

Todo Vehículos

Web destinada a la compra, venta y alquiler de vehículos

Índice

A) Introducción.....	2
B) Instalación.....	3
C) Guía del usuario.....	5
C.1) Header	6
C.1.a) Usuario conectado	6
C.1.b) Usuario desconectado.....	7
C.2) Main	8
C.2.a) Últimos añadidos.....	9
C.2.b) Venda o alquile su coche.....	10
C.2.c) Comprar	11
C.2.d) Alquilar	12
C.2.e) Catálogo.....	14
C.2.f) Productos	15
C.2.g) Mensajería.....	16
C.3) Footer	17
D) Guía del programador	18
D.1) Organización	19
D.1.a) Base de datos.....	20
D.2) Código	21
E) Mejoras	25

A) Introducción

Todo Vehículos es una página web **intermediaria** de compra, venta y alquiler de vehículos tales como coches y motos.

La función principal de Todo vehículos es hacer de intermediador entre compradores y vendedores y/o arrendatarios y arrendadores.

Todo Vehículos presenta una interfaz amigable mediante la que el usuario puede ofertar, adquirir u/o informarse sobre uno o varios productos de su interés.

Posteriormente mencionaremos el porqué de la filosofía empleada tanto en el ámbito visual como en el de la programación.

Las funciones que se pueden realizar actualmente en la web son:

- Crear un usuario, logearse y deslogearse.
- Ver la variedad de productos como sus detalles.
- Vender y/o alquilar un producto.
- Escribir mensajes entre usuarios.
- Escribir comentarios en cada página del producto.
- Solicitar un pedido de compra o alquiler.
- Emplear el buscador para localizar el producto deseado.

B) Instalación

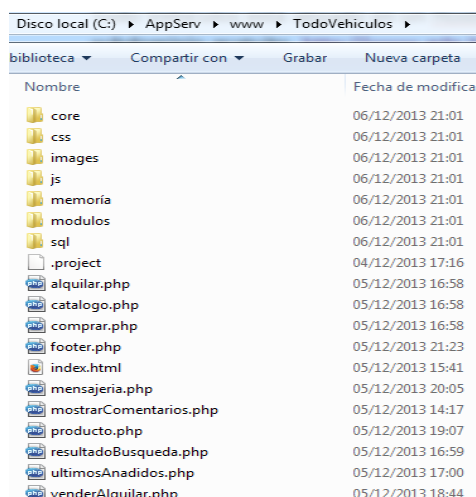
Todo Vehículos está alojado en un Hosting gratuito (Hostinger) y abierto al público mediante el subdominio gratuito, <http://www.gdp.hol.es/ToDoVehiculos/>.

En caso de necesitar ejecutarlo en un servidor local, deberemos realizar los siguientes pasos:

Paso 1: Descargar [appserver](#).

Paso 2: Instalar appserver siguiendo las instrucciones indicadas por el propio ejecutador (Marcar todas las casillas en la instalación).

Paso 3: Una vez instalado, buscaremos el directorio **www** de la carpeta AppServ (por defecto se instala en **C:**) e introduciremos en él el archivo **ToDoVehiculos.rar** y lo descomprimiremos.



Nombre	Fecha de modificación
core	06/12/2013 21:01
css	06/12/2013 21:01
images	06/12/2013 21:01
js	06/12/2013 21:01
memoria	06/12/2013 21:01
modulos	06/12/2013 21:01
sql	06/12/2013 21:01
.project	04/12/2013 17:16
alquiler.php	05/12/2013 16:58
catalogo.php	05/12/2013 16:58
comprar.php	05/12/2013 16:58
footer.php	05/12/2013 21:23
index.html	05/12/2013 15:41
mensajeria.php	05/12/2013 20:05
mostrarComentarios.php	05/12/2013 14:17
producto.php	05/12/2013 19:07
resultadoBusqueda.php	05/12/2013 16:59
ultimosAnadidos.php	05/12/2013 17:00
venderAlquiler.php	05/12/2013 18:44

Nos fijaremos que se generará una carpeta denominada **ToDoVehiculos** que albergará todos los ficheros de la web.

Figura 1. “www/Todovehiculos/”.

Paso 4: Ahora, procederemos a entrar en el fichero ‘**conexión.php**’ dentro de **ToDoVehiculos/core** y editar la variable **\$password = “ ”** por **\$password = “contraseña introducida en la instalación del appserver”**.

```
$username = "root"; // Usuario de la base de datos (por defecto root)
$password = ""; // Contraseña de la base de datos
$hostname = "localhost"; // Dirección IP/DNS que alberga la base de datos (por defecto localhost)
$dbd = "todovehiculos"; // Nombre de la base de datos de la web
```

Paso 5: Abriremos nuestro navegador e introduciremos en la barra de URL “**localhost/phpMyAdmin**” para seguidamente acceder. Una vez nos aparezca una ventana de **login** de phpMyAdmin introduciremos como usuario **root** (si no hemos modificado el usuario en la instalación, de haberlo hecho introducir el usuario correspondiente) y como **contraseña** la especificada en la instalación del appserver.

