



**UNIVERSIDAD DE MÁLAGA**

**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA**

**INGENIERO TÉCNICO EN INFORMÁTICA DE SISTEMAS**

**GESTIÓN WEB 2.0 GASTRONÓMICA**

**Realizado por:**

**JAVIER CERVILLA CONSTANT**

**Dirigido por:**

**JOSÉ GALINDO GÓMEZ**

**Departamento:**

**LENGUAJES Y CIENCIAS DE LA COMPUTACIÓN**



**UNIVERSIDAD DE MÁLAGA**  
**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA**  
**INGENIERO TÉCNICO EN INFORMÁTICA DE SISTEMAS**

Reunido el tribunal examinador en el día de la fecha, constituido por:

Presidente/a D./D<sup>a</sup>. \_\_\_\_\_

Secretario/a D./D<sup>a</sup>. \_\_\_\_\_

Vocal D./D<sup>a</sup>. \_\_\_\_\_

para juzgar el proyecto Fin de Carrera titulado: \_\_\_\_\_

realizado por D./D<sup>a</sup> \_\_\_\_\_

tutorizado por D./D<sup>a</sup> \_\_\_\_\_

, y en su caso, dirigido académicamente por D./D<sup>a</sup> \_\_\_\_\_  
\_\_\_\_\_

ACORDÓ POR \_\_\_\_\_ OTORGAR LA CALIFICACIÓN  
DE \_\_\_\_\_

Y PARA QUE CONSTE, SE EXTIENDE FIRMADA POR LOS COMPARCIENTES  
DEL TRIBUNAL, LA PRESENTE DILIGENCIA

Málaga a \_\_ de \_\_\_\_\_ del 201\_\_

El/La Presidente/a

El/La Secretario

El/La Vocal

Fdo:

Fdo:

Fdo:



## AGRADECIMIENTOS

En este apartado quisiera dedicar unas palabras a todas las personas que me han ayudado a lo largo de la carrera y a algunas personas que me hubiera gustado que me vieran finalizarla. Comienzo:

A mis padres por animarme a estudiar una carrera, por su paciencia, por su esfuerzo, compresión, sacrificio y por animarme en los momentos difíciles. A mi madre en particular por darme esos “tironcillos de orejas” cuando me hacían falta, sin ellos seguramente no habría acabado la carrera y a mi padre por ayudarme con el proyecto y con la exposición. Muchísimas gracias a los dos.

A mi hermana Nuria y a mi novia Jessica por apoyarme, comprenderme y animarme en los momentos difíciles. Muchas gracias.

A mis amigos de la resi por ayudarme y aportarme ideas: Jaime, Paco, Meli, Marina, Adri...Muchas gracias.

A mi tutor José Galindo por el esfuerzo, facilidades, dedicación y apoyo que me ha dado durante la realización del proyecto. Muchas gracias por sus consejos y por contestar tan rápido a mis correos.

Para finalizar me gustaría dedicarle este proyecto a dos personas a las que le hubiera encantado verme acabar la carrera: a mi abuela Lola recientemente fallecida y a mi abuelo Emilio que tan sólo pudo verme empezar la carrera, para ellos dos mi especial dedicatoria. Estén donde estén estarán siempre orgullosos de mí. Nunca me olvidaré de vosotros.

# ÍNDICE DE CONTENIDOS

<b>CAPÍTULO 1. INTRODUCCIÓN.....</b>	<b>1</b>
<b>1.1. MOTIVACIÓN .....</b>	<b>1</b>
<b>1.2. OBJETIVOS.....</b>	<b>2</b>
<b>1.3. ESTRUCTURA DE LA MEMORIA .....</b>	<b>3</b>
<b>CAPÍTULO 2. ANÁLISIS Y REQUISITOS .....</b>	<b>5</b>
<b>2.1. ANÁLISIS DE LA WEB .....</b>	<b>5</b>
<b>2.2. REQUISITOS FUNCIONALES .....</b>	<b>11</b>
<b>2.2.1. OFERTA DE SERVICIO WEB. PUBLICACIÓN DE RECETAS .....</b>	<b>12</b>
<b>2.2.2. BÚSQUEDA DE RECETAS .....</b>	<b>12</b>
<b>2.2.3. GESTIÓN DE USUARIOS .....</b>	<b>13</b>
<b>2.3. REQUISITOS NO FUNCIONALES .....</b>	<b>13</b>
<b>2.3.1. REQUISITOS DE PRESENTACIÓN .....</b>	<b>13</b>
<b>2.3.2. REQUISITOS TECNOLÓGICOS .....</b>	<b>13</b>
<b>CAPÍTULO 3. HERRAMIENTAS.....</b>	<b>15</b>
<b>3.1. LENGUAJES .....</b>	<b>15</b>
<b>3.1.1. DISEÑO DE LA WEB: HTML Y CSS .....</b>	<b>16</b>
<b>3.1.2. SERVIDOR: PHP .....</b>	<b>16</b>
<b>3.1.3. CLIENTE: JAVASCRIPT .....</b>	<b>17</b>
<b>3.2. APLICACIONES.....</b>	<b>17</b>
<b>3.2.1. EDITORES: DREAMWEAVER CS5 .....</b>	<b>17</b>
<b>3.2.2. DISEÑO DIAGRAMAS ENTIDAD-RELACIÓN: DIA .....</b>	<b>17</b>
<b>3.2.3. DISEÑO DE LA BASE DE DATOS: MYSQL WORKBENCH .....</b>	<b>18</b>
<b>3.2.4. SERVIDOR: WAMPSERVER 2.2 X32 .....</b>	<b>18</b>
<b>3.3. INSTALACIÓN .....</b>	<b>19</b>
<b>3.3.1 WAMPSERVER 2.2 X32.....</b>	<b>19</b>
<b>3.3.2. DREAMWEAVER CS5 .....</b>	<b>20</b>
<b>3.3.3. MYSQL WORKBENCH.....</b>	<b>21</b>
<b>3.3.4. DIA 0.97.1 .....</b>	<b>22</b>

<b>CAPÍTULO 4. DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS .....</b>	<b>23</b>
<b>4.1. DISEÑO DE LA BD. MODELO CONCEPTUAL .....</b>	<b>25</b>
<b>4.1.1. DEFINICIÓN DE OBJETIVOS .....</b>	<b>25</b>
<b>4.1.2. IDENTIFICACIÓN DE ENTIDADES Y ATRIBUTOS .....</b>	<b>27</b>
<b>4.1.3. DISEÑO FINAL EN MYSQL WORKBENCH .....</b>	<b>28</b>
<b>CAPÍTULO 5. DISEÑO HTML: ELEMENTOS UTILIZADOS .....</b>	<b>31</b>
<b>5.1. ELEMENTOS HTML .....</b>	<b>33</b>
<b>5.1.1. ENLACES .....</b>	<b>34</b>
<b>5.1.2. IMÁGENES .....</b>	<b>34</b>
<b>5.1.3. LISTAS .....</b>	<b>35</b>
<b>5.1.4. TABLAS.....</b>	<b>36</b>
<b>5.1.5. FORMULARIOS .....</b>	<b>37</b>
<b>5.1.6. MARCOS DE PÁGINAS .....</b>	<b>40</b>
<b>CAPÍTULO 6. DISEÑO E IMPLEMENTACIÓN EN PHP.....</b>	<b>43</b>
<b>6.1. BREVE DESCRIPCIÓN DE LOS APARTADOS DE LA WEB.....</b>	<b>44</b>
<b>6.2. VISIÓN DE UN USUARIO NO REGISTRADO.....</b>	<b>46</b>
<b>6.3. VISIÓN Y PRIVILEGIOS DE UN USUARIO REGISTRADO .....</b>	<b>47</b>
<b>6.4. VISIÓN Y PRIVILEGIOS DE USUARIO ADMINISTRADOR.....</b>	<b>47</b>
<b>6.5. FUNCIONES INCLUIDAS EN TODAS LAS DEMÁS .....</b>	<b>47</b>
<b>6.6. MENÚ DE USUARIO REGISTRADO .....</b>	<b>50</b>
<b>6.7. MENÚ ADMINISTRADOR .....</b>	<b>52</b>
<b>6.8. PÁGINAS ESTÁTICAS.....</b>	<b>55</b>
<b>6.9. VALIDACIONES JAVASCRIPT Y LIBRERÍA JQUERY .....</b>	<b>58</b>
<b>CAPÍTULO 7. HOJAS DE ESTILO EN CASCADA: CSS .....</b>	<b>61</b>
<b>7.1. INTRODUCCIÓN .....</b>	<b>61</b>
<b>7.2. EXPLICACIÓN DEL CÓDIGO UTILIZADO .....</b>	<b>64</b>
<b>CAPÍTULO 8. MANUAL DE USUARIO .....</b>	<b>79</b>
<b>8.1. CABECERA .....</b>	<b>80</b>
<b>8.1.1. REGISTRO DE USUARIOS .....</b>	<b>81</b>
<b>8.1.2. AUTENTICACIÓN DE USUARIOS .....</b>	<b>81</b>
<b>8.1.1.1. MODIFICAR CLAVE.....</b>	<b>82</b>
<b>8.1.1.2. MODIFICAR DATOS .....</b>	<b>82</b>
<b>8.1.1.3. ELIMINAR USUARIO .....</b>	<b>83</b>

<b>8.1.1.4. CERRAR SESIÓN .....</b>	<b>83</b>
<b>8.2. MENÚ HORIZONTAL.....</b>	<b>83</b>
<b>8.2.1. BOTÓN INICIO.....</b>	<b>83</b>
<b>8.2.2. BOTÓN NUESTRA WEB.....</b>	<b>84</b>
<b>8.2.3. BOTÓN BUSCAR RECETAS .....</b>	<b>84</b>
<b>8.2.4. BOTÓN ENVIAR RECETAS.....</b>	<b>85</b>
<b>8.2.5. BOTÓN GLOSARIO.....</b>	<b>87</b>
<b>8.2.5.1. ENLACE AGREGAR NUEVO TÉRMINO .....</b>	<b>87</b>
<b>8.2.6. BOTÓN AYUDA .....</b>	<b>88</b>
<b>8.2.7. BOTÓN ÁREA GESTIÓN.....</b>	<b>89</b>
<b>8.2.7.1. BOTÓN GESTIÓN USUARIOS .....</b>	<b>89</b>
<b>8.2.7.2. BOTÓN VALIDAR RECETAS .....</b>	<b>90</b>
<b>8.2.7.3. BOTÓN VALIDAR INGREDIENTES .....</b>	<b>91</b>
<b>8.2.7.4. BOTÓN VALIDAR TÉRMINO .....</b>	<b>91</b>
<b>8.3. COLUMNA IZQUIERDA .....</b>	<b>92</b>
<b>8.3.1. BÚSQUEDA POR PROVINCIA .....</b>	<b>93</b>
<b>8.3.2. BÚSQUEDA POR DIETA .....</b>	<b>94</b>
<b>8.3.3. BÚSQUEDA POR UTENSILIOS.....</b>	<b>95</b>
<b>8.3.4. BÚSQUEDA POR DIFICULTAD.....</b>	<b>95</b>
<b>8.3.5. BÚSQUEDA DE RESTAURANTES .....</b>	<b>95</b>
<b>8.3.6. ÚLTIMAS CINCO RECETAS.....</b>	<b>96</b>
<b>CAPÍTULO 9. CONCLUSIONES Y AMPLIACIONES .....</b>	<b>97</b>

# CAPITULO 1

# INTRODUCCIÓN

En este apartado describiremos brevemente los apartados del proyecto. Primero mostraremos los motivos que nos han impulsado a realizar este trabajo. En segundo lugar describiremos los objetivos que dicho proyecto debe satisfacer.

## 1.1. MOTIVACIÓN

El apasionante mundo de la cocina es un arte que tarde o temprano nos acaba gustando a todos. Todos los días comemos, mejor o peor, así que de gastronomía, el que más y el que menos puede opinar.

Al principio las recetas de cocina se iban propagando de generación en generación. Más tarde, surgieron libros o recetarios así como los programas de televisión basados en la cocina. ¿Quién no recuerda ver a su madre o a su abuela viendo la televisión y apuntando los ingredientes de alguna receta para luego hacerla en casa?

Hoy en día las tecnologías han avanzado mucho y ya se pueden encontrar recetas de cocina de cualquier lugar del mundo con las palabras adecuadas y un solo *click*.

Debido a la gastronomía tan diversa que poseemos en España, y en particular en Andalucía, he optado por hacer una página web de recetas de cocina que pueda ser una herramienta útil y asequible para personas que deseen aprender a cocinar, ya sean expertos o noveles en la materia pero el ámbito y las pretensiones de esta web serán globales.

## 1.2. OBJETIVOS

Los principales objetivos de “Gestión Web 2.0 Gastronómica” son los que abarcan el diseño de la página web básica, la base de datos, desarrollo de módulos.

- **Diseño de la página web**, realización de una página web dinámica, amena, atractiva, interactiva y de fácil manejo de recetas de cocina.
- **Base de datos**, diseño e implementación de una completa y flexible base de datos de recetas de cocina, que permita la consulta, modificación, inserción y eliminación de datos, así como su conexión y desconexión.
- **Desarrollo por módulos**, en ellos dotaremos de funcionalidad a nuestra página web de forma progresiva e intuitiva. Algunas de dichas funcionalidades son:
  - ✓ **Localizar por parte del usuario la receta deseada**, según la búsqueda deseada. El sistema permitirá hacer búsquedas variables, tales como buscar recetas por algún ingrediente principal, o con una lista de ingredientes variable, por lugar de procedencia o plato típico de una región concreta, por características nutricionales, etc.
  - ✓ **Gestionar nuevas recetas**, e incluirlas en la base de datos.
  - ✓ **Incluir una lista detallada de restaurantes**, a los que puedan acudir los usuarios de la página web atendiendo a su especialidad, lugar, etc. Estos

restaurantes estarán georeferenciados de forma que se muestren los que se encuentren cerca del lugar deseado. También se permitirán búsquedas de restaurantes según diversos criterios.

- ✓ **Registro y login de usuario**, para controlar el acceso a diversas áreas de los usuarios y dotarlos de ciertos privilegios.
- ✓ **Inclusión de fotos** por cada receta.
- ✓ **Añadir un glosario sobre técnicas de cocina**, en donde se definirán numerosas técnicas para la elaboración de recetas.

### 1.3. ESTRUCTURA DE LA MEMORIA

La memoria la hemos estructurado de la siguiente forma:

- **Capítulo 1:** en él haremos una breve introducción.
- **Capítulo 2:** en este capítulo haremos un estudio de diferentes páginas web de recetas de cocina.
- **Capítulo 3:** en él describiremos los programas utilizados para la realización de nuestro proyecto.
- **Capítulo 4:** en este capítulo describiremos nuestra base de datos.
- **Capítulo 5:** en este capítulo explicaremos los elementos HTML utilizados y algunos ejemplos de utilización en nuestro proyecto.
- **Capítulo 6:** explicaremos los elementos JAVASCRIPT y PHP utilizados en nuestro proyecto.
- **Capítulo 7:** explicaremos los estilos CSS aplicados a nuestra página web.
- **Capítulo 8:** en este capítulo haremos un manual para el usuario.
- **Capítulo 9:** finalizaremos el proyecto con nuestras conclusiones y unas posibles mejoras para el futuro.



# CAPITULO 2

## ANÁLISIS Y REQUISITOS

En este apartado haremos un estudio sobre diversas páginas web para obtener información sobre las características y servicios que ofrecen, para después desarrollar nuestra página web con unas características y servicios similares a las estudiadas.

### 2.1. ANÁLISIS DE LA WEB

#### ➤ *Web recetasdecocina*

Como podemos apreciar en la Figura 2.1 esta página web contiene un menú principal en el lado izquierdo en el que podemos observar que se estructura en referencia a los tipos de ingredientes. También consta de un enlace llamado “panel de usuarios” en

el cual pueden acceder los usuarios ya registrados así como los que se quieran registrar por primera vez. Los usuarios registrados recibirán un completo menú en su *e-mail* todos los días, esa es la única ventaja de estar registrado, ya que, esta web no permite añadir nuevas recetas a los usuarios. En la parte central de la página vienen las recetas favoritas de los usuarios, según el tipo de plato o bebida que sea. También cada día actualiza la página poniendo un menú distinto.



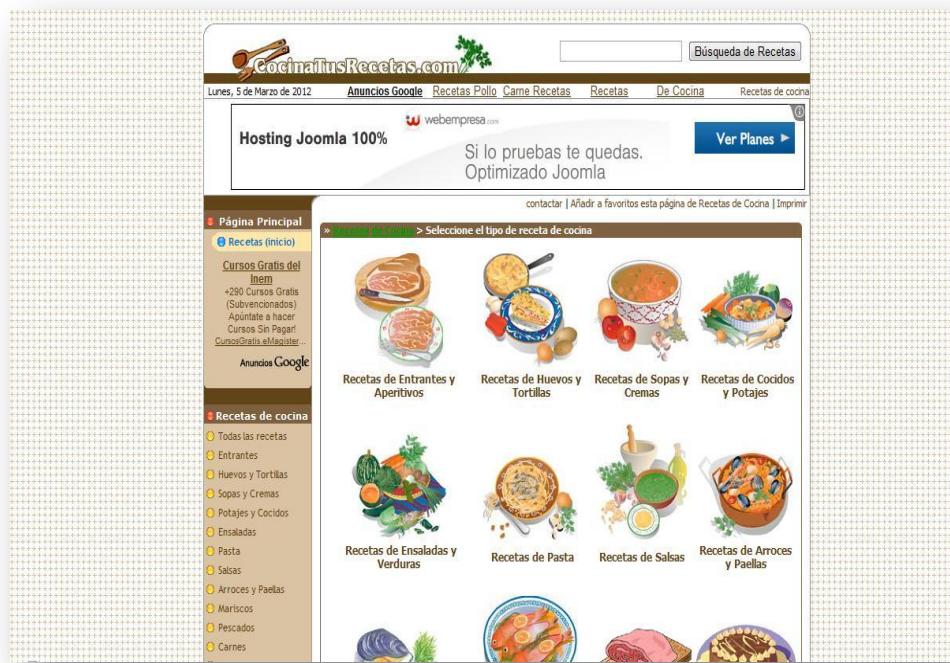
**Figura 2.1. Imagen de la web recetasdecocina.**

En lo que respecta al buscador de recetas, podemos realizar dos tipos de búsquedas, por título de la receta o por ingredientes y descripción. Tras realizar la búsqueda introduciendo las palabras deseadas aparecerán las diversas recetas (como máximo 50). Cada receta contendrá un número, el nombre de la receta y la categoría (verduras, pescados, carnes, etc.) a la que pertenece [1].

#### ➤ *Web cocinatusrecetas*

En la Figura 2.2 observamos que esta web contiene un menú principal en el lado izquierdo dividido en cuatro categorías: recetas de cocina, recetas de bebidas, según su época del año y según el país. Cuando pinchamos sobre cualquiera de los enlaces nos

aparece un listado de recetas. Cada receta contiene una imagen y un nuevo enlace en la que nos aparecerá su descripción. En la parte central de página aparecen imágenes de algunos de los distintos tipos de recetas que podemos encontrar. Pinchando sobre las respectivas imágenes observamos que el funcionamiento es el mismo que el de las diferentes secciones del lateral izquierdo.



**Figura 2.2. Imagen de la web cocinatusrecetas.**

Con lo que respecta al buscador, éste se encuentra en la parte superior de la página y permite búsquedas por palabras claves, mostrando todas las recetas que en su título o ingredientes contengan las palabras clave introducidas [2].

#### ➤ *Web recetario-cocina*

En el lateral izquierdo como se observa en la figura 2.3 se estructuran las diferentes categorías que podemos encontrar. Ésta página también contiene una sección que permite conectarse o registrarse a los usuarios. También permite añadir recetas con tu nombre, tener un recetario personalizado, leer titulares y escribir comentarios con tu nombre a los usuarios ya registrados.

Con lo que respecta al buscador de recetas, éste está situado en la parte superior de la página y permite realizar búsquedas introduciendo palabras clave que busca en el título de la receta. Aparece un listado con las recetas que contienen las palabras clave. Por cada

receta aparece el nombre, el número de lecturas, el número de votos, la categoría y la persona que la envió [3].



**Figura 2.3. Imagen de la web recetario-cocina.**

#### ➤ *Web lasrecetasdelchef*

En esta página como bien podemos observar en la Figura 2.4 encontramos en el lateral izquierdo todas las categorías que podemos encontrar, en la parte central podemos ver las recetas más destacadas. En esta página también se pueden añadir nuevas recetas sin necesidad de registro previo, tan solo necesitamos rellenar el formulario que nos aparece al pinchar en el botón “envíanos tu receta”.



**Figura 2.4. Imagen de la web lasrecetasdelchef.**

Con lo que respecta al buscador de nuevo se sitúa en la parte superior de la página y realiza las búsquedas en función a las palabras clave que introduzcas mostrando todas las recetas que contengan dichas palabras [4].

### ➤ *Web recetas*

En ésta página web encontramos las distintas secciones en el lateral izquierdo como vemos en la Figura 2.5. En el lateral derecho de la página podemos encontrar la receta de la semana y las recetas más populares. Para el registro de los usuarios tan sólo es necesario introducir el e-mail para recibir novedades y recetas. En el apartado izquierdo también encontramos un apartado que pone “envíanos tus recetas”, desde este apartado cualquier persona, este registrada o no, puede hacer pública una nueva receta con tan solo llenar un formulario, también puede hacer público su nombre y su correo si lo desea.



**Figura 2.5. Imagen de la web recetas.**

El buscador de recetas está situado en la parte superior de la página web y permite buscar por título de la receta o por ingredientes que contiene. Existe otra búsqueda más exhaustiva en la que debemos introducir el título o ingredientes, el tipo de plato, la nacionalidad, la dificultad, que no contenga ciertos ingredientes y seleccionar que sea baja en calorías o vegetariana [5].

