|  |
| --- |
| PROYECTO FIN DE GRADO: SEARCH & FIND |
| TRABAJO REALIZADO POR:  *JAIRO GARCIA APARICIO*  *VICTOR RUIZ MARTINEZ* |
|  |
|  |

**ÍNDICE GENERAL**

**1.** Introducción . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2

**2.** Planificación previa . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

**3.** Análisis . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

3.1. Requisitos generales . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

3.2. Casos de uso . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

**4.** Diseño . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

4.1. Arquitectura . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

4.1.1 Capa de presentación . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

4.1.2 Capa de dominio o aplicación . . . . . . . . . . . . . . . . . . . . . . . 12

4.1.3 Capa de datos . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18

4.2. Interfaz gáfica . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25

4.2.1 Usabilidad . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 26

**5.** Pruebas . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 27

**6.** Despliegue . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 29

6.1. Operaciones . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 29

6.2. Mantenimiento . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 30

**7.** Bibliografía . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 31

**1. Introducción**

Este documento describe el trabajo realizado en el proyecto final de desarrollo de aplicaciones web. El proyecto consiste en el desarrollo de un sitio web para alquilar y publicar apartamentos.

El proyecto está enfocado para que sea accesible desde navegadores de Internet pero en ordenadores y no para que sea accesible desde un móvil o tablet.

El sitio web está dedicado para gente mayor de edad, la cual esté en disposición de un apartamento el cual quiere ser alquilado. También está dedicado para la gente que busca apartamento para alquilarlo, ya sea por vacaciones, como por un tiempo más prolongado.

En cuanto a los usuarios, la página tiene varios tipos:

- Por un lado, están los usuarios no registrados, que únicamente pueden acceder a información general sobre el apartamento publicado y contactar con el vendedor a través de un número de teléfono o un correo electrónico.

- Por otro, están los usuarios registrados, que pueden acceder a una mayor funcionalidad dentro de la aplicación, donde pueden incluir un apartamento en alquiler con todos los datos que quiera aportar para que su apartamento quede expuesto en el sitio web de un modo gratuito. Además, también podrá acceder a un apartado donde se muestran todos los apartamentos del usuario que ha iniciado sesión y podrá eliminarlos cuando desee.

El proyecto ha sido realizado por Victor Ruiz Martinez y Jairo Garcia Aparicio, para el instituto I.E.S Melchor Gaspar de Jovellanos en la localidad de Fuenlabrada. La idea de crear este sitio web fue una propuesta común. Habíamos realizado varios ejercicios en clase y queríamos ampliar esos ejercicios haciendo algo más complejo y que tuviera diferentes partes de lo que hemos aprendido durante este curso. La finalidad de ello es que sea útil, sencilla y sobre todo fácil de manejar.

Licencia para la aplicación web:

Para esta aplicación web se hará uso de una licencia Creative Commons. Para ello, primeramente se ha de acceder a la página web oficial de Creative Commons (http://creativecommons.org/) y rellenar el formulario.

Sus principales características son:

* Las licencias Creative Commons sirven para expresar los usos que permite el autor sobre su obra a los demás usuarios, no los usos que se dispone a hacer de su obra.
* Estas licencias se centran exclusivamente en el ejercicio de los derechos de explotación.
* Solo el autor puede decidir el uso de estas licencias.
* El autor ha de concentrar y mantener la titularidad en exclusiva de los derechos de explotación.
* Antes de seleccionar la jurisdicción de la licencia debemos asegurarnos que efectivamente como autores se nos aplica la ley de Propiedad Intelectual de la jurisdicción en cuestión.
* El uso de estas licencias no supone ningún tipo de registro de la Propiedad Intelectual.
* Se trata de las licencias Copyleft más desarrolladas y sofisticadas. El Copyleft permite ofrecer la posibilidad de copiar, difundir, modificar o incluso hacer un uso comercial de nuestra obra a un público genérico sin renunciar a ningún tipo de protección legal por ello.

**2. Planificación previa**

La planificación previa fue uno de los problemas que más nos ha causado problemas en el proyecto. Empezamos el proyecto componiendolo por tres personas, decidimos hacer reuniones periódicas, para dividir el trabajo y para ver el avance del proyecto. La idea de lo que queríamos crear estaba clara, pero a la hora de empezar a realizar el proyecto , empezamos un poco más tarde de lo previsto por distintas circunstancias.

Empezamos a realizar el proyecto de manera conjunta, en las reuniones previstas con mi compañero Jairo y Jose Luis. Este último tuvo algunos problemas para acudir a las reuniones planificadas. Empezamos por crear la base de datos con una estructura básica en MySQL Workbench.

Realizamos la página principal de manera conjunta, pero con pocas funcionalidades. Pasado un tiempo, veíamos que íbamos retrasados con el proyecto así que decidimos dividir el trabajo.

Jairo realizó el Login y el Register, es decir el manejo de usuarios con el uso de sesiones. También se propuso el realizar la parte visible de la página como colores, botones…

Victor realizó el Publish Free que consiste en incluir un apartamento. También cree la lógica del botón contactar para unir los id de los apartamentos con los usuarios y comente el código.

Jose Luis empezo a realizar un puto de la memoria y una pagina de contacto pero cuando la probamos en su momento no funcionaba.

Tuvimos problemas con nuestro compañero Jose Luis por ciertos motivos y dejó de formar parte de nuestro proyecto. Esto nos afectó bastante porque estuvimos esperando mucho tiempo al trabajo de un compañero y ahora teníamos que recuperar nosotros dos solos.

La mayoría de las cosas se nos han ido ocurriendo a medida que íbamos avanzando el proyecto a Jairo y a mi. Por ejemplo, yo di la idea de un botón perfil donde se imprimieran todos los apartamentos del usuario que estuviera logueado. A Jairo se le ocurrió el botón borrar , para eliminar un apartamento.

A partir de nuestro problema, realizamos más sesiones de trabajo en común y por separado para avanzar todo lo posible y presentar el proyecto lo mejor posible.

**3. Análisis**

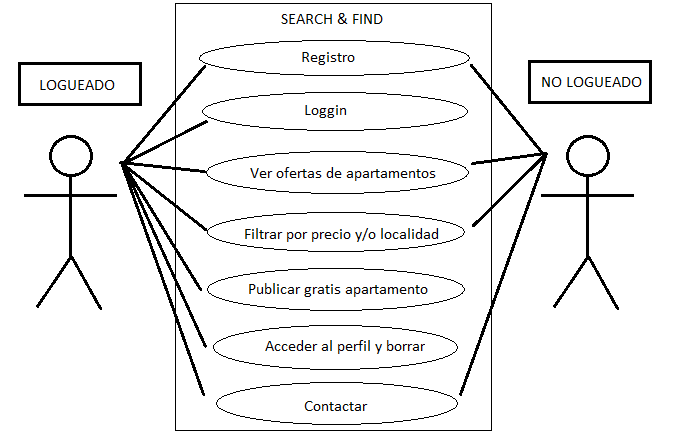
Después de haber planificado el Proyecto. Procederemos a explicar los requisitos generales y los casos de uso que puede haber en nuestro sitio web. El objetivo de los casos de uso es apreciar el funcionamiento de la pagina segun los roles y distintos casos de uso.

