

# Memoria Proyecto

Gestor de contenido  
para un portal de comunicación



# Contenido

Resumen	7
Abstract	9
<b>1. Introducción</b>	<b>13</b>
1.1 Justificación del proyecto	14
1.2 Caso de estudio	14
1.3 Estructura del documento	14
1.4 Descripción general	15
1.5 Objetivos	16
1.6 Requisitos	16
<b>2. Análisis</b>	<b>17</b>
2.1 Casos de uso	17
2.2 Diagrama de clases	19
2.3 Diagramas de secuencia	20
<b>3. Diseño</b>	<b>23</b>
3.1 Capa presentación	24
El Front-end .....	24
El Back-end .....	28
3.2 Capa Lógica	30
3.3 Capa de datos	32
<b>4. Implementación</b>	<b>33</b>
4.1 Tecnologías	33
4.1.1 HTML .....	33
4.1.2 CSS .....	33
4.1.3 Javascript .....	34
4.1.4 PHP .....	35
4.1.5 MySQL .....	35
4.1.6 XML .....	36
4.1.7 AJAX .....	36

<b>4.2 Herramientas</b>	<b>37</b>
<b>4.3 Detalles de implementación</b>	<b>38</b>
4.3.1 Ejemplo del Funcionamiento del buscador del index	38
4.3.2 Ejemplo creación de un microsite	41
4.3.3 La clase Microsite	45
4.3.4 Conexión a la Base de datos	47
4.3.5 Uso de Ajax	47
4.3.6 Funciones	48
<b>5. Plan de proyecto</b>	<b>49</b>
5.1 Equipo de trabajo	49
5.2 Plan de trabajo	49
5.3 Análisis de costes	50
<b>6. Trabajos futuros</b>	<b>53</b>
<b>7. Conclusiones</b>	<b>55</b>
<b>Anexos</b>	<b>57</b>
Anexo.1 Recursos utilizados	57
Anexo.2 Bibliografía	57

# Resumen

Este gestor será completamente editable por medio de menús y formularios los cuales han sido realizados con las tecnologías web actuales como HTML, PHP, MySQL. Se le ha dotado del máximo dinamismo posible, tanto en contenido como en personalización, además de realizarlo buscando su buen posicionamiento en internet.

Se busca también que el portal sea escalable, permitiendo la inserción de nuevos módulos que se puedan crear posteriormente, en el caso mostramos, una empresa de equipamiento médico.



# Abstract

This manager will be fully editable using menus and forms which have been made with current web technologies such as HTML, PHP, MySQL. It has given its the maximum possible dynamic in content and personalization, well I do it looking its good positioning on the internet.

This proposals also aims It is also that the portal is scalable, allowing the insertion of new modules that may be created later, in this case www.show, a medical equipment company.



# 1. Introducción

En la actualidad las empresas necesitan mantener una presencia en Internet constante y actualizada, son muchas las soluciones que podemos encontrar para que una empresa este presente en la red. Las páginas web mas comunes son las que partiendo de un primer enfoque que describe a nuestra empresa, podemos incluir categorías y noticias con la información deseada.

Para facilitar la actualización de dicha información existe la posibilidad mediante gestores de contenidos (CMS) como Joomla, Wordpress u otros que nos permiten introducir con cierta facilidad estos contenidos, aunque en muchas ocasiones estos usuarios necesitan una formación previa para manejarlos.

En este proyecto se pretende crear un gestor web, pero con requerimientos específicos que hacen que el uso de estos CMS no resulten la opción mas optima.

Por un lado, se busca que su uso sea fácil, permitiendo a usuarios inexpertos poder introducir contenido sin previa formación y que la información del portal este separada dando importancia a cada producto y no a la empresa en si mismo. Esto se consigue haciendo que el gestor sea capaz de crear diferentes sitios ubicados en otras tantas carpetas en el servidor, para poder direccionarlos mediante diferentes dominios si fuese necesario.

Estos diferentes sitios podrán tener una estructura diferente, según sean las necesidades de cada cliente o producto, pudiendo incluir secciones independientes y diferentes entre ellos.

El portal también dispondrá de una página principal a modo de índice, donde se mostrara un buscador donde estarán listadas las páginas publicadas.

## 1.1 Justificación del proyecto

Una empresa tiene una gran variedad de productos y quiere disponer de un medio para poder promocionarlos en Internet. Dado que sus productos son de varias líneas diferentes y dentro de cada línea hay multitud de productos y desea enfocarlos a otros tantos sectores, desea que cada uno de ellos pueda tratarse de forma independiente.

Hasta el momento actual su solución era crear distintas páginas realizadas de forma independiente en Html y publicarlas en Internet, mediante la contratación de diferentes dominios.

Nuestro proyecto le proporciona la posibilidad de poder crear estos sitios de forma unificada, mediante un solo gestor capaz de crear estos sitios web, manteniendo un sistema central para su creación y/o edición.

## 1.2 Caso de estudio

El portal es completamente configurable por lo que su aplicación puede ser utilizada para distintos tipos de empresas.

Una editorial que desea ir publicando sitios de sus diferentes publicaciones y así dado que cada una tiene su propia identidad poder comercializarlos y promocionarlos por separado. También podría ser utilizado por un centro de formación que desea promocionar sus cursos o masters, o incluso un centro comercial donde desea que cada una de las tiendas del centro puedan tener su página web independiente.

En nuestro caso se ha aplicado a una empresa que se dedica a comercializar maquinaria e instrumentación médica, los cuales quiere promocionar por separado pudiendo publicar diferente información de cada uno de ellos.

## 1.3 Estructura del documento

Esta memoria se ha estructurado de la siguiente forma:

**Introducción:** En este capítulo se describe el proyecto y se explican de modo general sus componentes, así como sus objetivos.

**Análisis:** Descripción de la estructura del proyecto y sus funcionalidades, así como los diagramas UML más significativos.

**Diseño:** Niveles de la arquitectura de la aplicación (presentación, lógico y de persistencia).

**Implementación:** Las tecnologías implementadas y descripción de las herramientas utilizadas para la creación.

**Conclusiones:** Conclusión desde un punto de vista personal sobre la aplicación y su finalidad.

**Bibliografía:** Relación de la documentación consultada, para la realización de esta memoria.

**Anexos:** Información complementaria para extender alguno de los puntos expuestos en los capítulos anteriores.

**Guía del usuario:** Descripción de cómo usar el gestor, con ayuda e imágenes para facilitar su aprendizaje.



## 1.4 Descripción general

La estructura del portal se basara en una parte pública (Front-end) y en una privada (Back-end).

El Front-end es la parte que se muestra a los usuarios, donde podrán acceder a la portada principal, que muestra una lista, mediante un buscador para acceder a los diferentes productos. También pueden acceder a los sitios directamente si se asignan diferentes dominios, esto es posible, debido a que la creación de cada uno de ellos se realizará en una carpeta independiente.

El portal, tendrá como parte principal la WEB, donde la empresa publicará sus noticias, fotos, etc. La red social donde los usuarios podrán compartir sus opiniones y comentarios, además este también será un medio donde los administradores de la web podrán enviar comunicaciones a sus usuarios.

El Back-end es el gestor, donde el usuario administrador puede crear y editar la información publicada.

The screenshot displays two main pages of the Mindray website:

- Homepage:** Shows a woman in a surgical cap, a banner about Mindray's history, and a sidebar for site search and navigation.
- Product Page (BC-6800):** Features a smiling boy, a banner with the text "Calidad excelente", a detailed product description, a video player, and a contact form.

A dashed arrow points from the "BC-6800" link in the sidebar of the homepage to the product page, indicating the creation of a separate microsite.

**Vista del Front-end**

The screenshot shows the Mindray Back-end interface with the following sections:

- Microsites:** A list of existing microsites (BC-6800, Z6 Sistema de ecografía de diagnóstico, DigiEye 760 Dual, MEC-1000, EC-2300) with delete buttons.
- Principal:** A configuration screen for the BC-6800 microsite, including fields for logo, alias, URL, and various settings.
- Posicionamiento:** A configuration screen for SEO settings, including meta descriptions and tags.

**Vista del Back-end**

## 1.5 Objetivos

El objetivo principal del proyecto es la construcción de un portal capaz de publicar productos o servicios en forma de "microsites", donde en cada uno de ellos, podamos editar los contenidos que tengamos creados, añadiendo nuevas secciones, fotografías o cualquier contenido que se necesite.

Además del objetivo principal se requiere:

- Gestor de contenido realizado a medida con interfaz amigable para facilitar el uso por parte del administrador.
- Ahorro de costes: forma eficiente y económica de distribuir la información entre los clientes, sustituyendo los medios clásicos.
- El interfaz de la Web será desarrollado para que soporte diferentes dispositivos.
- Fácil adaptación y configuración a la infraestructura tecnológica de la organización, así como gestión y manipulación de la información.
- Que esté disponible en todas las plataformas informáticas.
- Sencilla integración de información multimedia.
- Utilización de estándares públicos y abiertos, independientes de empresas externas, como puede ser TCP/IP o HTML.

## 1.6 Requisitos

- Para acceder a la web no se requiere ningún hardware específico, solamente acceso a Internet y un navegador.
- La aplicación se visualizará en la pantalla del usuario, de forma sencilla e intuitiva.
- El sistema estará disponible en todo momento.
- El sistema garantizará la seguridad del mismo.
- El sistema tan solo hará uso de los recursos demandados.
- El sistema estará monitorizado por un soporte técnico.
- El sistema debe tener control de estadísticas de visitas (google analytics).



# 2. Análisis

En este capítulo, se describe la fase de análisis del proyecto, su estructura y funcionalidad mediante diagramas UML, con lo que tendremos un modelo con los diferentes actores que interactúan con los objetos del sistema a través de las relaciones y acciones que cada uno de ellos podrá llevar a cabo.

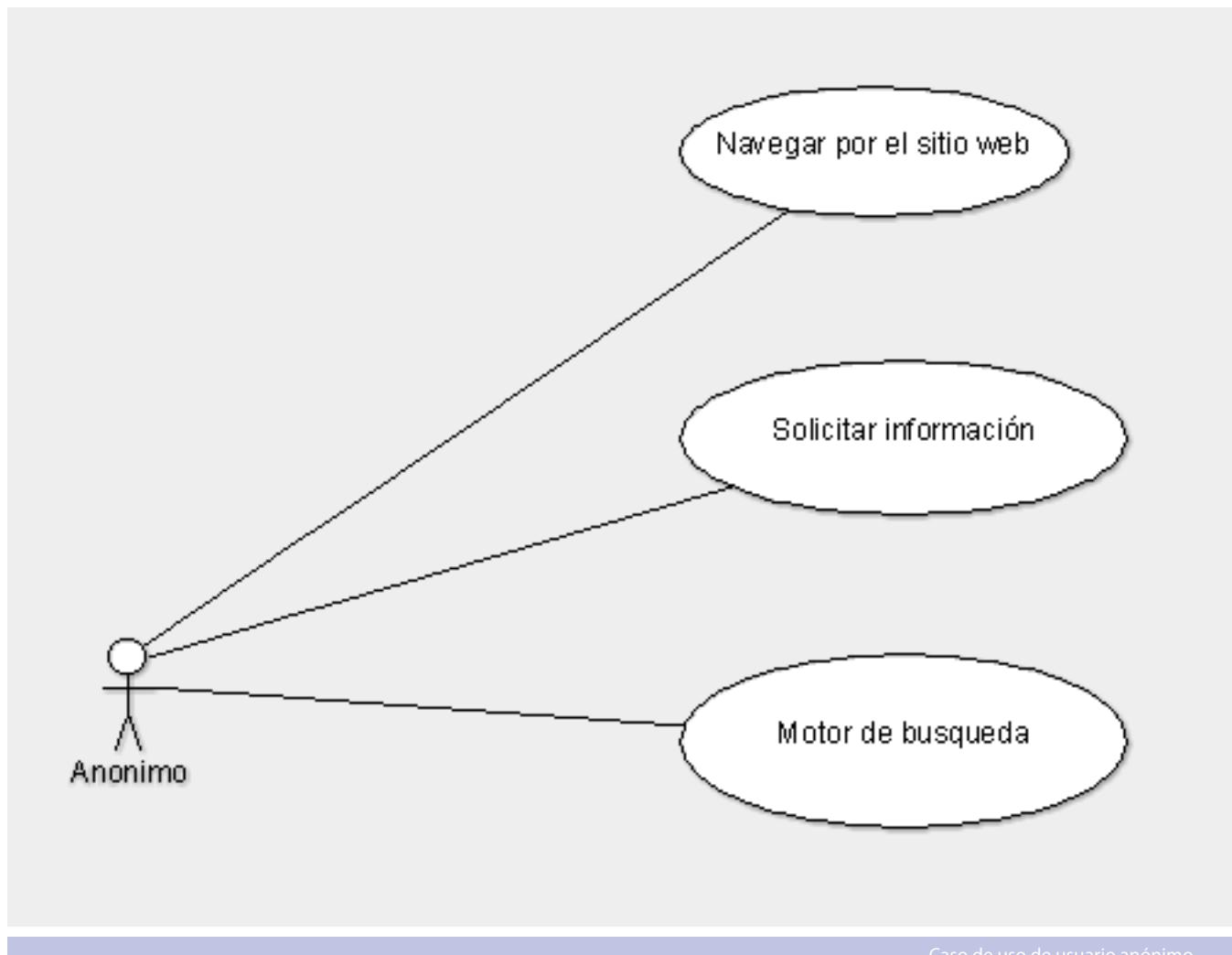
## 2.1 Casos de uso

Un caso de uso es una descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso. Los personajes o entidades que participarán en un caso de uso se denominan actores. En el contexto de ingeniería del software, un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal, sobre el propio sistema(1).

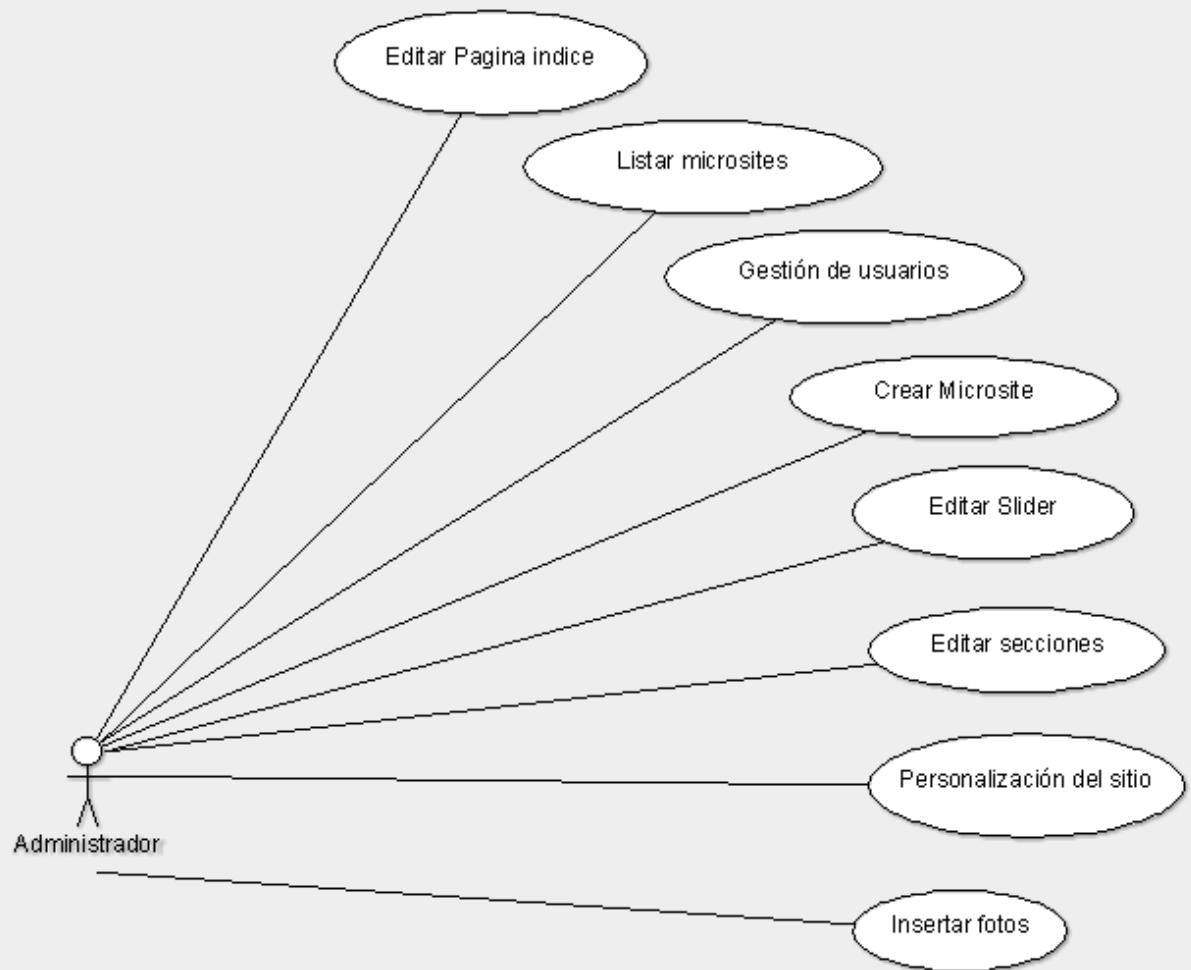
Los actores:

Actor anónimo: Usuario visitante.