### ➤ *Web quierococinar*

La estructura de esta página web es algo más simple que las anteriores como podemos apreciar en la Figura 2.6, contiene en la parte central imágenes de las distintas

categorías de recetas. Pinchando aparece un listado con las diferentes recetas. No permite registrarse, ni añadir recetas.



**Figura 2.6. Imagen de la web quierococinar.**

Ésta página no contiene buscador, ya que, pinchando sobre las diferentes imágenes aparece un listado de recetas [6].

#### ➤ *Web acocinar*

En la Figura 2.7 podemos observar que de nuevo el menú con las diferentes categorías se encuentra en el lateral izquierdo de la página. En la parte superior central existe una serie de categorías entre las cuales se encuentra un listado con algunos restaurantes.



**Figura 2.7. Imagen de la web a cocinar.**

En cuanto a las búsquedas muestra una lista de recetas de número variable por palabras clave [7].

### ➤ *Web recetasdiarias*

En ésta página web como podemos observar en la Figura 2.8 volvemos a encontrar las diferentes categorías en el lateral izquierdo. En la parte central de la página encontramos una lista con las últimas recetas introducidas. En el lateral derecho también podemos encontrar un top 10 de recetas.



**Figura 2.8. Imagen de la web recetasdiarias.**

Con lo que respecta al buscador se encuentra en el lateral derecho y permite hacer una búsqueda por palabras clave, mostrando todas las recetas que contengan dichas palabras clave en su título [8].

## 2.2. REQUISITOS FUNCIONALES

La interfaz de la página web debe dotar al usuario de la mayor comodidad posible al navegar por ella.

## **2.2.1. SERVICIOS DE LA PÁGINA WEB**

- **Presentación de la página web:** la página se dividirá en cinco apartados: cabecera, menú horizontal, columna izquierda, cuerpo y pie.
- **Realización de consultas:** la página web realizará la consulta correspondiente al introducir los datos que se deseen.
- **Publicación de recetas:** cualquier usuario que lo deseé podrá añadir una receta a la página web, el único requisito será que esté registrado con anterioridad. El administrador podrá exigir la validación de cada receta antes de publicarse. Se diferenciará entre dos tipos de usuarios: chef y aprendiz. El chef no necesitará validación por parte del administrador, mientras que el aprendiz si deberá ser validado por éste.
- **Modificación y eliminación de recetas:** el administrador de la página web y el usuario responsable de la receta podrán modificar o eliminar dicha receta en cualquier momento.

## **2.2.2. BÚSQUEDA DE RECETAS**

- **Realización de búsquedas:** cualquier usuario, ya esté registrado o no, podrá hacer búsquedas en la página web. El usuario introducirá sus preferencias eligiendo entre un conjunto de posibilidades: el título de la receta, ingredientes principales, tipo de restaurante, buscar platos que puedan hacerse con una lista de ingredientes variables teniendo en cuenta si son opcionales o no, etc.
- **Búsquedas excluyentes:** cualquier usuario podrá realizar dichas búsquedas, este registrado o no. Estas búsquedas consistirán, al contrario que las otras, en no contener cierto tipo de ingredientes, pudiendo ser muy útiles para personas con intolerantes o alérgicas.
- **Resultados de búsquedas:** las búsquedas se realizarán en la base de datos que crearemos para almacenar las recetas. Los resultados se mostrarán un listado con un número concreto de recetas.

### **2.2.3. GESTIÓN DE USUARIOS**

- **Registro de usuarios:** cualquier usuario tendrá la posibilidad de registrarse en la página web, llenando un formulario con sus datos personales. Dicho formulario contendrá como mínimo un nombre de usuario y correo electrónico que sean válidos.
- **Baja de usuarios:** cualquier usuario podrá darse de baja en cualquier momento, tan solo tendrá que introducir su correo y su contraseña en su panel de control.

## **2.3. REQUISITOS NO FUNCIONALES**

Los requisitos funcionales se dividen en requisitos de presentación y requisitos tecnológicos.

### **2.3.1. REQUISITOS DE PRESENTACIÓN**

- **Interfaz útil y fácil de manejar:** crearemos una página web que contenga un aspecto atractivo para el usuario y que se pueda manejar con facilidad para todas aquellas personas que deseen usarla.

### **2.3.2. REQUISITOS TECNOLÓGICOS**

- **Acceso a la página web:** la web debe estar disponible para todos los usuarios, ya estén registrados o no.
- **Acceso a la base de datos:** la base de datos debe estar disponible en cualquier momento.



# CAPÍTULO 3

# HERRAMIENTAS

En este capítulo describiremos los lenguajes, las aplicaciones y la instalación de los programas que vamos a utilizar para la realización del proyecto.

## 3.1. LENGUAJES

Haremos una breve mención de los lenguajes que utilizaremos para el diseño, servidor y cliente de la página web.

### **3.1.1. DISEÑO DE LA WEB: HTML Y CSS**

Para la generación del esqueleto de la página web optamos por HTML (*HyperText Markup Language*) y CSS (*Cascading Style Sheets*) entre otros.

- **HTML:** es un lenguaje muy básico y de uso bastante sencillo que se utiliza para la creación de páginas web. Su uso está basado en un conjunto de etiquetas que sirven para definir los distintos elementos que compondrán la página web. Para desarrollar una página web usando código HTML tan solo es necesario un editor de texto, al cual deberemos indicarle la extensión, que debe ser *html* o *htm* [9,10].
- **CSS:** también llamadas hojas de estilo en cascada. Es un lenguaje que se usa para definir la presentación de un documento escrito en HTML o XML. La intención de las CSS es la de separar la estructura de un documento de su presentación. Presenta una sintaxis muy sencilla y consiste en una serie de reglas. Algunas de las principales ventajas que presenta el uso de CSS son el control centralizado de la presentación de un sitio web completo, la separación del contenido de la presentación, la optimización del ancho de banda de la conexión y la mejora en la accesibilidad del documento [11,12].

### **3.1.2. SERVIDOR: PHP**

Para el servidor por su parte he elegido PHP porque es un lenguaje de programación interpretado, totalmente gratuito y que utiliza multitud de librerías reutilizables, diseñado para la creación de páginas web dinámicas. Habitualmente es usado del lado del servidor, pero también se puede usar desde una interfaz de línea de comandos. Es sencillo y su sintaxis es muy parecida a la de C y Perl. Se puede utilizar en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas de manera gratuita. Permite la conexión a diferentes servidores de bases de datos tales como MySQL u Oracle [13,14].

### **3.1.3. CLIENTE: JAVASCRIPT**

Para el lado del cliente he elegido JAVASCRIPT, ya que es un lenguaje de programación interpretado, que se define como orientado a objetos. Tiene una sintaxis similar a C, aunque adopta los nombres y las convenciones del lenguaje Java. Todos los navegadores actuales son capaces de entender el código JAVASCRIPT, aunque si utilizamos un navegador antiguo, pueden surgir problemas. Se utiliza normalmente en el lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas así como en bases de datos locales al navegador. El código JAVASCRIPT va incrustado en los documentos HTML y realizan acciones como petición de datos, mostrar mensajes, crear animaciones, etc.

## **3.2. APLICACIONES**

### **3.2.1. EDITORES: DREAMWEAVER CS5**

Para editar la web podríamos haber utilizado cualquier editor de texto, pero por comodidad usaremos Dreamweaver cs5, que es una aplicación dedicada a la construcción, diseño y edición de sitios y aplicaciones web. Es el programa más utilizado para el diseño y la programación web. La principal ventaja que posee es el gran poder de personalización y ampliación del mismo, ya que, sus rutinas están basadas en JAVASCRIPT, lo que les dota de una gran flexibilidad. Permite al usuario utilizar la mayoría de los navegadores web para previsualizar las páginas web. También permite el uso de extensiones, ofreciendo de esta manera funcionalidades añadidas a la aplicación [16].

### **3.2.2. DISEÑO DIAGRAMAS ENTIDAD-RELACIÓN: DIA**

Para realizar los diagramas entidad- relación utilizaremos DIA. Esta aplicación es un editor de diagramas que incorpora las herramientas necesarias para crearlos o modificarlos de manera muy fácil e intuitiva. Incluye diversas herramientas de dibujo para introducir diferentes elementos geométricos a nuestras composiciones, pudiendo editar sus propiedades de manera muy fácil y cómoda. También dispone de un espacio

cuadriculado para organizar nuestros diagramas y de un sistema de capas. Permite importar y exportar los dibujos realizados a los formatos más conocidos, además de tener su propio formato para editar el documento posteriormente. Algunas de las características más importantes son: su facilidad de uso, permite utilizar capas, exporta a la mayoría de formatos conocidos y que contiene una gran variedad de herramientas de dibujo. Por su parte entre las desventajas cabe destacar que posee un interfaz bastante mejorable [20].

### **3.2.3. DISEÑO DE LA BASE DE DATOS: MYSQL WORKBENCH**

Para realizar el diseño gráfico de la base de datos he optado por MySQL Workbench que es una herramienta para el modelado de bases de datos. Con esta herramienta se pueden crear, diseñar y validar esquemas de bases de datos, así como documentar bases de datos ya existentes o hacer migraciones a MySQL. Dispone de una representación visual de los objetos que permite entender rápidamente los esquemas y trabajar eficientemente con bases de datos simples y complejas. Además abarca desde la fase de diseño hasta la de implantación y mantenimiento, ofreciendo en cada etapa funciones que ayudan a automatizar el proceso. Entre las principales características que presenta destacan, la generación de scripts SQL, la validación de modelos y esquemas, la representación visual de las tablas, vistas, procedimientos almacenados y claves foráneas, ingeniería inversa para base de datos MySQL y que soporta MySQL 5, requisito imprescindible para que sea compatible con la base de datos de nuestro servidor local, en nuestro caso de WampServer [21].

### **3.2.4. SERVIDOR: WAMPSERVER 2.2 X32**

Para comprobar el funcionamiento de la página web durante su fase de desarrollo no será necesario colgarla en internet, sino que puede instalarse un servidor local en un ordenador doméstico. Dicho servidor contendrá la versión de Apache 2.2.21, la versión de MySQL 5.5.16 y la de phpMyAdmin 3.4.5.

➤ **APACHE:** es un servidor web HTTP de código abierto. Las principales ventajas que presenta son las siguientes: modularidad, multiplataforma, extensible y fácil de

encontrar ayuda o soporte [17]. Su uso principalmente está basado en el envío de páginas web estáticas y dinámicas en la World Wide Web.

- **MYSQL:** es un sistema gestor de bases de datos relacional, multiusuario, rápido, robusto y fácil de usar [18]. Se adapta bien a la administración de datos en entornos de red, especialmente en arquitecturas cliente servidor.
- **PHPMYADMIN:** permite administrar bases de datos MySQL tanto de manera local como remotamente, a través de un interfaz web fácil de usar. Está escrito en PHP y permite realizar todas las funciones típicas de MySQL [19].

### 3.3. INSTALACIÓN

En este apartado describiremos brevemente el proceso de instalación de los programas que necesitaremos para cumplir los objetivos que requiere nuestro proyecto.

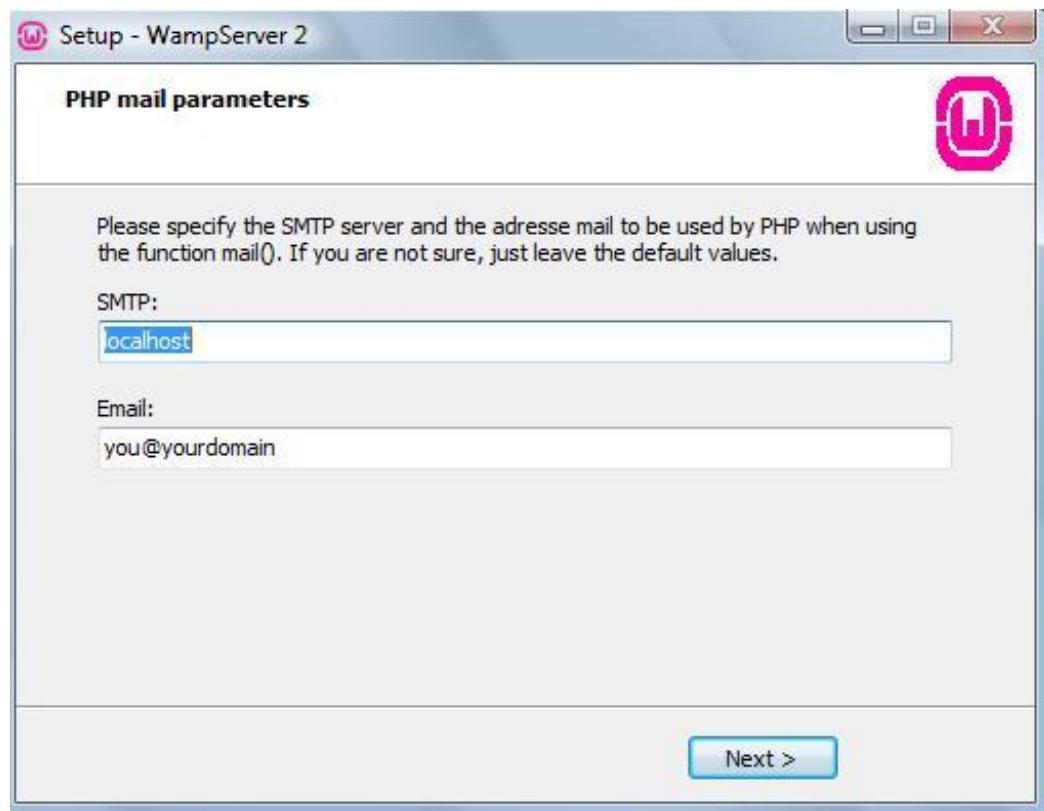
#### 3.3.1. WAMPSERVER 2.2 X32

WAMP es un acrónimo usado para definir una infraestructura de internet que utiliza las siguientes herramientas. Como sistema operativo utiliza Windows, como servidor web utiliza Apache, como gestor de base de datos utiliza MySQL como gestor de bases de datos y PHP usualmente como lenguaje de programación. Permite servir páginas HTML a internet, además de poder gestionar datos en ellas, al mismo tiempo que proporciona lenguajes de programación para desarrollar aplicaciones web [15].



Figura 3.1. Instalación de Wampserver.

Como podemos observar en la Figura 3.1 aquí aparecen las versiones de Apache, MySQL y phpMyAdmin citadas anteriormente y que utilizaremos para realizar el proyecto.

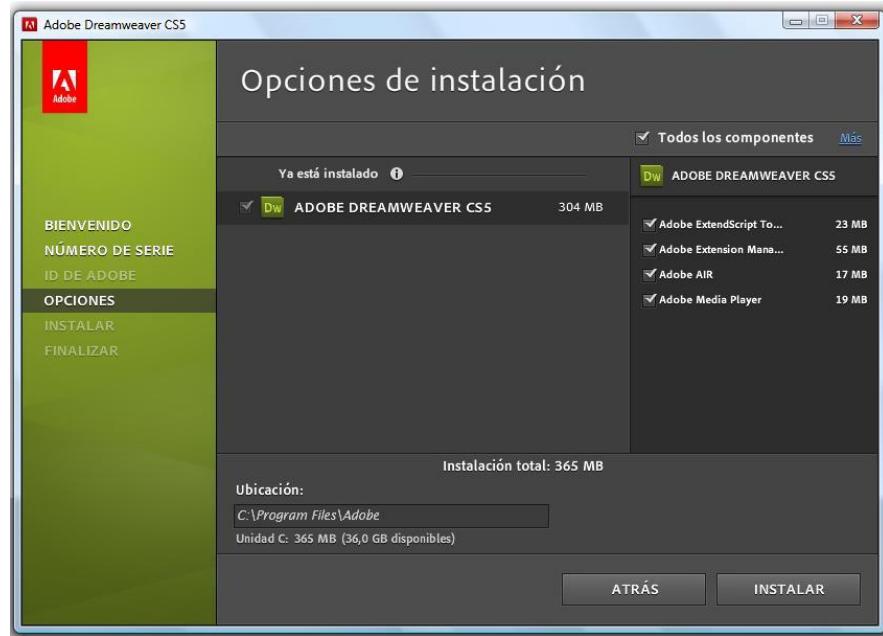


**Figura 3.2. Imagen de parámetros de configuración Wampserver.**

En la Figura 3.2 configuramos los parámetros para que Wampserver funcione correctamente. Al poner *localhost* en nuestro navegador si la instalación no ha dado ningún error, nos debería salir el menú de inicio de Wampserver donde aparecerán las herramientas que utilizaremos.

### 3.3.2. DREAMWEAVER CS5

La instalación de adobe Dreamweaver CS5 es de manera muy intuitiva, no hay que configurarlo de una manera especial, tan sólo hay que introducir el número de serie y seguir los pasos que se nos indican. Como podemos observar en la Figura 3.3 después de introducir el número de serie, el ejecutable nos preguntará que componentes deseamos instalar y su ubicación.



**Figura 3.3. Imagen opciones de instalación de Dreamweaver CS5.**

Una vez elegidas las opciones deseadas y pulsando el botón instalar, tan sólo nos restarán unos minutos para poder utilizar nuestro editor en nuestro ordenador.

### 3.3.3. MYSQL WORKBENCH 5.0 OSS

Esta es la versión gratuita de MySQL Workbench. El proceso de instalación de este programa es muy sencillo. Tan sólo hay que ejecutar el programa de instalación y pulsar el botón Siguiente de todas las ventanas que aparecen.



**Figura 3.4. Imagen instalación de MySQL Workbench 5.0 OSS.**

Por último como podemos apreciar en la Figura 3.4, nos indica donde se almacenará nuestro programa. Pulsamos en el botón *install* y ya tenemos instalado MySQL Workbench en nuestro equipo.

### 3.3.4. DIA 0.97.1

El primer paso del proceso de instalación de este programa será elegir el lenguaje en el que queremos que nos lo instale. En segundo lugar aceptaremos el acuerdo de licencia.



Figura 3.5. Instalación de DIA 0.97.1.

En tercer lugar como podemos apreciar en la Figura 3.5 elegiremos los componentes que deseamos instalar en nuestro ordenador. Pulsamos siguiente y en unos minutos ya podremos editar nuestros diagramas.

# CAPÍTULO 4

## DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS

Como ya reseñamos en el Capítulo 3, para el desarrollo y gestión de la base de datos utilizaremos phpMyAdmin y MySQL y SQL.

Se ha escogido phpMyAdmin porque facilita la interacción con las bases de datos y algunos servidores de internet, ya que suelen tener este gestor de bases de datos para el manejo de ellas. Una vez definida la base de datos, generamos en MySQL Workbench un *script* con todas las tablas y sus atributos, así como las diferentes relaciones existentes entre las diferentes tablas. Para su creación en phpMyAdmin basta con ejecutar en la consola de SQL dicho *script*.

Hemos elegido MySQL porque es mucho más útil para el acceso a las bases de datos tener un motor o servidor que haga las funciones de intérprete entre aplicaciones y usuarios con las bases de datos. Algunas de las principales ventajas que ello conlleva son las siguientes:

- **Seguridad:** consta de un sistema de privilegios y contraseñas que es muy flexible y seguro. Las contraseñas son seguras porque todo el tráfico de contraseñas está cifrado cuando se conecta a un servidor. Esto permite compartir datos sin que peligre la integridad de la base de datos o protegiendo determinados contenidos.
- **Portabilidad:** es un lenguaje estandarizado, por lo que las consultas hechas en SQL son fácilmente trasladables a otras plataformas o sistemas.
- **Conectividad:** permite realizar conexiones entre diferentes máquinas con distintos sistemas operativos.
- **Escalabilidad:** es posible manipular bases de datos muy grandes, del orden de 6000 tablas y 50 millones de registros.
- **Potencia:** SQL es un lenguaje muy potente a la hora de realizar consultas a las bases de datos.
- **Multiacceso:** las bases de datos pueden ser accedidas de manera simultánea por distintos usuario y/o aplicaciones.
- **Multihilo:** por lo que pueden beneficiarse de ello los sistemas multiprocesador.
- **Rendimiento:** presenta una gran velocidad al realizar las operaciones, lo que le hace ser uno de los gestores de bases de datos que mejor rendimiento presenta.
- **Fiabilidad:** MySQL está escrito en C y C++ y ha sido probado en multitud de compiladores [22].

Para el correcto desarrollo y funcionamiento de la base de datos durante su desarrollo deberemos tener siempre presente lo siguiente:

- **Consistencia de datos:** una misma información sólo puede ser almacenada una vez en la base de datos, si fuera almacenada más veces, se daría una redundancia de datos provocando una inconsistencia de datos.
- **Integridad de datos:** si la base de datos está disponible para varios usuarios de manera simultánea, deberán existir mecanismos para que las claves únicas no se repitan, para que se respeten las dependencias de existencia y que aseguren que las relaciones entre registros se mantengan de forma coherente [23].

## 4.1. DISEÑO DE LA BD. MODELO CONCEPTUAL

En este apartado realizaremos un estudio de los datos necesarios para realizar nuestra base de datos. Definiremos los objetivos que debe cumplir nuestra base de datos, identificaremos las entidades y las relaciones existentes entre ellas, haremos un primer diagrama, identificaremos los atributos que utilizaremos, así como las claves principales de las entidades y por último verificaremos el modelo.

### 4.1.1. DEFINICIÓN DE OBJETIVOS

Para la definición de nuestros objetivos será necesaria la distinción entre los diferentes usuarios que pueden navegar por nuestra página web. Podremos distinguir 3 tipos de usuarios: el administrador, el chef y el usuario normal, cada uno de ellos tendrá unos privilegios distintos que detallaremos a continuación.

- ✓ **Usuario normal.** Todos los usuarios que se registren en la página web, podrán modificar sus datos, clave o darse de baja cuando lo deseen. Además podrán enviar nuevas recetas, eso sí, el administrador deberá validarlas para que puedan ser vistas por los demás usuarios.
- ✓ **Usuario chef.** Podrá realizar las mismas operaciones que un usuario normal, la única ventaja que presenta respecto a éste es que no hará falta que las recetas que publique en la página web sean validadas por el administrador, ya que, daremos por supuesto que es un experto en el tema de la cocina.
- ✓ **Usuario Administrador.** Podrá realizar las mismas operaciones que un usuario normal. Será el encargado de validar las recetas de los usuarios normales para que puedan ser vistas por los demás usuarios. También se encargará de la edición de nuevos ingredientes en el caso de que les faltara algún atributo, podrá eliminar a los usuarios que le den un mal uso a la página web, dar de alta restaurantes e introducir nuevas palabras del glosario.

