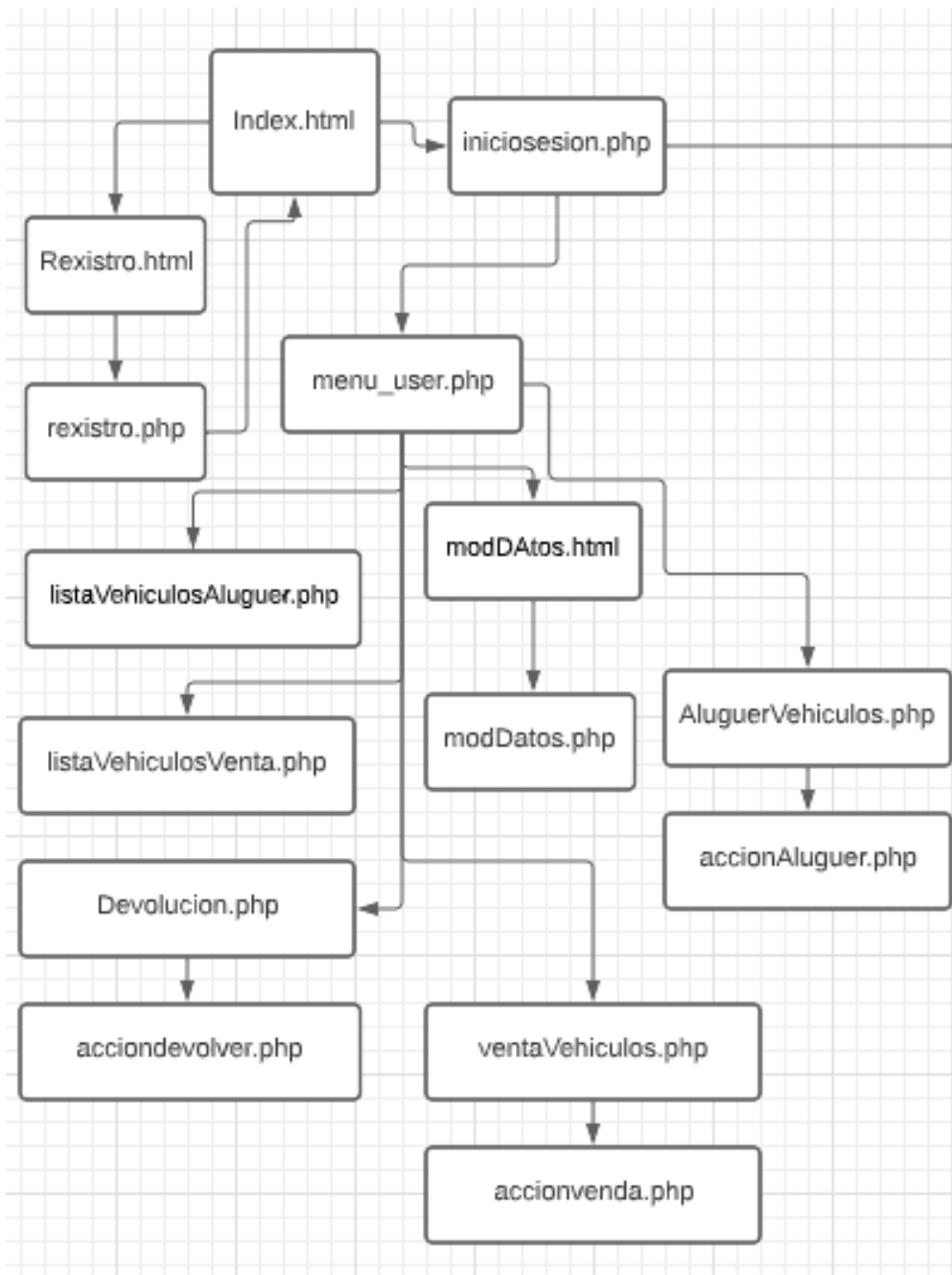