Admin: Administrador del portal web.

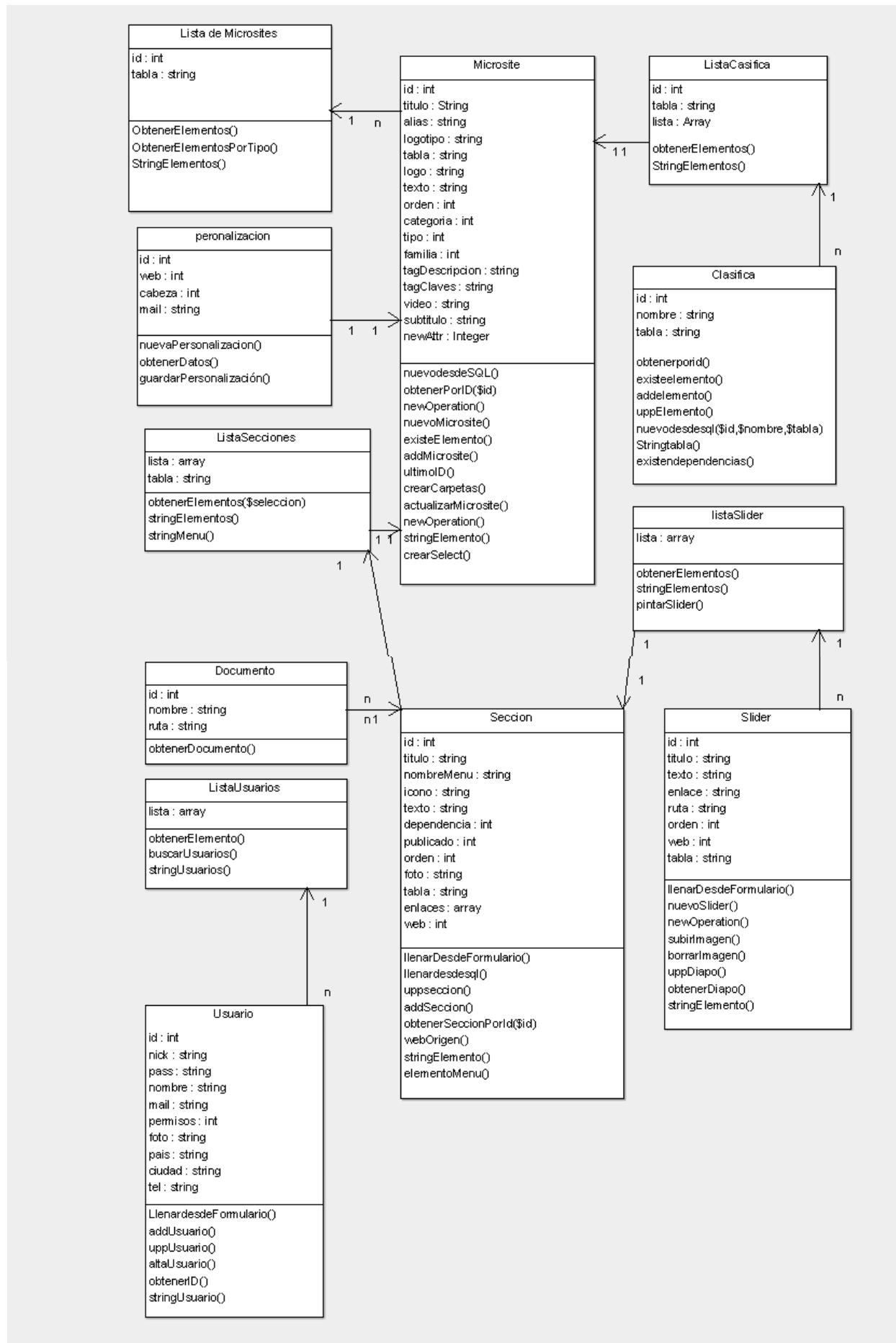


\* Fuente: wikipedia



Caso de uso de usuario administrador

## 2.2 Diagrama de clases



## 2.3 Diagramas de secuencia

Un diagrama de secuencia, muestra la interacción de un conjunto de objetos, en una aplicación a través del tiempo y se modela para cada caso de uso. A continuación mostraremos algunos procesos de nuestro portal mostrando estos diagramas.

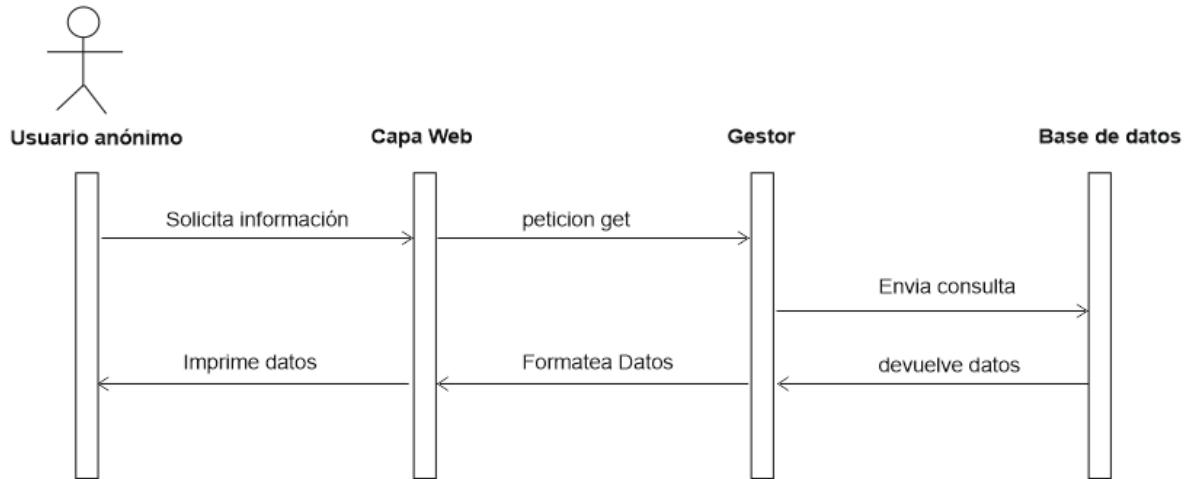


Diagrama de secuencia petición por get

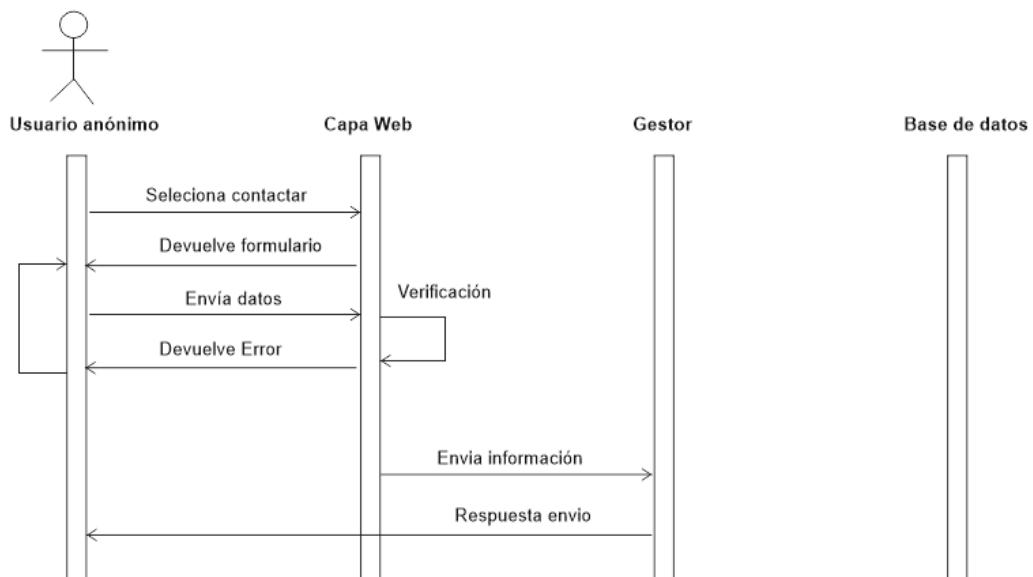


Diagrama de secuencia petición de información

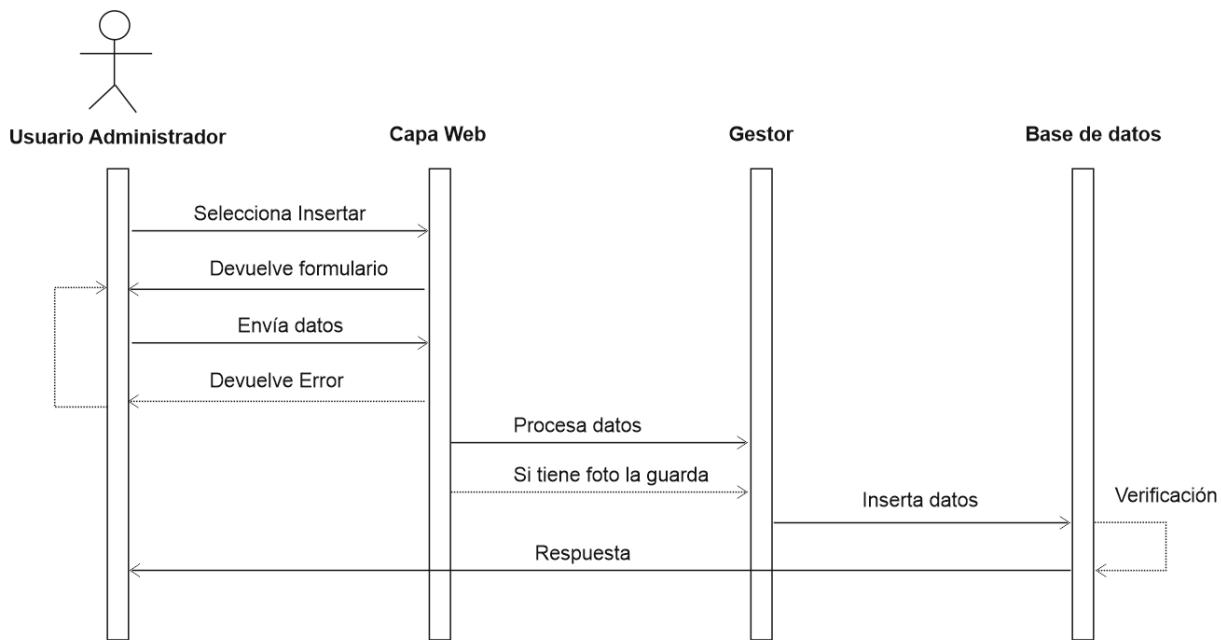


Diagrama de secuencia inserción de contenido

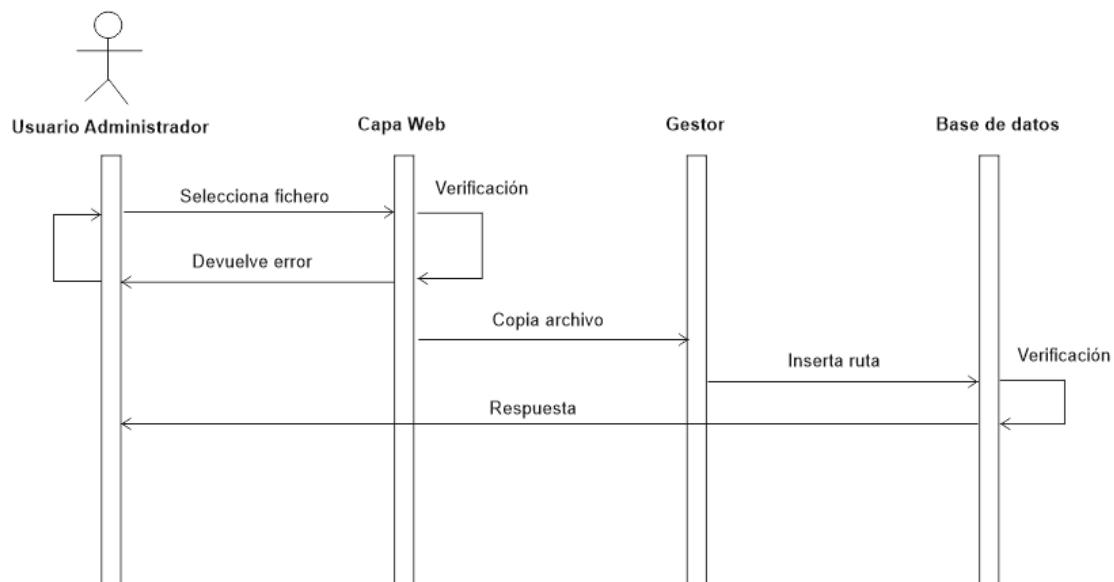


Diagrama de secuencia subir un archivo al servidor

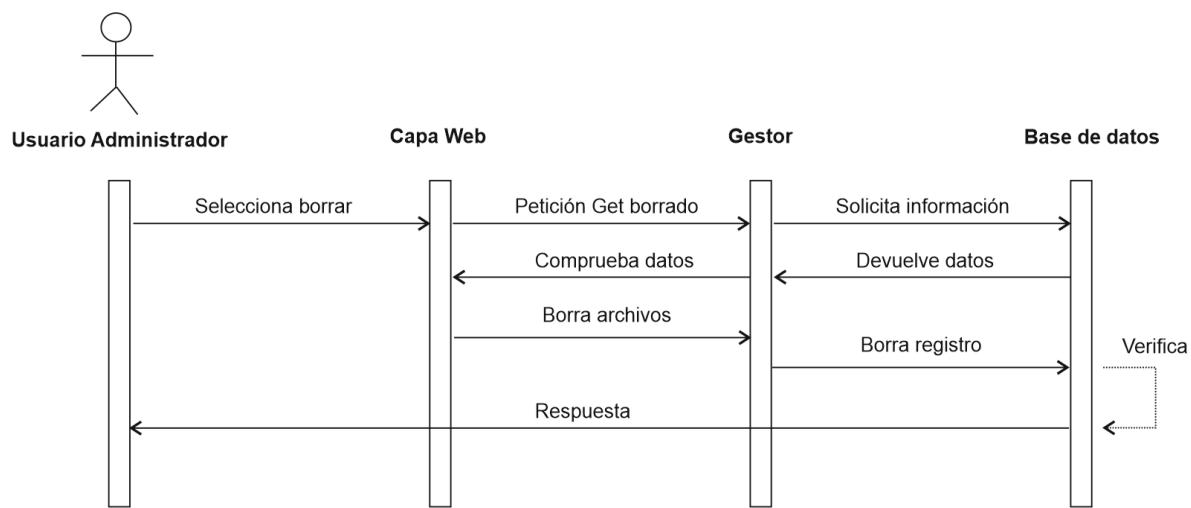


Diagrama de secuencia borrar secciones



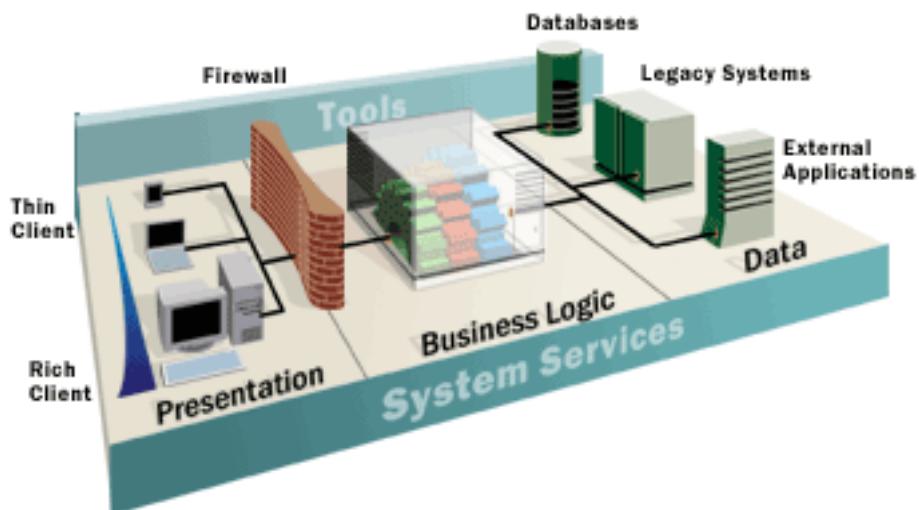
# 3. Diseño

La implementación se ha realizado por capas, esto facilita la creación y modificación del código, dado que según lo que estemos haciendo, podemos actuar directamente en los elementos afectados sin necesidad que las otras se vean afectadas.

