página web. Los indicadores más importantes a trabajar son el número y la calidad de los enlaces, presencia en redes sociales, menciones en medios locales, autoridad de la marca y rendimiento en los resultados de búsqueda, es decir, el CTR (Click Through Rate, porcentaje de clics que recibe una web en comparación con el número de impresiones que recibe la web en los motores de búsqueda) que tengan nuestros resultados en un motor de búsqueda.

8.2 Acciones de mejora de SEO

El primer paso para tener un buen posicionamiento en los buscadores es generar contenidos de calidad que interesen a la audiencia objetivo, ya que una página con un gran número de visitas tiene muchos puntos a la hora de ser posicionada en las primeras posiciones de las búsquedas de los principales buscadores.

Por otra parte, también es importante tener en cuenta una serie de pautas que harán que estos contenidos se indexen correctamente por parte de los motores de búsqueda y puedan ayudar a dar más visibilidad a la página. A continuación se tratan las principales pautas a tener en cuenta a la hora de querer mejorar el posicionamiento de la web.

8.2.1 Introducir Meta-datos

Introducir los meta-datos adecuados, descripción y palabras clave, alineados con el contenido de la página. Los meta-datos sirven para dar información a los motores de búsqueda acerca de la temática y los contenidos de nuestro sitio web.

- La **meta-descripción** es el párrafo que muestran los motores de búsqueda justo debajo del título de cada enlace de los resultados, permite dar información al buscador y a los posibles visitantes sobre la temática de la página. Con una buena descripción es posible conseguir que aumente el número de personas que hagan clic en el enlace de la web una vez lo han visualizado en la página de resultados de búsqueda. Por lo tanto, es importante expresar de forma concisa y resumida el contenido de nuestra página.

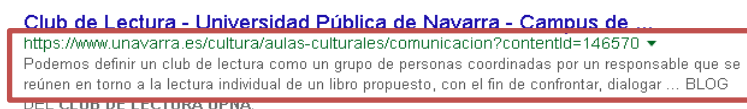



Ilustración 38 . SEO. Resultado de búsqueda en Google. Meta-descripción.

- Las **palabras clave** aportan información del sitio web a los motores de búsqueda. No es recomendable poner más de 10 aproximadamente, ya que si se abusa de las palabras clave los motores de búsqueda penalizarán la página.

1 de julio de 2016

Para establecer los metadatos en Joomla 3 se deberán seguir los siguientes pasos. Para establecer los meta-datos del sitio, de aplicación todas las páginas por herencia, a no ser que se explicita lo contrario en el elemento de menú correspondiente se ha de ir al Panel de Control > Sistema > Configuración Global > Configuración de los metadatos, y buscar los campos:

- **Metadescripción del Sitio.** Se informará la descripción del sitio.
- **Metapalabras clave.** Se informarán las palabras clave del sitio.

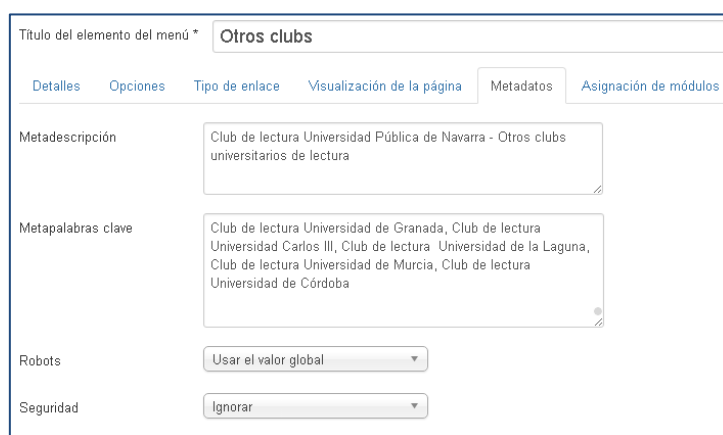


The screenshot shows the 'Configuración de los metadatos' (Global Metadata Configuration) interface. It includes three main sections: 'Metadescripción del sitio' (Site description) with a text area containing a paragraph about the University of Navarra reading club; 'Metapalabras clave' (Meta-keywords) with a text area listing 'libro, Novela, Literatura estadounidense, Literatura española, Narrativa, Comentario de Libros, cuento, UPNA, lectura, universidad'; and 'Robots' with a dropdown menu set to 'Seguir con indexación' (Follow global settings).

