



ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES Y DE  
TELECOMUNICACIÓN

Titulación :  
INGENIERO TÉCNICO EN INFORMÁTICA DE GESTIÓN

Título del proyecto:  
*Recuerdos en papel: Diseño y elaboración de sitio web para  
promoción de empresa fotográfica*

Maitane Itoiz Lleida

Jesus Villadangos

Pamplona, 9 de Septiembre del 2011



# Agradecimientos

*Mediante estas líneas me gustaría darle las gracias a las siguientes personas:*

*Principalmente a mi familia, padres y hermana, por el continuo apoyo recibido y total confianza depositada sobre mi.*

*A Jesus Villadangos, por aceptar defender la idea y ayudarme a llevarla a cabo.*

*A Yon Garin, por brindarme la oportunidad de formar parte de este nuevo proyecto y depositar en mi toda su confianza y responsabilidad. Por todas esas horas invertidas y todas las que nos quedan.*

*A toda la gente de Torino, por dejarme compartir este maravilloso año con todos vosotros y por permitirme conseguir todos los logros deseados. Con especial mención a aquellos que habéis ayudado a que este proyecto se llevase a cabo.*

*A la comunidad de desarrolladores de Drupal, por su documentación y por hacer accesible su código a todos los demás.*

*A todos aquellos que, directa o indirectamente, habéis contribuido a sacar este proyecto adelante.*

*Mil gracias.*

## *Esker onak*

*Lerro hauen bitartez eskerrak eman nahi dizkiet ondorengo pertsoneri:*

*Familiari, guraso eta ahizpari bereziki, beti ondoan egon zaretelako eta nigan konfidantza osoa izan duzuelako.*

*Jesus Villadangos-i, ideia hau defendatzea onartzeagatik eta aurrera eramaten laguntzeagatik.*

*Yon Garin-i, proiektu berri honetan partaide izateko aukeratzeagatik eta bere konfidantza osoa nigan jartzeagatik. Eman dituzun hainbat orduengatik eta aurrean ditugun beste hainbatengatik.*

*Torino-ko jendeari, urte zoragarri hau zuekin disfrutatzeke aukera izan dudalako eta baita jarritako helburu guztiak lortzen laguntzeagatik. Alipamen berezia proiektu hau burutzen lagundu duzuen haiei.*

*Drupal-eko komunitateari, zure dokumentazioagatik eta zuen kodea guztion eskura jartzeagatik.*

*Zuzenean edo zeharka, proiektu hau aurrera ateratzen lagundu duzuen guztiei.*

*Mila esker bihotz bihotzez.*



# Índice de contenido

<b>Capitulo 1: Introducción.....</b>	<b>8</b>
<b>Capitulo 2 : Análisis de los requisitos .....</b>	<b>9</b>
<b>2.1. Objetivos generales del sistema.....</b>	<b>9</b>
<b>2.2. Funcionamiento y Rendimiento requerido .....</b>	<b>9</b>
<b>2.3. Definición de requisitos.....</b>	<b>10</b>
<b>2.3.1. Requisitos de software .....</b>	<b>10</b>
2.3.1.1. Gestión de la aplicación web mediante un Content Management System (CMS): Drupal.....	10
2.3.1.1.1. Requisitos de Drupal .....	10
2.3.1.3. Requisitos establecidos por el cliente.....	11
<b>2.3.2. Requisitos de hardware.....</b>	<b>13</b>
<b>Capitulo 3: Herramientas .....</b>	<b>14</b>
<b>3.1. Drupal vs Joomla .....</b>	<b>14</b>
<b>3.2. Drupal.....</b>	<b>14</b>
<b>3.2.1. Instalación sobre Mac .....</b>	<b>15</b>
3.2.1.1. XAMPP .....	16
3.2.1.2. Instalación.....	17
<b>3.2.2. Funcionamiento .....</b>	<b>20</b>
3.2.2.1. Módulos incluidos en el core.....	20
3.2.2.2. Módulos adicionales .....	22
3.2.2.2.1. Instalando módulos.....	22
3.2.2.2.2. Actualizar módulos.....	23
3.2.2.2.3. Módulos adicionales instalados .....	23
<b>3.2.3. Backup de la Base de datos y otras recomendaciones.....</b>	<b>24</b>
<b>Capitulo 4: Análisis y diseño .....</b>	<b>25</b>
<b>4.1. Interfaces según funcionalidades de la web .....</b>	<b>25</b>

4.2. Análisis de la base de datos .....	46
<b>Capitulo 5: Desarrollo .....</b>	<b>48</b>
5.1. jQuery .....	48
5.1.1. Nivo Slider .....	49
5.1.2. Colorbox .....	50
5.1.3. Script creados en JavaScript .....	51
5.2. Modificación de la base de datos .....	56
<b>Capitulo 6: Gestión del proyecto .....</b>	<b>59</b>
6.1. Ciclo de vida de los requisitos.....	59
6.1.1. Reunión N°1: definición de los requisitos.....	59
6.1.2. Reunión N°2: Enero del 2011 .....	60
6.1.3. Reunión N°3: Mayo del 2011 .....	61
6.1.4. Reunión N°4: Julio del 2011 .....	62
6.1.5. Reunión N°5: Agosto del 2011 .....	62
6.1.6. Reunión N°6: Agosto del 2011 .....	63
<b>Capitulo 7: Conclusiones .....</b>	<b>64</b>
<b>Capitulo 8: Líneas futuras .....</b>	<b>66</b>
<b>Capitulo 9: Bibliografía .....</b>	<b>67</b>
Referencia bibliográfica.....	67
Referencia en línea.....	67

# Capítulo 1: Introducción

Internet es una de las herramientas más potentes en cuanto a medios de comunicación actuales. La mayoría de las empresas cuentan con una página web para dar a conocer al mundo sus servicios. De esta manera, el mercado de una empresa no se limita a una ciudad, región o país, su mercado es ilimitado pudiendo alcanzar cualquier rincón del mundo y brindando la posibilidad de llegar a millones de clientes potenciales ubicados alrededor del planeta. Pero además de alcanzar a estos consumidores ubicados en distintas partes del mundo, permite a las empresas promocionar y ofrecer sus productos o servicios las 24 horas del día, los 365 días del año. Difundir información sobre un negocio, producto o servicios se convierte en una tarea un poco más fácil. Cualquier consumidor puede tener acceso a las características de los productos, precios, presupuestos, ubicación física o comunicarse rápidamente con la empresa mediante formularios de contacto, envío de correo directo o redes sociales.

De este modo, el cliente al que se le va a diseñar la página web, la empresa de fotografía “Recuerdos en Papel” quiere darse a conocer más allá de los ámbitos locales y ha optado por lanzarse la mundo del world wide web.

Para ello requieren de una página web que permita a los usuarios conocer sus productos mediante un sencillo click, formas de trabajar, visualización de trabajos realizados... así como poder realizarse sus propios presupuestos o tener acceso a sus reportajes fotográficos ya realizados con anterioridad desde cualquier lugar.



## Capítulo 2 : Análisis de los requisitos

### 2.1. Objetivos generales del sistema

El cliente desea crear una aplicación web que le permita mostrar al público en general sus productos. Mediante la página web, quiere dar a conocer cuál es su método de trabajo así como varios ejemplos sobre trabajos realizados. Además de esto, desea permitir a usuarios registrados poder acceder a los proyectos que han realizado con la empresa y poder verlos allá donde estén.

Por otro lado, también desea permitirle al usuario que cree él mismo vía online un presupuesto inicial con todo aquello que deseen y poder hacérselo llegar a la empresa para que, posteriormente, sea la empresa la que se ponga en contacto con el consumidor.

### 2.2. Funcionamiento y Rendimiento requerido

Se desea que la aplicación cuente con un diseño profesional que proyecte la imagen seria de la empresa pero que al mismo tiempo sea atractiva, a tal punto que capte la atención del usuario y lo incite a visitarla.

Además de esto, la navegación de nuestra página web debe ser sencilla, simple y clara de tal modo que el usuario pueda encontrar fácilmente lo que busca y, a la vez, debe ser rápida (debe cargar rápidamente), de modo que evitemos que el usuario pueda decidir irse por falta de paciencia.

## 2.3. Definición de requisitos

Desde un principio se dejó claro que cliente y desarrollador estarían continuamente en contacto para hacer un desarrollo conjunto de la aplicación. En este caso, tanto cliente como programador se encontraban un poco desorientados en una etapa inicial. El cliente tenía una idea general de lo que quería mostrar al público pero le era difícil concretarla y tampoco sabía si era posible desarrollar tal idea al completo. El desarrollador sugirió una serie de proposiciones para llevarlo a cabo, pero era necesaria una amplia documentación en el tema.

Bien es sabido que un desarrollo de software<sup>[6]</sup> depende totalmente de los requisitos del cliente. Por tanto, se ha optado por un desarrollo iterativo. La idea principal es producir frecuentemente prototipos que tengan un subconjuntos de los rasgos requeridos y para completar dicha decisión, se le presentará al cliente periódicamente el progreso de la aplicación de modo que, en caso de surgir problemas o desacuerdos, puedan solucionarse a tiempo.

### 2.3.1. Requisitos de software

#### 2.3.1.1. Gestión de la aplicación web mediante un Content Management System (CMS): Drupal

Desde la universidad se propuso la opción de desarrollar la aplicación mediante un CMS y tras debatir entre dos de ellos (Drupal y Joomla), se optó por utilizar el primero. Pese a que la curva de aprendizaje de Drupal es más extensa que la de Joomla, finalmente se decidió este CMS porque ofrece al desarrollador una flexibilidad y potencia superiores a las que ofrece Joomla. Módulos como Views, CCK, Panels, etc le dan a Drupal una enorme ventaja (más adelante hablaremos más específicamente sobre cada uno de ellos).

##### 2.3.1.1.1. Requisitos de Drupal

###### × **Base de datos**

Drupal, para su propia instalación, requiere de una base de datos relacional. Para ello, podemos valernos de PhpMyAdmin o sino, crear la base de datos directamente desde la línea de comandos de MySQL o PostgreSQL. Esta base de datos tiene que tener, a parte de un nombre, un usuario y contraseña que serán pedidos más adelante durante la instalación de Drupal. En nuestro caso, como comentaremos más adelante, utilizaremos PhpMyAdmin.

###### × **Apache**

Es un servidor web HTTP de código abierto. Implementa el protocolo HTTP y la noción de sitio virtual. Es usado principalmente para enviar páginas web estáticas y dinámicas en internet. Apache es el componente de servidor web en la popular plataforma de aplicaciones LAMP, junto con MySQL

y los lenguajes de programación PHP/Perl/Python. Mac OS X integra apache como parte de su propio servidor web. En nuestro caso, en vez de configurar los componentes que trae el sistema operativo de Mac integrado, hemos optado por utilizar uno de esos paquetes, más concretamente XAMPP (hablaremos de ello más adelante).

× **PHP**

Además de los mencionados anteriormente, también precisamos de un intérprete para lenguaje de script como es PHP<sup>[3]</sup>. Como ya hemos mencionado en el punto anterior, en nuestro caso hemos optado por la opción de instalar XAMPP que también incluye uno de estos intérpretes por lo que no tenemos que configurar ninguno de los apartados de PHP de Mac OS X.

### **2.3.1.3. Requisitos establecidos por el cliente**

En primer lugar, y por petición del cliente, el sitio debe estar claramente dividido en cinco apartados: Inicio, Galería, Clientes, Presupuestos y Blog.

× **Inicio**

La página principal del sitio tendrá un banner que sirva como primer contacto entre el usuario y los servicios ofrecidos por nuestro cliente. Dicho banner ocupará gran parte de la página y constará de cinco fotos que irán pasando periódicamente sin que el usuario tenga que pulsar ningún botón. En caso de querer avanzar o retroceder, también podrá realizarse mediante las flechas que se encuentran a la derecha y a la izquierda del banner. Además, clickando sobre una de las imágenes concretas, el sistema nos redirigirá a la galería en cuestión. La página de inicio también tendrá un apartado en el que le dará la bienvenida al cliente y, en la parte derecha de la pantalla, aparecerá una columna en la que la empresa nos permite acceder a su perfil en las diferentes redes sociales que tan de moda están actualmente. Más concretamente podemos acceder a Facebook, Twitter, Flickr y Blogspot.

× **Galería**

Desde este apartado se le quiere mostrar al cliente los servicios ofrecidos por la empresa. Dicho apartado constará de cinco subapartados que serán *bodas*, *conciertos*, *viajes*, *fotos de estudio* y *otros*, apartado en el que se le mostrará al consumidor otro tipo de trabajos realizados por los fotógrafos. Cada subapartado, contendrá una galería de fotos en la que se podrán ver, primero en miniatura y luego una a una, todas las fotos. Además de esto, en los apartados *conciertos* y *bodas*, las distintas galerías estarán claramente diferenciadas por el nombre del grupo al que pertenece el concierto o el nombre de la pareja que ha contraído matrimonio.

× *Clientes*

En el apartado *clientes*, el usuario de la página web tendrá acceso a la página que le permita iniciar sesión o recordar su contraseña en caso de haberla olvidado. En ningún momento se le permitirá al usuario registrarse, ya que esto correrá de la mano del administrador de la página web. Una vez loggeado, el usuario podrá tener acceso a diversas páginas como por ejemplo, podrá acceder a un apartado que le permita ver todos los reportajes de fotos que ha realizado con la empresa. Si así es querido, se le proporcionará un usuario y contraseña que puede ser distribuido a diferentes personas afines a ellos para que, de este modo, puedan disfrutar de todas las fotos ellos también. En un principio esta opción sólo estará activa para los reportajes de bodas. Por otra parte, se le dará opción al cliente de ver todos los presupuestos que ha pedido.

Una vez terminada la página web, será la encargada de administrar todo el tema de programación pero nada relacionado con reportajes, usuarios, entradas de blog o respuesta de presupuestos. Por ello, tenemos que crear un apartado de *administración* que le permita a nuestro cliente tener control sobre estas acciones. De este modo, crearemos un menú que le facilite el trabajo. Dentro de dicho menú podremos acceder a las entradas de blog, crear nuevas galerías de fotos, administrar usuarios y crear nuevos presupuestos en respuesta a las solicitudes recibidas.

× *Presupuestos*

Mediante un formulario, se le dará la oportunidad al cliente de poder confeccionar él mismo un presupuesto inicial. El cliente deberá introducir sus datos personales (*nombre, apellidos, teléfono de contacto y dirección de correo electrónico*) así como el tipo de reportaje que desea realizar. Este presupuesto quedará almacenado en la base de datos y nuestro cliente recibirá una notificación de que un nuevo presupuesto ha sido almacenado.

Como ya hemos comentado, nuestro cliente también tendrá una parte de presupuestos dentro de su menú. Desde aquí podrá responder a esas solicitudes de presupuestos que le han llegado desde los usuarios de la página. Los datos serán extraídos de una relación de tablas de la base de datos y se le mostrarán automáticamente por pantalla, haciéndole un baremo inicial del importe del reportaje solicitado. El fotógrafo podrá hacer modificaciones sobre el presupuesto (como por ejemplo añadirle un descuento) y una vez establecido el precio final será almacenado en una tabla en la base de datos. Este presupuesto se lo haremos llegar al cliente mediante correo electrónico o pactando una cita con él para hablarlo personalmente.

× *Blog*

La empresa desea tener un blog en el que poder ir colgando diferentes post para compartir con el usuario de la página trabajos, viajes, ideas, momentos de ocio... Este apartado contará con la estructura típica de un blog de contenidos. Cualquier navegante que visite la web podrá comentar las diferentes entradas posteadas sin tener la necesidad de loggarse ni tampoco ser usuario registrado.

Además de estos apartados, tendremos otros como por ejemplo una página con información sobre la empresa y un formulario de contacto que permita al usuario contactar con la misma.

En cuanto al *lado administrador*, en un principio será el desarrollador de la página web el que se encargue de administrar contenidos, privilegios y demás, por lo que no es preciso desarrollar una vista para dicho trabajo ya que el theme de drupal que instalemos nos permite realizar todas y cada una de estas tareas.

### **2.3.2. Requisitos de hardware**

Como la página va a estar expuesta al mundo en el amplio mundo de internet, precisamos de un servidor donde poder alojar nuestros datos. Además de esto, necesitaremos un dominio es decir, un nombre único a través del cual se identifica nuestro sitio.

## Capítulo 3: Herramientas

### 3.1. Drupal vs Joomla

En un principio se pensó en utilizar como CMS Joomla. La razón principal de utilizar este programa era el previo uso del mismo en las clases teóricas de la universidad. Mientras realizaba el segundo curso de la carrera he tenido la posibilidad de utilizarlo para realizar un pequeño proyecto del laboratorio de sistemas operativos y por lo tanto tenía cierto conocimiento sobre él.

Sin embargo, me he documentado sobre estos dos programas antes de empezar a desarrollar las ideas y he decidido finalmente utilizar Drupal. Comunidades expertas en sistemas gestores de contenidos exponían Drupal como un programa significativamente más potente y mucho más flexible que Joomla. Pese a que la mayoría añadía que para conseguir un sitio web profesional con Drupal hay que invertir mucho tiempo, su amplia comunidad de desarrolladores inclinaba la balanza sobre este lado indiscutiblemente.

