



Centro Profesional
Universidad Europea Madrid
LAUREATE INTERNATIONAL UNIVERSITIES

**ESCUELA
POLITÉCNICA**

**CICLO FORMATIVO DE GRADO SUPERIOR
EN
DESARROLLO DE APLICACIONES MULTIPLATAFORMA**

OFERTAPAS

MIGUEL ANGEL DE LEÓN GARCÍA

CURSO 2013-2014

Resumen

Cualquier persona con un pub o un restaurante puede publicar sus ofertas en “Ofertapas”. Ellos pueden crear sus propias ofertas y publicarlas gratis en la web.

Después de publicar en ofertapas.com, será visible para todos los usuarios que acceden a la página web que buscan disfrutar de unas tapas en Madrid.

El usuario registrado puede acceder a su perfil y cambiarlo cuando quiera. También puede seleccionar algunas fotos y publicarlas en la web.

Los clientes podrán encontrar la dirección del pub o del restaurante en la web. Ellos podrán encontrar el mejor precio y el mejor lugar de Madrid para cenar, comer o simplemente tomar una cerveza fría.

Los clientes pueden buscar ofertas diarias por distrito, y de cada oferta pueden ver los precios, una imagen de la oferta, y también información sobre el bar o restaurante. Para ver estas ofertas no se requiere registro.

Los propietarios podrán publicar tantas ofertas como deseen y modificar las existentes. Para el propietario de Bar se requerirá registro si quieren publicar ofertas.

El sitio web ha sido desarrollado utilizando PHP, HTML5, CSS, AJAX, y también se ha utilizado como base de datos MYSQL.

Palabras Clave

Asíncrono, Hoja de estilo en cascada, jQuery, Procesador Hiper texto, cualquiera, clientes, Lenguaje de Modelado Unificado, dueño del bar, oferta...

Abstract

Any owner of a bar or restaurant can publish their offers in “Ofertapas.com”. They can create their own deals and publish them on the website for free.

After an offer is published in “ofertapas.com”, will be visible for all the users that access to the website looking for a bar in Madrid to enjoy the Spanish tapas.

The customers will be able to find the address of pub or restaurant on website. They will be able to find the best price and the best place of Madrid for dinner, lunch or just drink a cold beer.

Customers can search daily offers by district, and in each offer they can see price, a picture of the offer, and also information about the bar or restaurant. To see these offers is not required register.

Bar owner`s can publish as many offers as they want, also they can modify their existing offers. For Bar owner`s will be required register if they want to publish offers.

The website has been developed using PHP, HTML5, CSS, AJAX, and also has been used MYSQL as database.

Keywords

Asynchrony, Cascading Style Sheets, jQuery, Hypertext Pre-processor, Any owner, customers, Unified Modeling Language, Bar owner`s, offer ...

Agradecimientos

Después de dos años creo que estoy muy satisfecho con el curso realizado, ya que he adquirido las bases de programación para seguir avanzando en el desarrollo de aplicaciones en distintos lenguajes. También sé que esto sólo acaba de empezar y espero seguir aprendiendo lo mucho que me queda por conocer del mundo de la programación.

He aprendido a crear a partir de una idea inicial aplicaciones web y móvil con las últimas tecnologías del mercado (Android, Java, JavaScript, PHP, HTML5..) .

Todo este aprendizaje no habría sido posible sin la ayuda de mis compañeros y profesores a los cuales agradezco la ayuda que me han ofrecido en todo momento

Tabla de Contenidos

PARTE I. INTRODUCCIÓN A LA INVESTIGACIÓN	3.
CAPÍTULO 1. INTRODUCCIÓN.....	5.
1.1. Planteamiento y justificación del trabajo.....	6.
1.2. Hipótesis y objetivos	7.
1.3. Organización del trabajo fin de ciclo.....	8.
PARTE II. ESTADO DEL ARTE.....	10.
CAPÍTULO 2. Estado del Arte	11.
2.1. Tecnologías utilizadas	12.
2.1.1. Html.....	13.
2.1.2. Css	13.
2.1.3. PHP.....	14.
2.1.4. jQuery.....	14.
2.1.5. AJAX.....	15.
2.1.6 MySQL	15.
2.2. Herramientas utilizadas	16.
PARTE III. DESARROLLO DE LA INVESTIGACIÓN	18.
CAPÍTULO 3. Especificación de Requisitos.....	19.
3.1. Descripción general	20.
3.1.1. Perspectiva del producto.....	20.
3.1.2. Funciones del producto.....	20.
3.2. Requisitos Específicos.....	21.
3.2.1. Requisitos de interfaces externas.....	21.
3.2.2. Requisitos Funcionales	21.
CAPÍTULO 4. Análisis	24.
4.1. UML	25.
4.1.1. Diagrama de clases	25.
4.2. Diagrama Casos de uso	26.
4.2.1. Caso de uso: Anónimo.....	27.
4.2.2. Caso de uso: Usuario Registrado.....	27.

CAPÍTULO 5. Diseño	29.
5.1. Arquitectura del sistema	30.
5.1.1. Nivel de Presentación	30.
5.1.1.1. Interfaz Gráfica.....	31.
5.1.2. Nivel de Aplicación	37.
5.1.3. Nivel de Persistencia	38.
 CAPÍTULO 6. Evaluación y Pruebas	 39.
6.1. Prueba enlaces rotos	40.
6.2. Prueba validación con Html	40.
6.3. Prueba validación Css.....	40.
6.4. Prueba compatibilidad móvil	40.
6.5. Prueba compatibilidad navegadores	41.
6.6. Prueba compatibilidad resoluciones	42.
6.7. Evaluación casos de uso	43.
 PARTE IV. CONCLUSIONES	 46.
 CAPÍTULO 7. Conclusiones.....	 47.
7.1. Aportaciones originales	48.
7.2. Trabajos relacionados	49.
7.3. Líneas de investigación futuras	49.
 BIBLIOGRAFÍA	 51.
 PARTE V. ANEXOS	 52.
A. Anexo 1. Implementación.....	53.
A.1.1 Plantilla.php.....	53.
A.1.2.Comunicación entre Ver Ofertas publicadas y Editar oferta.	54.
A.1.3. Validar dinámicamente si el email de registro existe.	55.

PARTE I

Introducción a la Investigación

Capítulo 1 Introducción

En este capítulo hablaremos sobre los problemas actuales de los pequeños locales en el mercado hostelero, cómo puede ayudar nuestro portal web y los objetivos que se quieren cumplir con este proyecto.

Este capítulo se estructura en:

- Planteamiento y justificación del trabajo.

En este apartado comentaremos porqué hemos utilizado un portal web y no hemos realizado una aplicación.

Se justificará qué tiene nuestro portal que otros similares no tienen así como los problemas que soluciona.

- Hipótesis y objetivos.

Se explica de una manera clara qué persigue nuestra idea.

- Organización del trabajo fin de ciclo.

Explicamos cómo se desarrolla el resto de la memoria.

1.1 PLANTEAMIENTO Y JUSTIFICACIÓN DEL TRABAJO.

Las páginas web después de sufrir la fiebre de las aplicaciones han vuelto pero ésta vez con más fuerza, la aparición de HTML5 y el desarrollo de otras tecnologías relativamente nuevas como AJAX y jQuery combinado con las conocidas Php y Css hacen que crear una web se pueda igualar a una aplicación en cuanto a la demanda actual de consulta de datos móvil se refiere.

Si nuestra información la podemos ofrecer en una tecnología que no necesita instalación y puede consultarse desde el móvil igualmente... **¿Por qué no hacerlo así?**

El proyecto presentado surge por la idea de dos motivos:

1. Con la aparición de páginas de ofertas (groupon, groupalia..) junto con las franquicias (100 montaditos, Lizarrán, Cañas y tapas, Fridays...) significan un gran competidor a locales más pequeños de Madrid, este importante número de bares, restaurantes o pubs queda aislado, ya que no pueden competir en precios y menos en publicidad.

Otros portales de ofertas cobran una parte de la oferta que paga el usuario, además de la rebaja del precio de la oferta dejando apenas sin beneficios e incluso perdiendo dinero sólo por darse a conocer. Evidentemente los locales independientes no pueden correr esos riesgos.

2. Ofertapas nace de la demanda constante de ofertas por parte de los usuarios de internet y para que el gran número de locales hosteleros existentes en la capital, puedan darse a conocer a través de la red y a la vez servir de puente entre clientes y locales de manera que esa comunicación se traduzca en un aumento de beneficios para el propietario del local.

Con el portal “www.ofertapas.com” queremos que todos los bares, pubs o restaurantes, por pequeños que sean, puedan publicar sus ofertas de manera rápida, sencilla y gratuita.

Los posibles clientes podrán conocer estos “pequeños locales”, en un mismo portal de ofertas sin que sean eclipsados por las grandes franquicias.

Los posibles propietarios que se registren no tendrán que compartir beneficios con el portal web ajustando más el precio de sus ofertas.