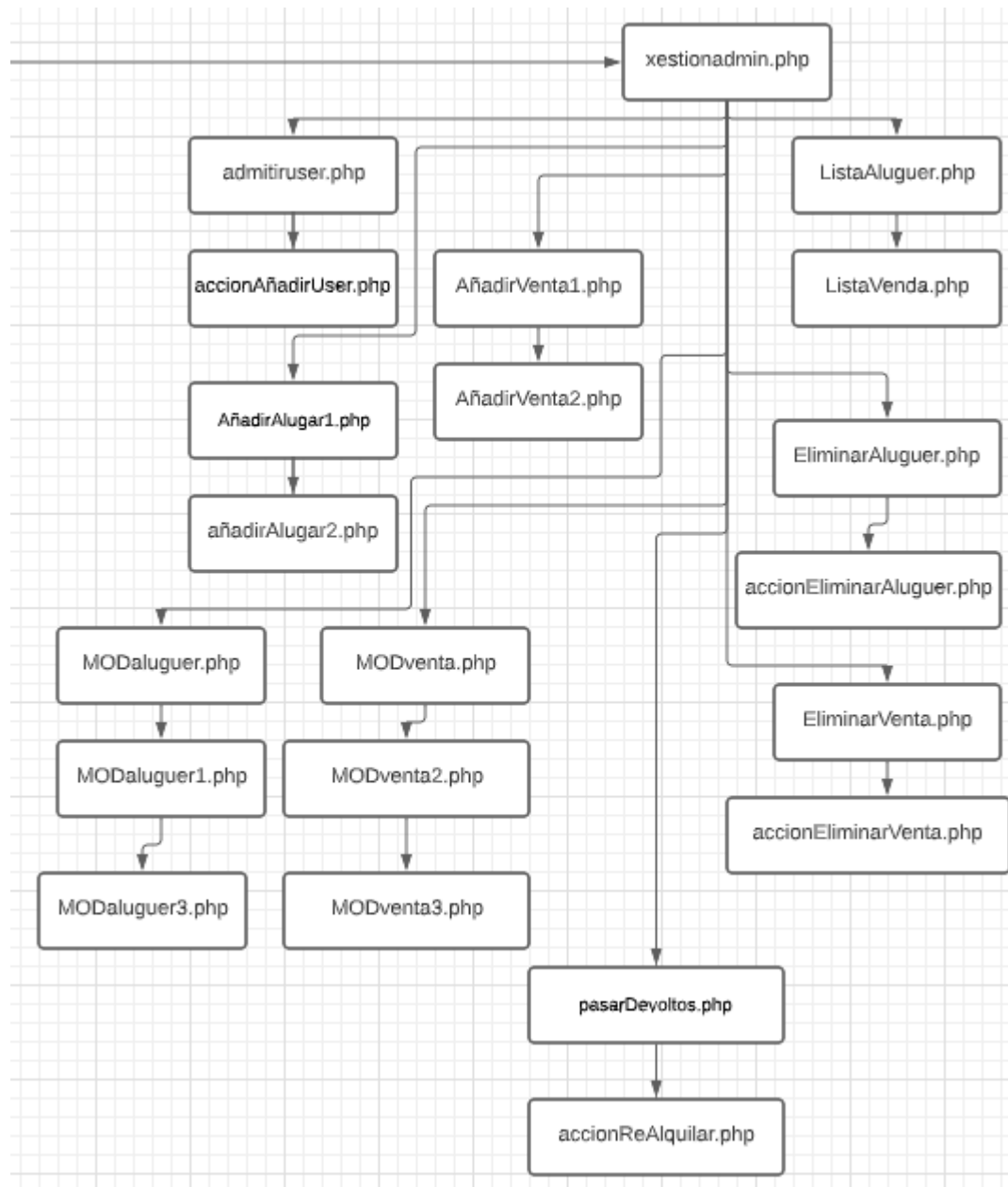


