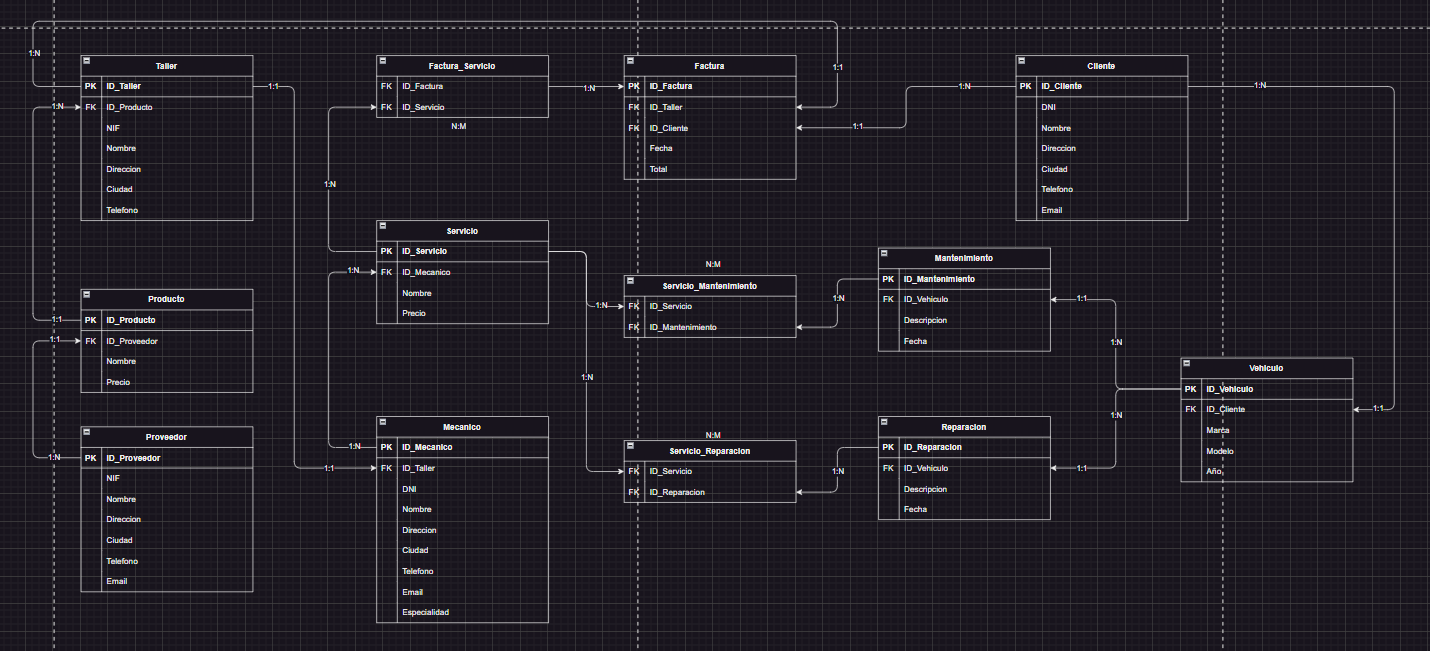
Elige una base de datos, la que quieras, pero al menos que tenga 6 tablas.

Con ella has de hacer:

* El diseño del modelo entidad-relación
* CREARLA ENTERA DESDE PHP
* Comprobar que se crea todo correctamente en PHPMYADMIN
* Inserta datos en esas tablas desde php
* Comprueba que todos los datos estén correctamente insertados
* Crea al menos dos formularios en php, los que quieras con el motivo que quieras y los explicas, para qué y con qué finalidad están hechos
* Utiliza al menos 3 funciones (aparte de las absolutamente necesarias para todo lo anterior) y me dices para que sirven en tu código concretamente.

Mi idea con esta práctica es que podamos usar la mayoría de cosas de las que hemos hecho uso hasta ahora, cuantas más añadas mejor. Y sobre todo intenta que haya pantallazos de todo a la entrega y que todo esté correcta y concretamente explicado.

1. Modelo Entidad-Relación.

* Proveedor - Producto:

Un proveedor puede suministrar muchos productos, pero un producto solo puede ser suministrado por un proveedor. Esto representa una relación uno a muchos entre Proveedor y Producto.

* Taller - Producto:

Un taller puede tener muchos productos, pero un producto pertenece solo a un taller. Esto es una relación uno a muchos entre Taller y Producto.

* Taller - Mecánico:

Cada taller puede tener muchos mecánicos, pero un mecánico solo trabaja en un taller. Esto sugiere una relación uno a muchos entre Taller y Mecánico.

* Taller - Factura:

Cada taller puede proporcionar muchas facturas, pero una factura pertenece a un taller. Esto es una relación uno a muchos entre Taller y Factura.