**3.1. Requisitos generales**

1. Rápida velocidad de carga
2. Acceso a internet
3. Cualquier tipo de explorador
4. Cualquier Sistema Operativo (Preferible Windows 10).
5. Caracteristicas para instalar Windows 10:

* Procesador: Procesador a 1 GHz o más rápido o sistema en un chip (SoC)
* RAM: 1 GB para 32 bits o 2 GB para 64 bits
* Espacio en disco duro: 16 GB para un SO de 32 bits o 32 GB para un SO de 64 bits

**3.2. Casos de uso**



* Cuando se ingresa por primera vez en la página web, aparecen todos los apartamentos que están en disposición de ser alquilados, cada uno de ellos incluye un botón para contactar con el vendedor. Aparte en la zona central aparecerá un filtrado donde se pueden filtrar los apartamentos por precio , por localidad o por ambos a la vez. El usuario no está registrado por lo que aparece un botón donde se puede registrar de forma gratuita.
* Si el usuario desea registrarse para acceder a una funcionalidad mayor se deberá acceder al botón Log-in en la esquina superior derecha de la página principal. Una vez accedido, se redirige a una nueva página donde se puede loguear si ya estás registrado con un email y la contraseña. Si accedes correctamente con las credenciales, serás redirigido automáticamente a la página principal y si no las introduces bien aparecerá un mensaje donde dice que las credenciales no son correctas. Por otro lado aparece un apartado llamado Register para registrarse si lo desea.
* Si accedemos a la parte de Register, nos llevará a una ventana nueva donde puedes registrarte gratuitamente. Deberás introducir un email, una contraseña y repetirla para que esté correctamente escrita y un número de teléfono de contacto. Una vez dando al botón crear se muestra un mensaje el cual dice que el usuario está correctamente registrado. Se permanece en la misma ventana pero aparece un link llamado Login el cual habrá que acceder para ser logueado.
* Una vez logueado en la página del Login, como dije antes será redirigido automáticamente a la página principal pero esta incluye varias funcionalidades al estar logueado.

-Se incluye un mensaje el cual muestra que estas iniciado la sesión como el usuario llamado como el nombre del email y al lado un botón Logout en el cual si se le da click la sesión del usuario se cierra y la página principal vuelve al estado como si no hubiera iniciado sesión.

-Incluye un botón en la parte superior derecha en el cual podrás publicar un apartamento de forma gratuita, que si se hace click será redirigido a otra página.

- Un botón Profile el cual si se accede a ella se redirige a otra página diferente.

* Si accedemos al botón Publish Free, nos redirigirá a otra página en la cual aparece un formulario. En este formulario se debe ingresar el nombre, el email, el precio, la localidad, descripción y adjuntar una foto del apartamento. Una vez cumplimentado esos campos saldrá un mensaje en el cual se diga que el apartamento ha sido publicado y aparecerá el último en la página principal con los datos adjuntados.
* Si accedemos al botón Profile desde la página principal, nos redirigirá a una página en la cual aparecen todos los apartamentos publicados por el usuario que tiene la cuenta activa. En esta página se podrán borrar los apartamentos cuando quiera el usuario a través del botón Delete.
* Todas las ofertas de apartamentos incluyen un botón de contacto, que si accedemos a él nos redirige a otra página en la cual nos aparece un formulario para rellenarlo mostrando interés por ese apartamento. Aparece un numero de telefono y un email, en formulario relleno se enviará a la persona que publica la oferta, pero el correo será enviado por la cuenta de email de nuestro sitio web.

**4. Diseño**

Después de haber analizado y desarrollado los requisitos generales y casos de usos, en este apartado se detalla el diseño de la aplicación que hemos desarrollado. El objetivo principal es determinar desglosadamente la arquitectura de la aplicación y otras características como el modelo de datos y la interfaz gráfica.

**4.1. Arquitectura**

Cuando decidimos el diseño de la aplicación nos aventuramos a aplicar la arquitectura de cliente/servidor de 3 capas, debido a que será una aplicación web instalada en un servidor remoto.

Con esta arquitectura tenemos varias ventajas al permitirnos separar la lógica de diseño de la lógica de negocio. Los códigos de una capa serán transparentes para otras y se comunicarán mediante servicios, facilitando así el desarrollo en paralelo y mejorando también la mantenibilidad y la reusabilidad. Estas capas proporcionan servicios en la capa superior y a su vez, estos servicios, se utilizan como medio de comunicación con la capa inferior.

* Capa de presentación: Es la capa con la que interacciona el usuario final de la aplicación y proporciona una interfaz gráfica. Aquí se representan los datos del sistema a los usuarios, se gestionan las interacciones que pueda haber y se presentan los posibles cambios sobre la aplicación según las peticiones de los usuarios. Esta capa se comunica con la capa de dominio.

* Capa de dominio o aplicación: Es el núcleo central de la aplicación donde se desarrolla la lógica de negocio. Esta capa es invisible para el usuario. Recibe las peticiones de la capa superior, y tras validarlas, las procesa y se realizan consultas correspondientes a la capa de datos si lo necesita. Como consecuencia, comunica los resultados obtenidos a la capa de presentación. Esta capa proporciona servicios hacia la capa de presentación y se nutre de los servicios de la capa de datos.

* Capa de datos: En esta capa se realiza la interacción con el sistema gestor de base de datos (SGBD). Recibe solicitudes de la capa de dominio y se encarga de hacer consultas como almacenamiento o recuperación de información existente en el gestor. Esta capa también es totalmente invisible para el usuario y accesible para la capa de dominio.



**4.1.1 Capa de presentación**

Como antes hemos indicado, la capa de presentación es la que se comunica directamente

con los usuarios finales mediante la interfaz gráfica. Es la zona en la que se representan visualmente los datos que contiene y es a su vez el que recibe las operaciones de los usuarios y las refleja en el dominio del sistema.

Las tecnologías utilizadas en esta capa han sido HTML y CSS y un framework de google para obtener un aspecto visual más atractivo que a continuación las detallaremos:

* HTML

HyperText Markup Language (HTML), es un lenguaje estándar amplio utilizado para la elaboración de páginas web y altamente soportado por los navegadores. Es el elemento de construcción más básico para la creación y presentación visual del contenido de una página web. Su funcionalidad principal es describir semánticamente la estructura de la página, pero no se encarga de la apariencia y la funcionalidad. En cuanto a la definición de la estructura del contenido de las páginas web, HTML define los elementos que componen una página web, como por ejemplo imágenes, títulos, enlaces, formularios, mediante del uso de etiquetas o marcas (tags).

- CSS

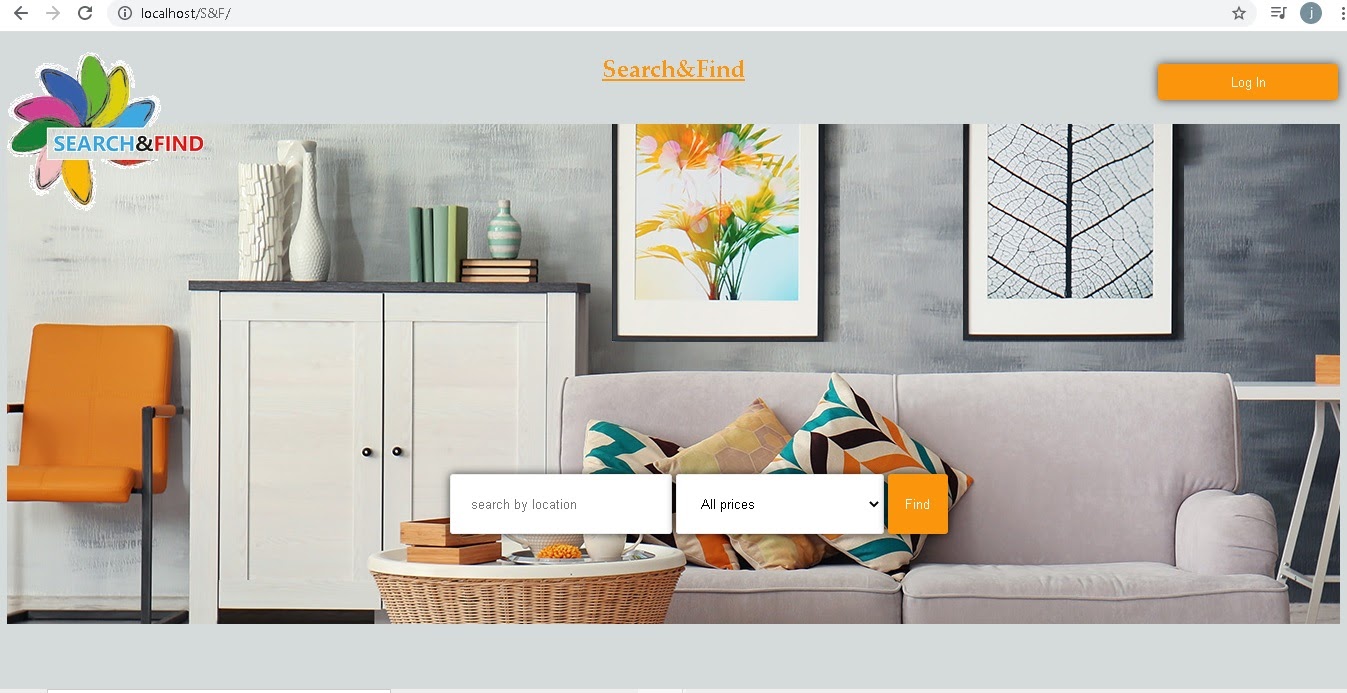
CSS es el acrónimo de Cascading Style Sheets (Hoja de estilos en cascada). Es un elemento básico complementario al HTML cuya funcionalidad es establecer la apariencia de la página web. Junto con HTML, es una tecnología muy popular para crear páginas web visualmente atractivas. Nace con la finalidad de separar el contenido de la forma de presentarse. Esta separación ayuda a reducir la complejidad en la estructura del documento HTML, mejorar la flexibilidad y reusabilidad de los códigos.