Se ha centrado en una única ciudad para darle cercanía al portal, por ser una ciudad importante de turismo nacional e internacional y por la proximidad de hacer publicidad para captar clientes para las ofertas y propietarios para que se registren.

1.2. HIPÓTESIS Y OBJETIVOS

La finalidad es crear un portal web de ofertas, en el que los bares, restaurantes y pubs de Madrid puedan ofertar sus productos a los visitantes del portal.

El portal de ofertas, en adelante Ofertapas, tratará de cumplir dos objetivos esenciales:

- 1- Servir de referente en el sector hostelero de la capital, creando un portal claro y ordenado, en el que quede patente para todo el que lo visite, de la oferta gastronómica en la ciudad, de la zona que elija y las cualidades del local de la oferta que seleccione.

Para ayudar al visitante a encontrar lo que está buscando, se han crearán listas desplegables para que el posible cliente busque según su criterio de búsqueda.

- 2- Albergar una intranet para que los dueños de locales, puedan ofertar y editar los productos que ofrecen de forma rápida y sencilla.

1.3. ORGANIZACIÓN DEL TRABAJO FIN DE CICLO

Este documento presenta una estructura en etapas, común en el desarrollo de un proyecto software. Estas etapas serán las siguientes:

Parte II Estado del Arte

- **Estado del arte:** En esta etapa, se describen las diferentes tecnologías utilizadas y herramientas usadas en el desarrollo del proyecto.

Parte III Desarrollo de la Investigación

- **Especificación de requisitos:** En esta fase se reúnen todos los requisitos funcionales que debe tener el producto final.

- **Análisis:** El Análisis describe la estructura y funcionalidad del producto mediante diagramas que permiten comprender el sistema y su funcionamiento. Se incluyen diagrama de clases UML y casos de uso, que describen los principales comportamientos del portal.

- **Diseño:** En esta fase se explican los diferentes niveles que componen la arquitectura de la aplicación (nivel de presentación, nivel lógico y nivel de persistencia).

- **Evaluación y pruebas:** En este apartado se efectúan una serie de pruebas para evaluar el correcto funcionamiento del sistema.

Parte IV Conclusiones

Aquí se exponen las conclusiones de la investigación, bibliografía e incluyendo próximas mejoras, trabajos relacionados...

Parte V Anexos

Anexo 1 - Implementación del código.

PARTE II

Estado del Arte

Capítulo 2 Estado del Arte

En esta etapa, se describen los siguientes apartados:

- Las diferentes tecnologías utilizadas.
- Las herramientas usadas en el desarrollo del proyecto.

2.1. TECNOLOGÍAS UTILIZADAS

Para realizar este proyecto se han utilizado varias tecnologías y lenguajes, ya sea para el aspecto visual de la web, como del comportamiento y la dinámica.

Para el aspecto visual de la página se ha utilizado en conjunto HTML y Css.

El primero para la estructura y el segundo para el aspecto.

Para el comportamiento de la web se ha utilizado PHP en el lado del servidor y jQuery (Javascript) para el lado del cliente.

Con PHP nos ocupamos de las acciones de acceso a la base de datos y de envío y recepción de datos desde la web a la base de datos y viceversa, así como de la comprobación de la sesión iniciada y autenticación.

Con JQuery hemos conseguido que la interfaz sea dinámica, es decir, que al realizar una petición, por ejemplo al rellenar el email en el formulario de registro, se comprueba si ese email ya ha sido dado de alta sin tener que recargar la web.

Para la base de datos se ha utilizado MySQL.

Veamos un poco más detalladamente cada una de ellas.

2.1.1. HTML

El HTML, Hyper Text Markup Language (Lenguaje de marcación de Hipertexto) es el lenguaje de marcas de texto utilizado normalmente en la www (World Wide Web).

Fue creado en 1986 por el físico nuclear Tim Berners-Lee, el cual tomo dos herramientas preexistentes: El concepto de Hipertexto (Conocido también como link o ancla) que permite conectar dos elementos entre si y el SGML (Lenguaje Estándar de Marcación General) que sirve para colocar etiquetas o marcas en un texto que indique como debe verse.

HTML es un sistema de etiquetas que no requiere compilador. Estos documentos pueden ser mostrados por los navegadores.

2.1.2. CSS

Hojas de Estilo en Cascada (Cascading Style Sheets), es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura.

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.

CSS funciona a base de reglas. Las hojas de estilo están compuestas por una o más de esas reglas aplicadas a un documento HTML o XML.

La regla tiene dos partes: un selector y la declaración.

A su vez la declaración está compuesta por una propiedad y el valor que se le asigne.

```
h1 {color: red;}
```

2.1.3. PHP

PHP es un lenguaje de programación interpretado, de uso general de código del lado del servidor originalmente diseñado para la creación de páginas web dinámicas

PHP es un acrónimo recursivo que significa PHP Hypertext Pre-processor (inicialmente PHP Tools, o, Personal Home Page Tools).

Fue creado originalmente por Rasmus Lerdorf en 1994.

Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos.

Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno.

2.1.4. jQuery (Javascript)

JavaScript es un lenguaje de programación interpretado.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web.

jQuery es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.

Las aplicaciones en internet son cada vez más complejas, ya que incorporan efectos visuales, auto-completar, animaciones etc. el desarrollar todos estos conceptos desde cero puede resultar complicado sobre todo si tenemos que presentar la solución con muy poco tiempo, en este tipo de situaciones el empleo de librerías como el jQuery nos facilitan el desarrollo de la aplicación.

2.1.5. AJAX

AJAX, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas.

Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, mejorando la interactividad, velocidad y usabilidad en las aplicaciones.

2.1.6. MySQL

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario.

Al principio carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de ello, atrajo a los desarrolladores de páginas web con contenido dinámico, justamente por su simplicidad.

Poco a poco los elementos de los que carecía MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre.

Ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor.

Soporta gran cantidad de datos. MySQL Server tiene bases de datos de hasta 50 millones de registros.

Permite escoger entre múltiples motores de almacenamiento para cada tabla.

2.2. HERRAMIENTAS UTILIZADAS

Para la implementación de la web y la base de datos se han utilizado diferentes aplicaciones o herramientas que son las siguientes:

- Para la creación de HTML y Css, así como los Scripts de PHP y Javascript se ha utilizado Adobe Dreamweaver CS6.

- La base de datos MySQL ha sido mantenida mediante PhpMyAdmin, una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web, utilizando Internet.

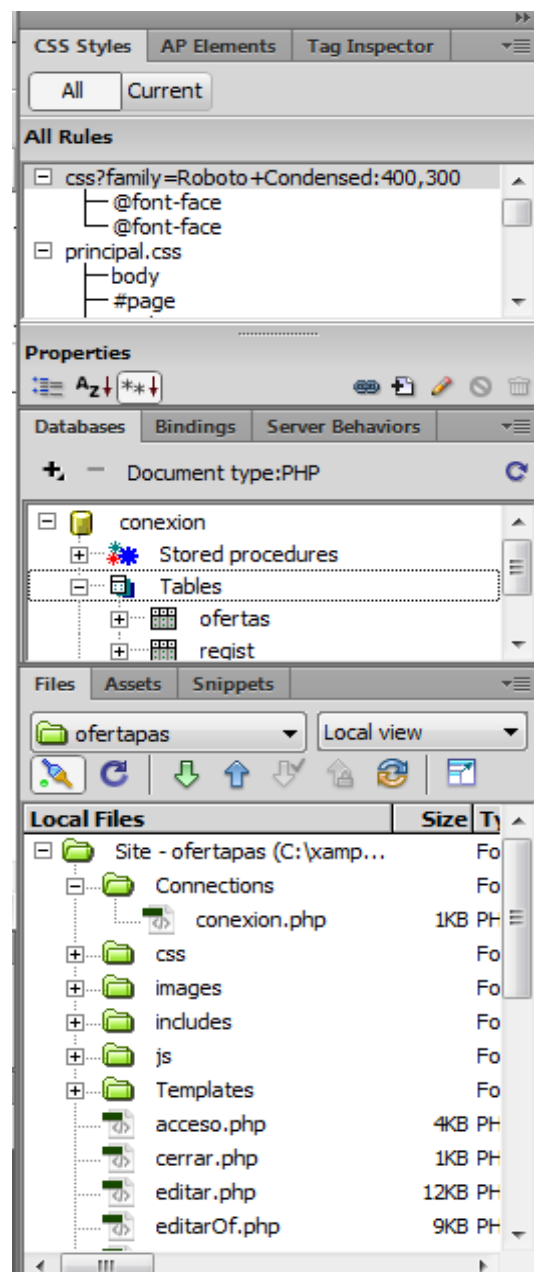
- Para las pruebas y comprobaciones de la web se han utilizado distintos navegadores (Firefox, Internet Explorer y Chrome).

Dreamweaver CS6 es un software fácil de usar que permite crear páginas web profesionales.

Se puede crear tablas, editar marcos, trabajar con capas, insertar comportamientos JavaScript, etc., de una forma muy sencilla y visual.