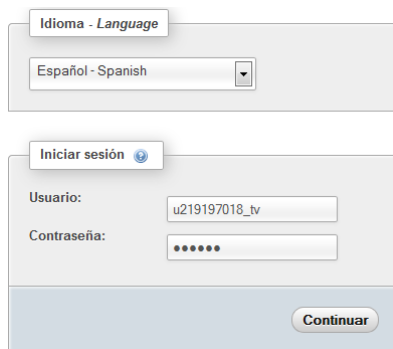


Figura 2. “Login de phpMyAdmin”.

Paso 6: Una vez dentro del phpMyAdmin, buscaremos la opción crear nueva base de datos y la llamaremos **todovehiculos**. Una vez creada la base de datos buscaremos la opción importar e importaremos el archivo “**todovehiculos.sql**” disponible en **www/ TodoVehiculos /sql/** y aceptaremos.

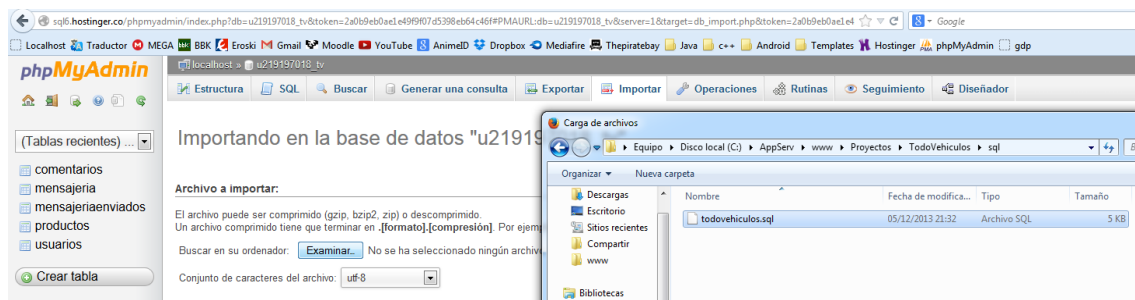


Figura 3. “importar en phpMyAdmin”.

Paso 7: Finalmente, accedemos a la dirección ‘**localhost/ TodoVehiculos**’ y ya podremos observar la web en su total esplendor.

C) Guía del usuario

Para poder mostrar el sitio web de una forma más sencilla y gráfica vamos a proceder a particionar a Todo Vehículos en 3 zonas estáticas: Header, Main y Footer.



Figura 4. "Web particionada".

B.1) Header

En el 'Header' de Todo Vehículo podemos encontrar en la parte izquierda una imagen con el logo de la web, que, al pulsarse nos llevará inmediatamente a la página principal de la web.



Figura 5. "Logo".

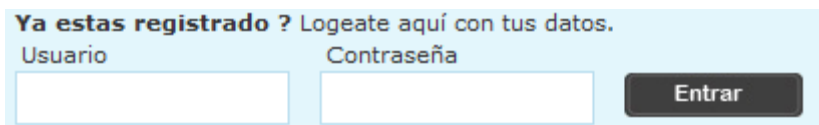
B.1.a) Usuario conectado

En caso de que el usuario no esté logeado, en el lateral derecho del 'Header' podemos observar una imagen acompañada de las palabras "Crear cuenta", que, al ser presionada nos redireccionará a un formulario emergente con sus respectivos datos a rellenar para crear una cuenta que nos permita comprar, escribir comentarios, vender, y alquilar vehículos.

The image shows a user interface for creating an account. At the top, there is a light blue button with a small icon of two people and the text 'CREAR CUENTA'. Below this, a modal window titled 'Registro' (Registration) is displayed. The form contains several input fields: 'Usuario:' and 'Nombre:' on the first line, 'Apellido1:' and 'Apellido2:' on the second line, 'Correo:' on the third line, 'Repetir correo:' on the fourth line, 'Contraseña:' and 'Repetir contraseña:' on the fifth line, and 'Codigo de seguridad:' on the sixth line. The security code field shows a green box with the text 'Jhu98qwr'. At the bottom of the form is a button labeled 'Registrarse'.

Figura 6. "Crear cuenta".

Debajo de “Crear cuenta” se encuentra un pequeño formulario con los campos Usuario y Contraseña que, en caso de introducir los datos acertados nos permitirá realizar las funciones de acceder a la sección de mensajería , escribir comentarios, comprar, alquilar, vender o alquilar vehículos. Si los datos introducidos fuesen incorrectos, el servidor web notificara al usuario de que se ha producido un error.

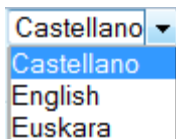


Ya estas registrado ? Logeate aquí con tus datos.

Usuario	Contraseña	Entrar
<input type="text"/>	<input type="password"/>	

Figura 7. “Login”.

Finalmente, mediante el desplegable de la esquina derecha podremos cambiar el idioma de la web por alguno de los que se muestren a continuación. (SIN IMPLEMENTAR).



Castellano ▼

- Castellano
- English
- Euskara

Figura 8. “Selección de idioma”.

B.1.b) Usuario desconectado

En caso de que el usuario este registrado, en vez de ver los botones de “crear cuenta” y “loguearse”, encontrará una imagen de un muñequito correspondiente a la configuración del usuario (SIN IMPLEMENTAR), un icono de una llave que corresponde a la configuración de los vehículos en venta o alquiler (SIN IMPLEMENTAR), un icono de un correo que pertenece a la sección de mensajería que desarrollare posteriormente, y un botón de desconexión, para poder salir de la sesión.