Ilustración 39. SEO. Configuración Global de Joomla 3

- La opción **Robots** se ha de informar a "Seguir con indexación", esto permitirá que los contenidos se indexen correctamente por parte de los motores de búsqueda.

Es posible establecer metadatos específicos en cada una de las páginas, para poder ajustar el posicionamiento de algunas páginas en concreto. Para ello se ha de ir a la configuración del elemento de menú correspondiente > Metadatos. Una vez aquí, estableceremos específicamente los campos "Metadescripción" y "Metapalabras clave" para esta página en concreto.



The screenshot shows the 'Metadatos' (Metadata) configuration for a specific menu item titled 'Otros clubs'. It features several tabs: 'Detalles', 'Opciones', 'Tipo de enlace', 'Visualización de la página', 'Metadatos', and 'Asignación de módulos'. The 'Metadatos' tab is active, showing: 'Metadescripción' (Club de lectura Universidad Pública de Navarra - Otros clubs universitarios de lectura); 'Metapalabras clave' (Club de lectura Universidad de Granada, Club de lectura Universidad Carlos III, Club de lectura Universidad de la Laguna, Club de lectura Universidad de Murcia, Club de lectura Universidad de Córdoba); 'Robots' (Usar el valor global); and 'Seguridad' (Ignorar).

Ilustración 40 . SEO. Configuración de los metadatos específicos para una página.

1 de julio de 2016

8.2.2 URL's amigables

En SEO son muy importantes las URL's amigables ya que van a permitir que la web genere direcciones limpias y que contendrán palabras relevantes. Para habilitar las URL's amigables en Joomla 3 se deben de seguir los siguientes pasos:

1. Renombrar el fichero "htaccess.txt" situado en la raíz del servidor web a ".htaccess".
2. Ir al panel de control de Joomla 3 > Sistema > Configuración Global y en la sección "Configuración SEO" se han de marcar "URLs amigables" y "Usar la reescritura de URLs"-

Ilustración 41 . SEO. Reescritura URL's

Una vez hecho el cambio las URL's generadas se modificaran de la siguiente manera:

Original: http://clubdelecturaupna.es/index.php?option=com_content&view=article&id=3&Itemid=107

URL amigable: <http://clubdelecturaupna.es/proximas-lecturas>

Por defecto, Joomla construirá la URL automáticamente a partir del campo "Título del elemento de menú", pero es posible definir el valor de la URL amigable de forma manual a través del campo "Alias" en la configuración del elemento del menú.

Ilustración 42. SEO. Configuración del "Alias"

1 de julio de 2016

8.2.3 Google Webmasters Tools

Las herramientas para webmasters de Google son un servicio gratuito que permite a los creadores de páginas web comprobar el estado de la indexación de sus sitios en el buscador y optimizar su visibilidad.

El servicio dispone de herramientas que permiten a los webmasters:

- Enviar y comprobar un mapa del sitio (Sitemap).
- Comprobar y ajustar la frecuencia de indexación, y ver con qué frecuencia GoogleBot visita un sitio determinado. El robot de rastreo web de Google ("araña"). El rastreo es el proceso mediante el cual el robot de Google descubre páginas nuevas y actualizadas y las añade al índice de Google.
- Producir y comprobar un archivo robots.txt
- Enumerar los enlaces de páginas internas y externas a ese sitio.
- Ver qué palabras clave en las búsquedas de Google han llevado a ese sitio, y la tasa de clic en las palabras de esa lista.
- Ver estadísticas sobre cómo Google indexa el sitio, y los errores que haya podido encontrar.
- Fijar un dominio por defecto (por ejemplo, hacer que clubdelecturaupna.es tenga preferencia sobre www.clubdelecturaupna.es), determinando que URL aparecerá en los resultados de las búsquedas.

Para crear una cuenta de Google WebMaster Tools, se debe entrar con una cuenta de Google a:

<https://www.google.com/webmasters/tools/home?hl=es>

1. Una vez creada la cuenta, tenemos que añadir el sitio web para poder gestionarlo. Para ello, podemos emplear la opción "Añadir una propiedad".