Además incluye un software de cliente FTP completo, permitiendo entre otras cosas trabajar con mapas visuales de los sitios web, actualizando el sitio web en el servidor sin salir del programa.

La imagen de la derecha corresponde al entorno de trabajo de la web presentada.



PARTE III

Desarrollo de la Investigación

Capítulo 3 Especificación de Requisitos

La especificación de requisitos tiene como finalidad conocer lo que se demanda y se espera obtener de la aplicación, muy importante para el desarrollo de la misma.

El capítulo 3 se divide en:

- Visión global y descripción general del producto.
- Funciones del producto, operaciones permitidas y restricciones generales.
- Requisitos específicos.

3.1. DESCRIPCIÓN GENERAL

El producto a desarrollar es un portal web de ofertas en línea.

El objetivo final es dar a conocer los locales registrados por los visitantes y aumentar así sus clientes.

Los visitantes de la web podrán informarse sobre la variada oferta gastronómica de Madrid.

Para acceder a la intranet, cada local deberá estar registrados previamente y dado de alta en la base de datos.

3.1.1 Perspectiva del producto

La aplicación que desarrollamos es totalmente independiente.

El sitio web podrá ser visitado por cualquier usuario, independiente de la plataforma y del navegador que utilice.

3.1.2 Funciones del producto

Operaciones permitidas

En este apartado se enumerarán las operaciones permitidas para cada uno de los usuarios.

- Anónimo:

Ver la oferta gastronómica según sus intereses, mediante la opciones de búsqueda del portal (zona o barrio, metro, tipo de oferta...).

- Registrado:

Mostrar información de su local.

Mostrar información acerca de los productos de su local mediante la publicación de ofertas.

Restricciones generales

Para acceder a la web no se requiere ningún hardware específico, solamente acceso a Internet y un navegador, ya que el servidor es el que se encarga de gestionar las transacciones sobre la base de datos y de interpretar el código PHP de las páginas.

Solo el administrador, tiene posibilidad de modificar cualquier dato de la web o la base de datos.

Los usuarios registrados sólo podrán modificar sus propios datos personales así como las ofertas que publiquen.

Supuestos y dependencias

Deberá haber un correcto funcionamiento de la base de datos y se deberán realizar copias de seguridad periódicamente.

3.2. REQUISITOS ESPECIFICOS

3.2.1 Requisitos de interfaces externas

Interfaces de usuario

La aplicación se visualizará en la pantalla del usuario, de forma sencilla e intuitiva.

Interfaces hardware

En nuestra aplicación no se necesita ningún interfaz hardware específico, puede ser visitada desde cualquier dispositivo que pueda conectarse a internet y navegar.

Interfaces software

La aplicación está desarrollada en PHP y JQuery, sobre Windows 7, mediante Dreamweaver CS6.

Su base de datos se basa en MySQL. De este modo puede funcionar sobre cualquier navegador y cualquier sistema operativo.

3.2.2 Requisitos funcionales

Almacenamiento y mantenimiento de los datos

Operaciones permitidas para usuario anónimo:

Solicitar información de la oferta.

Introducción: Se realiza una búsqueda según las preferencias del visitante.

Entradas: Búsqueda personalizada.

Proceso: Se envía una consulta para mostrar las ofertas.

Salida: Consulta del visitante.

Solicitar información del local.

Introducción: Visualizar ficha del local de la oferta seleccionada.

Entradas: Datos del perfil del usuario seleccionado.

Proceso: Consultar ficha del local.

Salida: Ficha del local.

Operaciones permitidas para el propietario del local:

Acceder a la intranet mediante su email.

Introducción: Identificarse mediante su email y contraseña.

Entradas: email y contraseña.

Proceso: Conectarse a la base de datos mediante su email para acceder a la intranet.

Salida: Intranet del propietario del local.

Modificar perfil.

Introducción: Modificar perfil del local registrado en la base de datos.

Entradas: Datos del perfil del usuario registrado.

Proceso: Modificado del perfil en la tabla registro de la base de datos.

Salida: Confirmación del modificado.

Crear Oferta.

Introducción: Insertar oferta en la base de datos.

Entradas: Datos de la oferta.

Proceso: Insertar oferta en la tabla ofertas de la base de datos.

Salida: Confirmación de la creación de la oferta.

Modificar Oferta.

Introducción: Modificar oferta en la base de datos.

Entradas: Datos de la oferta.

Proceso: Modificado de la oferta en la tabla ofertas de la base de datos.

Salida: Confirmación del modificado de la oferta.

Borrar Oferta.

Introducción: Borrar la oferta seleccionada.

Entradas: Datos la oferta.

Proceso: Eliminar la tupla de la oferta seleccionada.

Salida: Confirmación de borrado.

Seguridad

La aplicación no mostrará en ningún momento el email de sus usuarios para protegerlos de posibles fraudes con sus datos personales.

El local que desee utilizar los servicios de la Web tendrá identificarse dentro de la página con su email y clave.

El servidor protegerá todas las transacciones que se puedan realizar tanto del usuario hacia el servidor como el servidor al usuario.

Para proteger las contraseñas se utilizará md5, (Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5) un algoritmo de reducción criptográfico de 128 bits ampliamente usado para encriptar las contraseñas en la base de datos.

Mantenimiento

El mantenimiento se llevará a cabo por la persona asignada y autorizada previamente, con conocimientos sobre el sistema. Se realizarán las pertinentes copias de seguridad de la base de datos periódicamente.

Capítulo 4

Análisis

El siguiente apartado nos describe la fase de análisis del proyecto.

En esta etapa se analiza la aplicación que vamos a crear y se describe su estructura y funcionalidad mediante un diagrama UML, con lo que entender su funcionalidad nos será más sencillo.

Al final de esta fase tendremos un modelo con los diferentes actores que interactúan con los objetos del sistema a través de diversas relaciones y las acciones que cada uno de ellos podrá llevar a cabo.

4. UML

UML es un lenguaje de modelado que utilizaremos para describir nuestro sistema, para detallar y construir los métodos y atributos que estarán presentes en él.

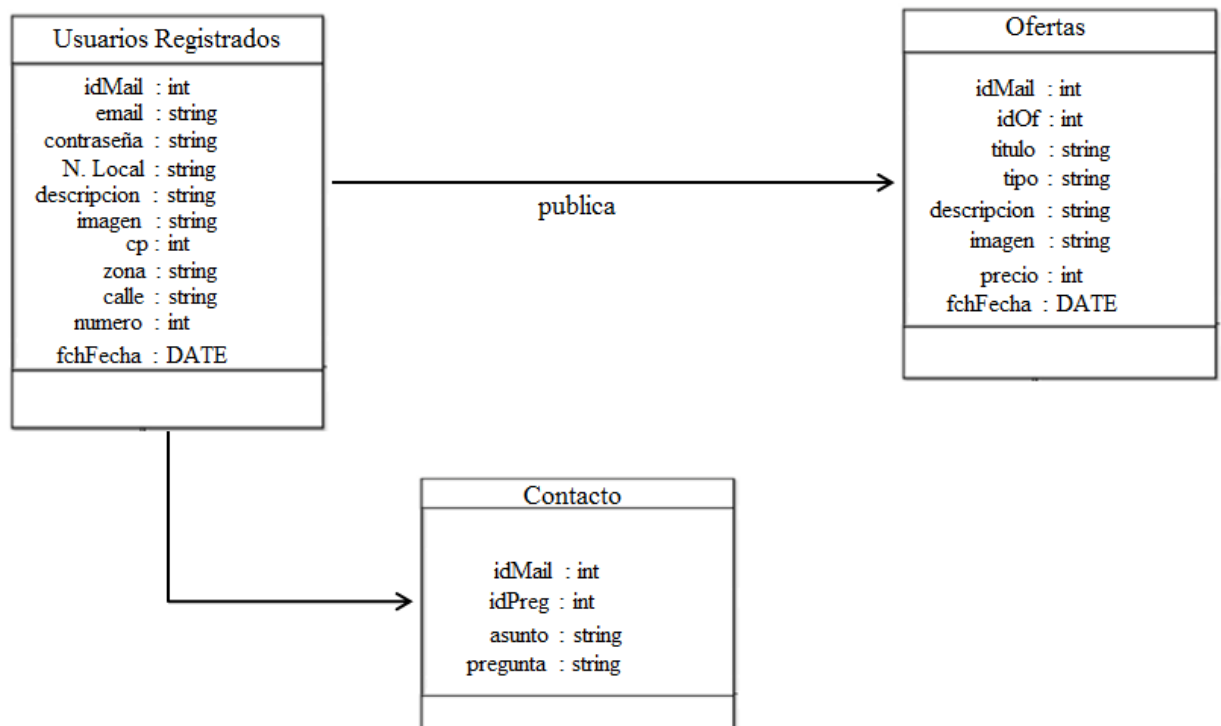
Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

Ofrece un estándar para describir un "plano" del sistema (modelo).

Realizaremos un diagrama para poder entender mejor nuestra aplicación.