* Cliente - Vehículo:

Un cliente puede tener muchos vehículos, pero un vehículo pertenece solo a un cliente. Esto representa una relación uno a muchos entre Cliente y Vehículo.

* Servicio - Mecánico:

Un servicio puede ser realizado por muchos mecánicos, pero un mecánico puede realizar varios servicios. Esto es una relación muchos a muchos entre Servicio y Mecánico, que se implementa mediante la tabla intermedia Servicio\_Mecanico.

* Reparación - Vehículo:

Cada reparación está asociada a un vehículo, pero un vehículo puede tener múltiples reparaciones. Esto representa una relación uno a muchos entre Reparación y Vehículo.

* Servicio - Reparación:

Un servicio puede estar relacionado con muchas reparaciones y una reparación puede requerir varios servicios. Esto también es una relación muchos a muchos, implementada a través de la tabla intermedia Servicio\_Reparación.

* Mantenimiento - Vehículo:

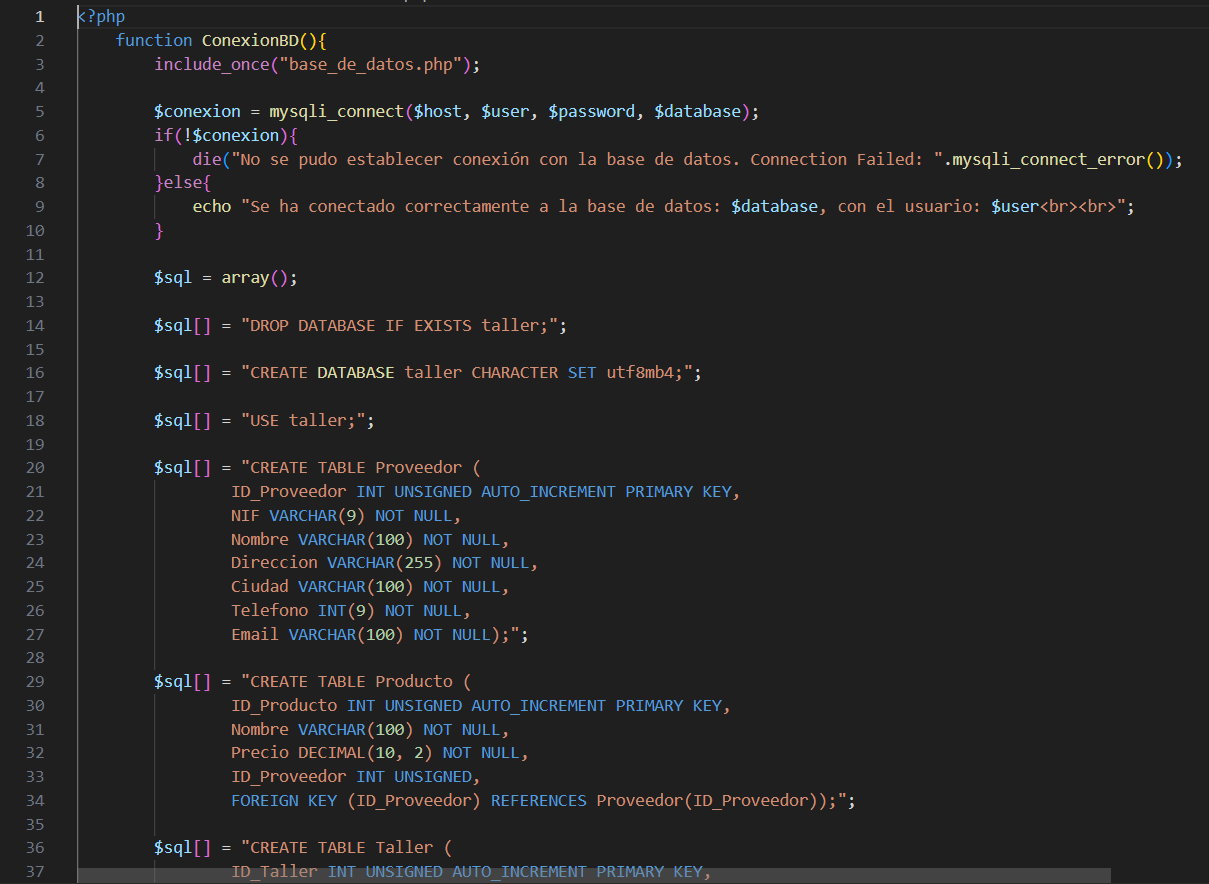
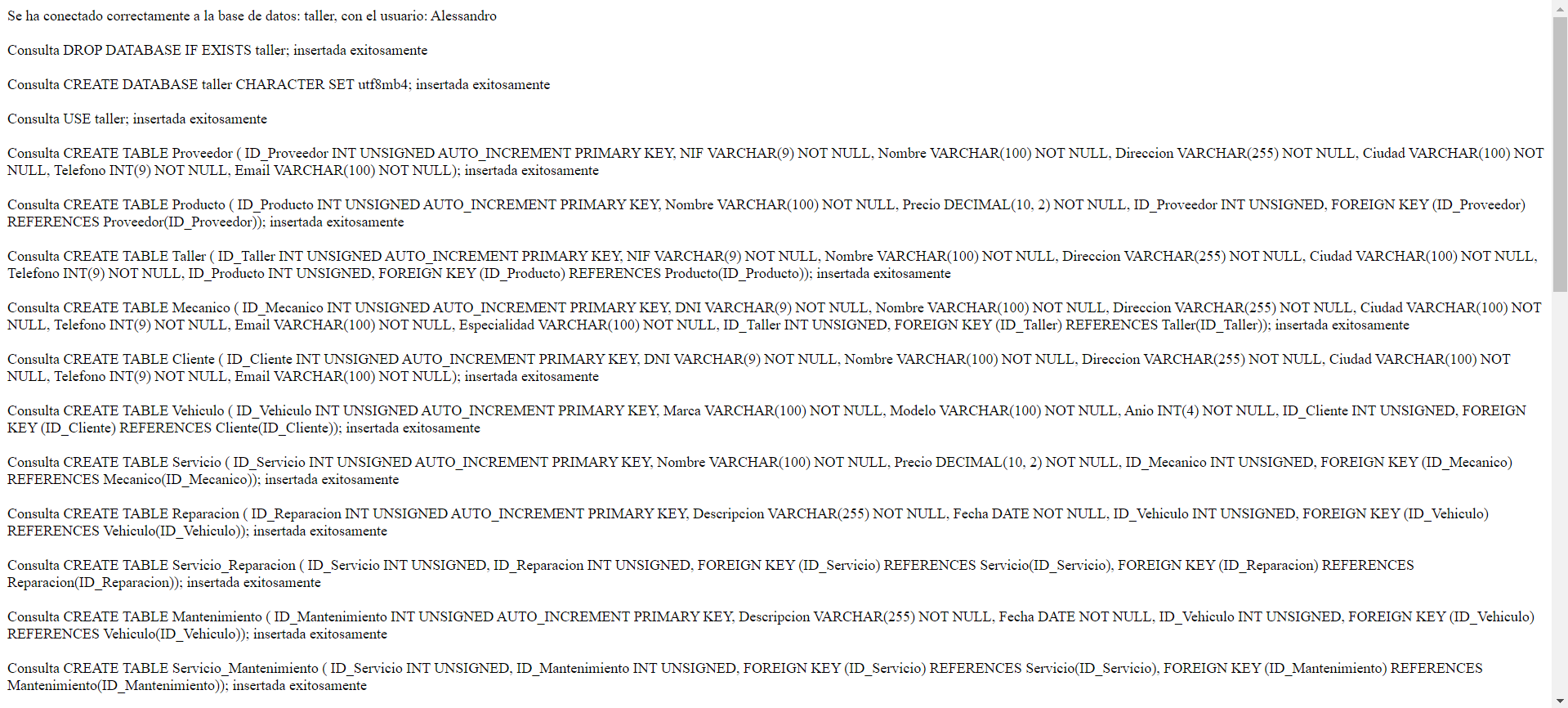
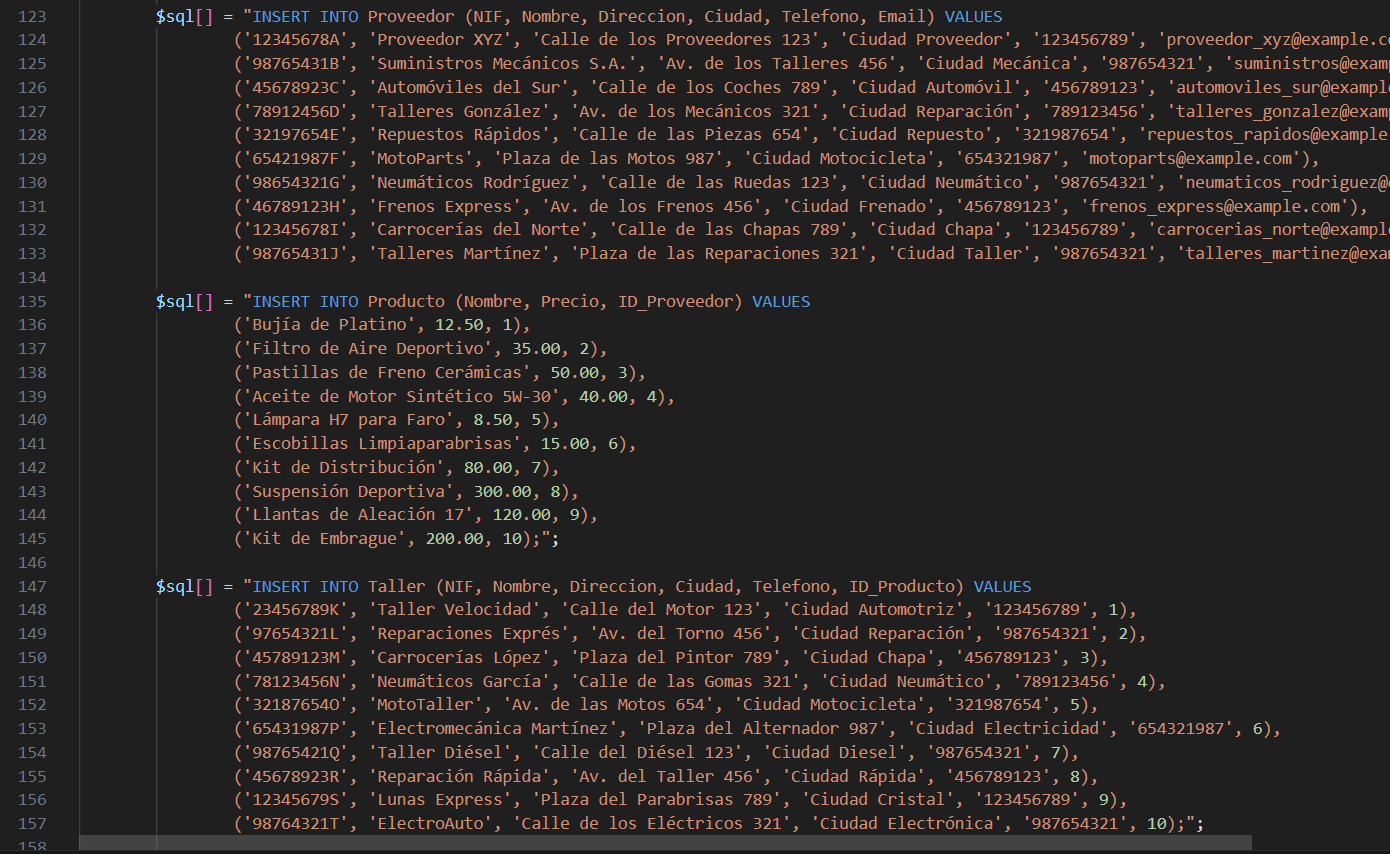
Similar a la relación entre Reparación y Vehículo, un mantenimiento está asociado a un vehículo, pero un vehículo puede tener múltiples mantenimientos. Esto es una relación uno a muchos entre Mantenimiento y Vehículo.