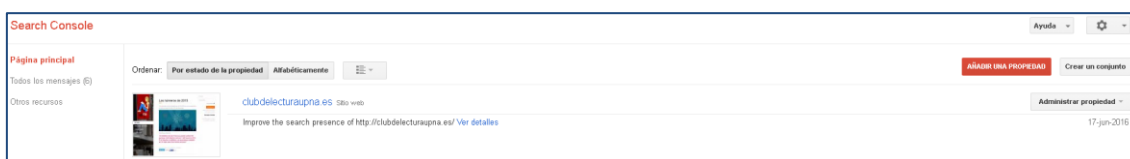


Ilustración 43. SEO. Google Webmasters

2. Para añadir un nuevo sitio web a la cuenta, previamente se deberá verificar la propiedad del mismo, de esta forma, Google comprobará quien es el propietario.
3. Para verificar el sitio web se deberá subir un fichero HTML, descargado de Google Webmasters Tools, a la raíz de la web.
4. Una vez verificada la cuenta, ya se podrán realizar acciones con el dominio.

8.2.4 Mapa del sitio con OSMaP

El mapa del sitio web(Sitemap) es un fichero donde están ubicadas las URL's del sitio web que han de indexar los buscadores. Existen dos tipos de sitemaps que motores de búsqueda como Google puede identificar e interpretar. La estructura de estos sitemaps puede estar en formato XML, RSS o Atom.

- XML: los sitemaps XML contienen todas las URLs de la página web.
- RSS/Atom: los feeds RSS/Atom sólo muestran los últimos cambios o las novedades de la web.

Google, motor de búsqueda que tomamos como referencia ya que es el más extendido en España, recomienda emplear ambos tipos, tanto los sitemaps XML como los feeds RSS/Atom, ya que estos últimos ayudarán a que los contenidos más nuevos se indexen rápidamente. Ambos sitemaps se pueden agregar en Google Webmaster Tools en la sección de sitemaps.

Los campos más importantes de los sitemaps son la URL y su fecha de modificación, ambos campos deben cumplir una serie de requisitos que detallamos a continuación.

8.2.4.1.1 Buenas prácticas para las URLs de los sitemaps:

- Incluir sólo URLs que existen y que son accesibles por el robot de Google.
- No incluir URLs que se han desautorizado en el archivo robots.txt.
- Incluir sólo URLs canónicas, ya que son las únicas que indexará Google. Al mismo tiempo evitar duplicar URLs dentro del sitemap.

8.2.4.1.2 Buenas prácticas para establecer la fecha de última modificación en un sitemap.

- Sólo se ha de actualizar la fecha de publicación de un artículo cuando el contenido ha cambiado significativamente. Hay que tener en cuenta que algunos plugins que se encargan de la generación del Sitemap de los Sistemas Gestores de Contenido actualizan la fecha aunque sólo modifiques el artículo para poner una coma.
- La fecha de última modificación debe tener el formato correcto:
 - XML: en este caso se emplea el formato W3C Datetime.
 - RSS: se emplea el formato especificado en la RFC822.
 - ATOM: formato especificado en la RFC3339.

8.2.4.1.3 Buenas prácticas para los sitemaps

XML

- Actualizar el sitemap al menos una vez al día, en caso de añadir contenido nuevo, y después hacer ping a Google.

1 de julio de 2016

- Maximizar el número de URLs que contiene un sitemap. Google tiene unos límites a la hora de procesar los sitemaps y sólo procesará sitemaps que contengan 50000 URLs y de un tamaño máximo (sin comprimir) de 10 MB.

Feeds RSS/Atom.

- Añadir al feed la URL de los contenidos nuevos y si se ha modificado considerablemente un artículo existente además de añadirlo al feed hay que modificar la fecha de actualización.
- Uso del protocolo PubSubHubbub para notificar a los agregadores de noticias y suscriptores sobre los nuevos contenidos publicados en tu web. El protocolo PubsubHubbub consiste en que el cliente simplemente está a la escucha, y es este protocolo quien le envía la información, lo que hace que la información aparezca en tiempo real, nada más publicarse

La generación de este documento la realizaremos de forma automática con la herramienta OSMMap.

Para consultarlo se deberá ir a Panel de control > Componentes > OSMMap y allí clicar en XML en el bloque “Sitemap Links”.