- Framework Google

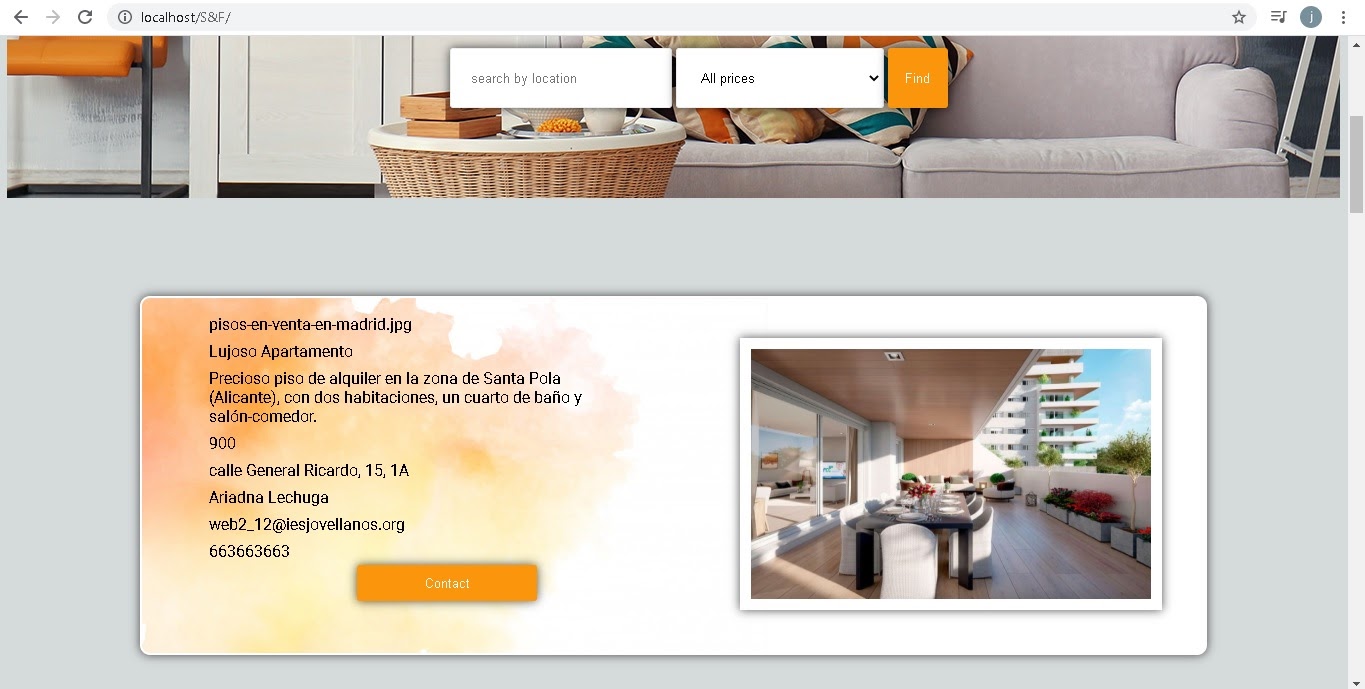
Es un framework o conjunto de herramientas de front-end de código abierto, su funcionalidad principal es agilizar el diseño de las páginas web. Ofrece plantillas de diseño como menú de navegación, formularios, botones y otros elementos basados en HTML y CSS.

A continuación mostramos algunas imágenes de la pantalla principal de nuestra aplicación:

*Foto de index.php*



*Foto de index.php(2)*



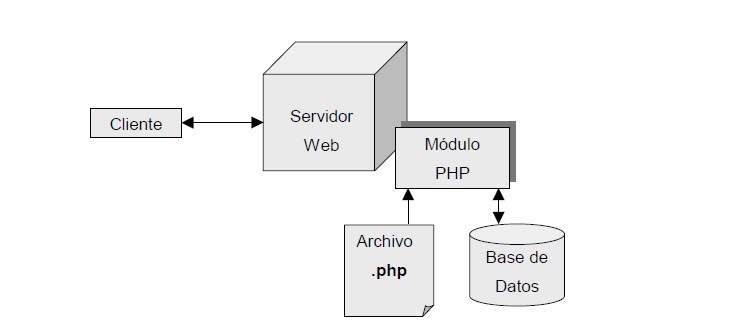
**4.1.2 Capa de dominio o aplicación.**

Esta capa se encarga de desarrollar e implementar toda la lógica de negocio de la nueva aplicación. Proporciona servicios web a la capa de presentación u a otros posibles sistemas externos que lo necesiten.

Las tecnologías que hemos utilizado para la implementación de esta aplicación son PHP como el módulo de lenguaje interpretado y Apache como Servidor Web.

* PHP

Es uno de los mejores soportes para la Programación Orientada a Objetos, mejoras de rendimiento, mejor soporte para MySQL y XML, iteradores de datos y excepción de errores. Se puede entender también como una extensión para servidores web,poniéndose entre cliente y servidor.



PHP se comporta como un módulo de Apache, que extrae código dentro de las páginas, lo ejecuta en el servidor y envía el resultado al cliente. El cliente no puede visualizar el código del programa, solamente su resultado. Además es importante destacar que al ser un lenguaje de script, los programas en PHP no se compilan, sino que únicamente se interpretan, por lo que es más lento en ejecutarse como programas hechos en C, que requieren estar compilados. No obstante, su gran ventaja es que un cambio en el código en PHP tiene un efecto inmediato, y en C por ejemplo, es necesario volver a compilarlo. Resulta importante aclarar que sólo son interpretados por el módulo PHP, es decir pasados por parte del Núcleo de Apache a él, los archivos que hayan sido definidos en la configuración del servidor como tales, dado que la interpretación de código es un proceso que necesita un tiempo mayor que el empleado en enviar una página tal como esté. Normalmente se emplea la extensión .php, aunque también se utilizan otras como .php3, .php4, y phtml.