Con este breve resumen sobre los objetivos procederemos a distinguir las diferentes entidades y sus interrelaciones para poder realizar el diagrama E-R.

Los **restaurantes** tendrán asociados una serie de datos que los identificarán como su nombre, ciudad, provincia, dirección, teléfono, web, longitud, latitud y un campo observaciones. Por otra parte cada restaurante podrá tener asociada una serie de **imágenes** para facilitar la promoción de dicho restaurante.

Por otra parte cada **receta** contendrá información sobre su título, duración, dificultad, preparación, una serie de campos lógicos (horno, batidora, etc.) y un campo observaciones por si el propietario de la receta viera conveniente añadir algún comentario adicional. A su vez por cada receta podrá contener una serie de **imágenes**.

Por otra parte tenemos la entidad **usuario**, que contendrá los siguientes datos: nombre del usuario, apellidos, correo, clave, ciudad, país, dirección, teléfono, ocupación, nivel (chef o principiante) y un campo observaciones. Una serie de usuarios podrán ser **administradores** de la página web, con lo que ello conlleva, mantenimiento de la página web, validación de recetas, edición de nuevos ingredientes, etc. Por otra parte cualquier usuario podrá publicar **recetas** en la página web.

Por su parte cada **ingrediente** estará formado por los siguientes datos: un nombre, calorías, proteínas, hidratos, grasas (totales, saturadas, monoinsaturadas, poliinsaturadas) sodio y fibra. Todos estos valores serán referentes a una porción de 100 gramos de cada alimento, así como un campo observaciones. Un ingrediente podrá contener también una serie de imágenes, al igual que restaurantes y recetas.

Por último la entidad **glosario** será formada por una expresión y su correspondiente definición. Esta entidad será independiente de todas las demás.

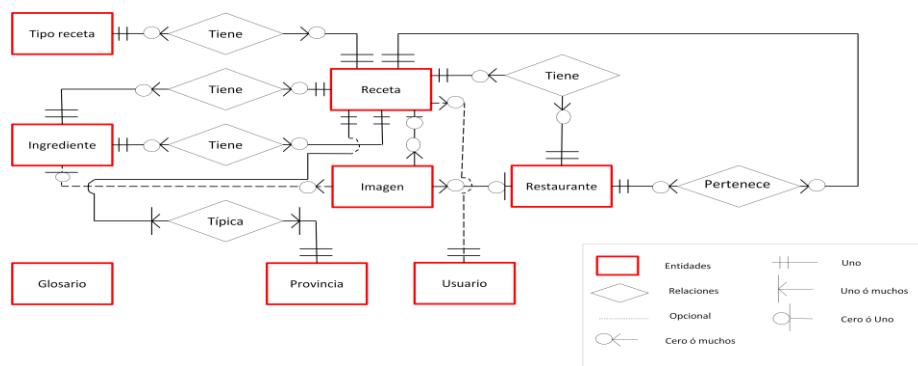


Figura 4.1. Diagrama E-R.

Con todo lo reseñado anteriormente un posible esbozo del diagrama E-R podría ser el que podemos apreciar en la Figura 4.1. Este diagrama no posee interrelaciones, dominios, etc. Su función será enfocar inicialmente el concepto y a partir de él comenzar a desarrollarlo. Una vez obtenidos los objetivos del problema a modelar pasamos al siguiente apartado.

#### 4.1.2. IDENTIFICACIÓN DE ENTIDADES Y ATRIBUTOS

Para almacenar la información en la base de datos, crearemos las siguientes entidades:

- ✓ **Usuario.** Contiene la información relativa de los distintos usuarios que se registrarán en la página web. Para cada usuario registrado en nuestra web, almacenaremos su **nombre**, **apellidos**, **correo**, **clave**, **ciudad**, **país**, **dirección**, **teléfono**, **ocupación**, **tipo\_usuario** (para saber si es chef o no), **admin** (para saber si es administrador) y un campo **observaciones** por si el usuario tiene algo que reseñar. Su clave primaria será llamada **id\_usuario** y será un entero con valor autoincrementado.
- ✓ **Glosario.** Contiene la información de diversas palabras relacionadas con técnicas de cocina. Esta entidad consta de dos atributos, una **expresión** y su correspondiente **definición**. Su clave primaria será el atributo llamado **expresión**.
- ✓ **Restaurante.** En esta entidad almacenaremos los datos correspondientes a cada restaurante. Por cada restaurante almacenaremos su **nombre**, **especialidad**, **ciudad** y **provincia** donde puede ser visitado, su **dirección**, su **teléfono** de contacto, así como las coordenadas de su **latitud** y su **longitud** donde puede ser encontrado. También almacenaremos un campo **observaciones** por si hubiera algo que reseñar. Su clave primaria se llamará **id\_rest** y será un entero autoincrementado.
- ✓ **Receta.** Por su parte de cada una de las recetas almacenaremos su **título**, su **duración**, su **dificultad**, y su modo de **preparación**, si utiliza **horno**, **batidora**, **microondas**, **thermomix**, si es apta para **celiacos**, **light**, **vegetariana** o **vegana**. También almacenaremos si está **validada**, su **fecha** de subida y el número de **comensales** para los que está pensada. Por último incluiremos un atributo **observaciones** por si alguien quisiera poner algún comentario adicional. Su clave primaria se llamará **id\_receta** y será un entero autoincrementado.

- ✓ **Imagen.** Para esta entidad almacenaremos los siguientes atributos: un nombre de la imagen, llamado **nombre\_imagen**, su **ruta** y una **descripción**. Su clave primaria será un entero autoincrementado llamado **id\_imag**.
- ✓ **Ingrediente.** Por último de cada ingrediente almacenaremos su **nombre**, **calorías**, **proteínas**, **hidratos**, **grasas totales**, **grasas saturadas**, **grasas monoinsaturadas**, **grasas poliinsaturadas**, **sodio** y **fibra** que contiene cada 100 gramos. También almacenaremos un campo **observaciones** por si fuera necesario introducir algún comentario adicional. Su clave primaria se llamará **id\_ingred** y será un entero autoincrementado.
- ✓ **Tipo\_receta.** Contendrá por su parte un único atributo, llamado **tipos\_receta**, el cual almacenará los diferentes tipos existentes de recetas (arroz, pescado, pasta, carne, etc.). Cada receta podrá pertenecer a su vez a varios tipos a la vez. Por ejemplo, la receta “Bacalao con tomate”, podría pertenecer al tipo pescado y primer plato. Su clave primaria será un entero autoincrementado llamado **id\_tipo\_receta**.
- ✓ **Provincia.** En esta entidad almacenaremos las diferentes provincias andaluzas. Constará de un único atributo: **nombre**, en el cual almacenaremos todas las provincias andaluzas (Huelva, Cádiz, Almería, Granada, Córdoba, etc.) y su clave primaria será un entero autoincrementado llamado **id\_provincia**.
- ✓ **Ingredientes\_principales:** esta entidad tendrá dos atributos: la **cantidad** de ingrediente utilizado en la receta y un campo **aclaraciones**, por si hubiera que hacer alguna puntuación.
- ✓ **Ingredientes\_opcionales:** tendrá los mismos atributos que los ingredientes principales, un atributo **cantidad** y otro llamado **aclaraciones**.

#### **4.1.3. DISEÑO FINAL EN MYSQL WORKBENCH**

En este apartado plasmaremos todo lo reseñado en los puntos anteriores utilizando MySQL Workbench para el diseño de las entidades y de sus interrelaciones. Para que quede más claro y legible el diseño de la base de datos, en la siguiente página podrán ver el resultado final en la Figura 4.2.

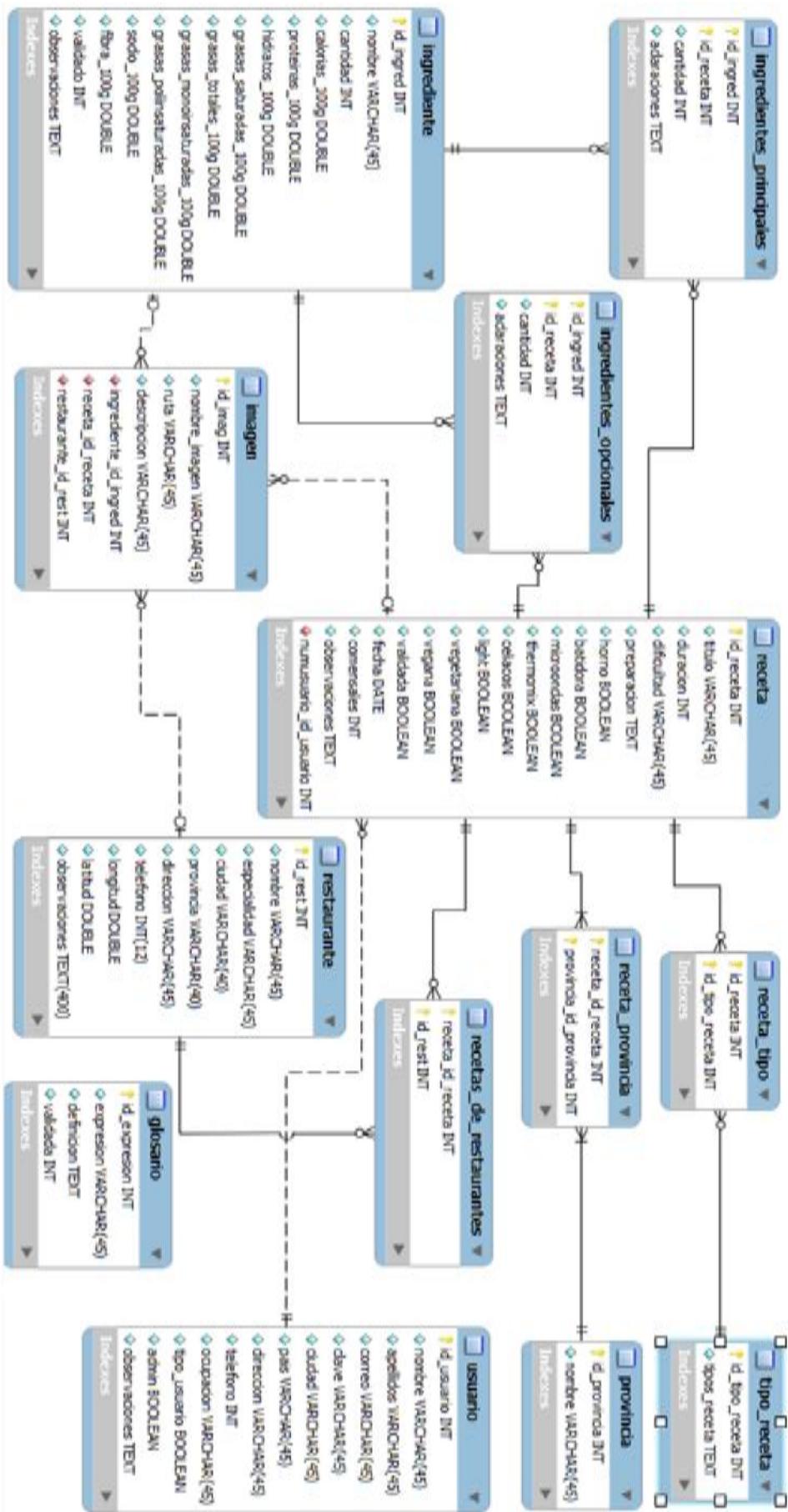


Figura 4.2. Diseño final en MySQL Workbench.



# CAPÍTULO 5

## DISEÑO HTML:

## ELEMENTOS

## UTILIZADOS

HTML, acrónimo de *HyperText Markup Language* («lenguaje de marcado de hipertexto»), hace referencia al lenguaje de marcado predominante para la elaboración de páginas web que se utiliza para describir y traducir la estructura y la información en forma de texto, así como para complementar el texto con objetos tales como imágenes. El HTML se escribe en forma de «etiquetas», rodeadas por corchetes angulares (<>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento.

Todas las funciones realizadas en la elaboración de nuestra página web contienen código HTML para su correcto funcionamiento. Al estar incrustado junto al código PHP lo más

conveniente será explicarlo en capítulos distintos. Algunas de las ventajas que presenta HTML serán las siguientes:

- Fácil de aprender y de entender, puedes hacer una página web con tan solo saber varias etiquetas HTML.
- Es estable, la mayoría de los lenguajes de programación se basan en HTML.
- Compatible con todos los navegadores web.
- Portable. HTML puede ser escrito usando cualquier editor de texto, ya que, no necesita un programa para desarrollar aplicaciones [24].

Con lo que respecta a la sintaxis, las etiquetas HTML suelen presentar dos partes, una apertura de forma general (*<p>*) y su correspondiente cierre (*</p>*).

Las partes en las que se puede dividir un documento HTML serán las siguientes:

- Un documento HTML deberá estar delimitado por las etiquetas *<html>* y *</html>*. Las mayúsculas y las minúsculas se reconocen igual, a la hora de escribir las etiquetas. Dentro de este documento distinguiremos dos partes principales:
- El encabezado: delimitado por *<head>* y *</head>* donde se colocarán etiquetas de índole informativa, como puede ser el título de nuestra página web.
- El cuerpo: que será donde colocaremos el texto y las imágenes y que estará delimitado por las etiquetas *<body>* y *</body>*.

El resultado será un documento con la siguiente estructura como el que podemos apreciar en la Figura 5.1.

```
<html>
  <head>
    <title> "Título de la página" </title>
  </head>
  <body>
  .....
  .....
  .....
  </body>
</html>
```

**Figura 5.1. Partes de un documento HTML.**

## 5.1. ELEMENTOS HTML

Existen una gran variedad de etiquetas en HTML. Algunas de las etiquetas básicas de HTML utilizadas en la creación de la página web serán las siguientes:

- **Párrafos:** en HTML un párrafo se definirá utilizando la etiqueta `<p>` y su correspondiente cierre. Se utilizan para introducir un bloque de texto. Los párrafos se podrán alinear a la derecha, izquierda o al centro con el atributo `align`. Ejemplos de esta etiqueta se pueden observar en los procedimientos destinados a la ayuda de los usuarios, en la página de inicio o en el apartado referente a nuestra web.
- **Saltos de línea:** para introducir un salto de línea en nuestro código utilizaremos la etiqueta `<br>`. A esta etiqueta no hace falta ponerle cierre. Esta etiqueta ha sido utilizada en casi todos los procedimientos creados.
- **Encabezados:** son párrafos con formato, es decir, con un tamaño de letra predefinido y en negrita. Existen diferentes tamaños, desde `<h1>` hasta `<h6>`, siendo `<h1>` el de mayor tamaño. A todos ellos habrá que ponerle sus correspondientes etiquetas de cierre. Algunos ejemplos de encabezados los podemos encontrar en nuestra página de ayuda (`h3`) o en la página de inicio, glosario o modificar datos de un usuario (`h1`).
- **Etiqueta `<div>`:** para aplicar estilo a una parte más amplia de una página HTML. Además con esta etiqueta también podremos crear divisiones en la página a las que podremos aplicar una serie de atributos para modificar su comportamiento. Su uso más destacado será convertir esa división en una capa cuyo comportamiento será muy diferente dentro del navegador. Esta etiqueta la he utilizado en casi todos los procedimientos creados.
- **Etiqueta `<span>`:** se utiliza para dar estilo a una pequeña parte de una página HTML. Hay que ponerle su correspondiente etiqueta de cierre. Esta etiqueta la he utilizado por ejemplo en la cabecera de la página web.
- **Negrita:** para poner texto en negrita podremos utilizar dos etiquetas: `<b>` y `<strong>` con sus correspondientes etiquetas de cierre. Esta etiqueta también la he utilizado bastante en casi todos los procedimientos creados.

A continuación pasaré a explicar algunas de las etiquetas HTML más complejas utilizadas en el diseño de la página web.

### 5.1.1. ENLACES

Un **enlace** o *link* es texto o imágenes en un sitio web que un usuario puede pinchar para tener acceso o conectar con otro documento. Los enlaces permiten conectar un sitio web o página web con otra, o incluso con una sección específica de una determinada web.

Se definen mediante la etiquetas `<a>` y `</a>` y su atributo más importante será `href` en el que se indicará la página o el archivo de destino. Dependiendo de este destino podemos distinguir entre los siguientes tipos de enlaces:

- **Enlaces internos:** los que se dirigen a otras partes dentro de la misma página.
- **Enlaces locales:** los que al pincharlos serán los encargados de dirigir al usuario a otra página dentro del mismo sitio web.
- **Enlaces remotos:** los que redirigen al usuario a otras páginas de otros sitio web.
- **Enlaces con direcciones de correo:** utilizados para crear un mensaje de correo que será enviado a una dirección determinada.
- **Enlaces con archivos:** para que los usuarios al pincharlos puedan descargar la información contenida en ese archivo.

Los enlaces que hemos utilizado en el diseño de la página web han sido principalmente los enlaces locales. Gracias a estos enlaces el usuario que pinche sobre ellos podrá ver el contenido de las diferentes páginas que conforman la web. Por ejemplo al pulsar el botón inicio del menú horizontal, se mostrará el contenido correspondiente al archivo “*inicio.php*”. De manera similar se procederá con todos los botones.

### 5.1.2. IMÁGENES

En este apartado describiré como se puede colocar una imagen usando HTML. Para colocar una imagen se utilizará la etiqueta `<img>`. Los principales atributos que pueden acompañar a una imagen serán los siguientes:

- **Src:** indica la ruta hacia el archivo que contiene la imagen.
- **Alt:** servirá para colocar una breve descripción de la imagen.
- **Height:** define la altura en pixels de la imagen.
- **Width:** define la anchura en pixels de la imagen.

- **Border:** define un borde que acompaña la imagen. El tamaño de este borde será indicado en pixels. Si queremos que el borde desaparezca pondremos este atributo a cero.
- **Vspace:** indica el espacio libre en la vertical que hay que colocar entre la imagen y otros elementos que la rodean.
- **Hspace:** igual que el anterior pero indicando el espacio libre en el plano horizontal.
- **Lowsrc:** Sirve para que el navegador muestra la imagen de baja resolución mientras se descarga la imagen original y el usuario pueda verla mientras tanto.

Las imágenes en HTML se utilizan para mostrar las fotos correspondientes a cada receta y restaurante y para mostrar las imágenes del menú y del submenú del administrador.

### 5.1.3. LISTAS

Para hacer enumeraciones de elementos podemos utilizar las listas. En HTML existen tres tipos de listas: las listas desordenadas, las listas ordenadas y las listas de definición.

- **Listas no numeradas:** serán delimitadas por las etiquetas `<ul>` y `</ul>` (*unordered list*) y cada uno de sus elementos será definido por la etiqueta `<li>` y su correspondiente cierre. También se podrá definir el tipo de viñeta utilizado mediante el atributo *type* que se situará dentro de la etiqueta `<ul>`. Este atributo puede tomar los siguientes valores: *disc*, *circle* y *square*, que se corresponden con un disco, un círculo o un cuadrado respectivamente.
- **Listas numeradas:** las listas ordenadas sirven para mostrar la información de manera numerada. Para ello utilizaremos las etiquetas `<ol>` y `</ol>` (*ordered list*) y los elementos que la compongan serán definidos igual que en las listas desordenadas, es decir, utilizando la etiqueta `<li>`. Al igual que las listas desordenadas permiten la posibilidad de modificar su estilo, es posible elegir entre números, letras minúsculas y mayúsculas y números romanos en minúscula o mayúscula utilizando el atributo *type* dentro de la etiqueta `<ol>`. Los valores que puede tomar este atributo serán los siguientes: 1 para ordenar por números, a

para ordenar por letras minúsculas, A para ordenar por letras mayúsculas, i para ordenar por números romanos en minúscula e I para ordenar por números romanos en mayúscula.

- **Listas de definición:** sirven para definir un conjunto de elementos con paras concepto-descripción, es decir, se especifican varios términos por su nombre y se escribe una definición para cada uno de ellos. Cada elemento es mostrado junto a su definición, uno tras otro.

Para realizar una lista de definición se utilizará la etiqueta `<dl>` y `</dl>` (*definition list*). La etiqueta para definir los diferentes elementos será `<dt>` (*definition item*) y para las definiciones será `<dd>` (*definition definition*).

En el proyecto hemos utilizado las listas desordenadas para mostrar los datos de cada restaurante (ciudad, provincia, dirección, teléfono y especialidad) y para mostrar los enlaces a las diferentes tipos de receta de la columna izquierda.

#### 5.1.4. TABLAS

Una tabla será un conjunto de celdas organizadas dentro de las cuales podremos alojar diferentes contenidos. HTML dispone de una gran variedad de etiquetas para crear tablas. La más importante es la etiqueta `<table>` y su correspondiente cierre `</table>`: serán las encargadas de definir la tabla. Dentro de ellas colocaremos las demás etiquetas:

- ✓ **Etiquetas `<tr>` y `</tr>`:** serán las encargadas de definir las filas que contendrá la tabla. Las tablas serán descritas por líneas de izquierda a derecha.
- ✓ **Etiquetas `<td>` y `</td>`:** serán las encargadas de definir cada celda. Dentro de estas etiquetas se almacenará el contenido que deseemos.

A continuación se hace un breve resumen de los atributos que pueden acompañar a cada línea de la tabla o a celdas en concreto.

Entre los atributos que permiten modificar toda una fila o una celda en concreto estarán los siguientes:

- **Align:** justifica el contenido de la celda del mismo modo que si fuera un párrafo.

- **Valing:** podremos elegir donde queremos el texto de la celda. El texto podrá ponerse arriba, en el centro o debajo de la celda.
- **Bgcolor:** da color a la celda o línea elegida.
- **Bordercolor:** define el color del borde de la celda o línea elegida.