Además, se había lanzado al mercado la nueva versión de Drupal (*Drupal 7*) que agrupaba todos los módulos importantes (antes adicionales pero totalmente imprescindibles) dentro del core haciéndolo más potente que Joomla. Por tanto, se decidió utilizar Drupal pese al nulo conocimiento del mismo.

### 3.2. Drupal

Drupal<sup>[1][2]</sup> es un sistema de gestión de contenidos (CMS, acrónimo de *Content Management System*) para la web. Se basa en una base de datos relacional donde almacena el texto y los enlaces entre el contenido. Las páginas se construyen dinámicamente para ser presentadas al usuario en respuesta a una "petición web" mediante un navegador.

El contenido en Drupal se almacena como *nodos*. Un nodo es un objeto de contenido genérico, que se puede corresponder con una "página", un "artículo", una "entrada de blog", etc... A algunos tipos de nodo, dependiendo de la configuración, los usuarios pueden añadir comentarios (los comentarios

no son considerados como nodos). Los nodos o los comentarios pueden necesitar ser aprobados por un moderador antes de ser publicados.

La presentación y el diseño de las páginas son gestionadas mediante *temas* intercambiables. La presentación estándar de una página de Drupal ('Tema'), consiste en tres columnas. La columna del centro es la "Columna de contenido". La más típica muestra resúmenes de los nodos publicados más recientemente por orden de fecha. Si hace click en uno de los resúmenes, el contenido completo del nodo es mostrado en la columna central.

Las columnas izquierda y derecha se llaman normalmente "barras laterales". Las barras laterales pueden mostrar *bloques* o información relacionada. Los bloques a menudo contienen enlaces para navegar hacia otros nodos. Por ejemplo, puede haber bloques mostrando los últimos artículos enviados o los más populares. Para nuevas instalaciones, se muestran el bloque de inicio de sesión y el bloque de navegación que contiene un menú de acciones que se pueden llevar a cabo. Diferentes menús pueden ser mostrados en el bloque de navegación, dependiendo de lo que esté haciendo y de qué privilegios o permisos ("roles") tenga el usuario.

Los bloques también pueden aparecer o no dependiendo de en qué parte del sitio web esté y de qué acciones está realizando, y también de los privilegios que tenga. Por ejemplo, el bloque de inicio de sesión no aparece si la sesión ya ha sido iniciada, o el de "artículos más recientes" puede no mostrarse si no hay artículos disponibles. El administrador puede habilitar/deshabilitar distintos bloques siguiendo las opciones de *Estructura >> Bloques*.

Los nodos se pueden clasificar en categorías, también llamadas *taxonomías*. Los foros son un ejemplo de nodos de contenido organizados por categoría. Las categorías pueden ser jerárquicas lo que significa que una categoría "padre" contiene múltiples categorías "hijos".

A menudo, nuevas funcionalidades son implementadas en Drupal como *módulos*. Una vez que un administrador añade un archivo o carpeta de un módulo en el subdirectorio "módulos", la opción de utilizar el módulo aparece en la sección Módulos. Si el administrador activa el módulo, las funcionalidades asociadas con él se activan y aparecen en la sección correspondiente de Drupal. Un módulo puede definir un nuevos tipos de nodos, crear nuevas opciones en el menú o proporcionar nuevos bloques que estarán disponibles para mostrar en las barras laterales.

### 3.2.1. Instalación sobre Mac

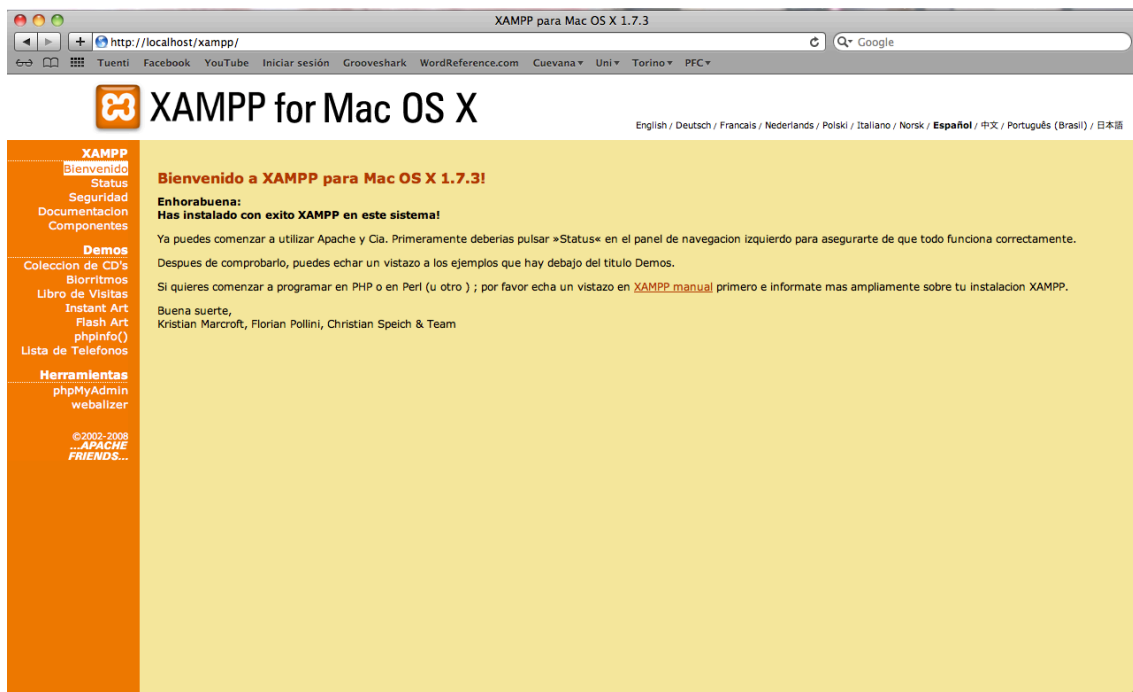
Como ya hemos comentado en el apartado *Requisitos de Drupal*, para correr un sitio Drupal se necesitan los siguientes programas y herramientas: Apache, PHP y MySQL. Mac OS X por defecto trae un servidor apache que podríamos utilizar y complementarlo con la instalación de MySQL y Php. Sin embargo, hemos optado por la opción de instalar un paquete que permita instalarlos los tres, XAMPP.

### 3.2.1.1. XAMPP

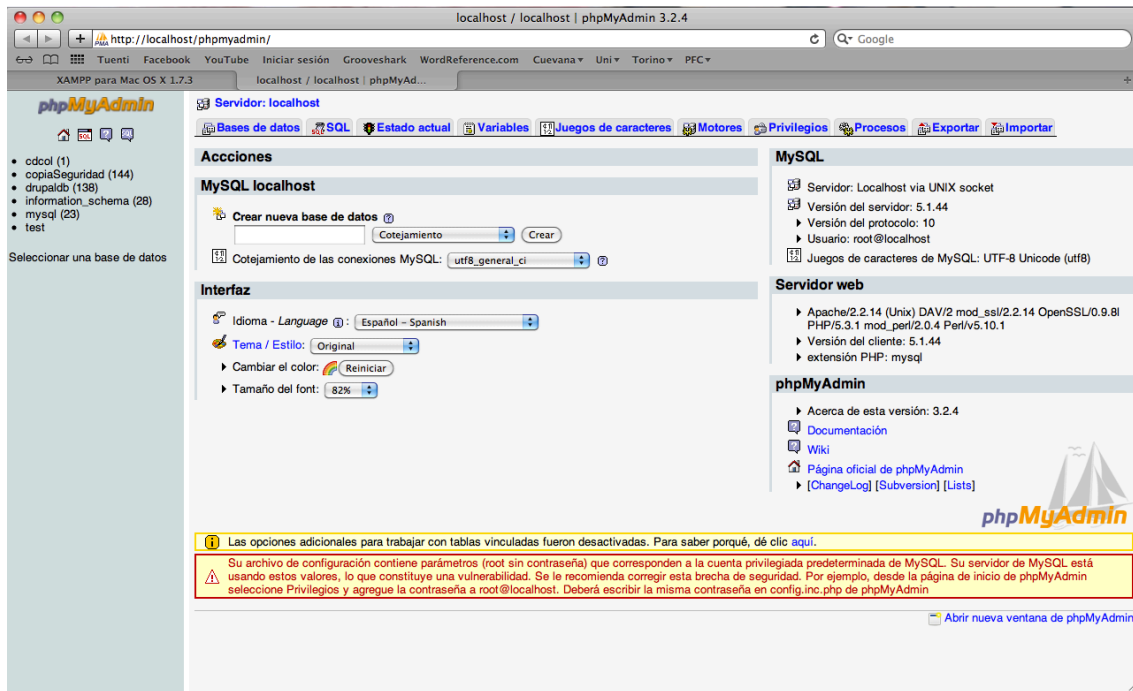
Es un paquete de código libre y gratuito. Consiste básicamente en un servidor HTTP de Apache, MySQL e intérpretes para scripts de PHP y lenguajes de programación Perl. El programa se distribuye bajo términos de la licencia de GNU y actúa como servidor web libre, capaz de servir páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, Linux, Solaris y Mac OS X y se utiliza principalmente para proyectos de desarrollo web. Este software es útil mientras se está creando páginas webs dinámicas utilizando lenguajes de programación como PHP o JavaScript.

Requiere únicamente de un archivo .zip, .tar o .exe que puede ser descargado desde <http://www.apachefriends.org/en/xampp-macosx.html>. Una vez descargado, no hay que realizar ningún cambio sobre nuestro sistema operativo. Es actualizado regularmente para incorporar las últimas actualizaciones de Apache/MySQL/Php y Perl. Módulos como OpenSSL o PhpMyAdmin también vienen incluidos dentro del paquete. Instalar XAMPP conlleva mucho menos tiempo que instalar cada uno de los componentes por separado. Por eso se convierte en una herramienta muy potente.

Cuando ya tenemos la aplicación instalada en nuestro sistema, sólo tendremos que ejecutarla y hacerla correr insertando el usuario y contraseña de nuestro sistema. Lo mismo tendremos que hacer en caso de que queramos pararlo.





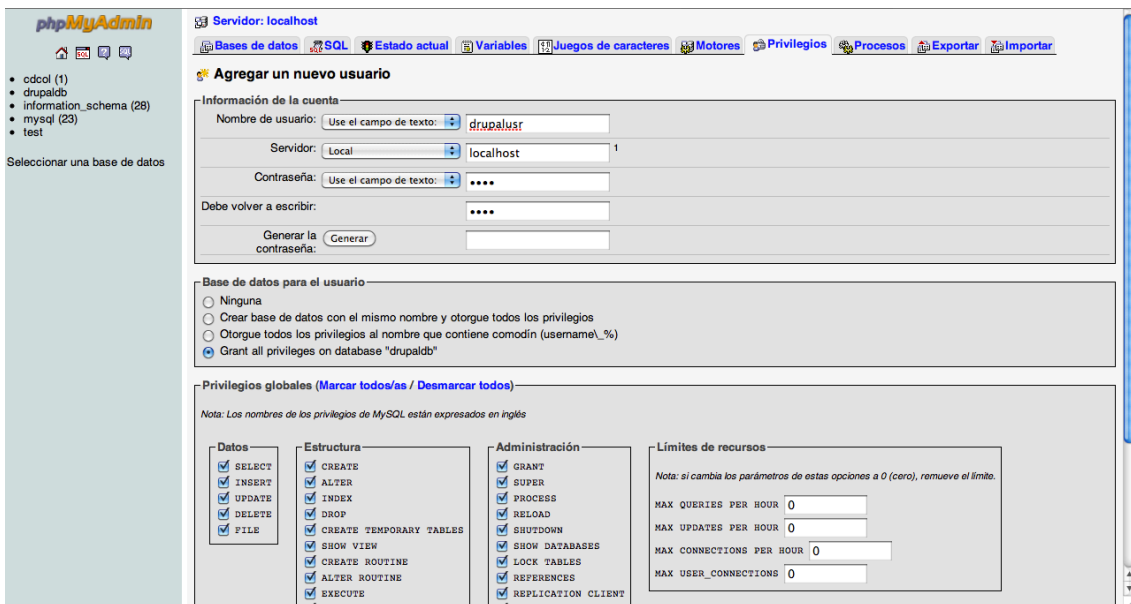
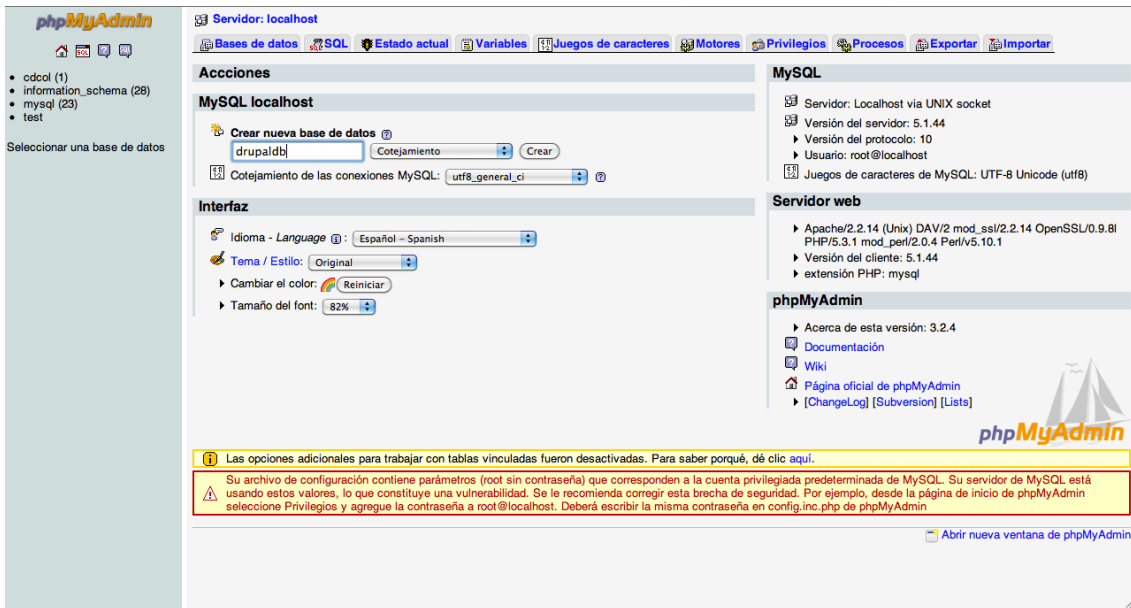


### 3.2.1.2. Instalación

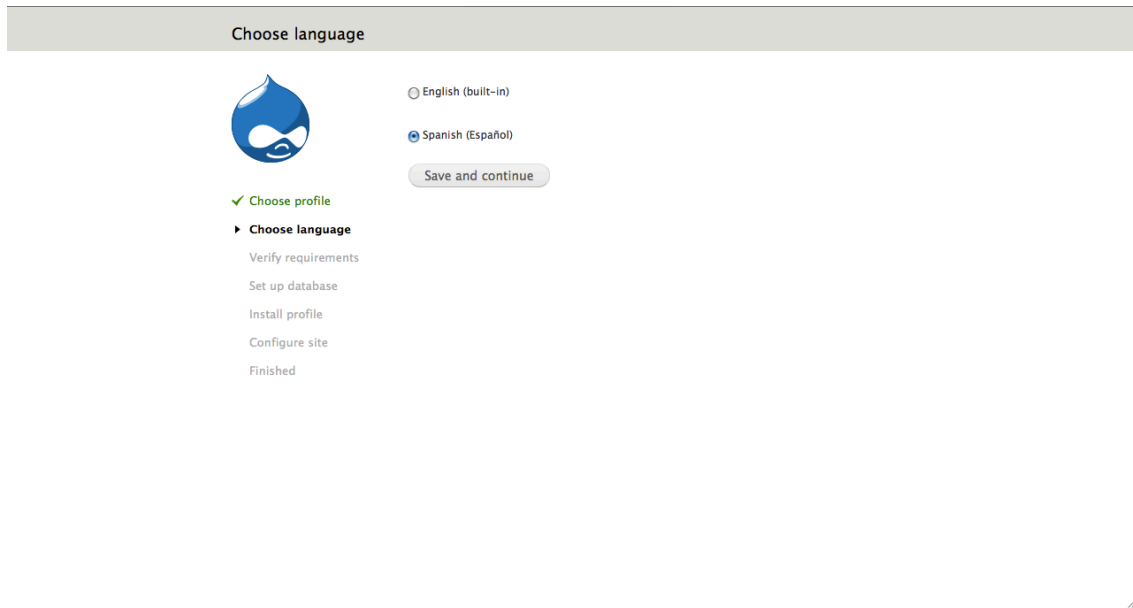
Lo primero que tenemos que hacer para instalar drupal es descargarlo. Podemos descargar el paquete de instalación (.tar o .zip) directamente desde [www.drupal.org/download](http://www.drupal.org/download). La última versión que ha salido al mercado es la 7 por lo que procederemos a descargar esta. Muchos de los módulos que antes había que añadirlos manualmente una vez tuviésemos Drupal instalado en nuestro sistema, han pasado a ser parte del core.

