



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Gestión Web De Alojamiento Vacacional

Proyecto Final de Carrera

Ingeniería Técnica en Informática de Sistemas

Autor: Mohammed El Amine Boumediane

Director: Félix Buendía García

Septiembre 2012

Índice

1. Introducción	5
1.1- Objetivos	5
1.2- Contexto	5
1.3- Estructura	8
2. Especificación de requisitos (ERS).....	9
2.1- Propósito.....	9
2.2- Ámbito	9
2.3- Definiciones, Siglas y Abreviaturas	9
2.4- Referencias.....	10
2.5- Visión Global.....	10
2.6- Descripción General.....	10
2.6.1- Perspectiva del Producto	10
2.6.2- Funciones del Producto.....	10
2.6.3- Características del Usuario.....	12
2.6.3- Restricciones	13
2.6.4- Supuestos y Dependencias	13
2.7- Requisitos Específicos	13
2.7.1- Requisitos de Interfaces Externos	13
3. Análisis.....	24
3.1- Diagrama de clases	24
3.2- Diagrama de casos de uso	26
3.3- Diagrama de secuencia.	27
3.3.1- Iniciar sesión.....	28
3.3.2- Crear inmueble	29
3.3.3- Eliminar inmueble.....	30
4. Diseño	31
4.1- Capa Interfaz.....	31
4.1.1- Parte Pública	32
4.1.2- Parte privada.....	33

4.2-	Capa de negocio	35
4.3-	Capa de persistencia	37
5.	Implementación	38
5.1-	Tecnologías de implementación web.....	38
5.1.1-	PHP.....	38
5.1.2-	JQuery.....	39
5.1.3-	Java Script.....	39
5.1.4-	MySQL	40
5.1.5-	Apache.....	40
5.2-	Descripción de la implementación	40
5.2-1.	Capa de presentación	40
5.2-2.	Capa de aplicación.....	42
5.2-3.	Capa de persistencia.....	44
6.	Pruebas	47
6.1-	Resolución de pantalla.....	47
6.2-	Validación del código CSS.....	47
6.3-	Compatibilidad entre navegadores y Resolución de pantalla	48
6.3-1.	Explorer.....	49
6.3-2.	Chrome.....	52
6.3-3.	Firefox	55
7.	Conclusiones	59
8.	Anexos.....	60
8.1-	Manual de usuario	60
8.1-1.	Usuario registrado.....	60
8.1-2.	Usuario Administrador	61
9.	Bibliografía	63



Índice De Ilustración

Ilustración 1: Web Enalquiler.....	6
Ilustración 2: Web Fotocasa.....	7
Ilustración 3: Interfaz de Usuario Parte Pública.....	14
Ilustración 4: Interfaz de Usuario Parte Privada.....	15
Ilustración 5: Diagrama de clases.....	25
Ilustración 6: Actores del sistema.....	26
Ilustración 7: Casos de uso general.....	27
Ilustración 8: Diagrama de secuencia, iniciar sesión.....	28
Ilustración 9: Diagrama de secuencia, crear inmueble.....	29
Ilustración 10: Diagrama de secuencia, eliminar inmueble.....	30
Ilustración 11: Arquitectura tres capas.....	31
Ilustración 12: Página de inicio.....	32
Ilustración 13: Menú derecha.....	33
Ilustración 14: Inicio de sesión.....	34
Ilustración 15: Estructura del usuario administrador.....	34
Ilustración 16: Estructura del usuario registrado.....	35
Ilustración 17: Arquitectura de capas.....	36
Ilustración 18: Modelo Entidad-Relacion.....	37
Ilustración 19: Código de la cabecera en HTML.....	41
Ilustración 20: Ejemplo del código CCS.....	41
Ilustración 21: Menú superior.....	42
Ilustración 22: Código php Clase Inmueble.....	43
Ilustración 23: Código PHP clase apartavacaciones.....	44
Ilustración 24: conexión MySQL.....	45
Ilustración 25: Código creación DDBB.....	46
Ilustración 26: Validation del código CCS.....	48
Ilustración 27: Navegadores más usados.....	49
Ilustración 28: IE parte pública 800 x 600.....	50
Ilustración 29: IE parte privada 800 x 600.....	50
Ilustración 30: IE parte pública 1600 x 1200.....	51
Ilustración 31: IE parte privada 1600 x 1200.....	52
Ilustración 32: Chrome parte pública 800 x 600.....	53
Ilustración 33: Chrome parte privada 800 x 600.....	53
Ilustración 34: Chrome parte pública 1600 x 1200.....	54
Ilustración 35: Chrome parte privada 1600 x 1200.....	55
Ilustración 36: Firefox parte pública 800 x 600.....	56
Ilustración 37: Firefox parte privada 800 x 600.....	56
Ilustración 38: Firefox parte pública 1600 x 1200.....	57
Ilustración 39: Firefox parte privada 1600 x 1200.....	58
Ilustración 40: Edición de perfil.....	60
Ilustración 41: Gestión de inmuebles.....	61

1. Introducción

Este documento, describe el trabajo realizado en el Proyecto Final de Carrera de Ingeniería Técnica en Informática de Sistemas, cursado en la Escuela Técnica Superior de Ingeniería Informática de la Universidad Politécnica de Valencia.

En este proyecto que trata el ámbito de la web, se ha elaborado y diseñado una inmobiliaria online, con el propósito de mejorar los portales inmobiliarios, para que sea una cómoda herramienta de trabajo de los agentes.

En esta memoria se describen todas las fases del proyecto, desde la idea y la especificación de requisitos, hasta la implementación y la ejecución por los usuarios finales.

1.1- Objetivos

La finalidad del proyecto, consiste en la creación de un portal web para una inmobiliaria, que sirve para dar información al usuario no registrado, para facilitar las citas a los usuarios registrados y para que sea una herramienta fundamental de trabajo para el usuario administrador. Hay tres tipos de usuarios, que son los siguientes:

- Usuarios no registrados: ver y buscar inmuebles y registrarse.
- Usuarios registrados: privilegios del usuario no registrado y la posibilidad de modificar/consultar su propio perfil, contactar con el dueño del inmueble y crear/consultar citas a través del servicio de citas. Pueden ver y eliminar todos sus mensajes.
- Usuario administrador: privilegios de los dos usuarios citados anteriormente, y además puede crear, modificar y borrar los inmuebles y servicios, ver y eliminar tanto sus mensajes como el de los otros usuarios y puede ver los usuarios.

1.2- Contexto

Este portal web, se ha realizado orientado a un sector inmobiliario, donde los usuarios podrán buscar apartamentos, casas, chalets, bajos... para alquilar, en multitud de idiomas.

La novedad de esta web inmobiliaria, es que el usuario registrado y el administrador, tienen un servicio de citas. Al seleccionar el apartado de citas, se accede a un calendario, en el que se puede elegir la fecha, una vez seleccionada, ésta queda guardada, y se puede volver a consultar. Con el servicio de citas se cita al agente del inmueble, siempre que tenga un hueco en su agenda.

Por otro lado, el usuario registrado y el administrador, podrá contactar con el dueño del inmueble. Usando el formulario de contacto, se enviará automáticamente un e-mail al dueño, y se guardará una copia de estos mensajes enviados.

Las webs inmobiliarias que hemos tomado como ejemplo, son unas de las más populares del mercado, se nombran a continuación:

- Enalquiler: www.enalquiler.com
- Fotocasa: www.fotocasa.es
- Fotoalquiler: www.fotoalquiler.com

A continuación, se muestra una imagen de la web Enalquiler:

The screenshot shows the Enalquiler website interface. At the top, there is a navigation bar with links for HOME, LA COMUNIDAD, ALERTAS, COMPARADOR PRECIOS, MI LISTA (0), and TRASLADO DE PISO. A button for 'Publica GRATIS tu anuncio' is also visible. The main search area is titled 'Buscar piso en alquiler' and includes filters for 'Tipo de vivienda' (Piso, Ático, Dúplex, Loft, Estudio, Casa / Chalet), 'Provincia' (Madrid, Barcelona, Málaga, Valencia / València, A Coruña, Alava), and 'Precio máximo / mes' (Indiferente, 500 €, 600 €, 800 €, 1.000 €, 1.200 €, 1.500 €). There are also dropdowns for 'Dormitorios' and 'Amueblado', and a checkbox for 'Con foto'. A 'Buscar' button is at the bottom of the search filters.

Below the search area, there is a 'La Comunidad' section with a 'Pregunta a La Comunidad' sub-section. Two questions are listed: '¿Los perros pueden subir en el ascensor?' and '¿Puede poner una cláusula el propietario para poder entrar a jardín y garaje?'. Each question has a 'Responder' button.

On the right side, there is a 'Síguenos en...' section with social media icons for Facebook, Twitter, and Google+. Below that is a 'Publica Gratis tu anuncio' section with a 'Añadir vivienda' button and an 'Acceso Propietarios' link. At the bottom right, there is a 'PUBLICIDAD' section for a blue car, 'AUTO SCOUT 24', with a 'Buscar' button.

Ilustración 1: Web Enalquiler.

Aquí se muestra una imagen de la web Fotocasa:



Ilustración 2: Web Fotocasa.

El portal inmobiliario Fotocasa, es muy conocido y tiene un elevado número de visitas. Lo primero que aparece en el portal son los términos de búsqueda, y una vez definidas las características, te muestra los inmuebles con dichas características.

En alquiler es uno de los portales inmobiliarios más importantes en la actualidad, una de las cosas que más me han gustado de este portal, es que en la misma página de inicio, muestra los inmuebles más relevantes, de precio más económico.

En la web Fotoalquiler, la característica que más destaca, es que eliges la ciudad en la que quieres buscar el inmueble con el mapa, y no puedes visitar los inmuebles sin elegir la población.

La web que se ha desarrollado en el proyecto, la novedad que aporta respecto a las anteriores, es que tiene el servicio de citas. Un servicio que sirve para quedar con el agente y concertar una visita a la vivienda, de forma muy sencilla y rápida, sólo se tiene que estar registrado para acceder a este servicio. Por otro lado, las páginas anteriores no te dejan tener contacto directo con el dueño. Otra novedad es el servicio de contacto, este servicio está a disposición de los usuarios registrados, y sirve para enviar mensajes directamente al dueño del inmueble.

1.3- Estructura

La estructura que se ha seguido para desarrollar este documento, y las fases que se han seguido para desarrollar el proyecto, se explican a continuación.

En primer lugar, se ha realizado la **especificación de requisitos**, en ésta se hace una captura de los requisitos que debe cumplir el proyecto, se ha hablado de los requisitos finales y de los usuarios.

A continuación, se trata el **análisis**. En cuanto al análisis, se reúnen todos los diagramas realizados, que describen la funcionalidad, los contenidos y la forma de interactuar.

El siguiente paso es el **diseño**, que describe el proyecto mediante la implementación de la web, con una determinada arquitectura. Este proyecto consta de tres niveles, que son:

- Interfaz
- Negocios
- Persistencia

Una vez finalizado el diseño, sigue la **implementación**. Durante la implementación se describe la tecnología, las herramientas y los tres niveles de arquitectura.

La evaluación, es el siguiente paso, en el que se evalúan los primeros resultados de la aplicación, se harán las pruebas oportunas para comprobar el correcto funcionamiento de la aplicación.

Y el documento finaliza con las **conclusiones**, donde se especifican los resultados obtenidos del proyecto.

2. Especificación de requisitos (ERS)

2.1- Propósito

El propósito de la especificación de requisitos, es establecer las características que requiere la aplicación, desarrollar su funcionalidad y satisfacer las necesidades de los diferentes usuarios.

En este apartado, se va a describir la gestión de un portal inmobiliario, y las mejoras que se pueden aportar mediante el software.

2.2- Ámbito

Este proyecto, que se llama “Aparta-Vacaciones”, es una web inmobiliaria y da soporte a los usuarios de la aplicación.

La aplicación da soporte a tres tipos de usuario, descritos anteriormente, y cada uno tiene acceso a una parte diferente, en total la aplicación consta de dos partes. Los usuarios no registrados, acceden a la parte pública, que es accesible para todo el mundo e incluye la información sobre los inmuebles en alquiler.

Los usuarios registrados y el usuario administrador, tienen acceso a la parte privada, y pueden realizar acciones diferentes según el tipo de usuario. Acciones como crear y modificar su perfil, servicio de citas, crear, modificar y borrar inmuebles, servicios y usuarios.

2.3- Definiciones, Siglas y Abreviaturas

- Sitio web: Es un conjunto de documentos y páginas web que tratan de un mismo tema, con una página de inicio. Tiene un dominio y una dirección de internet específicos.
- Servidor web: Servidor web o servidor HTTP es un programa, cuya función es procesar una aplicación del lado del servidor. Para ello, realiza conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente, y así genera o cede una respuesta en cualquier lenguaje o aplicación del lado del cliente.

2.4- Referencias

- "Una Guía para la Realización y Supervisión de Proyectos Final de Carrera (PFC) en el ámbito de la Web" de la Editorial UPV (REF. 247). ISBN 978-84-8363-325-0.
- Ejemplos de webs inmobiliarias: Enalquiler, Fotocasa, Fotoalquiler, Idealista.com.
- Otros Proyectos Final de Carrera.

2.5- Visión Global

Todo seguido, se va a proceder a explicar la descripción general de la web, con sus funciones, las características de cada usuario y las especificaciones de los requisitos que aparecen en el proyecto.

2.6- Descripción General

2.6.1- Perspectiva del Producto

Este portal inmobiliario, se ha desarrollado como herramienta de trabajo para los agentes inmobiliarios, a la vez que facilita la búsqueda de inmuebles a los clientes.

Para acceder a esta aplicación, es necesaria una conexión de internet, y usar un navegador web.

Los lenguajes de programación que se han usado, son PHP, JQuery, Java Script y HTML. Estos están vinculados al servidor con la base de datos MySQL. Todo ello, se ejecuta sobre el servidor Apache.

2.6.2- Funciones del Producto

Son dos las principales funciones, que se dividen en dos partes, una es la parte pública y la otra la parte privada. Cada parte tiene asignada un tipo de usuario. Por lo tanto, a continuación se citan las funciones que van a realizar cada usuario.

Parte Pública

Cualquier usuario que accede a la web, puede realizar diferentes acciones, sin registrarse, estas acciones se describen a continuación:

- Visitar página inicial: En la que pueden ver los inmuebles más relevantes, elegir la categoría que más les interese o ver más detalles del inmueble. Además pueden iniciar sesión.
- Visitar las ofertas: Consulta de las ofertas más relevantes.
- Buscar inmuebles: Se tiene la opción de buscar un apartamento, introduciendo la referencia, el precio o la categoría, etc.
- Contactar: Se puede consultar el teléfono de atención al cliente, la dirección, el mail, etc.
- Visitar inmuebles ordenados por categorías, como: chalet, piso, casa y apartamento.

Parte Privada

Usuario registrado

Tienen los mismos privilegios que el usuario anterior, pero además, pueden:

- Visitar “Mi perfil”: Se modifican o consultan los datos del usuario registrado.
- Contactar: Se rellena un formulario para contactar con el dueño del inmueble.
- Crear citas: Se crea una cita para visitar un inmueble.
- Consultar citas: Se consulta una o varias citas que han sido creadas anteriormente.
- Eliminar citas: Se eliminan las citas que ya han pasado de fecha, o a las que no se podrá asistir.
- Gestionar mensajes: Se pueden ver y borrar, pero sólo sus propios mensajes.

Usuario administrador

Tienen los mismos privilegios que los dos usuarios anteriores, pero además, pueden:

- Gestionar inmuebles: Se crea un nuevo inmueble, se modifica uno ya existente, una vez se ha seleccionado o se elimina.
- Crear servicio Cita/Mensaje: Se crea un nuevo servicio.
- Modificar servicio Cita/Mensaje: Se modifica el servicio seleccionado, y se cambian los datos existentes por otros nuevos.
- Borrar servicio Cita/Mensaje: Se elimina un servicio seleccionado.
- Contactar: Se rellena el formulario para contactar por mail con el dueño del inmueble.
- Ver mensajes: Consultar los mensajes que ha enviado, y los que han enviado los demás usuarios registrados.
- Eliminar mensajes: Se elimina los mensajes deseados, los creados por el administrador o los creados por otros usuarios registrados.
- Crear Usuario: Se crea un usuario nuevo (registrado o administrador)
- Borrar Usuario: Se elimina un usuario ya existente.
- Modificar Usuario: Modificar todos los usuarios ya existentes.

2.6.3- Características del Usuario

- Usuarios no registrados: Pueden ver y buscar inmuebles, además de registrarse. Solo acceden a la parte pública de la web.
- Usuarios registrados: Éstos tienen todos los privilegios del usuario no registrado, y además tienen la posibilidad de modificar o consultar su propio perfil, contactar con el dueño del inmueble y crear, consultar o eliminar citas a través del servicio de citas. También pueden eliminar sus mensajes, que se crean al contactar con el dueño. Accede a la parte privada de la web.