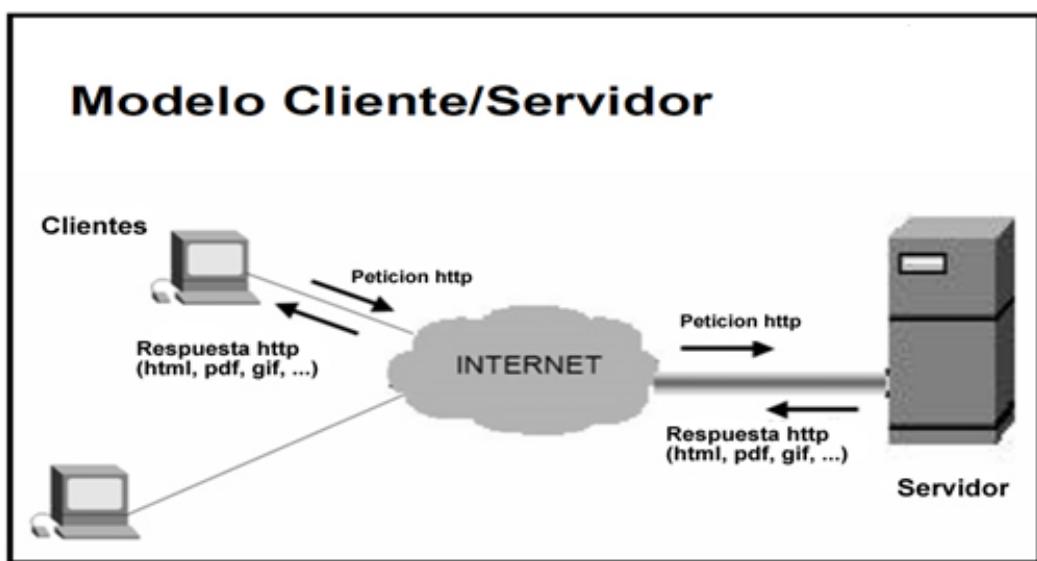
**Capa presentación:** La capa de presentación es la interfaz gráfica de nuestro sistema, esta debe ser amigable y fácil de usar. Esta capa también se denomina capa de usuario pues es lo que está visible al mismo, esta se comunica únicamente con la capa de negocio.

**Capa lógica:** Es donde reside la lógica de nuestro programa, aquí es donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación y con la de datos o persistencia. De esta última es de donde recibe los datos de nuestro gestor.

**Capa de datos:** Es donde tenemos los datos almacenados, está formada por un gestor de base de datos de donde recuperamos o almacenamos la información solicitada por la capa de negocio.



El portal está basado en una arquitectura cliente/servidor en las que las tareas se reparten entre el proveedor de servicios (El Servidor) y los demandantes llamados clientes (Terminal del usuario).



<http://eddiewebdotorg.wordpress.com/>

### 3.1 Capa presentación

La capa de presentación es la interfaz de la web. En el caso de nuestro portal, esta consta de dos partes, la parte para los visitantes o usuarios anónimos (Front-end) y la parte del gestor para los administradores del sistema (Back-end).

La interfaz debe ser intuitiva y a la vez sencilla para que así los usuarios no se pierdan. En nuestro caso como el objetivo es la creación de contenido en diferentes sitios separados entre si, esto resulta más sencillo, separando así el contenido y mostrando un simple menú con las opciones disponibles.

La codificación de la interfaz está realizada con los lenguajes Html 5, css y javascript teniendo en cuenta los distintos exploradores existentes y ofreciendo compatibilidad con antiguos exploradores que no soporten Html 5.

A continuación se describen las partes más significativas del portal.

#### El Front-end

El front-end debemos recordar que es configurable desde el gestor por lo que su aspecto se puede ver modificado, en nuestros ejemplos mostramos la demo que se ha realizado para la presentación del proyecto.

The screenshot shows the main page of the Mindray website. At the top is a navigation bar with links: Portada, Servicio, Acerca de Mindray, Red, and Contacto. Below the navigation is the Mindray logo and a large banner image of a smiling person's face. A text overlay on the banner says "banner cambiado 4" and "osdfosdfa". Below the banner is a section titled "Mindray en este siglo" with a small image of a modern building and some descriptive text. To the right of this section is a "Lista de Sitios" (List of Sites) search interface with fields for Buscador, Categoría, Familia, and Tipo, along with a "Buscar" button and a "Ver todos" link. At the bottom of the page is a footer with the Mindray logo, address "Esperanza, 38 - Bajo - 28001 Madrid - España", and phone number "Teléfono: 91000000".

Annotations on the left side:

- Menú de la página principal, aquí podemos añadir todas las secciones que deseemos, en esta página dado que es el
- Slider, en la página principal tenemos la opción de poner diapositivas para promocionar elementos, esto puede ser una buena opción para destacar productos
- Lista de los distintos micro-sites creados. Al entrar se mostrarán todos los sitios publicados, esto cambiara en cuanto usemos el buscador, pues se filtrarán según los parámetros indicados

Annotations on the right side:

- Logo principal, aquí se muestra el logo que tengamos configurado en el gestor pudiéndolo cambiar según la empresa.
- Texto de presentación. Mediante el gestor podremos publicar un texto de presentación de la empresa.
- Buscador. Mediante el buscador podremos filtrar los resultados, buscando por una palabra clave o seleccionando las distintas categorías, familias o tipos de producto.

Bottom left callout box:

En esta imagen se muestra la portada de la página principal, la cual además de tener la posibilidad de incluir textos y secciones, muestra un índice de los sitios publicados. También disponemos de un buscador para facilitar la navegabilidad.

Bottom right callout box:

Pie. En el pie de la página se mostrara la información de ubicación y contacto de la empresa, estos datos también son configurables.

## Sección página principal

Portada    Servicio    Acerca de Mindray    Red    Contacto

### mindray

En las secciones podemos incluir una imagen principal identificativa, además de poder incluir las que deseemos en el contenido.

Prompt and professional services maximize device performance



#### Política de Servicio

Como uno de los principales innovadores emergentes en el sector de equipos médicos de diagnóstico en todo el mundo, en Mindray ofrecemos a nuestros clientes un exhaustivo servicio de asistencia junto con unos productos de gran calidad a un precio asequible. Gracias a nuestra tecnología independiente y amplia experiencia en atención al cliente, en Mindray continuamos ofreciendo asistencia *in situ*, al tiempo que ayudamos a nuestros clientes a controlar los costes de mantenimiento, mitigar los riesgos médicos, ampliar su negocio en el sector y crear una medicina clínica de más valor.

##### Política de garantía

Mindray garantiza al distribuidor que los productos no tienen defectos o deficiencias en lo que respecta al diseño, los materiales y la mano de obra durante un período establecido a partir de la fecha de envío de fábrica. El distribuidor no será responsable de ningún servicio realizado a los usuarios finales.

Para los reactivos de diagnóstico *In vitro* que se encuentran dentro del período de garantía, el distribuidor será responsable de destruir los reactivos defectuosos o que no cumplen las normas, de acuerdo con las leyes y normativas locales una vez que haya recibido un aviso previo de confirmación por escrito de Mindray, así como de conservar los registros de destrucción relacionados.



##### Queda excluido de la cobertura de garantía:

- 1 Cualquier daño derivado de un uso distinto al especificado en el manual del usuario, ya sea intencionado o por error, o causado durante una reparación o mantenimiento no realizados por un representante del servicio de Mindray o un representante de Mindray autorizado;
- 2 Cualquier daño derivado de una negligencia o una realización incorrecta del mantenimiento periódico especificado por Mindray;
- 3 Cualquier situación en la que la etiqueta del número de serie original de Mindray o las marcas de identificación del producto hayan sido alteradas o eliminadas;
- 4 Los materiales consumibles especificados en los manuales de servicio, incluyendo, entre otros, papel, electrodos, elementos desechables u otros materiales de un solo uso, materiales de muestreo o adaptador, filtro, luz y sonda.
- 5 Cualquier daño derivado de un transporte o almacenamiento incorrecto.
- 6 Cualquier producto de otro fabricante.



Catálogo de servicios Mindray

### mindray

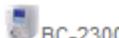
Esperanza, 38 - Bajo - 28001 Madrid - España  
Teléfono: 91000000

En las secciones además de poder incluir texto y fotos, podemos incluir la descarga de documentos, como por ejemplo catálogos de productos, manuales técnicos, etc.

Como comentábamos en la imagen anterior en la página principal podemos añadir secciones para ampliar la información de nuestra empresa.

Como ya hemos descrito anteriormente, el portal será capaz de crear diferentes sitios a los que podremos acceder desde dominios independientes o desde la pagina mostrada en las imágenes anteriores.

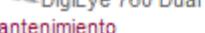
### Análisis



### Diagnóstico



Z6 Sistema de ecografía de diagnóstico

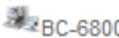


### Mantenimiento



### Documentación

### Repuesto



### Lista



### Microsite

### Portada Microsite

Buscador, en las microsites de forma opcional, podemos incluir el buscador para poder acceder a otros sitios, manteniendo así una estructura global con el resto.

Cada microsite tiene su propio slider donde incluir imágenes.

Menú donde se mostrarán las diferentes secciones creadas para la microsite. Cabe destacar que es una de las grandes ventajas de este portal, pues podemos personalizar cada microsite con las secciones que queramos, independientemente de su contenido o nombre.

Opcionalmente, podemos incluir un video en la microsite, con una demo del producto o promoción.

**BC-6800**

Calidades excelentes

Analizador automático para hematología

El innovador monitor con creaciones ecológicas IMEC de Mindray le ayuda a reducir la huella ecológica de su hospital y contribuye a la sostenibilidad del medioambiente.

Gracias al diseño optimizado del hardware, el monitor IMEC consume un 50% menos que los monitores de paciente convencionales. En este modo, la vida útil de la batería aumenta y se puede prescindir del ventilador, por lo que su funcionamiento es más eficiente y, por tanto, más ecológico. Además, su estructura es ligera y delgada pero muy resistente, por lo que su transporte resulta muy sencillo.

Sus parámetros de medición preconfigurados, que satisfacen todos los requisitos normativos clínicos, hacen posible la utilización del IMEC en distintos servicios hospitalarios, como urgencias, salas de reanimación, servicios generales y consultas externas.

- El diseño sin ventilador ayuda a reducir la contaminación acústica.
- 2,8 a 3,8 Kg de peso, lo que permite una mayor facilidad para trasladar y almacenar.
- Permite una rápida respuesta de ecogiro y un manejo de 8 bolas.
- La batería de litio-níquel permite un funcionamiento continuo máximo de 4 horas.
- Gran capacidad de almacenamiento de datos.
- Una 43 horas de grabación de ónreas de 1000.
- 120 horas de tendencias gráficas y tabulares.
- 1000 mediciones NIBP.
- 100 eventos de alarma.

El puerto USB permite la transferencia de datos del paciente a un ordenador y la copia de configuraciones personalizadas de usuario a otros IMECs.

**About Mindray**

**SOLICITUD DE INFORMACIÓN SIN COMPROMISO**

Nombre \* \_\_\_\_\_  
Apellido \* \_\_\_\_\_  
E-mail \* \_\_\_\_\_  
Teléfono \* \_\_\_\_\_  
España \_\_\_\_\_  
Selecciona provincia \* \_\_\_\_\_  
Enviar \_\_\_\_\_

**mindray**  
Esperanza, 38 - Bajo - 28001 Madrid - España  
Teléfono: 91000000

Formulario que aparece en todas las microsites para solicitar información del producto en cuestión.

En la imagen anterior mostramos el buscador. Este buscador se puede incluir o no en las microsite, dependiendo si queremos que tenga total independencia o no.

Su funcionamiento consiste en escribir una palabra clave o seleccionar unas de las categorías, familias o tipos y automáticamente nos mostrara los sitios coincidentes.

The screenshot shows a search interface with social media sharing icons (Facebook, Twitter, LinkedIn) at the top left. Below them are four dropdown menus: 'Buscador' (Search), 'Categoría' (Category), 'Familia' (Family), and 'Tipo' (Type), all set to 'Todos' (All). To the right of these is a magnifying glass icon. The 'mindray' logo is in the top right corner. A 'Cerrar' (Close) button is in the top right of the main content area. The main content area displays a list titled 'Sitios encontrados:' (Sites found) with five items: 'BC-6800', 'Z6 Sistema de ecografía de diagnóstico', 'DigiEye 760 Dual', 'MEC-1000', and 'BC-2300'. Below this list is a large thumbnail image of a medical device labeled 'BC-6800'.