Ilustración 10. SEO. Sitemap links

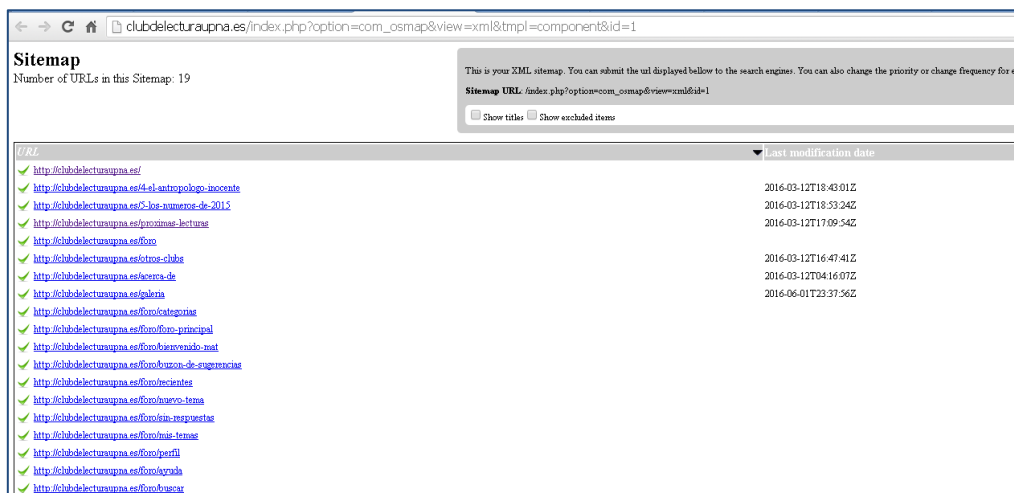


Ilustración 44. SEO. Contenido Sitemap

Con el mapa del sitio generado, se deberá copiar la URL de esta página, con el fin de añadir el mapa del sitio web en nuestra cuenta de Google Webmasters Tools. Es necesario realizar este paso, para indicarle a Google dónde se encuentra nuestro mapa del sitio web, y que pueda rastrearlo correctamente. Para hacer esto, primero se debe

1 de julio de 2016

que acceder a la cuenta de Google Webmasters Tools creada anteriormente e ir al sitio web anterior, y pinchar en la opción Rastreo > Sitemaps > Añadir o probar un sitemap.

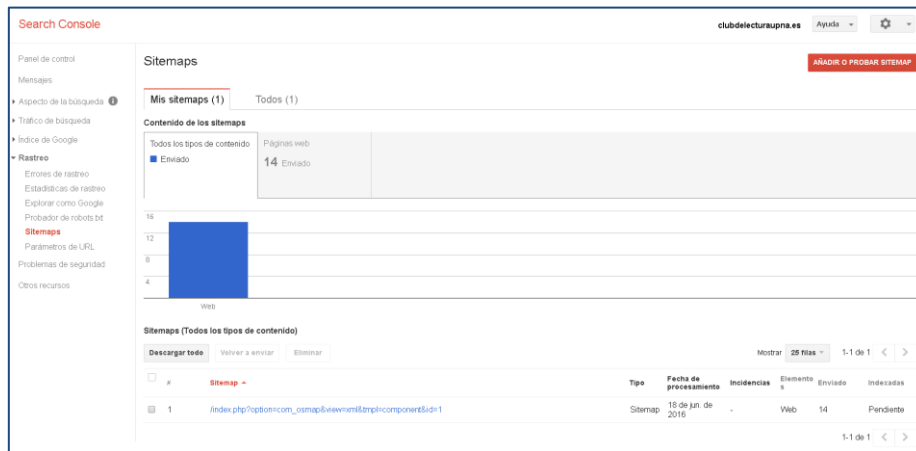


Ilustración 45.SEO. “Añadir o probar sitemap” en Google Webmasters Tools

Una vez hecho esto, se asegura Google rastreará la web e indexará las páginas, con lo que éstas aparecerán en los resultados de búsqueda.

8.2.5 Informes SEO con Google Analytics.

Para poder monitorizar el tráfico de la web y poder ver si las acciones llevadas a cabo para mejorar el posicionamiento están funcionando, se deberá dar de alta la web en Google Analytics con una cuenta de Google: <https://analytics.google.com/analytics/>

Una vez realizada el alta en Google Analytics será necesario añadir a la web el código que permite a Google hacer el seguimiento. Para ello se deberá ir al Panel de Control > Extensiones > Plantillas > Jp_rooms>Sttings>Opciones y pegaremos el código proporcionado por Google en “Additional Scripts”.