- Usuario administrador: Tiene los privilegios de los anteriores, y además puede crear, modificar y borrar los inmuebles, los servicios y los usuarios, ver y eliminar sus mensajes y los mensajes de los otros usuarios. En cuanto a las citas, también podrá ver y eliminar sus citas y la de los demás usuarios. Accede a la parte privada de la web.

2.6.3- Restricciones

Esta web se ha desarrollado para que funcione con cualquier navegador, así que para acceder al portal inmobiliario, sólo se necesita una conexión a internet.

2.6.4- Supuestos y Dependencias

Esta aplicación no necesita un hardware específico, ya que el mismo servidor es el encargado de darnos acceso a la aplicación y de interpretar los scripts PHP.

Es aquí donde encontramos la única dependencia, con el servidor, en este caso Apache.

El servidor web, tendrá las siguientes funciones:

- Procesar los scripts PHP.
- Proporcionar un servidor de base de datos.
- Disponer de un servidor FTP a la hora de transferir archivos.

2.7- Requisitos Específicos

2.7.1- Requisitos de Interfaces Externos

- Interfaces de usuario

La aplicación tiene un diseño único para todos los usuarios, tal como vemos en las siguientes imágenes.

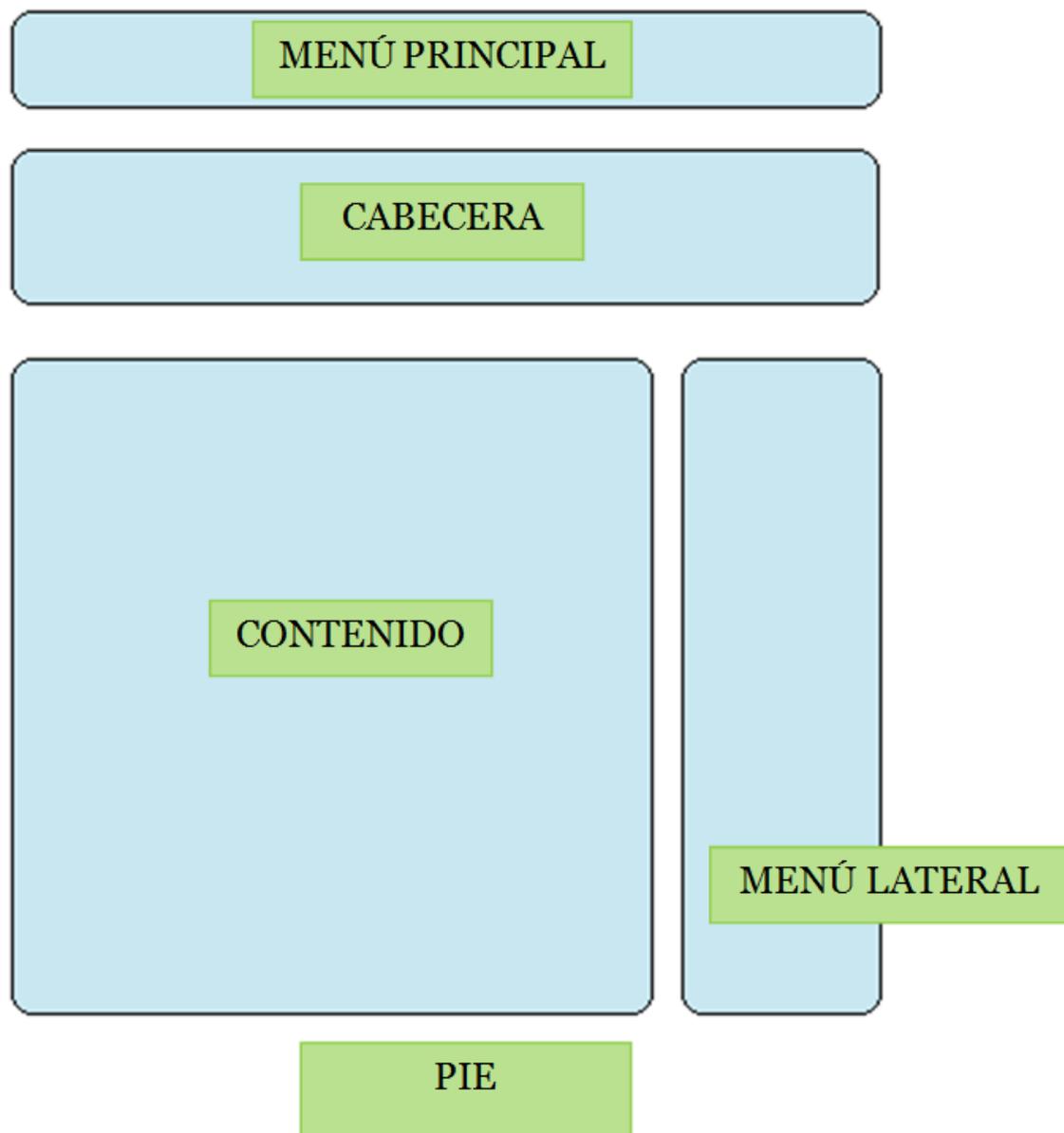


Ilustración 3: Interfaz de Usuario Parte Pública.

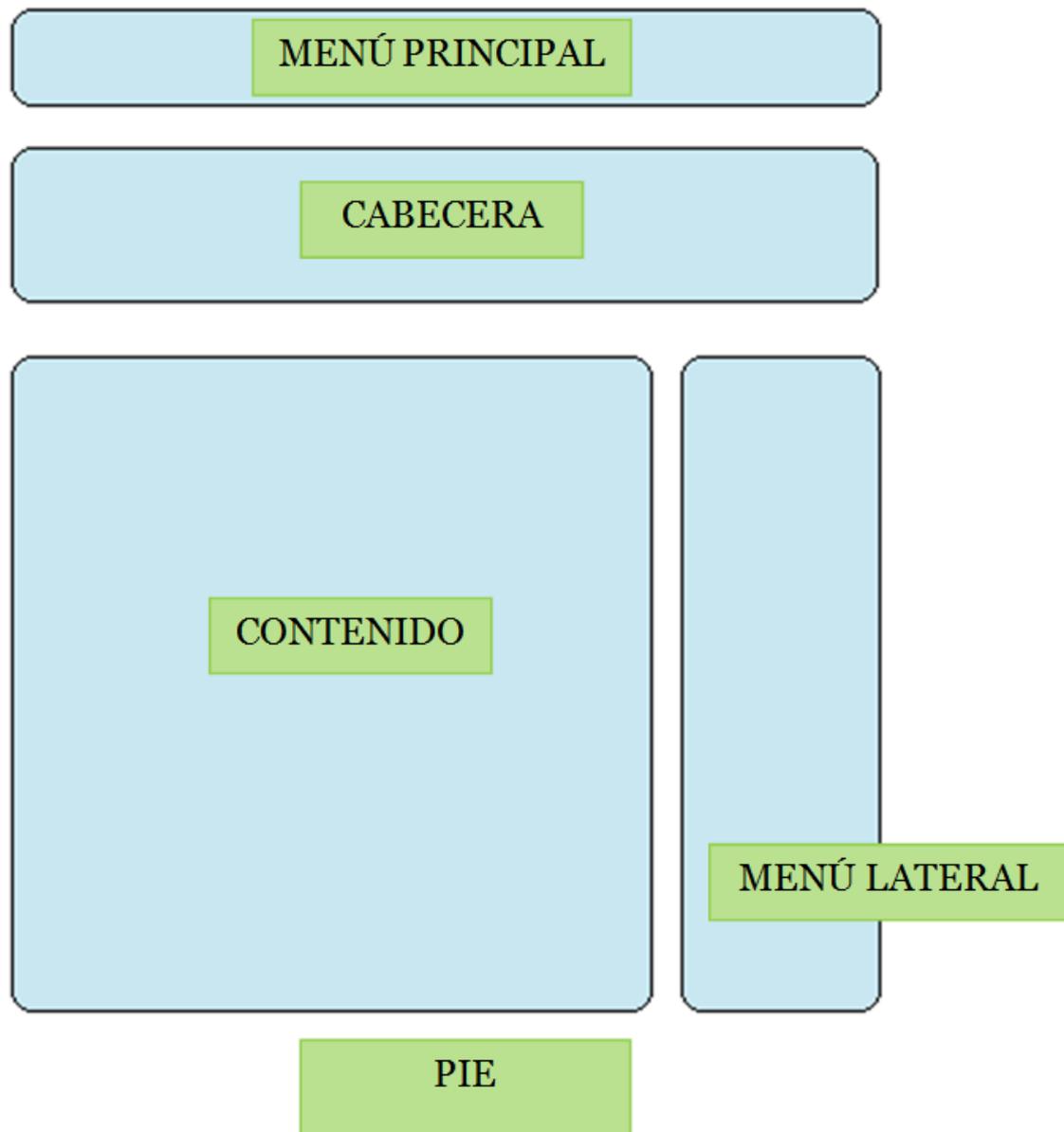


Ilustración 4: Interfaz de Usuario Parte Privada.

- Interfaces de hardware

Al ser una aplicación web, no importa el hardware que se use.

- Interfaces de software

Para que funcione la aplicación, sólo necesitará una conexión a internet y un navegador web, ya que la aplicación se ha desarrollado en PHP, JQuery, Java Script y HTML. Y la base de datos se basa en el lenguaje MySQL.

- Interfaces de comunicaciones

Las comunicaciones se efectuarán siguiendo el protocolo HTTP mediante conexiones TCP/IP.

2.7.2- Requisitos Funcionales

Aquí se describen las funciones de la web, se han clasificado según el tipo de usuario:

- Usuarios no registrados

Ver información:

Propósito	Ver el portal web.
Entrada	-
Proceso	Se carga la web.
Salida	Se visualiza la web.

Buscar inmuebles:

Propósito	Buscar un inmueble.
Entrada	Referencia, Categoría, Precio, Dormitorios, Numero de Camas, Plazas, Distancia del Mar, Municipio o Provincia
Proceso	Se carga el apartamento que cumple con las condiciones de la búsqueda.
Salida	Se visualiza la el resultado de la búsqueda.

Registrarse:

Propósito	Crear perfil del usuario.
Entrada	Foto del usuario, Nombre, Usuario, Contraseña, Email
Proceso	Se almacenan los datos introducidos.
Salida	Se visualiza un mensaje de bienvenida.

- Usuarios registrados

Este usuario tiene los mismos privilegios del usuario no registrado (a excepción de registrarse, porque ya está registrado) y además los siguientes privilegios:

Iniciar sesión:

Propósito	Identificarse en la aplicación web.
Entrada	Nombre de usuario y contraseña.
Proceso	Verificar los credenciales introducidos.
Salida	Se accede a la parte privada, si son correctos los datos introducidos.

Ver información:

Propósito	Ver el portal web.
Entrada	-
Proceso	Se carga la web.
Salida	Se visualiza la web.

Modificar perfil:

Propósito	Modificar los datos existentes del perfil
Entrada	Datos personales del usuario y contraseña.
Proceso	Se modifican los datos existentes por los deseados.
Salida	Se modifican los datos correctamente o falla la modificación.

Consultar perfil:

Propósito	Consultar sus datos del perfil
Entrada	Datos personales del usuario.
Proceso	Se recuperan los datos del usuario de la base de datos.
Salida	Se muestran los datos personales del usuario.

Contactar con el dueño del inmueble:

Propósito	Contactar con el dueño del inmueble
Entrada	Datos del usuario y el mensaje.
Proceso	Se manda un mensaje al administrador, para que sepa que un usuario registrado quiere contactar con él.
Salida	Se muestra un mensaje, en el que dice si se ha realizado con éxito o ha fallado.

Crear cita:

Propósito	Pedir cita con el agente inmobiliario.
Entrada	Datos de la cita y del usuario.
Proceso	Mostrar la cita de consulta.
Salida	La cita se muestra de forma correcta o falla.

Consultar cita:

Propósito	Consultar la cita almacenada anteriormente.
Entrada	Datos de la cita.
Proceso	Mostrar la cita de consulta.
Salida	La cita se muestra de forma correcta o falla.

Modificar cita:

Propósito	Modificar fecha de la cita.
Entrada	Datos de la cita.
Proceso	Se modifica la fecha existente por los nuevos datos.
Salida	Se modifica la cita o falla la modificación.

- Usuario administrador

Inicio sesión:

Propósito	Identificarse en la aplicación web.
Entrada	Nombre de usuario y contraseña.
Proceso	Verificar los credenciales introducidos.
Salida	Se accede a la parte privada, si son correctos los datos introducidos.

Mostrar información:

Propósito	Ver los inmuebles.
Entrada	-
Proceso	Se carga la web.
Salida	Se visualiza la web.

Crear Usuarios:

Propósito	Crear perfil del usuario.
Entrada	Datos personales del usuario y una contraseña.
Proceso	Se almacenan los datos introducidos.
Salida	Se visualiza el listado de los usuarios.

Crear inmueble o servicio:

Propósito	Crear un nuevo inmueble.
Entrada	Datos del inmueble.
Proceso	Insertar inmueble en la base de datos.
Salida	Inmueble insertado correctamente o no.

Modificar inmueble:

Propósito	Modificar datos del inmueble o servicio.
Entrada	Datos del inmueble o servicio.
Proceso	Se modifican datos existentes por los nuevos datos introducidos.
Salida	Se modifican los datos o falla la modificación.

Borrar inmueble:

Propósito	Borrar uno o varios inmuebles.
Entrada	Referencia apartamento.
Proceso	Eliminar registro de la base de datos.
Salida	Se elimina de forma correcta o incorrecta.

Ver mensajes:

Propósito	Ver los mensajes.
Entrada	-
Proceso	Mostrar los mensajes
Salida	Se muestran los mensajes de forma correcta o falla.

Ver usuario:

Propósito	Ver los datos del usuario.
Entrada	Datos personales usuario.
Proceso	Mostrar el usuario de consulta.
Salida	El usuario se muestra de forma correcta o falla.

2.7.3 Obligaciones de diseño

- Estándares cumplidos:

Esta web sigue los estándares de cualquier web, además tiene acceso seguro, así sólo pueden acceder a la privada los usuarios con permiso mediante contraseña. El idioma de la web es el castellano.

- Limitaciones hardware:

Al ser una web, no necesita un hardware específico. El servidor donde se encuentra la web, debe estar conectado a internet las 24 horas del día.

2.7.4- Atributos

- Seguridad

Es importante la seguridad en esta web, para que no todos los usuarios accedan a la administración, etc. Así pues, cualquier visitante que quiera acceder a la parte privada tendrá que identificarse. Los usuarios registrados y el administrador tendrán un nombre de usuario y contraseña. La información de acceso se guarda en la base de datos. Una vez se haya logado el usuario registrado o el administrador, se les re direccionará a la parte privada de la web.

- Facilidades de mantenimiento

El administrador se encargará del mantenimiento de la web. Pero si quiere hacer algún cambio y requiere un cambio de la base de datos, debe acudir al programador de la web.

- Portabilidad

Cualquier sistema operativo o plataforma puede soportar la aplicación, ya que se ha diseñado con tecnología libre. Por este motivo, se puede acceder a la web desde cualquier navegador.

- Otros Requisitos

La base de datos que usa el portal es MySQL, es aquí donde se almacena toda la información de la web. Mediante PHP y su API de acceso a bases de datos MySQL, el servidor web hace las consultas a la base de datos.

Hay que tener en cuenta, que para poder introducir datos en la base de datos, es necesario ser un usuario registrado o el administrador, es por esto que la aplicación se entregará con un usuario ya creado.



3. Análisis

Para desarrollar el análisis se ha usado el lenguaje gráfico UML (Unificado de Modelado), que es el más usado hoy en día. UML es un lenguaje gráfico, que ofrece un estándar para describir el plano del sistema; incluyendo procesos, funcionamiento, expresiones de lenguajes de programación y esquemas de base de datos y componentes reutilizables.

Nos vamos a centrar en tres tipos de diagramas UML:

- Diagrama de clases.
- Diagrama de casos de uso.
- Diagrama de secuencia.

3.1- Diagrama de clases

El diagrama de clases, que lo podemos ver en la figura que hay a continuación, nos sirve para definir la estructura de la aplicación, visualizando las relaciones que hay entre las clases y los atributos.