Memoria Programador

Página de Inicio	3
index.html	3
inicio sesion.php	4
Rexistro usuario	4
rexistro.html	4
rexistro.php	4
Usuarios	4
xestionadmin.php	4
menu user.php	4
Acciones Usuarios Normales	5
listaVehiculosAluguer.php	5
listaVehiculosVenta.php	5
modDAtos.html	5
modDatos.php	5
AluguerVehiculos.php	5
accionAluguer.php	5
Devolucion.php	6
acciondevolver.php	6
ventaVehiculos.php	6
accionventa.php	7
Acciones Administración	7
admitiruser.php	7
accionAñadirUser.php	7
AñadirAlugar1.php	7
añadirAlugar2.php	8
AñadirVenta1.php	8
añadirVenta2.php	8
MODaluguer.php	8
MODaluguer2.php	8
MODaluguer3.php	8
MODventa.php	8
MODventa2.php	8
MODventa3.php	9
EliminarAluguer.php	9
accionEliminarAluguer.php	9
EliminarVenta.php	9
accionEliminarVenta.php	9

ListaAluguer.php	10
ListaVenda.php	10
pasarDevoltos.php	10
accionReAlquilar.php	10





Página de Inicio

index.html

En el `index.html` encontramos la primera página de la web que contiene un form contiene dos forms uno para iniciar sesión y otro que contiene un botón para llevarnos a la página de registro

inicio sesion.php

Este archivo php recoge las credenciales de usuario y contraseña del form del html de index.php, primero comprobamos que no hayan campos vacíos, después nos conectamos a la base de datos y comprobamos que el usuario existe:

- si este no existe nos redirige a rexistro.htm.
- si el usuario existe comprueba si la contraseña y el usuario son válidos en la tabla de usuarios:
 - Si la contraseña esta es errónea nos da un aviso y nos redirige a index.html.
 - Si el usuario es válido comprueba el tipo de usuario en la tabla de usuario, si es administrador lo redirige a xestionadmin.php, y si es un usuario normal a menu_user.php.

Rexistro usuario

rexistro.html

Es un form con los campos que tiene que rellenar el usuario para registrarse, nos llevará a rexistro.php

rexistro.php

Recoge y guarda en variables los campos introducidos por el usuario en el form de registro y comprueba que no tenga campos vacíos, en caso de que haya uno vacío da el aviso de cuales estan vacios, después se conecta a la base de datos y realiza un insert con las variables que guardan los campos con los datos para añadirlos a la tabla novo_rexitro

Usuarios

xestionadmin.php

Lo primero que hace es mostrarnos la sesión del usuario, contiene un html que dentro tiene un form con los botones de todo lo que el administrador puede hacer y nos redirija en consecuencia.

menu user.php

Lo primero que hace es mostrarnos la sesión del usuario, contiene un html que dentro tiene un form con los botones de todo lo que el usuario puede hacer y nos redirija en consecuencia.

Acciones Usuarios Normales

listaVehiculosAluguer.php

Funciona de la siguiente manera, hace un select de vehiculos_aluguer para obtener los datos de los vehículos en alquiler, los guarda en una variable y con un while que itere para que a través de echos nos muestre los vehículos con sus datos.

listaVehiculosVenta.php

Funciona de la siguiente manera, hace un select de vehiculos_venta para obtener los datos de los vehículos en venta, los guarda en una variable y con un while que itere para que a través de echos nos muestre los vehículos con sus datos.

modDAtos.html

Este html contiene un form donde recogemos los datos del usuario que puede modificar y los envía a modDatos.php.

modDatos.php

Nos traemos los datos de modDAtos.html y los guardamos en variables y junto al sesion user ya tenemos todos los datos, con if comprobamos que campos estan vacios, en los que pasan dos cosas:

- Si el campo está vacío nos mostrará un mensaje indicando que faltan datos
- Si el campo tiene datos, este hará un update en la tabla usuarios actualizando dicho campo

Los campos funcionan por separado por tanto no hace falta actualizar todos a la vez sino que se pueden actualizar independientemente ya que cada if es para un campo.

AluguerVehiculos.php

Para alugar tenemos, primero hacemos un select de vehiculos_aluguer en el cual recogemos en una variable los datos y con un bucle for que contiene un while este contiene un form construido con echos para así colocar una checkbox y luego listar los datos del modelo, este form enviará el modelo seleccionado del checkbox a accionaluguer.php

accionAluguer.php

En este nos traemos el modelo del vehículo de AluguerVehiculos.php y lo metemos en una variable, con esta variable hacemos un select vehiculo_aluguer filtrado por el modelo de antes, guardamos el select en una variable con la que despues haremos un while para que nos muestre los datos del modelo y los guarde en variables, ahora tenemos un if que comprueba si hay ese modelo disponible:

- Si no lo hay, lo indica mediante un mensaje

- Si lo hay a la variable que guarda la cantidad le resta una y hace un update del modelo en la tabla `vehiculo_aluguer`, después de hacer esto, hacemos un select de `vehiculo_alugado` filtrando por el modelo y usuario, lo guardamos en una variable y con un if comprobamos si existe ya esa entrada en la tabla a través de un `numrows`,
 - Si es igual a 1, es decir ya hay ese modelo alquilado por ese usuario, se hace un update aumentando en 1 la cantidad de dicho modelo
 - Si es igual a 0, es decir que no existe esa entrada en la tabla, este hace un insert en la tabla con todos los datos que tenemos almacenados en las variables y el usuario que lo alquila.