Existen otros atributos que solo pueden ser asignados a una celda y que no son válidos para una línea o conjunto de líneas. Entre ellos destacan los siguientes:

- **Background:** nos permitirá colocar un fondo a una celda, a partir de una imagen.
- **Height:** define la altura cada una de las celdas. Esta altura se podrá definir en pixels o en porcentaje. Este atributo no funciona en todos los navegadores, por lo que hay que tener especial cuidado al usarlo.
- **Width:** define la anchura de la celda. Se puede expresar en pixels o como porcentaje.
- **Colspan:** expande una celda horizontalmente.
- **Rowspan:** expande una celda verticalmente.

Las tablas han resultado de gran utilidad en este proyecto, pues han servido para agrupar información. Un claro ejemplo sería la tabla de las recetas en las que se muestra la información nutricional de cada receta. Por otra parte también me han servido para estructurar los contenidos de los formularios. La mayoría de los formularios realizados están maquetados mediante tablas. Claros ejemplos de ello se pueden encontrar en los procedimientos para modificar los datos de un usuario o cambiar su clave.

### 5.1.5. FORMULARIOS

Los formularios en HTML serán muy importantes, ya que, de esta manera podremos intercambiar información con nuestros visitantes. En nuestro caso los formularios los utilizaremos para que los usuarios que lo deseen se registren, puedan enviar recetas, modificar sus datos personales, cambiar su contraseña, etc.

Los formularios serán definidos mediante las etiquetas `<form>` y `</form>`. Algunos de los atributos que pueden acompañar a dicha etiqueta serán los siguientes:

- **Action:** define el tipo de acción a llevar a cabo con el formulario. Existen dos posibilidades:

- ✓ El formulario es enviado a una dirección de correo electrónico.
- ✓ El formulario es enviado a un programa o *script* que es el que procesa su contenido. Para que sea el propio formulario el que se encargue de procesarlo pondremos la siguiente línea de código: `action=<?Php echo $_SERVER['PHP_SELF']?>`, de esta forma es el mismo formulario el que se encarga de procesar los datos como corresponda.

En el primero de los casos la información es enviada a la dirección de correo especificada en la etiqueta *action*.

En la segunda opción los datos son pasados a un programa o función que será el encargado de procesar los datos de manera conveniente. Este será el método usado para procesar los formularios de mi página web.

- **Method:** este atributo es el encargado de especificar la forma en la que el formulario será enviado. Los posibles valores que puede tomar este atributo serán dos: *post* y *get*. En la mayoría de los formularios utilizaré el método *post*. Para mostrar algunos mensajes de error utilizaré el método *get*.
- **Enctype:** se utiliza para indicar la forma en la que viajará la información que se mande a través del formulario. Para enviar un correo el valor de este atributo deberá ser “*text/plain*”, consiguiendo así, que es envíe el contenido del formulario como texto plano dentro del correo.

Por otro lado si queremos que el formulario sea procesado por otra función podremos omitir este atributo.

Ahora repasaremos los elementos de un formulario que sirven para introducir texto, entre ellos podemos distinguir los que sirven para almacenar textos cortos y textos largos.

- **Etiqueta input:** sirven para almacenar textos cortos. Su sintaxis será de la forma `<input>` Los atributos que pueden acompañar a esta etiqueta serán los siguientes:
  - ✓ **Type:** indica el tipo de los datos contenidos en la caja. En este caso serán de tipo “*text*”.
  - ✓ **Name:** indica el nombre que recibirá el contenido de la caja. Este nombre es de gran importancia a la hora de procesar los datos.
  - ✓ **Size:** define el tamaño de la caja de texto, es decir, el número de caracteres visibles. Si al escribir un usuario llega al final de la caja, el texto se

almacenará, pero irá desapareciendo parte del texto almacenado a la izquierda.

- ✓ **Maxlength:** indica el tamaño máximo del texto, que puede ser escrito en el campo. De existir este atributo, el navegador no permitirá introducir más caracteres de los indicados.
- ✓ **Value:** expresa un valor inicial en el campo, es decir, el valor que aparecerá por defecto en dicho campo.
- **Etiqueta input (texto oculto):** en determinadas ocasiones es necesario ocultar el texto para dotarlo de confidencialidad. Será igual que una etiqueta input normal lo único que hay que poner es en el atributo *type* que sea tipo “*password*”. Este tipo de input será muy útil para almacenar las contraseñas de los usuarios que quieran registrarse.
- **Etiqueta textarea:** esta etiqueta es utilizada para almacenar textos largos. Su sintaxis será de la forma `<textarea> </textarea>`. Este tipo de campos serán los elegidos para almacenar el modo de preparación de las recetas de cocina, los campos observaciones del registro de usuario, etc. Los atributos para indicar el tamaño de las cajas serán:
  - **Rows:** que indica el número de líneas del texto.
  - **Cols:** que indica el número de columnas del texto.

También podrán utilizarse los atributos utilizados para la etiquetas *input*.

Ahora describiremos brevemente otros elementos que pueden utilizarse dentro de un formulario:

- **Lista de selección única:** es un tipo de menú desplegable que nos permitirá seleccionar una única opción de las propuestas. Su sintaxis será la siguiente `<select> </select>`. Entre estas dos etiquetas pondremos las opciones que se pueden seleccionar con la etiqueta `<option> </option>`. Este tipo de selección única lo utilizaré por ejemplo, para seleccionar la provincia a la que se asociará cada receta, en el apartado de envío de recetas. Algunos de los atributos asociados a estas listas son los siguientes:
  - ✓ **Size:** indica el número de elementos mostrados a la vez en la lista. El resto de valores se verán desplazando la barra lateral de desplazamiento que aparecerá en el lado derecho.

- ✓ **Selected**: muestra la opción elegida por defecto.
- ✓ **Value**: define el valor que será enviado al programa encargado de realizar el procesamiento de los datos. Este atributo será muy útil, ya que, a cada provincia o tipo de receta le asignaremos un número que posteriormente será procesado para su inserción en la base de datos, de manera que es más fácil de procesar que un texto o palabra.
- **Lista de selección múltiple**: la sintaxis será la misma que la de la lista de selección única, lo único que en la etiqueta de apertura *select* habrá que ponerle el atributo “*multiple*”. Estas listas de selección múltiple las hemos utilizado por ejemplo, para elegir el tipo de cada receta, ya que, una receta puede pertenecer a su vez a varios tipos. Es aconsejable no utilizar listas de selección múltiple, ya que, para seleccionar varias opciones es necesario pulsar las teclas *ctrl* o *shift* y esto puede ser desconocido para el navegante.
- **Cajas de validación**: será un *input* en el que su *type* será “*checkbox*”. Se usan en el registro de usuarios para saber si el usuario registrado es chef o no y al enviar las recetas para que el usuario pueda seleccionar si una receta utiliza horno, batidora, es apta para celíacos, etc. Su funcionamiento es muy simple y se activa o desactiva haciendo un *click* sobre la caja.

En las páginas que reciben datos siempre se ha intentado usar formularios claros. Ejemplos pueden verse en el código de la columna izquierda para buscar los restaurantes o para registrarse los nuevos usuarios.

### 5.1.6. MARCOS DE PÁGINA

Esta etiqueta es la encargada de crear un espacio dentro de una página web, para incrustar otra web. Es un cuadrado cuyas dimensiones se definen con la etiqueta *<iframe>* en el cual se carga la página web que tiene asociada. Esa página web tendrá sus propios contenidos y estilos, que serán independientes del contexto donde se está mostrando. Además será totalmente funcional, es decir, si tiene enlaces se mostrarán en ese mismo espacio, y si tienes *scripts* o aplicaciones dentro se ejecutarán también de manera automática en ese espacio reservado.

Entre los contenidos que podemos mostrar dentro del espacio reservado por esta etiqueta se encuentran los siguientes:

- **Banners:** se invocan a través de un *iframe* pidiendo los datos del *banner*, que generalmente será un servidor de *banners*, que podría estar en otra página web.
- **Visualizar contenido de terceros:** se pueden utilizar para mostrar bloques de noticias o novedades que se publican en otras webs.
- **Interfaces de usuario:** en él se realizan ciertas actividades que se realizan de manera automática y que procesa otra página web.

Con lo que respecta a la sintaxis para hacer uso de ella se deberán utilizar las etiquetas *<iframe>* e *</iframe>*, dentro de la primera se podrán definir una serie de atributos entre los que destacan los siguientes:

- **Src:** indica la página web que se mostrará en el espacio reservado para la etiqueta *<iframe>*.
- **Width:** define la anchura del recuadro reservado.
- **Height:** define la altura de la etiqueta *<iframe>*.
- **Name:** para especificar un nombre asociado a la etiqueta *<iframe>* para referirnos si fuera necesario posteriormente.
- **Id:** indica el nombre del *<iframe>* para poder referirnos a él posteriormente.
- **Frameborder:** para definir si queremos el *<iframe>* con borde. A cero no se especifica borde a uno sí.
- **Scrolling:** indica si quiere que aparezcan barras de desplazamiento para que se muestre el contenido del *<iframe>* completamente. Existen tres posibles valores: *yes*, *no* y *auto*. “*Yes*” indica que aparezcan siempre las barras de desplazamiento, “*no*” indica que no se muestren nunca y “*auto*” (que es el valor por defecto) indica que se muestren cuando sean necesarias.
- **Marginwidth:** define el margen de la izquierda y derecha que debe tener la página que va dentro de esta etiqueta, con respecto al borde. Este margen va en *pixels* y prevalece respecto al margen que pueda tener la página web mostrada.
- **Marginheight:** tiene la misma función que el anterior, pero en este caso se especificará el tamaño del margen por arriba y por abajo.
- **Margin:** especifica la alineación del *<iframe>* y éste se especificará al igual que el de las imágenes.

- **Style y class:** definen el aspecto del *<iframe>* mediante hojas de estilo.

Esta etiqueta ha sido muy útil para georeferenciar los restaurantes dentro del marco indicado y mostrar así el contenido de google maps pasándole la longitud y latitud del restaurante deseado.

# CAPÍTULO 6

## DISEÑO E IMPLEMENTACIÓN EN PHP

PHP (*Hypertext Pre-processor*) fue creado originalmente por Rasmus Lerdof en 1994, sin embargo la implementación principal de PHP es producida actualmente por The PHP Group y sirve como estándar para PHP al no existir una especificación formal. Algunas de las principales características de PHP son:

- Es un lenguaje de programación que está orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno, por lo que se presenta como una alternativa de fácil acceso para todos.

- Facilidad de aprendizaje por su similitud con los lenguajes más utilizados para la programación estructurada (como C o Perl) y porque en su desarrollo se simplificaron distintas especificaciones, como es el caso de la definición de las variables primitivas.
- El código fuente escrito en PHP es invisible al navegador web y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Capacidad de expandir su potencial utilizando módulos.
- Posee una amplia documentación en su sitio web oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Permite aplicar técnicas de programación orientada a objetos.
- No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- Tiene manejo de excepciones (desde PHP 5).

Pero no todo son ventajas a la hora de utilizar PHP entre los principales inconvenientes encontramos los siguientes:

- Como es un lenguaje que se interpreta en ejecución, para ciertos usos puede resultar un inconveniente que el código fuente no pueda ser ocultado.
- Debido a que es un lenguaje interpretado, un script en PHP suele funcionar considerablemente más lento que su equivalente en un lenguaje de bajo nivel, sin embargo este inconveniente se puede minimizar con técnicas de caché tanto en archivos como en memoria [25].

## **6.1. BREVE DESCRIPCIÓN DE LOS APARTADOS DE LA WEB**

A continuación describiremos los apartados de los que consta la página web. Nuestra página se dividirá en cinco partes: una cabecera, un menú horizontal, una columna izquierda, un cuerpo y un pie de página.

- **Cabecera:** la cabecera será común a todas las páginas de nuestra web. En ella podemos encontrar el logotipo de nuestra web y en el lateral superior izquierdo encontraremos un apartado para el registro de usuarios y otro para el *logueo* de los usuarios ya registrados anteriormente. Para los usuarios que introduzcan su clave y su correo correctamente aparecerá un menú en el que si lo desean podrán modificar sus datos, cambiar su clave, darse de baja o cerrar la sesión anteriormente iniciada.
  
- **Menú horizontal:** en él podemos encontrar un listado de botones que nos mostrarán diferentes tipos de información.
  - ✓ **Inicio:** muestra un mensaje de bienvenida para los visitantes de la página web.
  - ✓ **Nuestra web:** en ella haremos una breve enumeración de las características más importantes de la dieta mediterránea. También se hará una referencia a las proteínas, hidratos de carbono y lípidos que debe comer una persona adulta al día.
  - ✓ **Buscar recetas:** si pinchamos sobre este botón podemos encontrar un buscador de recetas en la parte superior de la página en cual podremos buscar recetas que contengan varios ingredientes y que no contengan otros al mismo tiempo. También se mostrarán todas las recetas de la base de datos.
  - ✓ **Enviar recetas:** esté será el botón destinado a que los usuarios que lo deseen puedan colaborar con la web mandando sus recetas, para ello deberán estar registrados con anterioridad.
  - ✓ **Glosario:** en este apartado podremos encontrar una serie de expresiones relacionadas con el mundillo de la cocina y sus correspondientes definiciones.
  - ✓ **Ayuda:** en este apartado describiremos brevemente el contenido de la página web para situar al visitante y que pueda encontrar mejor los contenidos que deseé.
  - ✓ **Área gestión:** este botón sólo lo podrán ver los administradores de la página web.
  
- **Columna izquierda:** la columna izquierda será la encargada de realizar las búsquedas. Se podrán realizar las siguientes búsquedas:

- ✓ **Búsqueda por provincia:** aquí se mostrarán las recetas de la provincia seleccionada.
  - ✓ **Consultas por dieta:** aparecerán cuatro enlaces según el tipo de dieta: vegetariana, vegana, celíaco y light. Pinchando sobre cada uno de ellos aparecerán las recetas del tipo buscado.
  - ✓ **Consultas por utensilios:** se podrán hacer búsquedas de recetas por utensilios utilizados: horno, batidora, microondas y thermomix.
  - ✓ **Consultas por dificultad:** aparecerán cinco enlaces en los que se podrán buscar las recetas por su dificultad: muy baja, baja, media, alta o muy alta.
  - ✓ **Búsquedas de restaurantes:** se podrá buscar restaurantes por su provincia y especialidad principal.
  - ✓ **Últimas cinco recetas:** en este apartado se mostrarán las últimas cinco recetas insertadas (atendiendo a su fecha de subida) en la base de datos.
- **Cuerpo de la página:** será el encargado de mostrar el contenido correspondiente de los botones del menú horizontal y de la columna izquierda, a excepción de las últimas cinco recetas, que se mostrarán en una página en blanco distinta.
- **Pie:** al final de la página aparecerá mi nombre y el año de finalización del proyecto fin de carrera.

Atendiendo a los diferentes tipos de usuario que puede haber, éstos tendrán una visión distinta de la página web, así como, podrán gozar de una serie de privilegios.

## 6.2. VISIÓN DE UN USUARIO NO REGISTRADO

Podrá ver todo el contenido de la página web anteriormente reseñado, lo único que no podrá ver será el formulario de envío de recetas y el de envío de nuevos términos que estará reservado a los usuarios registrados anteriormente. Para ello los usuarios nuevos que quieran registrarse encontrará en la cabecera, justo encima del *login* de usuarios un enlace a un formulario de registro.

## 6.3. VISIÓN Y PRIVILEGIOS DE UN USUARIO REGISTRADO

Al igual que el usuario no registrado podrá ver los mismos contenidos pero podrá enviar recetas o nuevos términos. También al hacer el *login* de usuario aparecerá en la parte superior de la cabecera un listado en el usuario podrá cambiar su contraseña, modificar sus datos, darse de baja o cerrar la sesión iniciada anteriormente. También debemos tener en cuenta que se podrá distinguir entre dos tipos de usuarios registrados:

- **Usuario chef:** a la hora de enviar una receta esta receta será publicada directamente en la página web, es decir, no hará falta validación de la receta por parte del administrador.
- **Usuario registrado no chef:** por su parte los demás usuarios que suban una receta y no sean chefs deberán esperar a que el administrador valide su receta para que pueda ser publicada en la página web.

## 6.4. VISIÓN Y PRIVILEGIOS DE USUARIO ADMINISTRADOR

La visión será la misma que la de un usuario registrado, al identificarse en el sistema (*loguearse*) podrá modificar sus datos, cambiar la contraseña, darse de baja o cerrar la sesión. En cuanto al menú horizontal aparecerá un nuevo botón llamado área gestión, que al pulsarlo nos muestra un submenú en la parte inferior, en el cual el administrador podrá validar o eliminar recetas, ingredientes o términos e incluso podrá eliminar a usuarios que estén realizando un mal uso de la web.

## 6.5. FUNCIONES INCLUIDAS EN TODAS LAS DEMÁS

Estas funciones serán el esqueleto de nuestra página web y por ello deberán ser incluidas en todas las demás para que siempre tenga el mismo diseño. Estas funciones serán

la cabecera, el menú horizontal, la columna izquierda, el pie y las funciones comunes. A continuación se explicará brevemente su código.

- **Cabecera:** en nuestra cabecera tendremos un formulario para que los usuarios ya registrados introduzcan sus credenciales y puedan enviar recetas, agregar nuevos términos, modificar sus datos, clave o cerrar la sesión en caso de coincidir su correo y clave con las registradas en la base de datos. También constará de un enlace a un formulario para las personas que quieran registrarse por primera vez como podemos observar en la parte superior derecha de nuestra cabecera. La cabecera de nuestra página web la podemos observar en la parte superior de la Figura 8.1 del manual de usuario.
- **Columna izquierda:** dicha columna será la encargada de mostrar los distintos tipos de recetas. Las búsquedas se podrán hacer de diversas maneras: se podrán buscar las recetas correspondientes a cualquier provincia andaluza, por dietas, por tipo de utensilios utilizados y por dificultad. También se podrán buscar restaurantes que sean de una determinada provincia y sean de una determinada especialidad. Por último en esta columna izquierda aparecerá un enlace a las 5 últimas recetas insertadas en nuestra base de datos. Para diseñarla he utilizado un formulario en el que se puede seleccionar una sola provincia, para las recetas de cada provincia. Para los tipos de dieta, utensilios utilizados y dificultad hemos utilizado una lista desordenada sin decoración en cada uno de sus elementos. Por su parte para los restaurantes colaboradores he utilizado un formulario en el que los datos de los *select* se pasan por el método *post*. Por último para mostrar las 5 últimas recetas hago un acceso a nuestra base de datos y llamo a mi función creada en el archivo “funciones\_comunes.php”. Todo el resultado de lo explicado anteriormente lo podemos observar en la Figura 8.18 del manual de usuario.
- **Funciones\_comunes.php:** en ella podemos encontrar las funciones que se encargarán de realizar la conexión y desconexión con la base de datos, una función para comprobar si un usuario está autenticado y una función para seleccionar las últimas 5 recetas introducidas en la base de datos. También abriremos una sesión para almacenar los datos que necesitamos para tratarlos posteriormente.

- ✓ **Función para autenticar a un usuario:** será llamada `estas_autenticado` y no se le pasará ningún parámetro. Comprueba si existen datos de sesión correspondientes al nombre y al correo, si existen devuelve `true`, en caso contrario devolverá `false`. Para comprobar si existen datos en dichas variables de sesión utilizaremos `isset`, que comprueba si dicha variable no está vacía.
- ✓ **Función para conectar con la base de datos:** está función constará de tres líneas de código. En la primera almacenaremos en una variable (`$conexión`) seleccionaremos con la función `mysql_connect` la conexión que estará disponible en toda la página web, en nuestro caso “`localhost`”, el usuario será `root` y la contraseña será la cadena vacía, que son las que vienen por defecto al instalar WampServer. En la segunda línea elegiremos la base de datos a la que deseamos conectarnos, en nuestro caso `bd_recetas` que es donde guardaremos la información de las recetas, usuarios, etc. Por último haremos un `return` de la conexión para utilizarla posteriormente cuando nos haga falta.
- ✓ **Función para la desconexión de la base de datos:** su implementación es muy sencilla, ya que, sólo hay que llamar a la primitiva `mysql_close` pasándole el valor de la conexión abierta anteriormente. Esta función es bastante importante, ya que, si la página llegara a ser muy visitada y no se cerraran correctamente las conexiones podría llegar un momento en el que no podrían realizarse más. En MySQL el número de conexiones simultáneas está en torno a unas 100.
- ✓ **Función para mostrar las últimas 5 recetas:** lo primero que haremos será conectar con la base de datos. Luego seleccionaremos con un `select` las 5 últimas recetas ordenadas descendenteamente por fecha y la almacenaremos en una variable (`$sql`). A continuación guardamos en otra variable (`$registros`) el resultado de ejecutar la sentencia `select`, mediante un `mysql_query` al que le pasamos la variable que contiene la sentencia `select` y la conexión que abrimos al principio. Por último retornaremos el resultado almacenado en dicha variable (`$registros`).
- **Menú:** será el menú principal que verán todos los usuarios de la web, a excepción de los administradores, que verán un botón más llamado área gestión y estará formado por los botones que podemos observar en la Figura 8.1 del manual de usuario.

- **Pie:** esta función será llamada “pie.php” y simplemente muestra mi nombre junto al año de finalización del proyecto. El pie de nuestra página web lo podemos observar en la parte inferior de la Figura 8.1 del manual de usuario.

## 6.6. MENÚ DE USUARIO REGISTRADO

Un usuario registrado podrá modificar sus datos, modificar su clave, darse de baja o cerrar sesión con el menú que le aparecerá en la parte superior derecha de la cabecera de la página web.