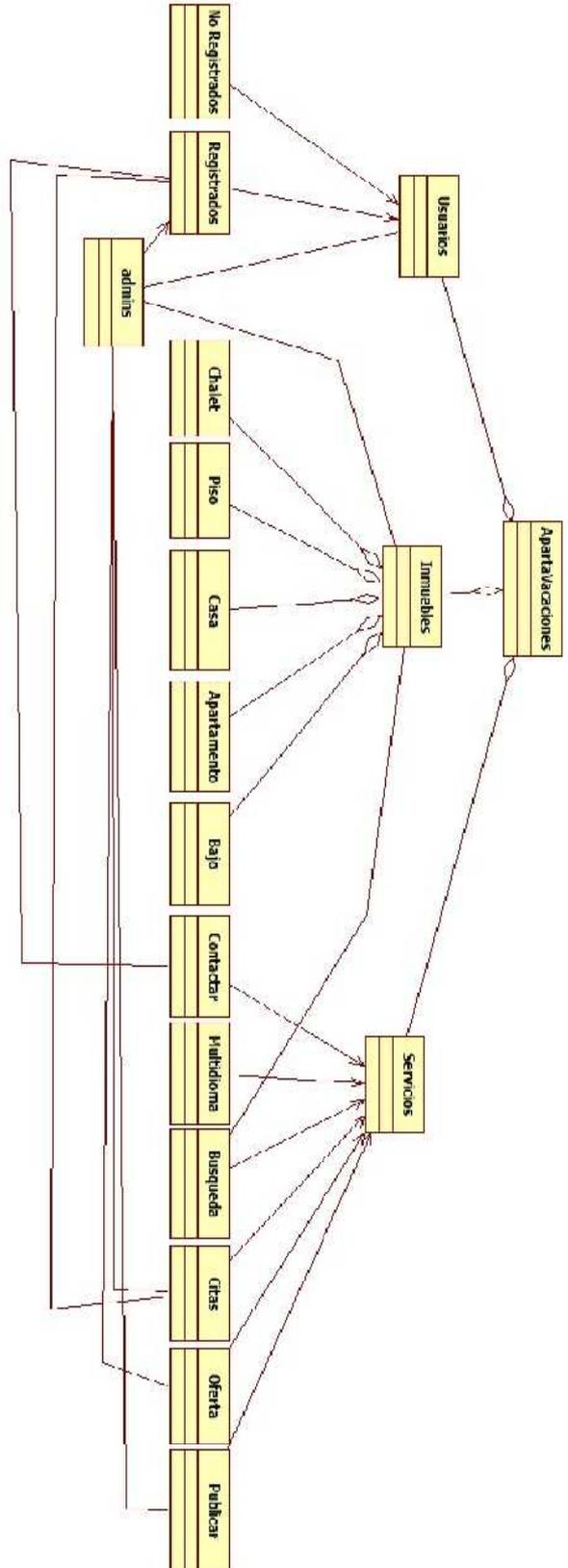


Ilustración 5: Diagrama de clases.

3.2- Diagrama de casos de uso

Con el diagrama de casos de uso, se puede observar los tipos de actores que acceden a la web, sus acciones, una vez están haciendo uso de la aplicación y las relaciones que hay entre ellos.

En el diagrama que hay a continuación, se observa los distintos tipos de usuarios, todos se generan a raíz del actor usuario:

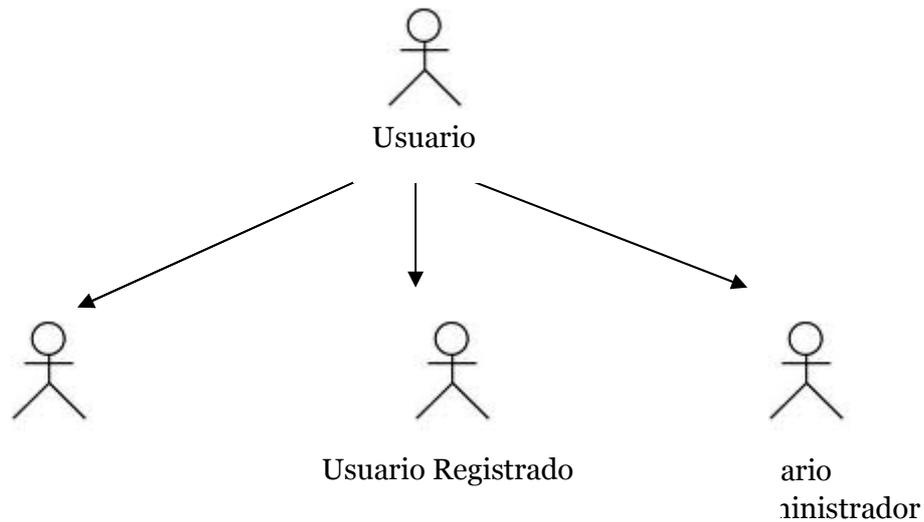


Ilustración 6: Actores del sistema

En el diagrama que hay a continuación, se observan las diferentes funciones de cada usuario:

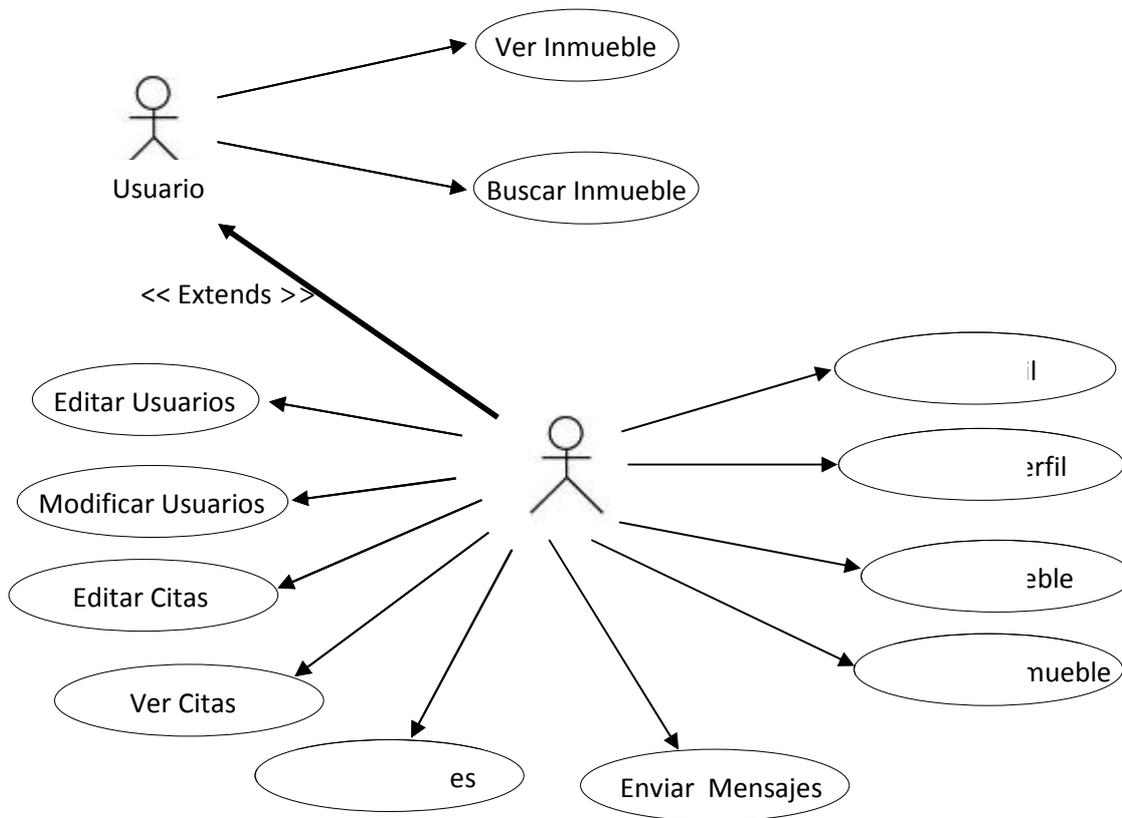


Ilustración 7: Casos de uso general.

3.3- Diagrama de secuencia.

Este diagrama, se usa para distinguir las operaciones que se producen dentro del mismo sistema y las operaciones de los usuarios.

Así pues, estos son los diagramas que corresponden a esta aplicación:

3.3.1- Iniciar sesión

Este diagrama de secuencia, representa el inicio de sesión, que será uno de los más usados, ya que cada usuario registrado, tiene que iniciar sesión para poder acceder a la parte privada.

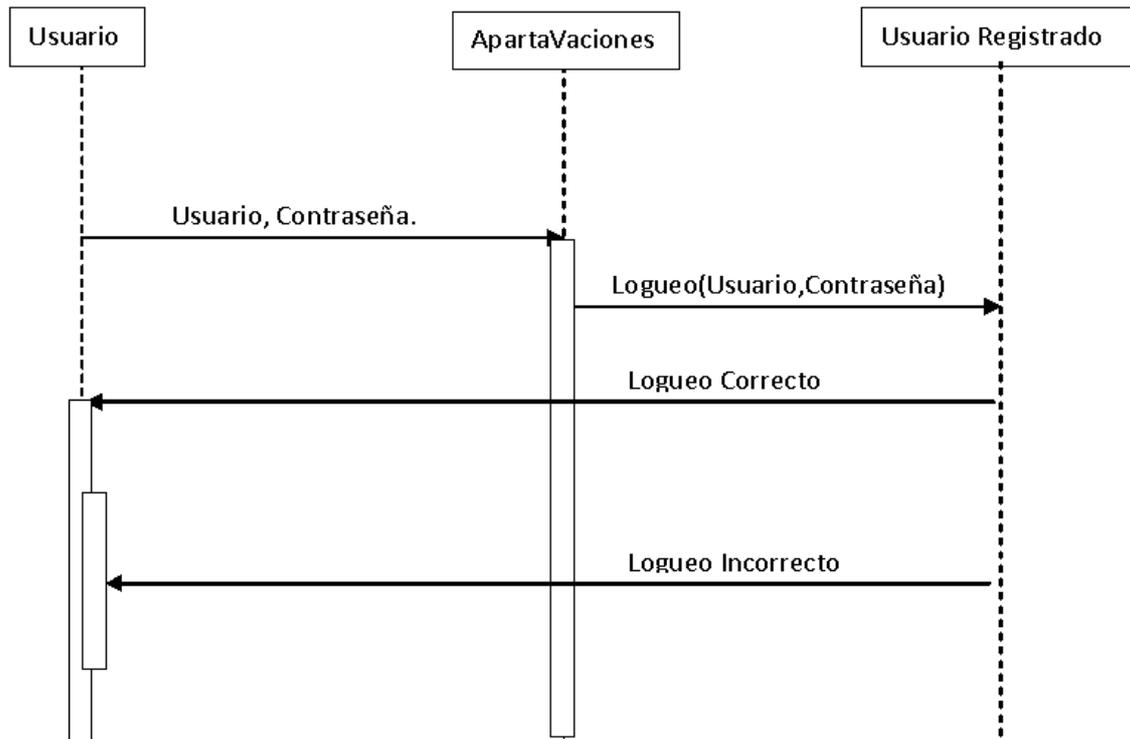


Ilustración 8: Diagrama de secuencia, iniciar sesión

3.3.2- Crear inmueble

Para este caso, el usuario registrado crea y publica un anuncio en la web, pero para ello tiene que introducir imágenes del inmueble, darle una referencia, poner el precio, la descripción, etc.

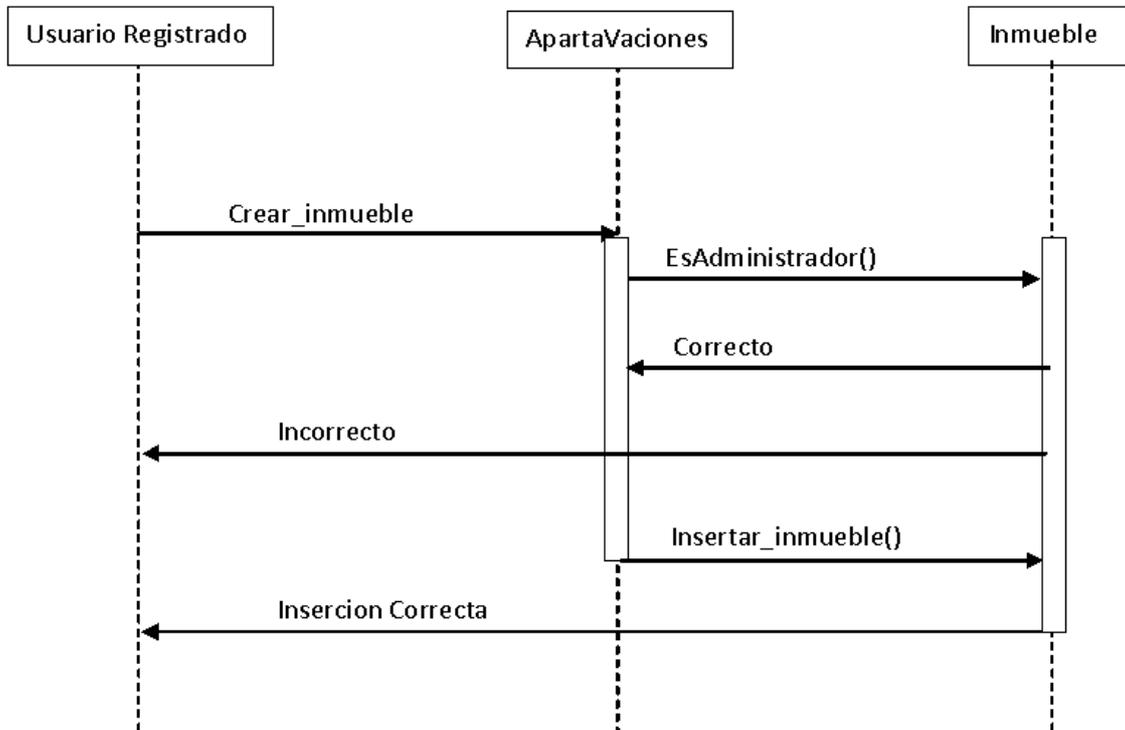


Ilustración 9: Diagrama de secuencia, crear inmueble.

3.3.3- Eliminar inmueble

Para eliminar inmueble, sólo lo puede hacer el usuario administrador, que una vez haya iniciado la sesión, lo puede eliminar, se puede ver a continuación, en el diagrama:

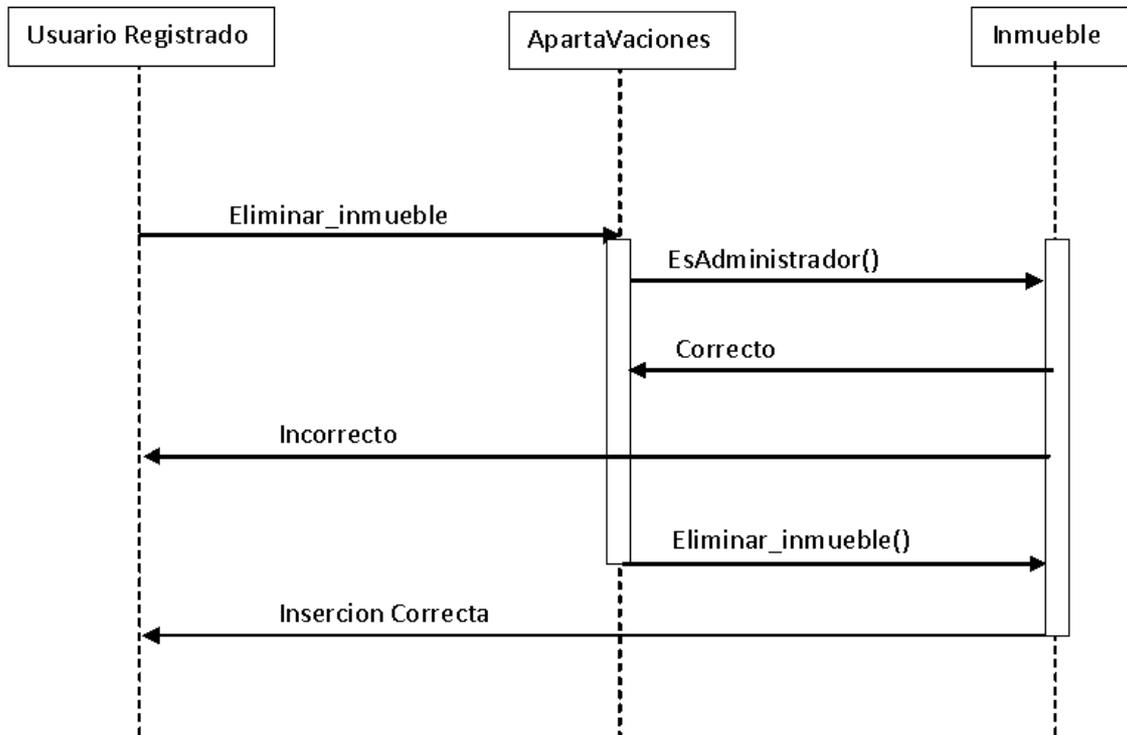


Ilustración 10: Diagrama de secuencia, eliminar inmueble.

4. Diseño

Como diseño de esta web, se ha elegido una arquitectura de tres capas, considerando que, de este modo los cambios en una de las capas, no repercutirá en las demás. Así, se asegura la independencia entre las distintas capas.