Devolucion.php

Empieza por un select de la tabla `vehiculo alugado` filtrado por el usuario, lo guardamos en una variable, a continuación abrimos un form con echos y después con un bucle for hacemos los options para un desplegable en el cual mostraremos los modelos disponibles para devolver por el usuario, cerramos el form y nos enviamos a `acciondevolver.php`

acciondevolver.php

En este nos traemos el modelo del vehículo de `devolucion.php` y lo metemos en una variable, con esta variable hacemos un select `vehiculo_alugado` filtrado por el modelo de antes, guardamos el select en una variable con la que después haremos un while para que nos muestre los datos del modelo y los guarde en variables, ahora tenemos un if que comprueba si hay más de 1 disponible:

- Si solo hay 1, hace un delete de ese modelo y el usuario quitandolo de la tabla `alugado`
- Si hay más de 1 a la variable que guarda la cantidad le resta una y hace un update del modelo en la tabla `vehiculo_alugado`, después de hacer esto, hacemos un select de `vehiculo_devolto` filtrando por el modelo y usuario, lo guardamos en una variable y con un if comprobamos si existe ya esa entrada en la tabla a través de un `numrows`,
 - Si es igual a 1, es decir ya hay ese modelo devuelto por ese usuario, se hace un update aumentando en 1 la cantidad de dicho modelo
 - Si es igual a 0, es decir que no existe esa entrada en la tabla, este hace un insert en la tabla con todos los datos que tenemos almacenados en las variables y el usuario que lo devuelve.

ventaVehiculos.php

Para alugar tenemos, primero hacemos un select de `vehiculos_venta` en el cual recogemos en una variable los datos y con un bucle for que contiene un while este contiene un form construido con echos para así colocar una checkbox y luego listar

los datos del modelo, este form enviará el modelo seleccionado del checkbox a accionvender.php

accionvenda.php

Nos traemos el modelo de venta Vehiculos.php y lo metemos en una variable, hacemos un select de vehiculos_venta filtrado por el modelo que guardamos, lo guardamos en una variable y con un while listamos los datos del modelo y los guardamos en una variable, con un if comprobamos si hay 1 o mas vehiculos a la venta:

- si hay mas de uno, le restamos 1 a la cantidad y hacemos un update con la nueva cantidad en vehiculos_venta
- si solo hay uno hacemos un delete de ese modelo en la tabla

Después de esto hacemos un select de la tabla usuario filtrado por el usuario de la sesión y listamos los datos y los metemos en variables, con todos los datos en variables con date guardamos la fecha de ese momento en una variable, con todo esto crearemos un fopen que creara un archivo que tendra el usuario y la fecha como nombre de archivo y los datos del modelo que compra y los datos del usuario que lo compra.

Acciones Administración

admitiruser.php

Empezamos haciendo un select de la tabla novo_rexistro y lo metemos en una variable para hacer un while mostrando los datos de los usuarios nuevos hacemos un form con echos que nos lleve a accionAñadirUser.phpm

accionAñadirUser.php

Para ello haremos un select de la tabla novo_rexistro, lo guardamos en una variable para hacer un while que metera los datos en variables, siguiendo estando dentro del bucle hacemos un insert en la tabla usuario con los datos de las variables, todos estos usuarios seran del tipo normal, y aun dentro del while haremos un delete del usuario de la tabla novo_rexistro haciendo asi que el bucle itere tantas veces como usuarios haya en novo_rexistro y por cada iteracion hará el insert del usuario en la tabla usuario y el delete de la tabla novo_rexistro para quitarlo de esta.

AñadirAlugar1.php

Es un php que contiene la sesion del usuario y contine un html con un form dentro que recoge los datos del vehiculo a añadir a alugar, este form nos lleva a añadiralugar2.php

añadirAlugar2.php

Este php se encarga de guardar en variables los datos traídos de añadirAlugar1.php, después comprueba que las variables no están vacías, si hay una vacía salta un mensaje de error en la que está vacía, una vez comprobado que no hay ninguna vacía, se procederá a realizar el insert en la tabla vehiculo_aluguer

AñadirVenta1.php

Es un php que contiene la sesión del usuario y contiene un html con un form dentro que recoge los datos del vehículo a añadir a alugar, este form nos lleva a añadirVenta2.php

añadirVenta2.php

Este php se encarga de guardar en variables los datos traídos de añadirVenta1.php, después comprueba que las variables no están vacías, si hay una vacía salta un mensaje de error en la que está vacía, una vez comprobado que no hay ninguna vacía, se procederá a realizar el insert en la tabla vehiculo_venta