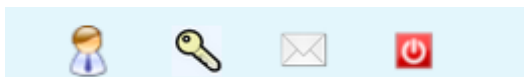


Figura 9. “header logeado”.

B.2) Main

Dentro del 'Main' podremos encontrar un buscador de vehículos que nos permitirá encontrar cualquier variedad de vehículos sin tener que desplazarnos por la web. Para ello deberemos introducir en el formulario la palabra a buscar. Una vez seleccionado el botón buscar, el navegador nos enviará a una página con todos los productos que se asocian a los valores especificados en el formulario. **(SOLO ESTA IMPLEMENTADO EL CAMPO BUSCAR).**

En caso de que la búsqueda no coincida con ningún producto cotejado en la base de datos notificará al usuario mediante un mensaje de error.

The screenshot displays the 'BUSCAR VEHÍCULO' (Search Vehicle) form. It includes input fields for 'Buscar' (Search), 'Tipo' (Type), 'Version', 'Provincia' (Province), 'Antigüedad' (Age), 'Precio' (Price), 'Modelo' (Model), 'Kms', 'Estado' (Status), 'Cambio' (Gear), and 'Combustible' (Fuel). A 'Buscar' button is located at the bottom of the form.

Below the form, the results for 'Audi' are shown. The first result is an 'Audi A3' described as 'Seminuevo, bien cuidado, pocos kilometros, color azul cromado, genial para ciudad' with a price of '15.000€'. It includes a 'Detalles' button and a red 'Comprar' button. The results section indicates 'Resultados: 1'.

Below the results, the search for 'motoreta' is shown, resulting in a message: 'No se encontraron resultados para: motoreta'.

Figura 10. "Búsqueda de vehículos".

En la parte derecha de nuestro "Main" podemos observar los diversos botones que nos conducirán a las diversas funcionalidades de nuestra web. A continuación se procederá a explicar en detalle las finalidades que alberga cada botón.

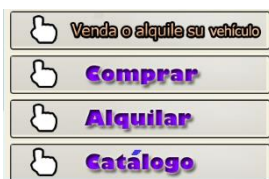


Figura 11. "menú".

B.2.a) Últimos añadidos

La única manera de acceder a esta sesión es mediante el logo, o cuando se introduce la ruta del dominio que alberga la web sin ninguna extensión.

En ella podemos encontrar los 10 últimos vehículos añadidos . En la columna derecha se mostrarán los últimos vehículos disponibles para alquilar, y en la izquierda los últimos vehículos para comprar.

(Comprar) Últimos añadidos	(Alquilar) Últimos añadidos
 <p>Suzuki Kawasaki ninja</p> <p>-</p> <p>Precio: 300€ 0</p> <p>Detalles</p> <p>Comprar</p>	 <p>Scooter Znen</p> <p>Nueva Vespino de Scotter con una mejor suspension que sus anteriores modelos</p> <p>Precio: 1.000€ 0</p> <p>Detalles</p> <p>Alquilar</p>
 <p>Audi A3</p> <p>Seminuevo, bien cuidado, pocos kilometros, color azul cromado, genial para ciudad</p> <p>Precio: 15.000€ 0</p> <p>Detalles</p> <p>Comprar</p>	 <p>Mercedes Clase-A</p> <p>Seminuevo, bien cuidado, pocos kilometros, color azul cromado, genial para ciudad</p> <p>Precio: 1.000€ 0</p> <p>Detalles</p> <p>Alquilar</p>

Figura 12. “Últimas noticias | index.html”.

B.2.b) Venta o alquiler su coche

Mediante este apartado, el usuario generará un anuncio en el sitio web para poder vender o alquilar un vehículo. Para ello, deberá rellenar un formulario donde se le pedirá información del vehículo. Para acceder a esta sección se deberá estar registrado y logeado, sino el acceso estará obstruido por un mensaje de error.

Vender o alquilar >> Introduzca los siguientes datos:

****Si dejan algún campo vacío no se realizará el registro. En caso de no estar seguro en algún campo, escribir un "-" (sin comillas).**

Nombre del vehículo	<input type="text"/>	Precio(€)	<input type="text"/>
Tipo	<input type="text"/>	Modelo	<input type="text"/>
Para	<input type="text"/>	Combustible	<input type="text"/>
Kms realizados	<input type="text"/>	Estado	<input type="text"/>
Antigüedad	<input type="text"/>	Cambio	<input type="text"/>
Carrocería	<input type="text"/>	Color	<input type="text"/>
Equipamiento	<input type="text"/>	URL de la Imagen del vehículo	<input type="text"/>
Versión	<input type="text"/>		

Descripción

Siguiente

Necesita acceder a su cuenta para entrar en esta sección. Si no dispone de una puede [crear una cuenta](#) .

Figura 13. "Venta o alquiler su vehículo".

B.2.c) Comprar

Mediante la sección “Comprar” el usuario dispondrá de una visión global de todos los vehículos disponibles para comprar.

En caso de querer comprar un vehículo bastará con presionar la imagen “Detalles” situada a la derecha del vehículo a solicitar y automáticamente el usuario será redireccionado a una página con todos los datos del producto seleccionado. En dicha página nos encontraremos con un botón de ‘Comprar’ que, en caso de ser presionado, se mandará un mensaje al propietario y al solicitante de que desean comprarle el vehículo y que por favor se ponga en contacto con el solicitante.