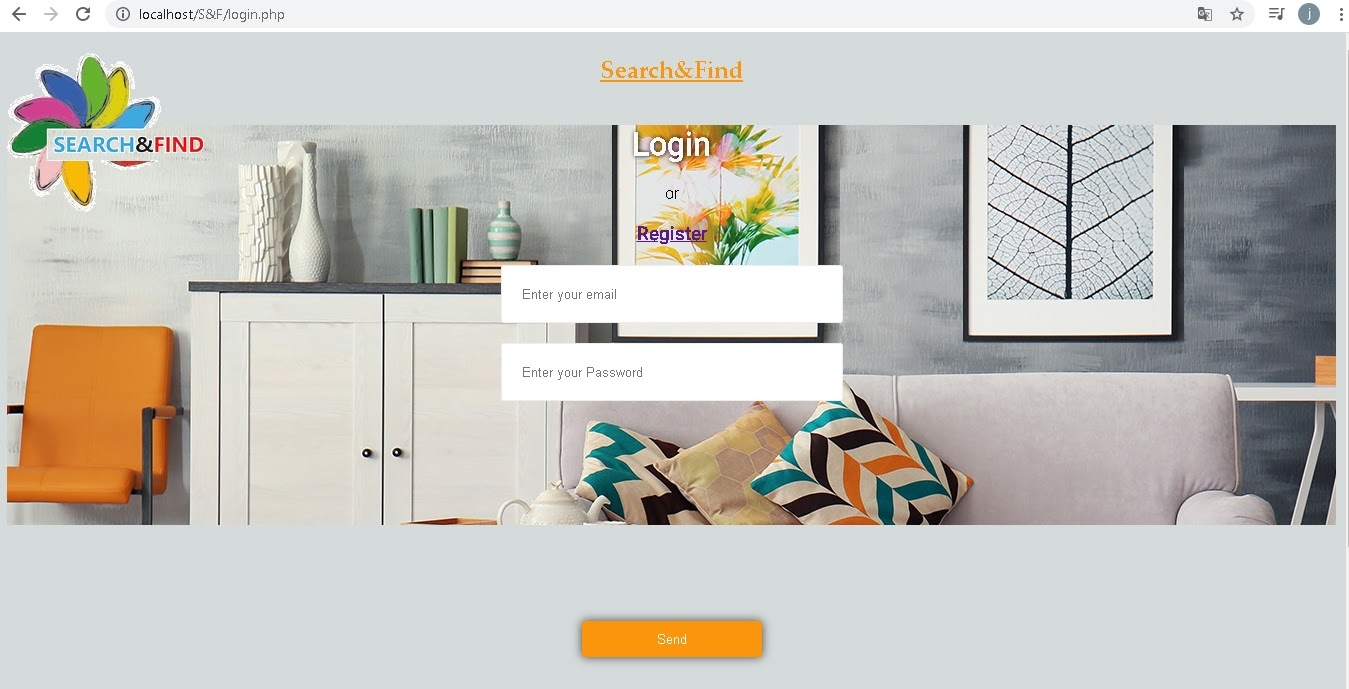
En nuestro proyecto, el módulo de lenguaje interpretado se encarga de atender las peticiones del Servidor Web, las interpreta y construye la respuesta que va a ser destinada al cliente, junto con accesos a la Base de Datos y al disco si fuera necesario.

Las funciones de este módulo en nuestro proyecto son:

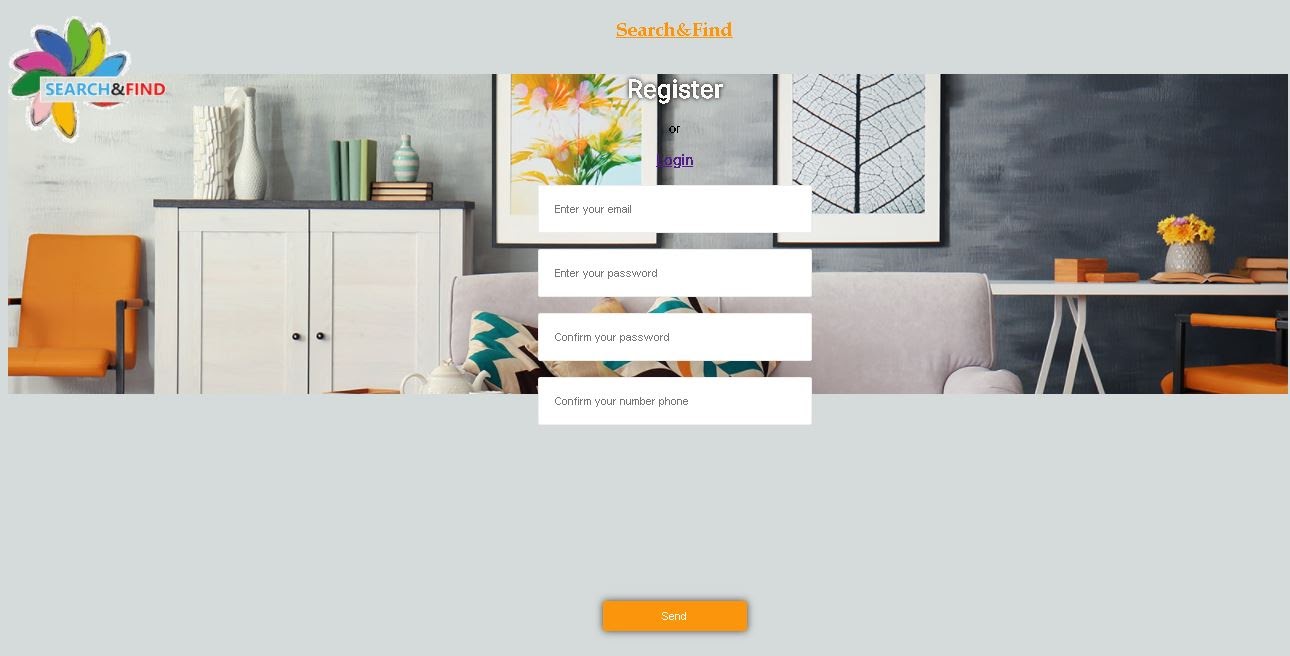
- Registro y logueado

* Para acceder a determinadas secciones se encarga de presentar al usuario la necesidad de autenticarse así como el proceso de inicio de sesión.
* Gestiona labores de cerrado y destrucción de sesiones de los usuarios.