Una vez que tenemos el paquete de instalación descargado, lo descomprimos e insertamos la carpeta que se nos ha creado dentro de la carpeta *htdocs* de nuestro XAMPP. En nuestro caso estará en la ruta *Aplicaciones/XAMPP/htdocs*. Antes de iniciar ninguna instalación, debemos crear la base de datos donde se nos copiarán todos los archivos de drupal. Como el paquete XAMPP trae PhpMyAdmin, accedemos a dicha página y creamos la base de datos. Crearemos también un nuevo usuario y contraseña y le otorgaremos a este usuario todos los permisos. En nuestro caso será:

*Nombre de la bdd:* drupaldb  
*Usuario:* drupalusr  
*Pass:* root



Creando la base de datos, procederemos a la instalación. Desde nuestro explorador accedemos a la carpeta que hemos creado (<http://tusitio/nombredelacarpeta>) y se nos lanzará el script de instalación. En nuestro caso será <http://localhost/recuerdos>. Lo primero que tendremos que hacer es seleccionar el perfil. Elegiremos standard. Luego, nos dará la opción de decir en qué idioma estará nuestro sitio, ofreciéndonos la posibilidad de traducirlo a cualquier otro idioma. Nosotros lo traduciremos a español.



Una vez traducida, nos pedirá que insertemos los datos de la base de datos creada anteriormente. En el campo nombre de la base de datos, usuario y contraseña introducimos los indicados un poco más arriba.



Cuando hayamos superado este paso, drupal pasará a instalar el perfil y configurar las traducciones y luego nos pedirá que configuremos el sitio que estamos creando. Tendremos que indicarle datos como *nombre del sitio*, *el usuario administrador*, *contraseña* y *correo electrónico* entre otros.

<p>✓ Configurar traducciones</p> <p>► Configurar sitio</p> <p>Terminar traducciones</p> <p>Terminado</p>	<p>Los correos electrónicos automáticos, como la información de registro, se enviarán desde esta dirección de correo. Use una dirección que tenga como terminación el dominio de su sitio para evitar que estos correos sean considerados spam.</p>
	<p><b>CUENTA DE MANTENIMIENTO DEL SITIO</b></p> <p><b>Nombre de usuario *</b></p> <input type="text" value="admin"/> <p>Se permite la utilización de espacios; los signos de puntuación no están permitidos a excepción de puntos, guiones altos y guiones bajos.</p> <p><b>Dirección de correo electrónico *</b></p> <input type="text" value="recuerdosmail@recuerdos.es"/> <p><b>Contraseña *</b></p> <input type="password" value="*****"/> <b>Fortaleza de la contraseña: Bueno</b> <p><b>Confirmar contraseña *</b></p> <input type="password" value="*****"/> Las contraseñas coinciden: sí <p>Para que su clave sea más robusta:</p> <ul style="list-style-type: none"> <li>• Añadir letras mayúsculas</li> <li>• Añadir signos de puntuación</li> </ul>
	<p><b>CONFIGURACIÓN DEL SERVIDOR</b></p> <p><b>País predefinido</b></p> <input type="text" value="España"/> <p>Elija el país predeterminado del sitio.</p> <p><b>Zona horaria predefinida</b></p>

De este modo, procederá con las instalación hasta el final sin que el usuario tenga que interferir en nada más.

### 3.2.2. Funcionamiento

En este apartado se procederá a explicar como se realiza la instalación/ actualización de los módulos.

#### 3.2.2.1. Módulos incluidos en el core

Como ya se ha comentado anteriormente, actualmente está en el mercado la versión 7 del core de Drupal. Esta versión incluye muchos de los módulos que antes había que instalar manualmente pero que eran totalmente imprescindibles. Por tanto, la comunidad de desarrolladores consideró adecuado introducirlos dentro del core. A continuación listamos todos los módulos que podemos encontrar por defecto que como ya están instalados, sólo es necesario habilitarlos o deshabilitarlos en función de si se quieren usar o no.

##### × *Módulos obligatorios*

- ✦ *Block*: Controla el montaje visual de los bloques con los que se construye una página.
- ✦ *Color*: Permite a los administradores cambiar el esquema de color de los temas compatibles con él.
- ✦ *Comment*: Permite a los usuarios comentar y discutir contenido publicado.
- ✦ *Field*: API para añadir campos a entidades tales como nodos y usuarios.

- ✦ *Field SQL storage*: Almacena los datos del campo en una base de datos SQL.
- ✦ *File*: Define un tipo de campo de archivo.
- ✦ *Filter*: Filtra el contenido preparándolo para presentación.
- ✦ *Locale*: Agrega la capacidad de gestionar idiomas y permite la traducción de la interfaz de usuario a idiomas que no son inglés.
- ✦ *Menu*: Permite a los administradores personalizar el menú de navegación del sitio.
- ✦ *Node*: Permite que se envíe el contenido al sitio y que se despliegue en páginas.
- ✦ *Options*: Define controles de selección, casillas de selección y botones para opciones para los campos de texto y números.
- ✦ *System*: Gestión de la configuración general del sitio por administradores.
- ✦ *Taxonomy*: Permite la categorización del contenido.
- ✦ *Text*: Define tipos de campo de texto simple.
- ✦ *User*: Administra el registro de usuarios y el sistema de inicio de sesión.

#### ✦ *Módulos opcionales*

- ✦ *Agregator*: Integra contenido sindicado (canales de noticias RSS, RDF y Atom).
- ✦ *Blog*: Activa los blogs multi usuario.
- ✦ *Book*: Permite a los usuarios crear y organizar los contenidos relacionados en un índice.
- ✦ *Contact*: Habilita el uso de formularios de contacto personal y a nivel de sitio.
- ✦ *Content translation*: Permite que el contenido sea traducido a diferentes idiomas.
- ✦ *Contextual links*: Proporciona enlaces contextuales para ejecutar acciones relacionadas con los elementos de una página.
- ✦ *Dashboard*: Proporciona una página de panel de control en la interfaz administrativa para organizar las tareas administrativas y monitorizar la información de su sitio.
- ✦ *Database logging*: Anota y registra eventos del sistema en la base de datos.
- ✦ *Field UI*: Interfaz de usuario para el API de campos.
- ✦ *Forum*: Proporciona foros de discusión.
- ✦ *Help*: Administra la presentación de la ayuda en pantalla.
- ✦ *Image*: Proporciona herramientas de manipulación de imágenes.
- ✦ *List*: Define tipos de campos de listas.
- ✦ *Number*: Define tipos de campos numéricos.
- ✦ *Overlay*: Presenta la interfaz administrativa de Drupal en una capa superpuesta.
- ✦ *Path*: Permite a los usuarios renombrar URL.

- ✦ *PHP filter*: Permite la evaluación de fragmentos de código PHP.
- ✦ *Poll*: Permite que su sitio capture votos sobre diferentes tópicos en forma de preguntas de opción múltiple.
- ✦ *RDF*: Enriquece su contenido con metadatos para que otros programas entiendan mejor sus relaciones y atributos.
- ✦ *Search*: Permite la búsqueda de palabras en todo el sitio.
- ✦ *Shortcut*: Permite a los usuarios gestionar las listas personalizables de enlaces de atajos.
- ✦ *Statistics*: Guarda estadísticas de acceso al sitio.
- ✦ *Syslog*: Anota y registra eventos del sistema en el syslog.
- ✦ *Testing*: Proporciona un entorno para ejecución de pruebas unificado y funcional.
- ✦ *Toolbar*: Proporciona una barra de herramientas que muestra los elementos de menú de administración y los enlaces de otros módulos.
- ✦ *Tracker*: Activa el seguimiento del contenido reciente para los usuarios.
- ✦ *Trigger*: Activa las acciones que serán disparadas en ciertos eventos del sistema, como cuando se crea un nuevo contenido.
- ✦ *Update manager*: Comprueba las actualizaciones disponibles y puede instalar módulos y temas de forma segura a través de una interfaz web.

### 3.2.2.2. Módulos adicionales

A parte de los módulos que podemos encontrar por defecto en el core, la comunidad de desarrolladores pone a nuestra disposición una gran cantidad de módulos adicionales que si se desean se pueden utilizar.

#### 3.2.2.2.1. Instalando módulos

Este otro tipo de módulos podemos descargarlos desde <http://drupal.org/project/Modules>.

Los módulos adicionales se distribuyen en una carpeta que incluye los archivos de cada módulo. Esta carpeta está en formato (tar.gz) y el equipo de desarrollo de Drupal recomienda que los módulos adicionales sean instalados en el directorio *sites/all/modules* en vez de instalarlos en el directorio *modules* del directorio de instalación de Drupal. Esto es así para facilitar la actualización de las versiones de Drupal (por ejemplo, de la versión 7.4 a la versión 7.7). Cabe destacar que la instalación de Drupal no incluye el directorio *sites/all/modules*. Sólo incluye la el directorio *all* dentro del directorio *sites*. Para instalar módulos adicionales es recomendable crear primero el directorio *modules* dentro del directorio *sites/all*.

Una vez creado el directorio, se descarga el módulo compatible para la versión que se tiene instalada y se copia en la carpeta mencionada anteriormente. Se inicia sesión en nuestro sitio con la cuenta de Administrador

y se va a la sección de módulos. En esa sección se puede ver que ya está listado el módulo copiado en la carpeta sites/all/modules pero aún no está instalado. Se debe activar primero y luego correr el script *update.php* dentro del sitio.

### 3.2.2.2. Actualizar módulos

Drupal libera actualizaciones de sus versiones cuando existen regularmente problemas de seguridad en alguna de sus versiones, incluyendo parches y actualizaciones de los módulos *core-opcional* y *core-obligatorios*. La forma más simple de actualizar es quitando la versión anterior e instalando la versión más reciente. Para esto, se borran los archivos de la versión que se quiere actualizar y se copian los archivos de la nueva versión.

Para actualizar módulos que no son parte de la instalación de Drupal se tiene que realizar una acción similar: se borra la carpeta del módulo a actualizar y posteriormente se copian los archivos (dentro de su carpeta correspondiente) del módulo que se actualiza a la carpeta (*sites/all/modules*). Posteriormente se ejecuta de nuevo *update.php* y al terminar de ejecutarse este script el módulo queda actualizado.

### 3.2.2.3. Módulos adicionales instalados

- × *Chaos Tool Suite (ctools)*: Es un conjunto de APIs y una herramienta para mejorar el trabajo del desarrollador. También contiene un módulo llamado *Page Manager* que consiste en gestionar páginas. En particular, maneja las páginas del panel, pero a medida que van creciendo, será capaz de manejar mucho más que solo los paneles. Podemos descargarlo en: <http://drupal.org/project/ctools>
- × *Date/Time*: Contiene campos de tipo fecha y de hora flexibles al uso en otros módulos. <http://drupal.org/project/date>
- × *References*: Define las características básicas comunes para los tipos diferentes de referencias como son las referencias a nodos y a usuarios. Para descargarlo, visitar la siguiente URL: <http://drupal.org/project/references>
- × *JQuery*: Actualiza la versión de jQuery del core de Drupal a una versión más nueva. [http://drupal.org/project/jquery\\_update](http://drupal.org/project/jquery_update)
- × *Wysiwyg*: permite editar contenidos con editores tipo lado-cliente. <http://drupal.org/project/wysiwyg>
- × *IMCE*: Cargador de imágenes/archivos y navegador que soporta directorios personales. <http://drupal.org/project/imce>
- × *Backup and migrate*: Permite hacer copias de seguridad o migrar la base de datos de drupal rápidamente y sin datos innecesarios. [http://drupal.org/project/backup\\_migrate](http://drupal.org/project/backup_migrate)
- × *Colorbox*: Plugin personalizable de *lightbox* para jQuery 1.3 y 1.4. <http://drupal.org/project/colorbox>
- × *Fill PDF*: Módulo que permite imprimir en pdf los resultados obtenidos de formularios. <http://drupal.org/project/fillpdf>

- × *LoginToboggan*: Este módulo realiza modificaciones en la página de login que viene por defecto con Drupal de modo que podamos redireccionar el login a la URL que queramos una vez iniciado sesión en el sistema o iniciar sesión con una dirección de correo electrónico en vez de un nombre de usuario.  
<http://drupal.org/project/logintoboggan>
- × *Libraries*: Permite el uso compartido de librerías externas.  
<http://drupal.org/project/libraries>
- × *Nice Menus*: Permite crear menús desplegables en Css/jQuery.  
[http://drupal.org/project/nice\\_menus](http://drupal.org/project/nice_menus)
- × *Token*: Proporciona una interfaz para el Api de Token y algunos “Tokens” que faltan en el core. <http://drupal.org/project/token>
- × *Captcha*: Permite añadir módulos captcha en nuestros formularios en modo de autenticación.  
<http://drupal.org/project/captcha>
- × *Views*: Crea listas personalizadas y consultas desde nuestra base de datos. <http://drupal.org/project/views>

Junto con este módulo se instala a su vez Views UI, que proporciona una interfaz administrativa al módulo Views. Si este módulo no se pueden crear o editar las vistas.

- × *Views Nivo Slider*: Se utiliza para mostrar una vista con Nivo Slider de jQuery. [http://drupal.org/project/views\\_nivo\\_slider](http://drupal.org/project/views_nivo_slider)
- × *WebForm*: Permite la creación de formularios y cuestionarios.  
<http://drupal.org/project/webform>

### 3.2.3. Backup de la Base de datos y otras recomendaciones

Es muy recomendable respaldar la base de datos para que, en caso de que algo grave sucediera, tuviésemos un respaldo del sitio en el estado anterior a la instalación/actualización de algún módulo. Entre los módulos adicionales que hemos instalado encontramos uno que nos permite hacer backups de modo seguro y de manera muy sencilla.

También es recomendable poner el sitio en *modo de mantenimiento* mientras se está haciendo la actualización o cuando estamos creando la copia de seguridad de nuestra base de datos para evitar posibles accesos al sitio o consultas que hagan que nuestra actualización se vea interferida. Para ello accedemos mediante la ruta <http://tussitio/user/?q=admin/config/development/maintenance> o accediendo por *configuración/modo de mantenimiento*.

Además de esto, se le recomienda al usuario leerse los manuales de los módulos que se vayan a instalar para ver si requieren de algún otro requisito o modificación sobre nuestro sistema.



## Capítulo 4: Análisis y diseño

### 4.1. Interfaces según funcionalidades de la web

La página contará con apartados claramente diferenciados teniendo en cuenta el tipo de usuario que accede a la misma. Por tanto, vamos clasificar los interfaces teniendo en cuenta los usuarios a los que van destinados y sus funcionalidades.



#### **Usuario anónimo**

##### **Inicio:** Página principal

Como ya hemos comentado anteriormente, la página principal cuenta con un banner central que es lo primero que capta la atención del usuario y que parte la página en dos. Encima de este tenemos el nombre de la empresa y un menú que nos permite acceder a las diferentes secciones de la página web. Bajo el banner, encontramos un texto de bienvenida a la página web y un menú que nos permite seguir a los fotógrafos (y empresa) en las diferentes redes sociales.



### ☀ **Galería:** Página de selección de galerías

Si pulsamos sobre el menú en el enlace *Galería* accederemos a esta página. Dicha página consta de 5 imágenes (las mismas que podemos encontrar en el banner) que nos redirigen a las galerías concretas. Si clickamos sobre *Bodas/Ezkontzak* nos llevará a otra página que nos mostrará los diferentes reportajes de bodas que hemos realizados.

Si por el contrario clickamos sobre *Conciertos/Kontzertuak* nos llevará a otra página en la que nos mostrará los nombres de los grupos que hemos retratado. *Viajes/bidaiak*, *Estudios* y *Otros/Besteak* nos llevan directamente a las galerías de fotos donde podemos ver una a una las fotos realizadas.

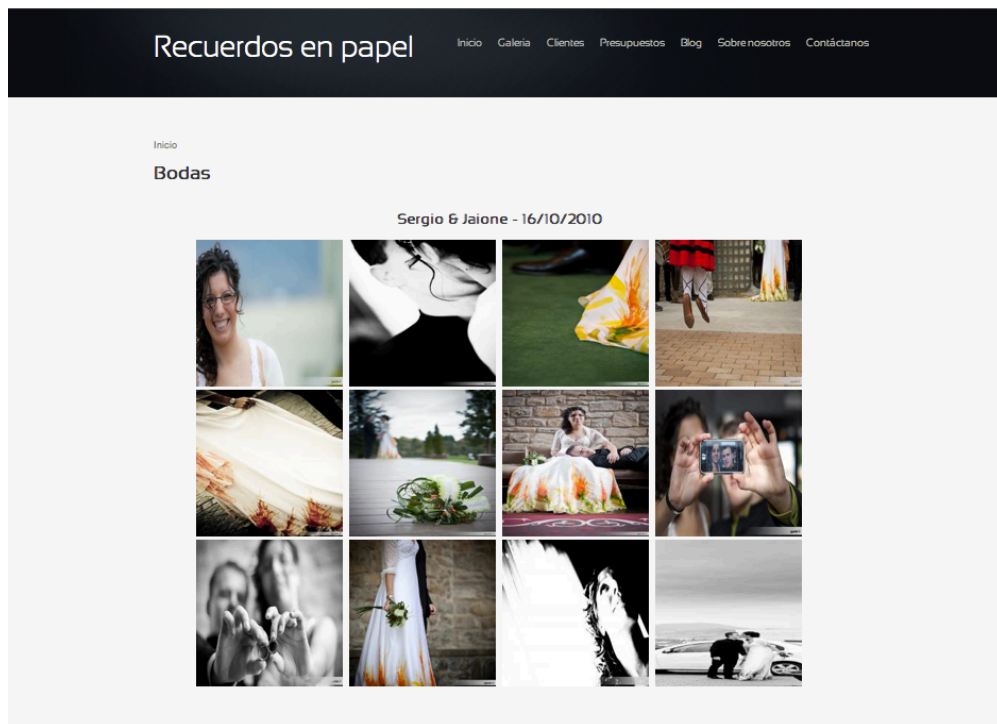


- ~ **Galería/Bodas:** Página de selección de reportaje de boda concreto.