En caso de no estar registrado y logeado, en la página de detalles del producto no encontrará ningún botón de compra sino un error que le incitará a logearse para proceder con la compra.

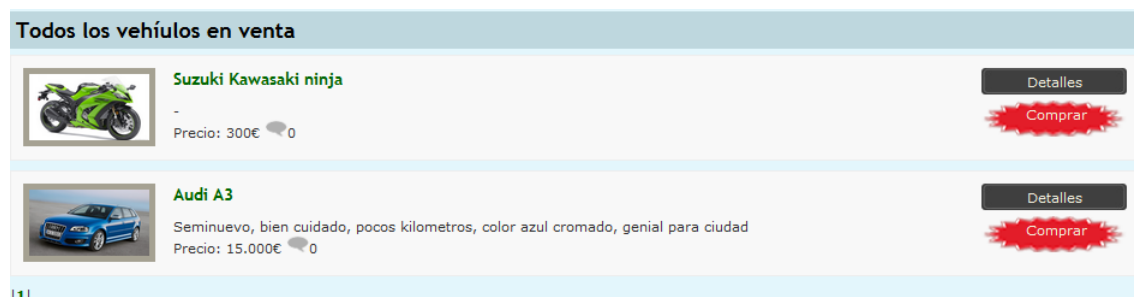


Figura 14. “Comprar”.



Figura 15. “Detalles (usuario no logeado)”.

Datos de Suzuki



Modelo: Kawasaki ninja
Versión: Basico
Kms realizados: 7.000
Antigüedad: 2001
Carrocería: -
Combustible: Gasolina
Estado: Seminuevo
Cambio: Manual
Color: Verde
Precio: 300€
Propietario: Squald

Descripción: -
Equipación: -

Comprar


Figura 16. “Detalles (usuario logeado)”.

B.2.d) Alquilar

El apartado “Alquilar” es básicamente igual que “Comprar” con la diferencia de que vez de mostrar los vehículos a la venta se visualizarán los vehículos disponibles para alquilar.


En caso de querer alquilar un vehículo bastará con presionar la imagen “Alquilar” en la página de detalles del producto a solicitar y automáticamente el propietario del vehículo y el solicitante recibirán un correo para ponerse ambos en contacto

Todos los vehículos en alquiler



Scooter Znen
 Nueva Vespino de Scotter con una mejor suspension que sus anteriores modelos
 Precio: 1.000€ 0

Detalles
 Alquilar



Mercedes Clase-A
 Seminuevo, bien cuidado, pocos kilometros, color azul cromado, genial para ciudad
 Precio: 1.000€ 0

Detalles
 Alquilar

Figura 17. “Alquilar”.

Datos de Scooter



Modelo: Znen

Versión: Básico

Kms realizados: 0

Antigüedad: 2013

Carrocería: -

Combustible: Gasolina

Estado: Nuevo

Cambio: Manual

Color: Negra

Precio: 1.000€

Propietario: Squald

Necesitas estar logeado para poder Alquilar!!!

Descripción: Nueva Vespino de Scotter con una mejor suspension que sus anteriores modelos

Equipacion: -

Figura 18. “Detalles (usuario no logeado)”.

Datos de Mercedes



Modelo: Clase-A

Versión: Básico

Kms realizados: 1.500

Antigüedad: 2010

Carrocería: Aluminio cromado

Combustible: Gasolina

Estado: Seminuevo

Cambio: Automático

Color: Plateado

Precio: 1.000€

Propietario: Squald

Alquilar

Descripción: Seminuevo, bien cuidado, pocos kilometros, color azul cromado, genial para ciudad

Equipacion: Ninguna

Figura 19. “Detalles (usuario logeado)”.

B.2.e) Catálogo

Dentro de la sección “Catálogo” el usuario dispondrá de una visión global de todos los vehículos disponibles tanto para comprar como para alquilar.

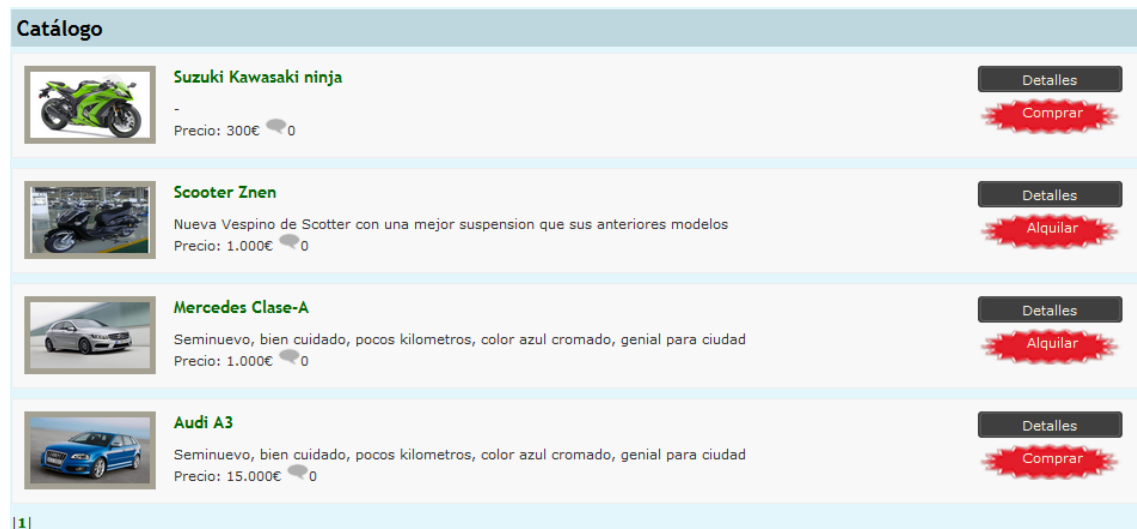


Figura 20. “Catálogo”.