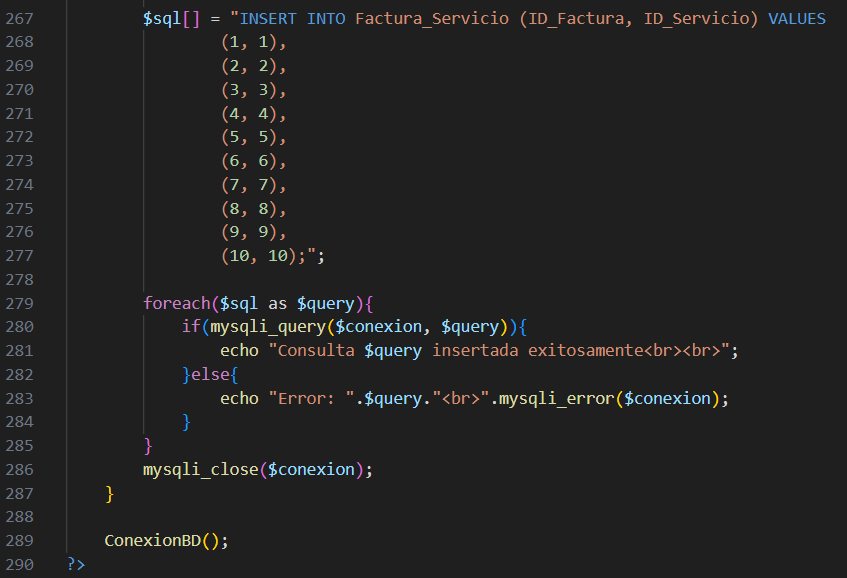
* Servicio - Mantenimiento:

Al igual que la relación entre Servicio y Reparación, un servicio puede estar relacionado con muchos mantenimientos y un mantenimiento puede requerir varios servicios. Esta es otra relación muchos a muchos, implementada a través de la tabla intermedia Servicio\_Mantenimiento.