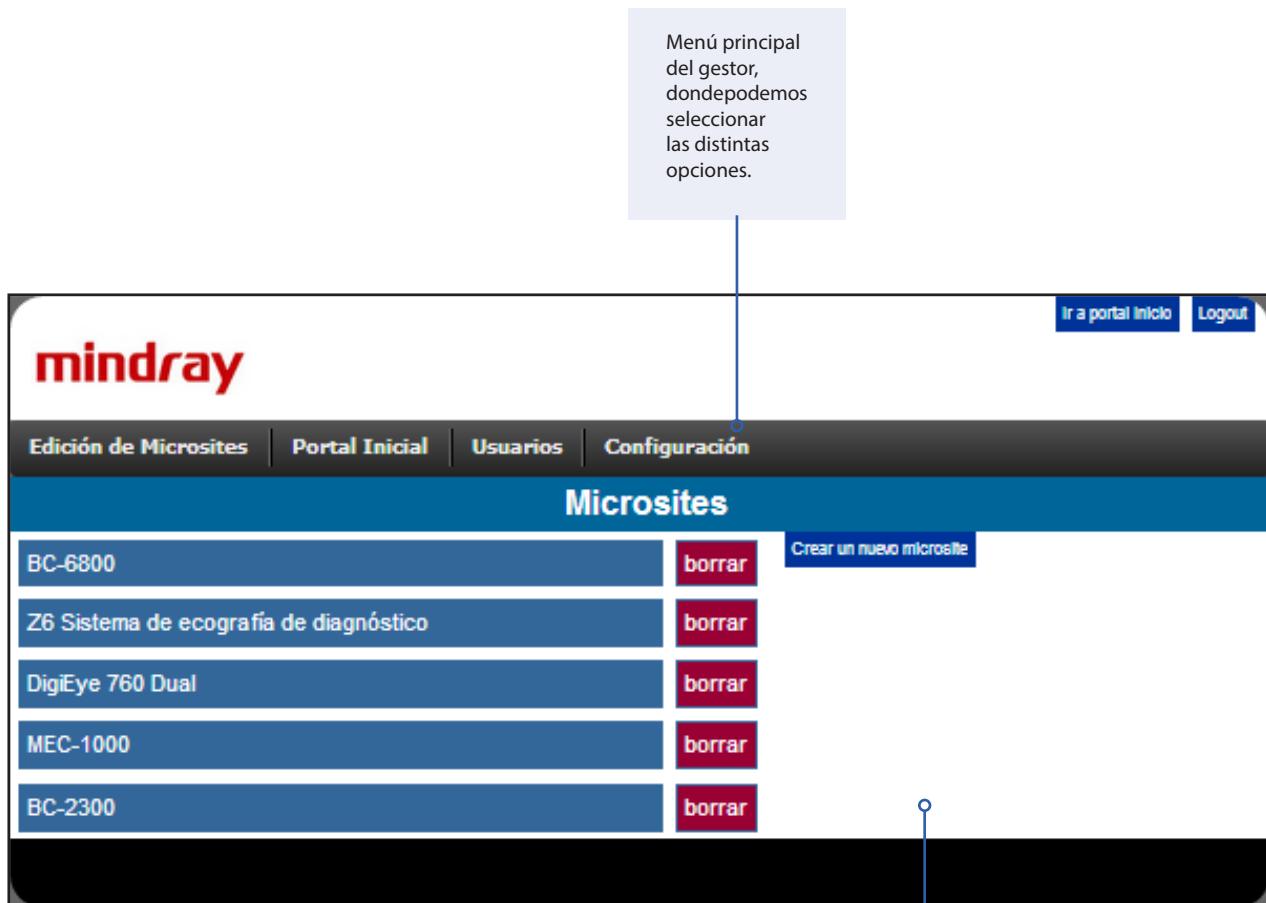
### Sección Microsite

The screenshot shows a product page for the 'BC-6800'. At the top, there are social media sharing icons and four dropdown menus: 'Buscador' (Search), 'Categoría' (Category), 'Familia' (Family), and 'Tipo' (Type), all set to 'Repuesto' (Replacement part), 'Farmaceutico' (Pharmaceutical), and 'Instalada' (Installed), respectively. The 'mindray' logo is in the top right. The main title 'BC-6800' is displayed above a large image of a medical device. Below the image is a horizontal menu bar with links: 'Portada' (Home), 'Características' (Features), 'Garantía' (Warranty), and 'Contacto' (Contact). The 'Características' section contains a heading 'Características de la maquina' (Machine characteristics) followed by a bulleted list of features. To the right of this list is a form titled 'SOLICITUD DE INFORMACIÓN SIN COMPROMISO' (Request for information without commitment) with fields for 'Nombre' (Name), 'Apellidos' (Last name), 'E-mail' (Email), 'Teléfono' (Phone), 'España' (Spain), 'Selecciona provincia' (Select province), and a 'Enviar' (Send) button. At the bottom of the page, there are two download links: 'Documento de Información 1' (Information document 1) and 'Documento Instrucciones 2' (Instructions document 2). The 'mindray' logo and address 'Esperanza, 38 - Bajo - 28001 Madrid - España' are at the bottom.

En las secciones de los microsites podemos incluir textos, fotos y descarga de documentos.

## El Back-end

El Back-end es el gestor al que solo tendrán acceso los administradores, en esta parte podrán mantener actualizado y configurado el portal mediante formularios, como principal objetivo de la creación de esta interfaz se ha buscado la simplicidad en los procesos para facilitar su uso.



Según lo que deseemos hacer, la zona central irá cambiando, mostrándonos los formularios necesarios para la creación de los contenidos. Estas opciones han sido estudiadas para que su facilidad de uso sea notable, mostrando diálogos de ayuda en todo momento.

**mindray**

Edición de Microsites | Portal Inicial | Usuarios | Configuración

### Microsites

BC-6800	<a href="#">borrar</a>
Z6 Sistema de ecografía de diagnóstico	<a href="#">borrar</a>
DigiEye 760 Dual	<a href="#">borrar</a>
MEC-1000	<a href="#">borrar</a>
BC-2300	<a href="#">borrar</a>

[Crear un nuevo microsite](#)

**mindray**

Edición de Microsites | Portal Inicial | Usuarios | Configuración

### Microsites

- [Principal](#)
- [Portada](#)
- [Slider](#)
- [Secciones](#)
- [Posicionamiento](#)
- [Galería](#)
- [Configuración](#)

**Principal**

Logo:  Ningún archivo seleccionado  
Usar Logotipo insertado para la microsite ▾

Título: BC-6800  
Alias es: Nuevo\_Microsite  
URL actual: 192.168.1.100:8080/microsites/Nuevo\_Microsite

Categoría:	Repuesto	Tipo:	Instalada	Familia:	Farmacéutico
<a href="#">Editar</a>					

**mindray**

Edición de Microsites | Portal Inicial | Usuarios | Configuración

### Microsites

- [Principal](#)
- [Portada](#)
- [Slider](#)
- [Secciones](#)
- [Posicionamiento](#)
- [Galería](#)
- [Configuración](#)

[Ver Microsite](#)

**Edición Slider**

[Añadir una Diapositiva](#)

[Editar](#) [Borrar](#)

[Editar](#) [Borrar](#)

[Editar](#) [Borrar](#)

**mindray**

Edición de Microsites | Portal Inicial | Usuarios | Configuración

### Microsites

- [Principal](#)
- [Portada](#)
- [Slider](#)
- [Secciones](#)
- [Posicionamiento](#)
- [Galería](#)
- [Configuración](#)

[Ver Microsite](#)

**Edición Slider**

**Nueva diapositiva**

Foto:  Ningún archivo seleccionado

Título: Calidades excellentes

Telón:

Enlace:

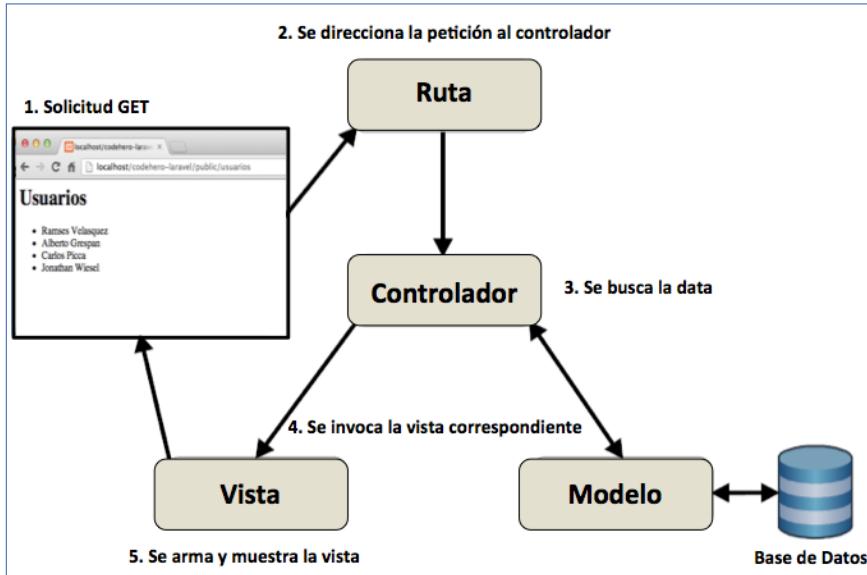
[Guardar](#) [Cancelar](#)

## 3.2 Capa Lógica

En la capa lógica se definen los comportamientos y reglas de los objetos. En este nivel recibimos las peticiones de los usuarios y se envían las respuestas una vez procesadas y obteniendo los datos necesarios de la siguiente capa de persistencia (BD).

Permite relacionar la capa de interface de usuario con la capa de datos. Aquí encontramos el código que permite llevar a cabo esta relación, además se hacen comprobaciones para validar los datos, dotando así de seguridad nuestro sistema.

En PHP el código se reparte por distintos archivos, así como las funciones de las mismas, pero siguiendo el MVC, se ha separado de modo que tenemos nuestras clases con los métodos y luego las partes controladoras de cada acción que construyen la interface.



Fuente: <http://bextlan.com/>

## Modelo

```

78 class Microsite {
79
80
81     private $id;
82     private $titulo;
83     private $alias;
84     private $logotipo;
85     private $tablaDB;
86     private $logoPrincipal;
87     private $texto;
88     private $orden;
89     private $categoria;
90     private $tipo;
91     private $familia;
92     private $tagDescripcion;
93     private $tagClaves;
94     private $video;
95     private $subtitulo;
96
97
98
99     public function __construct() {
100
101         $this->tablaDB = 'microsite';
102     }
103
104     public function getId() {
105         return $this->id;
106     }
107
108     public function getTitulo() {
109         return $this->titulo;
110     }
111
112     public function getAlias() {
113         return $this->alias;
114     }

```

```

<?php
include('includesGT/default.inc.php') ;
require_once 'funciones/class.secciones.php';

$listaSeccion = new ListaSeccion();
$listaSeccion->setTablaDB('secciones');
$seleccion = $_GET['id'];
$listaSeccion->obtenerElementos($seleccion);

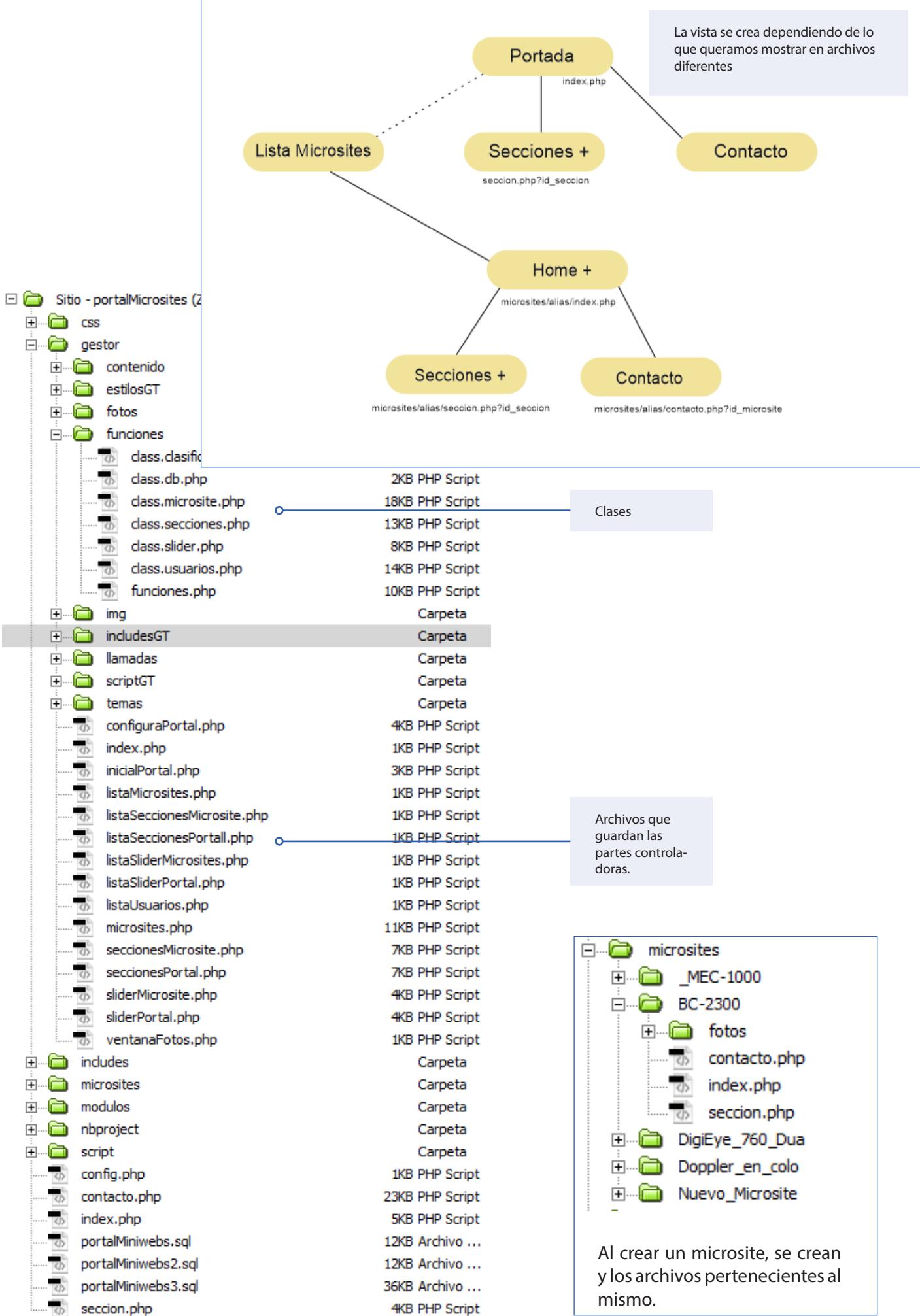
?>

<div id="subtitulo">Lista de sitios</div>
<div id="buscaWeb">
    <form name="buscador" method="post">
        <table>
            <tr>
                <td> <label>Buscador:</label><br><input type="text" name="texto" placeholder="Buscar" onchange="buscar()" /></td>
                <td> <label>Categoria:</label><?php echo $microsites::crearSelect('categoria',0) ?></td>
                <td> <label>Familia:</label><?php echo $microsites::crearSelect('familia',0) ?>
                <td> <label>Tipo:</label><?php echo $microsites::crearSelect('tipo',0) ?>
                <td><input type="button" onclick="buscar()" value='Buscar' /></td>
                <td><input type="button" onclick="window.location.reload()" value='Ver todos' /></td>
            </tr>
        </table>
    </form>
</div>
<div id="listaWeb">

