

## "AutoData Manager: Sistema Integral para la Gestión de Información Vehicular"

Andrés Felipe Pulido Prieto - 129790

Juan David Ardila Escobar - 130530

Juan Diego Gómez Fonseca – 129393

Ingeniería de Sistemas

Universidad ECCI

31 de agosto del 2025

## ***1. Introducción***

El presente proyecto tiene como finalidad el diseño e implementación de una base de datos estructurada y segura, que sirva como soporte para la gestión de información en un sistema orientado al manejo de vehículos y sus características. A lo largo del desarrollo, se aplicaron buenas prácticas de modelado mediante diagramas entidad-relación, casos de uso y secuencias, que permiten representar de manera clara la interacción entre los actores y el sistema.

Además, se incorporaron procesos de normalización para garantizar la integridad de los datos, así como medidas de seguridad y roles que delimitan los accesos según los perfiles de usuario. Paralelamente, se diseñó una interfaz de interacción que complementa la funcionalidad del sistema, brindando una experiencia más intuitiva. Este trabajo incluye la documentación técnica necesaria, lo cual facilita la comprensión del sistema y asegura la escalabilidad de la solución en escenarios futuros.

## ***2. Objetivo General***

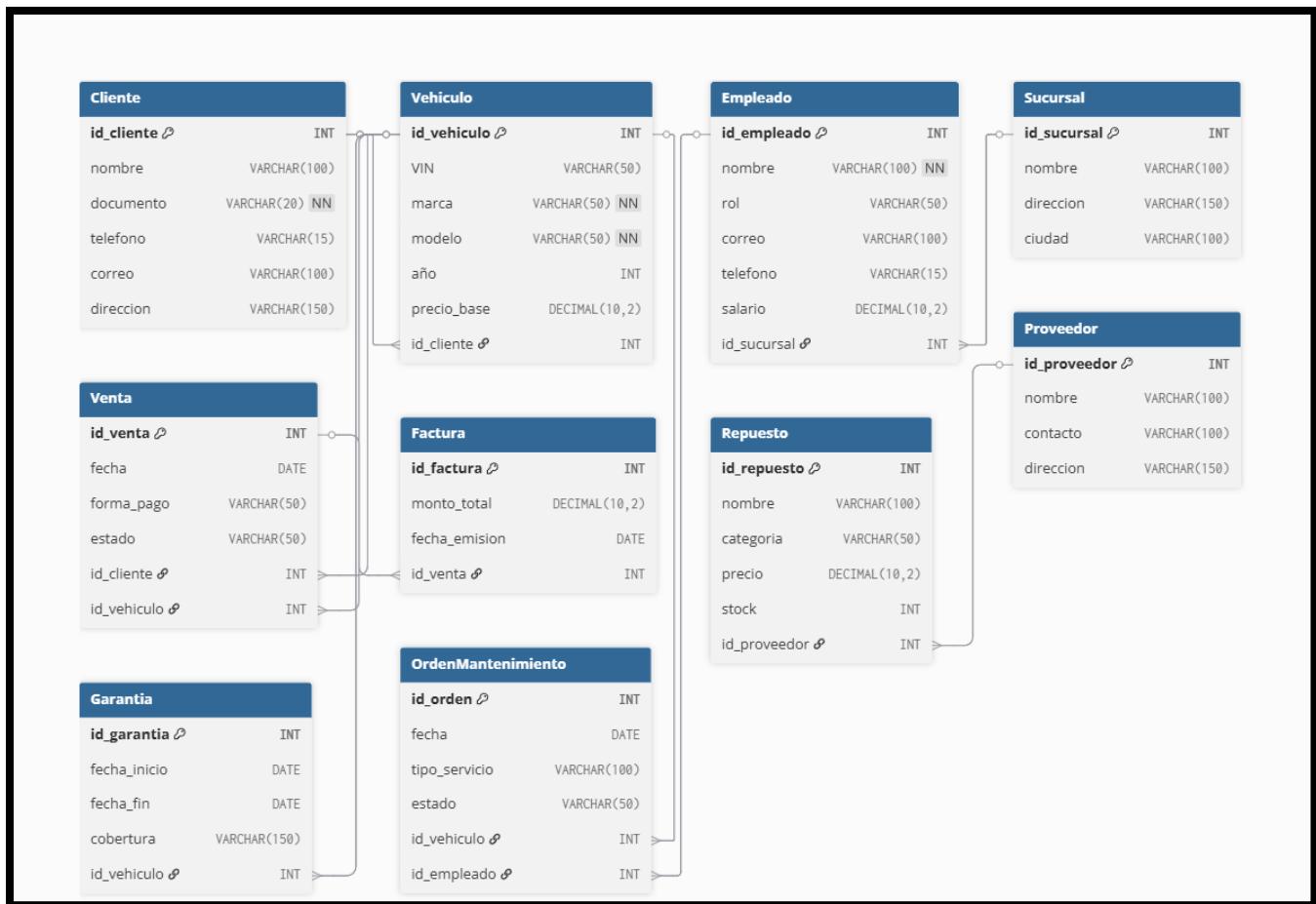
Diseñar e implementar una base de datos relacional optimizada y segura, acompañada de sus respectivos diagramas, normalización, roles de usuario e interfaz gráfica, con el fin de garantizar una gestión eficiente y confiable de la información relacionada con los vehículos y su administración.