- **Eliminar usuario:** para borrar un usuario ya registrado de la base de datos he utilizado dos funciones llamadas “borrarUsuario.php” y “borrarUsuario2.php”. La primera función contendrá el formulario a llenar por el usuario para darse de baja. Para ello el usuario deberá introducir correctamente su correo y su clave. Como en casi todas las funciones esta función tendrá incluidas las funciones comunes, la cabecera, el menú y la columna izquierda, así como el pie de la página web. Los datos introducidos en el formulario serán pasados a la función “borrarUsuario2.php” mediante el método *post*. Esta segunda función será la encargada de tratar los datos introducidos en el formulario. Primero realiza la conexión a la base de datos, luego almacena en dos variables los datos introducidos en los *input* del formulario, esos datos serán el correo y la clave. Luego nos creamos una sentencia SQL (*delete*) en la que eliminamos al usuario que esté almacenado en la base de datos y que coincida con los datos introducidos en el formulario. Si esta sentencia se ejecuta bien, se mostrará un mensaje confirmando la eliminación del usuario, en caso erróneo se mostrará un mensaje de error, por último cerraremos la conexión con nuestra base de datos. En esta función también se incluirán las funciones comunes, la cabecera, el menú, la columna izquierda y el pie de página. Una imagen del formulario para eliminar a un usuario la podemos observar en la Figura 8.5 de nuestro manual de usuario.
- **Cerrar sesión:** esta función será bastante sencilla y será la encargada de cerrar una sesión iniciada con anterioridad. Será llamada “logout.php”. En ella en primer

lugar incluiremos las funciones comunes que es donde iniciamos las variables de sesión y posteriormente eliminaremos todas las variables de sesión, destruiremos la sesión y redirigiremos a la página de inicio al usuario.

- **Modificar clave:** este apartado también constará de dos funciones llamadas “*modificarClave.php*” y “*modificarClave2.php*”. Ambas incluirán las funciones comunes, cabecera, menú, columna izquierda y pie de la página web. La función “*modificarClave.php*” mostrará por pantalla un formulario en el que aparecerá ya el correo del usuario que desea cambiar la clave. El usuario deberá introducir su clave antigua y clave nueva dos veces. Es importante que las dos claves nuevas coincidan, ya que, de no ser así no se cambiará la contraseña. Los datos del formulario se pasarán mediante el método post a la función “*modificarClave2.php*” que será la encargada de realizar la conexión con la base de datos. A renglón seguido almacenaremos las dos claves nuevas en dos variables, luego compararemos esas dos variables mediante la función para comparar dos cadenas *strcmp*, en caso afirmativo nos crearemos una sentencia SQL (*update*) en la que cambiaremos la clave del usuario almacenado en la base de datos en la que el correo y la clave antigua introducidas en el formulario coincidan. Si la sentencia se ejecuta correctamente mostraremos un mensaje confirmando que la clave ha sido cambiada, mientras que si hay algún error también se mostrará su correspondiente mensaje de error, pasándole el error por *get* al *index* de nuestra página web. Por último cerraremos nuestra conexión con nuestra base de datos. El resultado se puede observar e la Figura 8.4 de nuestro manual de usuario.
- **Modificar datos:** al igual que los dos apartados anteriores también constará de dos funciones para poder modificar los datos de un usuario registrado. Estas dos funciones incluirán las funciones comunes, cabecera, menú, columna izquierda y pie de la página web y se llamarán “*modificarDatos.php*” y “*modificarDatos2.php*”. La primera función será la encargada de mostrar un formulario con los datos del usuario *logueado* anteriormente. Para ello nos creamos un formulario en el que aparecerán todos los datos del usuario: nombre, apellidos, correo, clave, ciudad, país, dirección, teléfono, ocupación, si es chef o no (*checkbox*) y un campo observaciones por si el usuario quisiera añadir algo. El usuario encontrará la misma información que introdujo cuando se registró por primera vez en los *input* del

formulario. El usuario modificará los datos del formulario que desee y éstos serán pasados por el método *post* a la función “modificarDatos2.php”. Esta función realizará la conexión con la base de datos, y actualizará la base de datos mediante una sentencia SQL (*update*). En dicha sentencia se actualizarán todos los campos modificados anteriormente por el usuario que su correo coincida con el almacenado en la base de datos. Para saber si se ha marcado el *checkbox*, nos crearemos una variable e inicialmente la pondremos a cero. Si se encuentran datos en el *checkbox*, es decir, si se ha marcado, la cambiamos a uno. Si la sentencia SQL se ejecuta correctamente se mostrará un mensaje confirmando que los datos han sido modificados correctamente, en caso contrario mostraremos el correspondiente mensaje de error. Por último nunca olvidaremos cerrar nuestra conexión con la base de datos iniciada anteriormente. Una imagen del formulario para modificar los datos de un usuario se puede observar en la Figura 8.2 de nuestro manual de usuario.

Todos los enlaces a estas funciones que aparecen en nuestra cabecera se pueden observar en la Figura 8.3 del manual de usuario.

## 6.7. MENÚ DE ADMINISTRADOR

El administrador o administradores de la página web verán un botón nuevo llamado “Área Gestión” al autenticarse, en el que pulsando sobre él aparecerá un submenú botones nuevos en el que podrá eliminar un usuario y validar recetas e ingredientes.

- **Área gestión:** es la función que muestra que al pinchar sobre el botón del menú horizontal área gestión nos mostrará un submenú con las distintas opciones que puede realizar el administrador. La función será llamada “areagestion.php” y en ella incluiremos las funciones comunes, cabecera, menú, columna izquierda y el menú de gestión de los administradores. Luego mostraremos un mensaje de bienvenida para el administrador e incluiremos el pie de página. Una imagen de este botón la podremos ver en la Figura 8.13 de nuestro manual de usuario.

- **Menú administrador:** este submenú aparecerá al pulsar el botón del menú horizontal “ÁREA GESTIÓN” y como bien sabemos sólo será visible para el administrador. Estará formado por cuatro botones: uno para gestionar a los usuarios, llamado “GESTIÓN USUARIOS”, otro para validar o eliminar recetas, llamado “VALIDAR RECETAS”, otro para validar o eliminar los ingredientes, llamado “VALIDAR INGRED.” y otro validar o eliminar términos llamado “VALIDAR TÉRMINO”. Cada botón al pulsarlo nos llevará a las funciones correspondientes para eliminar un usuario, validar una receta y validar un ingrediente respectivamente. También se mostrará un mensaje de bienvenida para el administrador, como podemos observar en la Figura 8.13 de nuestro manual de usuario.
  
- **Gestión de usuarios:** constará de dos funciones “*gestionUsuarios.php*” y “*gestionUsuarios2.php*”. Como siempre ambas incluirán a funciones comunes, cabecera, menú, columna izquierda y pie de nuestra página web. La primera función será la encargada de mostrar los datos de un usuario, mediante una tabla, por su parte la segunda función será la encargada de eliminar al usuario si el administrador lo estimara oportuno. El administrador tan sólo tendría que pulsar el botón eliminar de la persona que deseara eliminar. Al finalizar nunca nos olvidaremos cerrar la conexión con nuestra base de datos. El resultado de todo esto se puede observar en la Figura 8.14 del manual de usuario.
  
- **Validar ingrediente:** constará de dos funciones “*validaingrediente.php*” y “*procesarValidarIngrediente.php*”. La primera incluirá las funciones comunes, cabecera, menú, columna izquierda y pie de nuestra página web. La primera función realizará la conexión con la base de datos y seleccionará los ingredientes que no estén validados (campo validado a 0) mediante una sentencia *select*. Esta sentencia se almacenará en un registro y si el número de columnas es igual a cero se mostrará un mensaje diciendo que no hay ingredientes por validar. Mientras el número de ingredientes por validar sea mayor que cero, se mostrará la información de cada ingrediente en un formulario (nombre, calorías, proteínas, hidratos, grasas saturadas, monoinsaturadas, poliinsaturadas y totales, sodio, fibra y su correspondiente campo de observaciones). También mediante un *input* de tipo *hidden* (oculto) almacenaremos el identificador de la receta, para utilizarlo

posteriormente para buscar el ingrediente y actualizarlo. Despu s de realizar la consulta cerraremos la base de datos. La funci n “*procesarValidarIngrediente.php*” ser  la encargada de ver que bot n ha sido pulsado. Primero abrimos una nueva conexi n con nuestra base de datos y luego comprobamos que bot n ha sido pulsado. Si se pulsa el bot n validar hacemos un *update* con la informaci n contenida en el formulario actualizando los datos del ingrediente que coincide con el identificador del campo oculto del formulario. En dicha actualizaci n cambiaremos el valor del campo validado a uno. Si por el contrario pulsamos en el bot n eliminar haremos un *delete* del ingrediente que coincide con el identificador del campo oculto del formulario. Tambi n mostraremos un mensaje para el administrador indicando que el ingrediente ha sido validado o eliminado. Por  ltimo cerraremos la conexi n con nuestra base de datos. El resultado del formulario, as  como el dise o de cada ingrediente se podr  observar en la Figura 8.16 de nuestro manual de usuario.

- **Validar receta:** el funcionamiento ser  exactamente igual que el de las funciones de validar un ingrediente. Constar  de dos funciones “*validareceta.php*” y “*procesarValidarReceta.php*”. Lo  nico que cambian ser n los nombres de los campos del formulario, ya que, en vez de mostrarse los datos de un ingrediente se mostrar n los campos de una receta, es decir, por cada receta se mostrar  su t tulo, duraci n, dificultad, preparaci n, si utiliza horno, batidora, microondas, thermomix, si la receta es apta para cel acos, light, vegetariana o vegana (todos estos campos anteriores son campos son checkbox), duraci n, fecha de subida, el n mero de comensales a los que va dirigido y un campo observaciones. El formulario tambi n contendr  un *input* oculto para despu s a la hora de actualizar la receta la podamos localizar y modificar correctamente. El contenido y el dise o del validador de recetas se podr  observar en la Figura 8.15 del manual de usuario.
  
- **Validar t rmino:** el funcionamiento ser  id ntico al de validar una receta o ingrediente. Tambi n constar  de dos funciones llamadas “*validarTermino.php*” y “*procesarValidarTermino.php*”. Lo  nico que cambia son los nombres de los campos del formulario. Estos nombres se pueden apreciar en la Figura 8.17 del manual de usuario.

## 6.8. PÁGINAS ESTÁTICAS

En estas páginas el usuario podrá ver la información pero no podrá modificar ni enviar datos. Estas páginas siempre contendrán el mismo contenido y serán las siguientes:

- **Inicio:** será la primera página que verá el usuario al visitar nuestra página web. En ella se mostrará un mensaje de bienvenida y una breve descripción del contenido de la página web, como se puede apreciar en la Figura 8.6 del manual de usuario.
- **Ayuda:** esta función será llamada “*ayuda.php*” y su implementación constará de los siguientes pasos. En primer lugar, incluiremos las funciones comunes, cabecera, menú y columna izquierda de nuestra página web. Luego mediante párrafos iremos comentando los contenidos y secciones en los que se dividirá nuestra web como podemos apreciar en la Figura 8.12 de nuestro manual de usuario. Por último incluiremos el pie de nuestra página web.
- **Glosario:** esta función será la encargada de mostrar los distintos términos almacenados en la base de datos y será llamada “*glosario.php*”. En primer lugar incluiremos las funciones comunes, cabecera, menú y la columna izquierda. Después conectamos con nuestra base de datos y hacemos una consulta seleccionando todos los datos existentes en la tabla glosario guardándolos en un registro. Luego sacaremos del registro una fila con la función *mysql\_fetch\_array* y hasta que la fila no sea vacía, iremos mostrando su correspondiente nombre y expresión. Por último cerraremos la conexión con la base de datos e incluiremos nuestro pie de página. Un ejemplo de cómo se muestran los datos del glosario puede observarse en la Figura 8.10 del manual de usuario.
- **Enlace agregar nuevo término:** a través de este enlace los usuarios registrados que lo deseen podrán agregar nuevos términos a nuestro glosario. El formulario para agregar un nuevo término lo podemos observar en la Figura 8.11 de nuestro manual de usuario. Los campos expresión y definición serán obligatorios.
- **Nuestra web:** como siempre incluiremos las funciones comunes, cabecera, menú y columna izquierda. Luego encontraremos cuatro párrafos: en el primero de ellos

haremos referencia a la dieta mediterránea y en los tres siguientes hablaremos de las proteínas, hidratos y lípidos que debe tomar una persona al día. Por último incluimos el pie de página. Parte del contenido de nuestra web lo podemos observar en la Figura 8.7 de nuestro manual de usuario.

- **Enviar receta:** los datos que deberán llenar los usuarios registrados para mandar recetas serán los siguientes: título de la receta, tipo de plato, provincia, dificultad de la receta, número de comensales para los que está pensada, si utiliza horno, batidora, microondas, thermomix, si es una receta vegetariana, vegana, apta para celíacos, light, tiempo de preparación, introducir una foto si lo estimara oportuno, introducir los ingredientes principales y opcionales, el modo de preparación, su correo y un campo observaciones por si quisiera reseñar algo más como podemos observar en las Figura 8.9 de nuestro manual de usuario. Todos estos datos del formulario serán procesados por otra función que será la encargada de insertar la receta en la base de datos, si los datos fueran correctos.
- **Mostrar recetas:** las búsquedas de recetas se harán de diferentes tipos: por provincias, por tipo de receta, por utensilios utilizados, por dificultad y por las últimas cinco recetas. Por otra parte, los restaurantes se mostrarán por provincia y especialidad principal. Comenzaremos explicando brevemente el funcionamiento para mostrar las recetas por provincias y tipo de receta (vegetariana, vegana, light o celíaca) que se realizará de la misma forma. Primero realizamos la conexión con nuestra base de datos. A continuación montaremos un *select* en base a la consulta que queremos realizar. Para la búsqueda por provincias, montaremos un *select* reuniendo las tablas de receta\_provincia y receta primero y después el resultado lo agruparemos con la tabla provincia. Para los tipos de recetas será más sencillo, tan sólo deberemos seleccionar las recetas que tengan marcadas las casillas de vegetariana, vegana, light o celíacos. En la Figura 8.8 del manual de usuario mostramos la primera receta de la provincia de Huelva. Para los distintos tipos de receta la búsqueda será similar, lo único que cambiará será el título dependiendo de la provincia que se busque o el tipo de receta deseado.

También habrá una búsqueda un poco más avanzada para personas que deseen buscar recetas que contengan algunos ingredientes y que al mismo tiempo no contengan otros. Esta búsqueda aparecerá siempre en la parte superior de la

página y será independiente de la búsqueda que realice el usuario, ya que siempre se mostrará. Para ello hemos utilizado la función *explode* que será la encargada de reconocer los diferentes ingredientes que ha introducido el usuario en su búsqueda. Esta función no tiene en cuenta los espacios en blanco y te va dando ingrediente a ingrediente. Estos ingredientes por separado los iremos concatenando en nuestro *select* para hacer la búsqueda en nuestra base de datos. Lo mismo ocurrirá con los ingredientes que no queramos que aparezcan en nuestra búsqueda pero en nuestro *select* deberemos negarla. El usuario también deberá incluir la provincia, dificultad, si contiene horno, batidora, etcétera. Al final ejecutaremos nuestra consulta con todo concatenado para seleccionar las recetas que contengan y no contengan a la vez los ingredientes que desee el usuario.

Por otro lado el formato utilizado para mostrar el contenido de las recetas será el siguiente:

- 1º. En la parte superior se mostrará el título de la receta.
- 2º. A continuación se mostrarán los ingredientes principales y opcionales que contiene cada receta y se calculará el aporte nutricional de cada receta como se puede observar en la tabla llamada aporte nutricional de la Figura 8.19 de nuestro manual de usuario. Dicho aporte será calculado para cada receta con ingredientes principales e ingredientesopcionales en el caso de que la receta tuviera.
- 3º. A continuación se mostrará el modo de preparación de la receta.
- 4º. Por último se mostrarán las fotos que estén asociadas con dicha receta y su dificultad, tiempo de preparación, número de comensales y utensilios utilizados.

Para cada restaurante, como se puede apreciar en la Figura 8.20 de nuestro manual de usuario el formato con el que se mostrarán los datos será el siguiente:

- 1º. En la parte superior se mostrará el nombre del restaurante.
- 2º. A continuación se mostrará la foto correspondiente a dicho restaurante y sus datos: ciudad, provincia, dirección, teléfono y especialidad, todo ello mediante una lista desordenada (*unordered list*).
- 3º. Por último en el hueco que queda al lado izquierdo de la foto y los datos del restaurante, se mostrará la georeferenciación del restaurante utilizando *google maps*.

Por último cuando se pinche sobre el enlace de alguna de las cinco últimas recetas, el contenido aparecerá en una pestaña distinta. El formato para mostrar las cinco últimas recetas será el mismo que para mostrar cualquier otra receta.

- **Registro usuario:** para registrar a un usuario en nuestra base de datos utilizaremos dos funciones: “*registroUsuario.php*” y “*formularioRegistro.php*”. La primera función será la encargada de procesar los errores, en el caso de que hubieran, e insertar en nuestra base de datos al nuevo usuario. La segunda función será la encargada de pedir los datos necesarios para registrar al nuevo usuario mediante un formulario. Este formulario se podrá observar en la Figura 8.2.

## 6.9. VALIDACIONES JAVASCRIPT Y LIBRERÍA JQUERY

Para realizar las validaciones de los formularios de modificar clave, modificar datos, y el registro de usuario, hemos utilizado JAVASCRIPT. Nos hemos creado una función para validar el formulario, en la cual, nos creamos una variable mediante expresiones regulares para cada elemento que queramos validar (nombre, apellidos, correo, clave, ciudad, país, dirección, teléfono y ocupación). Luego comprobamos que los datos introducidos en nuestro formulario se adaptan a nuestros patrones o variables creadas anteriormente, en caso negativo se muestra en pantalla un mensaje diciendo que los datos introducidos no son válidos. Hasta que no sean todos los campos escritos correctamente no dejará modificar los datos del usuario o modificar su clave o registrar a un nuevo usuario. Para que el formulario que queramos pueda validar los datos pondremos el siguiente código: *onsubmit="return validacion(this)"*.

Los patrones con los que compararemos serán los que se pueden observar en la Figura 6.1. Los patrones para el nombre, para la ciudad, para el país y para la ocupación podrán contener letras mayúsculas y minúsculas y espacios en blanco. Por su parte el teléfono deberá estar formado por nueve dígitos. La dirección será igual que el nombre, pero podrá contener también dígitos y los caracteres especiales ° y ª. La clave por su parte deberá estar formada por entre seis y quince caracteres de longitud. Estos caracteres deberán contener al menos un dígito, una letra minúscula y una letra mayúscula. Por último el correo deberá estar formado por al menos dos letras mayúsculas, minúsculas o dígitos antes de la arroba y

detrás podrá llevar como mínimo dos letras mayúsculas, minúsculas, dígitos o guión bajo o guión normal, seguido de un punto y de dos a cuatro letras mayúsculas o minúsculas, seguido de un punto y de dos a cuatro letras mayúsculas o minúsculas.

```
var er_nombre = /^[a-z][A-Z]|\á|é|i|ó|ú|ñ|ü|\s)+$/  
var er_correo=^[a-zA-Z0-9_-]{2,}@[a-zA-Z0-9_-]{2,}\\. [a-zA-Z]{2,4}(\.[a-zA-Z]{2,4})?$/  
var er_clave=/^(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{6,15}$/  
var er_ciudad = /^[a-z][A-Z]|\á|é|i|ó|ú|ñ|ü|\s)+$/  
var er_pais = /^[a-z][A-Z]|\á|é|i|ó|ú|ñ|ü|\s)+$/  
var er_direccion = /^[a-z][A-Z][0-9]|\á|é|i|ó|ú|ñ|ü|^|\s)+$/  
var er_telefono =^{\d{9}}$/;  
var er_ocupacion = /^[a-z][A-Z]|\á|é|i|ó|ú|ñ|ü|\s)+$/
```

**Figura 6.1. Patrones de validación.**

Para añadir un nuevo ingrediente debemos crearnos una nueva fila vacía idéntica a la que aparecía antes de llenar los datos del ingrediente. La creación de esta fila la conseguiremos utilizando un *script* de JAVASCRIPT cuyo funcionamiento describiremos a continuación. Si el objeto está lleno, es decir, se han introducido la cantidad, la medida y el nombre del ingrediente, al pulsar sobre el botón de añadir ingrediente creamos un nuevo objeto vacío idéntico al anterior, en el que podremos almacenar un nuevo ingrediente. Así procederemos con todos los ingredientes que queramos introducir. Para cada campo anterior comprobaremos la existencia de datos. En el caso de estar llenos los tres campos al pulsar el botón de añadir ingrediente se añadirá un nuevo ingrediente. Este funcionamiento será exactamente igual tanto para los ingredientes principales e ingredientes opcionales.

También para enviar una foto adjunta con una receta debemos comprobar una serie de aspectos antes de subirla a nuestra base de datos, también los comprobaremos con *javascript*. Primero especificamos la carpeta a la cual queremos que se suba nuestra foto. Los errores pueden ser producidos por varios motivos: que se introduzca una imagen distinta de *jpg*, *gif* o *pjpeg* o que la foto sea mayor de 300kb, en cuyo caso se mostrará el siguiente mensaje de error: "compruebe que la foto sea una imagen en formato *gif* o *jpg* y de tamaño inferior a 300Kb.", que el título de la foto a subir no esté formada por letras mayúsculas o minúsculas, dígitos, guiones bajos o guiones, en cuyo caso se mostrará el siguiente mensaje de error: "el nombre de la foto contiene caracteres no válidos.", o que el nombre de la foto ya exista, en cuyo caso se pedirá que se le cambie el nombre a la foto antes de subirla de nuevo.

Por último la función encargada de producir el efecto de encendido de la hornilla cuando se pone el cursor encima de algunos de los botones del menú es la que se puede

apreciar en la Figura 6.2. Al posicionar el ratón sobre el botón este cambia progresivamente a la imagen que tenemos superpuesta debajo de la inicial.

```
<script type="text/javascript" src="js/jquery.min.js"></script>
<script type='text/javascript'>
$(document).ready(function(){
  $("img.a").hover(
    function() {
      $(this).stop().animate({"opacity": "0"}, "normal");
    },
    function() {
      $(this).stop().animate({"opacity": "1"}, "normal");
    });
});
</script>
```

Figura 6.2. Imagen código función JQUERY.