Las tres capas son:

- Interfaz
- Negocios
- Persistencias

La estructura de directorios y ficheros será la siguiente:

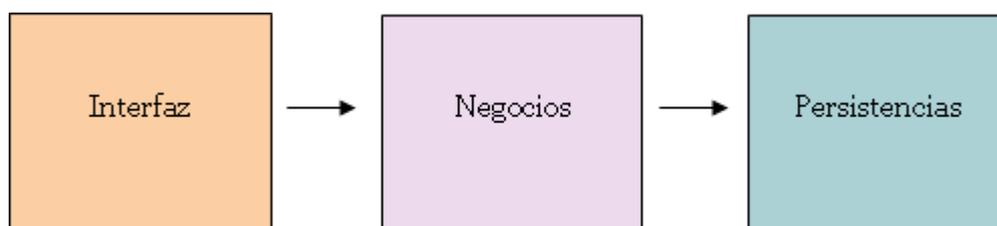


Ilustración 11: Arquitectura tres capas.

4.1- Capa Interfaz

Esta capa es la que se muestra al usuario, e interactúa con él, también capta toda la información que éste introduce.

Es importante que esta capa sea muy intuitiva y fácil de usar, ya que el usuario interactuará sobre ella.

Para hacer más clara la interfaz al usuario, lo que se ha hecho es diferenciar una zona privada de otra pública.

4.1.1- Parte Pública

Como podemos ver más abajo, en la imagen, a la parte pública de la web, puede acceder cualquier visitante, es decir, cualquier tipo de usuario.

Los contenidos a los que se puede acceder son, por ejemplo; ver los inmuebles, las imágenes, las diferentes categorías, etc.

Al abrir la web, lo primero que aparece es la página de inicio, como vemos a continuación.

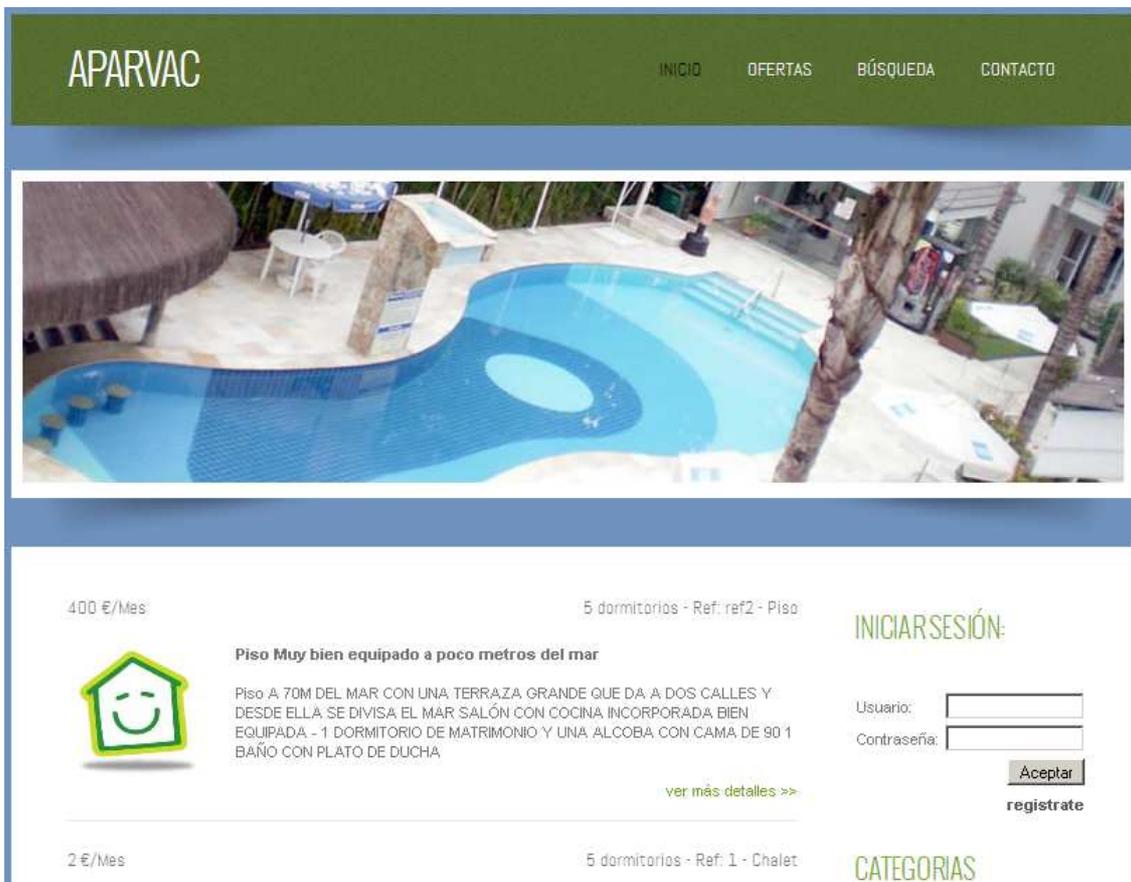


Ilustración 12: Página de inicio.

En la página de inicio, hay dos menús, el de la parte de arriba, contiene:

- Ofertas
- Búsqueda
- Contacto

Y otro menú, en la parte derecha, en el que se puede:

- Iniciar sesión o registrarse
- Visitar las diferentes categorías, en las que se dividen los apartamentos.

El menú de arriba, como es más importante, se ha puesto en la parte más visible de la web, y el menú de la derecha, se ha colocado ahí, para que antes de moverse por las diferentes categorías, vean los inmuebles de la primera página.

The screenshot displays a real estate website interface. On the left, there are four property listings, each with a price per month, a title, a description, and a 'ver más detalles >>' link. The listings are: 1) 'Piso Muy bien equipado a poco metros del mar' for 400 €/Mes, 5 dormitorios, Ref: ref2 - Piso; 2) '3' for 2 €/Mes, 5 dormitorios, Ref: 1 - Chalet; 3) 'Chalet a todo lujo' for 2100 €/Mes, 2 dormitorios, Ref: 0120 - Chalet; 4) 'Chalet primera linea del mar' for 2900 €/Mes, 1 dormitorio, Ref: 0121 - Chalet. On the right side, there is a 'INICIAR SESIÓN:' section with fields for 'Usuario:' and 'Contraseña:', an 'Aceptar' button, and a 'regístrate' link. Below that is a 'CATEGORIAS' section with a list of categories: 'Chalet', 'Piso', 'Casa', 'Apartamento', and 'Bajo'.

Ilustración 13: Menú derecha.

4.1.2- Parte privada

A la parte privada de la web, sólo tienen acceso los usuarios registrados y el administrador. Aunque las acciones de estos usuarios son diferentes entre sí, tienen el mismo sistema.

Para acceder a la sesión, el Inicio de Sesión, se encuentra en la parte derecha de la web, accesible para todos los usuarios, al introducir el nombre de usuario y la contraseña, el sistema detecta inmediatamente el tipo de usuario.

INICIARSESIÓN:

Usuario:

Contraseña:

Ilustración 14: Inicio de sesión.

En la parte privada del administrador (que tiene los mismos permisos que el usuario registrado y además puede crear, modificar y borrar los inmuebles y servicios, ver y eliminar tanto sus mensajes como el de los otros usuarios y puede ver los usuarios) tiene su propio menú.

Así pues, para realizar todas estas acciones, en su sesión tiene el siguiente menú, que las podemos ver en la imagen que hay a continuación:

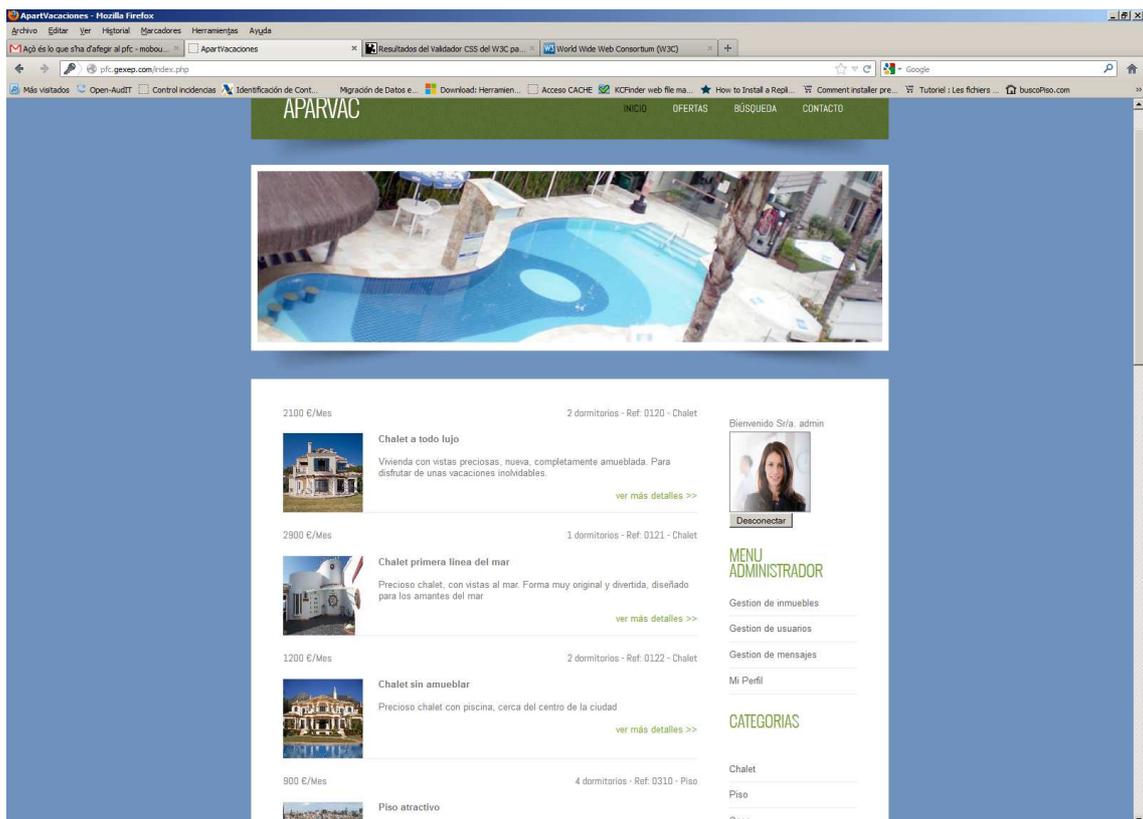


Ilustración 15: Estructura del usuario administrador

Por otro lado, el usuario registrado, que tiene menos permisos que el administrador (privilegios del usuario no registrado y la posibilidad de modificar o consultar su

propio perfil, contactar con el dueño del inmueble y crear o consultar citas a través del servicio de citas. Pueden ver y eliminar todos los mensajes) en su sesión tiene el siguiente menú:

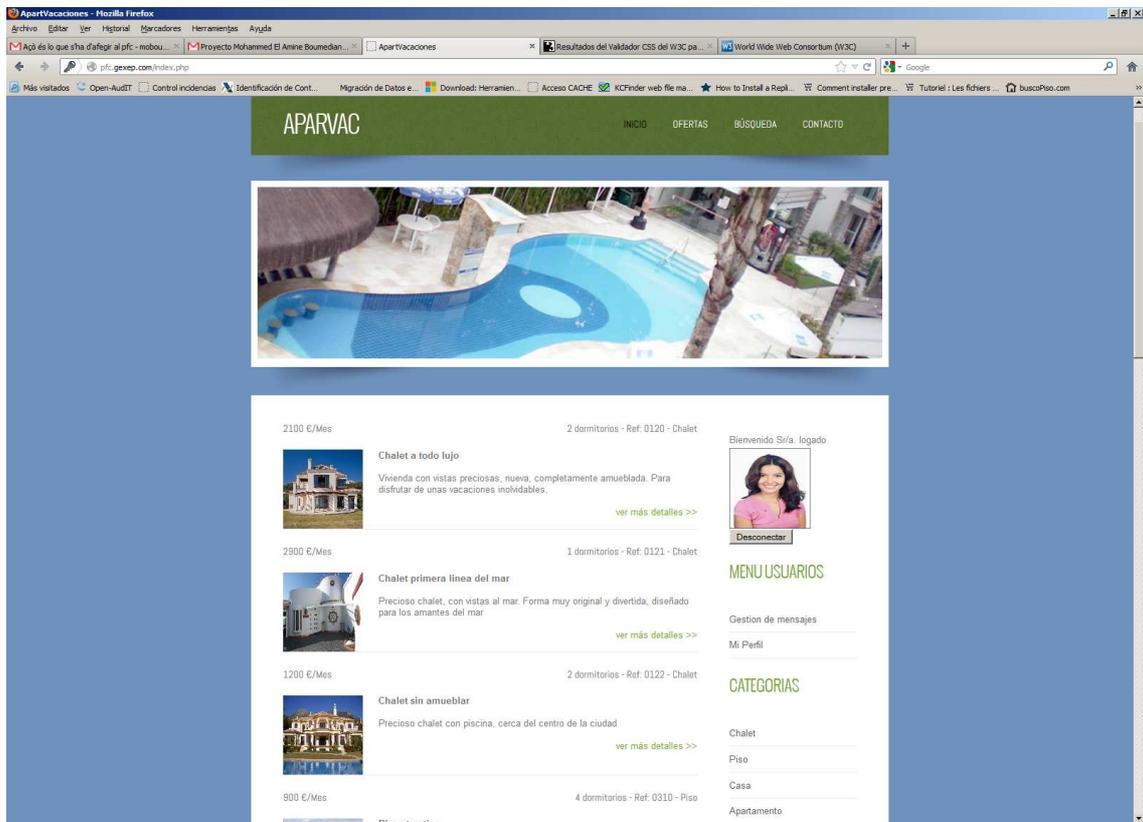


Ilustración 16: Estructura del usuario registrado

4.2- Capa de negocio

La funcionalidad de la capa de negocio, es interactuar entre la capa interfaz y la persistencia. Esta capa es la encargada de dar la funcionalidad a la aplicación.

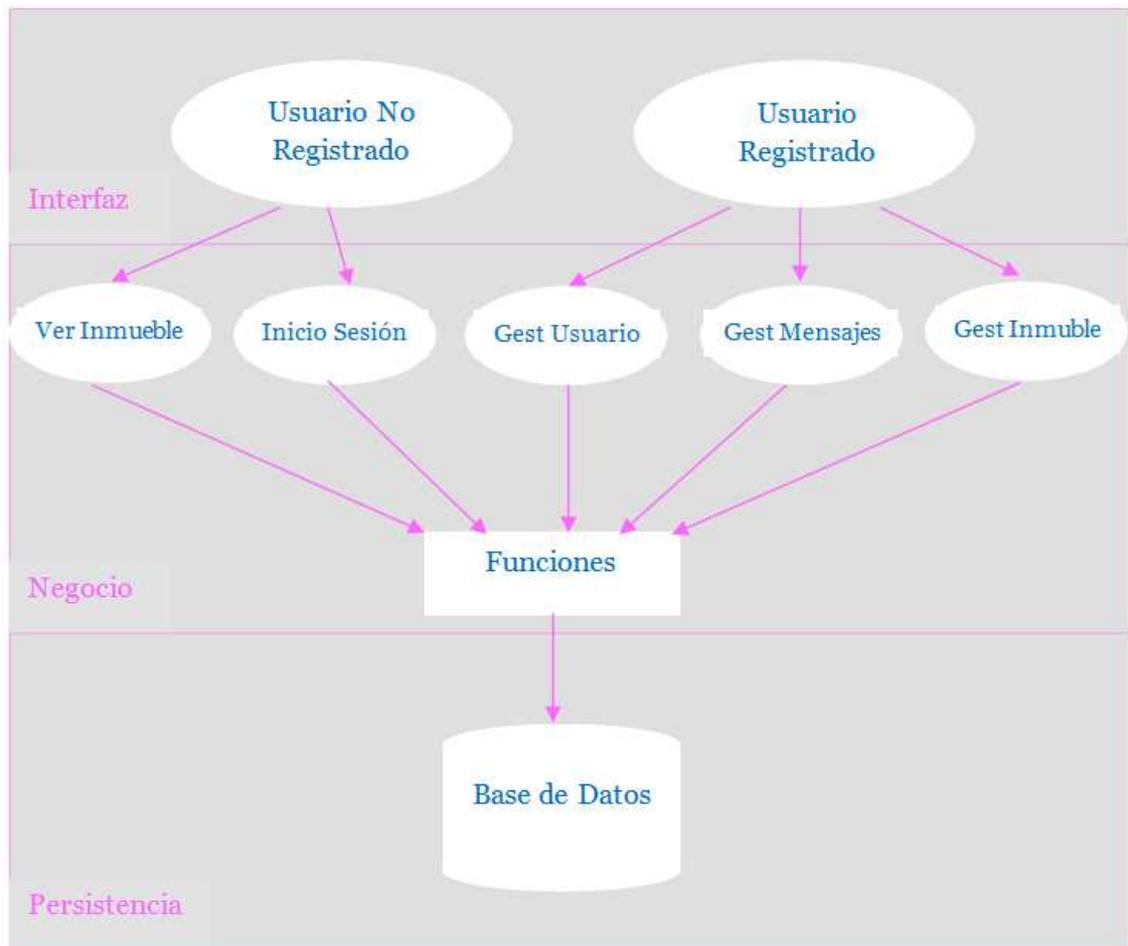


Ilustración 17: Arquitectura de capas

4.3- Capa de persistencia

Esta capa está compuesta por la base de datos. El diseño de esta capa no ha sido sencillo, pero finalmente se ha adoptado este modelo, ya que da mejor funcionalidad a la aplicación. Seguidamente, se puede ver el diagrama, con diversas relaciones entre las tablas de la base de datos.