## ***3. Objetivos Específicos***

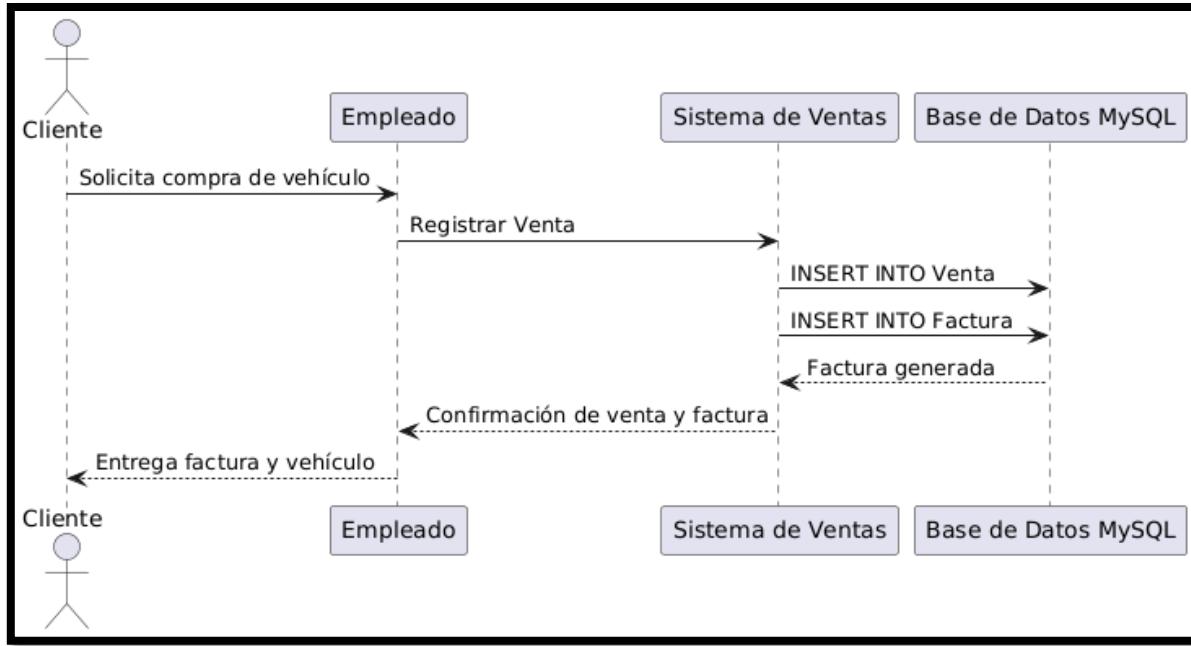
- Diseñar y estructurar una base de datos relacional que permita almacenar de forma organizada y segura la información de los vehículos, incluyendo características técnicas, propietarios y estado actual.
- Implementar una interfaz gráfica en JavaScript que facilite la interacción con los datos de los vehículos, permitiendo registrar, consultar, modificar y eliminar información de manera intuitiva.
- Optimizar los procesos de consulta y gestión de datos vehiculares mediante la integración de funciones de búsqueda y filtrado, garantizando rapidez, precisión y usabilidad en el sistema.