Para paginar los resultados he utilizado una función JAVASCRIPT llamada “*PHPPaging.php*” que es libre y totalmente autoconfigurable. Para las búsquedas de recetas y de términos se mostrarán los resultados paginados de manera que se muestren cuatro por página. Al posicionarnos sobre el enlace de los resultados se mostrará un mensaje con los resultados que se están mostrando y el total que hay. Para que los resultados se muestren paginados hay que realizar cuatro pasos:

1º. Introducir la función dentro del código que queremos paginar. Para ello incluimos la siguiente orden: *require\_once("PHPPaging.php")*.

2º. Creamos un nuevo objeto *PHPPaging*: *\$paging=new PHPPaging*.

3º. Introducimos la sentencia SQL que queramos paginar, *\$paging->agregarConsulta(\$sql)*, y ejecutamos: *\$paging->ejecutar()*.

4º. Recogemos los resultados mediante la orden *\$fila=\$paging->fetchResultado()*.

5º. Por último mostraremos la barra de paginación de la siguiente forma: *\$paging->fetchNavegacion()*.

Por último para mostrar un mensaje de confirmación cuando queremos eliminar una receta, ingrediente o término desde el menú del usuario hay que poner el siguiente código dentro del formulario: *onclick="return confirm('\u00BF Desea realmente eliminar la receta?');"*.

# CAPITULO 7

## HOJA DE ESTILOS EN CASCADA: CSS

En este capítulo describiremos brevemente como surgieron las CSS (*Cascading Style Sheets*) y algunas de las ventajas más importantes que suponen su uso. Por último explicaremos de manera exhaustiva la hoja de estilos de nuestra página web.

### 7.1. INTRODUCCIÓN

Las hojas de estilo en cascada o *Cascading Style Sheets* (CSS) nacieron ante las limitaciones que presenta HTML a la hora de aplicarle forma a un documento. Esto se debe a que HTML fue creado para uso científico que difiere bastante del uso actual.

Para solucionar estos problemas los diseñadores inventaron nuevas técnicas tales como la utilización de imágenes transparentes para ajustarlas, utilización de etiquetas que no son estándares de HTML, etc. Estas técnicas han causado infinidad de problemas en las páginas web a la hora de su visualización en las diferentes plataformas.

Además de estas dificultades los diseñadores se han visto frustrados por la dificultad que se encontraban a la hora de maquetar las páginas web [26].

Otro antecedente que ha hecho necesario el desarrollo de las CSS es que las páginas web tienen mezclado junto a su código HTML el contenido del documento con las etiquetas necesarias para darle formato. Esto tiene como principal inconveniente que la lectura de los códigos HTML se hace pesada y muy difícil a la hora de buscar o depurar errores en las páginas. En definitiva, el hecho de tener texto mezclado con etiquetas incrustadas para darle formato supone una gran desventaja, ya que, se degrada su utilidad.

El modo de funcionamiento de las CSS consiste en definir mediante una sintaxis especial, la forma de presentación que se aplicará a:

- **Una web entera:** se puede definir los estilos de toda una web de una sola vez.
- **Un documento HTML:** se puede definir el estilo de un pequeño trozo de código o a toda la página HTML.
- **Una etiqueta en concreto:** se puede llegar a definir varios estilos diferentes para una sola etiqueta. Esto es muy importante, ya que, otorga una gran potencia a la hora de programar nuestra página. Por ejemplo, podríamos definir diferentes tipos de párrafos: que tuvieran diferentes colores de letra o incluso que tuvieran márgenes distintos.

La potencia de esta tecnología salta a la vista. Entre las principales ventajas que supone el uso de las CSS se encuentran las siguientes:

- **Control del diseño:** las CSS son muy útiles para separar el contenido del diseño, siendo esto muy útil cuando se quiere cambiar el aspecto del diseño de una página web, ya que, sin hojas de estilo se tendría que cambiar obligatoriamente página a página. Sin embargo, utilizando hojas de estilo, se puede cambiar el aspecto de la página web modificando únicamente la hoja de estilos. Con esto logramos una mayor uniformidad en el diseño y ahorro de tiempo.

- **Redefinición de etiquetas:** se puede redefinir el aspecto de visualización de una etiqueta. Con la hoja de estilos se puede hacer, por ejemplo, que todos los encabezamientos `<h2>` de nuestra página tengan un color o un tamaño determinado. Si no usáramos hojas de estilo se debería redefinir cada vez que usase dicha etiqueta.
- **Uso de etiquetas para su misión:** gracias a las hojas de estilo, podremos usar cada etiqueta para lo que realmente se creó. Por ejemplo, la etiqueta `<font>` se utilizará sólo en la hoja de estilos, mientras que la etiqueta `<table>` se usará para presentar datos tabulados.
- **Personalización:** nos permite definir aspectos concretos del documento, facilitando su diseño.
- **Maquetación:** se pueden colocar los elementos con una mayor precisión utilizando pocas líneas de código.
- **Respeto de estándares:** CSS será respetuoso con los estándares, lo que implicará que sea más funcional en los diferentes navegadores y tendrá un código más sencillo.
- **Tamaño:** se reducirá considerablemente el peso de los ficheros, disminuyendo el ancho de banda consumido y acercando el contenido de la página web a los usuarios de manera más rápida.
- **Usabilidad:** las CSS permiten a los usuarios definir hojas de estilo que se adapten mejor a sus características.
- **Visibilidad:** los robots de motores de búsqueda usan navegadores en modo texto como Lynx. Definiendo el diseño en la hoja de estilo le ahorras trabajo al buscador, ya que, podrá indexar tus documentos con mayor rapidez. Además reduce la utilización de etiquetas [27].

Las CSS son bastante recientes, por lo que no todos los navegadores las soportan. En concreto sólo los navegadores de Netscape a partir de versión 4 en adelante y de Microsoft a partir de la versión 3 son capaces de comprender la sintaxis de las CSS. Además cabe destacar que no todos los navegadores implementan las mismas funciones de hojas de estilos [28].

## 7.2. EXPLICACIÓN DEL CÓDIGO UTILIZADO

A continuación pasaremos a detallar los diferentes estilos aplicados en nuestra página web, de manera que cualquier usuario no experto, pudiera realizar los cambios que estimara oportunos. Para ello utilizaremos el juego de caracteres UTF8.

```
1 @charset "utf-8";
2 body {
3     background : #C0D9D9 url(..../imagenes/mantel.jpg) repeat;
4     font : 8pt Verdana, Geneva, Arial, Helvetica, sans-serif;
5     color : #666666;
6     margin : 20px 0px 20px 0px;
7     text-align: center;
8 }
9 #cabecera{
10     height : 339px;
11     width: 900px;
12     margin-left:auto;
13     margin-right:auto;
14     background-image: url('..../imagenes/banner.png');
15 }
```

Figura 7.1. Código CSS del cuerpo y cabecera.

Empezaremos por el cuerpo y la cabecera. Como podemos observar en la Figura 7.1, el cuerpo de nuestra página web tendrá como color de fondo o *background* un color azul clarito. Los diferentes colores se formarán mezclando rojo, verde y azul (RGB). También llevará asociado una imagen de fondo que será un mantel andaluz repitiéndose éste tanto en el eje x como en el eje y. Con el atributo *font* elegiremos el tamaño de letra utilizado, en este caso 8pt (puntos) y el tipo de letra a utilizar. Para el tipo de letra el orden en el que aparece es importante ya que el navegador buscará el primer tipo de letra y en caso de no encontrarlo pasará al siguiente hasta encontrar alguno. Con el atributo *color* elegiremos el color de letra

para el cuerpo de nuestra página web, en esta ocasión he elegido un gris clarito. Con el atributo *margin*, como podemos apreciar en la línea 7 de la Figura 7.1, asignamos un margen hacia arriba, hacia la derecha, hacia abajo y hacia centro respectivamente. Estos márgenes se pueden poner de manera individual utilizando *margin-top*, *margin-right*, *margin-bottom* y *margin-left* respectivamente. El texto quedará alineado en el centro utilizando el atributo *text-align*.

Por su parte, como podemos apreciar en la Figura 7.1 para la cabecera de nuestra página web tendrá una anchura de 900 pixels y una altura de 339 pixels. Los márgenes a la izquierda y a la derecha serán puestos en auto para que la cabecera se quede en el centro. Como fondo tendrá una imagen png llamada banner que tendrá los bordes redondeados.

```

16  #login{
17      width: 170px;
18      height: 90px;
19      position: relative;
20      top: 17px;
21      left: 703px;
22  }
23  #login p{
24      color: #390;
25      font-family: Arial black;
26      font-size: 14px;
27      margin: 0px;
28  }
29  #login a{
30      color:red;
31  }
32  #login a:hover{
33      color:#f90;
34  }

```

**Figura 7.2. Código CSS login.**

Para el login utilizaremos el atributo *position* que aparece en la línea 19 de la Figura 7.2 para ponerlo en la posición que queramos dentro del espacio reservado anteriormente para la cabecera. Para ello le indicaremos el margen hacia arriba y el margen hacia la izquierda que queramos. Estos márgenes se pueden apreciar en las líneas 20 y 21 de la Figura 7.2 respectivamente. En esta ocasión el espacio reservado dentro de la cabecera para el login será de 170 pixels de anchura y 90 pixels de altura.

Por otro lado para los párrafos que haya dentro del *login* tendremos los siguientes atributos: el color de la letra será color verde pistacho, el tipo de letra será *Arial-black*, el tamaño de letra será de

14 pixels y no tendrá margen. Todos estos atributos se pueden observar en las líneas 24, 25, 26 y 27 de la Figura 7.2.

Para los *links* o enlaces que aparezcan en el *login* utilizaremos el rojo, para destacarlos, como se indica en la línea 30 de la Figura 7.2. Cuando nos ponemos encima del enlace se cambia a un color naranja clarito, este atributo se puede observar en la línea 33 de la Figura 7.2.

El código para el menú lo explicaré ayudándome de la Figura 7.3. Para el menú utilizaremos un color de fondo marrón oscuro y su tamaño será de 900 pixels de anchura y 98 pixels de altura.

```
35 #menu{  
36     background-color:#660000;  
37     width:900px;  
38     height:98px;  
39 }  
40 div.fadehover {  
41     position: relative;  
42 }  
43 img.a {  
44     position: absolute;  
45     left: 0;  
46     top: 0;  
47     z-index: 10;  
48 }  
49 img.b {  
50     position: absolute;  
51     left: 0;  
52     top: 0;  
53 }
```

Figura 7.3. Código CSS menú.

Para todos los *div* de la clase *fadehover* utilizaremos una posicion relativia. Tendremos una imágenes de tipo a y otra de tipo b que estarán superpuestas una encima de otra. Tanto las imágenes de tipo a como las de tipo b tendrán posicionamiento absoluto, de 0 pixels hacia arriba y de 0 pixels hacia la izquierda. Al tener un posicionamiento absoluto en ambas imágenes el atributo *z-index* nos indicará que imagen estará encima de la otra. En nuestro caso estará la imagen de tipo a. Al posicionarnos sobre una imagen de la clase a, tendremos un script de JAVASCRIPT que utilizará JQUERY. Esta librería nos ayudará a crear el efecto de transición de una imagen de tipo a a otra de tipo b. Esta librería cambia la opacidad de la imagen de tipo a, que al principio es 1, a 0, para que pueda verse la imagen de tipo b.

El menú del administrador tendrá un color de fondo blanco y tendrá una anchura de 555 pixels y una altura de 98 pixels. También flotará a la izquierda como se puede apreciar en las líneas 55,56 y 57 de la Figura 7.4 respectivamente.

```
54  #menu_gest{  
55      background-color:#fff;  
56      width:555px;height:98px;  
57      float:right;  
58  }
```

**Figura 7.4. Menú administrador.**

Todos los párrafos de nuestra página web tendrán unos márgenes a la izquierda y a la derecha de 10 pixels y un tamaño de letra de 15 pixels como se puede apreciar en la Figura 7.5.

```
59  #estiloglosario, #mensajeBienvenida, #nuestraweb,#estiloayuda,#estilonuevouser{  
60      width: 718px;  
61      float: right;  
62      background-color:#fff;  
63      text-align: left;  
64  }  
65  #estiloglosario p,#mensajeBienvenida p, #nuestraweb p,#estiloayuda p{  
66      margin-left: 10px;  
67      margin-right: 10px;  
68  }  
69  p{  
70      margin-left: 10px;  
71      margin-right: 10px;  
72      font-size:15px;  
73  }
```

**Figura 7.5. Código CSS glosario, bienvenida, etc.**

Como podemos observar en la Figura 7.5 el estilo del glosario, del mensaje de bienvenida, de la página nuestra web, el estilo de la ayuda y el estilo de los nuevos usuarios tendrán un tamaño de 718 pixels, flotarán a la izquierda y, tendrán un color de fondo blanco, el texto estará alineado a la izquierda.

Por otra parte todos los párrafos incluidos en el glosario, en el mensaje de bienvenida, en nuestra web y en la ayuda tendrán un margen a izquierda y derecha de 10 pixels, como se puede apreciar en las líneas 66 y 67 de la Figura 7.5.

Para cada palabra del glosario le pondremos un borde de color negro, sólido de 1 pixel de tamaño. El margen entre palabra y palabra será de 5 pixels y el margen a la izquierda y derecha será de 1 pixel como se puede apreciar en las líneas 75, 76, 77 y 78 respectivamente de la Figura 7.6.



```
74 #palabraglosario{  
75     border:#000 1px solid;  
76     margin-bottom:5px;  
77     margin-left:1px;  
78     margin-right:1px;  
79 }  
80 #nuestraweb h3, #estiloayuda h3,.saludo{  
81     color:#000;  
82 }  
83 #pie{  
84     clear : both;  
85     color : #cccccc;  
86     text-align : right;  
87     margin : 10px 10px 0px 10px;  
88     padding-bottom:10px;  
89 }
```

**Figura 7.6. Código CSS glosario, h3 y pie.**

Para los encabezados de tipo *h3* de nuestra web, ayuda el saludo de bienvenida tendremos un color de letra negro como podemos apreciar en la línea 81 de la Figura 7.6.

Con lo que respecta al pie de nuestra página web en la línea 84 de la Figura 7.6 indicamos que no se ponga nada a la izquierda y a la derecha de nuestro pie. El color de la letra será un tono grisáceo y se alinearán a la derecha. También tendrá unos márgenes hacia arriba, izquierda, abajo y derecha de 10, 10, 0 y 10 pixels respectivamente. En la línea 88 indicamos que dentro del contenedor del pie cree un margen de 10 pixels hacia abajo.

Las letras de los *checkbox* que se muestran con cada receta tendrán un tamaño de letra de 11 pixels.

```
90  #tablalogicos{  
91      font-size:11px;  
92  }  
93  #menuusuario{  
94      margin-left:25px;  
95      text-align:left;  
96  }  
97  #dologin{  
98      font-size:10px;  
99      text-align:left;  
100 }
```

Figura 7.7. Código CSS *login*, menú usuario y *checkbox*.

Por otra parte, el menú de usuario tendrá un margen a la izquierda de 25 pixels y el texto estará alineado a la izquierda como podemos apreciar en las líneas 94 y 95 de la Figura 7.7.

Para le *login*, por su parte tendrá un tamaño de fuente de 10 pixels y el texto estará alineado a la izquierda como se observar en las líneas 98 y 99 de la Figura 7.7.

Para todos los *span* del acceso de usuarios utilizaré un tamaño de letra de 11 pixels.

```
101 #acceso span{  
102     font-size:11px;  
103  }  
104 #saludoAdmin{  
105     float:right;  
106     width:720px;  
107  }  
108 #lista_res{  
109     width: 718px;  
110     float: right;  
111     background-color:#fff;  
112  }
```

Figura 7.8. Código CSS acceso usuarios, listas y saludo administrador.

Para el saludo del administrador he optado por un tamaño de 720 pixels y que flote a la derecha como podemos apreciar en las líneas 106 y 105 de la Figura 7.8 respectivamente.

Por su parte para el listado de restaurantes la anchura será de 718 pixels y flotará a la derecha. El color de fondo utilizado será el blanco como se puede observar en las líneas 109, 110 y 111 de la Figura 7.8.

Para el estilo de cada receta he optado por ponerle un margen hacia arriba y hacia abajo de 10 pixels, un borde negro sólido de 1 pixel, un tamaño de 714 pixels y que flote hacia la derecha. Como color de fondo he optado por el blanco. Todo esto se puede observar en las líneas 114, 115, 116, 117, 118 y 119 respectivamente de la Figura 7.9.



```
113 #estiloreceta{  
114     margin-top:10px;  
115     margin-bottom:10px;  
116     border:#000 solid 1px;  
117     width:714px;  
118     float:center;  
119     background-color:#fff;  
120 }  
121 #estilo_rest{  
122     width: 718px;  
123     float: right;  
124     background-color:#fff;  
125 }  
126 #estilo_rest ul{  
127     text-align:left;  
128 }
```

Figura 7.9. Código CSS recetas y restaurantes.

Para los restaurantes he optado por un tamaño de 718 pixels y que floten hacia la derecha. El color de fondo elegido será el blanco. Por otra parte el texto estará de la lista desordenada que haya en los restaurantes estará alineada a la izquierda como se puede observar en las líneas 122, 123, 124 y 127 de la Figura 7.9.

Para las descripciones de los restaurantes aparece un nuevo atributo que se puede ver en la línea 130 de la Figura 7.10. Con este atributo prohibimos que nos salgamos del contenedor asignado a cada restaurante. Tiene un tamaño del 100 % es decir llena todo el contenedor asignado y flota a la izquierda.

```

129  #descrip_rest{
130      overflow: hidden;
131      width: 100%;
132      float: left;
133  }
134  #titulo_principal{
135      background-color:#FFF;
136  }
137  #ingredientes{
138      float: left;
139      margin-left: 33px;
140      text-align: left;
141      margin-top: 6px;
142      width: 262px;
143  }

```

**Figura 7.10. Código CSS restaurantes y recetas.**

Para cada foto que contenga una receta flotarán a la izquierda, su texto estará centrado, se prohíbe poner cosas a la derecha y a la izquierda de cada foto y tendrá un margen hacia debajo de 10 pixels como se puede apreciar en las líneas 145, 146, 147 y 148 de la Figura 7.11.

```

144  #foto_receta{
145      float: left;
146      text-align: center;
147      clear: both;
148      margin-bottom: 10px;
149  }
150  #foto_rest{
151      float: left;
152      text-align:left;
153      margin-right: 10px;
154  }

```

**Figura 7.11.Código CSS fotos.**

Por otra parte cada foto de un restaurante flotará a la izquierda, su texto estará centrado y tendrá un margen a la derecha de 10 pixels como se puede observar en las líneas 151, 152 y 153 de la Figura 7.11.

Para cuando los enlaces de tipo A han sido visitados, para seleccionar el color del enlace activo, para la selección del elemento que tiene el foco y para seleccionar el enlace a

otras páginas no visitadas, usaremos un color amarillo claro, como podemos apreciar en la Figura 7.12. El color de la fuente será el negro.

```
155 A, A:ACTIVE, A:FOCUS, A:LINK, A:VISITED {  
156     color : #FFFF33;  
157     font-weight: bold;  
158 }  
159 A:HOVER {  
160     color: #f90;  
161 }
```

**Figura 7.12. Código CSS enlaces.**

Por otro lado cuando se posicione el cursor sobre los enlaces, el color elegido será un color anaranjado.

Para los INPUTS de todos los formularios el tamaño de la letra será de 8 puntos como se puede apreciar en la línea 163 de la Figura 7.13.

```
162 INPUT {  
163     font-size : 8pt;  
164 }  
165 H1{  
166     font-size: 17px;  
167     text-align:center;  
168     color:#000;  
169     background-color:#FFF;  
170 }  
171 H2{  
172     text-align:left;  
173     margin-left:10px;  
174 }  
175 H3{  
176     color: #fff;  
177     font-size:13px;  
178     text-align:center;  
179 }
```

**Figura 7.13. Código CSS formulario y encabezados.**

Para los encabezados de tipo H1 el tamaño de letra será de 17 pixels, su texto estará centrado, su color de letra será negro y tendrá un color de fondo blanco.

Por otro lado para los encabezados de tipo H2 su texto estará alineado a la izquierda y tendrá un margen a la izquierda de 10 pixels.

Por último para los encabezados de tipo H3 el color de la letra será el negro, el tamaño de la fuente será de 13 pixels y su texto estará centrado.

Todos estos atributos se pueden ver en las líneas 166, 167, 168, 169, 172, 173, 176, 177 y 178 respectivamente de la Figura 7.13.

Los bordes de cada receta por su parte tendrán un borde sólido de 2 pixels de color grisáceo. Su texto estará alineado a la izquierda, su tamaño será de 900 pixels y su margen se ajustará de manera automática. Todos estos atributos se pueden ver en las líneas 181, 182, 183 y 184 de la Figura 7.14.

```
180 #borde{  
181     border: 2px solid #cccccc;  
182     text-align: left;  
183     width: 900px;  
184     margin: auto;  
185 }  
186 #contenedor{  
187     text-align: center;  
188     width: 900px;  
189     margin: auto;  
190     background-color : #fff;  
191 }
```

**Figura 7.14. Código CSS bordes y contenedor.**

Por lo que respecta al contenedor de la página web su texto estará centrado, tendrá un tamaño de 900 pixels, su margen se ajustará automáticamente y su color de fondo será blanco como podemos observar en las líneas 187, 188, 189 y 190 de la Figura 7.14.

Para la columna izquierda de nuestra página web he optado por darle una anchura de 180 pixels, que flote a la derecha y porque tenga un margen hacia debajo de 20 pixels dentro de la columna izquierda. Su color de fondo será el marrón. Todos estos atributos se pueden observar en la Figura 7.14 en las líneas que van desde la 187 hasta la 190.