Mediante una foto del reportaje realizado se nos permite elegir la galería de fotos que queremos ver. Será una selección de fotos de todas las realizadas el día del enlace.

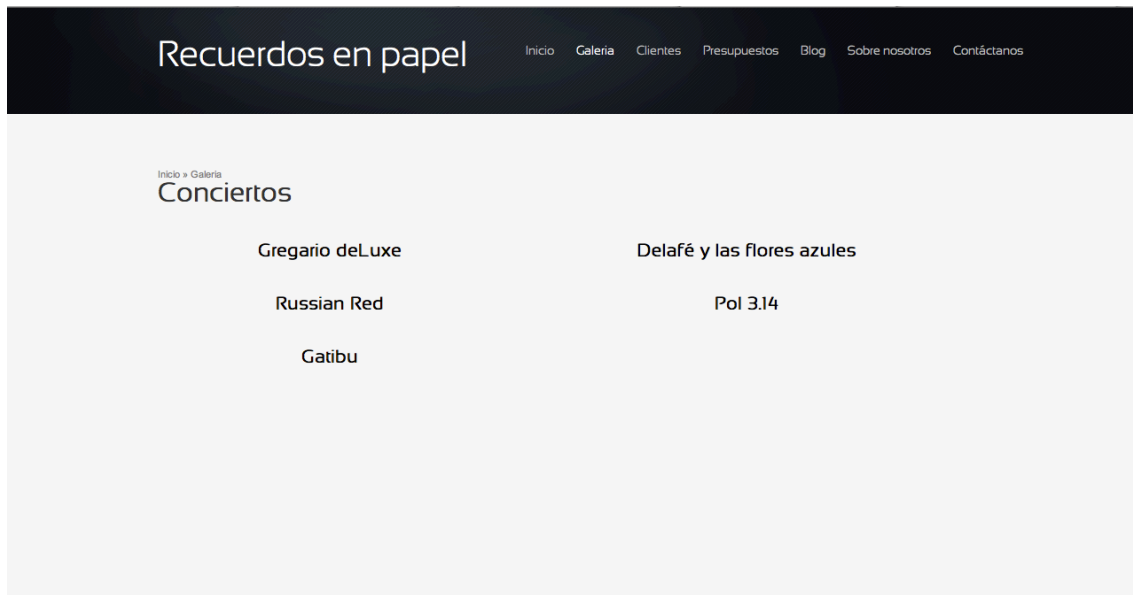


● **Galería de bodas**



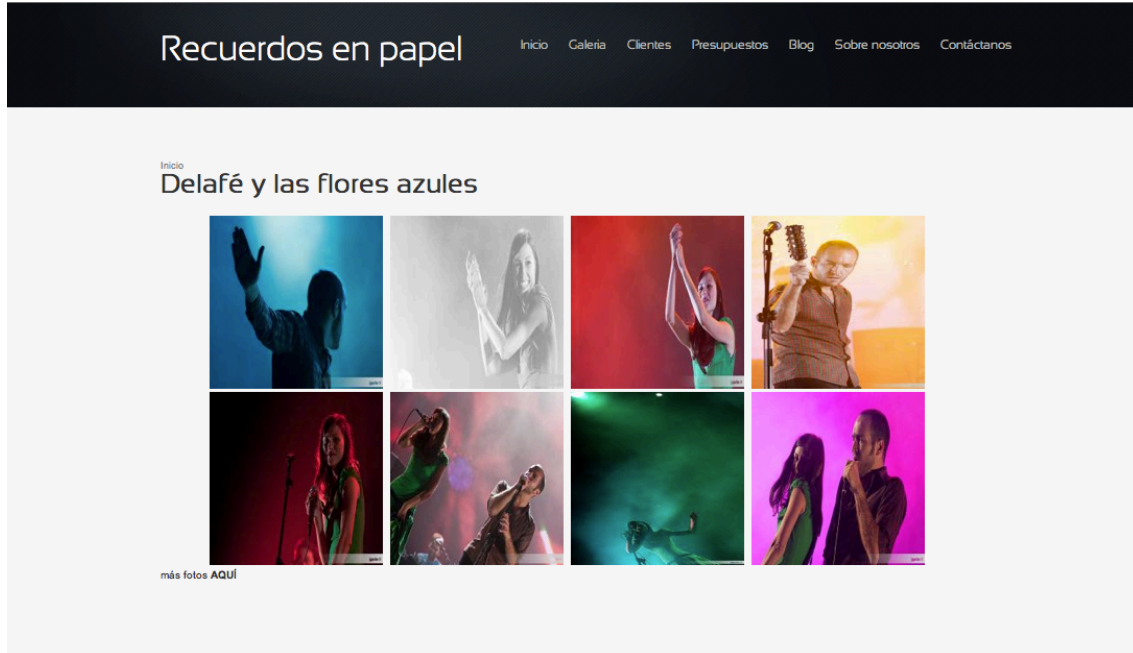
~ **Galería/Conciertos:** Página de selección del reportaje de un concierto concreto.

Si en vez de clickar sobre bodas, lo hacemos sobre *conciertos/kontzertuak* lo que obtendremos será una página con todos los nombres de los grupos a los que hemos retratados en sus respectivos conciertos. Clickando sobre el nombre de cada grupo, accederemos a la galería concreta.



● **Galería de conciertos**

Además de ver la fotos del grupo en concierto, debajo de todas las fotos y para enseñar todo el producto en su totalidad a aquel cliente que disponga de más tiempo o tenga mayor curiosidad, tenemos un link que nos permite acceder a la galería de fotos de dicho concierto en la página flickr.



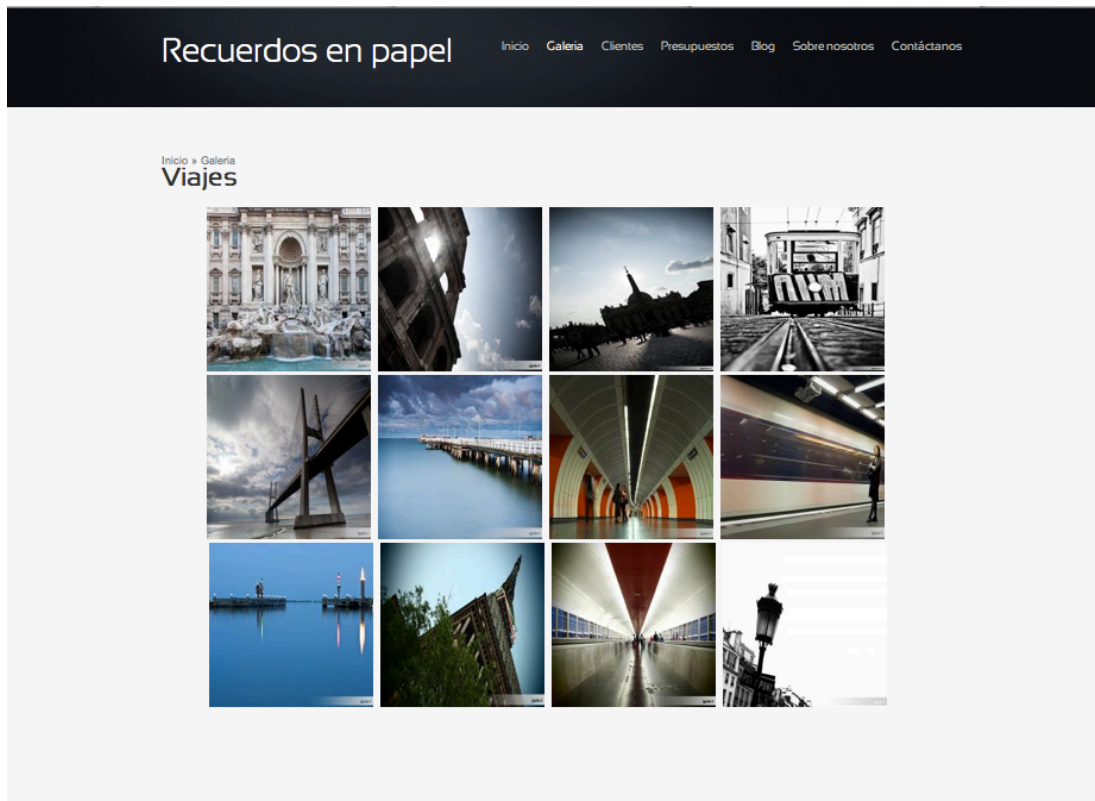
## ~ Galería/Estudio



Las fotos que podemos encontrar en esta galería son fotos realizadas a personas. No existe límite de edad ni ninguna otra característica concreta. Las fotos van desde bebés recién nacidos hasta gente joven o ancianos que quieren retratar el paso del tiempo o un momento de su vida en concreto, porque cualquier momento puede ser especial.

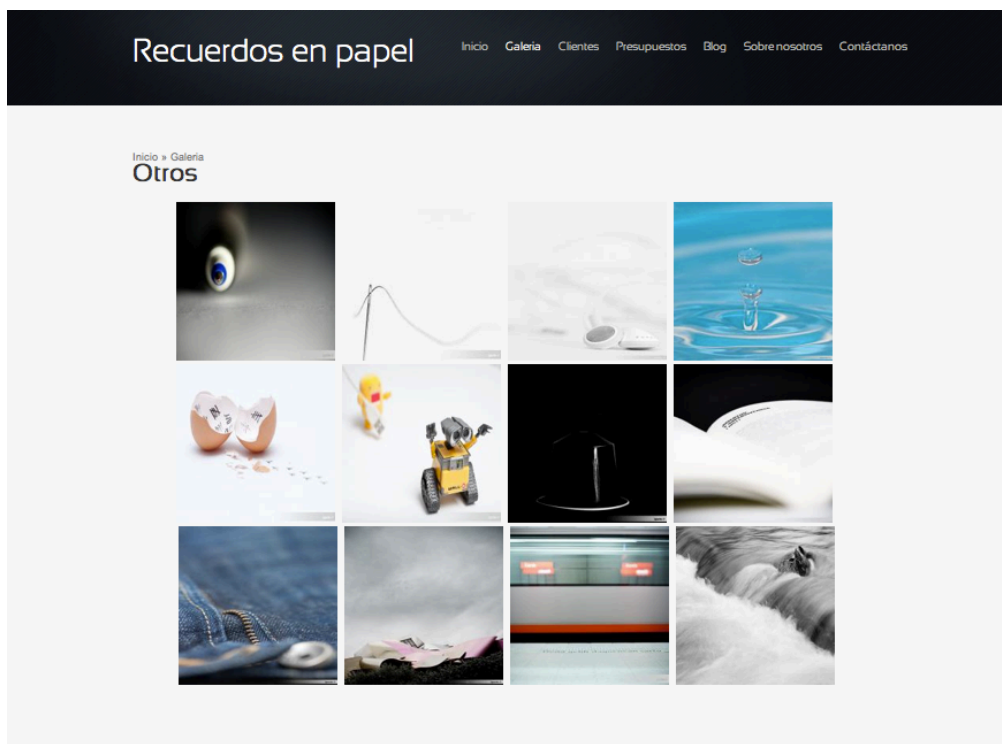
## ~ Galería/viajes

Dentro de esta galería la empresa quiere mostrarle al público las instantáneas tomadas durante los diferentes viajes realizados. Aunque dentro de su propuesta de servicios no esté incluido el paisajismo, no se descarta el poder realizar reportajes que incluyan el medio ambiente como marco central.



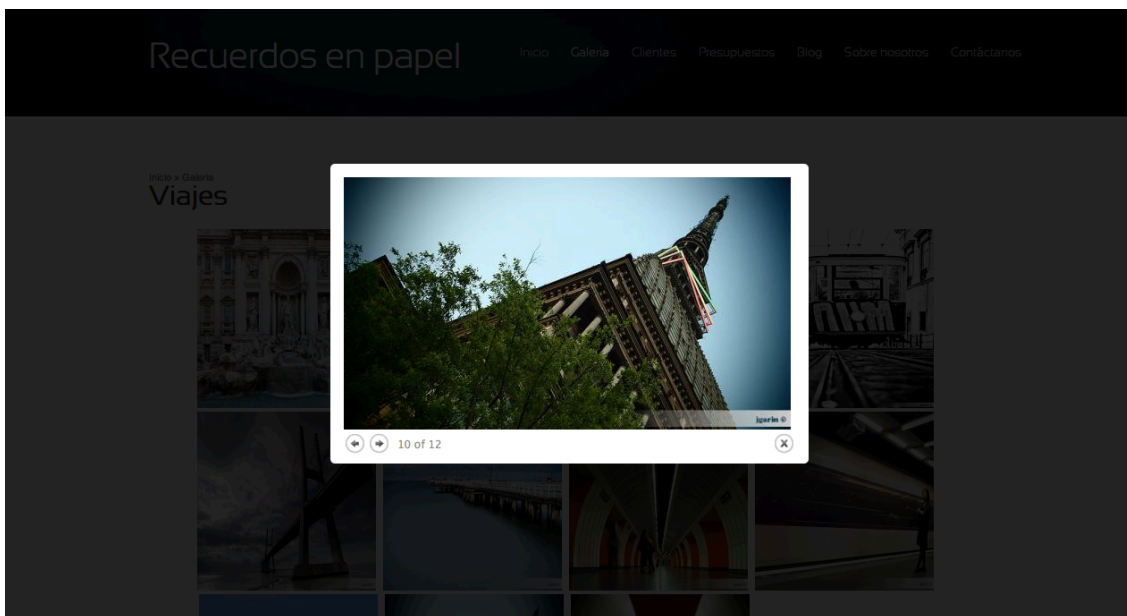
### ~ Galería/Otros

La galería *Otros/Besteak* incluye fotos minimalistas, fotos hechas por que sí, momentos de ocio, inspiraciones... Fotos incluidas dentro de esta galería pueden ser utilizadas como imagen de empresas, publicidad, etc.

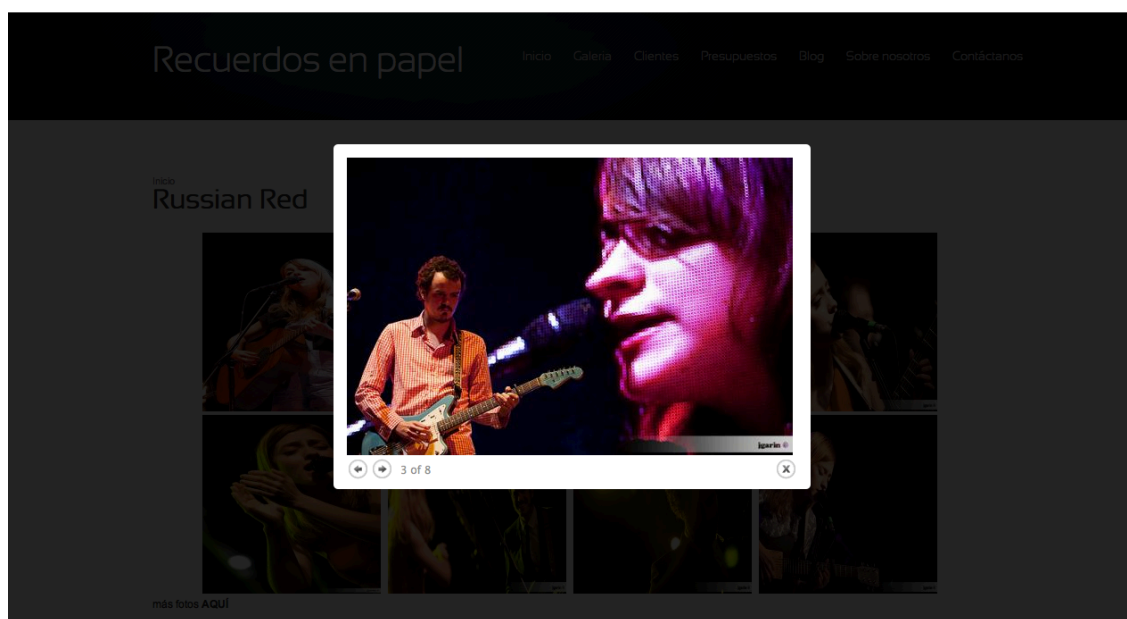


## ☀ Vista individual de las imágenes

Dentro de cada galería particular podemos ver una a una las fotos. Clickando sobre cualquiera de ellas se nos abrirá en tamaño original y bien dándole a los botones derecha o izquierda, o bien clickando sobre las flechas que aparecen en el popup podemos ir pasando las fotos para verlas todas ellas.