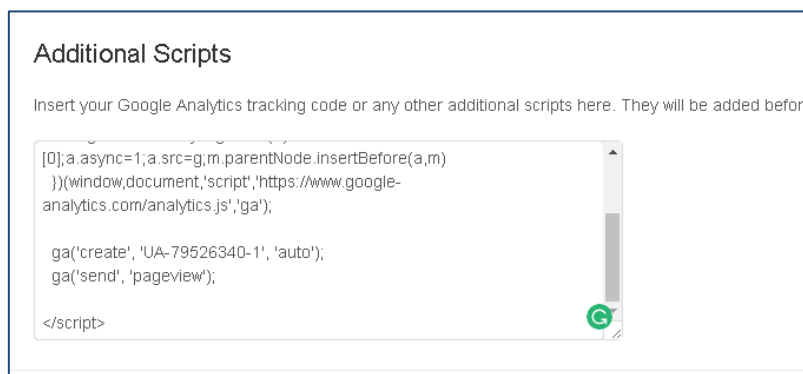


Ilustración 46. SEO. Añadir o probar un sitemap en Google Webmasters Tools

Google Analytics trae por defecto varios informes, entre ellos algunos relacionados con el posicionamiento. Estos informes están bajo *Adquisición*>Todo el tráfico>*Canales* podremos ver la distribución de fuentes de nuestro sitio. El tráfico puede ser:

1 de julio de 2016

- **Tráfico de búsqueda orgánico.** Cuando el usuario llega al hacer clic en un resultado de una búsqueda de buscador.
- **Tráfico de búsqueda de pago.** Cuando el usuario llega tras hacer clic en un anuncio en buscadores.
- **Tráfico directo.** El usuario llega directamente, tecleando la URL en el navegador por ejemplo, o a través desde marcadores.
- **Tráfico de referencia.** El usuario llega haciendo clic en un enlace desde otro sitio web.

Adquisición	Default Channel Grouping	Adquisición			Comportamiento			Conversiones	
		Sesiones	% de nuevas sesiones	Nuevos usuarios	Porcentaje de rebote	Páginas/sesión	Duración media de la sesión	Porcentaje de conversiones del objetivo	Consec de obj
Todo el tráfico		89.709	82,66 % % del total: 100,00 % (89.709)	74.151	56,04 % Promedio del sitio: 56,04 % (0,00 %)	3,58	00:02:10	0,00 %	0
1. Palabras clave	1. Organic Search	82.929 (92,44 %)	82,85 %	68.704 (82,65 %)	55,86 %	3,55	00:02:08	0,00 %	0
2. AdWords	2. Direct	5.955 (6,64 %)	80,64 %	4.802 (6,48 %)	58,89 %	4,11	00:02:39	0,00 %	0
3. Social	3. Referral	809 (0,90 %)	78,74 %	637 (0,86 %)	54,02 %	3,09	00:02:55	0,00 %	0
4. Optimización en buscadores	4. Social	16 (0,02 %)	50,00 %	8 (0,01 %)	43,75 %	3,50	00:04:06	0,00 %	0

Ilustración 47. SEO. Google analytics

El objetivo SEO estará enfocado en el tráfico orgánico, para ello encontraremos en “Adquisición” una sección específica para el tráfico de buscadores. Estos informes nos proporcionarán:

- Las palabras clave que los usuarios escribieron en el buscador para llegar a nuestro sitio web.
- Podremos ver qué visitas llegan desde estas palabras clave y que métricas tiene asociada cada palabra, como el porcentaje de rebote, tiempo medio de sesión, etc. De esta forma podremos saber qué palabras claves tienen mejor rendimiento y así potenciarlas.
- Se deberá asociar la cuenta de Google Webmaster Tools a Google Analytics, ello nos permitirá acceder a tres nuevos tipos de informes:
 - **Consultas de búsqueda**, con el CTR, clics, impresiones y posición media de cada consulta
 - **Páginas de destino**, con las mismas métricas que las consultas
 - **Resumen geográfico**, también con las mismas métricas y la información de los países que nos generan tráfico orgánico

1 de julio de 2016

8.2.6 Eliminar las URL's duplicadas

Las URL's duplicadas son URL's que apuntan al mismo contenido de la página. Esta duplicidad perjudica a la puntuación de la página en los análisis de los robots, por tanto conviene evitarlo en la medida de lo posible. Por ejemplo para evitar esta duplicidad en las peticiones de nuestra web sin www o con www se deberá forzar una redirección, normalmente, las webs son visibles tanto si se pone en el navegador el nombre del dominio con www o sin www, lo que provoca que toda la web esté duplicada.