#### 4. Diagramas:

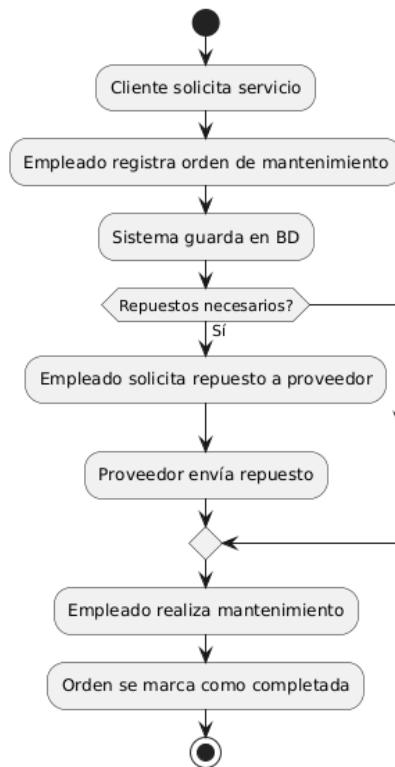
##### a. Diagrama de Clase:



**b. Diagrama de Secuencia (Compra de un vehículo con Factura)**



**c. Diagrama de Actividad (Mantenimiento de un vehículo)**



## 5. Características:

- a. **Estructurada y relacional:** Diseñada bajo el modelo relacional, con 10 tablas principales y relaciones claras mediante llaves primarias y foráneas.
- b. **Integridad referencial:** Se aplican llaves foráneas para garantizar la consistencia entre clientes, vehículos, ventas, facturas, empleados y proveedores.
- c. **Normalización:** La base de datos está organizada hasta la 3<sup>a</sup> Forma Normal (3FN), eliminando redundancias y evitando anomalías en inserción, actualización y eliminación.
- d. **Escalabilidad:** Su diseño permite agregar más tablas o relaciones en el futuro (ej. nuevos servicios, sucursales, historial de mantenimientos).
- e. **Seguridad y control**
  - i. Restricciones como UNIQUE en documentos y correos de clientes.
  - ii. Uso de tipos de datos apropiados (ej. DECIMAL para precios, DATE para fechas).
- f. **Soporte Para la gestión Integral:** Administra información de clientes, ventas, vehículos, facturas, empleados, proveedores, repuestos y mantenimientos en un solo sistema.
- g. **Compatibilidad:** Creada en MySQL/PHP, pero portable a otros motores SQL como PostgreSQL o SQL Server.
- h. **Optimización para consultas:** Las tablas están preparadas para consultas frecuentes como:
  - i. historial de compras de un cliente,
  - ii. repuestos por proveedor
  - iii. facturas generadas en un rango de fechas,
  - iv. servicios de mantenimiento realizados por empleado.

## 6. Roles

- a. **Admin:** Puede realizar cualquier cambio en la DB, agregar, borrar modificar llaves, columnas es decir un acceso total al sistema.
- b. **Empleado:** Puede registrar y consultar contenido de las tablas y columnas, pero esta limitado no puede ni agregar, borrar o eliminar llaves, columnas y tablas.
- c. **Cliente:** Puede consultar y actualizar su propia información, pero tiene las limitaciones del rol de Empleado.