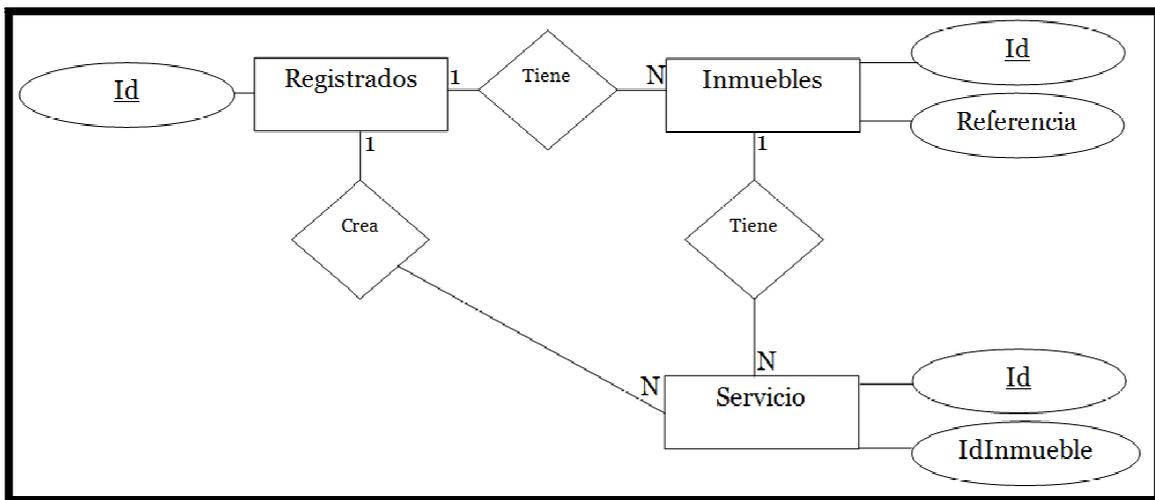


Ilustración 18: Modelo Entidad-Relacion

5. Implementación

Son varias las tecnologías que se han usado para desarrollar la aplicación web, éstas se describen a continuación.

5.1- Tecnologías de implementación web

Los lenguajes que se han usado para la programación son PHP, JQuery, Java Script y HTML. La base de datos sobre la que se ha trabajado es MySQL, y todo esto se ejecuta sobre el servidor Apache.

A continuación se procede a la explicación de cada uno.

5.1.1- PHP

PHP (*Hypertext Pre-processor*) es un lenguaje de programación interpretado o *framework* para HTML, diseñado originalmente para la creación de páginas web dinámicas. El uso que tiene en la actualidad, es para la creación de otros programas, como por ejemplo, las aplicaciones que tienen una interfaz gráfica, usando las bibliotecas Qt o GTK+.

Una de las cosas que cabe destacar es que se considera software libre. La mayoría de los servidores web, los sistemas operativos o las plataformas lo despliegan, sin coste alguno.

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones.

Lo que distingue a PHP de algo lado-cliente como Javascript, es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá los resultados de ejecutar el script, sin ninguna posibilidad de determinar qué código ha producido el resultado recibido. El servidor web puede ser incluso configurado para que procese todos los archivos HTML con PHP y entonces no hay manera que los usuarios puedan saber que tienes debajo de la manga.

5.1.2- JQuery

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

jQuery es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privativos.¹ jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

5.1.3- Java Script

JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

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

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. JavaScript se interpreta en el agente de usuario, al mismo tiempo que las sentencias van descargándose junto con el código HTML.

5.1.4- MySQL

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones.

Está desarrollado en su mayor parte en ANSI C.

MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

5.1.5- Apache

El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.12 y la noción de sitio virtual. Cuando comenzó su desarrollo, se basó inicialmente en el popular código NCSA HTTPd 1.3, pero más tarde fue reescrito por completo.

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

5.2- Descripción de la implementación

5.2-1. Capa de presentación

En esta capa se han usado tecnologías como HTML, CSS y JAVASCRIPT para darle forma y estructura al portal.

A continuación se verá como se implementado la cabecera en html:

```
C:\Users\aboumediane\Desktop\PFC\Backups\CodigoFinal\Final\includes\header.php - Notepad++
Archivo Editar Buscar Vista Codificación Lenguaje Configuración Macro Ejecutar Plugins Ventana ?
header.php
1 <div id="header-wrapper" class="container">
2   <div id="header" class="container">
3     <div id="logo">
4       <h1><a href="index.php">aparvac</a></h1>
5     </div>
6     <div id="menu">
7       <ul>
8         <li class="current_page_item"><a href="index.php">Inicio</a></li>
9         <li><a href="ofertas.php">Ofertas</a></li>
10        <li><a href="busqueda.php">Búsqueda</a></li>
11        <li><a href="contacto.php">Contacto</a></li>
12      </ul>
13    </div>
14  </div>
15  <div></div>
16 </div>
```

Ilustración 19: Código de la cabecera en HTML

```
C:\Users\aboumediane\Desktop\PFC\Backups\CodigoFinal\Final\style.css - Notepad++
Archivo Editar Buscar Vista Codificación Lenguaje Configuración Macro Ejecutar Plugins Ventana
header.php style.css
1 body {
2   margin: 0;
3   padding: 0;
4   background: #C5DA99 url(images/img01.jpg) repeat;
5   font-family: Arial, Helvetica, sans-serif;
6   font-size: 14px;
7   color: #787878;
8 }
9
10 h1, h2, h3 {
11   margin: 0;
12   padding: 0;
13   text-transform: uppercase;
14   font-weight: normal;
15   font-family: 'Oswald', sans-serif;
16   font-weight: 200;
17   color: #79A62E;
18 }
19
20 h1 {
21   font-size: 2em;
22 }
23
24 h2 {
25   font-size: 2.8em;
26 }
27
28 h3 {
29   font-size: 1.6em;
30 }
31
32 p, ul, ol {
33   margin-top: 0;
34   line-height: 180%;
```

Ilustración 20: Ejemplo del código CSS

En la siguiente imagen se verá como ha quedado reflejada la asociación entre los selectores y los elementos:



Ilustración 21: Menú superior

5.2-2. Capa de aplicación

En esta capa, se ha implementado mediante el lenguaje php, aunque con el objetivo de mejorar la apariencia de la web, que se vea bien en todos los navegadores y sea más dinámica, también se ha usado el lenguaje HTML, Javascript, JQuery.

Todo esto, se puede observar con estas imágenes del código:

```
C:\Users\aboumediane\Desktop\PFC\CodigoFinal\Final\Clases\inmueble.class.php - Notepad++
Archivo  Editar  Buscar  Vista  Codificación  Lenguaje  Configuración  Macro  Ejecutar  Plugins  Ventana  ?
DDBB.SQL  Untitled.xml  index.php  editar.php  guardarinmueble.php  FileSizeControlierVASS.vbs  inmueble.class.php
1  <?php
2  /*****inmueble*****/
3  class inmueble{
4      private $Id;
5      private $Referencia;
6      private $Categoria;
7      //private $Propietario;
8      private $Titulo;
9      private $Precio;
10     private $Descripcion;
11     private $Dormitorios;
12     private $NumCamas;
13     private $Plazas;
14     private $DisMar;
15     private $Municipio;
16     private $Provincia;
17     private $Publicado;
18     private $Oferta;
19     private $Image1;
20     private $Image2;
21     private $Image3;
22     private $Image4;
23
24
25     function __construct($id="", $Referencia="", $Titulo="", $Precio="", $Descripcion="")
26     {
27         $this->id=$id;
28         $this->Referencia=$Referencia;
29         $this->Categoria=$Categoria;
30         $this->Titulo=$Titulo;
31         $this->Precio=$Precio;
32         $this->Descripcion=$Descripcion;
33         $this->Dormitorios=$Dormitorios;
34         $this->NumCamas=$NumCamas;
35         $this->Plazas=$Plazas;
36         $this->DisMar=$DisMar;
37         $this->Municipio=$Municipio;
38         $this->Provincia=$Provincia;
39         $this->Publicado=$Publicado;
40         $this->Image1=$Image1;
41         $this->Image2=$Image2;
42         $this->Image3=$Image3;
43         $this->Image4=$Image4;
44     }

```

Ilustración 22: Código php Clase Inmueble

En la imagen anterior se describe como hemos creado el objeto inmueble para poder crear, eliminar y modificar inmuebles de la capa de negocio.

```

1  <? //session_start();?>
2  <?php
3
4  include_once("conexion.class.php");
5  include_once("inmueble.class.php");
6  include_once("usuarios.class.php");
7  include_once("servicios.class.php");
8
9  class apartavacaciones{
10     var $con;
11     function apartavacaciones(){
12         $this->con=new DBManager;
13     }
14
15     function Consultar(){
16         if($this->con->conectar()==true){
17             return mysql_query("select * from inmuebles");
18         }
19     }
20
21     function Consultar_inmuebles(){
22         $Idarray=0;
23
24         if($this->con->conectar()==true){
25             $result= mysql_query("select * from 'inmuebles'");
26             while ($row = mysql_fetch_assoc($result)) {
27                 $inmueble=new inmueble;
28                 $inmueble->setid($row["Id"]);
29                 $inmueble->setReferencia($row["Referencia"]);
30                 $inmueble->setCategoria($row["Categoria"]);
31                 $inmueble->setTitulo($row["Titulo"]);
32                 $inmueble->setPrecio($row["Precio"]);
33                 $inmueble->setDescripcion($row["Descripcion"]);
34
35                 $inmueble->setDormitorios($row["Dormitorios"]);
36                 $inmueble->setNumCamas($row["NumCamas"]);
37                 $inmueble->setPlazas($row["Plazas"]);
38                 $inmueble->setDisMar($row["DisMar"]);
39                 $inmueble->setMunicipio($row["Municipio"]);
40                 $inmueble->setProvincia($row["Provincia"]);
41                 $inmueble->setPublicado($row["Publicado"]);
42                 $inmueble->setOferta($row["Oferta"]);
43                 if(empty($row["Image1"]))$inmueble->setImage1("image-apartamentos/Default.jpg");

```

Ilustración 23: Código PHP clase apartavacaciones

En la ilustración 12 se muestra parte del código responsable de crear y gestionar los objetos del proyecto.

5.2-3. Capa de persistencia

El objeto Conexión, se encuentra entre la capa de negocio y la de persistencia. Ésta interactúa entre los objetos y la base de datos MySQL.

Para que reflejar el uso de las tecnologías que se han usado en la implementación de la aplicación web, vamos a dejar unas imágenes a continuación.

Aquí se puede observar el código:

```
*C:\Users\aboumediane\Desktop\PFC\CodigoFinal\Final\Clases\conexion.class.php - Notepad++
Archivo  Editar  Buscar  Vista  Codificación  Lenguaje  Configuración  Macro  Ejecutar  Plugins  Ventana  ?
DDBB.SQL  Untitled.xml  index.php  editar.php  guardainmueble.php  FileSizeControlerVASS.vbs  inmue

1  <?php
2  //database-clientes      Tabla: paginaweb
3  class DBManager{
4      var $conect;
5
6      var $BaseDatos;
7      var $Servidor;
8      var $Usuario;
9      var $Clave;
10     function DBManager(){
11
12         $this->BaseDatos = "pfc";
13         $this->Servidor = "localhost";
14         $this->Usuario = "root";
15         $this->Clave = "-----";
16     }
17
18     function conectar() {
19         if (!$con=@mysql_connect($this->Servidor,$this->Usuario,$this->Clave)){
20             echo "<h1> [:(] Error al conectar a la base de datos</h1>";
21             exit();
22         }
23         if (!$@mysql_select_db($this->BaseDatos,$con)){
24             echo "<h1> [:(] Error al seleccionar la base de datos</h1>";
25             exit();
26         }
27         $this->conect=$con;
28         return true;
29     }
30 }
31 >>
```

Ilustración 24: conexión MySQL

Por otro lado, el código SQL, que es el encargado de generar las tablas con las que el objeto anterior ha interactuado, quedaría así, tal y como vemos en el ejemplo:

```

C:\Users\aboumediane\Desktop\PFC\CodigoFinal\Final\DDBB.SQL - Notepad++
Archivo  Editar  Buscar  Vista  Codificación  Lenguaje  Configuración  Macro  Ejecutar  Plugins  Ventana  ?
DDBB.SQL  Untitled.xml  index.php  editar.php  guardarinmueble.php  FileSizeControlerVASS.vbs  inmueble.class.php
19  -- Estructura de tabla para la tabla `inmuebles`
20  --
21  --
22  CREATE TABLE `inmuebles` (
23  `Id` int(11) NOT NULL auto_increment,
24  `Referencia` varchar(4) NOT NULL default '0',
25  `Categoria` varchar(100) NOT NULL,
26  `Titulo` varchar(100) NOT NULL,
27  `Precio` varchar(100) NOT NULL,
28  `Descripcion` varchar(1000) NOT NULL,
29  `Dormitorios` varchar(100) NOT NULL,
30  `NumCamas` varchar(100) NOT NULL,
31  `Plazas` varchar(100) NOT NULL,
32  `DisMar` varchar(100) NOT NULL,
33  `Municipio` varchar(100) NOT NULL,
34  `Provincia` varchar(100) NOT NULL,
35  `Publicado` varchar(1) NOT NULL,
36  `Oferta` varchar(1) NOT NULL,
37  `Image1` varchar(100) NOT NULL default 'image-apartamentos/Default.jpg',
38  `Image2` varchar(100) NOT NULL,
39  `Image3` varchar(100) NOT NULL,
40  `Image4` varchar(100) NOT NULL,
41  PRIMARY KEY (`Id`)
42  ) ENGINE=MyISAM  DEFAULT CHARSET=utf8 AUTO_INCREMENT=14 ;
43  --
44  --
45  -- Volcar la base de datos para la tabla `inmuebles`
46  --
47  --
48  INSERT INTO `inmuebles` VALUES (2, 'ref2', 'Piso', 'Piso Muy bien equipado a poco metros del mar', '
49  INSERT INTO `inmuebles` VALUES (11, '33', 'Casa', '55 laiala 11 ¢Atilde;¸plusmn;¸Atilde;¸plusmn;¸Ati
50  INSERT INTO `inmuebles` VALUES (10, '1', 'Chalet', '3', '2', '4 ¢Atilde;¸plusmn;¸Atilde;¸sup3;', '5'
51  -----
52  -----
53  -----
54  --
55  -- Estructura de tabla para la tabla `servicios`
56  --
57  --
58  CREATE TABLE `servicios` (
59  `Id` int(11) NOT NULL auto_increment,
60  `TipoServicio` varchar(100) NOT NULL,

```

Ilustración 25: Código creación DDBB

6. Pruebas

6.1- Resolución de pantalla

Hemos realizado las pruebas de los enlaces solo en la parte pública ya que la parte privada necesita usuario y contraseña para acceder. El resultado queda como sigue:

http://pfc.gexep.com/index.php	OK
http://pfc.gexep.com/ofertas.php	OK
http://pfc.gexep.com/busqueda.php	OK
http://pfc.gexep.com/contacto.php	OK
http://pfc.gexep.com/editarusers.php?id=Registro	OK
http://pfc.gexep.com/InmueblesXCategorias.php?Categoria=Chalet	OK
http://pfc.gexep.com/InmueblesXCategorias.php?Categoria=Piso	OK
http://pfc.gexep.com/InmueblesXCategorias.php?Categoria=Casa	OK
http://pfc.gexep.com/InmueblesXCategorias.php?Categoria=Apartamento	OK
http://pfc.gexep.com/InmueblesXCategorias.php?Categoria=Bajo	OK
http://pfc.gexep.com/content.php?id=14	OK

6.2- Validación del código CSS

Tras seguir las pautas de los estándares CSS hemos obtenido un resultado satisfactorio pasando la pagina por uno el validador CSS <http://jigsaw.w3.org/css-validator/>.