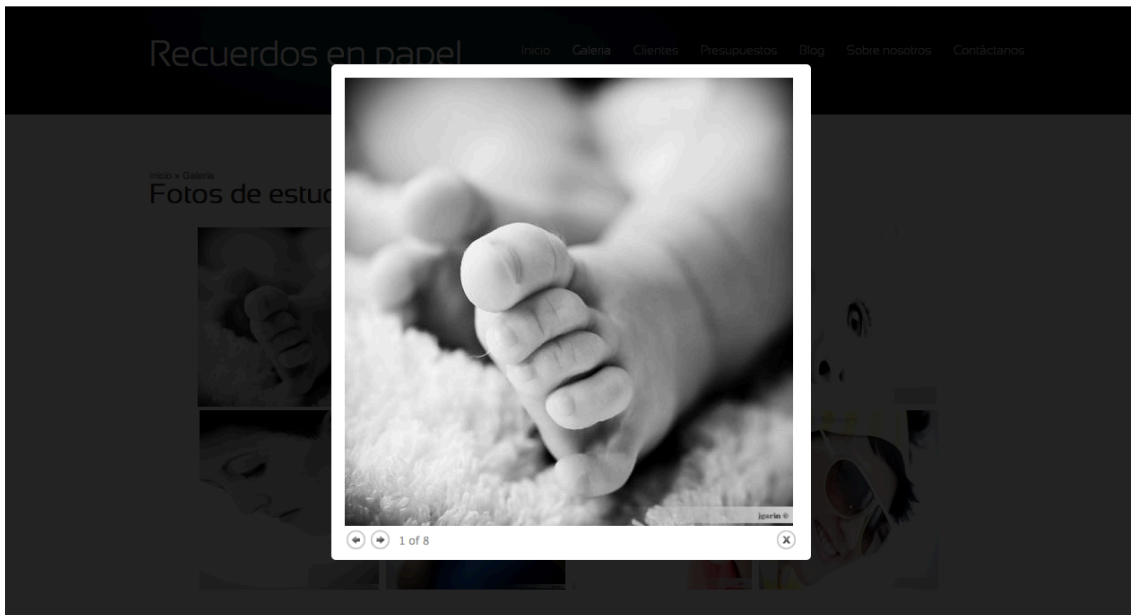


Mole Antonelliana - Torino



Russian Red - En concierto





Fotos de estudio

## ☀ Clientes

Si seguimos navegando por el menú, llegaremos a la pestaña clientes. Dentro de esta tenemos tres posibilidades. Podemos acceder a la página si somos usuarios registrados mediante nuestro nombre de usuario y contraseña.

¿Pero qué ocurre si se nos olvida la contraseña? Si al intentar acceder vemos que nos hemos olvidado de nuestra contraseña, podemos contactar con el administrador de la web para que nos la vuelva a mandar a nuestro correo electrónico.

Si por el contrario no somos usuarios registrados pero estamos bajo conocimiento de un nombre de usuario y contraseña invitados, podemos acceder a ver el reportaje de fotos completo de nuestros amigos.

A screenshot of the 'Cuentas de usuario' (User Accounts) page on the 'Recuerdos en papel' website. The page has a dark header with the site name and navigation menu. The main content area is light gray and contains the following elements:

- Header: 'Inicio' and 'Cuentas de usuario'.
- Sub-header: 'Inicia sesión para acceder a tus reportajes y otros datos'.
- Buttons: 'Iniciar sesión' and 'Solicitar una nueva contraseña'.
- Form fields: 'Nombre de usuario \*' and 'Contraseña \*'.
- Instructions: 'Escribe su nombre de usuario Recuerdos en papel.' and 'Escribe la contraseña asignada a su nombre de usuario.'
- Submit button: 'INICIAR SESIÓN'.
- Link: '¿Has visto las fotos de tus amigos? ¿Quieres volver a recordárlas? Entra aquí y hazlo!'.
- Access button: 'ACCEDER'.

## Presupuestos

Desde un principio el cliente tenía claro que quería darle la posibilidad al usuario de crearse él mismo su propio presupuesto. Por ello hemos creado un interfaz que primero, le pida al usuario sus datos personales.



Recuerdos en papel Inicio Galería Clientes Presupuestos Blog Sobre nosotros Contáctanos

Inicio  
Presupuesto

Indíquenos qué tipo de reportaje fotográfico desea realizar y en menos de 48h nos pondremos en contacto con usted

Nombre \*

Apellidos \*

Teléfono \*

E-mail \*

Indique una dirección de correo electrónico para poder contactar con usted

NEXT PAGE >

Una vez introducidos, pasamos a otro formulario con varias opciones. En este caso se le brinda al usuario la opción de elegir entre si desea crear un reportaje fotográfico sobre una boda o sobre otro tipo de eventos como serían *conciertos*, *fotos de estudio* o cualquier otro tipo de reportaje. Si este último caso es el elegido, el cliente tendrá que especificar qué tipo de reportaje fotográfico desea realizar.

Además de esto, si ha elegido como servicio *boda* tendrá que decir si quiere un reportaje completo o por el contrario sólo fotografiar parte del día. El usuario podrá elegir qué momento de la boda quiere fotografiar. Tendrá que indicar también si quiere un *álbum* que le permita recordar en papel aquellos momentos vividos o solamente quiere las fotos en formato digital.

Si se ha elegido *boda* o *concierto* tendrá que especificar también el lugar y la fecha concreta del evento. A parte de todo esto, el usuario también podrá dar una breve descripción sobre el reportaje que quiere realizar o cualquier comentario que quiera añadir.

Recuerdos en papel Inicio Galería Clientes Presupuestos Blog Sobre nosotros Contáctanos

Inicio  
**Presupuesto**

Elija el servicio que quiere contratar

<b>Boda</b>		<b>Album</b>	<b>Otros servicios</b>
<input type="checkbox"/> Preboda		<input type="checkbox"/> Si	<input type="checkbox"/> Conciertos
<input type="checkbox"/> Ceremonia		<input type="checkbox"/> No	<input type="checkbox"/> Estudio
<input type="checkbox"/> Banquete			<input type="checkbox"/> Otros
<input type="checkbox"/> Celebración			

Si ha elegido Otros, especifique qué tipo de servicio desea contratar

En caso de que quiera hacer un reportaje sobre Boda o Concierto, indique el día y el lugar del evento

Fecha: Ago 21 2011 Lugar:

Escriba una pequeña descripción sobre el reportaje que desea realizar

< PREVIOUS PAGE ENVIAR

## Blog

La empresa ha decidido crear un blog en el que poder ir publicando eventos fotográficos que vayan realizando. Su contenido será bastante variado, desde viajes realizados, escapadas ocasionales, conciertos, ... cualquier tipo de historia que el fotógrafo quiera compartir con sus clientes y visitantes de la página.

Cualquier usuario de la página tendrá acceso a este bloque y además, podrá postear cualquier cosa que quiera comentar al respecto sin tener que estar registrado en la misma.

La página de *Blogs* contendrá un resumen de lo que sería la entrada completa que ha posteado el fotógrafo. Si al cliente le ha interesado el post y quiere verlo completo, puede hacerlo clickando sobre el título del post o bien haciendo click sobre *leer más*.




## ☀ Sobre nosotros

La penúltima sección es una página estática sobre la empresa. Además, contiene un link que te envía directamente a la página de contacto para poder contactar con la empresa.



## Contáctanos



Recuerdos en papel Inicio Galería Clientes Presupuestos Blog Sobre nosotros Contáctanos

Inicio  
**Contactar**

Su nombre \*

Su dirección de correo electrónico \*





Asunto \*

Mensaje \*

Enviarle una copia.

**MANDAR MENSAJE**

Síguenos - Jarraitu

Mediante este formulario el usuario puede ponerse en contacto con la empresa. Solamente tiene que indicar su nombre, dirección de correo electrónico mediante el cual poder contactar con él, un asunto a modo de resumen y el mensaje que quiere hacer llegar.

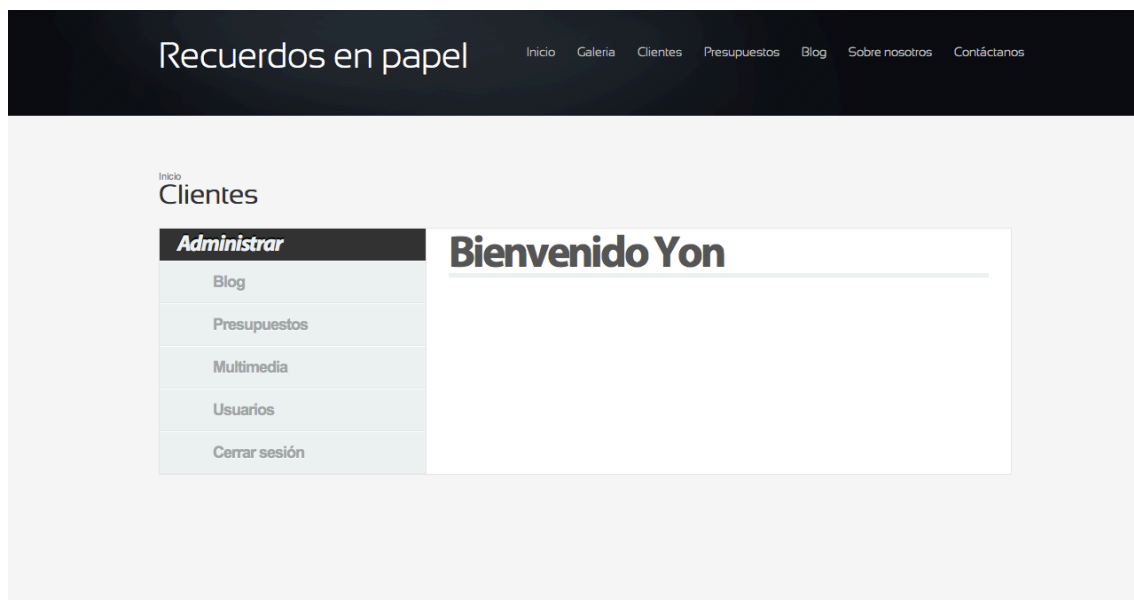
Además, en la parte derecha podemos encontrar de nuevo los enlaces a la redes sociales en las que la empresa o fotógrafos tienen perfiles creados.



**Usuario registrado: Yon** (cliente con privilegios de pseudoadministrador)

Consultando con la empresa, se llegó al acuerdo de que yo seré en un futuro la encargada de administrar la página web pero se requerirá de un usuario que tenga ciertos privilegios más allá de los privilegios que tuviese un simple usuario registrado. Por ello, se crea un pseudoadministrador que tenga permisos para administrar contenidos propios, administrar los presupuestos recibidos y crear respuestas a los mismos, administrar las galerías de imágenes y usuarios.

Por ello, cuando se inicia sesión con este usuario, encontramos un menú donde están todas estas opciones listadas en el lado derecho.

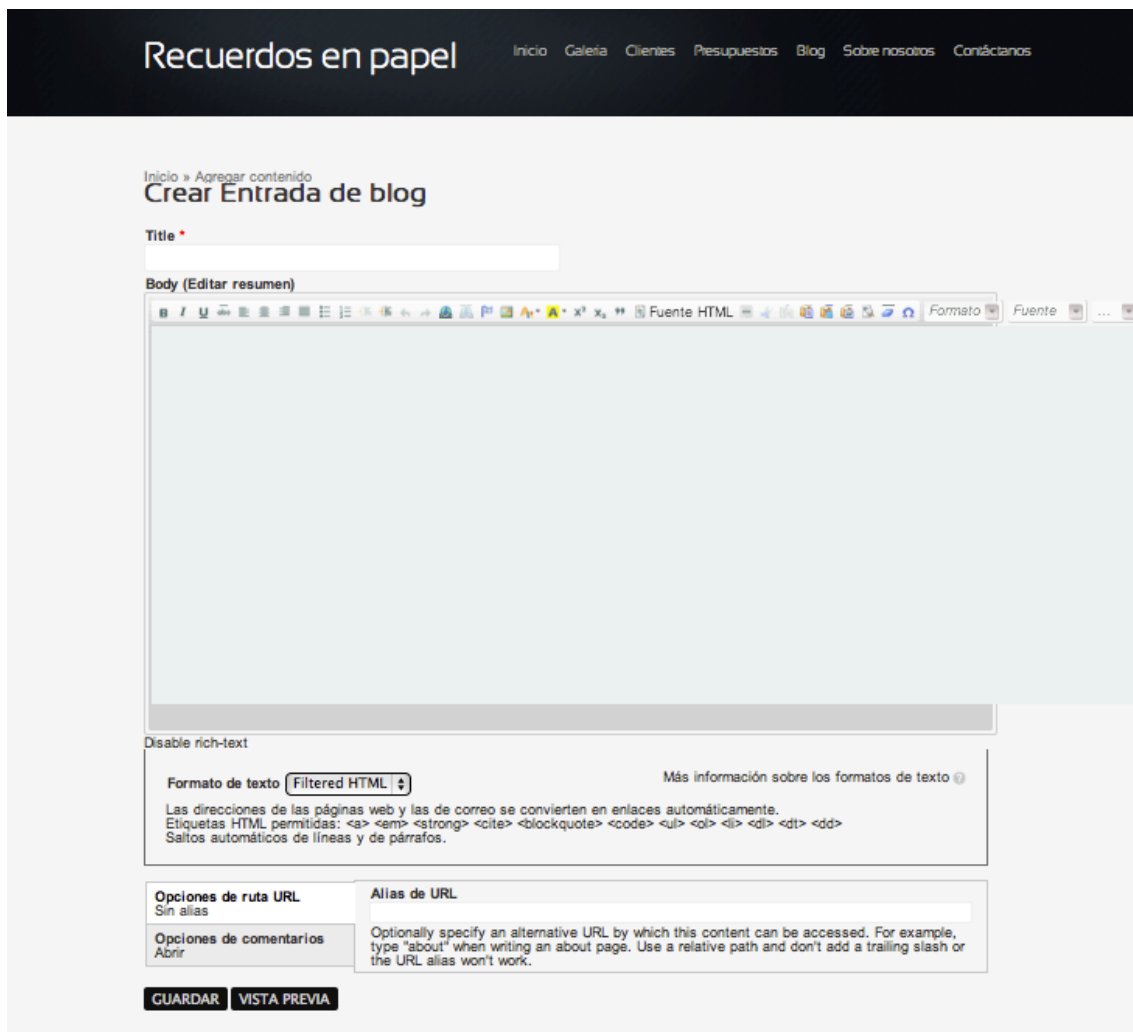


### ● **Blog/Listar Entradas**

Clickando sobre *Blog*, se nos desplegará una lista de opciones. Entre ellas podemos encontrar listar entradas, que nos lleva directamente a una página en la que tenemos un select con todas las entradas a los post realizados. Seleccionando el post que queramos y dándole a aceptar, nos redirigirá directamente a la página en cuestión.



● **Blog/Agregar Entradas**



Si en vez de elegir *listar entradas* elegimos agregar entradas, el enlace nos redigirá directamente a una página en la que crear un nuevo post.

### ● **Presupuestos/Crear Presupuestos**

La segunda opción que tenemos en este menú de administración es *Presupuestos*. En esta encontraremos *Crear presupuestos* o *Listar presupuestos*.

La primera opción nos lleva a una página en la que se nos listan todos los presupuestos que hemos recibido recientemente y seleccionaremos el deseado. Esta selección nos lleva directamente a una página en la que se realizará el presupuesto. Tras consultar la relación precio-servicios en la base de datos, obtendremos una aproximación de lo que sería el presupuesto inicial de respuesta.

Dicho presupuesto podrá ser modificado según las exigencias del cliente hasta quedar totalmente personalizado. Una vez tengamos el presupuesto definido, nos pondremos en contacto con el cliente para mandarle el presupuesto vía online o para acordar una cita para presentarlo personalmente.





**Recuerdos en papel** Inicio Galería Clientes Presupuestos Blog Sobre nosotros Contáctanos

Inicio  
**Clientes**

**Administrar**

- Blog
- Presupuestos
- Multimedia
- Usuarios
- Cerrar sesión

## Crear presupuesto

**Datos del cliente**

Nombre: Maitane  
Apellidos: Itoiz Lleida  
Teléfono: 948225773  
Dirección de correo: i.maitane@hotmail.com

PRESUPUESTO	
Servicio	Precio
Preboda	215 €
Ceremonia	650 €
Celebración	250 €

Total: 1115€  
Iva 18%: 200.7 €  
**Precio final: 1315.7 €**

[Guardar presupuesto](#)

● **Presupuestos/Listar presupuestos**

Si lo que se desea es ver los presupuestos que se han recibido, basta con acceder a Listar presupuestos y obtendremos una tabla en la que se mostrarán todos ellos. Estos presupuestos tendrán cada uno su número de presupuesto y la fecha en la que lo hemos recibido. Desde ahí mismo podremos acceder a verlo detalladamente, editarlo o bien eliminarlo.