Para evitar esto se deberá forzar una redirección, de contenido con www a sin o viceversa, para ello se debe añadir el siguiente código al inicio del fichero `.htaccess`, debajo de la activación del motor de redirecciones RewriteEngine On.

```
RewriteCond %{HTTP_HOST} !^clubdelecturaupna\.es  
RewriteRule (.*) http://clubdelecturaupna.es/$1 [R=301,L]
```

Esta regla sustituye “www.clubdelecturaupna.es” por el nombre seleccionado del dominio, y si se intenta acceder a cualquier dirección de la web con www, el servidor web redirigirá automáticamente a la misma dirección, pero sin www al inicio.

Esto es lo que se llama una redirección, y así nos evitaremos el problema de las URL's duplicadas debido al problema de las www.

El Google Webmaster Tools, se deberá marcar al buscador que utilice la URL sin WWW como dominio preferido, de esta manera se deberán realizar menos redirecciones.

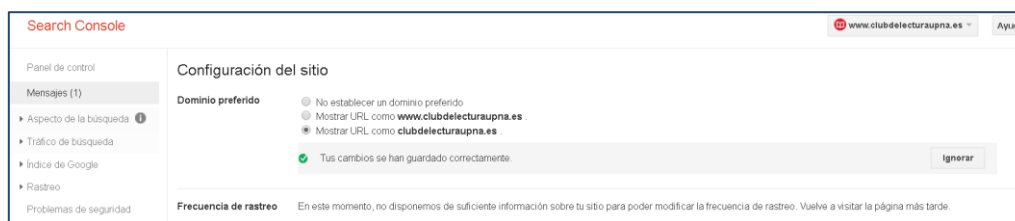


Ilustración 48. SEO. Google Webmaster Tools. Dominio preferido.

8.2.7 Contenido - Blog o sección de Noticias

Actualmente los robots de los buscadores dan mucha importancia al contenido de las páginas web y cada cuanto se actualiza este. Por ello se añadirá a la página, en la página principal, artículos que hablan de la temática de la página en formato Blog.

8.2.8 Etiquetado HTML

A la hora de redactar los contenidos de la web se debe tener en cuenta emplear correctamente el etiquetado HTML. Al crear/modificar un artículo, en el editor, se

1 de julio de 2016

podrán usar diversos encabezados (Encabezado 1 o h1, Encabezado 2 o h2, negrita o strong, etc), es importante utilizar estos encabezados y etiquetas HTML para los títulos de las subsecciones en las páginas, y que éstos contengan palabras relevantes y sean concisos.

El objetivo es que las palabras o frases que están incluidas en los encabezados y negritas de las páginas de nuestra web sean relevantes, porque los motores de búsqueda prestan especial atención a estas etiquetas HTML.

8.2.9 Peso de las imágenes de tus artículos

El tiempo que tarda una web en cargarse tiene relación directa con el tamaño que ocupa, esto es algo que los motores de búsqueda tienen en cuenta la hora del posicionamiento y uno de los puntos que más impactan es el peso de las imágenes. Una imagen sin optimizar en la web provocará un aumento significativo del tiempo de carga de la página, algo que puede ser perjudicial para el posicionamiento de la misma.

Algunos editores, como Gimp o Photoshop, disponen de modo optimizado al guardar, que nos permitirá guardar nuestras imágenes con un peso reducido, y manteniendo la calidad de las mismas. El peso deseable de las imágenes debe de ser de unos 15-90KB.

8.2.10 HTML5 y nuevas plantillas

Las páginas web que utilizan la tecnología HTML5 disponen de nuevas etiquetas que beneficiarán el SEO de las webs que las empleen. Gracias a ellas cada una de las partes de la página es interpretada por los motores de búsqueda como lo defina el desarrollador, con el uso de etiquetas como:

- header (cabecera de la web)
- menu (menú)
- article (cuerpo del artículo)
- etc.

A este concepto se le llama web semántica, porque permite que los motores de búsqueda entender la estructura de la web a través de las y dar importancia a ciertos contenidos sobre otros en la indexación.

Las nuevas plantillas de Joomla 3 utilizan HTML5 lo que permite disponer de estas características. Este etiquetado lo hace la plantilla internamente, y el redactor de

1 de julio de 2016

contenido únicamente ha de crear los artículos normalmente, y será la plantilla la que introduzca estas etiquetas que aportarán el significado semántico a los contenidos.

La plantilla utilizada en el proyecto siguen la filosofía de diseño web adaptable (RWD del inglés Responsive Web Design, es una filosofía de diseño y desarrollo cuyo objetivo es adaptar la apariencia de las páginas web al dispositivo que se esté utilizando para visualizarlas) .Esto permite que además de optimizar el SEO, el sitio web está adaptado para tabletas y teléfonos Smartphones.