4.1. Diagrama de clases

Un diagrama de clases describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro.



Un usuario se considerará anónimo mientras no se identifique en la aplicación.

Un usuario registrado será propietario del local.

De cada propietario conocemos sus datos personales, así como nombre, descripción y otros datos del local.

Un propietario pertenece a un distrito o zona. Cada zona o distrito se compondrá de los distintos locales que se encuentran registrados en esa área, agrupando de forma fácil, las ofertas de éstos locales.

4.2. Diagramas de casos de uso

Los diagramas de casos de uso describen el comportamiento de la aplicación, enfatizan lo que debe suceder en el sistema modelado.

En estos diagramas podemos observar varios actores, aquellos que podrán actuar con el sistema y se describe de este modo qué podrán hacer.

En nuestro sistema podemos distinguir varios actores:

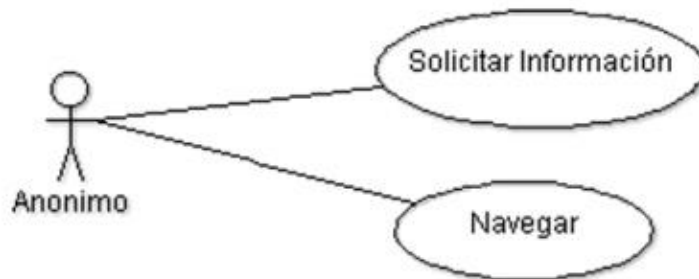
Anónimo es toda persona que accede a la aplicación sin identificarse (posibles Clientes), este actor podrá navegar por la web, pero no acceder a la intranet. Si podrá, en cambio, cualquier usuario registrado.

Usuario registrado es el propietario del local y accede a la aplicación para publicar y gestionar las ofertas que publica.

Administrador es aquel con acceso total, que puede modificar por completo la aplicación, es el que se encarga de su mantenimiento.

A continuación mostramos los casos de uso según cada actor.

4.2.1 Caso de uso: Anónimo

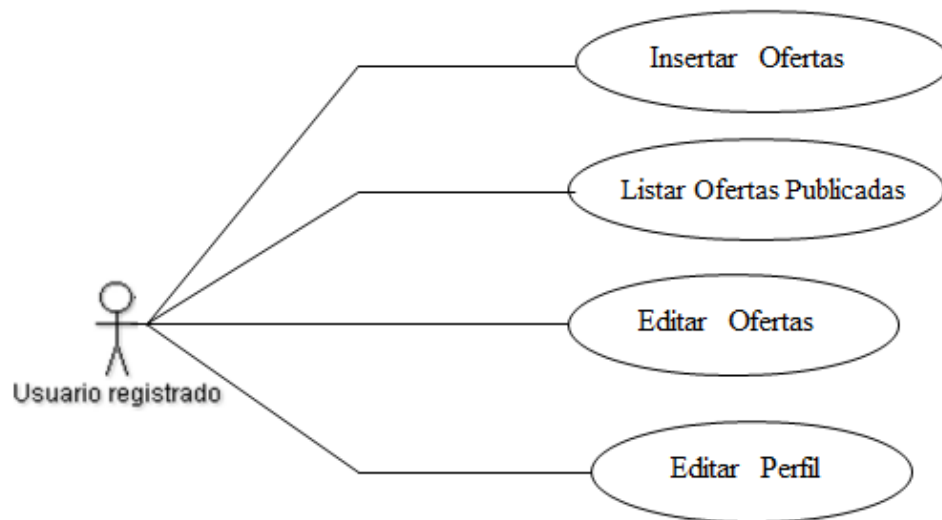


Se puede observar que el usuario anónimo solo puede realizar dos acciones en la aplicación, navegar por la web o solicitar información sobre la oferta o local.

4.2.2 Caso de uso: Usuario registrado

El usuario registrado debe identificarse mediante su usuario y contraseña para poder acceder a la intranet.





Cualquier usuario que se identifique podrá consultar sus datos personales, los de su local, así como listar y modificar las ofertas que publique.

4.2.3 Caso de uso: Administrador

El Administrador tiene permisos totales para modificar, añadir o borrar cualquier registro de la aplicación.

Capítulo 5

Diseño

En el capítulo 5 hablaremos sobre la arquitectura del sistema.

En ella encontramos tres niveles:

- Nivel de presentación.
- Nivel de aplicación.
- Nivel de persistencia.



5. ARQUITECTURA DEL SISTEMA

Nuestra aplicación está basado en el modelo Cliente / Servidor.

Los clientes realizan peticiones Servidor Web que le ofrece la respuesta, de este modo la capacidad de proceso está repartida entre los clientes y los servidores. Mediante esta arquitectura los accesos, recursos y la integridad de los datos son controlados por el servidor, de modo que un usuario registrado no puede acceder a aquellos datos a los que no esté autorizado.

La arquitectura de nuestra web estará estructurada de la siguiente manera:

- **Nivel de presentación:** Esta capa de nuestra arquitectura se encarga de la representación de la información para el usuario final, interactuando con él y comunicándose únicamente con el nivel de aplicación.
- **Nivel de aplicación:** Es donde se ubica el código de los programas, que se ejecutan, recibiendo las peticiones del usuario y enviándole las respuestas tras el proceso.
- **Nivel de persistencia:** En esta capa se encuentran los datos guardados en bases de datos y procesados por el nivel de aplicación.
A continuación se detallan cada uno de los niveles en nuestro proyecto.


5.1. Nivel de presentación

El nivel de Presentación está formado por todos los documentos que envía el servidor al cliente y que se muestran al usuario final de la aplicación.


En nuestra aplicación hay dos vistas parecidas pero diferentes de la web. La primera para usuarios en general, es decir todo aquel que visita el portal buscando ofertas (posible cliente) y la segunda, solo para registrados, los propietarios de local que utilizan la intranet para gestionar las ofertas que publican.

5.1.1 Interfaz gráfica


La primera vista de nuestra web se corresponde a la página de inicio. Ésta página es la que verán los posibles clientes al entrar a la web.




Gambas al ajillo
gambas al ajillo a precios populares. Especialidad mariscos
4 Euros - Mañana/Comida
EL Abuelo
Centro
Calle de la Victoria, 12 - 28012
[Más Información](#)




Caña y cerveza
Caña mas vino por 2 euros. Raciones y más.
2 Euros - Mañana/Comida
EL Abuelo
Centro
Calle de la Victoria, 12 - 28012
[Más Información](#)




Tapa gulas
Gratis al pedir tu cerveza
0 Euros - Todo el dia
EL Abuelo
Centro
Calle de la Victoria, 12 - 28012
[Más Información](#)



Tapas gratis
Consumiciones de mas de 15 e
tapas gratis
15 Euros - Mañana/Comida
bar yoaana
Latina
cava baja, 4 - 34534
[Más Información](#)



cubo mas bravas
cubo 12 botellines mas racion de bravas
5 Euros - Mañana/Comida
bar yoaana
Latina
cava baja, 4 - 34534
[Más Información](#)



mojitos
ven a cenar y gratis un mojito. También 2x1
4 Euros - Tarde/Cena
Bar Felix
Centro
Pez, 23 - 28008
[Más Información](#)

Se divide en los siguientes apartados:

1. En la parte superior, la búsqueda mediante una lista desplegable del criterio elegido.
2. En la parte izquierda de la web tenemos un pequeño menú de acceso. Con este menú podremos acceder a la página de inicio de la intranet o darnos de alta en el sistema.
3. En el cuerpo de la página, en la parte superior, el resultado de las búsqueda realizada. Al entrar en la web, aparecerán las ofertas publicadas. Más abajo aparece la galería de ofertas publicadas por los usuarios registrados, en ella veremos los datos y resultados de las consultas a la base de datos.
4. Si pulsamos el enlace “Más Información” el navegador nos redirigirá a la siguiente página relacionada con la oferta.