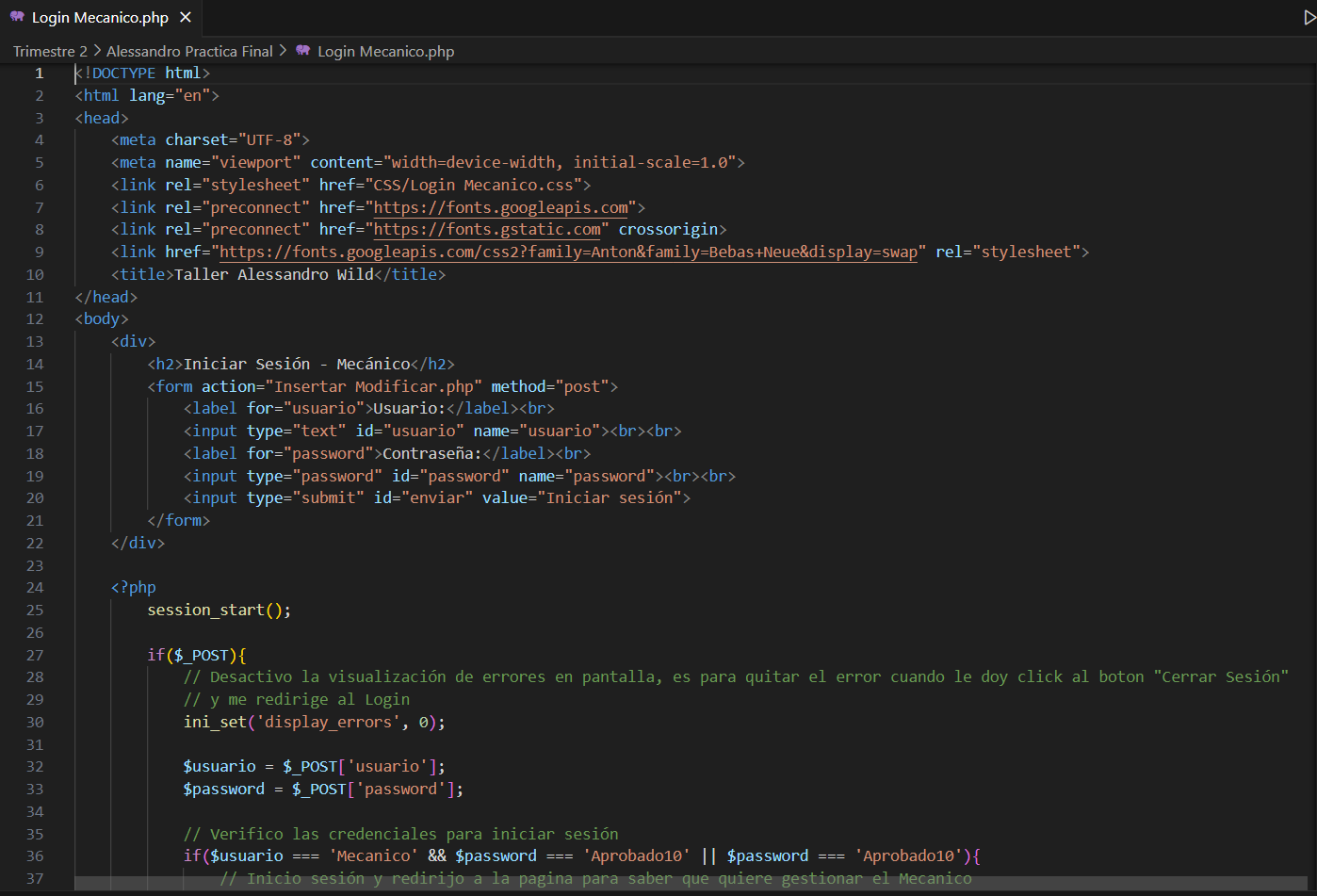
* Factura - Servicio:

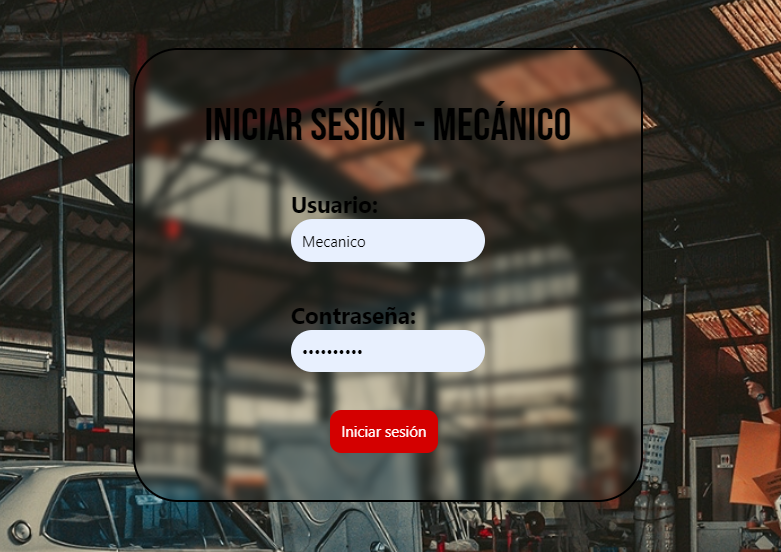
Una factura puede incluir múltiples servicios y un servicio puede estar presente en varias facturas. Esto es una relación muchos a muchos, implementada mediante la tabla intermedia Factura\_Servicio.

1. Creación de las tablas desde PHP.
2. Comprobación de la creación de las tablas.
3. Inserción de las tablas.

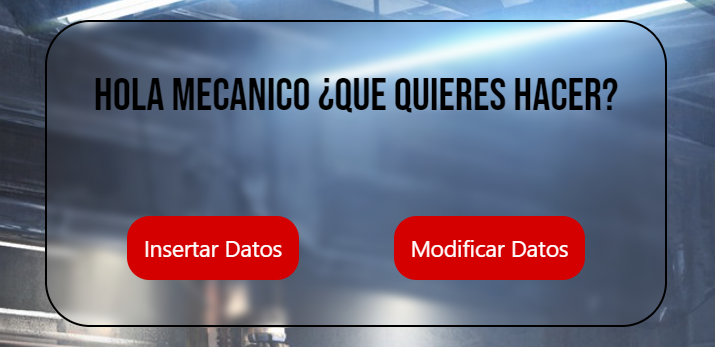


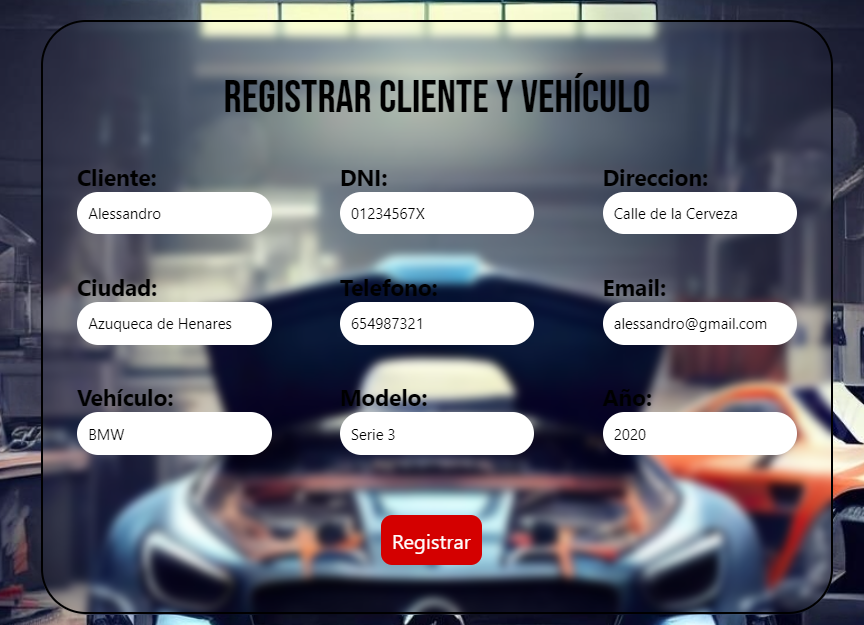
1. Comprobación de la inserción de los datos.
2. Creación de los formularios.

He creado un Login para que el Mecánico acceda a la Base de Datos del Taller y pueda gestionarla, añadiendo clientes y vehículos de los clientes.