**Recuerdos en papel** Inicio Galería Clientes Presupuestos Blog Sobre nosotros Contáctanos

Inicio  
**Últimas solicitudes de presupuesto recibidas**

Nº de presupuesto	Fecha de emisión	Ver detalladamente	Editar	Eliminar
1	08/21/2011 - 13:46	vista	Editar	eliminar
2	08/21/2011 - 13:47	vista	Editar	eliminar

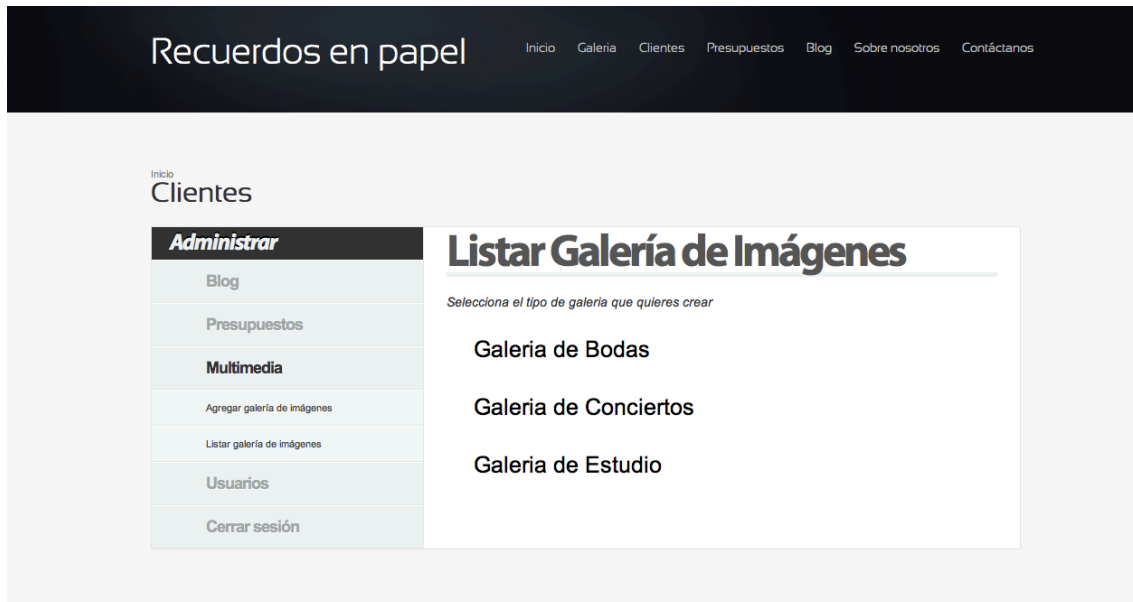
## ● Multimedia / Agregar galería de imágenes

La tercera parada de nuestro menú es Multimedia. En ella podemos agregar galerías de imágenes o bien listar las que ya tenemos creadas. Para agregar una nueva galería basta con seleccionar el tipo de galería que se quiere crear y seleccionar las fotos que queremos subir. Una vez subidas estas fotos, tendremos que indicarle qué usuario será el propietario de dichas imágenes y a qué grupo de taxonomía pertenecerá. De esta manera podemos hacer que el usuario al que se le ha realizado el trabajo sea el único al que se le autorice la visión de las fotos.



● **Multimedia/Listar galería de imágenes**

Para ver las galerías que tenemos creadas no tendremos más que seleccionar cual de las galerías queremos ver. Una vez seleccionada, la página consultará sobre la base de datos y nos mostrará aquellas galerías que coincidan con nuestra selección.



## ● Usuarios/ Crear usuario

Apartado usuarios. Desde aquí podemos administrar los usuarios que vayan a tener acceso a la página. Bastará con darles un nombre y una contraseña, así como indicar su dirección de correo electrónico, el estado en el que estará la cuenta (activa/bloqueada) y el idioma en el que verá el usuario la página (español por defecto).

The screenshot shows the 'Recuerdos en papel' website header with navigation links: Inicio, Galería, Clientes, Presupuestos, Blog, Sobre nosotros, and Contáctanos. The main content area is titled 'Inicio' and contains a form for creating a new user account. The form fields include: 'Nombre de usuario', 'Dirección de correo electrónico', 'Contraseña' (with a strength indicator), and 'Confirmar contraseña'. Below these fields are radio buttons for 'Estado' (Bloqueado, Activo) and 'Opciones de idioma' (Inglés (English), Español). A 'CREAR NUEVA CUENTA' button is at the bottom of the form.

## ● Usuarios/ Listar usuarios

Clickando sobre este enlace, nos mostrará una tabla con información sobre los usuarios. Desde su nombre, el estado, el rol que tomará, desde cuándo forma parte de nuestros usuarios, el último acceso que tuvo y un enlace que nos permitirá acceder a la página de edición.

**Recuerdos en papel** Inicio Galería Clientes Presupuestos Blog Sobre nosotros Contáctanos

Inicio  
**Personas**

**Agregar usuario**

Mostrar sólo los usuarios en los que

rol: cualquiera  
permiso: cualquiera  
estado: cualquiera **FILTRAR**

Opciones de actualización  
Desbloquear a los usuarios seleccionados **ACTUALIZAR**

<input checked="" type="checkbox"/>	Nombre de usuario	Estado	Roles	Miembro durante	Último acceso	Operaciones
<input type="checkbox"/>	Sergiojaione	activo		1 día 22 horas	hace 1 día 19 horas	editar
<input type="checkbox"/>	Maitane	activo		6 días 19 horas	hace 3 días 3 horas	editar
<input type="checkbox"/>	Yon	activo		YonAdmin3 semanas 1 día	hace 2 mins 23 segs	editar
<input type="checkbox"/>	admin	activo		administrator2 meses 4 días	hace 35 mins 40 segs	editar



## Usuario registrado

El usuario registrado tendrá acceso a sus propios reportajes y a sus presupuestos. Si elige ver sus reportajes, el sistema consultará la base de datos para mostrarle qué tipo de reportajes tienen creados y a partir de ahí mostrar una lista con los nombres.

Si por el contrario eligen ver los presupuestos, podrán consultar aquellos presupuestos que todavía están sin cerrar.

**Recuerdos en papel** Inicio Galería Clientes Presupuestos Blog Sobre nosotros Contáctanos

Inicio  
**Clientes**

**Acceder**

- Mis reportajes
- Mis presupuestos
- Cerrar sesión

**Bienvenido Sergiojaione**

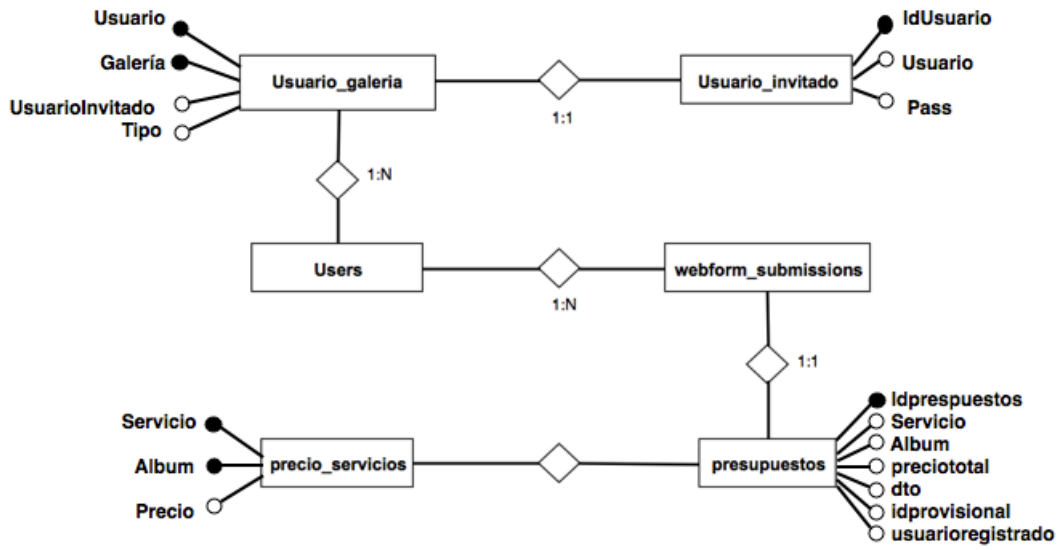
## 4.2. Análisis de la base de datos

Como ya se ha comentado en el apartado de instalación de Drupal, precisa de una base de datos donde poder instalar todo lo necesario para el correcto funcionamiento. Toda la información relacionada con nodos, formularios, vistas, taxonomía, usuarios, privilegios, ... estará incluido dentro de esta base de datos. Puede ocurrir que, a parte de todas estas tablas creadas por defecto, necesitemos alguna tabla diferente que satisfaga las necesidades concretas de nuestro proyecto.

En nuestro caso, necesitamos crear varias tablas adicionales ya que tenemos una serie de datos que no pertenecen a ninguna de las categorías que ya están creadas. Por ejemplo, necesitamos almacenar todo lo relacionado con el tema de los presupuestos realizados por el usuario. Cuando un usuario (ya sea registrado o no) realiza un presupuesto, dicho presupuesto se almacenará dentro de `webform_submissions` (tabla contenida por defecto entre las tablas de la base de datos de Drupal) y desde ahí le llegará al encargado de la página la notificación de un nuevo presupuesto. Existirá la oportunidad de responder a ese presupuesto y para ello se necesitará una tabla que exponga qué precio tendrá cada servicio, para poder consultarla y realizar un baremo aproximado. Además, una vez calculado cuánto es el presupuesto a realizar, este se guardará para poder mandárselos al cliente.

Por otra parte, una vez realizado el reportaje fotográfico, este estará expuesto en la página web para que los propios usuarios puedan verlo. Por tanto, se necesita saber qué galería le pertenece a qué usuario y además , si les ha proporcionado o no un usuario y contraseña invitado para que sus amigos y familiares puedan acceder a ver las fotos.

El E/R quedaría del siguiente modo:



## Capítulo 5: Desarrollo

Las aplicaciones que podemos ver y encontrar en internet son cada vez más complejas, ya que incorporan efectos visuales, drag and drop, auto-completar, animaciones etc. el desarrollar todos estos conceptos desde cero puede resultar complicado pero el empleo de librerías como jQuery nos facilitan el desarrollo de la aplicación. Otra ventaja paralela es despreocuparnos cuando codificamos en la compatibilidad de navegadores, ya que la librería resolverá esto.

### 5.1. jQuery

jQuery<sup>[3]</sup> es una librería Javascript de código abierto que ayuda a desarrollar funcionalidades complejas tanto en Javascript como en Ajax (Asynchronous JavaScript + XML). Facilita la interacción de los diferentes códigos cliente y proporciona un desarrollo basado en objetos y modular para realizar aplicaciones más interactivas y con comportamientos más aparentes. Así que es una librería a utilizar tanto por aquellos que empiezan con javascript, ya que es más sencillo de utilizar para desarrollar Ajax, como por aquellos que tengan experiencia utilizando lenguajes de cliente, en este caso facilitará el desarrollo y acelerará los tiempos de implementación. A través de jQuery se puede acceder al código DOM (Document Object Model) de las páginas HTML o XHTML y manipular sus elementos, así como cambiar propiedades de las CSS, reaccionar ante eventos o facilitar el uso de efectos y transiciones, entre otras cosas. Además, es modular y se puede extender mediante la gran cantidad de plugins existentes que complementan y habilitan funcionalidades extra que van desde efectos drag and drop a multitud de presentaciones de datos en forma de carrusel, de acordeon, etc.

Desde la versión 5, allá por 2006, se ha incluido jQuery en el core de Drupal, convirtiéndose así en la librería "oficial" de desarrollo en javascript y Ajax para módulos y plugins en aplicaciones de Drupal. Antes de incluir jQuery, otras librerías como Dojo o Prototype fueron consideradas, pero jQuery fue elegido gracias a su estabilidad, a su tamaño ligero, y sobre todo a la colaboración de su creador, John Resig, facilitando que la licencia de la librería permitiese integrarla en Drupal. La versión de jQuery agregada en el core de



Drupal varía según si es Drupal 6, que incluye jQuery 1.2, o 1.3 o Drupal 7, que incluye jQuery 1.4.4 en sus últimas subversiones. De todas formas, el módulo jQuery update permite actualizar la versión de jQuery de las diferentes versiones de Drupal.

### 5.1.1. Nivo Slider

Nivo Slider es un slideshow que corre bajo jQuery, muy sencillo de usar y de configurar. El slider se caracteriza por presentar imágenes de una manera muy elegante o informal según el gusto del consumidor. Se puede personalizar tanto hasta llegar a colocarle los efectos de transición entre cada imagen y la duración de cada una entre muchas otras opciones.

Esta constituido por tres partes:

- × *jQuery*
- × *El script del Nivo Slider*
- × *El CSS del Nivo Slider*

Posee características particulares como:

- × *16 efectos de transición*
- × *Se puede ajustar a nuestras necesidades, dirección y control de navegación*
- × *La versión sólo pesa 7kb*
- × *Apoya la vinculación en imágenes*
- × *Teclado de navegación.*

Nivo Slider ha sido probado en los siguientes navegadores: Internet Explorer v7+, Firefox v3+, Google Chrome v4, Opera v10.5, entre otros. Podemos descargarlo desde <http://nivo.dev7studios.com/>

Para crear el banner que está en la página principal hemos seguido los siguientes pasos. Hemos creado un nodo con todas las fotos que formarían parte del slider. Una vez tenemos todas las fotos subidas al servidor, lo que hacemos es crear una vista que muestre todas estas fotos. Le indicaremos que el modo de visualización de estas fotos será mediante views nivo slider y ahí mismo podemos configurar qué tipo de transición queremos, además del tiempo transcurrido entre una transición y otra y si queremos que al hacer click sobre la imagen, nos redireccione a una URL concreta. Guardaremos la vista como bloque para poder agregarla a nuestra página principal.

## 5.1.2. Colorbox

Interfaces como las galerías de imágenes incorporan un módulo, llamado *Colorbox*, que utiliza funciones jQuery para conseguir dicho efecto.

ColorBox es un plugin para jQuery 1.3 muy ligero (12kb) que permite hacer unos lightbox muy atractivos. Cada día se utilizan mas los lightbox y son más los plugins que se crean para ello. Este en especial es muy ligero, personalizable y fácil de utilizar.

Estas son unas de sus características mas interesantes:

- × *Soporta la utilización de fotos de forma individual o en grupos, ajax en línea, y contenido iframed.*
- × *Diversos Estilos. Apariencia completamente controlada a través de CSS, para que los usuarios puedan dar nuevos estilos a la caja.*
- × *EL comportamiento de los ajustes puede ser sobre-escrito, sin alterar el archivo de Javascript ColorBox.*
- × *HTML Totalmente discreto, no requiere cambios en los existentes HTML.*
- × *Efecto atractivo, con discreto degradado.*
- × *Precarga de imágenes de fondo para evitar la inflamación de unstyled.*
- × *Ligero: sólo 12kbs de JavaScript.*
- × *Escrito en jQuery y puede ser encadenado con otros comandos de jQuery.*
- × *Genera XHTML y CSS válido.*
- × *Compatible con cualquier navegador web.*

Crear una galería con *colorbox* es muy simple. Simplemente se tiene que crear un nodo con todas las imágenes que se quiera que pertenezcan a la galería. Una vez se tienen todas las fotos subidas, se crea una vista que muestre las fotos que se acaban de subir. Para que solamente se muestren esas fotos exactamente, a las fotos subidas anteriormente, se les pondrá un tag, una taxonomía o algún tipo de referencia que nos ayude en la selección. Una vez que las fotos están en el servidor, se elige la opción para que no las muestre con el módulo *colorbox* de modo que cada vez que hagamos click sobre la imagen, se abra la foto con el formato deseado. Para ello, el módulo le indica que tendrá que coger el estilo *colorbox imagefield* incluido en el css de *colorbox*.

### 5.1.3. Script creados en JavaScript

jQuery, además de permitir añadir módulos que realizan operaciones concretas, permite añadir código JavaScript<sup>[4][5]</sup> escrito por nosotros mismos. De este modo, podemos agregar script externos a drupal dentro del mismo.