bar yoaana



cubo mas bravas

cubo 12 botellines mas racion de bravas

5 Euros

Mañana/Comida



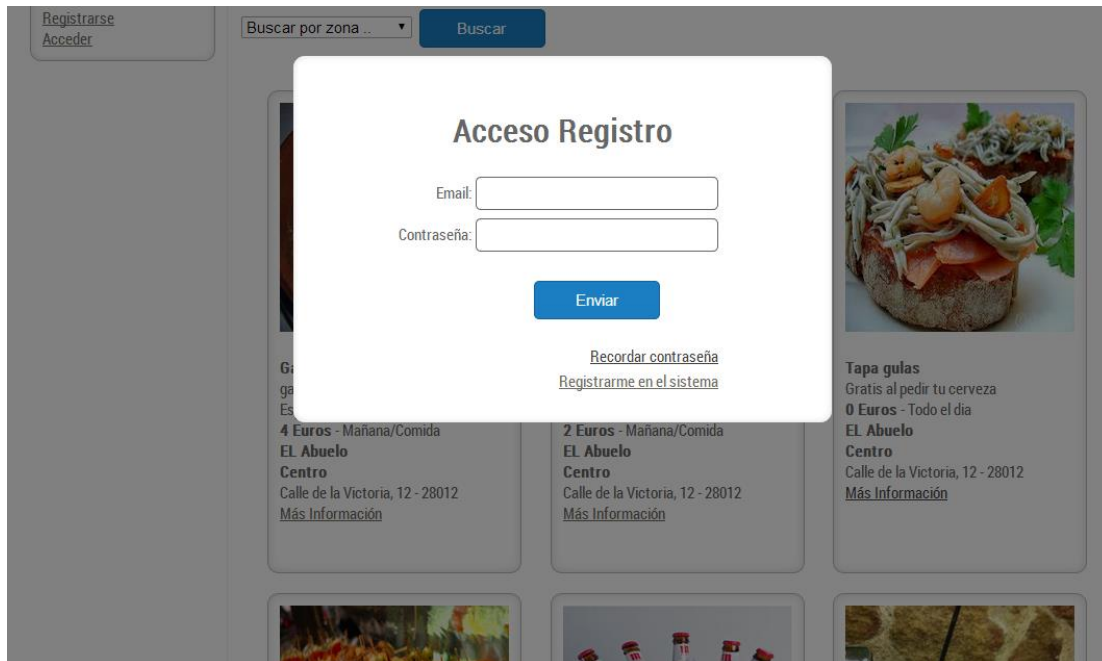
cava baja , 4 - 34534

Latina

Ven a nuestro bar - terraza. Tenemos todo tipo de raciones. Desde las 12 del mediodía hasta las 12 de la noche. Precios especiales a grupos

En ésta página hacemos una consulta para extraer información de dos tablas, la del perfil donde el propietario se registró y donde publicó la oferta.

Para acceder a la intranet se ha creado una página de login con css y javascript.



```
#boira {
  left:0;
  top:0;
  z-index:1;
  position: fixed;
  display: none;
  margin: auto;
  width: 100%;
  height: 100%;
  color: white;
  background: black;
  filter: alpha(opacity=50); opacity: 0.5; -moz-opacity: 0.50
}
```

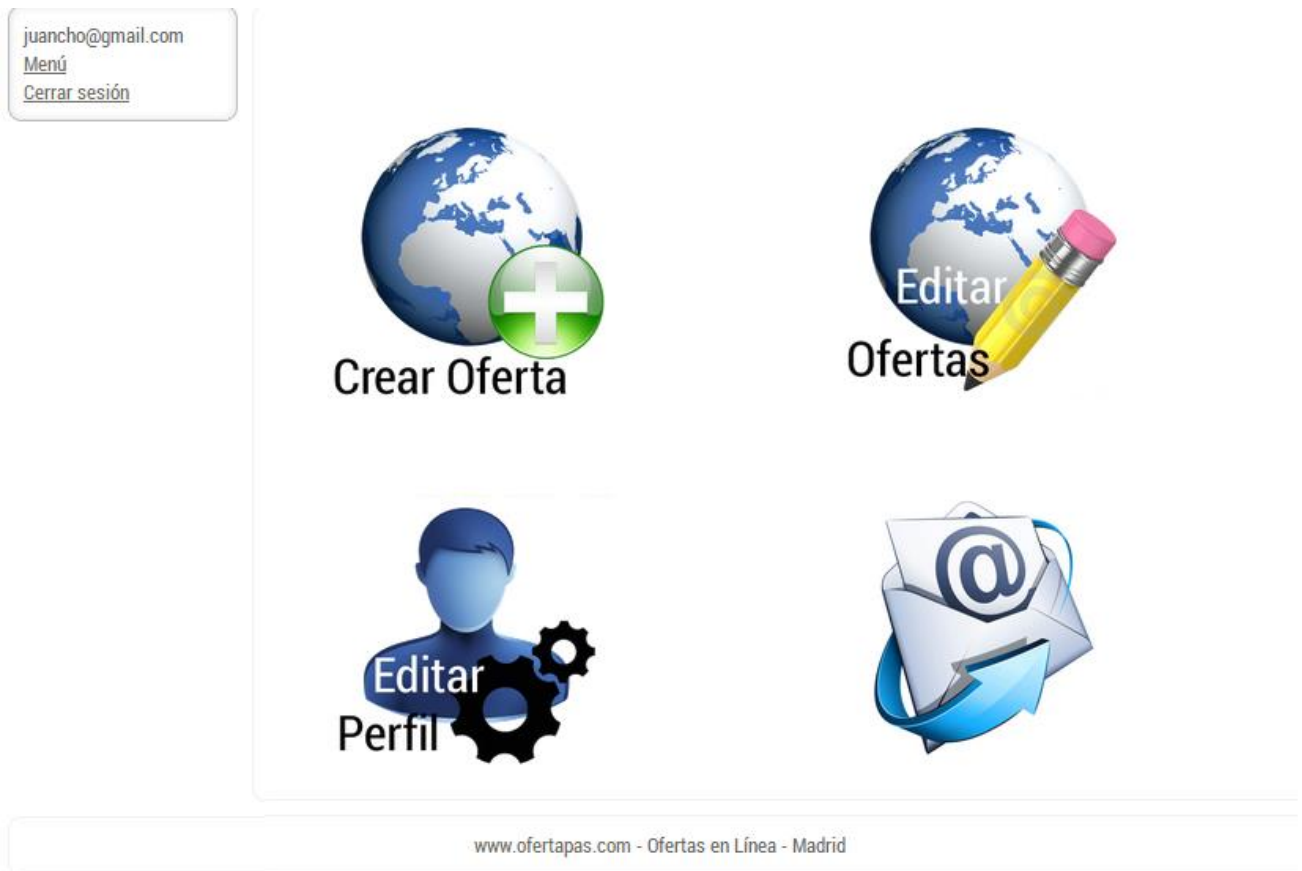
Detalle código Css que crea la capa para oscurecer la pantalla.

```
.veureDia {
  width:316px;
  height:283px;
  color: #5d5d5d;
}
```

Detalle código Css ventana flotante donde carga login.php.

Esta es la parte gráfica Css que irá combinado con código JavaScript más complejo.

La segunda vista se corresponde a la intranet, a la que accederán solo los usuarios registrados.



2. El menú de sesión se encuentra a la izquierda de la pantalla, con él podremos acceder al menú de la intranet, ver el email que ha iniciado sesión y cerrar sesión.

3. En el cuerpo de la página veremos las diferentes opciones que puede realizar el local:

- Insertar-Crear ofertas.

Insertar Oferta

Título de la oferta:

Tipo Oferta:

Descripción:

Imagen:

Precio:

- Ver / Editar las ofertas publicadas.

[Volver](#)



Tapas gratis

Consumiciones de mas de 15 e
tapas gratis

15 Euros - Mañana/Comida

bar yoanaa -

Latina

cava baja, 4 34534

[Editar](#)



cubo mas bravas

cubo 12 botellines mas racion de
bravas

5 Euros - Mañana/Comida

bar yoanaa -

Latina

cava baja, 4 34534

[Editar](#)



2 bocadillos calamares

sdfdsf

5 Euros - Tarde/Cena

bar yoanaa -

Latina

cava baja, 4 34534

[Editar](#)



Editar Oferta

Título de la oferta:

Tipo Oferta:

Descripción:

Imagen:

[Subir Imagen](#)

Precio:

[Editar Oferta](#)

- Editar perfil del usuario y local.

Modificar Usuario

Email:

Contraseña:

Repetir contraseña:

Nombre Local:

Descripción:

Imagen:

Código Postal:

Zona:

Calle:

Número:

- Contactar con el administrador.

Contacto

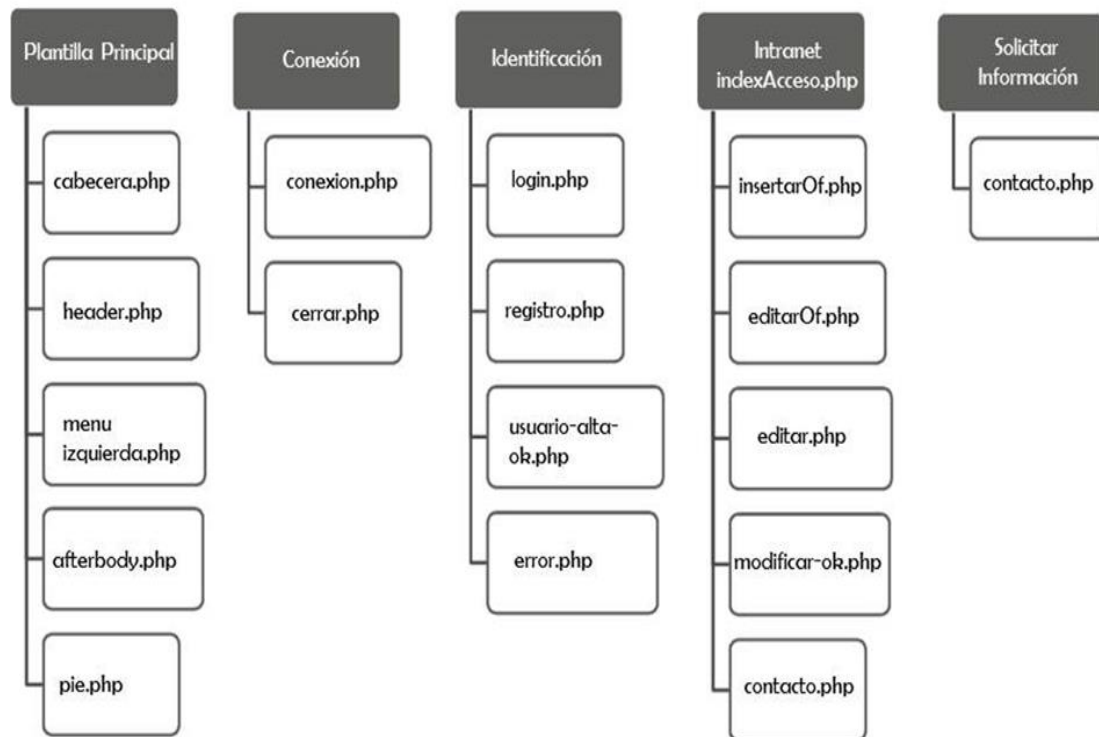
Usuario:

Asunto:

Pregunta:

5.2 Nivel de aplicación

Con esta capa se muestran las diferentes acciones que se pueden desarrollar en el sistema, asociando a cada una, los ficheros que contienen las funciones que deben utilizar.



5.3 Nivel de persistencia

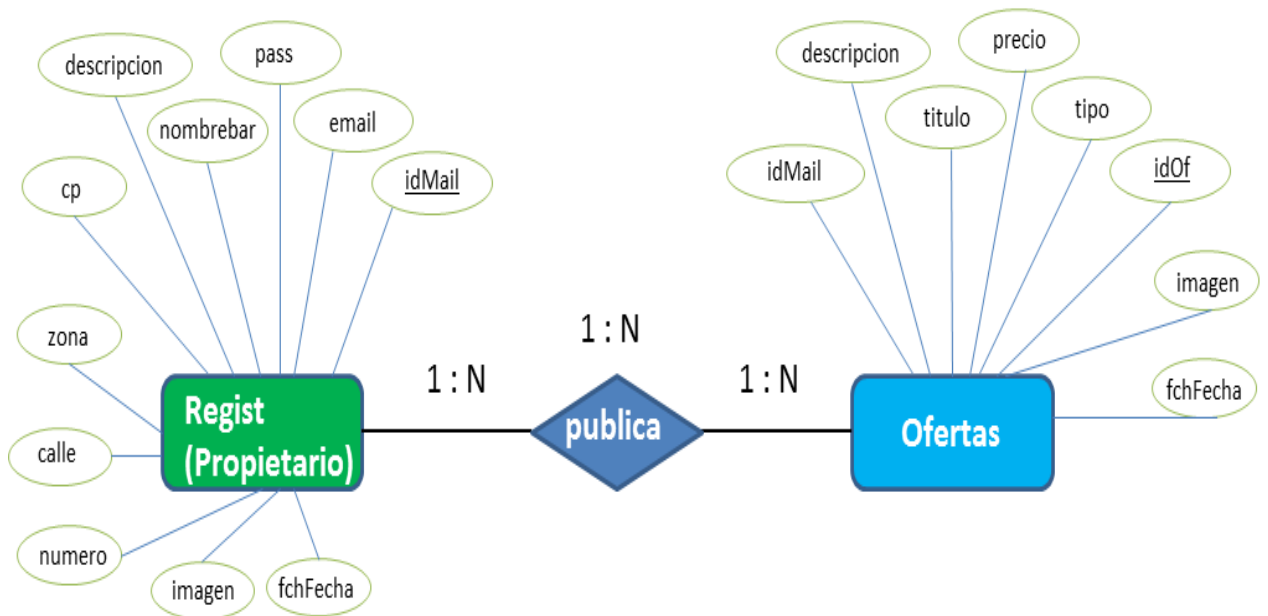
Es en esta capa donde los datos de nuestra aplicación están almacenados.

Base de datos

Para nuestro proyecto hemos creado una base de datos llamada “tapasdb” que contendrá en cada una de sus tablas los datos tanto de registro y ofertas, como las relaciones entre ellos.

Se muestra a continuación el diagrama entidad-relación que describe las relaciones entre los diferentes usuarios y objetos del sistema y las restricciones entre ellos.

Diagrama Entidad-Relación



Capítulo 6 Evaluación y pruebas

Para evaluar el correcto funcionamiento del producto se han realizado una serie de pruebas donde se comprueba la navegabilidad en diferentes navegadores y resoluciones.

Una vez acabado el diseño final se realizarán pruebas de enlaces rotos, html, css y de compatibilidad para el móvil en <http://www.w3.org>.

También he realizado fichas con algunos de los casos de uso.

6.1 Prueba de enlaces rotos

Realizaremos esta prueba para la comprobar que no haya ningún enlace todo en nuestra aplicación.

Para ello utilizamos la herramienta que nos ofrece

<http://validator.w3.org/checklink>.

Como ejemplo pondremos la validación de los enlaces en la página principal:

6.2 Validación de html

Se ha comprobará que el html cumple los estándares, mediante el validador

<http://validator.w3.org>

6.3 Validación de css

Igual que el código html, las hojas de estilo Css, pueden ser validadas mediante

<http://jigsaw.w3.org/css-validator>

6.4 Prueba de compatibilidad para móvil

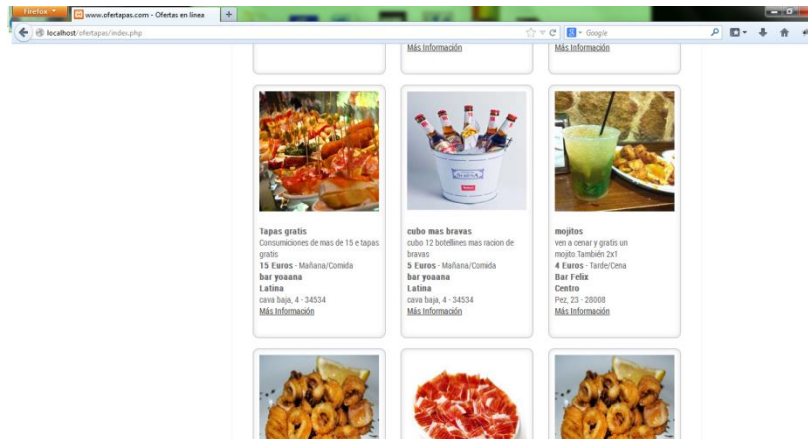
Comprobaremos también si nuestro portal sería visible a través de un terminal móvil mediante

<http://validator.w3.org/mobile>

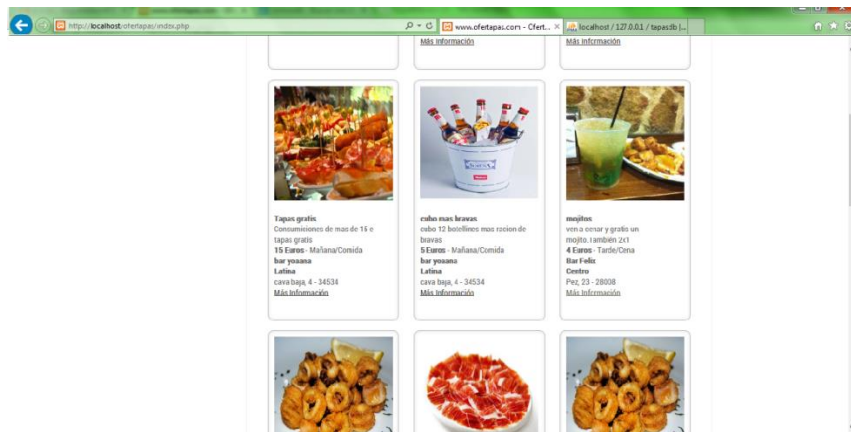
6.5 Compatibilidad con los navegadores

Comprobamos la compatibilidad de nuestra web en diferentes navegadores para asegurarnos de que cualquier usuario podrá entrar correctamente.