Como es probable que se cargue excesivamente la web al mostrar tantos productos, hemos aligerado la carga generando un sistema de paginación por números. Dicho sistema, genera un índice del total de páginas con productos adheridos a ellas, que podemos localizar en la zona superior del ‘Footer’.

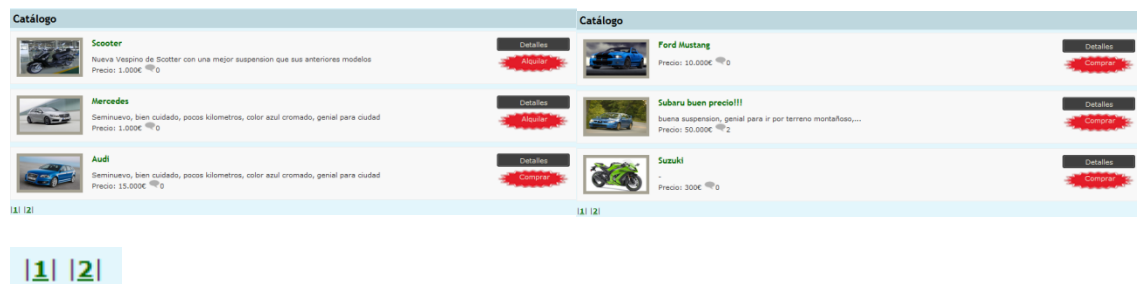


Figura 21. “Catálogo - paginación”.

* Este sistema de paginación es vigente en varias secciones de Todo Vehículos.

B.2.f) productos

Como ya hemos mencionado antes productos, que hace referencia al nombre 'Detalles' ya nombrado en el apartado 'Comprar' y 'Alquilar' es la página en la que se muestran los datos del producto seleccionado. Consta con un Subapartado de comentarios, donde cada usuario puede dejar constancia de su presencia. Cada producto alberga diferentes comentarios.

En caso de que el usuario no este logeado podrá ver los comentarios de los demás usuarios pero no podrá escribir ninguno. Para ello deberá registrarse y/o logearse.

Datos de Suzuki


Modelo: Kawasaki ninja
 Versión: Basico
 Kms realizados: 7.000
 Antigüedad: 2001
 Carrocería: -
 Combustible: Gasolina
 Estado: Seminuevo
 Cambio: Manual
 Color: Verde
 Precio: 300€
 Propietario: Squald

Necesitas estar logeado para poder Comprar!!!

Descripción: -
 Equipación: -


Comentarios

2013-12-06 05:37:16 Squald
 Me gusta

[\[1\]](#)

Necesita acceder a su cuenta para dejar un comentario. Si no dispone de una puede [crear una cuenta](#).

Figura 21. "producto | detalles (no logeado)".

Datos de Suzuki


Modelo: Kawasaki ninja
 Versión: Basico
 Kms realizados: 7.000
 Antigüedad: 2001
 Carrocería: -
 Combustible: Gasolina
 Estado: Seminuevo
 Cambio: Manual
 Color: Verde
 Precio: 300€
 Propietario: Squald

Comprar

Descripción: -
 Equipación: -

Comentarios

2013-12-06 05:37:16 Squald
 Me gusta

[\[1\]](#)

Escriba un comentario

Enviar

Figura 22. "producto | detalles (logeado)".

B.2.g) Mensajería

Mediante el sistema de mensajería instantánea incorporado en Todo Vehículos los usuarios podrán intercambiar información.

El sistema de mensajería consta de una sección de elementos recibidos denominada 'Bandeja de entrada', y de una sección de elementos enviados denominada 'Elementos enviados'.

Gracias a este sistema, cuando un usuario solicite un producto, el propietario del producto y el solicitante recibirán un mensaje que les permitirá mantenerse en contacto.



The screenshot shows a web interface for sending messages. At the top, there's a header with 'Bandeja de Entrada | Enviados'. Below it, a status bar says 'No hay mensajes en la bandeja de entrada'. The main section is titled 'Escribir Mensaje'. It contains three input fields: 'Usuario destinatario', 'Asunto', and a large 'Mensaje' text area. At the bottom right of the text area is a 'Siguiete' button.

Figura 23. "Mensajería".



The screenshot shows the 'Bandeja de Entrada | Enviados' section with two messages. Each message displays the sender, subject, date, and the message content, along with a 'Borrar Mensaje' link.

Remitente:	Asunto:	Fecha:	Mensaje:	Acción:
Servicio de TodoVehiculos	Nueva solicitud de para Alquilar el producto Scooter	2013-12-07 08:05:04	El usuario Squald desea Alquilar su producto: Scooter	Borrar Mensaje
Servicio de TodoVehiculos	Se ha mandado una solicitud al usuario Squald para Alquilar el producto Scooter	2013-12-07 08:05:04	Producto solicitado: Scooter a Squald	Borrar Mensaje

Figura 24. "Mensaje resultante de pulsar el botón alquilar del producto Scooter (El primero es el mensaje que recibirá propietario, y el segundo, el que recibirá el solicitante)".