Resultado: <http://jigsaw.w3.org/css-validator/validator?uri=http%3A%2F%2Fpfc.gexep.com%2Findex.php&profile=css3&usermedium=all&warning=1&vextwarning=&lang=es>

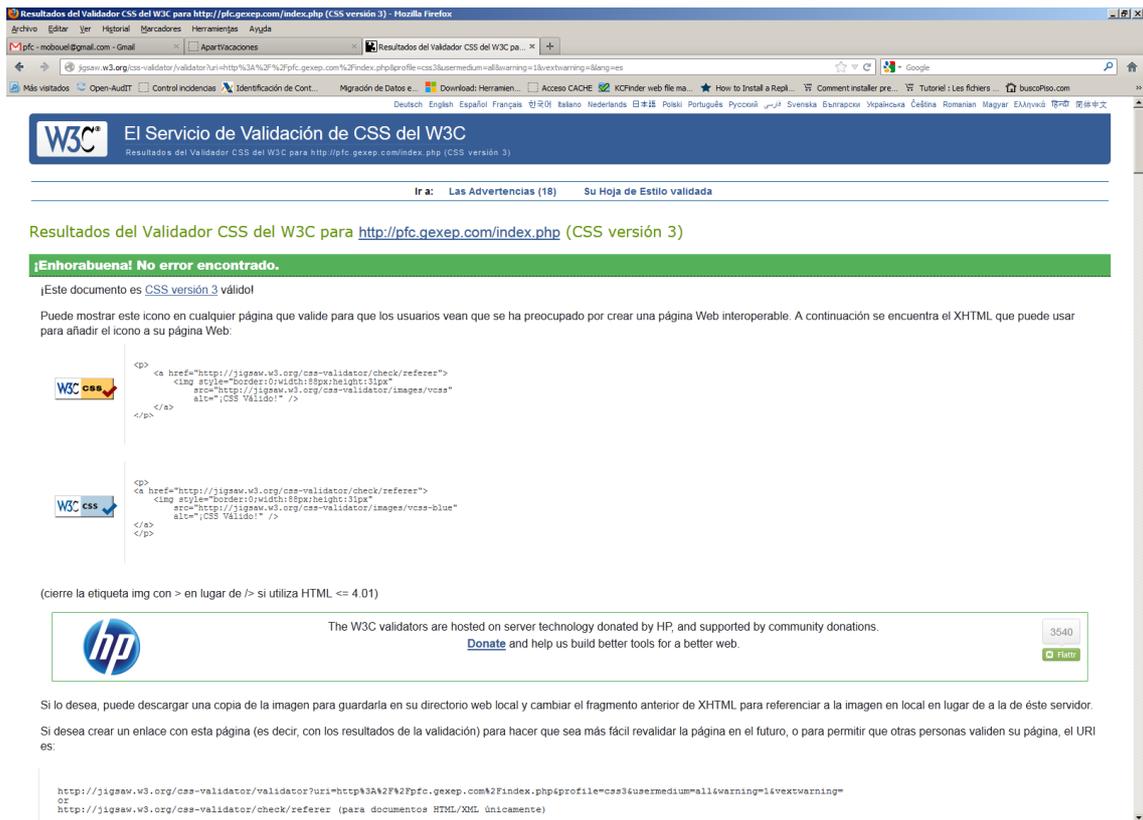


Ilustración 26: Validation del código CCS

6.3- Compatibilidad entre navegadores y Resolución de pantalla

Para evaluar la respuesta de la web, lo hemos comprobado sobre los principales navegadores, es decir, los que más se usan.

Gracias a esta página web <http://gs.statcounter.com>, nos muestra una gráfica con los navegadores más usados en la actualidad. Lo podemos ver a continuación

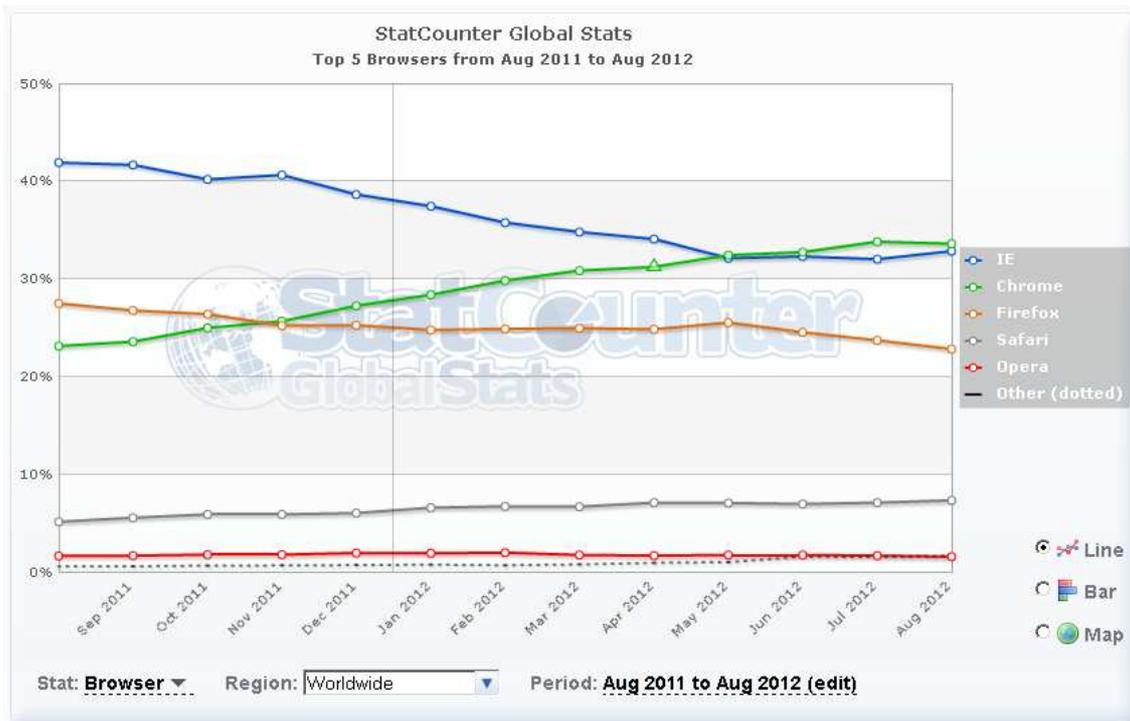


Ilustración 27: Navegadores más usados.

Así pues, en los apartados siguientes, vamos a evaluar la respuesta de la aplicación para estos navegadores.

6.3-1. Explorer

Para comprobar si es correcta la visualización de la web, se han usado diferentes resoluciones de pantalla.

Así pues, primero se ha comprobado con una resolución baja, de **800 x 600 píxeles**, tal y como aparece en la imagen a continuación:

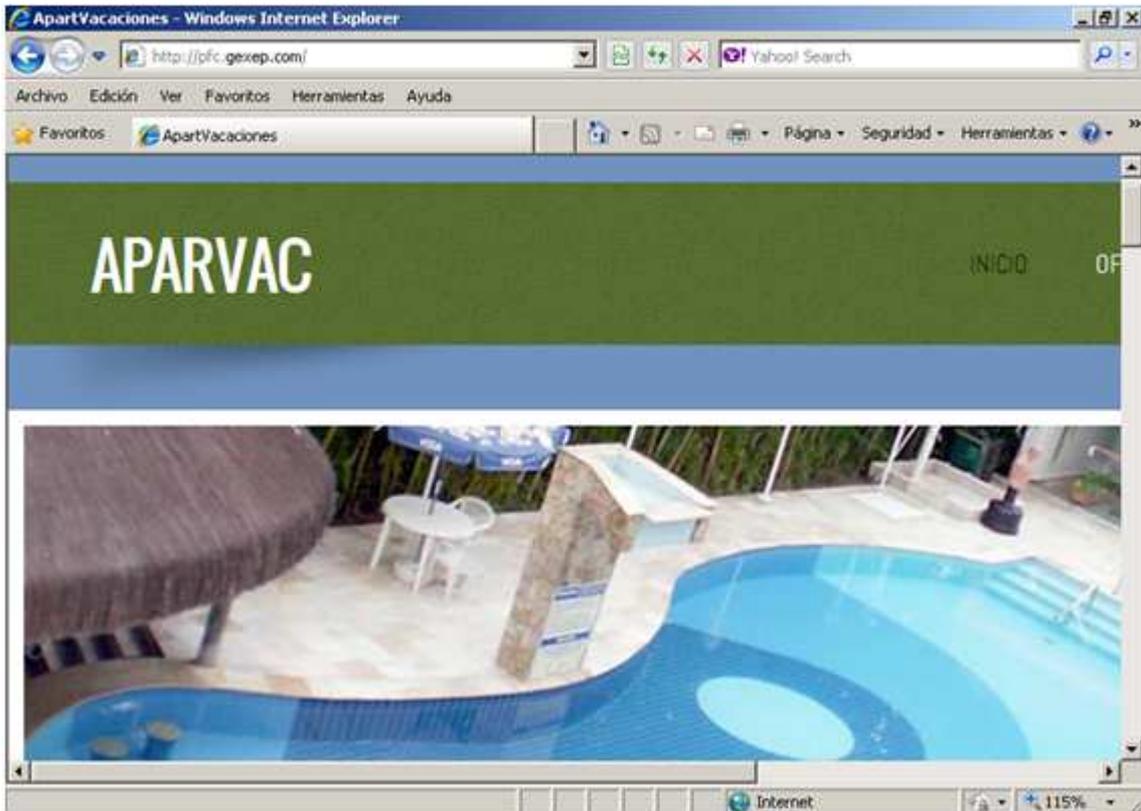


Ilustración 28: IE parte pública 800 x 600.

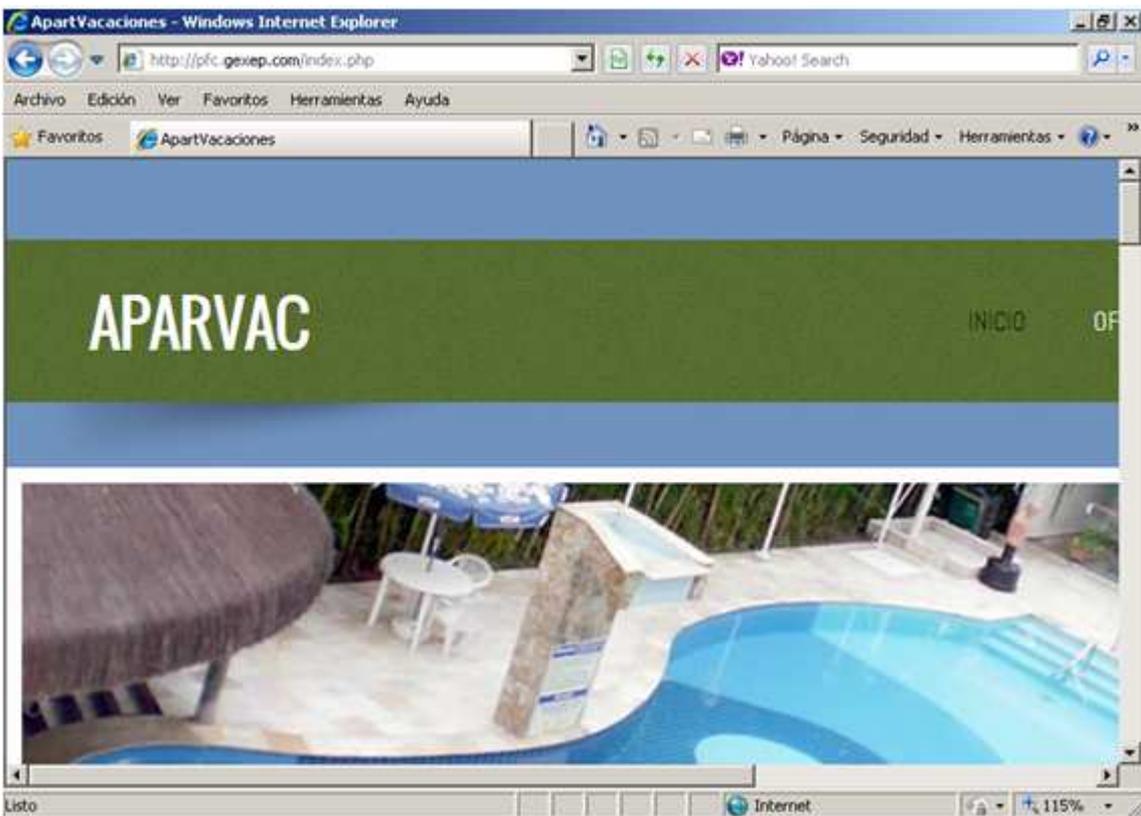


Ilustración 29: IE parte privada 800 x 600.

La visualización es correcta, aunque para ver completamente la web, es necesario usar los *scrolls*. También hay que tener en cuenta, que esta resolución apenas se usa en la actualidad.

Para una resolución de **1600 x 1200 píxeles**, la visualización es correcta, y en este caso, no se necesita el *scroll* lateral para visualizar la web completamente. Lo podemos comprobar en las imágenes que hay a continuación:

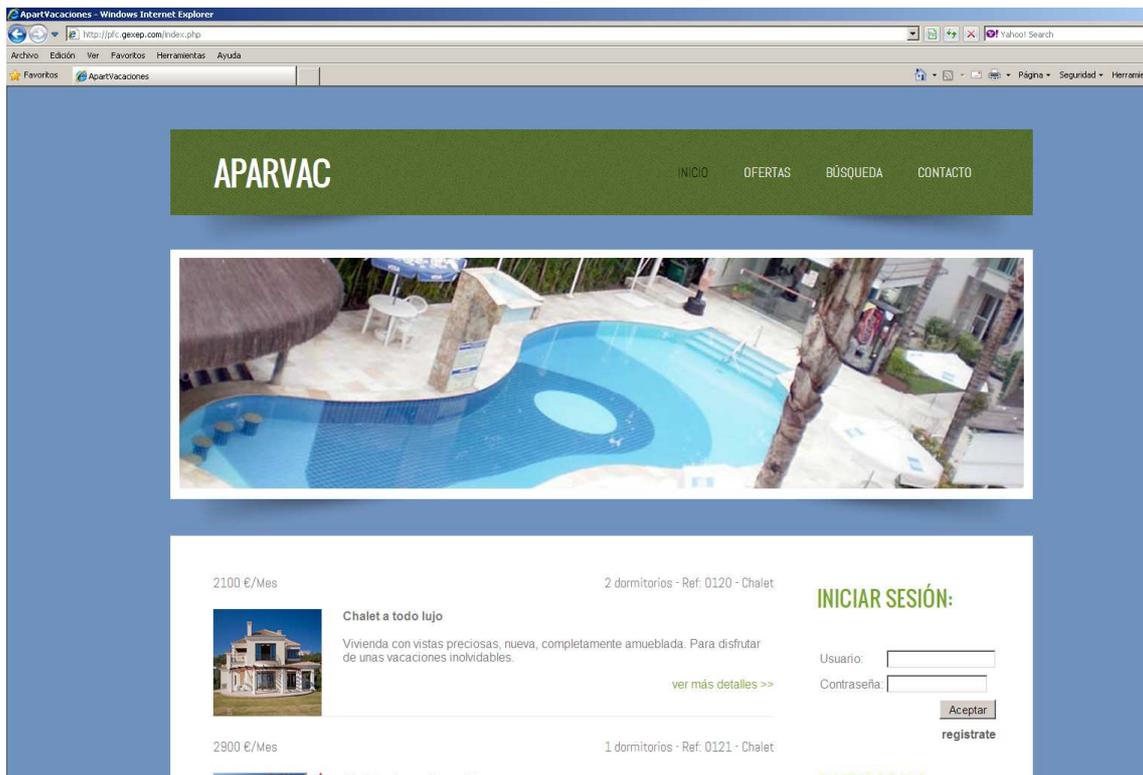


Ilustración 30: IE parte pública 1600 x 1200.