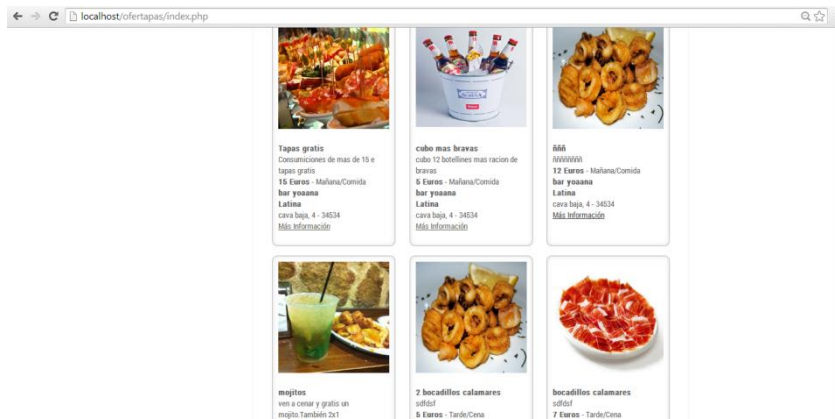
6.5.1 Mozilla Firefox



6.5.2 Internet Explorer



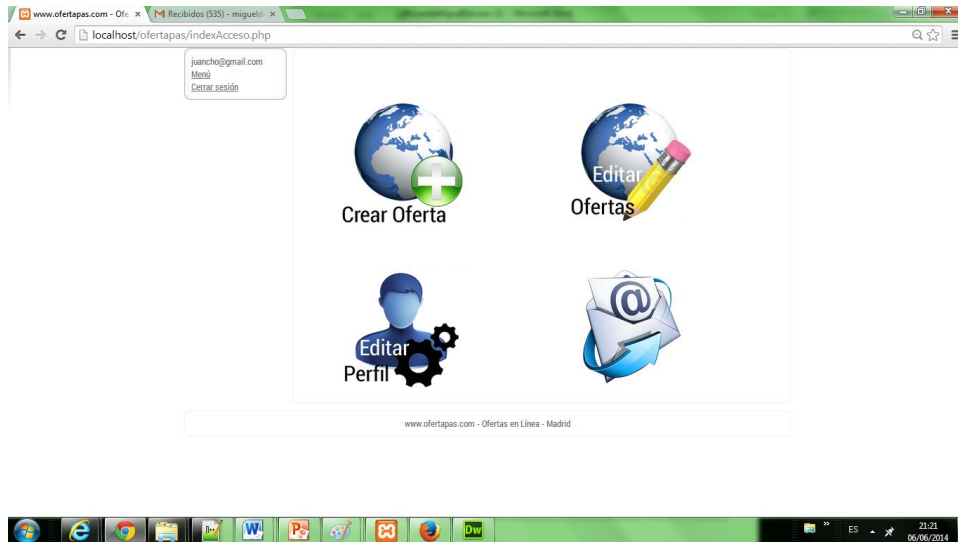
6.5.3 Google Chrome



6.6 Compatibilidad con diferentes resoluciones

Por último comprobamos si nuestra web es visiblemente compatible con diferentes resoluciones de pantalla, para asegurarnos de que se vea correctamente.

Resolución 1366 x 748 ppp.



Resolución 800 x 600 ppp.



6.7. Evaluación Casos de uso

Caso de uso	Solicitar Información	
Actores	Todos los usuarios	
Propósito	Solicitar información de las ofertas publicadas	
Resumen	Los visitantes de la web podrán buscar ofertas según sus preferencias de búsqueda.	
Precondiciones	Visitar la página	
Pos condiciones	El sistema mostrará las ofertas que coincidan con el campo seleccionado	
Flujo de eventos	Interacción con el usuario	Obligaciones del sistema
1. El visitante pulsa sobre la lista desplegable sobre el campo que quiere realizar la búsqueda. 2. Selecciona según sus preferencias la opción de la lista desplegable. 3. Pulsa el botón "Buscar"		El sistema mostrará una búsqueda donde coincida el campo seleccionado.
Extensiones sincronas	Si no se selecciona ningún campo de la lista desplegable, se mostrarán las ofertas según fecha de publicación.	

Caso de uso	Acceder a la intranet	
Actores	Cualquier usuario con clave	
Propósito	Acceder a la zona restringida de la web	
Resumen	Cualquier usuario que sea registrado puede intentar acceder a su zona privada mediante su email y contraseña	
Precondiciones	Ser usuario registrado	
Pos condiciones	El usuario accederá a la intranet	
Flujo de eventos	Interacción con el usuario	Obligaciones del sistema
	1. El usuario pulsa el enlace "Acceder". 2. Rellena los campos "Email" y "Contraseña". 3. Pulsa el botón "Entrar".	4. El sistema comprueba que los datos introducidos coinciden con la base de datos y redirige el navegador a la intranet.
Extensiones sincronas	Si no se rellena algún dato obligatorio el sistema lo comprobará y mostrará el error	

Caso de uso	Editar Perfil	
Actores	Usuario registrado	
Propósito	Consultar datos del perfil del local y poder modificarlos	
Resumen	Consultar y modificar los datos que rellenamos al registrarnos. Datos de nuestro local y contraseña de la cuenta	
Precondiciones	Ser usuario registrado y estar en la intranet	
Pos condiciones	El usuario ve sus datos en campos editables	
Flujo de eventos	Interacción con el usuario	Obligaciones del sistema
	1. El usuario pulsa el botón "Editar Perfil". 2. El sistema redirige el navegador a la página adecuada.	3. Se mostrará una página parecida a la del registro pero con los datos que rellenamos. 4. Al pulsar "Modificar" el sistema reescribirá los datos.
Extensiones sincronas	Si la sesión ha terminado, el sistema no permitirá que los datos se vean.	

Caso de uso	Insertar Oferta	
Actores	Usuario registrado	
Propósito	Crear una oferta de un producto de nuestro local	
Resumen	Se accede a la página adecuada y se rellenan los datos sobre la oferta	
Precondiciones	Ser usuario registrado y estar en la intranet	
Pos condiciones	La oferta estará publicada en la página principal del portal.	
Flujo de eventos	Interacción con el usuario	Obligaciones del sistema
	1. El usuario pulsa el botón "Crear Oferta". 2. El sistema redirige el navegador a la página adecuada.	3. Se rellena el formulario para Crear la oferta y se pulsa el botón "Insertar Oferta". 4. Se crea un registro en la tabla ofertas y ya podrá ser vista por los visitantes de la web.
Extensiones sincronas	Si la sesión ha terminado, el sistema no permitirá que los datos se vean.	

PARTE IV

Conclusiones

Capítulo 7

Conclusiones

Después de acabar el desarrollo de la web, y teniendo un margen de mejora bastante amplio, las conclusiones finales son muy positivas, ya que se ha logrado realizar la idea inicial al completo, una web dinámica con un primer diseño ordenado, en que consultar ofertas o publicirlas es muy sencillo y visual.

Se ha completado todo el proceso del sistema planteado como el registro e inicio de sesión así como las funciones de la intranet (publicar, editar ofertas y editar el perfil del local y contraseña del usuario).

Todavía no se ha hecho lo más difícil, que es publicitar la página web y que sea un portal de consulta habitual entre los usuarios de internet. Partimos con la ventaja de que Madrid es una ciudad donde hay un importante número de bares y restaurantes que recibe un gran número de turistas.

Emprender una idea no es nada fácil pero sí un proyecto tiene buena base se podrá convertir en uno de los pilares fundamentales para seguir desarrollándola.

Creo que es una idea con cierto carácter original, que busca ayudar al propietario y no beneficiarse de él.

Se publicitaría por el centro de Madrid con folletos informativos para los locales y otro para los turistas y posibles visitantes de la página.

Una forma de costear este gasto sería añadiendo publicidad en forma de pop – up según el distrito que busquen o debajo del menú de la izquierda.

7.1. APORTACIONES ORIGINALES

- Se ha mejorado el dinamismo de la web utilizando tecnologías no vistas en clase, como AJAX, utilizado dentro del registro en la validación del email para poder registrarse en el sistema, indicando dinámicamente si el email de registro está ya en la base de datos sin necesidad de recargar toda la página.

Email:

✓ E-mail correcto para el alta.

otir Email:

Los visitantes también realizan las búsquedas de las ofertas sin que se recargue la página.

- La validación de todos los campos del formulario, (registro e insertar) han de estar rellenos, de lo contrario el sistema no efectuará ninguna consulta a la base de datos y aparecerá un mensaje debajo del campo que no se haya rellenado. Este mensaje aparece utilizando Css y es aplicado al resto de páginas utilizando las reglas de cascada y herencia. También otros efectos visuales son gracias a Css, como las tablas redondeadas con sombra donde van las ofertas o la fuente de la letra.

Nombre Local:

✗ Debes escribir un nombre del local.

En cuanto a la originalidad de la idea, Ofertapas al contrario que el resto de páginas que ofrecen un servicio parecido, no sacará ningún beneficio de las ofertas que publiquen sus propietarios dejando así más margen de beneficio para ellos.

7.2. TRABAJOS RELACIONADOS

Se han utilizado varios documentos de asignaturas de clase tanto del primer como del segundo curso, para el comienzo del desarrollo del proyecto además de pruebas y formas de desarrollar las ideas planteadas al inicio de proyecto.

1er Curso.

- Lenguaje de marcas. Teoría. JavaScript.
- Entornos de desarrollo. Pruebas y planteamiento del proyecto.
- Programación. Conceptos de la programación orientada a objetos.
- Bases de datos. Aprendizaje de la realización de tablas para el modelo E.R y sus restricciones.

2do Curso.