B.4) Footer

En el “Footer” podremos encontrar una copia del menú para que los usuarios puedan desplazarse con mayor comodidad, un link “Soporte técnico” (**SIN IMPLEMENTAR**) que nos permitirá escribir un mensaje a la administración de Todo Vehículos para notificarles cualquier error, posible mejora, duda o queja. Y una escritura con el copyright.

[Inicio](#) | [Vender/Alquilar](#) | [Comprar](#) | [Alquilar](#) | [Catálogo](#) | [Soporte técnico](#) © Copyright - All Rights Reserved.

Figura 25. “Footer”.

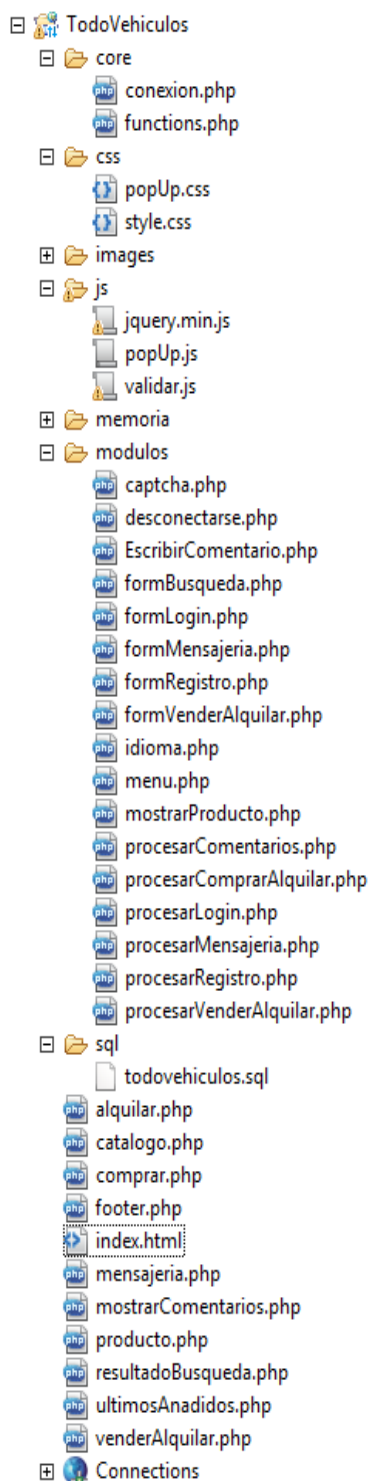
D) Guía del Programador

Todo Vehículos ha sido creado mediante la herramienta [Aptana Studio 3](#).

Esta herramienta es un excelente editor y gestor de proyectos que, nos permite tener la web actualizada en todo momento gracias a la opción de FTP que sube a nuestro servidor las modificaciones llevadas a cabo desde nuestro pc.

Las tecnologías empleadas han sido: **xhtml**, como base de todos los ficheros web; **css**, como núcleo del diseño web; **JavaScript** y varias de sus librerías (**Jquery** y **Ajax**), como herramientas avanzadas de programación web para dotar al sitio web de un aspecto más profesional; **php**, como herramienta de programación base y como mediador entre css, mysql y html; y por último, pero no menos importante **mysql**; como base de datos para almacenar la información.

Cabe indicar que aun pudiendo haber utilizado xml como “base de datos” o el propio java script, se ha llegado a la conclusión de que, al ser una web en la que se necesitan almacenar una gran variedad de datos (usuario, productos, comentarios,...), es más eficiente y seguro emplear mysql.



D.1) Organización

Para una mejor organización Todo Vehículos está formado por varios directorios que jerarquizan el código de nuestra web.

En la carpeta **core** podemos encontrar el archivo '**conexion.php**' que alberga los datos de acceso a la base de datos y, '**funciones.php**', que es el corazón de todo el funcionamiento lógico de nuestra web (desde la impresión de todos los datos de la base de datos hasta la obtención de la ip del usuario).

En la carpeta **css** encontramos los archivos que dotan de una estructura grafica a Todo Vehículos. Esta compuesta por '**popUp.css**', que almacena el código de diseño referente al popup de registro. Y, '**style.css**', que almacena el código de diseño de toda la web.

En **images**, guardamos todas las imágenes empleadas en el sitio web.

En **js** podemos encontrar un archivo denominado 'jquery.min.js' que sirve para poder utilizar tanto **JavaScript** como sus librerías (**Jquery**, **Ajax**,...). También podemos localizar un '**popUp.js**', que alberga las funciones JavaScript que permiten generar y tratar el popup de registro de usuarios. Y, un '**Validar.js**' que mantiene las funciones de validación de datos para el login y el registro de usuarios; un algoritmo para limitar mensajes de entrada y una sentencia para poder ver los comentarios escritos sin necesitar de recargar la web.

En la carpeta **modulos** se han almacenado todos los archivos referentes a formularios y procesamiento de información de formularios.






En la carpeta **sql** incluimos la estructura de la base de datos.

Figura 26. "Gestor de Aptana"

Todos los documentos de la carpeta raíz corresponden a la estructura de cada sección del menú.