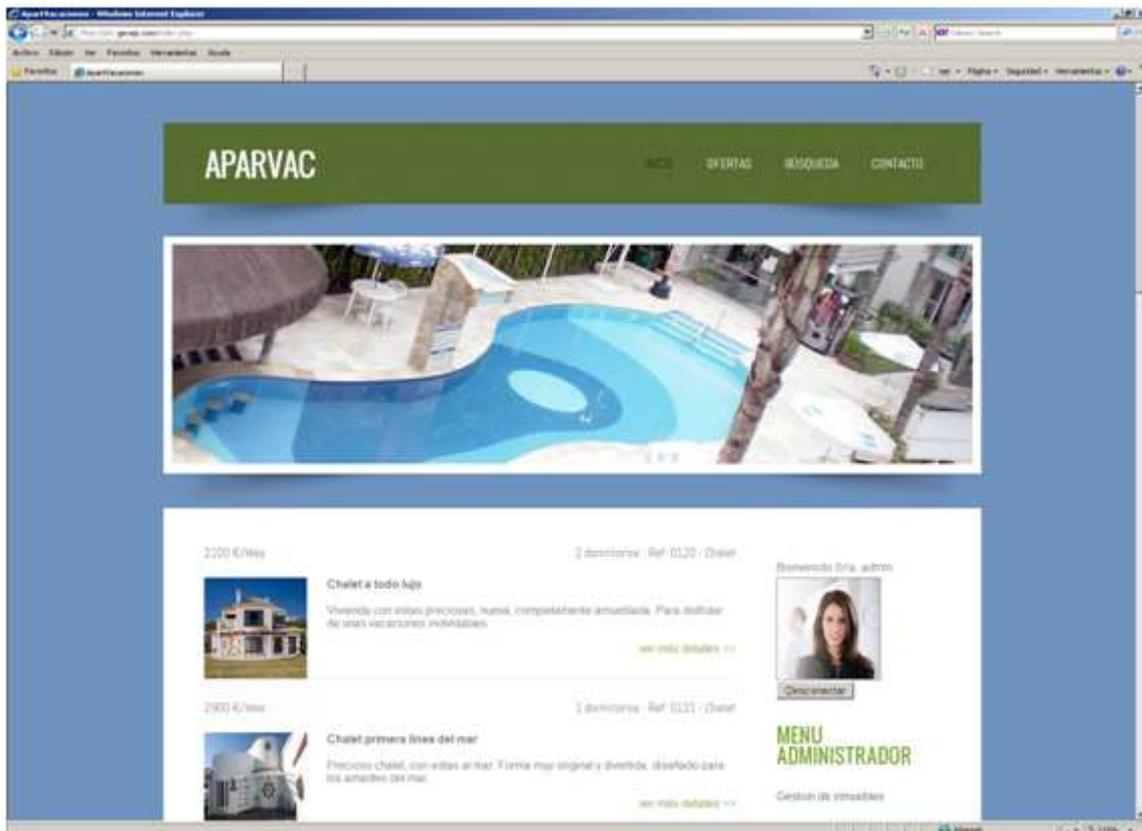


Ilustración 31: IE parte privada 1600 x 1200.

6.3-2. Chrome

Lo mismo ocurre en el navegador Chrome, si se usa una resolución de **800 x 600 píxeles**, aparecen los *scrolls*, pero la visualización es correcta. Lo podemos comprobar con las imágenes que hay a continuación:

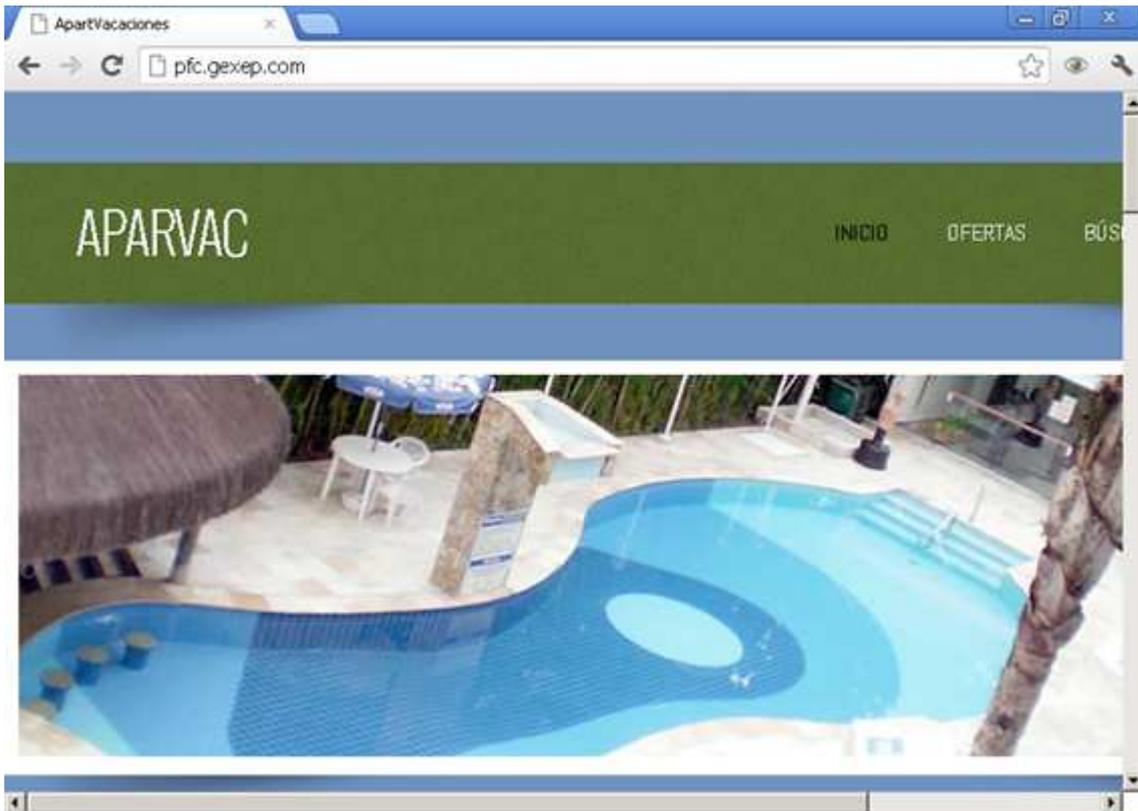


Ilustración 32: Chrome parte pública 800 x 600.

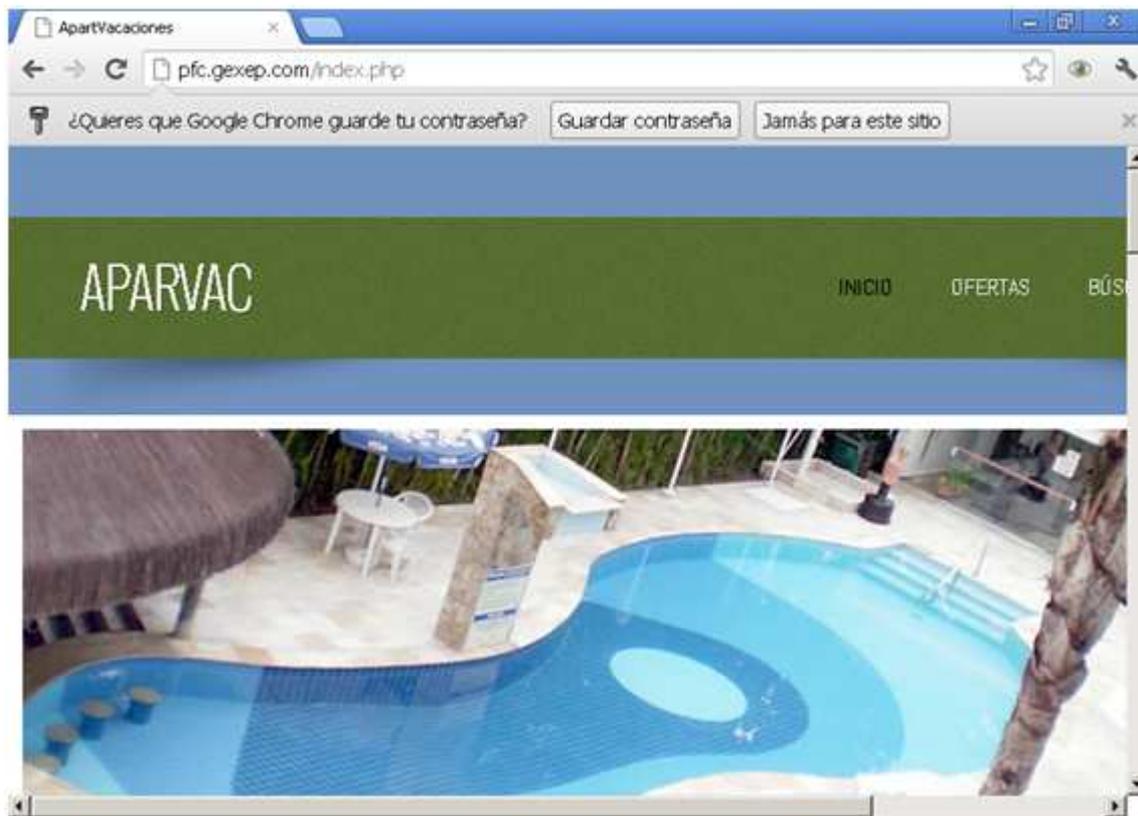


Ilustración 33: Chrome parte privada 800 x 600.

Para una resolución de **1600 x 1200 píxeles**, la visualización es correcta, y en este caso, no se necesita el *scroll* lateral para visualizar la web completamente. Lo podemos comprobar en las imágenes que hay a continuación:

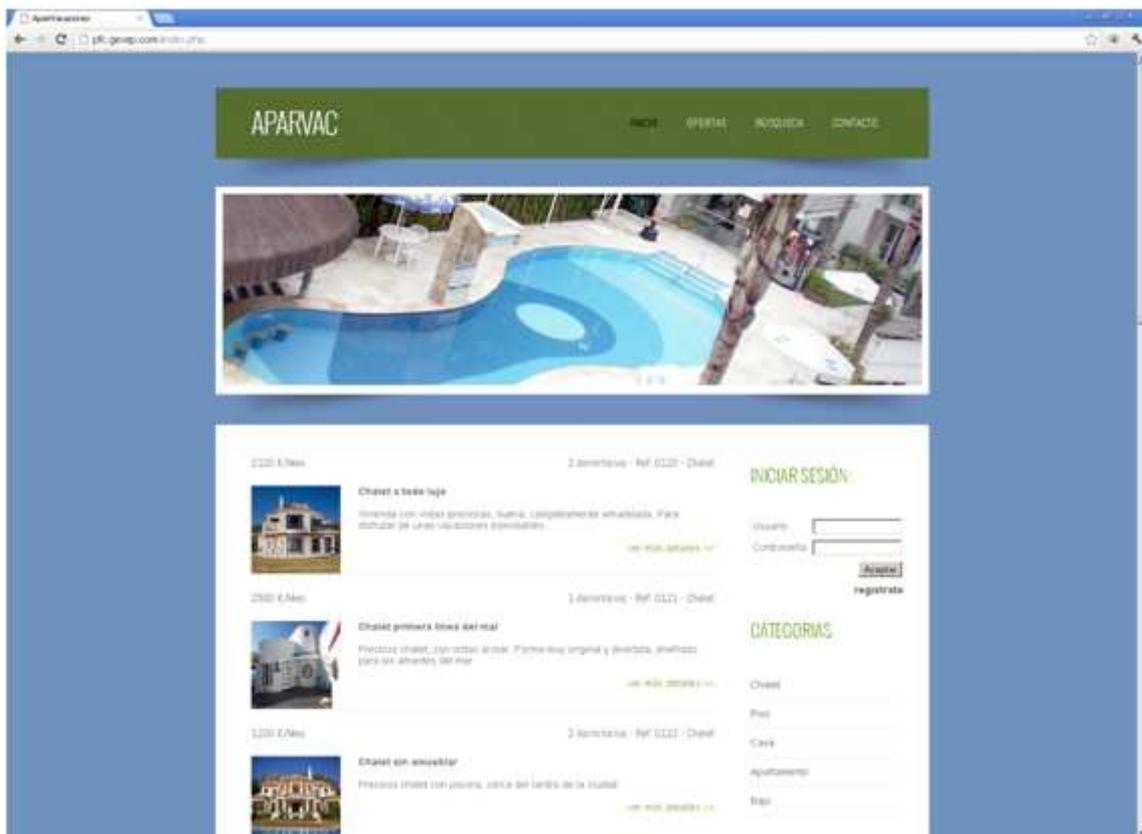


Ilustración 34: Chrome parte pública 1600 x 1200.

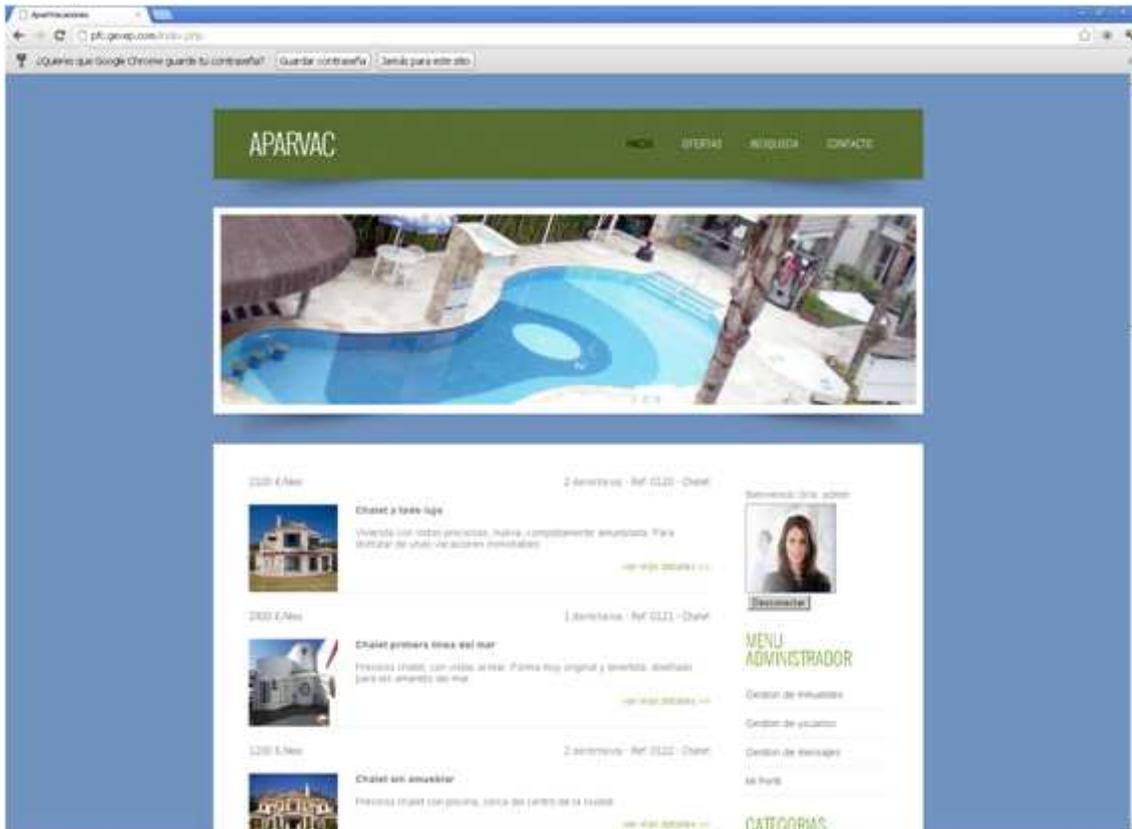


Ilustración 35: Chrome parte privada 1600 x 1200.

6.3-3. Firefox

Y de nuevo, nos encontramos con el mismo caso con el navegador Firefox, con **800 x 600 píxeles** se ve correctamente, pero aparecen los *scrolls*.

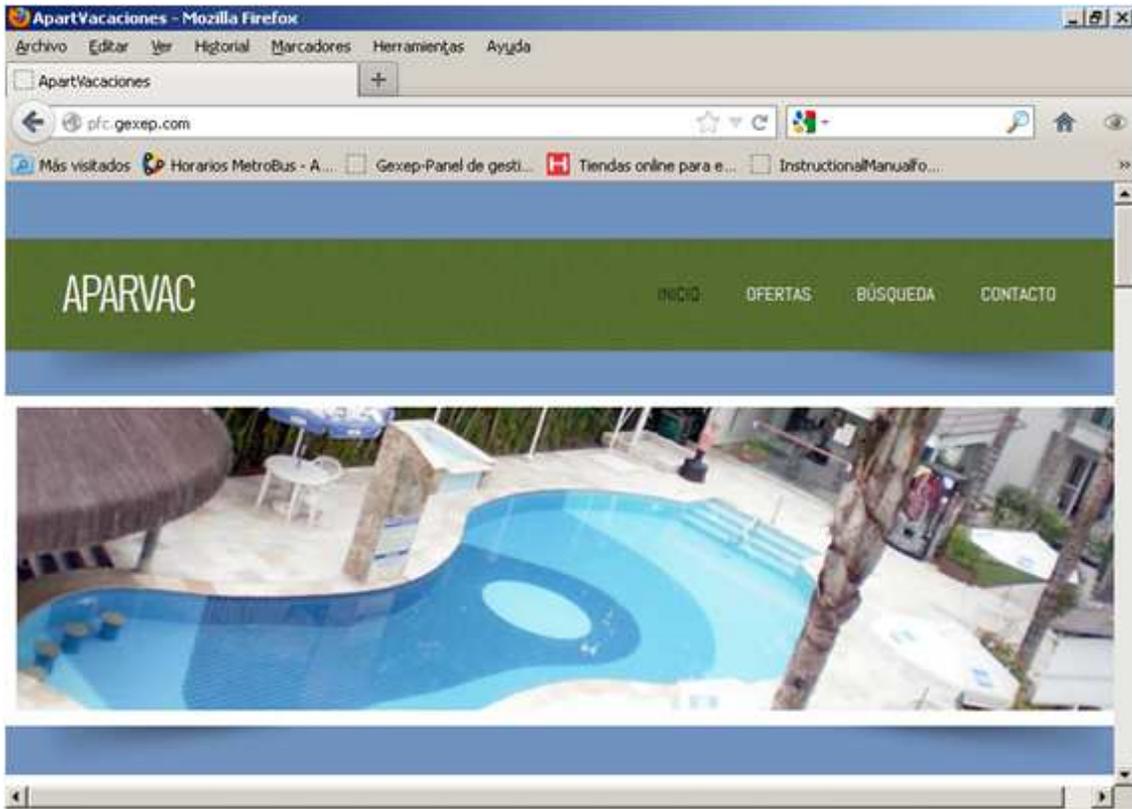


Ilustración 36: Firefox parte pública 800 x 600.