```

## Controlador

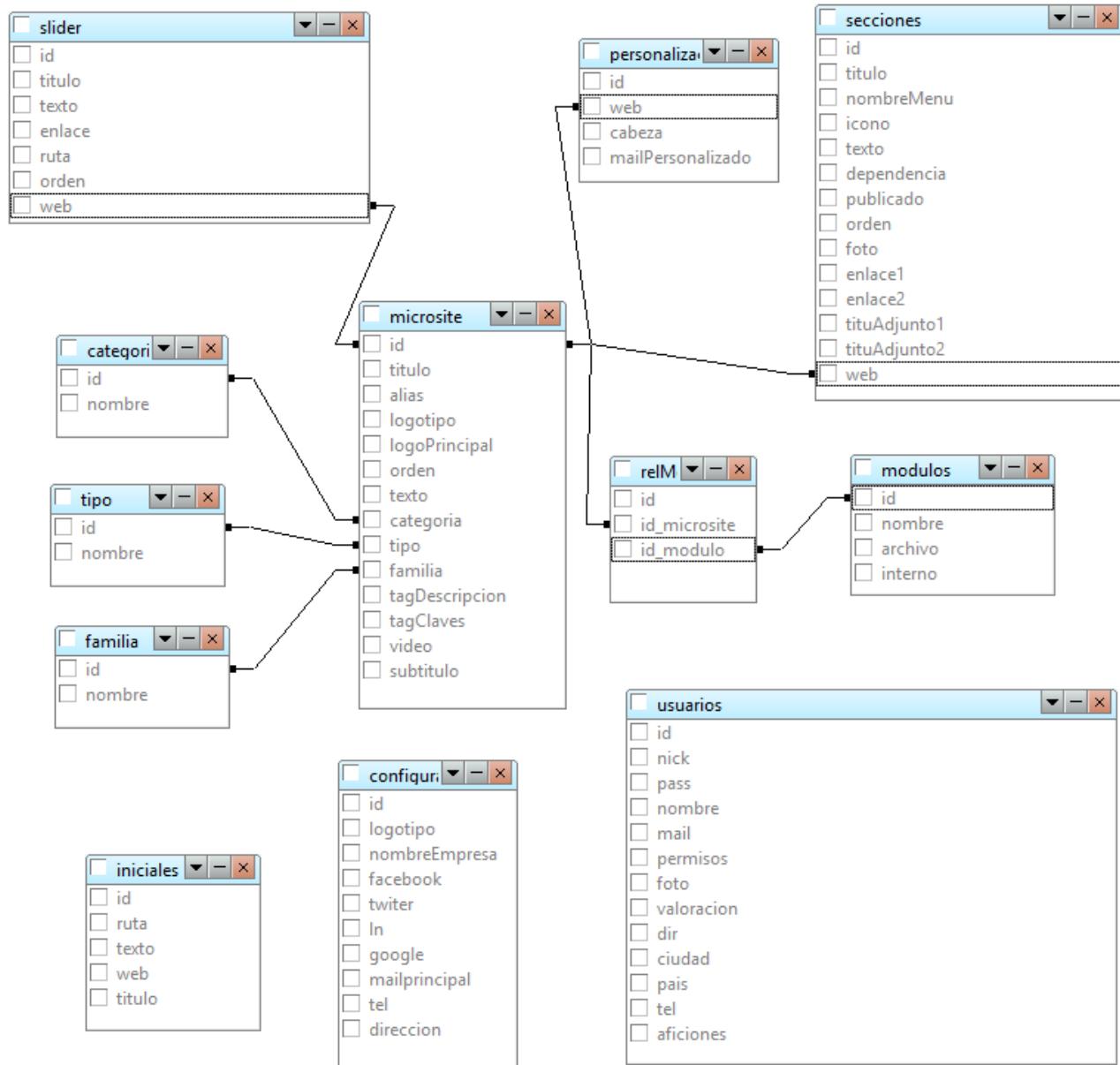
## Vista



### 3.3 Capa de datos

En la capa de datos es donde se almacena de forma persistente toda la información, en nuestro caso el gestor de base de datos utilizado es MySQL, muy utilizado en las aplicaciones web.

Nuestra aplicación hace uso de una única base de datos que contiene las tablas necesarias para poder construir los modelos lógicos de la capa anterior.



# 4. Implementación

## 4.1 Tecnologías

### 4.1.1 HTML

HTML (HyperText Markup Language) fue desarrollado por Tim Berners-Lee mientras estaba en el CERN, y popularizado con el navegador Mosaic desarrollado en el NCSA. Pero fue en los años 90 cuando se produjo la explosión de la Web.

Es un lenguaje de etiquetas usado por los diseñadores y programadores para crear páginas web, este lenguaje es un estándar reconocido por en todo el mundo y cuyas normas las define un organismo sin ánimo de lucro llamado World Wide Web Consortium (<http://www.w3.org/>).

Un documento HTML está formado por su contenido y etiquetas que delimitan sus partes pudiendo formatear la página aplicando el diseño o estructura deseada.



Actualmente HTML5 es la última actualización de este lenguaje de marcas, aunque realmente es un término que sirve para agrupar las nuevas tecnologías de desarrollo de aplicaciones web: HTML5, CSS3 y nuevas capacidades de JavaScript, sin las cuales ya no se concibe una página web.

### 4.1.2 CSS

Con las Hojas de estilo CSS (Cascading Style Sheets) podemos dotar a nuestras páginas HTML con diseños más estéticos, mas hoy en día que con HTML5, la tendencia es usar el menor número de atributos, dejando así todo lo que se refiere a la maquetación, aspecto, diseño, etc. a CSS.

Con las hojas de estilo no solo podremos hacer que nuestros textos y fotos se vean del modo que deseemos, además controlaremos toda la estructura de nuestra página, menús, columnas, etc.

## 4.1.3 Javascript

En contra de los lenguajes vistos hasta ahora, JavaScript (abreviado comúnmente "JS") es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas<sup>4</sup> aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

Con él conseguiremos mayor interacción entre las páginas y podemos usarlo junto a HTML creando HTML dinámico DHTML.

Otra de las particularidades de javascript es su flexibilidad, así podemos incluirlo en el código hasta en 3 formas diferentes para crear la página.

En el mismo documento,

```
<html>
  <head>
    <title>JavaScript en el propio documento</title>
    <script type="text/javascript">
      alert("Esto es un mensaje en una ventada del navegador");
    </script>
  </head>
  <body>
    <p>Un párrafo de texto.</p>
  </body>
</html>
```

O en un documento externo enlazándolo mediante la etiqueta <script>.

```
<html>
  <head>
    <title>Ejemplo de código JavaScript en el propio documento</title>
    <script type="text/javascript" src="archivoJavaScript.js"></script>
  </head>
  <body>
    <p>Un párrafo de texto.</p>
  </body>
</html>
```

Si lo introducimos en un documento externo debemos tener en cuenta que hay que definir el atributo src, que nos indicará la URL que se quiere enlazar



## 4.1.4 PHP

PHP es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.

Un ejemplo sería:

```
<html>
  <head>
    <title>Ejemplo</title>
  </head>

  <body>
    <?php
      echo "Hola, soy PHP";
    ?>
  </body>
</html>
```

El código de PHP, siempre estará delimitado por las etiquetas de inicio y final `<?php` y `?>` que permiten entrar y salir del modo PHP.

Una de las cualidades de PHP es la facilidad de su uso para principiantes, pero a la vez ofrece muchas características avanzadas para los programadores más experimentados o profesionales.

Lo que lo distingue de otros lenguajes como JS es que este último está del lado del cliente, mientras que PHP es ejecutado en el servidor, creando un HTML, que posteriormente será enviado al cliente. Así el cliente recibe el script, pero nunca sabrá el código subyacente del que procede.

## 4.1.5 MySQL

Es un sistema de base de datos operacional. MySQL es hoy en día uno de los más importantes en lo que hace al diseño y PROGRAMACIÓN DE BASE DE DATOS de tipo relacional. Cuenta con millones de aplicaciones y aparece en el mundo informático como una de las más utilizadas por usuarios del medio. El programa MySQL se usa como servidor a través del cual pueden conectarse múltiples usuarios y utilizarlo al mismo tiempo.

La historia del MySQL (cuya sigla en inglés se traslada a My Structured Query Language o Lenguaje de Consulta Estructurado) se remite a principios de la década de 1980. Programadores de IBM lo desarrollaron para contar con un código de programación que permitiera generar múltiples y extendidas bases de datos para empresas y organizaciones de diferente tipo. Desde esta época numerosas versiones han surgido y muchas de ellas fueron de gran importancia. Hoy en día MySQL es desarrollado por la empresa Sun Microsystems.

Una de las características más interesantes de MySQL es que permite recurrir a bases de datos multiusuario a través de la web y en diferentes lenguajes de programación que se adaptan a diferentes necesidades y requerimientos. Por otro lado, MySQL es conocida por desarrollar alta velocidad en la búsqueda de datos e información, a diferencia de sistemas anteriores. Las plataformas que utiliza son de variado tipo y entre ellas podemos mencionar LAMP, MAMP, SAMP, BAMP y WAMP (aplicables a Mac, Windows, Linux, BSD, Open Solaris, Perl y Python entre otras).

Otra de las características de MySQL es que las sentencias pueden formar parte de un programa o bien ejecutarse sobre un terminal.

En el primer caso, las sentencias SQL se ejecutarán cuando se ejecute el programa mientras que en el segundo caso las sentencias serán introducidas por teclado.

Estas sentencias serán enviadas al sistema gestor de bases de datos para que las ejecute. Posteriormente este sistema accederá a los datos, realizará las operaciones expresadas en la sentencia SQL, y, una vez ejecutadas, devolverá el resultado al terminal de origen para que las muestre al usuario.

## 4.1.6 XML

XML (Extensible Markup Language) es un lenguaje de etiquetas, es decir, cada paquete de información está delimitado por dos etiquetas como se hace también en el lenguaje HTML, pero XML separa el contenido de la presentación. Explicaremos esto con el siguiente ejemplo:

```
<H1>Mateo</H1>      <--- HTML  
<Nombre>Mateo</Nombre>  <--- XML
```

<H1> y <Nombre> son etiquetas. Ambas encierran el texto o paquete de información "Mateo". La etiqueta <H1> es de HTML, y se encarga de mostrar visualmente el texto "Mateo" en la página web en un tamaño determinado pero no dice nada del significado de "Mateo": si es una ciudad o un nombre, por ejemplo. En cambio la etiqueta <Nombre> es de XML y nos dice que "Mateo" es un nombre de persona, por lo tanto XML se preocupa del significado del texto que encierra y no de la apariencia de cómo se muestre el texto en la página web. Por eso se dice que XML es un lenguaje de etiquetas, que como hemos dicho anteriormente, separa el contenido de la presentación. Lo mismo se puede definir el lenguaje XML usando palabras más técnicas pero con el mismo significado que la definición anterior:

"XML describe el sentido semántico de los datos dejando de lado la presentación".

## 4.1.7 AJAX

AJAX (Asynchronous JavaScript and XML).

AJAX lo podemos usar para realizar peticiones HTTP asíncronas.

Así, cuando se visualiza una página que dispone de AJAX, la comunicación con el servidor HTTP queda abierta en segundo plano, lanzando los objetos que la página requiere. Esto es muy útil, porque podremos variar la página, sin necesidad de ir recargándola.

Esto ha resultado, borrar-  
dos, etc.



## 4.2 Herramientas

Para la realización del proyecto se han utilizado diversas herramientas para la edición del código y para gestionar la base de datos.

Las herramientas utilizadas principalmente son:

XAMPP: Programa para emular un servidor de forma local, y así poder desarrollar el proyecto.

Dreamweaver: Editor para escribir el código de nuestra aplicación, con asistentes para el desarrollo y gestión de archivos.

Sublime-text: Editor de textos con asistentes de código, usado principalmente en la codificación de PHP.

PhpMyAdmin: Gestor de bases de datos integrado en los servidores

## Photoshop: Programa para el tratamiento de imágenes.

Exploradores: Para realizar pruebas y visualizar los contenidos.



DW Archivo Edición Ver Insertar Modificar Formato Comandos Sitio Ventana Ayuda

Código Fuente Díndir Diseño En vivo Título: Gestor

microsites.php x dassmicrosite.php x seccioneMicrosite.php x portalMicrosites 2: portalMicroSites/gestor/seccionesMicrosite.php

Este página puede tener archivos relacionados dinámicamente que sólo el servidor puede detectar. Detalles | Preferencias

```
4
5
6 $sección = new Sección();
7 $sección->setTablaDB('secciones');
8 $modo = '';
9 $sección->setWeb($_GET['web']);
10 if(isset($_GET['id']) && !empty($_GET['id'])){
11     $id = intval($_GET['id']);
12     $sección->obtenerSecciónPorId($id);
13 }
14
15 }
16
17
18
19
20
21 if (isset($_POST) && !empty($_POST)) {
22
23     foreach ($_POST as $clave => $valor) {
24         $clave = addslashes($valor);
25     }
26
27     if(isset($_POST['publicado']) && $_POST['publicado'] == "yes") {
28         $publicado = 1;
29     }else{
30         $publicado = 0;
31     }
32
33     $enlace1 = $_FILES['enlace1'];
34     $enlace2 = $_FILES['enlace2'];
35
36
37
38 //ACTUALIZAR
39 if (isset($_POST['id']) && !empty($_POST['id'])) {
40
41     $id = intval($_POST['id']);
42
43     $sección->updSección($id, $título, $nombreMenu, $_FILES['icono'], $texto, $dependencia, $publicado, $_FILES['foto'],
44     $enlace1, $enlace2, $títuloAdjunto, $títuloAdjunto2);
45
46     $webDorigen = $sección->webDorigen();
47     header('Location:listaSeccionesMicrosite.php?id='.$webDorigen);
48
49
50
51
52
53
54
55
56
57
58
59
59
60
61
62
63
64
65
66
67
68
69
69
70
71
72
73
74
75
76
77
78
79
79
80
81
82
83
84
85
86
87
87
88
89
89
90
91
92
93
94
95
96
96
97
98
99
99
100
101
102
103
104
105
106
107
108
109
109
110
111
112
113
114
115
116
117
118
119
119
120
121
122
123
124
125
126
127
128
129
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
169
170
171
172
173
174
175
176
177
178
179
179
180
181
182
183
184
185
186
187
188
189
189
190
191
192
193
194
195
196
197
198
199
199
200
201
202
203
204
205
206
207
208
209
209
210
211
212
213
214
215
216
217
218
219
219
220
221
222
223
224
225
226
227
228
229
229
230
231
232
233
234
235
236
237
238
239
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
287
288
289
289
290
291
292
293
294
295
296
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
318
319
319
320
321
322
323
324
325
326
327
328
329
329
330
331
332
333
334
335
336
337
338
339
339
340
341
342
343
344
345
346
347
348
349
349
350
351
352
353
354
355
356
357
358
359
359
360
361
362
363
364
365
366
367
367
368
369
369
370
371
372
373
374
375
376
377
377
378
379
379
380
381
382
383
384
385
386
387
387
388
389
389
390
391
392
393
394
395
396
397
398
399
399
400
401
402
403
404
405
406
407
408
409
409
410
411
412
413
414
415
416
417
418
419
419
420
421
422
423
424
425
426
427
428
429
429
430
431
432
433
434
435
436
437
438
439
439
440
441
442
443
444
445
446
447
448
449
449
450
451
452
453
454
455
456
457
458
459
459
460
461
462
463
464
465
466
467
468
469
469
470
471
472
473
474
475
476
477
478
479
479
480
481
482
483
484
485
486
487
487
488
489
489
490
491
492
493
494
495
496
497
498
499
499
500
501
502
503
504
505
506
507
508
509
509
510
511
512
513
514
515
516
517
518
519
519
520
521
522
523
524
525
526
527
528
529
529
530
531
532
533
534
535
536
537
538
539
539
540
541
542
543
544
545
546
547
548
549
549
550
551
552
553
554
555
556
557
558
559
559
560
561
562
563
564
565
566
567
568
569
569
570
571
572
573
574
575
576
577
578
579
579
580
581
582
583
584
585
586
587
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
618
619
619
620
621
622
623
624
625
626
627
628
629
629
630
631
632
633
634
635
636
637
638
639
639
640
641
642
643
644
645
646
647
648
649
649
650
651
652
653
654
655
656
657
658
659
659
660
661
662
663
664
665
666
667
668
669
669
670
671
672
673
674
675
676
677
678
679
679
680
681
682
683
684
685
686
687
687
688
689
689
690
691
692
693
694
695
696
697
697
698
699
699
700
701
702
703
704
705
706
707
708
709
709
710
711
712
713
714
715
715
716
717
718
719
719
720
721
722
723
724
725
726
727
728
729
729
730
731
732
733
734
735
736
737
738
739
739
740
741
742
743
744
745
746
747
748
749
749
750
751
752
753
754
755
756
757
758
759
759
760
761
762
763
764
765
766
767
768
769
769
770
771
772
773
774
775
776
777
778
779
779
780
781
782
783
784
785
786
787
787
788
789
789
790
791
792
793
794
795
796
797
797
798
799
799
800
801
802
803
804
805
806
807
808
809
809
810
811
812
813
814
815
815
816
817
818
819
819
820
821
822
823
824
825
826
827
828
829
829
830
831
832
833
834
835
836
837
838
838
839
839
840
841
842
843
844
845
846
847
848
848
849
849
850
851
852
853
854
855
856
857
858
859
859
860
861
862
863
864
865
866
867
868
869
869
870
871
872
873
874
875
876
877
878
879
879
880
881
882
883
884
885
886
887
887
888
889
889
890
891
892
893
894
895
896
897
897
898
899
899
900
901
902
903
904
905
906
907
908
909
909
910
911
912
913
914
915
915
916
917
918
919
919
920
921
922
923
924
925
926
927
928
929
929
930
931
932
933
934
935
936
937
938
938
939
939
940
941
942
943
944
945
946
947
948
948
949
949
950
951
952
953
954
955
956
957
958
959
959
960
961
962
963
964
965
966
967
968
968
969
969
970
971
972
973
974
975
976
977
978
979
979
980
981
982
983
984
985
986
987
987
988
989
989
990
991
992
993
994
995
996
997
997
998
999
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1009
1010
1011
1012
1013
1014
1015
1015
1016
1017
1018
1019
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1038
1039
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1048
1049
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1068
1069
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1078
1079
1079
1080
1081
1082
1083
1084
1085
1086
1087
1087
1088
1089
1089
1090
1091
1092
1093
1094
1095
1095
1096
1096
1097
1098
1099
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1138
1139
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1148
1149
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1168
1169
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1178
1179
1179
1180
1181
1182
1183
1184
1185
1186
1187
1187
1188
1189
1189
1190
1191
1192
1193
1194
1195
1195
1196
1196
1197
1198
1199
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1218
1219
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1238
1239
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1248
1249
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1268
1269
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1278
1279
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1288
1289
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1298
1299
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1318
1319
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1338
1339
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1348
1349
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1368
1369
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1378
1379
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1388
1389
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1398
1399
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1418
1419
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1428
1429
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1438
1439
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1448
1449
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1468
1469
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1478
1479
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1488
1489
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1498
1499
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1518
1519
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1528
1529
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1538
1539
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1548
1549
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1568
1569
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1578
1579
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1588
1589
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1598
1599
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1618
1619
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1628
1629
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1638
1639
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1648
1649
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1668
1669
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1678
1679
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1688
1689
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1698
1699
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1718
1719
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1728
1729
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1738
1739
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1748
1749
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1768
1769
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1778
1779
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1788
1789
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1798
1799
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1818
1819
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1828
1829
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1838
1839
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1848
1849
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1868
1869
1869
1870
1871
1872
1873
1874
1875
1876
1877
1877
1878
1878
1879
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1888
1889
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1898
1899
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1918
1919
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1928
1929
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1938
1939
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1948
1949
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1968
1969
1969
1970
1971
1972
1973
1974
1975
1976
1977
1977
1978
1978
1979
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1988
1989
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1998
1999
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2138
2139
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2148
2149
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2198
2199
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2218
2219
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2248
2249
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2278
2279
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2298
2299
2299
2300
2301
2302
2303
2304
2305
23
```