D.1.a) Base de datos

Todo Vehículos almacena todos sus datos en una base de datos jerarquizada de la siguiente manera:

 comentarios
 mensajeria
 mensajeriaenviados
 productos
 usuarios

En **comentarios** se almacenan los comentarios de todos los usuarios (La función de vinculación con la base de datos se encuentra en '**functions.php**' y su llamada en '**produtco.php**').

Las tablas **mensajeria** y **mensajeriaenviados** almacenan todos los mensajes enviados entre usuarios (La función de vinculación

Figura 27. "tablas de la base de datos"

con la base de datos se encuentra en '**functions.php**' y su llamada en '**formLogin.php**' .

Gracias a la tabla **productos** disponemos de un lugar donde almacenar todos los datos de los vehículos que ofertan los usuarios (Esta tabla, seguida de la de usuarios, es indudablemente, la más empleada en nuestra web. Su código de vinculación se encuentra en '**functions.php**').

Mediante la tabla usuarios, disponemos de un servicio de usuarios, donde cada individuo dispone de una serie de datos privados protegidos por una contraseña.

D.1) Código

Dado que Todo Vehículos es una web muy compleja y dispone de demasiado código para documentar en unas pocas hojas vamos a explicar la parte del código más importante.

Para mostrar la estructura de nuestro código xhtml/html vamos a usar como ejemplo el fichero 'mensajeria.php'

```
<?php
session_start();
require_once('core/conexion.php');
require_once('core/functions.php');
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Todo Vehículos</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link rel="icon" href="images/motor.png" type="image/x-icon">
<link rel="stylesheet" href="css/style.css" type="text/css" />
<link rel="stylesheet" href="css/popUp.css" type="text/css" />
<script type="text/javascript" src="js/jquery.min.js"></script>
<script type="text/javascript" src="js/validar.js"></script>
<script type="text/javascript" src="js/popUp.js"></script>
</head>
<body>
<div id="wrap">
<div id="header">
<div id="sitename">
<h1 id="logo"><a href="index.html"></a></h1>
</div>
<div class="language">
<?php include_once('modulos/idioma.php');?>
</div>
<div id="useractions">
<?php
include_once('modulos/formLogin.php');
echo'</div>';
include_once('modulos/formRegistro.php');
?>
</div>
</div>
<div id="content">
<div id="home_main">
<?php include_once('modulos/formBusqueda.php'); ?>
</div>
<div id="home_sidebar">
<?php include_once('modulos/menu.php'); ?>
</div>
<div class="clear">&nbsp;</div>
<div id="main-buy">
<center><h1><a href='mensajeria.php?id=0'>Bandeja de Entrada</a> |
<a href='mensajeria.php?id=1'>Enviados</a></h1></center>

<?php mostrarMensajeria();
include_once('modulos/formMensajeria.php');?>
</div>
<div class="clear">&nbsp;</div>
</div>
<div id="footer">
<?php include_once('footer.php') ?>
</div>
</div>
</body>
</html>
```

Figura 28. "mensajería.php"

Mediante **sesión_start()** permitimos usar la variable `$_SESSION['Email']` para detectar si el usuario está logeado en la web o no.

Gracias a los **require_once** de los ficheros '**conexión.php**' y '**functions.php**' podemos utilizar las funciones de la base de datos y todas las funciones implementadas en el lenguaje **php** para nuestro sitio web.

También podemos distinguir las llamadas a los **css** para cargar el diseño en la web y los **js** para poder emplear las funciones programadas mediante **javaScripts** y sus derivados.

Finalmente queda destacar que se ha decidido separar por módulos todas aquellas partes que van a variar o que incluyen código algorítmico.

Por ejemplo la sección correspondiente a mostrar el login y todas sus derivados (mensajería, logeo, desconexión,...) se almacenan en **modulos/formLogin.php**.

La sección del formulario de registro se almacena en **modulos/formRegistro.php**.

El formulario de búsqueda se aloja en **modulos/formBusqueda.php**.

El menú podemos localizarlo en **modulos/menu.php**.

Y para el caso del footer, lo podemos localizar en **footer.php**

Estos son unos de los principales ficheros de nuestro sistema web.

En lo que a JavaScript/jquery/Ajax concierne, vamos a analizar en detalle un par de funciones de nuestro fichero **validar.js**

Recargar comentarios sin necesidad de recargar la página completa:

```
$('#enviarComentario').click(function(){
    no_error = validar_comentario();
    if (no_error) {
        var comentid = $("input#id").val();
        $.ajax({
            url: 'modulos/procesarComentarios.php',
            type: 'post',
            dataType: 'json',
            data: $('#comentariosForm').serialize(),
            success: function() {
                $('#c_ImprimirComentarios').load("mostrarComentarios.php?id="+ comentid);
                return false;
            }
        });
    }
});
```

Figura 29. "Recargar comentarios. Y evitar campos vacíos"

Cuando clickemos en el botón con la etiqueta `enviarComentario` y no tenga como parámetro un campo vacío, mandará mediante el método **POST** todos los datos del formulario a **procesarComentarios.php**, donde se ejecutarán las sentencias necesarias para almacenar el mensaje en la base de datos. Consecutivamente se cargará la web **mostrarComentarios.php?id=comentid** en la etiqueta `C_imprimirComentarios`. Lo negativo de este sistema es que necesitas generar un fichero independiente con el código e la función que deseas mostrar. Mediante `?id=comentid` estas mandando una sentencia **GET** a `mostrarComentarios` donde dices que `$_GET['id']` tiene el valor de `comentid`. En nuestro caso, `comentid` es el id de la producto en el que se está escribiendo el comentario.

```
<?php
require_once('core/conexion.php');
require_once('core/functions.php');
imprimirComentarios($_GET['id']);?>
```