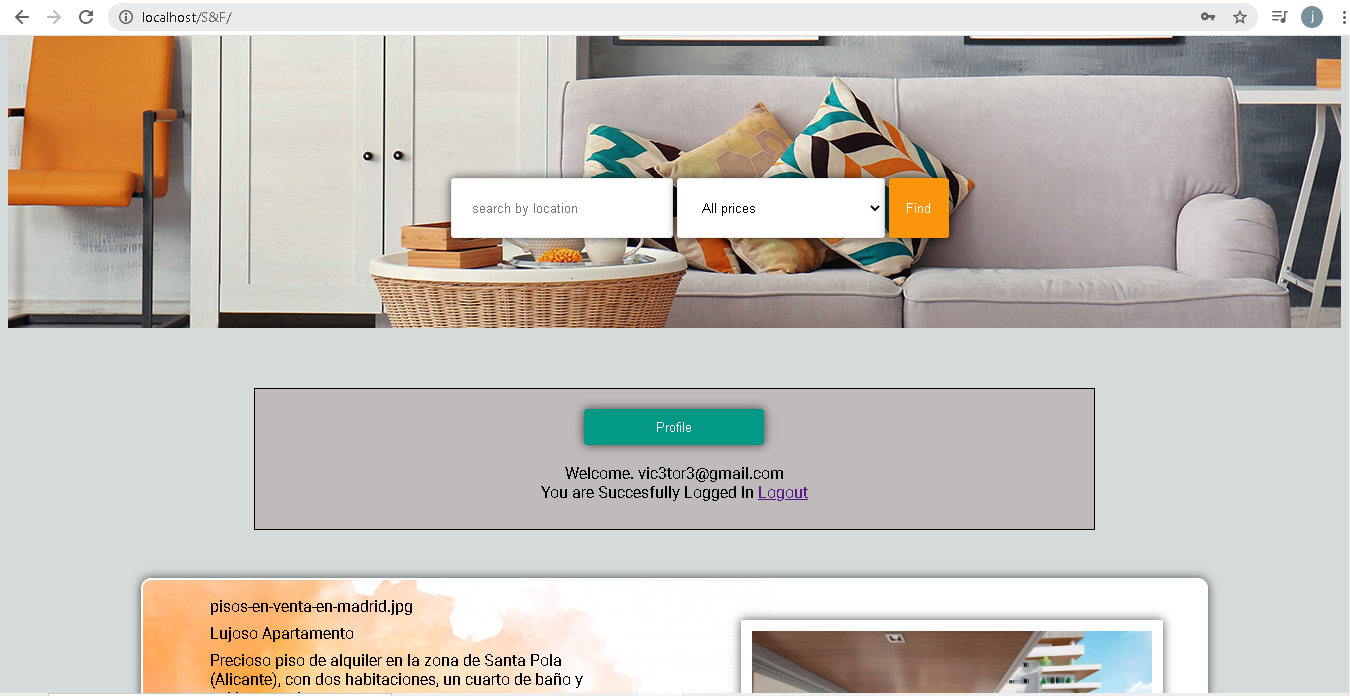
*Foto de login.php*



*Foto de register.php*



*Foto de logout.php*



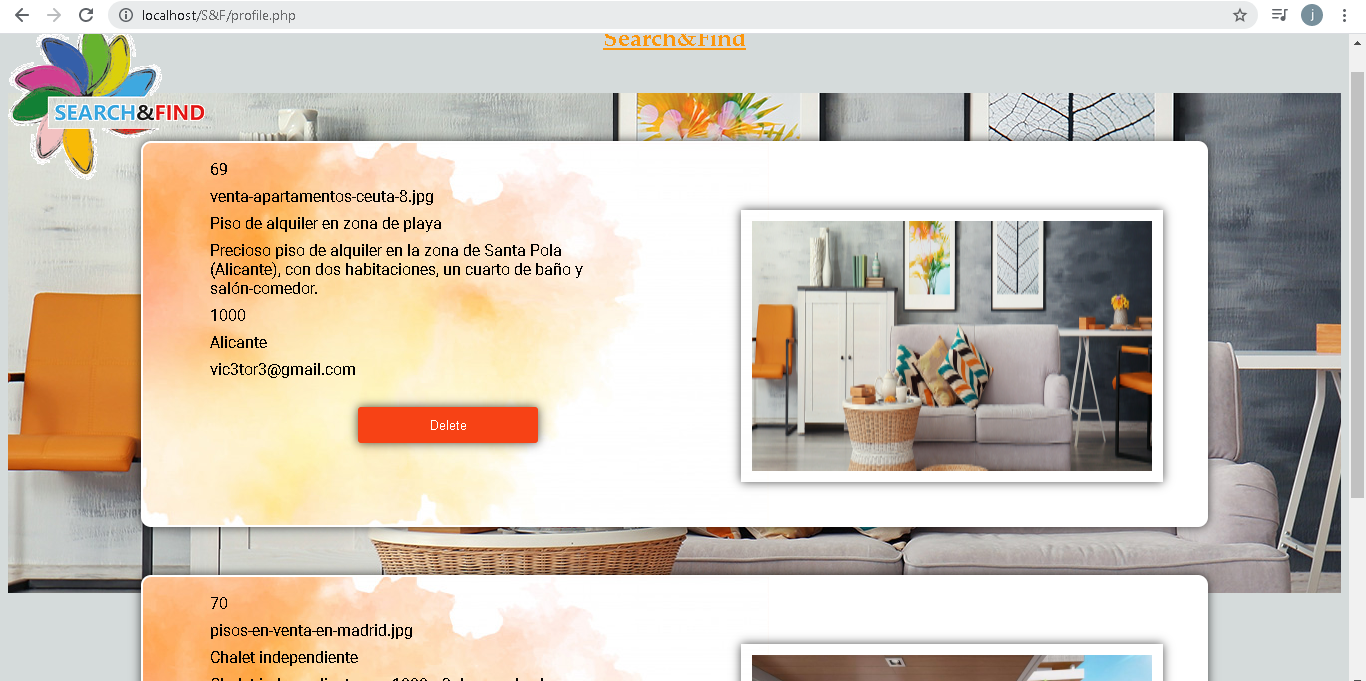
- Búsqueda o filtrado

* Requiere en la pantalla inicial el rellenado de parámetros de filtrado y muestra los resultados.
* Permite visualizar tantos los datos normales como los filtrados.
* Permite visualizar los datos agregados del usuario logueado en su perfil personal.
* En el caso de que se acceda a la página principal y se quiera contactar con un usuario en concreto, se puede solicitar la información relevante para realizar un envío de correo electrónico o a través de un número de contacto.

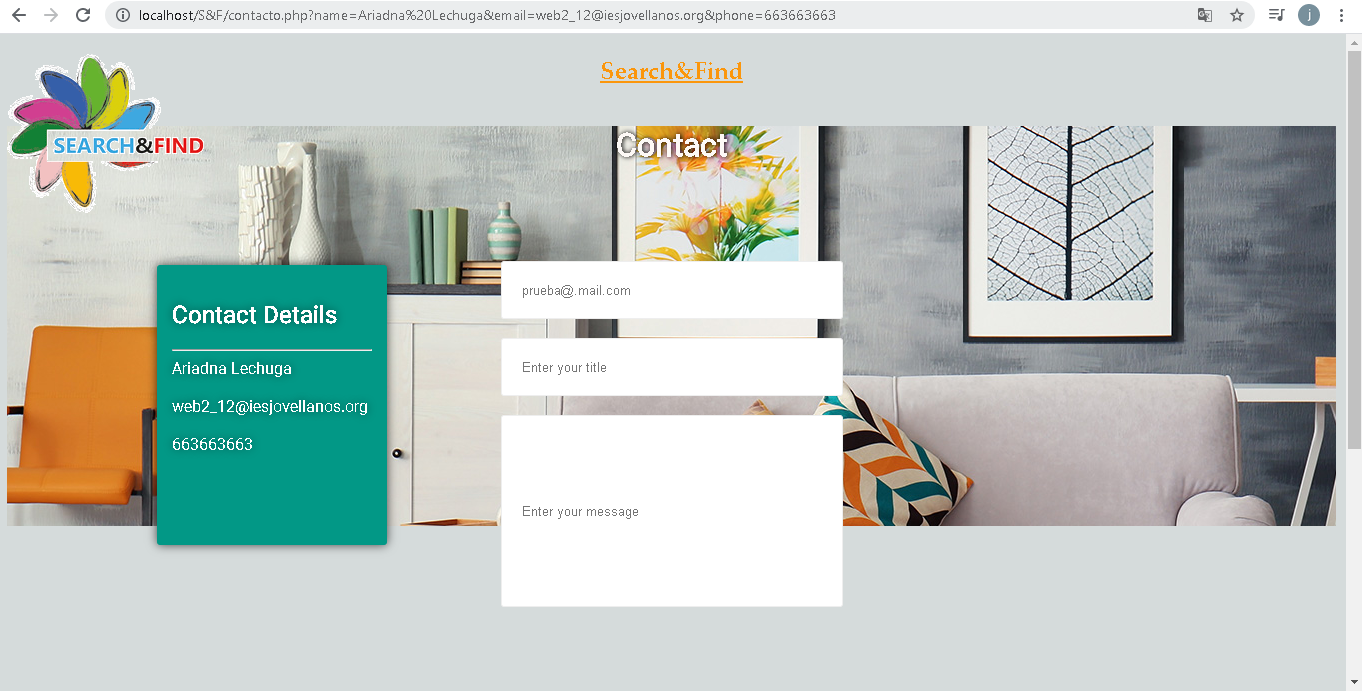
- Borrado y envío de correo.

* En el caso de que se esté logueado y se haya publicado un anuncio, este usuario tiene la posibilidad de borrar dicho anuncio.
* El cliente interesado en solicitar más información sobre un anuncio, puede enviar un correo electrónico a ese usuario en concreto, rellenando un sencillo formulario.

*Foto de profile.php*



*Foto de contact.php*



Estas funciones introducen o eliminan la información deseada de nuestra base de datos. A su vez, estas también cuentan con un archivo principal para cada una, que indica la presentación en pantalla de aspectos comunes, además de otros archivos que se le introducen por medio de las funciones de PHP include y require. Estos archivos que se introducen almacenan el código de diversas funciones además de invocar archivos con la configuración de la aplicación.