Por lo que respecta a los títulos de cada receta, éstos tendrán el título centrado, tendrá un tamaño de letra de 14 pixeles y su color será negro. Su color de fondo será un amarillo muy clarito y tendrá un borde sólido arriba de 1pixel de color grisáceo y un borde abajo de 1

pixel sólido negro como se puede apreciar en la Figura 7.15 en las líneas que van desde la 199 hasta la 204.

```
192 #col_izq{  
193     width:180px;  
194     float:left;  
195     padding-top: 20px;  
196     background-color: #600;  
197 }  
198 #tituloreceta{  
199     text-align:center;  
200     font-size:14px;  
201     color:#000;  
202     background-color:#F5F4C3;  
203     border-top : 1px solid #cccccc;  
204     border-bottom : 1px solid #000;  
205 }
```

**Figura 7.15. Código CSS columna izquierda y título receta.**

Por su parte para los campos de los formularios su texto estará centrado y para el cuerpo de nuestra página web, como podemos observar en la Figura 7.16 tendrá una anchura de 620 pixels, flotará a la izquierda, tendrá un margen a la izquierda de 8 pixels y unos márgenes dentro del contenedor de 12, 0, 10 y 0 pixels hacia arriba, izquierda, abajo y derecha respectivamente.

```
206 #campoform{  
207     text-align:center;  
208 }  
209 #cuerpo{  
210     float:left;  
211     width:620px;  
212     margin-left: 8px;  
213     padding: 12px 0px 10px 0px;  
214 }  
215 #navabajo{  
216     font-weight : bold;  
217 }  
218 #lateral{  
219     width: 200px;  
220     float:right;  
221     background-color: #EBF2FE;  
222     border-bottom : 1px solid #cccccc;  
223     border-left : 1px solid #cccccc;  
224 }
```

**Figura 7.16. Código CSS cuerpo y lateral.**

Para el navegador de abajo tendremos una fuente de color negrita, mientras que para el lateral de nuestra página web tendrá un tamaño de 200 pixels de anchura, flotará a la derecha y su color de fondo será azul clarito. Tendrá un borde abajo y arriba de 1 pixel sólido de color grisáceo. Todos estos atributos se pueden ver con más detalle en las líneas que van de la 219 a la 223 de la Figura 7.16.

```
225  #campotexto{  
226      float: left;  
227  }  
228  #campotexto input{  
229      width:100px;  
230  }  
231  #fbuscar form{  
232      margin-bottom : 0px;  
233      margin-top : 0px;  
234  }
```

**Figura 7.17. Campos de texto.**

Para los buscadores de la columna izquierda tendrán una serie de características: tendrá un margen dentro de la columna izquierda de 3 pixels hacia arriba y de 106 pixels hacia abajo, como se puede observar en la Figura 7.18.

```
235  #botonbuscar {  
236      padding-top : 3px;  
237      padding-left: 106px;  
238  }  
239  #botonbuscar input{  
240      border : 0px none;  
241  }
```

**Figura 7.18. Código CSS botones.**

Por lo que respecta a los INPUTS del buscador de la columna izquierda no tendrán ningún borde como se puede ver en la línea 240 de la Figura 7.18.

Para otras listas desordenadas que aparezcan en nuestra página web tendrán un margen de 5 y 10 pixels hacia arriba y derecha respectivamente. No tendrán ningún elemento de

decoración y tendrán un margen dentro de su contenedor principal de 4 pixels como se puede apreciar en las líneas que van desde la 243 a la 245 de la Figura 7.19.

```
242 #otras ul{  
243     margin : 5px 10px 0px 0px;  
244     list-style: none;  
245     padding: 0px 0px 0px 4px;  
246 }  
247 #otras li{  
248     padding-left: 14px;  
249     background: transparent url("images/bullet.gif") 0 2px no-repeat;  
250     margin-bottom: 10px;  
251 }
```

Figura 7.19. Código CSS listas desordenadas.

Por otro lado, para cada elemento de la lista, como podemos apreciar en la Figura 7.19 tendremos un margen dentro del contenedor principal de 14 pixels hacia la izquierda. Un color de fondo transparente con una imagen de un disco pasada a través de url de 2 pixels y que no se repite.

Para el cuerpo del lateral izquierdo de nuestra página web tendrá unos márgenes dentro de su contenedor principal de 5, 4, 13 y 10 pixels hacia arriba, izquierda, abajo y derecha respectivamente como se puede apreciar en la línea 253 de la Figura 7.20.

```
252 .cuerpolateral{  
253     padding: 5px 4px 13px 10px;  
254 }  
255 .titlat{  
256     border:#cccccc solid 1px;  
257     background-color:#68729E;  
258     color:#ffffff;  
259     font-size:8pt;  
260     text-transform : uppercase;  
261     padding: 7px 3px 7px 8px;  
262     margin: 0px 0px 8px 0px;  
263     font-weight : normal;  
264     letter-spacing : 2px;  
265 }
```

Figura 7.20. Código CSS lateral y títulos.

Para el buscador del lateral izquierdo, tendrá un borde sólido de 1 pixel de color grisáceo, el color de fondo será un morado claro, el color de la letra será negro y su tamaño será de 8 puntos. El texto será transformado en mayúscula, y tendrá unos márgenes dentro de su contenedor principal de 7, 3, 7 y 8 pixels hacia arriba, izquierda, abajo y derecha respectivamente. Tendrá un margen hacia debajo de 8 pixels. El tamaño de la letra será normal y el espaciado entre letras será de 2 pixels. Todo ello se puede apreciar en la Figura 7.20.

Para cada receta que se tenga que validar su aspecto será el siguiente: tendrá un borde negro de 1 pixel sólido. También tendrá unos márgenes hacia abajo, arriba, izquierda y derecha de 5, 5, 1 y 1 pixels respectivamente. Todos estos atributos se pueden observar en las líneas que van desde la 267 a la 271 de la Figura 7.21.

```

266 #valida_receta{
267     border: #000 solid 1px;
268     margin-bottom:5px;
269     margin-top:5px;
270     margin-left:1px;
271     margin-right:1px;
272 }
```

**Figura 7.21. Código CSS validar receta.**

Para cuando los enlaces del navegador han sido visitados, para seleccionar el color del enlace activo, para la selección del elemento que tiene el foco y para seleccionar el enlace a otras páginas no visitadas, usaremos un color amarillo claro, como podemos apreciar en la Figura 7.22.

```

273 A.enlacenav, A.enlacenav:VISITED, A.enlacenav:ACTIVE, A.enlacenav:FOCUS, A.enlacenav:LINK{
274     color: #ffff66;
275 }
276 A.enlacenav:HOVER{
277     color: #3F7DE3;
278 }
```

**Figura 7.22. Código CSS de enlaces.**

Por otro lado, cuando ponemos el ratón sobre los enlaces, el color elegido será el azul como podemos apreciar en la línea 277 de la Figura 7.22.

Para las casillas de verificación no se podrá poner nada ni al lado derecho ni al lado izquierdo, mientras que para los errores que se produzcan en los formularios, se mostrará un mensaje en color rojo, como podemos apreciar en las líneas 280 y 283 de la Figura 7.23.

```
279 .radio{  
280     clear : both;  
281 }  
282 .formerror{  
283     color:red;  
284 }  
285 .formularioValidareceta{  
286     border:#000 solid 1px;  
287 }
```

Figura 7.23. Código CSS errores y recetas.

Para finalizar para cada receta tendremos un borde sólido de un 1 pixel, de color negro como podemos apreciar en la Figura 7.23.

# CAPÍTULO 8

# MANUAL DE USUARIO

Al visitar nuestra página web lo primero que se encontrarán será la página de inicio. Ella estará compuesta por los siguientes componentes: una cabecera, un menú horizontal, una columna izquierda, un cuerpo y un pie de página. Todos estos componentes serán comunes a todas las páginas de las que constará nuestra web, lo único que cambiará será el contenido del cuerpo de la página dependiendo del botón o enlace que pinchemos, como se puede observar en la Figura 8.1. Dependiendo de los diferentes tipos de usuarios nuestra web cambiará su aspecto

y contenido. A continuación pasaremos a describir los diferentes componentes que forman parte de esta estructura básica y la visión para los diferentes tipos de usuarios.



**Figura 8.1. Imagen página inicio.**

## 8.1. CABECERA

Situada en la parte superior (Figura 8.1) está formada por nuestro logotipo, por un enlace para que se registren nuevos usuarios, y por un sistema de autenticación de usuarios que ya existan.

### **8.1.1. REGISTRO DE USUARIOS**

Para que se registre un nuevo usuario será necesario llenar un formulario con sus datos personales. Los campos que se deben llenar obligatoriamente serán los siguientes: nombre, apellidos, correo, clave, ciudad, país, dirección, teléfono y ocupación como se puede observar en la Figura 8.2. El campo chef servirá a los usuarios que lo hayan seleccionado para validar sus recetas automáticamente sin supervisión previa del administrador. Por último el campo observaciones servirá para que el usuario haga cualquier otra observación que crea conveniente.

## Muchas gracias por tu interés en registrarte con nosotros!

Nombre:\*

Apellidos:\*

Correo:\*

Clave:\*

Ciudad:\*

País:\*

Dirección:\*

Teléfono:\*

Ocupación:\*

Chef:

A standard rich text editor toolbar with various formatting options: bold (B), italic (I), underline (U), alignment buttons (left, center, right, justify), font size dropdown, font family dropdown, font format dropdown, and a set of decorative icons for lists, tables, and other document-related functions.

Observaciones:

ENVIAR

**Figura 8.2. Imagen registro de usuario.**

### **8.1.2. AUTENTICACIÓN DE USUARIOS**

Los usuarios registrados con éxito anteriormente podrán iniciar sesión introduciendo su correo y clave. En caso de no coincidir dicho correo y clave con el

almacenado en la base de datos se informará del error al usuario mediante un mensaje de color rojo en el cuerpo de nuestra página web. En caso de que los datos introducidos fueran correctos se mostrará un mensaje de bienvenida con el nombre del usuario y un menú con diferentes opciones como el que aparece en la Figura 8.3.



**Figura 8.3. Cabecera con un submenú para un usuario registrado.**

#### **8.1.1.1. MODIFICAR CLAVE**

Desde este enlace el usuario podrá modificar la clave en cualquier momento. Los datos que hay que rellenar para cambiar la clave satisfactoriamente los podemos observar en la Figura 8.4.

The form is titled "Cambiar clave:". It contains four input fields: "Correo:" with the value "cervolla@hotmail.com", "Clave antigua:" (empty), "Clave nueva:" (empty), and "Repita clave nueva" (empty). Below the fields is a "Cambiar clave" button.

**Figura 8.4. Imagen cambiar clave.**

#### **8.1.1.2. MODIFICAR DATOS**

Desde aquí un usuario registrado podrá modificar sus datos personales en cualquier momento, accediendo a un formulario como el de la Figura 8.2.

### 8.1.1.3. ELIMINAR USUARIO

Con esta opción accedemos al formulario de la Figura 8.5, con el que un usuario podrá darse de baja cuando lo desee, introduciendo su correo y clave.



El formulario se titula "Borrar usuario:". Contiene dos campos de texto: "Correo:" y "Clave:", ambos con cuadros vacíos. A la derecha de los cuadros hay un botón rectangular con el texto "DARSE DE BAJA".

**Figura 8.5. Imagen borrar usuario.**

### 8.1.1.4. CERRAR SESIÓN

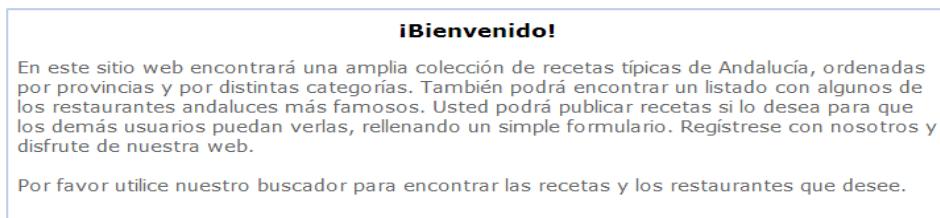
Este enlace será el encargado de cerrar la sesión cuando el usuario pulse sobre él volviendo a la página principal.

## 8.2. MENÚ HORIZONTAL

Este menú estará formado por seis botones para los usuarios no registrados y registrados que no sean administradores. Para los administradores de la página web dicho menú estará compuesto por un botón más llamado “ÁREA GESTIÓN”. A continuación describiremos brevemente el funcionamiento de cada uno de estos botones.

### 8.2.1. BOTÓN INICIO

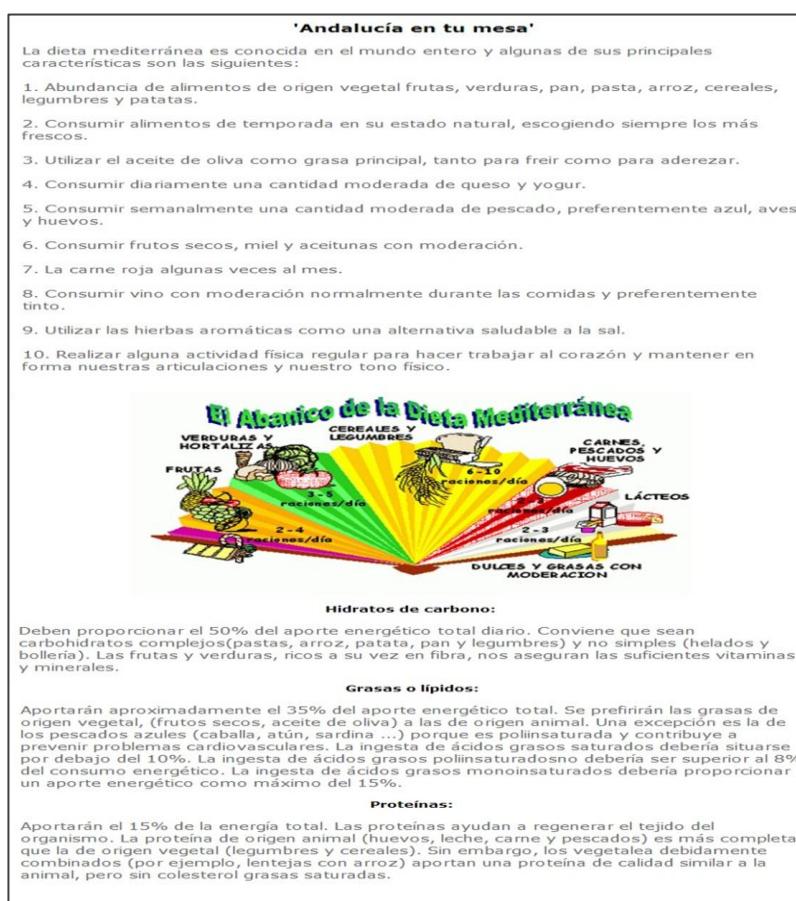
Al pinchar sobre este botón se mostrará en el cuerpo de nuestra web un mensaje de bienvenida para los visitantes de nuestra página web, como se puede apreciar en la Figura 8.6. También será la página de inicio al cargar nuestra página web.



**Figura 8.6. Imagen página inicio.**

## 8.2.2. BOTÓN NUESTRA WEB

Al pinchar sobre el botón “NUESTRA WEB” en el cuerpo de nuestra página web podremos ver algunas de las características principales de la dieta mediterránea e información sobre la cantidad recomendable de lípidos, hidratos y proteínas que un adulto debe consumir al día. Una imagen del contenido de dicho botón se puede observar en la Figura 8.7.



**Figura 8.7. Imagen del contenido del botón nuestra web.**

## 8.2.3. BOTÓN BUSCAR RECETAS

Al pulsar sobre el botón llamado “BUSCAR RECETAS” aparecerá el contenido de todas las recetas que haya en nuestra base de datos (cuatro recetas por página). También aparecerá un buscador avanzado de recetas en el que se podrán introducir los ingredientes

que desea que contenga y que no contenga la receta, su provincia, su dificultad y si la receta necesita horno, batidora, microondas o thermomix, como se puede apreciar en la Figura 8.8.

**Nuestras recetas:**

Ingredientes	<input type="text"/>
Que no contenga	<input type="text"/>
Provincia	--Elija provincia-- <input type="button" value="▼"/>
Dificultad	--Elija dificultad-- <input type="button" value="▼"/>
Horno	<input type="checkbox"/> Microondas <input type="checkbox"/>
Batidora	<input type="checkbox"/> Thermomix <input type="checkbox"/>
<input type="button" value="BUSCAR"/>	

**ARROZ A LA CUBANA**

<b>Ingredientes Principales:</b> 300 gr. arroz 150 gr. salchicha 1 huevo 100 gr. tomate	<b>APORTE NUTRICIONAL</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Elementos</th> <th>Con Ingredientes Principales</th> <th>Con Ingredientes Opcionales</th> </tr> </thead> <tbody> <tr> <td>Aporte Calórico</td> <td>483.74</td> <td>499.5783</td> </tr> <tr> <td>Aporte Proteico</td> <td>14.2589</td> <td>14.4356</td> </tr> <tr> <td>Aporte Hidratos</td> <td>80.7523</td> <td>84.2189</td> </tr> <tr> <td>Grasas Totales</td> <td>11.057</td> <td>11.102</td> </tr> <tr> <td>Saturadas</td> <td>14.2823</td> <td>14.3023</td> </tr> <tr> <td>Monoinsaturadas</td> <td>4.313</td> <td>4.3197</td> </tr> <tr> <td>Poliinsaturadas</td> <td>1.2227</td> <td>1.2377</td> </tr> <tr> <td>Sodio</td> <td>0.8126</td> <td>0.8128</td> </tr> <tr> <td>Fibras</td> <td>1.8</td> <td>2.225</td> </tr> </tbody> </table>	Elementos	Con Ingredientes Principales	Con Ingredientes Opcionales	Aporte Calórico	483.74	499.5783	Aporte Proteico	14.2589	14.4356	Aporte Hidratos	80.7523	84.2189	Grasas Totales	11.057	11.102	Saturadas	14.2823	14.3023	Monoinsaturadas	4.313	4.3197	Poliinsaturadas	1.2227	1.2377	Sodio	0.8126	0.8128	Fibras	1.8	2.225
Elementos	Con Ingredientes Principales	Con Ingredientes Opcionales																													
Aporte Calórico	483.74	499.5783																													
Aporte Proteico	14.2589	14.4356																													
Aporte Hidratos	80.7523	84.2189																													
Grasas Totales	11.057	11.102																													
Saturadas	14.2823	14.3023																													
Monoinsaturadas	4.313	4.3197																													
Poliinsaturadas	1.2227	1.2377																													
Sodio	0.8126	0.8128																													
Fibras	1.8	2.225																													
<b>Ingredientes Opcionales:</b> 50 gr. plátano																															

**Preparación:** Freímos los huevos en una sartén con abundante aceite caliente. Por otro lado freímos las salchichas hasta que se doren. Preparamos el arroz en el microondas. Montamos el plato con un poco de arroz con tomate por encima y un huevo frito y salchichas acompañándolo. También se puede poner plátano frito para que se ajuste más a la receta original.




**Dificultad:** Baja  
**Tiempo de preparación:** 20 minutos  
**Número de comensales:** 3 personas

**Utensilios necesarios:**

Horno:	<input type="checkbox"/>
Batidora:	<input type="checkbox"/>
Microondas:	<input type="checkbox"/>
Thermomix:	<input type="checkbox"/>

**Figura 8.8. Imagen botón buscar recetas.**

### 8.2.3. BOTÓN ENVIAR RECETA

El contenido para enviar una nueva receta a nuestra página web sólo será visible para los usuarios registrados, es decir, sólo podrán enviar recetas nuevas aquellos usuarios que se hayan registrado con anterioridad. Dicho contenido lo podemos observar en la Figura 8.9. Por el contrario si un usuario no registrado pulsa sobre este botón se

mostrará un mensaje diciéndole que sólo los usuarios registrados pueden mandar nuevas recetas. Los campos que hay que llenar obligatoriamente serán los siguientes: título de la receta, número de comensales, tiempo de preparación, modo de preparación y al menos que contenga un ingrediente principal. Para añadir un nuevo ingrediente pulsaremos sobre el botón “añadir ingrediente” y éste nos creará una fila vacía idéntica a la anterior para introducir el nuevo ingrediente. Para añadir los ingredientes opcionales se procederá de la misma manera que con los ingredientes principales, siendo éstos no obligatorios.

Para enviar una foto, el archivo deberá ser menor de 300 *kb* y tener un nombre que no esté contenido en la base de datos. Si ya estuviera la foto o su tamaño fuera menor, se mostrará un mensaje de error.

Por su parte el modo de preparación será obligatorio y podrá editarse como el usuario lo prefiera: letra en negrita, cursiva, subrayada, párrafos enumerados, texto centrado, justificado, a derecha, a izquierda, con distintos tamaño de letra, distinta tipografía, etcétera.

Muchas gracias por colaborar con la web.

**Título de la receta:**

**Tipo de plato:**  Arroz  
Pescado  
Pasta  
Carne

**Provincia:**  Huelva

**Nivel de dificultad:**  Muy Baja

**Calculada para:**  personas

**Horno**   
**Batidora**   
**Microondas**   
**Thermomix**   
**Vegetariana**   
**Vegana**   
**Celiacos**   
**Light**

**Tiempo de preparación:**  minutos

**Introduzca foto:**  Seleccionar archivo. No se ha seleccionado ningún archivo.

**Ingredientes Principales:**  
Indique por separado cantidad, medida e ingrediente.

Cantidad	Medida	Producto
<input type="text"/>	<input type="text"/> unidad o pieza <input type="button" value="&gt;"/>	<input type="text"/>

Una vez introducida la cantidad, medida y nombre del ingrediente, pulse "añadir ingrediente" para añadir un nuevo

**Ingredientes Opcionales:**  
Indique por separado cantidad, medida e ingrediente.

Cantidad	Medida	Producto
<input type="text"/>	<input type="text"/> unidad o pieza <input type="button" value="&gt;"/>	<input type="text"/>

Una vez introducida la cantidad, medida y nombre del ingrediente, pulse "añadir ingrediente" para añadir un nuevo

**Modo de preparación:**



**Correo:**

**Observaciones:**



**Figura 8.9. Imagen enviar receta usuario registrado.**

### 8.1.5. BOTÓN GLOSARIO

Al pulsar sobre el botón llamado “GLOSARIO” aparecerán una serie de expresiones y sus correspondientes definiciones (cuatro por página) con el formato que podemos apreciar en la Figura 8.10. También encontraremos un enlace para agregar nuevos términos.