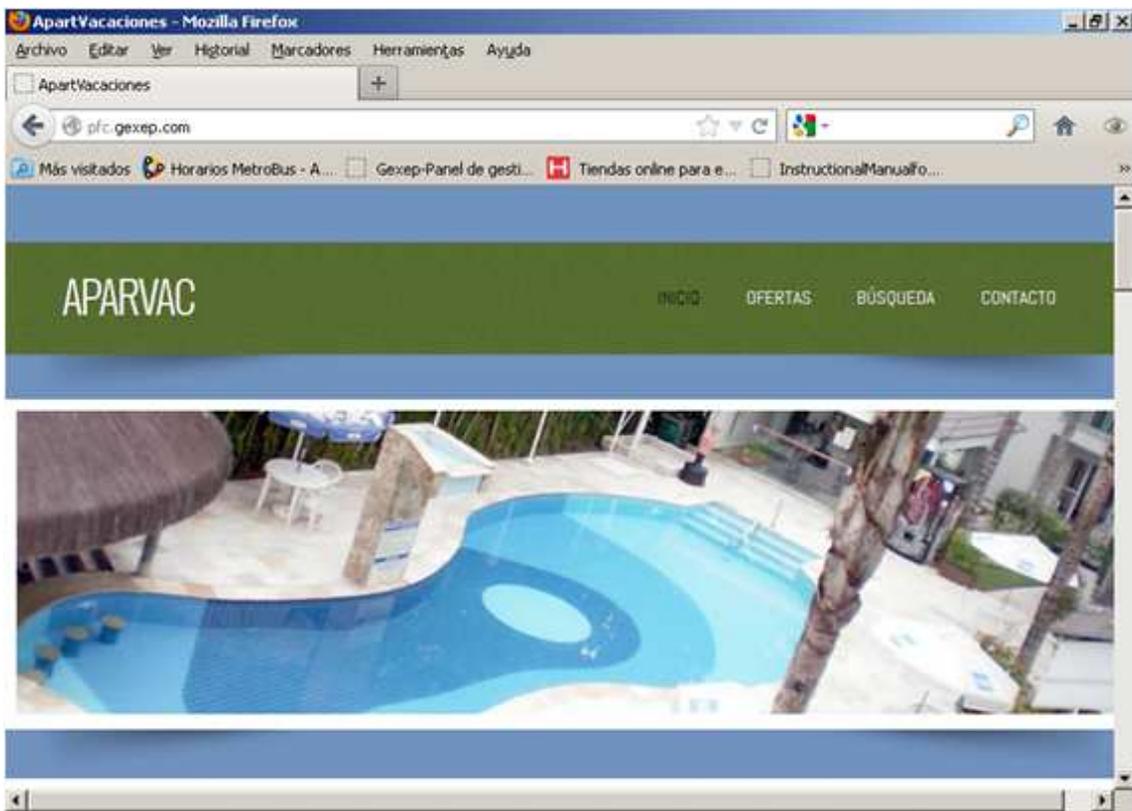


Ilustración 37: Firefox parte privada 800 x 600.

Para una resolución de **1600 x 1200 píxeles**, la visualización es correcta, y en este caso, no se necesita el *scroll* lateral para visualizar la web completamente. Lo podemos comprobar en las imágenes que hay a continuación:

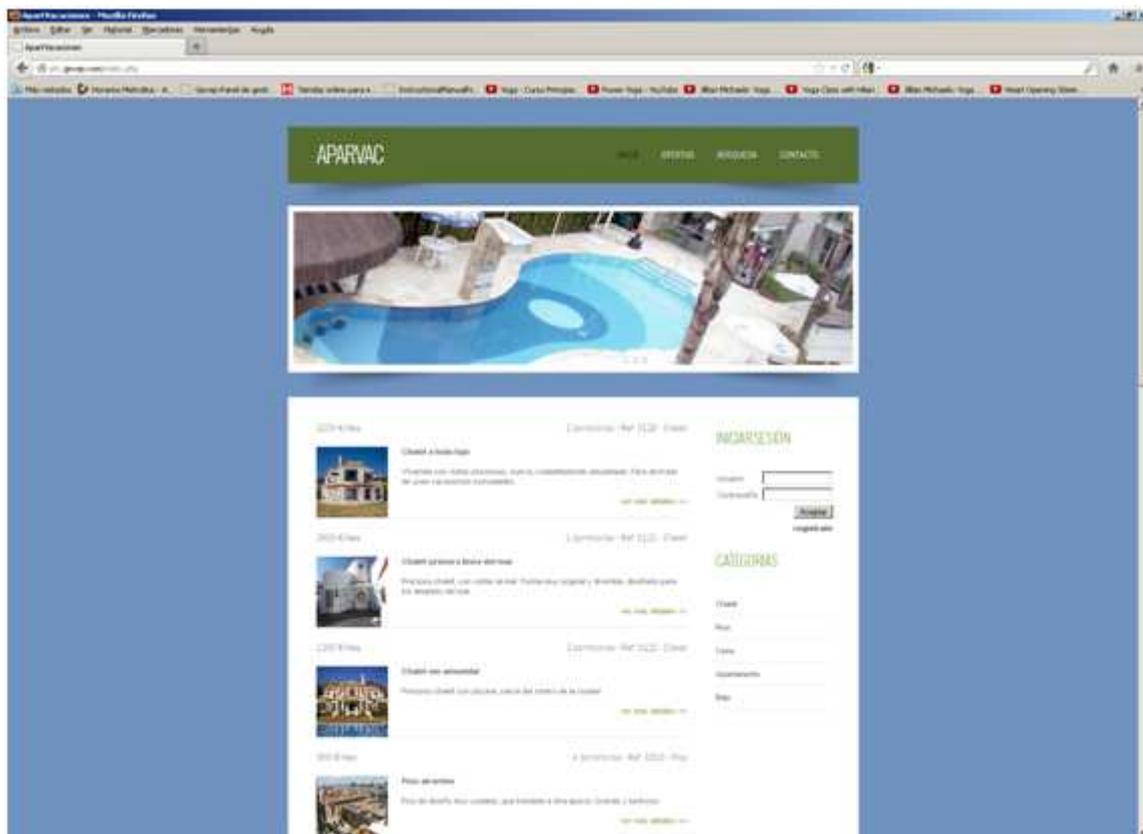


Ilustración 38: Firefox parte pública 1600 x 1200.

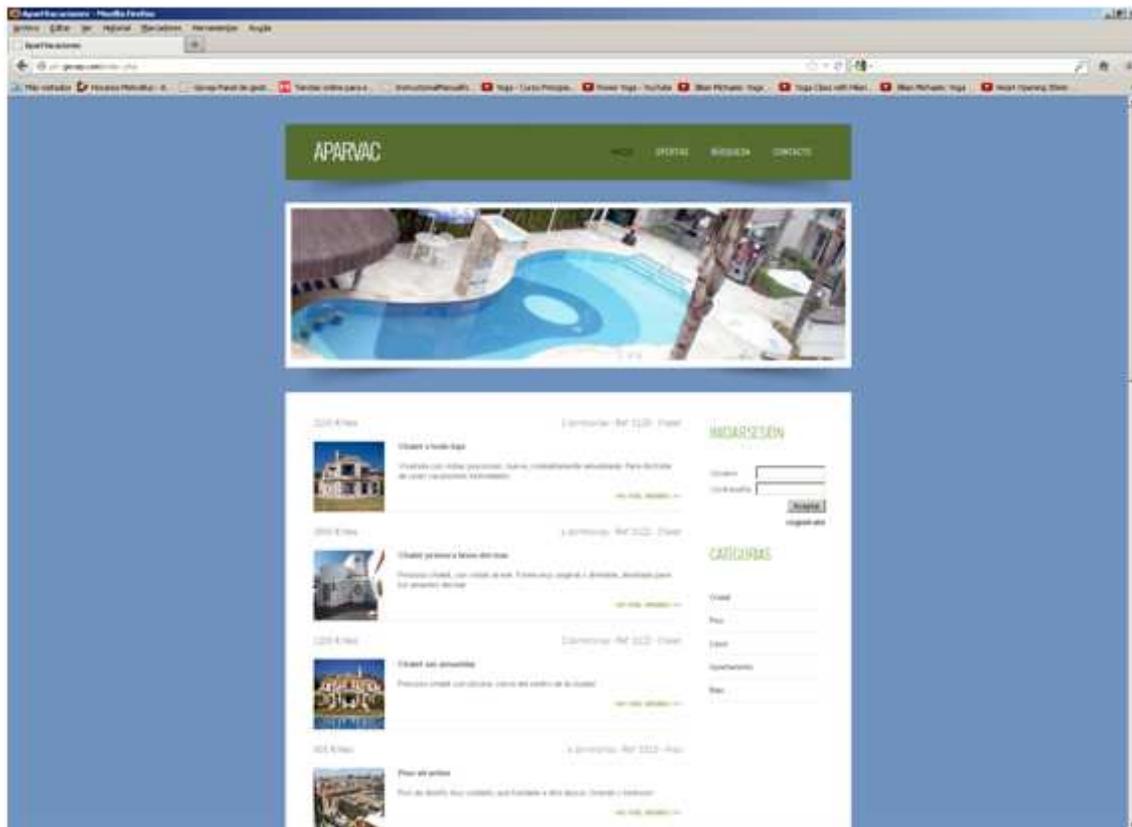


Ilustración 39: Firefox parte privada 1600 x 1200.

7. Conclusiones

En este proyecto, se ha realizado una web inmobiliaria, y se ha intentado y conseguido (al menos de forma subjetiva) que la web sea muy fácil de manejar e intuitiva para cualquier usuario.

Una de las finalidades del proyecto, es que los usuarios no registrados y registrados, puedan encontrar la información que buscan rápidamente y que además, puedan ponerse en contacto con el dueño del inmueble o concertar citas con el agente inmobiliario.

El aspecto novedoso de este proyecto, es el servicio de citas. Este servicio ha sido de gran utilidad para todos los usuarios que han tenido acceso a él, ya que es una forma muy fácil y rápida de concertar una cita en el instante preciso.

También ha sido interesante y satisfactorio la parte de enfocar la web al cliente, es decir hablar con él y saber qué es lo que quiere, y a raíz de ahí diseñar la aplicación, ya que así se va al grano, y una vez finalizada la web, evitas muchos problemas, como por ejemplo, alguna parte que no le guste al cliente final.

Por otro lado, los lenguajes de programación que se han usado, han sido: PHP, JQuery, Java Script y HTML. De estos lenguajes, PHP y JQuery no se han estudiado en la carrera.

Cabe destacar, que la aplicación se ha desarrollado orientada a objetos, ya que la mayoría de aplicaciones web no están orientadas a objetos.

Para finalizar, quiero explicar que este proyecto ha sido una oportunidad para aprender, ya que está dirigido a proyectos necesarios para la vida laboral.



8. Anexos

8.1- Manual de usuario

8.1-1. Usuario registrado

Registrarse: se rellenan los datos y clic en aceptar. Una vez registrado, se puede acceder al menú privado.

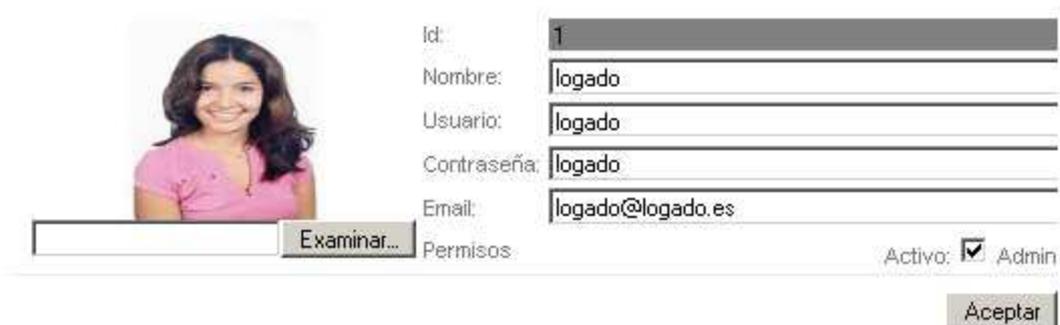
Iniciar sesión: se introduce el usuario y la contraseña, clic en Aceptar.

Gestionar mensajes: en este menú se pueden ver los mensajes creados por el mismo usuario, y para eliminar sus mensajes, clic en el botón rojo. En el mensaje de confirmación clic en Aceptar.

Mi perfil: Este menú es para consultar los datos introducirlos, si se quieren modificar, se cambia el dato anterior por el nuevo, y finalmente clic en Aceptar.

Crear cita: Dentro del inmueble que se quiere visitar (dentro de más detalles), aparece la opción: Concertar cita con un agente, clic aquí y se elige la fecha de la cita, se puede elegir la fecha que se quiera siempre y cuando el agente no tenga otra cita en la misma hora.

Contactar con el dueño del inmueble: Dentro del inmueble que se quiere visitar (dentro de más detalles), aparece la opción: Contactar con el dueño del inmueble, clic aquí y rellenar el formulario, finalmente clic en Enviar.



Id:	1
Nombre:	logado
Usuario:	logado
Contraseña:	logado
Email:	logado@logado.es
Permisos:	Activo: <input checked="" type="checkbox"/> Admin

Examinar... Aceptar

Ilustración 40: Edición de perfil

Desconectar: una vez se ha finalizado la visita a la web, clic en Desconectar.

8.1-2. Usuario Administrador

Iniciar sesión: se introduce el usuario y la contraseña, clic en Aceptar.

Gestión de inmuebles: dentro de este menú, hay varias posibilidades; que son:

- **Añadir inmueble:** Se cargan 4 imágenes, haciendo clic en Examinar, se buscan donde están guardadas, y se selecciona la imagen que se quiere subir. Seguidamente, se introduce la información que te pide del inmueble, una vez se introducen todos los datos clic en Aceptar.
- **Modificar inmueble:** Al hacer clic sobre el título del inmueble, te re direcciona a los datos que hay introducidos del inmueble, se cambia el dato que se quiere modificar y finalmente clic en Aceptar.
- **Eliminar inmueble:** Para eliminar un inmueble, clic sobre el botón rojo, y en el mensaje de confirmación clic en Aceptar.

+ Añadir Inmueble						
Ref.	Título	Categoría	Precio	Dorm.	Pub.	Ofe.
2	Piso Muy bien equipado a poco metro...	Piso	400 €/Mes	5		
11	55 lalala 11 ññññ ó Ä...	Casa	44 €/Mes	77		
10	3...	Chalet	2 €/Mes	5		
14	Chalet a todo lujo...	Chalet	2100 €/Mes	2		
15	Chalet primera linea del mar...	Chalet	2900 €/Mes	1		
16	Chalet sin amueblar...	Chalet	1200 €/Mes	2		
17	Piso atractivo...	Piso	900 €/Mes	4		
18	Casa reformada...	Casa	750 €/Mes	2		
19	Casa para estudiantes...	Casa	100 €/Mes	4		
20	Casa en el centro...	Casa	650 €/Mes	3		
21	Apartamento con vistas al mar...	Apartamento	700 €/Mes	2		
22	Excelente apartamento...	Apartamento	1200 €/Mes	3		
23	Apartamento a todo lujo...	Apartamento	1100 €/Mes	2		
24	Bajo familiar...	Bajo	500 €/Mes	4		
25	Bajo con jardín...	Bajo	450 €/Mes	3		
26	Bajo luminoso...	Bajo	450 €/Mes	4		
27	Casa con terraza...	Casa	910 €/Mes	2		
28	Bajo sin estrenar...	Bajo	600 €/Mes	3		
29	Chalet con jardin...	Chalet	3200 €/Mes	4		

Ilustración 41: Gestión de inmuebles

Gestión de usuarios: En el menú se pueden ver todos los usuarios, y además se puede:

- **Añadir Usuario:** Se rellenan los datos y se hace clic en Aceptar.
- **Eliminar Usuario:** clic sobre el botón rojo, y en el mensaje de confirmación clic en Aceptar.

9. Bibliografía

BUENDÍA GARCÍA F. 2008. Una guía para la realización y supervisión de proyectos final de carrera (PFC) en el ámbito de la web

Direcciones de internet:

<http://es.wikipedia.org>

<http://php.net>