#### × Menú de usuario registrado

Cuando iniciamos sesión como usuario registrado, dependiendo de qué usuario sea el que ha iniciado sesión se le cargará un menú u otro diferente. Independientemente de cual sea el que se cargue, estará escrito en código php que realiza lo siguiente (cambiarán las URL a las que será redireccionando):

```
<?php extract($_REQUEST);?>
<div id="pagina">
  <div id="menu">
    <h3>Administrar</h3>
    <ul>
      <li><span <?php if($p == "posts"){echo 'class="current"';} ?><a href="javascript:void(0);" id="link-posts">Posts</a></span>
        <ul>
          <li><a href="?p=posts&sec=agregar-post" class="add">Agregar</a></li>
          <li><a href="?p=posts&sec=listar-posts" class="list">Listar</a></li>
        </ul>
      </li>
      <li><span <?php if($p == "categorias"){echo 'class="current"';} ?><a href="javascript:void(0);" id="link-categorias">Presupuestos</a></span>
        <ul>
          <li><a href="?p=presupuestos&sec=elegir_presupuesto" class="add">Agregar</a></li>
          <li><a href="?p=presupuestos&sec=listar-presupuestos" class="list">Listar</a></li>
        </ul>
      </li>
      <li><span <?php if($p == "multimedia"){echo 'class="current"';} ?><a href="javascript:void(0);" id="link-multimedia">Multimedia</a></span>
        <ul>
          <li><a href="?p=multimedia&sec=agregar-galeria-imagenes" class="add">Agregar galer&iacute;a de im&aacute;genes</a></li>
          <li><a href="?p=multimedia&sec=listar-galeria-imagenes" class="list">Listar galer&iacute;a de im&aacute;genes</a></li>
          <li><a href="?p=multimedia&sec=agregar-galeria-video" class="add">Agregar galer&iacute;a de video</a></li>
          <li><a href="?p=multimedia&sec=listar-galeria-video" class="list">Listar galer&iacute;a de video</a></li>
        </ul>
      </li>
      <li><span <?php if($p == "usuarios"){echo 'class="current"';} ?><a href="javascript:void(0);" id="link-usuarios">Usuarios</a></span>
        <ul>
          <li><a href="?p=usuarios&sec=agregar-usuario" class="add">Agregar</a></li>
          <li><a href="?p=usuarios&sec=listar-usuarios" class="list">Listar</a></li>
        </ul>
      </li>
      <li><span <?php if($p == "cerrar"){echo 'class="current"';} ?><a href="/recuerdos/user/logout" id="link-usuarios">Cerrar sesión</a></span>
    </ul>
  </div> <!--/menu-->
  <div id="contenido">
    <?php
    if ($p==''){
      require("sites/all/menu/home.php");
      $p=1;
    } elseif (file_exists("sites/all/menu/$p.php")) {
      require("sites/all/menu/$p.php");
    } elseif ($p==404) {
      require("sites/all/menu/404.php");
    }
    if ($pres=='si'){
      require("sites/all/menu/paginas/presupuestos/crear_presupuesto.php");
    }
    if ($insertar=='si'){
      require("sites/all/menu/paginas/presupuestos/inserta_presupuesto.php");
    }
  >
  </div> <!--/contenido-->
</div><!--/pagina-->
```

Como podemos ver, se nos cargará un menú lateral en el que, en este caso, tendremos cuatro opciones para elegir. Cuando clickemos sobre cualquiera de ellas se nos activará el siguiente código JavaScript:

```
jQuery(document).ready(function(){
  //Menu desplegable
  jQuery("#menu ul li ul").hide();
  jQuery("#menu ul li span.current").addClass("open").next("ul").show
  ();
  jQuery("#menu ul li span").click(function(){
    jQuery(this).next("ul").slideToggle("slow").parent
    ("li").siblings ("li").find("ul:visible").slideUp("slow");
    jQuery(this).toggleClass("open");
    jQuery(this).parent("li").siblings("li").find
    ("span").removeClass ("open");
  });
});
```

Viendo los dos script, se puede ver que cada opción pertenece a una lista (etiqueta *ul*). Inicialmente la lista estará oculta y en el momento en el que se clicke sobre una de las opciones, se activará el método *span.current* que lo que hace es, desplegar la lista y mostrar las opciones que hay. Le indicamos que dicho despliegue se haga despacio. Una vez que la lista está abierta, aparecerán las diferentes opciones que se pueden realizar y si se clicka sobre cualquiera de ellas, enviará unas variables en la cláusula *href* con distintos valores dependiendo de donde se haya clickado. Al final del código tenemos una sentencia *if* que, dependiendo del valor de la variable que le ha llegado, redireccionará la página a una dirección o a otra diferente.

Una vez que esta página redirecciona a otro link, accederá a una página puente entre varias páginas. El código Php de estas dos será:

```
<?php
if ($sec==''){
    echo '<meta content="0; URL=?p='.$p.'&sec=listar-galeria-imagenes" http-equiv="Refresh" />';
} elseif (file_exists("sites/all/menu/paginas/$p/$sec.php")) {
    require("sites/all/menu/paginas/$p/$sec.php");
} else {
    echo '<meta content="0; URL=?p=404" http-equiv="Refresh" />';
}
?>
```

En caso de que la variable *\$sec* esté vacía, redireccionará la página a una página concreta. Si dicha variable tiene un valor asignado, lo que hará será concatenar esa variable con la variable *\$p* que se le ha pasado en la URL, creando de este modo la dirección que tiene que cargar. Si por el contrario ninguna de estas opciones es válida, mostrará una página de error.

Opciones como *Listar post* o *Crear presupuestos*, muestran una página intermedia en la que, tras consultar a la base de datos, muestra una lista desplegable con las opciones disponibles. Cuando elegimos la opción que queramos ver será cuando ya nos redirija a la página en cuestión.

### × *Listar post*

```
<h2> Listar entradas del blog </h2>
<div id="contenido">
<?php
$query="SELECT nid, title FROM {node} WHERE type='blog'";
$result=db_query($query);?>
<p style="font-size:17px"> Selecciona la entrada de blog que quiere ver </p>
<form method="POST">
<select name="entradas">
<option> Entradas de blog
<?php
while ($row = $result->fetchAssoc()){?>
<option value="<?php echo $row['nid'] ?>"> <?php echo $row['title'];
} ?>
</select>
<input type="submit" id="edit-submit" name="acceptar" value = 'Aceptar' class="form-submit" style="margin-left:10px"/>
</form>
<?php
if(isset($_POST['acceptar'])){
    $nid=$_POST['entradas'];
    drupal_goto('node/'.$nid);
}
?>
</div>
```

## × Crear presupuesto

```

<h2> Crear presupuesto </h2>
<div id = "contenido">
  <p style="font-size:17px">Elige el número del presupuesto que quieres responder </p>
  <div style="float:left; margin-right:10px">
    <?php $query="Select sid from {webform_submissions}";
    $pres="no";
    $result = db_query($query); ?>
    <form method="POST" action="?pres=si&p=1">
      <select name="lista">
        <option> nºpresupuesto
        <?php
          while ($row = $result->fetchAssoc()){ ?>
            <option value="<?php echo $row['sid'] ?>"> <?php echo $row['sid'];
          }?>
      </select>
      <input type="submit" id="edit-submit" name="aceptar" value = 'Aceptar' class="form-submit" />
    </form>
  </div>
</div>
</div>

```

Como se puede ver, en el primero de ellos una vez que tenemos el post que se desea ver seleccionado, nos redirige directamente a la página. En el segundo caso sin embargo, pasa a la página principal dos variables con valores extraídos del formulario. Dependiendo del valor de estas variables, una vez se haga el redireccionamiento a la página de crear presupuesto, volverá a consultar a la base de datos y teniendo en cuenta estas variables extraerá los datos correspondientes al presupuesto elegido.

La variable \$pres es transmitida desde la página de elegir presupuesto a la página crear presupuesto y dependiendo del valor que tenga esta variable se obtendrán los datos de la base de datos. Se realiza la consulta pasándole esta variable como parámetro. Con los datos que se extraen, se vuelve a consultar la base de datos para extraer información más concreta y se muestra el resultado obtenido por pantalla. Una vez se tiene el presupuesto que se quiere diseñar a gusto del cliente, se vuelve a llamar a la página principal pasándole varias variables que harán la llamada a la página de insertar presupuesto y le pasarán los valores que tiene que almacenar.

```

<head>
<?php
  $npres=$_POST['lista'];
  $banquete=false;$celebracion=false;$ceremonia=false;$reboda=false;$concierto=false;$estudio=false;$boda = false;
  $preciopre=0;$precioere=0;$precioiban=0;$preciocele=0;$preciocon30=0;$preciocon45=0;$estudio20=0;$estudio30=0;
  $suma=0;$insertar="no";
?>
</head>
<body>
<h2> Crear presupuesto </h2>
<?php
  $query = "SELECT form_key, data FROM {webform_submitted_data} NATURAL JOIN {webform_component} WHERE sid = $npres;";
  $result = db_query($query);
  $i = 0;
  while ($row = $result->fetchAssoc()){
    $componente[$i]=$row['form_key'];
    $dato[$i]=$row['data'];
    $i=$i+1;
  }
  for ($j=$i-1; $j>0; $j--){
    switch ($componente[$j]) {
      case "nombre":
        $nombre = $dato[$j]." ";
        break;
      case "apellidos":
        $apellidos = $dato[$j]." ";
        break;
      case "telefono":
        $telefono = $dato[$j]." ";
        break;
      case "correo":
        $mail = $dato[$j]." ";
        break;
    }
  }

```



```

<table style="font-size: 14px; border:1px solid">
  <tr>
    <td colspan=2 style="padding:10px;background-color:#EBF1F1"> <h3> PRESUPUESTO 1</h3></td>
  </tr>
  <tr>
    <td style="padding:10px 20px"> <b> Servicio </b> </td>
    <td style="font-size: 14px"> <b> Precio </b> </td>
  </tr>
  <tr>
    <td style="padding:10px 60px; font-size: 14px"> Estudio 20 fotos </td>
    <td style="padding-top:10px; font-size: 14px"> <?php echo $precios20." € </td>;
    $suma1 = $precios20;?>
  </tr>
  <tr><td colspan=2><hr></td></tr>
<?php } ?>
  <tr>
    <td align="right" style="padding:15px"> <b> Total: </b> </td>
    <td> <?php echo " ".$suma1."€" ?> </td>
  </tr>
  <tr>
    <td align="right" style="padding:15px"> <b> Iva 18%: </b> </td>
    <td> <?php $iva = ($suma1*0.18); echo $iva." €" ?> </td>
  </tr>
  <tr style="font-size:18px">
    <td align="right" style="padding:15px"> <b> Precio final: </b> </td>
    <td><?php $total=$suma1+$iva; echo $total." €" ?> </td>
  </tr>
  <tr>
    <td colspan=2 align="right" style="padding:30px">
      <form method="post" action="?p=1&insertar=si&npres=<?php echo $npres."&servicio=".$servicio."&album=".$album."&total=".$total?>">
        <input type="submit" value = "Guardar presupuesto" name="guardar">
      </form>
    </td>
  </tr>
</table>
&nbsp;

<?php if ($concierto){ ?>
<table style="font-size: 14px; border:1px solid">
  <tr>
    <td colspan=2 style="padding:10px;background-color:#EBF1F1"> <h3> PRESUPUESTO 2</h3></td>
  </tr>
  <tr>
    <td style="padding:10px 20px"> <b> Servicio </b> </td>
    <td style="font-size: 14px"> <b> Precio </b> </td>
  </tr>
  <tr>
    <td style="padding:10px 60px; font-size: 14px"> Concierto 45 fotos </td>
    <td style="padding-top:10px; font-size: 14px"> <?php echo $preciocon45." € </td>
    $suma2 = $preciocon45;?>
  <tr><td colspan=2><hr></td></tr>
  <?php } elseif ($estudio){ ?>
  <table style="font-size: 14px; border:1px solid">
    <tr>
      <td colspan=2 style="padding:10px;background-color:#EBF1F1"> <h3> PRESUPUESTO 2</h3></td>
    </tr>
    <tr>
      <td style="padding:10px 20px"> <b> Servicio </b> </td>
      <td style="font-size: 14px"> <b> Precio </b> </td>
    </tr>
    <tr>
      <td style="padding:10px 60px; font-size: 14px"> Estudio 30 fotos </td>
      <td style="padding-top:10px; font-size: 14px"> <?php echo $precios30." € </td>;
      $suma2 = $precios30;?>
    </tr>
    <tr><td colspan=2><hr></td></tr>
  <?php } ?>
  <tr>
    <td align="right" style="padding:15px"> <b> Total: </b> </td>
    <td> <?php echo " ".$suma2."€" ?> </td>
  </tr>
  <tr>
    <td align="right" style="padding:15px"> <b> Iva 18%: </b> </td>
    <td> <?php $iva = ($suma2*0.18); echo $iva." €" ?> </td>
  </tr>
  <tr style="font-size:18px">
    <td align="right" style="padding:15px"> <b> Precio final: </b> </td>
    <td><?php $total=$suma2+$iva; echo $total." €" ?> </td>
  </tr>

  <tr>
    <td colspan=2 align="right" style="padding:30px">
      <form method="post" action="?p=1&insertar=si&npres=
        <?php echo $npres."&servicio=".$servicio."&album=".$album."&total=".$total?>">
        <input type="submit" value = "Guardar presupuesto" name="guardar">
      </form>
    </td>
  </tr>
</table>
<?php }?>
</div>
</body>

```

Lo mismo ocurre cuando se elige Listar galería de imágenes pero en este caso en vez de tener una lista desplegable en la que elegir el tipo de galería que se quiere mostrar, aparecerán links que redirigirán a la página. Dicho link pasará una variable (como en las otras situaciones) que será la que determine qué datos vamos a extraer de la base de datos.

```

<h2>Listar Galerías de Imágenes</h2>
<?php
$query=NULL;
$galeria=$_SESSION['galeria'];
$query = "SELECT vista, galeria FROM {usuario_galeria} WHERE tipo='".$galeria."'";
$query=db_query($consulta);
if ($query==NULL) {
    echo "No hay ninguna galería creada";
} else {
    $i=0;
    while($row = $query->fetchAssoc() ) {
        $nombre[$i]=$row['galeria'];
        $album[$i]=$row['vista'];
        $i=$i+1;
    }
    $j=0;
    $l=$i-1;
    $a=$i-1;
    ?> <table width="100%"> <?php
    while ($l>=0){
        if ($j<$a){
            for ($k=0;$k<2;$k++){
                ?>
                <td style="padding:10px 40px">
                    <h1><a href ="/recuerdos/galeria/<?php echo $album[$j]?>"> <?php echo $nombre[$j]. "</a></h1></td>";
                    $j=$j+1;
                }
            } else {
                ?> <td font-family="Miriad Tilt" style="padding:10px 40px">
                    <h1><a href ="/recuerdos/galeria/<?php echo $album[$j]?>"> <?php echo $nombre[$j]. "</a></h1></td>";
                    $j=$j+1;
                }
            echo "</tr>";
            $l=$a-$j;
        }
        echo "</table>";
    }?>

```

Para que todo esto funcione correctamente, se tienen que realizar ciertos cambios en la base de datos de drupal añadiendo tablas que no existen entre las tablas que se crean predefinidamente al instalar drupal.

## 5.2. Modificación de la base de datos

Para satisfacer los intereses del cliente, se necesita modificar la base de datos de Drupal, como se he comentado en el punto anterior. Para ello crearemos unas tablas que quedan recogidas a continuación, tal y como hemos visto en el E/R.

### × *Relación Usuario-Galería*

Necesitamos crear una relación que nos permita tener almacenado a qué usuarios les pertenece qué galería. Por ello, crearemos una tabla que almacene nombres de usuario, galerías, el tipo de reportaje que han realizado y si se le ha proporcionado un nombre de usuario y contraseña invitado para poder acceder a la misma.



```
CREATE TABLE usuario-galeria (
    usuario          varchar(60),
    galeria          varchar(64),
    usuarioinvitado  int(11) NULL,
    tipo             varchar(64),
    PRIMARY KEY (usuario, galeria),
    FOREIGN KEY fk_usuarios (usuario)
        REFERENCES users(name),
    FOREIGN KEY fk_usuarioinvitado(usuarioinvitado)
        REFERENCES usuario_invitado(IdUsuario));
```

× *Relación usuarios\_invitados*

Si los clientes quieren, se les puede proporcionar un usuario con su correspondiente contraseña para divulgarla entre sus amistades para que de este modo accedan a ver los reportajes de fotos realizados. Para ello, hemos creado una tabla en la que almacenaremos el Id de usuario, el nombre y la contraseña.

```
CREATE TABLE usuarios_invitados (
    IdUsuario      int(11),
    Usuario        varchar(64),
    Pass           varchar(64),
    PRIMARY KEY (IdUsuario));
```

× *Presupuestos*

Necesitamos una relación en la que poder almacenar los presupuestos que creamos. Para ello, creamos una tabla que contenga los siguientes campos:

```
CREATE TABLE presupuestos (
    idpresupuestos  int(11),
    servicio        varchar(11),
    album           boolean,
    preciototal     int(11),
    dto             int(11) NULL,
    idprovisional   int(11),
    usuarioregistrado int(11),
    PRIMARY KEY (idpresupuesto),
    FOREIGN KEY fk_idprovisional (idprovisional)
        REFERECES webform_submissions (sid),
    FOREIGN KEY fk_usuarioregistrado (usuarioregistrado)
        REFERENCES users(uid));
```

× *Precio\_servicios*

Además de estas tablas anteriores, para crear los presupuestos se necesitan unos precios establecidos. Ante esta necesidad se crea una relación precio\_servicio de modo que almacena cuánto valdrá cada servicio ofrecido en relación a si contará con álbum o no.

```
CREATE TABLE precio_servicios (  
    servicio    varchar(11),  
    album      boolean,  
    precio     int(11),  
    PRIMARY KEY (servicio, album));
```

## Capítulo 6: Gestión del proyecto

En diciembre del 2010, la empresa se puso en contacto con el desarrollador para una posible cooperación. En la primera reunión entre estos dos, la empresa expuso su intención de crear una página web que le sirviese para exponer sus productos en internet pero no tenía claro cuáles iban a ser los contenidos de la misma. Por tanto, se hizo una primera aproximación de los requisitos del cliente y el desarrollador optó por hacer un borrador para mostrarle al cliente si eso era lo que quería o había que tomar otro camino diferente.