En estos archivos de configuración se almacenan juegos de caracteres para fechas, números o texto, datos de conexión a la Base de Datos, función de conexión a la Base de Datos que devuelve el resultado de la consulta que se le envía como argumento, nombre del archivo de login y register encargado de la autenticación, así como el nombre de la sesión del usuario que el servidor crea para cada cliente.

- Apache

Para el desarrollo y la aplicación de este proyecto ha sido indispensable la utilización de un servidor web.

El servidor web tiene la función de permanecer a la espera de peticiones HTTP de los clientes, que suelen llevarlas a cabo a través de un navegador web. Los clientes realizan peticiones HTTP al servidor, y este les responde con el/los contenido/s que los clientes solicitan. Los contenidos devueltos por el servidor suelen ser interpretados por los navegadores web de los clientes, que los muestran por pantalla acorde a unas fuentes, colores, disposición de textos y objetos, por lo que el servidor se limita a transferir el código de la página sin llevar a cabo ninguna interpretación de ella.

Nosotros hemos decidido implementar la utilización del servidor web Apache. Se trata de un servidor de código abierto cuyos inicios se remontan a 1995, actualmente se desarrolla dentro del proyecto HTTP Server de Apache Software Foundation, y que soporta plataformas UNIX, Windows y cualquier otra que implemente el protocolo HTTP/1.1. Es muy posiblemente el servidor web más conocido y utilizado.

Actualmente existen muchas versiones de Apache, nosotros hemos optado por utilizar la versión 2.4 que incluye las siguientes características:

* Está disponible para muchos Sistemas Operativos.
* Al ser una tecnología gratuita de código fuente abierto, su software es transparente y se puede saber concretamente lo que se ha instalado.
* Tiene diseño modular, lo que permite de una manera muy sencilla ampliar sus capacidades.
* Permite trabajar con CGI, Java, Perl y PHP, entre otros lenguajes de script, incluso en modo CGI o como módulos.
* Ofrece una alta configurabilidad en la creación y gestión de logs, por si se requiere tener un mayor control sobre el funcionamiento del servidor web.

Como ya hemos explicado anteriormente, hemos decidido utilizar el servidor web Apache por diversas opciones y porque hemos llegado a la conclusión que será el que mejor encajaría en nuestra aplicación. Nuestro servidor web administra y atiende las peticiones de los clientes, enviándolas al módulo de lenguaje interpretado cuando sea necesario.

La ubicación de los archivos deben estar organizados en la carpeta correspondiente para que el servidor pueda acceder a ellos. Estos archivos están agrupados en el mismo fichero y se compone de artículos en formato pdf, hojas de estilo, imágenes y todos los scripts de php que componen el proyecto.

**4.1.3 Capa de datos**

Esta capa se nutre del sistema gestor de base de datos (SGBD) a partir de peticiones de la capa de dominio. La organización consiste en el diseño de la base de datos donde se almacenan los datos de nuestra aplicación. En nuestro caso hemos decidido utilizar el sistema de administrador de base de datos MySQL Workbench.

* MySQL

MySQL GUI Tools Bundle es una [*suite*](https://es.wikipedia.org/wiki/Software_suite) de código abierto y de multiplataforma de aplicaciones de escritorio para la administración de servidores de bases de datos MySQL, para construir y manipular los datos en una base de datos MySQL. Fue desarrollada por MySQL AB y después por [Sun Microsystems](https://es.wikipedia.org/wiki/Sun_Microsystems) y lanzada bajo la licencia GPL. Su desarrollo fue detenido, y ahora solo es preservado bajo los archivos de descarga del sitio de MySQL.

MySQL GUI Tools Bundle fue superada por MySQL Workbench, y llegó al final de su vida con la publicación de la versión beta de MySQL Workbench 5.2, aunque el equipo de soporte de MySQL continuó proveyendo asistencia para MySQL GUI Tools Bundle hasta el 30 de junio de 2010. Futuros lanzamientos de MySQL Workbench añadirán un plugin de migración, añadiendo características equivalentes al componente MySQL Migration Toolkit de MySQL GUI Tools Bundle.

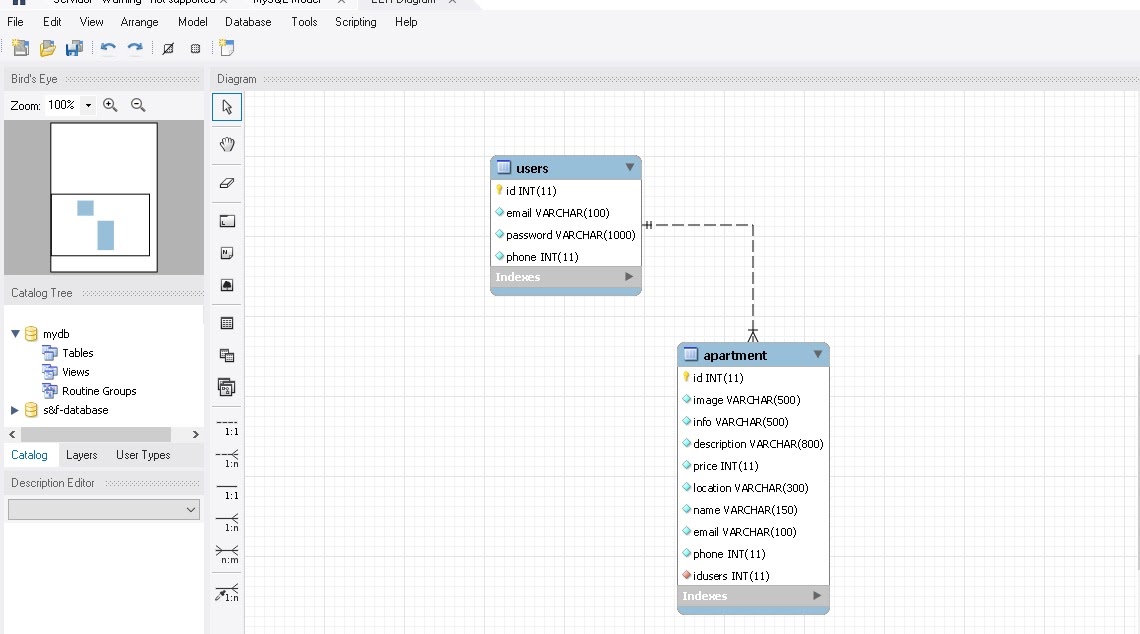
Hemos decidido utilizar MySQL Workbench debido a que era el administrador de base de datos que hemos aprendido en el módulo y nos daba una mayor seguridad frente a borrados accidentales. También vemos que funciona muy bien en diferentes sistemas operativos. Según la información que hemos podido obtener de otros sistemas de base de datos, MySQL nos presenta una buena relación entre calidad y usabilidad. En este caso, Oracle por ser un sistema de pago y costoso lo descartamos y Postgre SQL presenta una mayor lentitud que los demás y ya que no tenemos muchas entradas de datos también lo descartamos.

Actualmente la versión más reciente es la versión 6.3 y las características más importantes son las siguientes:

* Multiplataforma, soportando varios Sistemas Operativos como GNU/Linux, Mac OS X, SunOS y MS Windows, entre otros.
* Tiene disponibles APIs para C, C++, Eiffel, Java, Perl y PHP, entre otros lenguajes.
* Soporta multi-hilo (multi-threaded) para múltiples microprocesadores.
* Sencillez para añadir otro sistema de almacenamiento, permitiendo añadir una interfaz SQL a una base de datos propia.
* Reserva de memoria muy rápida basada en threads (hilos).
* Trabaja con funciones SQL implementadas mediante una librería altamente optimizada, con una rapidez tan grande como sea posible, y por lo que normalmente no se reserva memoria después de toda la inicialización para consultas.
* El servidor se puede utilizar como un programa separado para el caso de un entorno de red cliente-servidor. A su vez también está disponible como biblioteca y puede ser embebido en aplicaciones autónomas, con disponibilidad o no de red.