Dreamweaver

## 4.3 Detalles de implementación

Dado que mostrar todo el código en la aplicación en esta memoria sería algo tedioso, a continuación se muestran y describen algunas partes de la codificación que resultan interesantes.

### 4.3.1 Ejemplo del Funcionamiento del buscador del index



```
<?php

include 'gestor/funciones/class.db.php';
include 'config.php';
include 'gestor/funciones/funciones.php';
include 'gestor/funciones/class.slider.php';
include 'gestor/funciones/class.secciones.php';
require_once 'gestor/funciones/class.microsite.php';

$microsites = new ListaMicrosite();
$microsites->obtenerElementos();

$sql = "SELECT * FROM iniciales WHERE id=1";

$conexion = new Bd();
$res = $conexion->consulta($sql);

?>
```

En la parte controladora de esta página, incluimos los archivos necesarios que necesitamos, además de importar algunos datos iniciales.

Creamos el objeto ListaMicrosite que nos proporcionara todos los metodos para poder construir la lista de microsites que tiene nuestro sistema.

```
$txt= "<ul>";
$conexion = new Bd();

$sql = "SELECT id, nombre FROM categoria";
$resCat = $conexion->consultaLista($sql);

while(list($idCat,$nombreCat) = mysqli_fetch_array($resCat)){
    $txt .= "<li>$nombreCat";
    $txt .= "<ul>";
    $sql = "SELECT id, titulo, alias,logotipo FROM microsite WHERE categoria=".$idCat;
    $res = $conexion->consultaLista($sql);

    while(list($id,$titulo,$alias,$logotipo) = mysqli_fetch_array($res)){
        $txt .= "<li><img src='microsites/".$alias."/fotos/".$logotipo."' width='30px'>".$alias."/'.>". $titulo."</a>";
    }
    $txt .= "</ul>";
}
```

La lista es construida mediante bucles, mostrando inicialmente todos, mediante el buscador se pueden filtrar estas búsquedas, lo que irá remplazando la lista inicial, para no tener que refrescar la página al ejecutar estos filtros se realizó una codificación que mediante AJAX podemos realizar las consultas al servidor con los nuevos parametros y así crear el dinamismo.

Buscador:	Categoría	Familia	Tipo	<input type="button" value="Buscar"/>	<input type="button" value="Ver todos"/>
<input placeholder="Buscar" type="text"/>	<input type="button" value="Todos"/>	<input type="button" value="Todos"/>	<input type="button" value="Todos"/>		

```

<div id='buscaWeb'>
    <form name='buscador' method="post">
        <table>
            <tr>
                <td> <label>Buscador:</label><br><input type="text" name='texto' placeholder="Buscar" onchange=>
            </td>
                <td> <label>Categoría</label><?php echo $microsites::crearSelect('categoria',0) ?></td>
                <td><label>Familia</label><?php echo $microsites::crearSelect('familia',0) ?></td>
                <td><label>Tipo</label><?php echo $microsites::crearSelect('tipo',0) ?></td>
                <td><input type="button" onclick="buscar()" value='Buscar' /></td>
                <td><input type="button" onclick="window.location.reload()" value='Ver todos' /></td>
            </tr>
        </table>
    </form>
</div>

```

Dado que la lista de categorías, familias o tipos es editable, lo primero que tenemos que hacer es que estos elementos select de html se construyan con los datos correctos, para ello hemos creado el método crearSelect().

```

public function crearSelect($tabla,$elem ) {
    $txt = '<div id="'.$tabla.'"><select name="'.$tabla.'"'>
        <option value="0">Todos</option>';
    $conexion = new Bd();
    $sql = "SELECT * FROM ".$tabla." ORDER BY nombre ASC";
    $res = $conexion->consultaLista($sql);
    while (list($id, $nombre) = mysqli_fetch_array($res) ) {
        if($elem == $id){
            $txt .= "<option value='".$id."' selected>".$nombre."</option> ";
        }else{
            $txt .= "<option value='".$id."'>".$nombre."</option> ";
        }
    }
    $txt .= "</select></div>";

    return $txt;
}

```

Una vez tenemos las opciones del buscador tenemos que generar los eventos necesarios en javascript para que al realizar algún cambio en el mismo se haga la petición por Ajax para cambiar los datos en la página.

```

window.onload = function() {
    var contador = document.getElementsByTagName('select').length;
    for (i = 0; contador - 1; i++)
        document.getElementsByTagName('select')[i].onchange = buscar;

}

```

De este modo cada vez que se produzca un cambio en uno de los select se llamará a la función javascript buscar, que es la lanzadera de la petición Ajax.

```

function buscar() {
    var texto = ""
    var categoria, familia, tipo;

    texto = document.buscador.texto.value;
    categoria = document.buscador.categoría.value;
    familia = document.buscador.familia.value;
    tipo = document.buscador.tipo.value;

    busqueda(texto, categoria, familia, tipo);

}

```

Ahora veamos como se realiza la petición, podemos observar como las funciones javascript generan una ruta con los parámetros necesarios para que php construya el nuevo resultado y se lo devuelva a javascript para que lo pinte en la página.

```

function busqueda(texto, categoria, familia, tipo) {
    if (principal) {
        var myurl = 'gestor/llamadas/buscador.php';
        var ubica = 1;
    } else {
        var myurl = '../../../../../gestor/llamadas/buscador.php';
        var ubica = 0;
    }
    myRand = parseInt(Math.random() * 9999999999999999);
    modurl = myurl + '?rand=' + myRand + '&texto=' + texto + '&categoria=' + categoria + '&familia=' + familia + '&tipo=' +
    tipo + '&ubica=' + ubica;
    busca.open("GET", modurl, true);
    busca.onreadystatechange = busquedaResponse;
    busca.send(null);
}

function busquedaResponse() {
    if (busca.readyState == 4) {
        if (busca.status == 200) {
            var miTexto = busca.responseText;
            if(!principal){
                document.getElementById('resultadoBusqueda').style.display = "block";
            }
            document.getElementById('listaWeb').innerHTML = miTexto;
            //window.location.reload()
        }
    }
}

```

Podemos observar como llamamos a una página de PHP del servidor y luego cómo recuperamos el resultado con "responseText", para luego pintarlo en la página html mediante el id del elemento.

La página que llamamos mediante la función es la encargada de recibir los datos por Get en este caso y realizar la consulta en la Base de datos.

```

$ubica = intval($_GET['ubica']);

$texto = $_GET['texto'];
$categoría = $_GET['categoria'];
$familia = $_GET['familia'];
$tipo = $_GET['tipo'];

$ampliaSQL = "";
if($categoria != 0)
    $ampliaSQL .= "AND categoria='".$categoria."'";
if($familia != 0)
    $ampliaSQL .= "AND familia='".$familia."'";
if($tipo != 0)
    $ampliaSQL .= "AND tipo='".$tipo."' ";

$conexion = new Bd();
$txt = "<ul>";
$sql = "SELECT id, titulo, alias,logotipo FROM microsite WHERE titulo LIKE '%".$texto."%' ".$ampliaSQL."";;

$res = $conexion->consultaLista($sql);
$elementos = $conexion->numeroElementos();
while(list($id,$titulo,$alias,$logotipo) = mysqli_fetch_array($res)){
    if($ubica == 0){

        $txt .= "<li><img src='../../".$alias."/fotos/".$logotipo."' width='30' /><a href='../../".$alias."/'>".$titulo."</a>";
    }else{

        $txt .= "<li><img src='microsites/".$alias."/fotos/".$logotipo."' width='30' /><a href='microsites/".$alias."/'>".$titulo."</a>";
    }
    $txt .= "</li>";
}
if($elementos<1){
    $txt = "No se encontraron sitios con esos parámetros.";
}

echo $txt;

```

## 4.3.2 Ejemplo creación de un microsite

**Microsites**

**Principal** Ver Microsite

**Principal**  
Rellene este formulario con los datos de la nueva web y despues del guardado apareceran el resto de opciones

Logotipo:  
 Ningún archivo seleccionado

Título:

Alias:



Cuando creamos un microsite, lo primero que nos solicita el sistema es la inserción de un título y una imagen, este primer paso creara la estructura de carpetas en el servidor.

```
private function crearCarpeta(){

    if(mkdir("../microsites/".$this->alias, 0700)){

        /* Creacion de subcarpetas */
        mkdir("../microsites/".$this->alias."/fotos", 0700);

        /*crea archivo index*/
        $index = fopen("./temas/plantilla_index.php", 'r');
        $textoIndex = stream_get_contents($index);
        fclose($index);

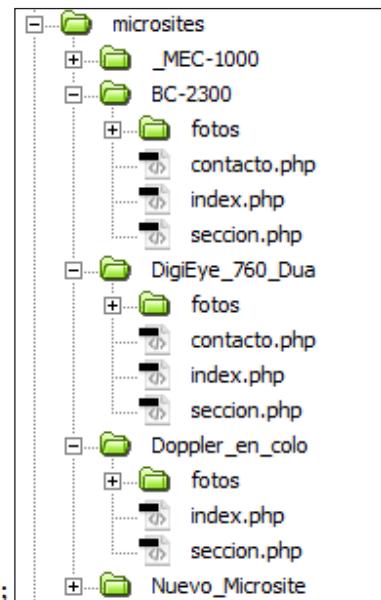
        /*codigo personalizado*/
        $ampliacion = '<?php $idWeb='.$this->id.'; ?>';
        $archivo = fopen("../microsites/".$this->alias."/index.php", 'w');
        fwrite($archivo, $ampliacion.$textoIndex);
        fclose($archivo);

        /*crea archivo seccion*/
        $seccion= fopen("./temas/plantilla_seccion.php", 'r');
        $textoSeccion = stream_get_contents($seccion);
        fclose($seccion);

        /*codigo personalizado*/
        $ampliacion = '<?php $idWeb='.$this->id.'; ?>';
        $archivo = fopen("../microsites/".$this->alias."/seccion.php", 'w');
        fwrite($archivo, $ampliacion.$textoSeccion);
        fclose($archivo);

        /*crea archivo contacto*/
        $contacto= fopen("./temas/plantilla_contacto.php", 'r');
        $textoContacto = stream_get_contents($contacto);
        fclose($contacto);

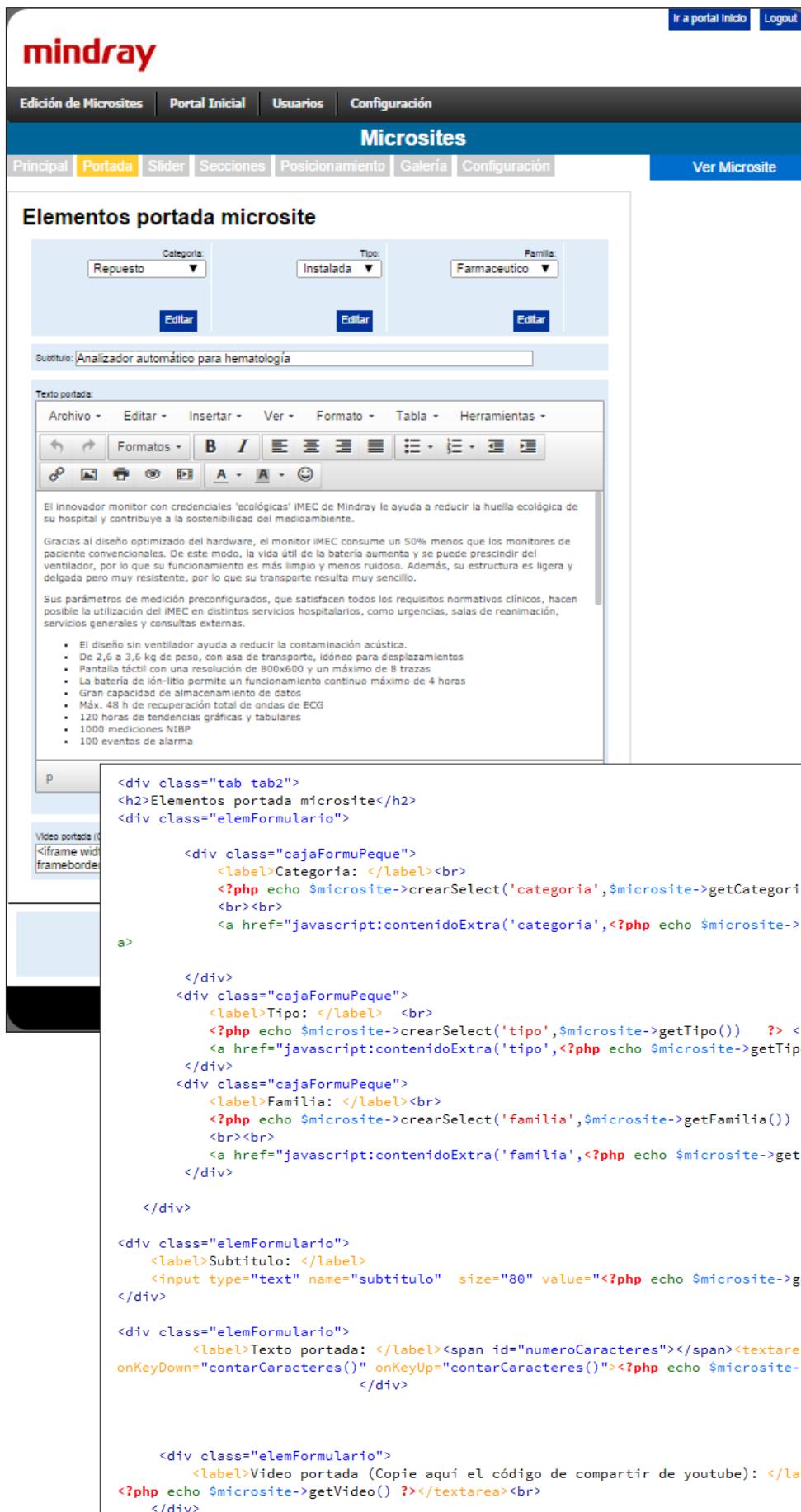
        /*codigo personalizado*/
        $ampliacion = '<?php $idWeb='.$this->id.'; ?>';
        $archivo = fopen("../microsites/".$this->alias."/contacto.php", 'w');
        fwrite($archivo, $ampliacion.$textoContacto);
        fclose($archivo);
    }
}
```



La clase Microsite dispone de un método a tal efecto, que vemos como crea la estructura de carpetas, tomando como nombre el alias (o en su defecto el título modificado). También se guardan los archivos insertando el código de la plantilla creada para cada sección y lo añade a los archivos.