- Sistemas de gestión empresarial. Teoría y ejercicios del módulo de php así como la conexión al servidor.
- Acceso a bases de datos. Conexión de nuestro sistema y el servidor.

7.3. LINEAS DE INVESTIGACION FUTURAS

- Subir más imágenes del local.

Estas imágenes se mostrarían junto con la que ya ha subido el usuario en el apartado “Más información” al pulsar sobre la oferta seleccionada. Mejorando así el apartado “Más información” de la oferta.

- Contenido de la página web y de las ofertas también en inglés.

Esta mejora se solucionaría de la siguiente manera:

Al entrar en la página, código para detectar el lenguaje del navegador.

Una vez detectado, si no es español, redirigirá al dominio www.ofertapas.com.

Los usuarios registrados, al insertar o al editar ofertas tendrán la opción para rellenar la oferta en inglés, mostrándose campos paralelos para este idioma.

- Control de las ofertas y usuarios desde la web.

La base de datos, en la tabla de los usuarios, hay un campo denominado “nivel” para distinguir si está registrado o no. El administrador tendrá un nivel distinto al de los usuarios registrados. Con esto se consigue que pueda aparecer sólo al administrador lo que indiquemos. Por ejemplo:

Tabla con listado e información de usuarios registrados y ofertas con permisos para borrar registro de la base datos.

- Mostrar mapa de los locales.

Al pulsar “Más Información” en las ofertas, se dirige a una página que muestra información sobre la oferta seleccionada y del local. La mejora sería que aparezca también un mapa de donde se encuentra el local.



El problema que plantea esta mejora es convertir en coordenadas, la dirección que introduce el usuario registrado de su local.

```
modo de la API  versión  inicio automático
<script type="text/javascript">
  google.load('maps', '2', {callback:simple});var map;
  function simple(){
    if (GBrowserIsCompatible()) {
      var map = new GMap2(document.getElementById("map"));
      map.setCenter(new GLatLng(20.0972, -81.6503), 4);}
  }
</script>
<div id="map" style="width:360px;height:200px;"></div>

          latitud  longitud  zoom
          ancho  altura
```

-Geolocalización con HTML 5.

- Compartir oferta en redes sociales.

Bibliografía

La principal fuente de consulta utilizada ha sido internet. A continuación se describen los enlaces que más importancia han tenido dentro de este proyecto.

PHP YA [web tutorial de php].[Abril 2014] .

Disponible en: <http://www.phpya.com.ar>

AJAX YA [web tutorial de ajax].[Finales de Abri, 2014] .

Disponible en: <http://www.ajaxya.com.ar>

Wikipedia [web enciclopedia libre].[Durante todo el proyecto] .

Disponible en: <http://es.wikipedia.org>

AJPDSOFT (foro programación)[web foro].[Abril, 2014] .

Disponible en:

<http://www.ajpdsoft.com/modules.php?name=News&file=article&sid=488>

Validar formularios css (web Daniel Martínez)[web blog].[Abril, 2014] .

Disponible en:

<http://coolvillage.es/ayuda-al-usuario-a-rellenar-formularios-con-css>

jQuery cargar Css con Ajax(web Dean Williams)[web blog].[Abril, 2014] .

Disponible en:

<http://deano.me/2012/09/jquery-load-css-with-ajax-all-browsers>

Ver detalle php(forosdelweb.com)[foro sobre programación web].[Abril, 2014] .

Disponible en:

<http://www.forosdelweb.com/f18/como-crear-pagina-detalle-php-925191>

PARTE V

Anexos

ANEXO 1: Implementación

Algunas partes de código que me parecen más interesantes son:

1. Plantilla.php

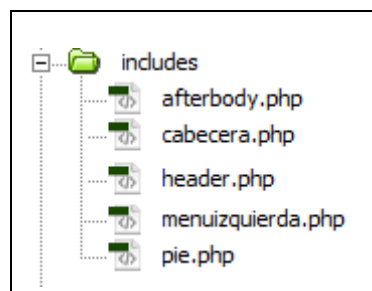
De ésta plantilla nace el resto de páginas principales del portal. En ella aparecen los includes que es un elemento del lenguaje PHP que nos permite incluir una página entera dentro de otra.

```
<!DOCTYPE html>
<html lang="es">
<head>
<link href='http://fonts.googleapis.com/css?family=Roboto+Condensed:400,300'
rel='stylesheet' type='text/css'>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<!-- TemplateBeginEditable name="doctitle" -->
<title>www.ofertapas.com - Ofertas en línea</title>
<!-- TemplateEndEditable -->
<!-- TemplateBeginEditable name="head" -->
<!-- TemplateEndEditable -->
<link href="../css/principal.css" rel="stylesheet" type="text/css" />
<?php include("includes/header.php"); ?>
</head>
<body>
<?php include("includes/afterbody.php"); ?>

<div class="container">
  <div class="header"><?php include("includes/cabecera.php"); ?></div>
  <div class="sidebar1"><?php include("includes/menuizquierda.php"); ?>
</div>

  <div class="content"><!-- TemplateBeginEditable name="Contenido" -->dddd<!--
TemplateEndEditable --></div>
  <div class="footer">
    <?php include("includes/pie.php"); ?>
  <!-- end .footer --></div>
<!-- end .container --></div>
</body>
</html>
```

Editando una página del include modificará el resto de documentos php.



2. Comunicación entre Ver Ofertas publicadas y Editar oferta.

Vista de la oferta en la página VerOf.php , que muestra las ofertas publicadas por el propietario. Se puede ver un enlace al final de cada oferta llamado “Editar”.



Al hacer clic sobre “Editar” el navegador nos redirecciona al siguiente enlace en el que se puede ver un número al final. Este número corresponde al id de la tabla oferta (autoincremental).

```
localhost/ofertapas/editarOf.php?id=26
```

```
<a href = editarOf.php?id=<?php echo $row_DatosProductos['idOfer'];?>><br>
```

A este enlace se pasa una variable que contiene el id de la oferta.

En la otra página, editarOf.php (editar oferta) , se coge la variable “ id” que contiene el id de la oferta. A continuación se hace una consulta pasándole el id de la oferta para mostrar los campos que rellenamos al crear esa oferta.

```
$id=$_GET['id'];

mysql_select_db($database_conexion, $conexion);
$query_DatosCategorias = "SELECT ofertas.idOfer, ofertas.titulo, ofertas.tipo,
ofertas.descripcion, ofertas.imagen, ofertas.precio, regist.idMail FROM
ofertas, regist WHERE ofertas.idOfer = '$id'";
```

La página que muestra es la siguiente:

Editar Oferta

Título de la oferta:

Tipo Oferta:

Descripción:

Imagen:

Precio:

www.ofertapas.com - Ofertas en Línea - Madrid

3. Validar dinámicamente si el email de registro existe en la base de datos.

Registro de Usuario

Email:

☒ E-mail correcto para el alta.

Repetir Email:

Contraseña:

Este código corresponde a la página registro.php, en él se valida de forma dinámica si el email introducido existe o no en la base de datos. El mensaje es cargado dinámicamente con Css y Ajax.

Se llama a la función “controlaremailunico()” cuando el ratón sale del campo Email. Una vez dentro de la función se coge el valor del campo email y se envía por POST la variable “emailinsertado” del campo email de la consulta realizada en funcionesajax.php, (formid = 3).

A continuación llamados a “resp” que será la respuesta de la consulta formid = 3.

Si es 1, el email no se encuentra en la base de datos.

Mostrará un aviso Css + jQuery.

```
<script type="text/javascript">

function controlaremailunico()
{
    var emailinsertado = document.formregistro.email.value;
    $.ajax({
        type: "POST",
        url:"includes/funciones-ajax.php",
        data: 'email='+emailinsertado+'&formid=3',
        success: function(resp)
        {
            //El email no se encuentra en la BD, el alta se puede dar.
            if (resp==1)
            {
                $("#aviso8").hide("fast");
                $("#exitol").show("slow");
                document.formregistro.botonAlta.disabled = false;
            }
            //El email se encuentra en la BD, no podemos dejar que se de de
            if (resp==0)
            {
                $("#exitol").hide("fast");
                $("#aviso8").show("slow");
                document.formregistro.botonAlta.disabled = true;
            }
        }
    });
}

</script>
```

Este código corresponde a la consulta a la base de datos de funcionesajax.php que es llamado desde la función controlaremailunico () en la página registro.php.

```
if ($_POST["formid"]==3)
{
    mysql_select_db($database_conexion, $conexion);
    $query_DatosFrecuentes = "SELECT * FROM regist WHERE email = '".utf8_decode
($_POST['email'])."'";
    $DatosFrecuentes = mysql_query($query_DatosFrecuentes, $conexion) or die(
mysql_error());
    $row_DatosFrecuentes = mysql_fetch_assoc($DatosFrecuentes);
    $totalRows_DatosFrecuentes = mysql_num_rows($DatosFrecuentes);

    if ($totalRows_DatosFrecuentes>0) echo "0";
    else echo "1";
}

?>
```