A continuación vamos a mostrar el diagrama de entidad-relación de la base de datos de nuestra aplicación y toda la información existente en la misma.

*Diagrama E/R*



*Tabla Apartment*

-- MySQL dump 10.13 Distrib 5.7.17, for Win64 (x86\_64)

--

-- Host: 127.0.0.1 Database: s&fdatabase

-- ------------------------------------------------------

-- Server version 5.5.5-10.4.11-MariaDB

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!40101 SET NAMES utf8 \*/;

/\*!40103 SET @OLD\_TIME\_ZONE=@@TIME\_ZONE \*/;

/\*!40103 SET TIME\_ZONE='+00:00' \*/;

/\*!40014 SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0 \*/;

/\*!40014 SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0 \*/;

/\*!40101 SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='NO\_AUTO\_VALUE\_ON\_ZERO' \*/;

/\*!40111 SET @OLD\_SQL\_NOTES=@@SQL\_NOTES, SQL\_NOTES=0 \*/;

--

-- Table structure for table `apartment`

--

DROP TABLE IF EXISTS `apartment`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!40101 SET character\_set\_client = utf8 \*/;

CREATE TABLE `apartment` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`image` varchar(1000) NOT NULL,

`info` varchar(500) NOT NULL,

`description` varchar(800) NOT NULL,

`price` int(11) NOT NULL,

`location` varchar(300) NOT NULL,

`name` varchar(150) NOT NULL,

`email` varchar(100) NOT NULL,

`phone` int(11) NOT NULL,

`idusers` int(11) NOT NULL,

PRIMARY KEY (`id`),

KEY `idusers\_idx` (`idusers`),

CONSTRAINT `idusers` FOREIGN KEY (`idusers`) REFERENCES `users` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION

) ENGINE=InnoDB AUTO\_INCREMENT=15 DEFAULT CHARSET=latin1;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Dumping data for table `apartment`

--

LOCK TABLES `apartment` WRITE;

/\*!40000 ALTER TABLE `apartment` DISABLE KEYS \*/;

INSERT INTO `apartment` VALUES (1,'piso\_playa.jpg','Piso de alquiler en zona de playa','Precioso piso de alquiler en la zona de Santa Pola (Alicante), con dos habitaciones, un cuarto de baÃ±o y salÃ³n-comedor.',650,'Alicante','Lucia Perez Lopez','lucia@mail.com',656656656,1),(6,'foto\_chalet.jpg','Chalet independiente','Chalet independiente con 1000m2 de parcela, dos habitaciones, 3 baÃ±os, piscina, y un salon comedor',1250,'Toledo','Joaquin Cruz Campo','Joaquin@mail.com',669669669,2),(7,'foto\_chalet2.jpg','Chalet independiente','Chalet independiente con 800m2 de parcela, 2 habitaciones, 2 baÃ±os, cocina, y un salÃ³n enorme',1100,'Toledo','Joaquin Cruz Campo','Joaquin@mail.com',669669669,2),(8,'chalet\_montana.jpg','Piso de alquiler en zona de montaÃ±a','Precioso piso en zona de montaÃ±a con 2 habitaciones, 1 cuarto de baÃ±o completo y salÃ³n-comedor',750,'Huesca','Francisco Garcia Garcia','paco@mail.com',662662662,3),(11,'piso\_Gijon.jpg','Hermoso piso en la periferia de GijÃ³n','Alquilo un pequeÃ±o piso en la zona del llano (GijÃ³n), con dos habitaciones, 1 baÃ±o, salÃ³n y cocina reformada.',470,'GijÃ³n','Lucia Perez Lopez','lucia@mail.com',656656656,1),(12,'loft\_barna.jpg','Maravilloso loft en el centro de Barcelona','Alquilamos un encantador loft en el centro de Barna, a escasos metros de la Sagrada Familia. 1 habitacion, 1 baÃ±o, cocina americana.',950,'Barcelona','Jose Luis Garcia Aparicio','pepe@mail.com',666666666,4),(13,'pisosmadrid.jpg','Piso de alquiler en Madrid','Lujoso piso en la zona de Arguelles. 2 habitaciones, 2 baÃ±os, terraza, cocina amueblada y salÃ³n.',1000,'Madrid','Jose Luis Garcia Aparicio','pepe@mail.com',666666666,4),(14,'Hurdes.jpg','Alquilo hermosa casa en las Hurdes','Se alquila hermosa casa en las Hurdes. ConstrucciÃ³n milenaria con gran terreno y buenos cimientos. 3 habitaciones, 1 baÃ±o y salÃ³n-comedor.',200,'Caceres','Francisco Garcia Garcia','paco@mail.com',662662662,3);

/\*!40000 ALTER TABLE `apartment` ENABLE KEYS \*/;

UNLOCK TABLES;

/\*!40103 SET TIME\_ZONE=@OLD\_TIME\_ZONE \*/;

/\*!40101 SET SQL\_MODE=@OLD\_SQL\_MODE \*/;

/\*!40014 SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS \*/;

/\*!40014 SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS \*/;

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

/\*!40111 SET SQL\_NOTES=@OLD\_SQL\_NOTES \*/;

-- Dump completed on 2021-06-16 18:29:10

*Tabla Users*

-- MySQL dump 10.13 Distrib 5.7.17, for Win64 (x86\_64)

--

-- Host: 127.0.0.1 Database: s&fdatabase

-- ------------------------------------------------------

-- Server version 5.5.5-10.4.11-MariaDB

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!40101 SET NAMES utf8 \*/;

/\*!40103 SET @OLD\_TIME\_ZONE=@@TIME\_ZONE \*/;

/\*!40103 SET TIME\_ZONE='+00:00' \*/;

/\*!40014 SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0 \*/;

/\*!40014 SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0 \*/;

/\*!40101 SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='NO\_AUTO\_VALUE\_ON\_ZERO' \*/;

/\*!40111 SET @OLD\_SQL\_NOTES=@@SQL\_NOTES, SQL\_NOTES=0 \*/;

--

-- Table structure for table `users`

--

DROP TABLE IF EXISTS `users`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!40101 SET character\_set\_client = utf8 \*/;

CREATE TABLE `users` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`email` varchar(100) NOT NULL,

`password` varchar(1000) NOT NULL,

`phone` int(11) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=latin1;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Dumping data for table `users`

--

LOCK TABLES `users` WRITE;

/\*!40000 ALTER TABLE `users` DISABLE KEYS \*/;

INSERT INTO `users` VALUES (1,'lucia@mail.com','$2y$10$UmTfeW/.yOPXtfUWcMQD6eWi4.bCid5izPqRdHIt.c3RVtHsdSL/O',0),(2,'Joaquin@mail.com','$2y$10$tz1.OIF/keJ7sRK/7GJ6qu0PcnUDHdk/qxN8yaMDUksmrHwjYtdca',0),(3,'paco@mail.com','$2y$10$vkf.GizZQqwZmDNIJW6ZTeE1RHGEQ0igIiOgTZgOURKpa7P1i.mx6',0),(4,'pepe@mail.com','$2y$10$LNQlyMBYgTDXVmPvBBGi8urgLDOhDNpbRVDt4dbD.kNAwh.z6lqzi',0);

/\*!40000 ALTER TABLE `users` ENABLE KEYS \*/;

UNLOCK TABLES;

/\*!40103 SET TIME\_ZONE=@OLD\_TIME\_ZONE \*/;

/\*!40101 SET SQL\_MODE=@OLD\_SQL\_MODE \*/;

/\*!40014 SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS \*/;

/\*!40014 SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS \*/;

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

/\*!40111 SET SQL\_NOTES=@OLD\_SQL\_NOTES \*/;

-- Dump completed on 2021-06-16 18:29:10

**4.2. Interfaz gáfica**

En el desarrollo de nuestra interfaz gráfica hemos tenido en cuenta varios factores que creemos que son importantes para la interacción del usuario con la misma.