Una vez creada la estructura de archivos y carpetas el sistema nos muestra el resto de opciones de edición de las que disponemos

En esta parte encontramos el corazón de nuestro gestor, donde por medio de pestañas podemos editar y introducir todo el contenido y la personalización de nuestra microsite.



The screenshot shows the Mindray CMS interface with the following details:

- Top Navigation:** Ir a portal inicio, Logout.
- Main Navigation:** Edición de Microsites, Portal Inicial, Usuarios, Configuración, Microsites, Principal, Portada, Slider, Secciones, Posicionamiento, Galería, Configuración, Ver Microsite.
- Section Header:** Elementos portada microsite.
- Form Fields:** Categoría: Repuesto, Tipo: Instalada, Familia: Farmaceutico, each with an 'Editar' button.
- Text Area:** Subtítulo: Analizador automático para hematología.
- Text Editor:** Texto portada with a toolbar.
- Text Content:** El innovador monitor con credenciales 'ecológicas' IMEC de Mindray le ayuda a reducir la huella ecológica de su hospital y contribuye a la sostenibilidad del medioambiente. Gracias al diseño optimizado del hardware, el monitor IMEC consume un 50% menos que los monitores de paciente convencionales. De este modo, la vida útil de la batería aumenta y se puede prescindir del ventilador, por lo que su funcionamiento es más limpio y menos ruidoso. Además, su estructura es ligera y delgada pero muy resistente, por lo que su transporte resulta muy sencillo. Sus parámetros de medición preconfigurados, que satisfacen todos los requisitos normativos clínicos, hacen posible la utilización del IMEC en distintos servicios hospitalarios, como urgencias, salas de reanimación, servicios generales y consultas externas.
  - El diseño sin ventilador ayuda a reducir la contaminación acústica.
  - De 2,6 a 3,6 kg de peso, con asa de transporte, idóneo para desplazamientos
  - Pantalla táctil con una resolución de 800x600 y un máximo de 8 trazas
  - La batería de ión-litio permite un funcionamiento continuo máximo de 4 horas
  - Gran capacidad de almacenamiento de datos
  - Max. 48 h de recuperación total de ondas de ECG
  - 120 horas de tendencias gráficas y tabulares
  - 1000 mediciones NIBP
  - 100 eventos de alarma
- Code Preview:** Shows the generated HTML and PHP code for the page elements.

A continuación mostramos parte del código, podemos ver como mediante formularios vamos solicitando la información y introduciendo los datos guardados en caso de que los ubiese.

En la siguiente página se mostrarán algunas de las diferentes opciones de las que disponemos, no se muestra el código por no alargar el documento en exceso.

**mindray**

Edición de Microsites | Portal Inicial | Usuarios | Configuración

**Microsites**

Principal | Portada | Slider | Secciones | Posicionamiento | Galería | Configuración

### Principal



Logo:

- Seleccionar archivo Ningún archivo seleccionado
- Usar Logotipo insertado para la microsite ▾

Título: BC-6800

El Alias es: Nuevo\_Microsite

URL actual: www.ingenierosdelsoftware.com/microsites/Nuevo\_Microsite

Nuevo\_Microsite

Dentro de las opciones de edición de las microsites, tambien dispondremos de la posibilidad de editar partes de la misma como los sliders o las secciones.

**mindray**

Edición de Microsites | Portal Inicial | Usuarios | Configuración

**Microsites**

Principal | Portada | Slider | Secciones | Posicionamiento | Galería | Configuración

### Elementos portada microsite

Categoría	Tipo	Familia
Reuesto	Instalada	Farmaceutico

Editar Editar Editar

Subtítulo: Analizador automático para hematología

Texto portada:

Archivo | Editar | Insertar | Ver | Formato | Tabla | Herramientas |

Formatos | B | I |

El innovador monitor con credenciales 'ecológicas' IMEC de Mindray le ayuda a reducir la huella ecológica de su hospital y contribuye a la sostenibilidad del medioambiente.

Gracias al diseño optimizado del hardware, el monitor IMEC consume un 50% menos que los monitores de paciente convencionales. De este modo, la vida útil de la batería aumenta y se puede prescindir del ventilador, por lo que su funcionamiento es más limpio y menos ruidoso. Además, su estructura es ligera y delgada pero muy resistente, por lo que su transporte resulta muy sencillo.

Sus parámetros de medición preconfigurados, que satisfacen todos los requisitos normativos clínicos, hacen posible la utilización del IMEC en distintos servicios hospitalarios, como urgencias, salas de reanimación, servicios generales y consultas externas.

- El diseño sin ventilador ayuda a reducir la contaminación acústica.
- De 2,6 a 3,6 kg de peso, con asa de transporte, idóneo para desplazamientos
- Pantalla táctil con una resolución de 800x600 y un máximo de 8 trazas
- La batería de ión-litio permite un funcionamiento continuo máximo de 4 horas
- Gran capacidad de almacenamiento de datos
- Máx. 48 h de recuperación total de ondas de ECG
- 120 horas de tendencias gráficas y tabulares
- 1000 mediciones NIBP
- 100 eventos de alarma

P Palabras: 365

Vídeo portada (Copie aquí el código de compartir de youtube):  
<iframe width="560" height="315" src="https://www.youtube.com/embed/ZJZ28-XtkI" frameborder="0" allowfullscreen></iframe>

**mindray**

Edición de Microsites | Portal Inicial | Usuarios | Configuración

**Microsites**

Principal | Portada | Slider | Secciones | Posicionamiento | Galería | Configuración

### Edición Slider

Añadir una Diapositiva



Editor Borrar ▾ ▾

Editor Borrar ▾ ▾

Editor Borrar ▾ ▾

Protocols and professional services delivered to enhance performance

**mindray**

Edición de Microsites | Portal Inicial | Usuarios | Configuración

**Microsites**

Principal | Portada | Slider | Secciones | Posicionamiento | Galería | Configuración

### Edición secciones

Añadir una Sección

Características de la maquina

Editor Borrar ▾ ▾

Garantía

Editor Borrar ▾ ▾

The screenshot shows a list of images in the 'Galería de fotos' section. Each item includes the image thumbnail, the full path and file name (e.g., '/gestorContenido/imagenes/Articulos\_38\_id1409263177.gif'), the original file name (e.g., 'Articulos\_38\_id1409263177.gif'), and the file size (e.g., '43.56 Kb').

Imagen	Nombre	Tamaño
	/gestorContenido/imagenes/Articulos_38_id1409263177.gif	43.56 Kb
	/gestorContenido/imagenes/Articulos_34_id1409885657.gif	43.74 Kb
	/gestorContenido/imagenes/Articulos_44_id1409264602.gif	30.16 Kb
	/gestorContenido/imagenes/Articulos_9_id1409264600.jpg	34.97 Kb
	/gestorContenido/imagenes/Articulos_28_id1409267130.gif	21.39 Kb
	/gestorContenido/imagenes/Articulos_28_id1409267130.jpg	8.03 Kb
	/gestorContenido/imagenes/Articulos_sub_banner_id2140926800.jpg	80 Kb

Cuando introducimos los datos, al guardar el objeto microsite se rellena con toda la información de los diferentes formularios y actualiza la base de datos con los nuevos datos.

A continuación mostramos el método para actualizar estos datos que vemos como recorre el post validando los datos introducidos y contruye la sentencia SQL para introducirlos.

The screenshot shows the 'Posicionamiento' section with fields for 'Tag descripción:' containing 'bonita pagina' and 'Tag palabras clave (Separa las palabras por comas):' containing 'que bien, todo fenomenal, esto es nuevo'.

```
public function actualizaMicrosite($datos,$archivos){
    $this->setId($datos['id']);
    $this->alias = $datos['alias'];

    $this->setTitulo($datos['titulo']);
    $this->setLogoPincipal($datos['logoPrincipal']);
    $this->setTexto($datos['texto']);
    $this->setCategoria($datos['categoria']);
    $this->setFamilia($datos['familia']);
    $this->setTipo($datos['tipo']);
    $this->setTagDescripcion($datos['tagDescripcion']);
    $this->setTagClaves($datos['tagClaves']);
    $this->setVideo($datos['video']);
    $this->setSubtitulo($datos['subtitulo']);

    $conexion = new Bd();
    $sql = "UPDATE ". $this->tablaDB . " SET ";
    $sql .= " titulo='".$this->titulo' ";
    $sql .= " , logoprincipal='".$this->logoPrincipal' ";
    $sql .= " , texto='".$this->texto' ";
    $sql .= " , categoria='".$this->categoria' ";
    $sql .= " , tipo='".$this->tipo' ";
    $sql .= " , familia='".$this->familia' ";
    $sql .= " , tagClaves='".$this->tagClaves' ";
    $sql .= " , tagDescripcion='".$this->tagDescripcion' ";
    $sql .= " , video='".$this->video' ";
    $sql .= " , subtítulo='".$this->subtitulo' ";

    if (strlen($archivos['ruta']['name']) >0) {
        $directorio = "../microsites/". $this->getAlias() . "/fotos/";
        borrarFoto($this->id, $this->tablaDB, 'ruta');
        $logotipo = subirFoto($archivos['ruta'], $directorio);
        $pos = strpos($logotipo, "/")+1;
        $this->logotipo = substr($logotipo, $pos);

        $sql .= " , logotipo = '$this->logotipo' ";
    }
    $sql .= " WHERE id=". $this->id;
    $res = $conexion->actualizar($sql);
    /*Personalizacion*/
    $personaliza = new Personalizacion($this->id);
    $personaliza->setCabeza($datos['cabeza']);
    $personaliza->setMailPersonalizado($datos['mailPersonalizado']);
    $personaliza->guardaPersonalizacion();
}
```

The screenshot shows the 'Configuración' section with a dropdown 'Mostrar tira superior' set to 'Si' and a field 'Mail Personalizado: (si desea usar el mail principal, dejar este campo en blanco)' containing 'correo@ejemplo.com'.

### 4.3.3 La clase Microsite

Ya hemos comentado que mostrar todo el código de nuestro proyecto en la memoria sería demasiado tedioso, si se desea examinar la codificación con mas detalle, se sugiere que se examines los archivos del proyecto implementado.

Pero dado que la clase microsite podriamos definirla como la mas importante de nuestro portal vamos a describir su estructura y alguno de los métodos utilizados,. Esto nos servira para comprender la metodología usada en la implantación pues el resto de clases tendrían una estructura similar aunque con las diferencias obvias portratarse de objetos diferentes.

```
private $id;
private $titulo;
private $alias;
private $logotipo;
private $tablaDB;
private $logoPrincipal;
private $texto;
private $orden;
private $categoria;
private $tipo;
private $familia;
private $tagDescripcion;
private $tagClaves;
private $video;
private $subtitulo;
```

Como en todas las clases lo primero que hacemos es definir los atributos de nuestra clase, tambien se han generado métodos de entrada y salida de los mismo validando los datos introducidos en cada caso.

```
public function setTexto($texto) {
    $this->texto = addslashes($texto);
}

public function setAlias($alias) {

    if(strlen($alias)<3){
        $alias = $this->titulo;
    }

    $str = limpiar_caracteres_especiales($alias);
    if(strlen($str)>15 ){
        $str= substr($str, 0, 15);
    }

    if($this->existeElemento('alias', $str)){
        $numeroAleatorio = rand(1,100);
        $str = $str.$numeroAleatorio;
    }

    $this->alias = $str;
}
```

En el caso del atributo alias, dado que lo usaremos como nombre para la carpeta del microsite, vemos como lo primero que hace es comprobar si lo han introducido y si no usa parte del título para crearlo, tambien vemos como comprueba si ya existe además de eliminar caracteres que nos puedan dar problemas con la función limpiar caracteres.

En nuestra clase existen varios métodos que se encargan de llenar el objeto con los datos ya sea si proceden desde el formulario como si lo estamos recuperando desde la base de datos, en los siguientes métodos vemos como partiendo de una consulta SQL el objeto es llenado con los datos.

Al proceder de la base de datos no es necesario la validación de los mismos, así que los asignamos directamente.

```
public function nuevoDesdeSql($id, $titulo, $alias,$logotipo,$logoPrincipal,$orden,$texto,$categoria,$tipo,
$familia,$tagDescripcion,$tagClaves,$video, $subtitulo){

    $this->id = $id;
    $this->titulo = $titulo;
    $this->alias = $alias;
    $this->logotipo = $logotipo;
    $this->logoPincipal = $logoPrincipal;
    $this->orden = $orden;
    $this->texto = $texto;
    $this->categoria = $categoria;
    $this->tipo = $tipo;
    $this->familia = $familia;
    $this->tagDescripcion = $tagDescripcion;
    $this->tagClaves = $tagClaves;
    $this->video = $video;
    $this->subtitulo = $subtitulo;
```

Una acción muy común en nuestra aplicación es que tenga que recuperar los datos de un objeto partiendo del identificador del mismo, para ello disponemos de un método que recupera esta información.

```
public function obtenerPorId($id){

    $sql = "SELECT * FROM ".$this->tablaDB." WHERE id=".$id;
    $conexion = new Bd();
    $res = $conexion->consulta($sql);
    $this->nuevoDesdeSql($res['id'],$res['titulo'],$res['alias'],$res['logotipo'],$res['logoPrincipal'],$res['orden'],
    $res['texto'],$res['categoria'],$res['tipo'],$res['familia'],$res['tagDescripcion'],$res['tagClaves'],$res['video'], $res[
    'subtitulo'] );
```

}

El construir el objeto partiendo de la base de datos también es un procedimiento habitual usado por la clase padre ListaMicrosite que se encarga de manejar un conjunto de objetos del tipo microsite.

```
public function obtenerElementosPorTipo($parametro){

    $conexion = new Bd();
    $sql = "SELECT * FROM ".$this->tabla." WHERE ".$condicion." ORDER BY orden ASC";
    $res = $conexion->consultarLista($sql);
    while ($list($id, $titulo, $alias,$logotipo,$logoPrincipal,$orden,$texto,$categoria,$tipo,$familia,$tagDescripcion,
    $tagClaves,$video,$subtitulo) = mysqli_fetch_array($res) ) {
        $fila = new Microsite();
        $fila->nuevoDesdeSql($id, $titulo, $alias,$logotipo,$logoPrincipal,$orden,$texto,$categoria,$tipo,$familia,
        $tagDescripcion,$tagClaves,$video,$subtitulo);
        //añadir al array de la clase clase
        array_push($this->lista, $fila);
    }
}
```

Otro tipo de métodos que encontraremos en la clase, son los de impresión que se encargan en pintar los datos para su visualización.

```
public function stringElemento(){

    $txt = "<div class='listaMicrosite'><a href='microsites.php?id=".$this->id."' class='editar'>".$this->titulo."</a>
<a href='javascript:borrarSitio(\".$this->id.\")' class='borrar'>borrar</a></div>";

    return $txt;
}
```

## 4.3.4 Conexión a la Base de datos

Para crear las conexiones a la base de datos se ha generado una clase a tal efecto, desde la cual podemos enviar las consultas SQL fácilmente usando los métodos creados al efecto.

```
public function __construct(){

    $this->conexion = new mysqli($this->server, $this->usuario, $this->pass , $this->basedatos);
    $this->conexion->select_db($this->basedatos);
    $this->conexion->query("SET NAMES 'utf8'");

}
```

## 4.3.5 Uso de Ajax

El portal hace uso de esta tecnología frecuentemente para evitar refrescos de página, ordenar resultados, borrado de elementos, actualizar los select, etc.