9 Procesos de mantenimiento del sistema

A continuación se detallan los procedimientos estándar para asegurar el buen funcionamiento de la página y poder recuperarla en caso de ocurrir algún desastre.

9.1 Copia Automática

Para la implementación del proceso de copia automática se creará una tarea programada semanal que lanzará el proceso de copia de seguridad. Para ello se deberá acceder a la opción de menú Avanzado > Tareas Programadas y crear una nueva tarea, para ello se ha de informar en “Comando a ejecutar” el nombre del script a ejecutar y en “Opciones comunes” se deberá seleccionar la periodicidad semanal.



Ilustración 49. Configuración de Copia de seguridad automática

Al clicar en “Guardar” se creará la tarea programada y aparecerá en la lista de tareas programadas actuales, desde donde se podrán revisar los resultados de ejecución o eliminarla cuando ya no sea necesaria.

Este script llamará a una URL que realizará la copia de seguridad mediante la aplicación Akeeba Backup. Para activar esta opción se ha de ir al panel de control de Joomla, Componentes > Akeeba Backup y clicar en el botón “Opciones”. Una vez cargada la pantalla de opciones se deberá ir a “Front-end backup” y marcar la opción “Enable front-end and remote backup”, indicar una palabra secreta y rellenar los datos de envío de correos (Email address, Email Subject y Email Body).

En Componentes > Akeeba Backup >Schedule automatic backups se podrá consultar el comando a ejecutar para realizar la copia de seguridad de forma programada y que será ejecutado desde el script “backup.sh”.

Backup.sh

```
wget --max-redirect=10000  
"http://clubdelecturaupna.es/index.php?option=com_akeeba&view=Ba  
ckup&key=5q9TCw8T4GQKsjstIEGYszkLQF3ywAWR" -O - 1>/dev/null  
2>/dev/null
```

1 de julio de 2016



Ilustración 50. Configuración de Copia de seguridad automática

9.2 Copia Manual

Para realizar una copia de seguridad del sistema utilizará el componente Akeeba Backup, para ello se deberá acceder al panel de control de la Administración de Joomla e ir a Componentes > Akeeba Backup. En la pantalla que se cargará se deberá clicar a “Backup Now” y después al botón “Backup Now!”.

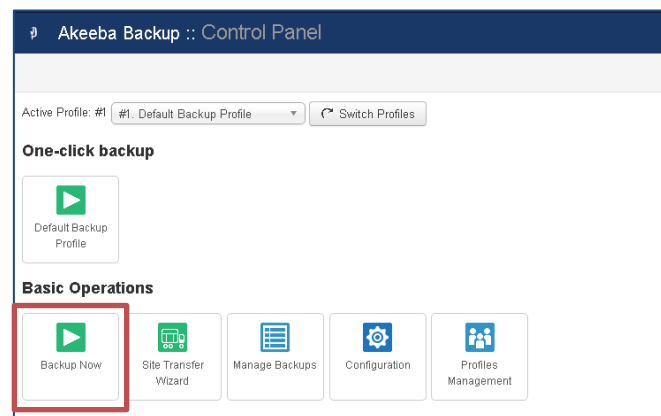


Ilustración 51. Copia de seguridad manual. Akeeba Backup.

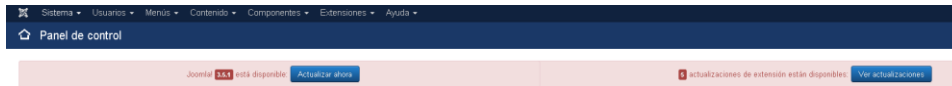
9.3 Restaurar copias de seguridad

Para restaurar una copia de seguridad se deberá acceder al panel de control de la Administración de Joomla e ir a Componentes > Akeeba Backup. En la pantalla que se cargará se deberá clicar a “Manage Backups”, en la pantalla que aparecerá se deberá marcar la casilla de la copia de seguridad a restaurar y clicar en el botón “Restore”. Desde el listado también se pueden descargar las copias de seguridad realizadas.

9.4 Actualización del gestor de contenido

El gestor de contenido Joomla tiene la característica de avisar en la pantalla por defecto del panel de control cuando hay actualizaciones de Joomla o de las extensiones instaladas.