Creemos que es muy importante que la información sea lo más limitada a lo necesario o indispensable para que el usuario no se pierda y pueda moverse con fluidez y rapidez.

También hemos tenido muy en cuenta la organización de la información, esta debe ser ordenada y visualmente bien estructurada.

En cuanto a los iconos y gráficos hemos intentados que sean los más reconocibles posibles, utilizando un tamaño perfectamente visible y colocados centrados o en alguno de los extremos superiores.

En cuanto a los colores hemos querido usar en casi su totalidad, sobre todo en el fondo de todas las páginas, una gama de colores planos de grises para dar cierta sobriedad y sofisticación. No obstante también hemos incorporado en menor cantidad varios tonos de la gama cromática, resaltando el color naranja que refuerza el entusiasmo, la felicidad y creatividad y el azul turquesa que genera cierta estabilidad y racionalidad.

**4.2.1 Usabilidad**

En cuanto a la usabilidad, hemos intentado crear una aplicación lo más sencilla posible con la que el usuario pueda cumplir con una tarea específica de manera efectiva, eficaz y obteniendo cierto grado de satisfacción. Hemos intentado colocar los botones estrictamente necesarios, haciendo desaparecer estos cuando no se necesitaban, hemos organizado gran parte de la información en cajas de texto e imágenes claramente visibles, y también hemos intentado resaltar ciertos apartados cuando creemos que el usuario lo va a necesitar.

El usuario puede acceder con facilidad a la aplicación y puede moverse a través de ella sin importar hardware o software,tampoco necesita tener amplios conocimientos en la materia para poder moverse por ella. Creemos que esta aplicación puede ser utilizada por casi todas las personas, exceptuando a personas con ciertas limitaciones físicas que no han sido contempladas.

El sistema es bastante útil e intuitivo para que el usuario pueda desempeñar sus actividades, publicar o alquilar uno o varios apartamentos.

En cuanto a calidad, no podemos comparar con verdadera exactitud nuestra aplicación respecto a otras del mismo segmento, pero hemos intentado llevar a cabo las pautas generales de este tipo de aplicaciones, incluyendo los apartados más relevantes.

Respecto a la fiabilidad de la aplicación, el sistema cumple con casi todos los requisitos para el uso por parte del usuario.

Creemos que hemos sido coherentes con el significado de objetos, formas y métodos de trabajo evitando que el usuario tenga que realizar actividades diferentes en tareas similares del sistema, a excepto del login y el register.

Hemos aplicado estilos css y html para que en cualquier navegador o equipo pueda ser visible con casi la misma calidad, imagen y tamaño, sin que experimente cambios bruscos o pierda funcionalidad.

El proyecto comprende un marco estructural con una carpeta denominada “css" en donde está la hoja de estilos.También hay un apartado con las imágenes usadas, así como la página index junto a otras que contienen el código básico de la aplicación web. También hemos utilizado ciertas propiedades para que la aplicación tenga la tecnología responsive sin necesidad de utilizar javaScript.

**5. Pruebas**

Al terminar cada apartado de la aplicación se han realizado diferentes pruebas para corroborar que todo funciona correctamente. Se han hecho pruebas sobre cada módulo de funcionalidades. A continuación se detallan:

* Se comprueba que la conexión a la base de datos es correcta y que la inserción, visualización y borrado de las jerarquías de datos es correcto y no se colapsa.
* Que la utilización del filtrado realiza las secuencias deseadas una vez que se selecciona el dato deseado y se selecciona la acción de “find”.
* Se hacen pruebas sobre el login y el register que no acepta diferentes contraseñas y que una vez registrado el usuario concuerda con los datos de la página de registro.
* Se comprueba que crea el perfil cuando se registra y logea así como la interrupción de la sesión cuando se aplica.
* Se visualiza correctamente los diferentes anuncios con el texto indicado, las imágenes y los botones de contacto.
* Se hacen pruebas para verificar que el usuario logueado es capaz de publicar un anuncio y aparece en su perfil una vez que se selecciona la acción de “profile”.
* Se comprueba que el enlace superior de “Search&find” redirige automáticamente a la página principal.
* Probamos que el botón de “delete” borra los anuncios del usuario que ha registrado es publicación.
* Se comprueba que mediante la acción de “contact” nos redirige correctamente a dicha página y una vez rellenado el formulario para contactar mediante correo electrónico se envía satisfactoriamente al usuario seleccionado.
* La organización de los diferentes elementos no sufren modificaciones extrañas cuando se minimiza la pantalla y tampoco pierden ningun tipo de funcionalidad.
* Se prueba la aplicación en diferentes navegadores con Google Chrome, Microsoft Egde y Mozilla y funciona correctamente.

**6. Despliegue**

En cuanto al despliegue de nuestra aplicación, podemos indicar que para instalarla debemos tener en cuenta los diferentes agentes que hemos utilizado a la hora de poder llevar a cabo este proyecto.

Como anteriormente hemos desarrollado, para el despliegue de la aplicación necesitamos un gestor o sistema de base de datos, un servidor web y un entorno de desarrollo.

No se especifica un sistema operativo en concreto ni tampoco tener una máquina con unas determinadas características técnicas.

Tampoco se ha especificado el explorador a utilizar, en nuestro caso hemos desplegado la aplicación en el explorador Google Chrome, aunque también se puede realizar dicho despliegue en otros exploradores conocidos.

Para que el usuario pueda navegar por ella, no necesita permisos especiales, excepto para realizar ciertas tareas, sin olvidar que requiere estar autenticado para tales efectos.

Los procesos de escritura en la Base de Datos como en disco, los puede realizar también el usuario autenticado.

**6.1 Operaciones**

A continuación, vamos a detallar las diferentes operaciones que se pueden realizar a la hora de que el usuario navegue por la aplicación con las diferentes opciones que se les plantea. En el apartado de la capa de dominio hemos desarrollado estos apartados incluyendo imágenes, por eso aquí sólo los vamos a nombrar.

Los usuarios puede realizar las operaciones de:

* Autenticado
* Borrado
* Inserción
* Filtrado
* Envío de correo
* Cierre de sesión

Todas estas operaciones se podrán realizar en función de los roles que tenga cada usuario, estando en el perfil de autenticado o en el perfil de usuario sin autenticado.

**6.2 Mantenimiento**

Generalmente y salvo algún error inesperado que pueda ocurrir en el funcionamiento, la aplicación se puede automantener.

En diferentes momentos de la utilización de la aplicación puede necesitar de la actualización del sistema de gestión de Base de Datos MySQL Workbench y el servidor Apache para el correcto funcionamiento de la misma.

Dado las preferencias del cliente se podría necesitar modificaciones específicas en el producto y su posterior mantenimiento periódico según demanda.

**7. Bibliografía utilizada**

[**https://www.lawebera.es/xhtml-css/posicion-capas-css.php**](https://www.lawebera.es/xhtml-css/posicion-capas-css.php)

[**https://es.wikipedia.org/wiki/Hoja\_de\_estilos\_en\_cascada**](https://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada)

[**https://es.wikipedia.org/wiki/HTML**](https://es.wikipedia.org/wiki/HTML)

[**https://www.ionos.es/digitalguide/servidores/know-how/servidor-web-definicion-historia-y-programas/**](https://www.ionos.es/digitalguide/servidores/know-how/servidor-web-definicion-historia-y-programas/)

[**https://es.wikipedia.org/wiki/MySQL\_Workbench**](https://es.wikipedia.org/wiki/MySQL_Workbench)

[**https://dspace.unl.edu.ec/jspui/bitstream/123456789/14156/1/Carri%C3%B3n%20Chamba%2C%20Rober%20Alonso.pdf**](https://dspace.unl.edu.ec/jspui/bitstream/123456789/14156/1/Carri%C3%B3n%20Chamba%2C%20Rober%20Alonso.pdf)