**GLOSARIO DE TÉRMINOS | [AGREGAR NUEVO TÉRMINO](#)**

**Expresión:** ADEREZAR

**Definición:** Agregar sal, aceite, vinagre, especias, etc, a ensaladas u otras preparaciones frías.

**Expresión:** ALÍÑAR

**Definición:** Aderezar o sazonar.

**Expresión:** BAÑO MARÍA

**Definición:** Baño de agua usado para cocinar lentamente alimentos que se encuentran dentro de un envase rodeado con agua hirviendo en un recipiente de mayor tamaño.

**Expresión:** BATIR

**Definición:** Mover y revolver alguna sustancia para que se condense o trabe, o para que se lique o disuelva.

[1](#) [2](#) [3](#) [Siguiente >](#)

**Figura 8.10. Imagen botón glosario.**

#### 8.1.5.1. ENLACE AGREGAR NUEVO TÉRMINO

Al igual que en el botón enviar receta, sólo podrán agregar nuevos términos al glosario los usuarios registrados con anterioridad. Si un usuario no está registrado se mostrará un mensaje diciendo que sólo los usuarios registrados pueden enviar nuevos términos. El formulario para enviar un nuevo término lo podemos observar en la Figura 8.11. Los campos expresión y definición serán obligatorios.

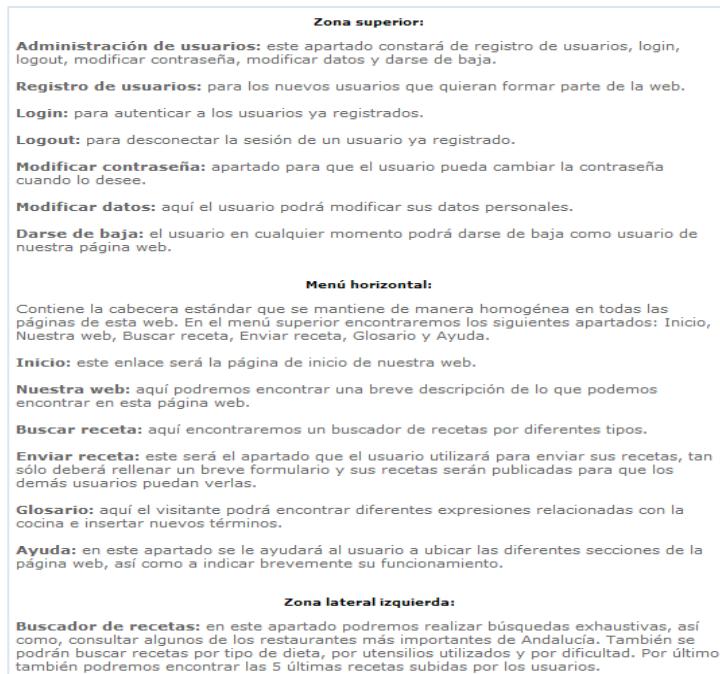
**AGREGAR NUEVO TÉRMINO**

Expresión:*	<input type="text"/>
Definición:*	     
<input type="button" value="ENVIAR"/>	

**Figura 8.11. Imagen agregar nuevo término.**

### 8.1.6. BOTÓN AYUDA

Al pulsar sobre el botón “AYUDA” aparecerá una breve descripción de los contenidos de la página web para facilitar la navegación al usuario. Dicho contenido lo podemos observar en la Figura 8.12.



**Figura 8.12. Imagen ayuda.**

## 8.1.7. BOTÓN ÁREA GESTIÓN

El contenido de este botón como se indicó anteriormente sólo será visible para los administradores de la página web. Al pulsar sobre este botón, como podemos apreciar en la Figura 8.13 aparecerá justo debajo un submenú con cuatro botones cuya utilidad explicaremos a continuación.



Figura 8.13. Imagen menú administrador.

### 8.1.7.1. BOTÓN GESTIÓN USUARIOS

En cualquier momento el administrador podrá dar de baja a un usuario que esté haciendo un mal uso de la página web, tan sólo deberá pulsar sobre el botón eliminar que podemos observar en la Figura 8.14.

Gestión de usuarios:			
Nombre	Apellidos	Correo	
Javier	Cervilla Constant	cervolla@hotmail.com	<button>ELIMINAR</button>
Jessica	González Ramet	jessica@hotmail.com	<button>ELIMINAR</button>
jaime	Peralto Galán	jaime@hotmail.com	<button>ELIMINAR</button>
Melina	Sánchez Gutiérrez	melina@hotmail.com	<button>ELIMINAR</button>

Figura 8.14. Imagen gestión de usuarios.

### 8.1.7.2. BOTÓN VALIDAR RECETAS

El administrador podrá validar una receta si el contenido de la misma es correcto, para que pueda ser mostrada en nuestra página web. Por el contrario si el contenido de la receta no le gusta podrá eliminar la receta pulsando sobre el botón eliminar. Al pulsar sobre el botón eliminar receta se nos pedirá una confirmación para eliminar la receta definitivamente. Una imagen del validador de recetas la podemos observar en la Figura 8.15. El administrador también tendrá la posibilidad de modificar la receta si lo estimara oportuno. Las recetas que no hayan sido validadas por el administrador no serán mostradas en la página web.

**Validador de recetas:**

Título:	ARROZ A LA CUBANA
Duración:	20
Dificultad:	Baja
Preparación:	<p>Freímos los huevos en una sartén con abundante aceite caliente. Por otro lado freímos las salchichas hasta que se doren. Preparamos el arroz en el microondas. Montamos el plato con un poco de arroz con tomate por encima y un huevo frito y salchichas acompañándolo. También se puede poner plátano frito para que se ajuste más a la receta original.</p>
Horno:	<input type="checkbox"/>
Batidora:	<input type="checkbox"/>
Microondas:	<input type="checkbox"/>
Thermomix:	<input type="checkbox"/>
Celiacos:	<input type="checkbox"/>
Light:	<input type="checkbox"/>
Vegetariana:	<input type="checkbox"/>
Vegana:	<input type="checkbox"/>
Fecha:	2012-06-04
Comensales:	3
Observaciones:	

**ELIMINAR** **VALIDAR**

**Figura 8.15. Imagen validar recetas.**

### 8.1.7.3. BOTÓN VALIDAR INGREDIENTES

El administrador deberá completar la información de un ingrediente que no estuviera almacenado anteriormente en nuestra base de datos. Dicha información la podemos observar en la Figura 8.16. También tendrá la posibilidad de eliminar un ingrediente no validado en cualquier momento.

**Validador de ingredientes:**

Nombre:	arroz
Calorías:	349 [kcal/100 gramos]
Proteínas:	6.9 [g/100 gramos]
Hidratos:	78.2 [g/100 gramos]
Grasas saturadas:	0.15 [g/100 gramos]
Grasas totales:	0.6 [g/100 gramos]
Grasas monoinsaturadas:	0.17 [g/100 gramos]
Grasas poliinsaturadas:	0.2 [g/100 gramos]
Sodio:	0.04 [g/100 gramos]
Fibra:	1.4 [g/100 gramos]

Observaciones:

B I U Font Size...  
Font Family. Font Format

ELIMINAR VALDAR

**Figura 8.16. Imagen validar ingredientes.**

### 8.1.7.4. BOTÓN VALIDAR TÉRMINO

Al igual que en los botones para validar recetas e ingredientes el administrador podrá eliminar y validar términos si lo cree conveniente como podemos observar en la

Figura 8.17. Los términos que no hayan sido validados por el administrador no se mostrarán en la página web, al igual que pasa con las recetas.

**Validador de términos:**

Expresión: ABRILLANTAR

B I U Font Size...  
Font Family. Font Format Font Image Link Email Phone

Definición: Dar brillo a pasteles o a otros preparados utilizando jalea, gelatina, manteca clarificada o yema de huevo.

ELIMINAR VALIDAR

Expresión: ACREMAR

B I U Font Size...  
Font Family. Font Format Font Image Link Email Phone

Definición: Dar consistencia untuosa, de crema o pomada, a un ingrediente de densidad media.

ELIMINAR VALIDAR

1

**Imagen 8.17. Botón validar término.**

### 8.3. COLUMNA IZQUIERDA

Estas búsquedas estarán disponibles para cualquier usuario y podrán realizarse en cualquier momento. Los resultados de estas búsquedas se mostrarán en diferentes páginas de cuatro en cuatro. El contenido de la columna izquierda lo podemos observar en la Figura 8.18.

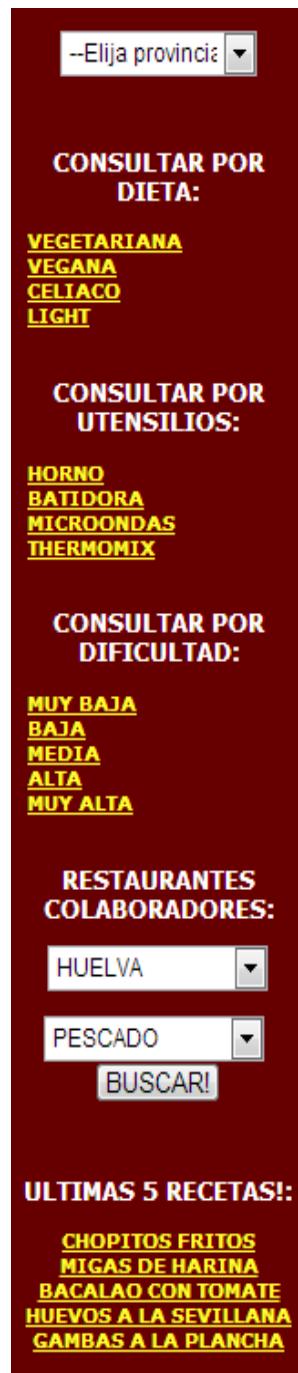


Figura 8.18. Imagen columna izquierda.

### 8.3.1. BÚSQUEDA POR PROVINCIA

Al seleccionar una provincia del menú desplegable se mostrarán todas las recetas de esa provincia almacenadas en nuestra base de datos (cuatro por página). El formato de las recetas se puede apreciar en la Figura 8.19 y será el mismo para cualquier tipo de búsqueda.

### Mostrando recetas de la provincia de HUELVA

Ingredientes	<input type="text"/>		
Que no contenga	<input type="text"/>		
Provincia	--Elija provincia-- <input type="button" value="▼"/>		
Dificultad	--Elija dificultad-- <input type="button" value="▼"/>		
Horno	<input type="checkbox"/>	Microondas	<input type="checkbox"/>
Batidora	<input type="checkbox"/>	Thermomix	<input type="checkbox"/>
Vegana	<input type="checkbox"/>	Vegetariana	<input type="checkbox"/>
Light	<input type="checkbox"/>	Celiacos	<input type="checkbox"/>
<input type="button" value="BUSCAR"/>			

#### ANCHOAS FRITAS

**Ingredientes Principales:**

250 gr. anchoas  
50 gr. ajo  
100 gr. aceite de oliva  
15 gr. sal

**Ingredientes Opcionales:**  
10 gr. guindilla.

APORTE NUTRICIONAL		
Elementos	Con Ingredientes Principales	Con Ingredientes Opcionales
Aporte Calórico	530.3333	531.9067
Aporte Proteico	24.9	24.9623
Aporte Hidratos	4.3583	4.5817
Grasas Totales	46.005	46.0417
Saturadas	24.9	24.9033
Monoinsaturadas	26.6717	26.6737
Poliinsaturadas	9.6667	9.6867
Sodio	5.2492	5.5492
Fibras	0.2	0.25

**Preparación:** Primero quitamos la cabeza y las tripas a las anchoas, las lavamos bien y las sazonamos con sal. Luego, en una sartén con bastante aceite, freímos los ajos y la guindilla. Antes de que queden doraditos, echamos las anchoas. Las freímos vuelta y vuelta. Servimos bien calientes.




**Dificultad:** Muy Baja  
**Tiempo de preparación:** 10 minutos  
**Número de comensales:** 3 personas

**Utensilios necesarios:**

Horno:   
 Batidora:   
 Microondas:   
 Thermomix:

**Observaciones:**  
Anchoas fritas.

Figura 8.19. Imagen buscador de recetas.

### 8.3.2. BÚSQUEDA POR DIETA

Al pinchar sobre estos enlaces se mostrarán las recetas ordenadas por tipo de dieta seleccionada: vegetariana, vegana, celíaca o light. El formato de las recetas mostradas será el mismo que en el apartado 8.3.1.

### **8.3.3. BÚSQUEDA POR UTENSILIOS**

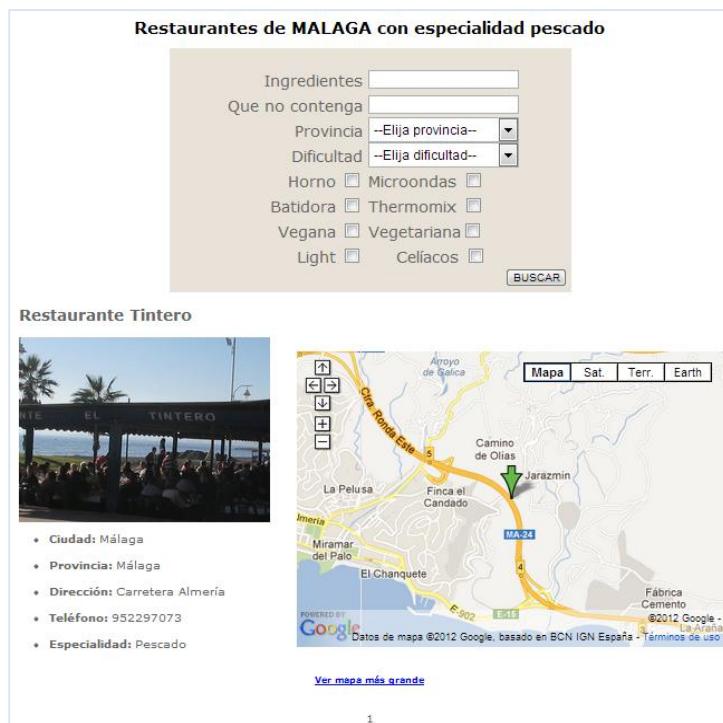
Este apartado será útil para buscar recetas que necesiten un tipo de utensilio determinado: horno, batidora, microondas, thermomix. El formato de las recetas mostradas será el mismo que en el apartado 8.3.1.

### **8.3.4. BÚSQUEDA POR DIFICULTAD**

En este apartado se mostrarán las recetas clasificadas por su dificultad: muy baja, baja, media, alta, muy alta. El formato de las recetas mostradas será el mismo que en el apartado 8.3.1.

### **8.3.5. BÚSQUEDA DE RESTAURANTES**

Para realizar una búsqueda será necesario introducir la provincia y la especialidad de los restaurantes buscados. Una imagen sobre el contenido de los restaurantes de la provincia de Málaga con especialidad pescado se puede observar en la Figura 8.20.



**Figura 8.20. Imagen buscador de restaurantes.**

### **8.3.6. ÚLTIMAS CINCO RECETAS**

Al pinchar sobre cualquiera de las cinco últimas recetas aparecerá una nueva pestaña con la información correspondiente a la receta pulsada. El formato de la receta será el mismo que el mostrado en el apartado 8.3.1.

# CAPITULO 9

# CONCLUSIONES Y AMPLIACIONES

Con este proyecto hemos intentado realizar una página web amena, atractiva, divertida e interactiva para los usuarios que la visiten. El objetivo es que los visitantes consulten sus contenidos y se registren e interactúen con la web, bien enviando nuevas recetas o agregando nuevos términos. La web está orientada a recetas de nuestra tierra, Andalucía, debido a la riqueza gastronómica que tenemos y a su tan conocida dieta mediterránea. La elaboración de este proyecto ha servido para aprender por mi cuenta *HTML, PHP, CSS* y algo de *Javascript* y *Jquery*, ya que, en la facultad por falta de tiempo no hay ninguna asignatura que explique y enseñe como realizar una página web desde cero.

La dificultad más importante que me he encontrado a lo largo de la realización del proyecto ha sido a la hora de introducir una nueva receta en nuestra base de datos, ya que, hay que introducir numerosos ingredientes principales y optionales e insertar los distintos campos del formulario en diferentes tablas.

Entre las posibles mejoras que podemos hacer en un futuro a nuestro proyecto se encuentran las siguientes:

- Ampliar la página web para que se puedan insertar recetas de cualquier punto de la geografía española, no sólo de Andalucía.
- Introducir publicidad de restaurantes colaboradores, con esa intención he creado las búsquedas de los restaurantes.
- Introducir un *captcha* en los formularios en los que se envíen datos, para así evitar el *spam* en nuestra web.
- Poner un límite de tiempo para que se cierre la sesión por sí sola cuando ha transcurrido un límite de tiempo sin actividad por parte del usuario.
- Ampliar las búsquedas que puedan hacer los usuarios por tipo de receta.
- Ampliar el buscador avanzado, visualizando los ingredientes almacenados en nuestra base de datos para que el usuario pueda seleccionarlos.
- Introducir un enlace en la cabecera para que el usuario pueda cambiar su clave si no se acuerda.
- Introducir un nuevo botón en la gestión de usuarios para que el administrador pueda cambiar la contraseña de los usuarios registrados, por si quisiera bloquearlos temporalmente.
- Mostrar iconos con los utensilios utilizados al lado de los *links* de las últimas cinco recetas.

- Cambiar la tabla provincia por otra llamada “zona”, ya que, provincia es algo muy concreto. Además las zonas podrían estar incluidas en otras zonas para así tener una jerarquía, por ejemplo, Andalucía es de España, pero a su vez lo es también de Europa
- Crear un nuevo botón en el área de administradores que permita eliminar recetas validadas.



# REFERENCIAS

- [1] Web recetasdecocina: [www.recetasdecocina.es](http://www.recetasdecocina.es) (Accedido en Marzo 2012)
- [2] Web cocinatusrecetas: [www.cocinatusrecetas.com](http://www.cocinatusrecetas.com) (Accedido en Marzo 2012)
- [3] Web recetario-cocina: [ww.recetario-cocina.com](http://www.recetario-cocina.com) (Accedido en Marzo 2012)
- [4] Web Las recetas del chef: [www.lasrecetasdelchef.net](http://www.lasrecetasdelchef.net) (Accedido en Marzo 2012)
- [5] Web Recetas.net: [www.recetas.net](http://www.recetas.net) (Accedido en Marzo 2012)
- [6] Web Quiero Cocinar: [www.quierococinar.com](http://www.quierococinar.com) (Accedido en Marzo 2012)
- [7] Web A Cocinar: [www.acocinar.com](http://www.acocinar.com) (Accedido en Marzo 2012)
- [8] Web Recetasdiarias: [www.recetasdiarias.com](http://www.recetasdiarias.com) (Accedido en Marzo 2012)
- [9] Wikipedia, HTML: <http://es.wikipedia.org/wiki/HTML> (Accedido en Marzo 2012)
- [10] Rubén Álvarez, 2001. Desarrolloweb, Manual de HTML:  
<http://www.desarrolloweb.com/manuales/21/>.
- [11] Wikipedia, Hojas de estilo en cascada: [http://es.wikipedia.org/wiki/Hojas\\_de\\_estilo\\_en\\_cascada](http://es.wikipedia.org/wiki/Hojas_de_estilo_en_cascada)
- [12] Miguel Ángel Álvarez, 2001. Desarrolloweb, Introducción a las CSS:  
<http://www.desarrolloweb.com/articulos/181.php>.
- [13] Wikipedia: <http://es.wikipedia.org/wiki/Php> (Accedido en Marzo de 2012)
- [14] Rubén Álvarez, 2001. Desarrolloweb, Introducción a la programación en PHP:  
<http://www.desarrolloweb.com/articulos/303.php>.
- [15] Wikipedia, WAMP: <http://es.wikipedia.org/wiki/WAMP> (Accedido en Marzo de 2012)
- [16] Wikipedia, Adobe Dreamweaver: <http://es.wikipedia.org/wiki/Dreamweaver> (Accedido en Marzo 2012)
- [17] Wikipedia, Servidor HTTP Apache: [http://es.wikipedia.org/wiki/Servidor\\_HTTP\\_Apache](http://es.wikipedia.org/wiki/Servidor_HTTP_Apache) (Accedido en Marzo 2012)
- [18] Wikipedia, MySQL: <http://es.wikipedia.org/wiki/MySQL> (Accedido en Marzo de 2012)
- [19] Wikipedia, phpMyAdmin: <http://es.wikipedia.org/wiki/PhpMyAdmin> (Accedido en Marzo de 2012)
- [20] Wikipedia, Dia (programa): [http://es.wikipedia.org/wiki/Dia\\_\(programa\)](http://es.wikipedia.org/wiki/Dia_(programa))
- [21] Wikipedia, MySQL Workbench: [http://es.wikipedia.org/wiki/MySQL\\_Workbench](http://es.wikipedia.org/wiki/MySQL_Workbench)
- [22] D. Zawodny, Jeremy y Balling, Derek J. (2004). MySQL Avanzado. Anaya. Málaga: Biblioteca Universitaria de Málaga.
- [23] Gilfillan, Ian (2003). La biblia de MySQL. Anaya. Málaga: Biblioteca Universitaria de Málaga.
- [24] Desarrolloweb, HTML a fondo: <http://www.desarrolloweb.com/html/>
- [25] Desarrolloweb, Programación en PHP: <http://www.desarrolloweb.com/manuales/12/>

- [26] Desarrolloweb, Manual de CSS, hojas de estilo: <http://www.desarrolloweb.com/manuales/manual-css-hojas-de-estilo.html>
- [27] Felipe Javier García Gallego, CSS desde cero: <http://css.infames.org/ventajas.html>
- [28] Miguel Ángel Álvarez, 2001. Desarrolloweb, Características y ventajas de las CSS: <http://www.desarrolloweb.com/articulos/182.php>