1 de julio de 2016

**Ilustración 52. Actualizaciones en Joomla**

- **Actualización Joomla.** Clicando sobre “Actualizar ahora” accederemos a la pantalla que resume que versión tiene el sistema actual y cual tendrá y que permite mediante el botón “Instalar la actualización”, actualizar el sistema Joomla.
- **Actualización extensiones.** Clicando en “Ver actualizaciones” iremos a la pantalla con la relación de actualizaciones a aplicar y podremos marcar cuales de ellas aplicar seleccionando las casillas a su izquierda y clicando “Actualizar”.

10 Conclusiones

Del análisis de las opciones que hay en el mercado para la gestión de contenidos y de los conocimientos adquiridos en este proyecto he llegado a la conclusión que el uso de una plataforma de Gestión de Contenido ahorra esfuerzo a la hora de poner en marcha un nuevo sitio web, permitiendo focalizarse en el contenido a añadir. El hecho que sean plataformas tan extensibles hace que se puedan adaptar a multitud de necesidades tan solo usando extensiones ya creadas por la comunidad o la empresa que las mantiene.

En el caso concreto del sistema gestor de contenido usado, Joomla, se ha mostrado como una plataforma muy completa y madura que ha permitido poner en marcha la página del Club de lectura en muy poco tiempo, permitiendo centrarse en el desarrollo de la extensión a medida y en el aseguramiento de la calidad de la misma mediante los diversos tipos de pruebas definidos en la estrategia de prueba, con el objetivo de garantizar que delante de cualquier cambio en el código desarrollado, si este tiene una prueba que lo valida, se detectaría el defecto de forma temprana.

Actualmente hay multitud de recursos libres de uso que se pueden utilizar para multitud de cosas, entre ellas la creación de sitios web, por ello todos los programas y librerías usados en este proyecto son de libre uso, algunos además de código libre también.

En el futuro, para próximos proyectos en los que estuviera involucrado recomendaría el uso de un sistema gestor de conocimiento de código libre para la implementación de la solución, analizando siempre la opción más adecuada a la necesidad a cubrir, ya que cada uno de los gestores más utilizados tiene unas características que les hace cubrir mejor cada tipo de necesidad.

1 de julio de 2016

11 Glosario de términos

Acrónimo	Descripción
CMS	Content Management System
WYSIWYG	What You See Is What You Get
HTML	HyperText Markup Language
WAI	Web Accessibility Initiative
W3C	World Wide Web Consortium
RSS	Really Simple Syndication
PHP	PHP: Hypertext Preprocessor
IIS	Internet Information Server
SEO	Search Engine Optimization
GPL	General Public License
MVC	Modelo Vista Controlador
SQL	Structured Query Language
CSS	Cascading Style Sheets
FTP	File Transfer Protocol
CTR	Click Through Rate
RWD	Responsive Web Design

12 Bibliografía

Joomla! Documentation. Sitio web de Joomla! [En línea].

<https://www.joomla.org/3/es/>

QUnit Documentation, Sitio web de QUnit, [En línea], <https://api.qunitjs.com/>

PHPUnit Documentation, Sitio web de PHPUnit, [En línea],

<https://phpunit.de/manual/current/en/index.html>

How to evaluate a content management system Robertson, J., *Step two* [en línea],

http://www.steptwo.com.au/papers/kmc_evaluate/

¿So, what is a content management system?, Robertson, J., *Step Two* [en línea].

http://www.steptwo.com.au/papers/kmc_what/index.html

The gilbane report - The classification & evaluation of

Content management systems, Frank Gilbane, Bluebill Advisors, Inc., [Archivo pdf]

<http://gilbane.com/artpdf/GR11.2.pdf>

What is a Content Management System?, Jonah Winters, [en línea],

http://bahai-library.com/what_is_cms#3

1 de julio de 2016

13 Webs de consulta

CMS Usage Statistics, BuiltWith, [En línea].

<http://trends.builtwith.com/cms>

PHP-Documentación, [En línea].

<http://php.net/manual/es/intro-what-is.php>

Desarrollo de un Componente MVC/Desarrollo de un Componente Básico, Joomla! Documentation, [En línea].

https://docs.joomla.org/J3.x:Developing_an_MVC_Component/Developing_a_Basic_Component/es

CSS Syntax and Selectors, W3C, [En línea].

http://www.w3schools.com/css/css_syntax.asp

Search Engine Market Share 2015 by Country – Infographic, JimmyData, [En línea]

<https://jimmydata.com/blog/search-engine-market-share-country-infographic/>

Akeeba Backup User's Guide, Akeeba, [En línea].

<https://www.akeebabackup.com/documentation/akeeba-backup-documentation.html>