### 6.1. Ciclo de vida de los requisitos

#### 6.1.1. Reunión N°1: definición de los requisitos

Una de las cosas claras que tenía el cliente era que la página tenía que diferenciar claramente tres apartados *Servicios*, *Presupuestos* y *Área de clientes*.

Con la ayuda del desarrollador se fueron determinando las opciones que tendrían cada uno de los apartados. El primer borrador quedó del siguiente modo:

- × *Apartado Servicios*
  - *Bodas*
  - *Conciertos*
  - *Fotos de estudio*
  - *Paisajismo*
  
- × *Apartado Clientes*
  - *Registro*
  - *Login*
  - *Presupuesto*
  - *Facturas*
  - *Reportajes*

- × *Apartado Presupuestos*
  - *Servicio requerido*
  - *Opción de libro digital*
  - *Lugar donde se realizará el reportaje*
    - *Estudio o particular*
    - *Kilometraje en caso de que sea fuera de la comarca de Pamplona*

Dentro del apartado Reportajes en Clientes, el usuario tendría la opción de ver todas las fotos realizadas en el reportaje. Además de eso, en caso de que el cliente así lo quisiese, se le proporcionaría un nombre de usuario y contraseña para que usuarios no registrados en la página puedan acceder con dichos datos a ver el reportaje. Este usuario y contraseña estarían diseñados para divulgarlo entre los amigos y familiares para que ellos también puedan disfrutar de las fotos.

En el apartado Presupuestos, el usuario podría crearse su propio presupuesto según sus intereses. Tendría que elegir el tipo de servicio que quiere contratar y si lo elegido es un reportaje de boda, debería que indicar qué momento de la boda quiere que sea retratado (preboda, ceremonia, banquete y celebración). Este presupuesto le llegaría directamente a la empresa que tras realizar el presupuesto completo se pondría en contacto con el usuario para acabar de determinar detalles.

### **6.1.2. Reunión N°2: Enero del 2011**

Semanas más tarde de la primera reunión, desarrollador y empresa se volvieron a juntar para una segunda aproximación de los requisitos. Desde la universidad se había propuesto la opción de desarrollar el software mediante un CMS y se le comentó esta opción al cliente. Juntos, pasaron a decidir cómo sería el interfaz de la página ya que había que elegir qué theme sería el encargado de darle estilo a la página o si había que diseñar un theme concreto porque ninguno de los existentes cumplía con lo que el cliente quería.

Tras ver los pros y contras, se optó por elegir uno de los ya creados, Andromeda, ya que cumplía perfectamente con la imagen que la empresa quería dar.

Además de la vista física que tendría la página, había que pulir un poco más lo comentado en la reunión anterior. El cliente por su parte añadía la idea de incorporar un blog a la página que le permitiese conectar con los usuarios que accediesen a ella. En dicho blog contaría historias de viajes realizados, reportajes, ideas o simples momentos en los que las imágenes eran protagonistas.

Por otra parte se consideró la idea de añadir un apartado de compras en el que el cliente podía vender fotos. Dicha idea se quedó un poco en el aire para debatirla más adelante.

Una vez determinado el diseño de las interfaces, el programador se puso a desarrollar las ideas teniendo siempre contacto continuo con la empresa para, en caso de duda o de problemas surgidos durante el desarrollo, poder consultarlo con la misma para llegar a una posible solución conjunta.

### 6.1.3. Reunión N°3: Mayo del 2011

Tras varios meses de investigación después de la segunda reunión, el programador ya tenía un primer borrador de lo que sería la página. Se le mostró al cliente la idea de poner un banner central que sirviese de bienvenida al cliente captando la atención del mismo. En dicho banner se mostrarían los servicios que la empresa oferta y servirían como link a las galerías concretas de cada servicio. Además, el cliente exigió la posibilidad de poner un apartado que permitiese al usuario acceder a los perfiles creados en las diferentes redes sociales. Por otra parte, el desarrollador necesitaba que la empresa le proporcionase fotos adecuadas para diseñar correctamente el banner (el tamaño correcto de las fotos).

Apartados como *Contacto* y *Sobre nosotros* estaban diseñados y el apartado *Galerías* todavía era un simple borrador. En esta reunión se debatió cómo iba a ser el estilo en el que se mostrasen las galerías de fotos y cómo se le iba a permitir al usuario acceder a las mismas. Además de fijar cuales serían los servicios ofrecidos cambiando los anteriores por *Bodas*, *Conciertos*, *Viajes*, *Fotos de estudio* y *Otros*.

A partir de esta reunión y hasta la siguiente, el desarrollador estuvo informándose sobre cómo llevar a cabo el diseño de las galerías pactado con el cliente. Tras mucho leer e investigar, surgió la idea de hacerlo con el módulo Colorbox, dándole de este modo un estilo más elegante al pase de diapositivas. También se eligió Nivo Slider como módulo para desarrollar el banner.

La siguiente reunión sería al cabo casi tres meses, tiempo en el que el desarrollador ya tenía más o menos terminadas las galerías de imágenes y el blog.

#### 6.1.4. Reunión N°4: Julio del 2011

Como ya hemos comentado, la mayoría de las páginas estáticas estaban más o menos determinadas. Sin embargo, al presentárselas al cliente, en el caso de las galerías de imágenes, decidió que quería cambiar el aspecto de la presentación de las mismas. La página que sirve para elegir qué galería queremos ver no le convencía del todo y se optó por cambiar los enlaces de texto por fotos que tendrían la misma función. También se decidió que el usuario no iba a tener la posibilidad de registrarse él mismo en la página web, sino que sería el administrador el que crease nuevas cuentas y las eliminase. Por tanto, la página de registro desaparecería. Otra de las páginas que desaparecería sería la página de facturas ya que la empresa decidió desechar la idea de mostrar las facturas por pantalla.

Por su parte, el fotógrafo tenía un blog personal en el que compartía vivencias en la red así que se empezó a escribir alguna de esas vivencias también en el blog de la página web para ver su correcto funcionamiento. Además, se decidió que cualquier usuario podría tener acceso al blog y tener la posibilidad de poder comentar los post que el fotógrafo publicase.

Una vez realizados estos cambios, se pasó a leer y documentarse sobre jQuery. Rondaba la idea de hacer un menú dinámico escrito en jQuery para la sección *Clientes* así que se comenzó a investigar sobre cómo introducir código jQuery dentro de los diferentes nodos de Drupal.

Además, ya se había empezado a desarrollar poco a poco la parte de *Presupuestos*. Se había optado por utilizar el módulo WebForms para hacer el formulario de selección de presupuesto. Constaría de dos partes, la primera parte sería donde el usuario indicaría sus datos personales y en la segunda parte, todo lo relacionado sobre el reportaje que quiere realizar. Una vez registrado el presupuesto en la base de datos le llegaría a la empresa y esta podría verla por pantalla o incluso imprimirlo en formato PDF.

#### 6.1.5. Reunión N°5: Agosto del 2011

Terminado el módulo de presupuestos, quedaba que la empresa nos mostrase cómo quería que fuese el formato del PDF.

Entonces, se comenzó a desarrollar todo el apartado de la sección *Clientes* en su totalidad. Se decidió que se iba a hacer un menú codificado en JavaScript que cambiaría dependiendo de cuál era el cliente que accedía a la página web. En caso de ser el usuario parte de la empresa, tendría un menú de tipo pseudoadministrador y por el contrario, si el que accedía era un simple usuario registrado cliente de la empresa, tendría un menú acorde a sus reportajes y presupuestos realizados. Además, este menú también sería dinámico porque en caso de ser un usuario registrado el que acceda a la página, el menú se cargará dependiendo de los reportaje que haya realizado.

Es decir, si el usuario ha realizado solamente un reportaje de boda, se le cargará en el apartado reportajes esa única galería. Si por el contrario ha realizado además de un reportaje de boda, un book personal, se le cargarán las dos opciones dentro del apartado *Mis reportajes*.

Se definieron también los privilegios que iba a tener el usuario registrado que forma parte de la empresa y cuáles iban a ser las funciones que va a llevar a cabo. Por tanto, se decidió que dicha persona iba a ser la encargada de administrar los usuarios, crear galerías de imágenes, administrar el blog y sus comentarios y todo el tema de presupuestos.

En lo referente a los presupuestos, la opción de que la empresa respondiese a la petición de presupuesto del cliente era una realidad pero no se tenía ninguna relación precio-servicio para poder llevar dicha acción a cabo así que, se le pidió a la empresa un documento en el que detallase los precios de los diferentes servicios dependiendo de cuáles eran los requisitos.

En esta misma reunión se decidió también deshacerse de la idea de crear un carro de la compra que permitiese la venta de imágenes porque no cumplía con la dinámica de la empresa.

### **6.1.6. Reunión N°6: Agosto del 2011**

En esta última reunión se le mostró al cliente cómo se había desarrollado toda la página web para ver si realmente cumplía lo requerido por la empresa. Ante la grata satisfacción del cliente se le brindó al desarrollador la oportunidad de seguir colaborando con la empresa en un futuro para realizar el mantenimiento de la página y hacerle frente a posibles modificaciones y requisitos nuevos que se quieran añadir.

## Capítulo 7: Conclusiones

Nunca había desarrollado un proyecto de esta envergadura, comenzar a trabajar con la idea de otra persona y desarrollar algo para cubrir todas sus necesidades y expectativas, haciendo un seguimiento completo de todos y cada uno de los apartados. Sabía que un buen planteamiento de los requisitos era vital para el ciclo de desarrollo de la aplicación, pero no creía que en realidad, fuese tan importante. Creo que el haber tenido plena cooperación por parte del cliente ha facilitado mucho el trabajo ya que en cualquier momento de incertidumbre, de error, de no saber cómo continuar ha estado siempre disponible para debatir ideas y posibles soluciones. Por ello, creo que es imprescindible tener al cliente informado continuamente sobre lo que estamos haciendo y cómo estamos desarrollándolo para que el proyecto avance sin ningún contratiempo. De echo, creo que es la base para que el trabajo final sea exitoso para ambas partes .

En lo referente a los lenguajes, nunca había utilizado un CMS. Personalmente creo, que es una ardua tarea partir desde cero con este tipo de gestor de contenidos, pero muy interesante al mismo tiempo. La curva de aprendizaje de Drupal es muy extensa, hay que invertir mucho tiempo en documentarse y leer como funciona cada uno de sus apartados. Tengo que admitir que hoy, ocho meses después de empezar a tener contacto con el mismo, sigo sin saber a la perfección cómo funciona. Su gran cantidad de módulos, teniendo en cuenta que algunos de ellos realizan tareas muy similares, requiere tener muy claro lo que se quiere conseguir para así integrar en el sistema el modulo que mejor cumpla con las necesidades. Además de lo anterior, creo es una herramienta muy potente en cuanto a importar código JavaScript se refiere, pero hay que saber transformar nuestro código en código jQuery de Drupal. Otra de las ventajas es que facilita mucho el trabajo físico, ya que se vale de theme's para determinar el aspecto de la página, aunque también hay que apuntar es muy limitado en el momento en el que decides salirte de lo que marca theme. Por tanto, creo que es útil utilizar esta herramienta en caso de que nuestro cliente no nos limite el diseño de la página y encuentre un theme que satisfaga sus deseos. Si por el contrario, la empresa tiene claro cómo quiere que sea el diseño de su página y quiere una funcionalidad muy específica, no recurriría al uso de un CMS para diseñarla.



En cuanto a los tiempo de las fases de desarrollos se refiere, en mi caso, yo misma he sido la que ha marcado estos tiempos ya que la empresa me ha otorgado total libertad en este aspecto. No obstante, creo que un buen planteamiento del periodo de cada etapa del desarrollo es muy importante. La primera de ellas, etapa en la que se analizan todos los requisitos que tendrá la aplicación, debería ser una de las más extensas y que más cuidadosamente se debe realizar, ya que un mal planteamiento en la misma puede encadenar errores durante todo el proceso. En mi caso, uno de los errores fue no tener en cuenta el proceso de aprendizaje de manejo de Drupal así como volver a utilizar herramientas olvidadas como PHP o familiarizarme con JavaScript, algo que tendré en cuenta para proyectos futuros. El tiempo requerido desde que se realiza la documentación sobre las herramientas a utilizar y el comienzo del manejo con fluidez de las mismas, es considerable. Esto en mi caso hizo que se retrasase mucho el comienzo de la programación de la página, corrección que realizaré para el siguiente proyecto.

Aún con todo esto, y teniendo en cuenta cuales eran mis capacidades iniciales para el cumplimiento de los requisitos del sistema, estoy satisfecha con lo conseguido. La empresa, a su vez, ha considerado que se han cumplido los requisitos que exigía y está totalmente satisfecha con la imagen que da la página, aunque aún quedan detalles por pulir, como por ejemplo la seguridad de la página o lo comentado en líneas futuras.

Considerando la falta de experiencia en mi caso, tanto en lenguajes como a la hora de realizar un sistema de esta amplitud y características y tratándose del primer trabajo en este ámbito de la carrera, creo que el resultado conseguido puede considerarse óptimo.

## Capítulo 8: Líneas futuras

Ante la satisfacción del cliente con el resultado, el programador pasará a formar parte de la empresa como administrador de la página web y desarrollador de posibles modificaciones y nuevos requisitos.

En la última reunión realizada con el cliente, se habló sobre lanzar la página a la red para tener una primera visión de cómo funcionaría. Por tanto, desarrollador y empresa se pusieron a buscar diferentes formas de hosting y dominios en red. Se ha tenido un primer contacto con un proveedor web pero todavía no se ha decidido nada sobre ello.

Por otro lado, la empresa ha solicitado la posibilidad de añadir Google Analytics dentro del código de la página para poder realizar un seguimiento de los usuarios que acceden al sitio web. Se trata de una solución de analítica web para empresas que proporciona información sobre el tráfico del sitio web que permite recopilar, ver y analizar datos. Al insertar un fragmento de JavaScript básico en las páginas, podemos ver qué contenido es el más visitado, cuál es el promedio de visitas de la página y la hora local de las visitas. Drupal tiene desarrollado un módulo que te permite añadirlo a tus páginas.

La empresa también ha planteado la posibilidad de enlazar mediante RSS el blog personal que tiene en Blogspot con el blog de la página web. Aunque todavía no está muy claro si se debería hacer un filtro a la hora de exportar los post o se haría una completa exportación de los mismos.

Como ya he comentado anteriormente, todavía quedan cosas por pulir y desarrollar del proyecto así que seguiré trabajando en ello.

## Capítulo 9: Bibliografía

### Referencia bibliográfica

1. Todd Tomlinson, J.K.V. (2010). *Pro Drupal 7 Development: Third Edition*, Apress
2. Tomlinson, T. (2010). *Beggining Drupal 7*, Apress.
3. Lengstorf, J. (2010). *Pro PHP and jQuery*, Apress.
4. Crockford, D. (2008). *JavaScript: The Good Parts*, O'Reilly Media, Inc.
5. Flanagan, D. (2011). *JavaScript: The Definitive Guide*, O'Reilly Media, Inc.
6. Fernando Alonso, L. M., Fco. Javier Segovia (2005). *Introducción a la Ingeniería del Software: Modelos de desarrollo de programas*, Delta Publicaciones.

### Referencia en línea

Documentación básica de la comunidad Drupal, fuentes para resolver problemas y seleccionar módulos.

1. XAMPP:
2. *Drupal*
  - 2.1. [www.drupal.org](http://www.drupal.org)
  - 2.2. *Getting Started With Drupal*
  - 2.3. *Lecciones de la comunidad*: <http://drupaldojo.com/>
  - 2.4. *Andromeda's handbook*: <http://drupal.org/node/1181286>
3. *Nivo Slider*: <http://nivo.dev7studios.com/>
4. *jQuery*
  - 4.1. <http://jquery.com/>
  - 4.2. *Manual*:
    - 4.2.1. <http://www.desarrolloweb.com/manuales/manual-jquery.html>
    - 4.2.2. <http://docs.jquery.com/Tutorials>
  - 4.3. *JavaScript y jQuery en Drupal*: <http://drupal.org/node/171213>
5. *Javascript*: <http://www.javascript.com/>
6. *Css*:
  - 6.1. <http://www.w3c.es/divulgacion/guiasbreves/hojasestilo>
  - 6.2. <http://www.w3.org/Style/CSS/>

Además de estos enlaces, también se han utilizado todos los mencionados anteriormente en el documento.