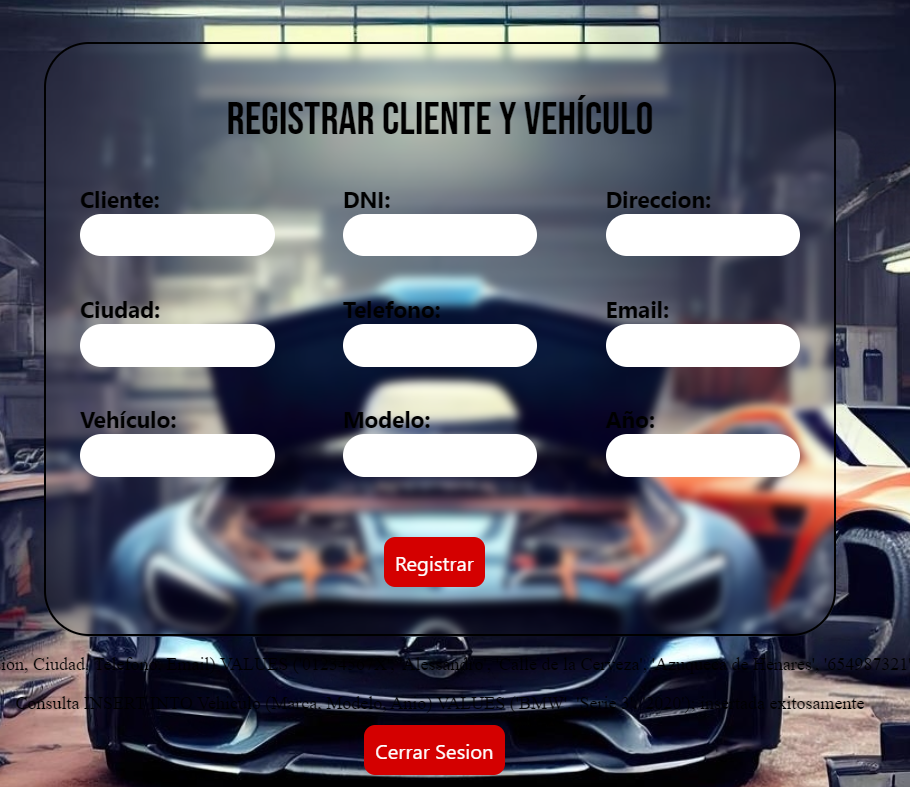
Puedes acceder con cualquier nombre de usuario (Mecánico, Maribel, Alessandro, etc.), suponiendo que eres el mecánico, sin embargo, la contraseña siempre va a ser “Aprobado10”, sino no vas a poder acceder para gestionar el Taller.

Una vez Iniciado sesión te redirige a otra página, la cual tienes 2 opciones, Registrar a un Cliente y su Vehículo o Modificar los datos del Cliente con su Vehículo.

En “Insertar Datos” tenemos otro formulario para registrar al Cliente con su Vehículo.

Al clicar el botón “Registrar” manda los datos a la Base de Datos.



Luego nos aparecerá otro botón para Cerrar Sesión si ya no queremos registrar a otro Cliente.

Y nos mandará al Login del Mecánico.

En el caso de Modificar los Datos del Cliente el proceso es muy parecido.

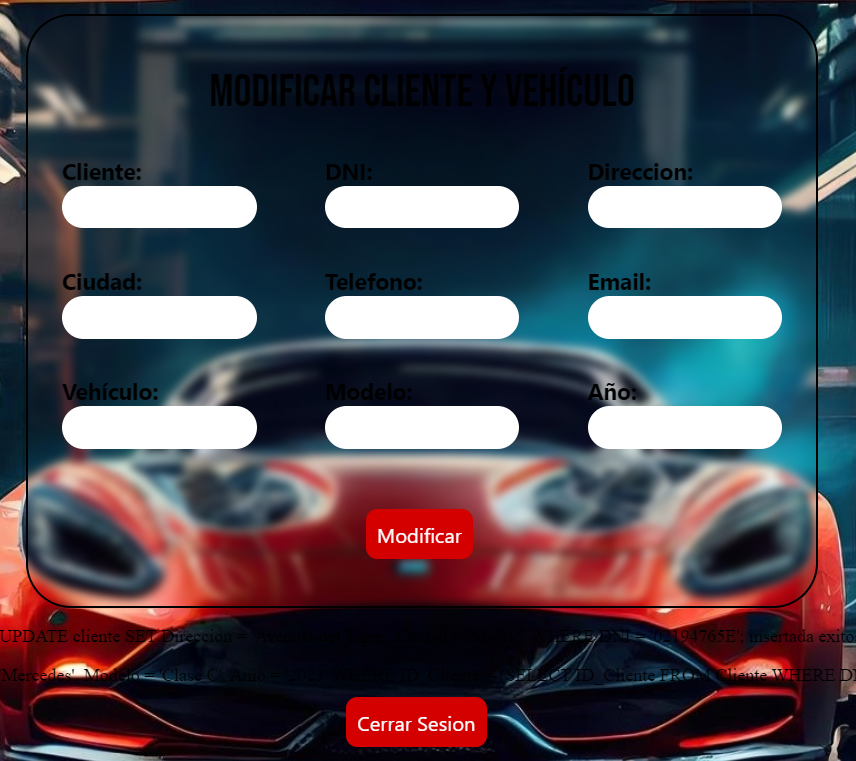
Solo que estamos obligados a rellenar todos los campos excepto el Teléfono y el Email (son opcionales porque no vamos a utilizar esos datos).