MODaluguer.php

Es un php que tiene un html con form en el que introduciremos el el modelo a modificar y esto nos llevará a ModAluguer2.php

MODaluguer2.php

Aqui guardaremos en una sesión el modelo y tendremos un html con un form donde introduciremos los datos a modificar y nos llevará a ModAluguer3.php

MODaluguer3.php

Abrimos la sesion donde recuperamos el modelo y guardamos en variables los datos traídos de ModAluguer2.php, a continuación comprobaremos de forma independiente con if si las variables contienen datos, si estos tienen datos hacemos un update con los datos en vehiculo_aluguer cada campo funciona de forma independiente, no hace falta rellenar todos campos a la vez.

MODventa.php

Es un php que tiene un html con form en el que introduciremos el el modelo a modificar y esto nos llevará a Modventar2.php

MODventa2.php

Aqui guardaremos en una sesión el modelo y tendremos un html con un form donde introduciremos los datos a modificar y nos llevará a Modventa3.php

MODventa3.php

Abrimos la sesion donde recuperamos el modelo y guardamos en variables los datos traídos de ModAluguer2.php, a continuación comprobaremos de forma independiente con if si las variables contienen datos, si estos tienen datos hacemos un update con los datos en vehiculo_venta, cada campo funciona de forma independiente, no hace falta rellenar todos campos a la vez.

EliminarAluguer.php

En este php tendremos un form formado por echos donde pediremos el modelo y la cantidad que queremos eliminar, y nos enviará a accionEliminarAluguer.php.

accionEliminarAluguer.php

Empezaremos por guardar en variables el modelo y la cantidad que traemos de EliminarAluguer.php, despues hacemos un select en la tabla vehiculo_aluguer filtrado por el modelo y lo metemos en una variable, con eso listamos la cantidad disponible del modelo para alquilar, creamos una variable en el que guardaremos la resta de la cantidad disponible menos la que queremos eliminar, con if comprobamos si la cantidad final es mayor que 0:

- Si es mayor que 0, hacemos un update cambiando la cantidad a la cantidad final.
- Si es menor que 0, entonces hace un delete para eliminar ese modelo de la tabla.

EliminarVenda.php

En este php tendremos un form formado por echos donde pediremos el modelo y la cantidad que queremos eliminar, y nos enviará a accionEliminarVenda.php.

accionEliminarVenda.php

Empezaremos por guardar en variables el modelo y la cantidad que traemos de EliminarVenda.php, despues hacemos un select en la tabla vehiculo_aluguer filtrado por el modelo y lo metemos en una variable, con eso listamos la cantidad disponible del modelo para alquilar, creamos una variable en el que guardaremos la resta de la cantidad disponible menos la que queremos eliminar, con if comprobamos si la cantidad final es mayor que 0:

- Si es mayor que 0, hacemos un update cambiando la cantidad a la cantidad final.
- Si es menor que 0, entonces hace un delete para eliminar ese modelo de la tabla.

ListaAluguer.php

Funciona de la siguiente manera, hace un select de vehiculos_aluguer para obtener los datos de los vehículos en alquiler, los guarda en una variable y con un while que itere para que a través de echos nos muestre los vehículos con sus datos.

ListaVenda.php

Funciona de la siguiente manera, hace un select de vehiculos_venda para obtener los datos de los vehículos en venta, los guarda en una variable y con un while que itere para que a través de echos nos muestre los vehículos con sus datos.

pasarDevoltos.php

Lo primero que hacemos es un select de la tabla vehiculo_devolto, lo guardamos en una variable y hacemos un while para mostrar todos los modelos que hay para devolver y tenemos un boton que nos llevará a accionReAlquilar.php

accionReAlquilar.php

Lo primero que hacemos es un select de la tabla vehiculo_devolto, lo guardamos en una variable y hacemos un while para guardar en variables los campos con los datos del modelo, hacemos un select de vehiculo_aluguer filtrado por modelo en el que sacamos el numrows, comprobamos que ese modelo exista en la tabla vehiculo_aluguer, como en la tabla de vehiculo_aluguer dejamos a 0 los modelos en vez de eliminarlos, solo tendremos que hacer un update para volcar los vehículos y borrarlos también de la tabla vehiculo_devolto por cada iteración del bucle.