Figura 30. “mostrarComentarios.php ”

Limitar la cantidad de letras por mensaje a a enviar (implementado en comentarios, en descripción del vehículo y en el mensaje del sistema de mensajería).

```
function ContarTexto(actual, cajita, max)
{
    if (actual.value.length > max)
        actual.value = actual.value.substring(0, max);
    else
        cajita.value = max - actual.value.length;
}
});
```

Figura 31. “Limitador de letras por mensaje ”

Su único cometido es, eliminar las letras que excedan el límite establecido. Los valores a introducir han de ser: la etiqueta del formulario donde se va a tratar, un `String` y un `int` con el tamaño máximo del string.

```
<td><label><textarea class="textarea" tabindex="50" name="msj_mensaje" id="msj_mensaje" cols="50" rows="19"
onKeyDown="ContarTexto(formMensajeria.msj_mensaje,formMensajeria.cajita,500);"
onKeyUp="ContarTexto(formMensajeria.msj_mensaje,formMensajeria.cajita,500);"></textarea></label></td>
```

Figura 31. “llamada a la función limitadora de letras de un mensaje ”

En lo que a **css** respecta, no hay mucho que expresar ya que no son más que comandos adheridos a una etiqueta en la que hay que tener presente en que capa se sitúa.

A continuación se muestra el style de alguno de los mensajes de error que se muestran mediante la activación de una sentencia **js**.

```
/* error -----*/
.error{display:none; color:#0F0}
.loginbtn {background: url('image/Loginbutton.jpg');}
.nota{border: 2px solid; color:#333333 font-size: 1.20em;}
```

Figura 31. “Etiqueta de los mensajes de error ”

Por ultimo vamos a mostrar un pequeño código referente al uso de **php** para conectarse a la base de datos y obtener una serie de datos.

Para ello primero debemos crear un código de apertura de la base de datos para poder acceder a su información.

```
//connection to the database
$dbhandle = mysql_connect($hostname, $username, $password) or die("Imposible conectarse a MySQL");
mysql_select_db($db,$dbhandle)or die("No se puede localizar la base de datos indicada");
```

Figura 32. “conexión con la base de datos”

Después basta con emplear el código apropiado para realizar una consulta con la base de datos, por ejemplo vamos a observar el código mediante el que imprimimos los comentarios asociados a un producto **\$id** y almacenados en la base de datos. Estos comentarios se mostrarán restringidos a 10 impresiones por página.

Los datos se obtienen mediante el comando “**SELECT**” que nos permite coger el TODO (*) de la tabla comentarios donde la columna **idProducto** coincida con **\$id**, ordenados por la id del comentario de manera descendente y limitados a 10 mensajes por pagina.

```
function totalComentarios($id){
    $tablaComentarios= mysql_query("select * from comentarios WHERE IdProducto='".$id."'") or die ("totalComentarios :: Error en la base de datos.");
    return mysql_num_rows($tablaComentarios);
}

function imprimirComentarios($id){
    if(isset($_GET['pag']))
        $pag = $_GET['pag'];
    else
        $pag = 1;
    $registrosPagina = 10;
    if($pag == 1)
        $listaInicio = 0;
    else
        $listaInicio = $registrosPagina * ($pag - 1);
    $tablaComentarios= mysql_query("select * from comentarios WHERE IdProducto='".$id."' ORDER BY IdComentario DESC LIMIT $listaInicio, $registrosPagina") or die ("imprimirComentarios");
    while($row = mysql_fetch_array($tablaComentarios)){
        echo "<div id='headings'>";
        echo "<p><div class='content'>";
        echo "<span class='fecha'>".$row['fecha'].</span>";
        echo "<span class='autor'>".$row['autor'].</span>";
        echo "<p><span class='article-body'>".$row['comentario'].</span></p></div>";
        echo "<div class='comment_box'></div>";
        echo "</div><p></div>";
    }
    $totalRegistros = totalComentarios($id);
    $totalPaginas = ceil($totalRegistros / $registrosPagina);
    for ($i = 1; $i <= $totalPaginas; $i++){
        echo "<strong><a href='producto.php?id=$id&pag=$i'>$i</a></strong>";
    }
}
```

Figura 33. “imprimirComentarios – se emplea en producto.php”

E) Mejoras a implementar

Existen numerosas mejoras que se podrían implementar a la web para dotar a esta de mayor funcionalidad de cara su uso. Estas mejoras están destinadas a los usuarios, los cuales podrían acceder a mayores tipos de opciones. Estas son las mejoras que se podrían implementar:

- Un buscador que de la opción de elegir, más detalladamente, que tipo de vehículo queremos; como por ejemplo, de poder buscar solo los coches que sean de gasolina.
- Un apartado donde el usuario pueda modificar toda su información personal: Nombre, Correo, contraseña,...
- Un apartado donde el usuario pueda modificar los vehículos que ha puesto a la venta o alquiler, y otro donde pueda ver que vehículos ha comprado o alquilado.
- Una opción de recuperación de contraseña.
- Etc.