Los demás datos si son necesarios dado el caso de si el Cliente a cambiado de vivienda o ha cambiado de Vehículo.

Esta vez vamos a modificar la Dirección de vivienda de Pedro Martinez que se ha mudado de Ciudad y ha cambiado su Hyundai por un Mercedes.

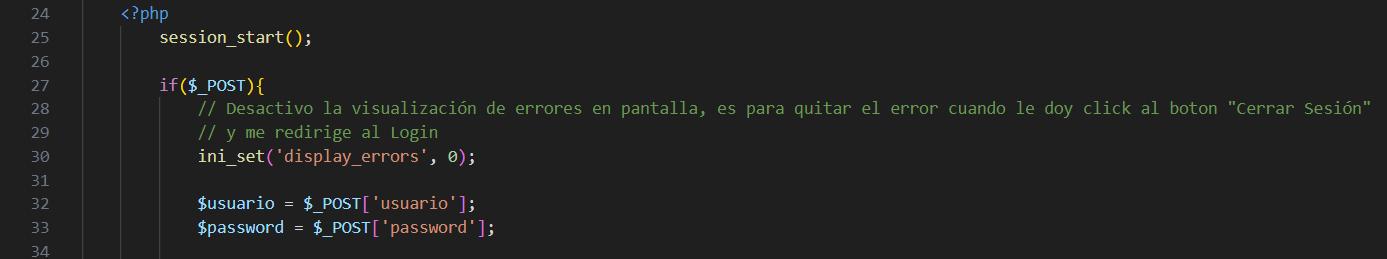
Vemos que se han actualizado los datos.



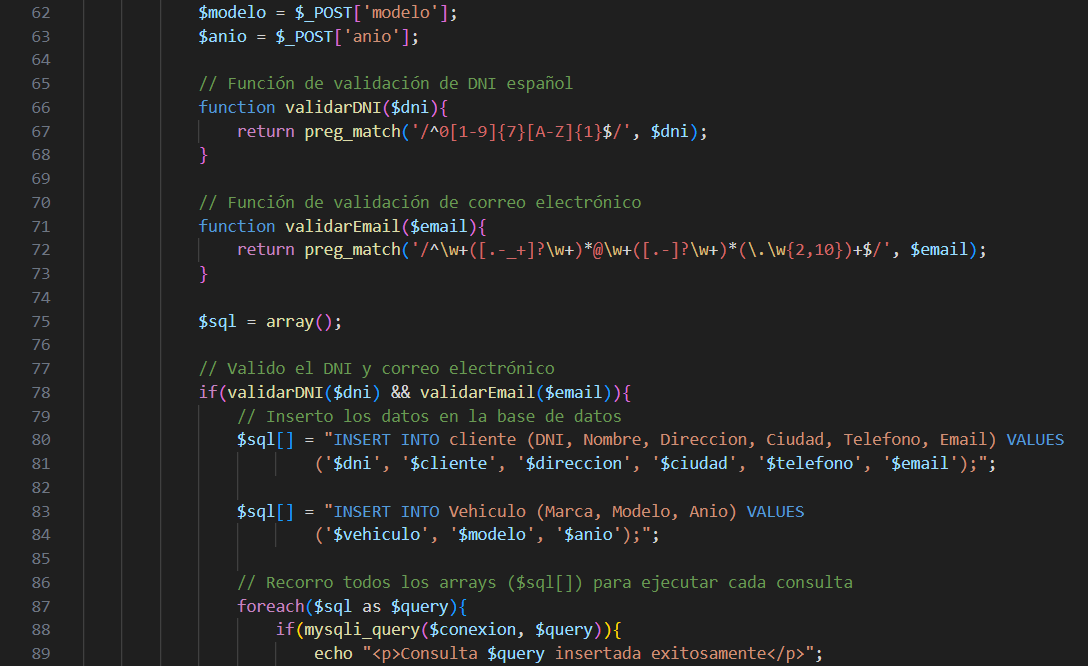
Y también nos aparece un botón de “Cerrar Sesión” una vez modificados los datos.

1. Funciones distintas que he utilizado.

ini\_set('display\_errors', 0);

Sirve para desactivar la visualización de errores en PHP, por ejemplo, al darle click al botón “Cerrar Sesion” y que me redirija a la pagina de Login del Mecanico, me aparecía un error de que no había insertado ningún valor en el campo usuario o password.



He creado 2 funciones que sirven para validar tanto el DNI como el Email utilizando expresiones regulares.

Y utilizo la función preg\_match(), que sirve para buscar un patrón específico dentro de una cadena de caracteres y comprobar si ese patrón está presente o no.