A continuación mostramos un ejemplo de la codificación de javascript para borrar uno de los microsites.



## 4.3.6 Funciones

El sistema tambien hace de forma recurrente llamadas a funciones, en muchos casos propias del lenguaje PHP, pero en otros casos llama a funciones propias.

A continuación mostramos algunas de las funciones que se han creado para el sistema.

```
function subirFoto($archivo,$directorio){
    $ruta = "";
    $tipo = $archivo['type'];
    $stamano = $archivo['size'];

    $nombreArchivo = limpiar_caracteres_especiales($archivo['name']);
    $nombreArchivo = cortarCadenaFinal($nombreArchivo, ".");

    if((strpos($tipo, 'gif') || strpos($tipo, 'jpeg') || strpos($tipo, 'png')) && $stamano < 1000000 ){
        $idUnico = time();

        if(strpos($tipo, 'gif')){
            $extension = '.gif';
        }else if(strpos($tipo, 'jpeg')){
            $extension = '.jpg';
        }else{
            $extension = '.png';
        }

        if(move_uploaded_file($archivo['tmp_name'],$directorio.$nombreArchivo.$idUnico.$extension)){
            $ruta = $directorio.$nombreArchivo.$idUnico.$extension;
        }else{
            echo "<script>alert('Error al subir el archivo al servidor')</script>";
        }
    }else{
        echo "<script>alert('La foto debe tener una extensión del tipo: jpg, gif o png')</script>";
    }

    return $ruta;
}
```

La función subirFoto() sube archivos al servidor dandole como parametros el archivo y el directorio, dado que es una acción recurrente en nuestra aplicación para la subida de logos y otros elementos si entidad suficiente para realizar una clase, se ha decidido añadir esta función que es usada por los distintos métodos de las clases..

```
function limpiar_caracteres_especiales($archivo) {
    $archivo = ereg_replace("[ààààà@]", "a", $archivo);
    $archivo = ereg_replace("[ÀÀÀÀÀ]", "A", $archivo);
    $archivo = ereg_replace("[éèèè]", "e", $archivo);
    $archivo = ereg_replace("[ÈÈÈÈ]", "E", $archivo);
    $archivo = ereg_replace("[íííí]", "i", $archivo);
    $archivo = ereg_replace("[ÌÌÌÌ]", "I", $archivo);
    $archivo = ereg_replace("[óòòòö]", "o", $archivo);
    $archivo = ereg_replace("[ÒÒÒÒÔ]", "O", $archivo);
    $archivo = ereg_replace("[úùùùü]", "u", $archivo);
    $archivo = ereg_replace("[ÙÙÙÙÜ]", "U", $archivo);
    $archivo = str_replace("[¿?\\]", "_", $archivo);
    $archivo = str_replace(" ", "_", $archivo);
    $archivo = str_replace("ñ", "n", $archivo);
    $archivo = str_replace("Ñ", "N", $archivo);
    ...
}
```

La función para limpiar caracteres es usada para cuando queremos guardar un archivo con el nombre proporcionado por el usuario, lo cual aunque es positivo para el posicionamiento, puede causar problemas en el guardado, así que eliminaremos cualquier carácter que pueda causar errores.

# 5. Plan de proyecto

## 5.1 Equipo de trabajo

Jefe de proyecto

Ingeniero en informática

5 años de experiencia en dirección de proyectos software

Conocimiento de metodologías de gestión de proyecto, análisis de riesgos, seguimiento, etc.

Analista Funcional

Ingeniero en informática

Más de 5 años de experiencia en análisis de aplicaciones software para el medio Internet.

Conocimiento de técnicas de análisis, modelado, metodología y usabilidad.

Tester

Ingeniero Técnico en informática

16 años de experiencia en ingeniería del software y desarrollo de aplicaciones empresariales con Servidores Apache y PHP.

Programadores

Ingeniero en informática preferiblemente

Más de 4 años de experiencia en desarrollo de aplicaciones PHP con bases de datos

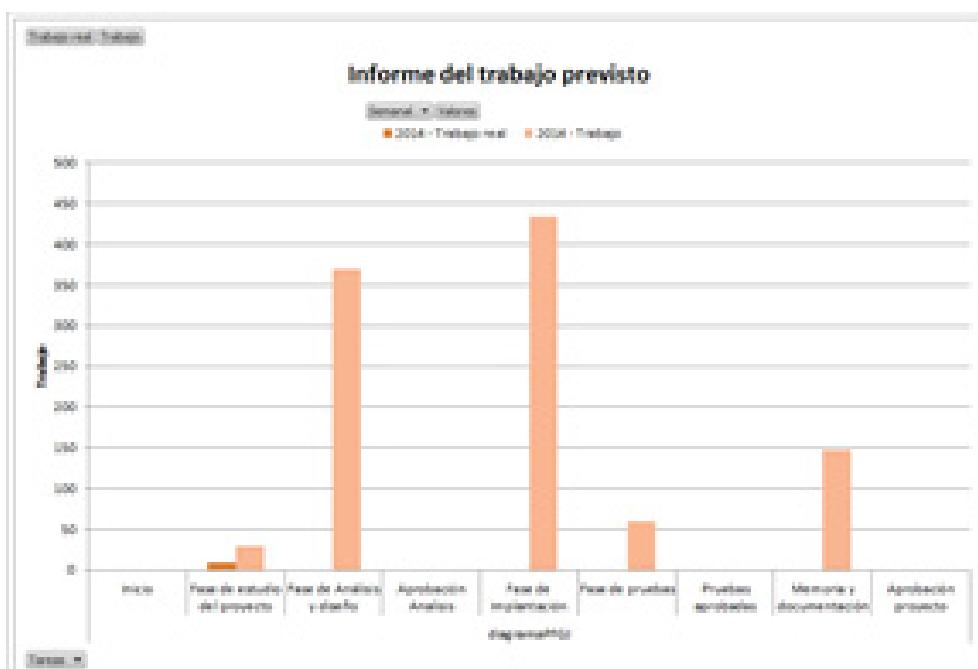
Diseñador

Titulado en Artes Gráficas

Más de 4 años de experiencia en desarrollo Web

Experto en HTML5 y CSS3

## 5.2 Plan de trabajo



Inicio	0 días	dom 15/06/14	dom 15/06/14		
<b>Fase de estudio del proyecto</b>	<b>2,13 días</b>	<b>dom 15/06/14</b>	<b>jue 19/06/14</b>	<b>1</b>	<b>Analista</b>
Solicitud del proyecto	2 horas	dom 15/06/14	lun 16/06/14	1	Analista
Busqueda de información	8 horas	dom 15/06/14	mar 17/06/14	3	Analista
Examen de herramientas similares	8 horas	lun 16/06/14	mié 18/06/14	4	Analista
Propuesta del proyecto	5 horas	mié 18/06/14	jue 19/06/14	5	Analista
<b>Fase de Análisis y diseño</b>	<b>13,5 días</b>	<b>jue 19/06/14</b>	<b>vie 18/07/14</b>	<b>6</b>	<b>Analista</b>
<b>Estudio de la tecnología</b>	<b>5,38 días</b>	<b>jue 19/06/14</b>	<b>mar 01/07/14</b>		<b>Analista</b>
Busqueda de soluciones Similares	1 hora	jue 19/06/14	jue 19/06/14	6	Analista
Busqueda de bibliografía y documentación	10 horas	jue 19/06/14	lun 23/06/14	9	Analista
Busqueda de APIS y herramientas a utilizar	5 horas	lun 23/06/14	mar 24/06/14	10	Analista
Estudio de la bibliografía y documetación	20 horas	mar 24/06/14	lun 30/06/14	11	Analista
Análisis de requisitos	2 horas	lun 30/06/14	lun 30/06/14	12	Analista
Generación de informe del estudio	5 horas	mar 01/07/14	mar 01/07/14	13	Analista
<b>Fase de Diseño</b>	<b>8,13 días</b>	<b>mié 02/07/14</b>	<b>vie 18/07/14</b>	<b>8</b>	<b>Analista</b>
Definir funcionalidades	5 horas	mié 02/07/14	mié 02/07/14	14	Analista
Definición de complementos	5 horas	jue 03/07/14	jue 03/07/14	16	Analista
<b>Diseño UML</b>	<b>5,63 días</b>	<b>vie 04/07/14</b>	<b>mié 16/07/14</b>	<b>17</b>	<b>Analista</b>
Modelado de datos	20 horas	vie 04/07/14	mié 09/07/14	17	Analista
Diseño de clases y casos de uso	20 horas	jue 10/07/14	mar 15/07/14	19	Analista
Estados	5 horas	mié 16/07/14	mié 16/07/14	20	Analista
Diseño base	8 horas	jue 17/07/14	vie 18/07/14	21	Analista
Documentacion	2 horas	vie 18/07/14	vie 18/07/14	22	Analista
Aprobación Analisis	0 días	vie 10/01/14	vie 10/01/14		Analista
<b>Fase de Implantación</b>	<b>17,5 días</b>	<b>lun 21/07/14</b>	<b>mié 27/08/14</b>	<b>7</b>	<b>Programador</b>
<b>Aplicación WEB</b>	<b>17,5 días</b>	<b>lun 21/07/14</b>	<b>mié 27/08/14</b>		<b>Programador</b>
Configuración del sistema	5 horas	lun 21/07/14	lun 21/07/14	23	Programador
<b>Front End</b>	<b>3,38 días</b>	<b>mar 22/07/14</b>	<b>mar 29/07/14</b>		<b>Programador</b>
Creación de plantillas	25 horas	mar 22/07/14	lun 28/07/14	27	Diseñador
Estilos	2 horas	mar 29/07/14	mar 29/07/14	29	Diseñador
<b>Back End</b>	<b>11,88 días</b>	<b>mar 29/07/14</b>	<b>lun 25/08/14</b>	<b>28</b>	<b>Programador</b>
Gestor de contenido	70 horas	mar 29/07/14	lun 18/08/14	30	Programador
Bases de datos	10 horas	lun 18/08/14	mié 20/08/14	32	Programador
Gestión de los diferentes dominios	10 horas	mié 20/08/14	vie 22/08/14	33	Programador
Gestor de usuarios y comentarios	5 horas	vie 22/08/14	lun 25/08/14	34	Programador
<b>Adaptación Móvil</b>	<b>1,5 días</b>	<b>lun 25/08/14</b>	<b>mié 27/08/14</b>	<b>31</b>	<b>Programador</b>
Creación de templates Comunes	10 horas	lun 25/08/14	mié 27/08/14	35	Diseñador
Libros de estilo para móvil	2 horas	mié 27/08/14	mié 27/08/14	37	Diseñador
Implantación en servidor	1 hora	mié 27/08/14	mié 27/08/14	38	Programador
<b>Fase de pruebas</b>	<b>3,75 días</b>	<b>jue 28/08/14</b>	<b>jue 04/09/14</b>	<b>25</b>	<b>Tester</b>
Pruebas del sistema	30 horas	jue 28/08/14	jue 04/09/14	39	Tester
Pruebas aprobadas	0 días	jue 04/09/14	jue 04/09/14	41	Tester
<b>Memoria y documentación</b>	<b>12,75 días</b>	<b>mar 15/07/14</b>	<b>mar 12/08/14</b>		<b>Analista</b>
Recopilación de informes anteriores	12 horas	vie 05/09/14	mar 09/09/14	41	Analista
Maquetación de la memoria	25 horas	mar 09/09/14	mar 16/09/14	44	Diseñador
Presentación Virtual	8 horas	mar 16/09/14	mié 17/09/14	45	Diseñador
Aprobación proyecto	0 días	mié 17/09/14	mié 17/09/14	46	

## 5.3 Análisis de costes

Para realizar el estudio de costes se tienen en cuenta los siguientes tipos de costes:

- Coste personal
- Coste Hardware
- Coste Software
- Costes generales

### PERSONAL

Analista: 27,75 días x 8 horas x 50 € = 11.100 €

Programador y diseñador: 20,75 días x 8 horas x 30 € = 4.980 €

Tester: 3,75 días x 8 horas x 35 euros = 1.050 €

TOTAL: 17.130 €

## COSTE HARDWARE

Equipo PC 1.500 €  
Hosting, Servidor dedicado 120 €/mes.  
3 meses para la creación 360 €.

## COSTE SOFTWARE

En el desarrollo del proyecto se utilizarán herramientas freeware, por tanto no tienen coste alguno.

Costes generales	Coste unitario	Unidades / Meses	Total
Alquiler oficina	300	3	900
Material oficina	50	3	150
Impresión / encuadernación del proyecto	250	3	750
Electricidad	45	3	135
Internet	20	3	60
		Total	1.995

Costes totales	Coste
coste personal	17.130,00
coste hardware	1.850,00
costes generales	1.995,00
Total	20.975,00



# 6. Trabajos futuros

El portal tiene muchas posibilidades de ampliación, y su evolución tomara varias caminos, desde mejoras en la interfaz insertando nuevos módulos como un sistema de banners, etc. Tambien un sistema de usuarios administradores donde se permita la edición de algunos de los sitios y no de su totalidad, de esta forma se podría clasificar el trabajo en varios grupos sin interferencias, y muchas mas opciones que podriamos añadirle a la funcionalidad del sistema.

A continuación se muestra una lista de las mejoras y ampliaciones que se consideran más interesantes.

Más opciones de personalización de los sitios, permitiendo así poder aplicar más creatividad por parte del usuario.  
Poner el sitio en diferentes idiomas.

Integración de RSS para promocionar la información publicada.

Mejorar la accesibilidad del portal web para aquellos usuarios con algún tipo de discapacidad.

Diferentes roles para los usuarios Administradores.

Posibilidad de guardar contenidos para facilitar la publicación de contenidos.

Sistema de valoración y puntuación de los sitios por parte de los usuarios.

Mayor integración con las redes sociales.

Mejoras en las interfaces móviles y en el posicionamiento.

Dado la popularidad de internet hoy en día y las ventajas que nos ofrece en el marketing y la promoción las posibilidades son muchas, y se espera un constante crecimiento de la aplicación.



# Anexos

## Anexo.1 Recursos utilizados

- Editor de texto enriquecido Tinymce - <http://www.tinymce.com>
- Librería JQUERY - <http://jquery.com/>
- Bxslider - <http://bxslider.com/>
- Phpmailer: - Librería php para envio de mails

## Anexo.2 Bibliografía

Para la realización del proyecto se han consultado las siguientes fuentes.

- Php.net
- [eddiewebdotorg.wordpress.com/](http://eddiewebdotorg.wordpress.com/)
- [bextlan.com/proyectos/alumnos/laboratoriosdelweb/temas/laravel/laravel-4-estructura-de-un-proyecto/](http://bextlan.com/proyectos/alumnos/laboratoriosdelweb/temas/laravel/laravel-4-estructura-de-un-proyecto/)
- Cursos y documentación de Jesús Conde
- Apuntes y documentación de las asignaturas cursadas en la UEM
- <http://librosweb.es/>
- <http://www.aulaclic.es>