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

	rol_id	nombre	descripcion
▶	1	Administrador	Acceso total al sistema, gestión de usuarios y datos
	2	Empleado	Acceso limitado para registrar y consultar información
*	3	Cliente	Acceso para consultar y actualizar su propia información
	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

	usuario_id	nombre	correo	rol_id	contraseña
▶	1	juanperez	juan.perez@mail.com	1	K4ys3r#202438+
	2	mariagarcia	maria.garcia@mail.com	2	Maria#20245%
*	3	carloslopez	carlos.lopez@mail.com	2	Carlos#202845%
	4	lauraquintero	laura.quintero@mail.com	3	laura123+
	NULL	NULL	NULL	NULL	NULL

## 7. Seguridad.

### a. Control de Usuarios:

- i. Cada usuario se guarda en la tabla Usuarios con un rol asignado (rol\_id).
- ii. Así, el sistema sabe qué puede y qué no puede hacer cada persona.
- iii. Esto permite auditar y controlar quién realiza cada acción.

**b. Contraseñas:**

- i. Política de Ciberseguridad, la contraseña debe ser cambiada cada 90 días.
- ii. La contraseña es de uso personal.
- iii. La contraseña debe ser alfanumérica, debe tener al menos 8 caracteres, y un carácter especial. (¡”#\$%&/).

**8. Tablas:**

<b>Tabla</b>	<b>Llave primaria (PK)</b>	<b>Llave foránea (FK)</b>	<b>Función</b>
Cliente	Id_Cliente	-	Almacena la información de los clientes, como nombre, documento, correo y dirección.
vehículo	Id_vehículo	id_cliente	Registra los vehículos asociados a cada cliente, incluyendo marca, modelo, año y precio base.
Empleado	Id_Empleado	id_sucursal	Contiene los datos de los empleados, su rol, correo, teléfono, salario y la sucursal a la que pertenecen.
Sucursal	Id_Sucursal		Guarda la información de las sucursales, incluyendo nombre, dirección y ciudad.
Proveedor	Id_Proveedor		Registra los proveedores de repuestos, con nombre, contacto y dirección.
Repuesto	Id_Repuesto	id_proveedor	Contiene los repuestos disponibles, su categoría, precio, stock y el proveedor asociado.
Venta	Id_Venta	id_cliente, id_vehiculo	Almacena las ventas realizadas, incluyendo fecha, forma de pago y estado. Relaciona cliente y vehículo vendidos.
Factura	Id_Factura	id_venta	Registra las facturas generadas por las ventas, con monto total y fecha de emisión.
Garantía	Id_Garantía	id_vehiculo	Registra las garantías asociadas a los vehículos, incluyendo fechas de inicio y fin, cobertura y vehículo correspondiente.
Orden Mantenimiento	Id_Orden Mantenimiento	id_vehiculo, id_empleado	Controla las órdenes de mantenimiento de vehículos, con fecha, tipo de servicio y empleado responsable.
Usuarios	Id_Usuarios		Guarda el registro de usuarios independientemente si es Admin, Empleado o Cliente
Roles	Id_Rol	rol_Id	Clasifica a los Usuarios en Roles para determinar delegaciones.

- *Todas las Tablas:*

The screenshot shows the MySQL Workbench interface with the title bar "Tablas DB". Below the title bar, there are tabs for "Items Cliente", "Items Empleado", and "Items Factura". The main area contains a command-line window with the following text:  
1 • USE motosdb;  
2 • SHOW TABLES;  
3  
Below the command window is a "Result Grid" table with the following data:  
Tables\_in\_motosdb  
cliente  
empleado  
factura  
garantia  
orden\_mantenimiento  
proveedor  
repuesto  
roles  
sucursal  
▶ usuarios  
vehiculo  
venta

- *Tabla de Clientes:*

The screenshot shows the MySQL Workbench interface with the title bar "Tablas DB". Below the title bar, there are tabs for "Items Cliente", "Items Empleado", and "Items Factura". The main area contains a command-line window with the following text:  
1 • USE motosdb;  
2 • SHOW TABLES;  
3 • DESCRIBE cliente  
Below the command window is a "Result Grid" table with the following data:  
Field Type Null Key Default Extra  
id\_cliente int(11) NO PRI NULL auto\_increment  
nombre varchar(100) NO NULL  
documento varchar(50) YES UNI NULL  
telefono varchar(20) YES NULL  
correo varchar(100) YES NULL  
direccion varchar(150) YES NULL

- *Tabla de Empleado:*

The screenshot shows the MySQL Workbench interface with the 'empleado' table selected. The table has the following structure:

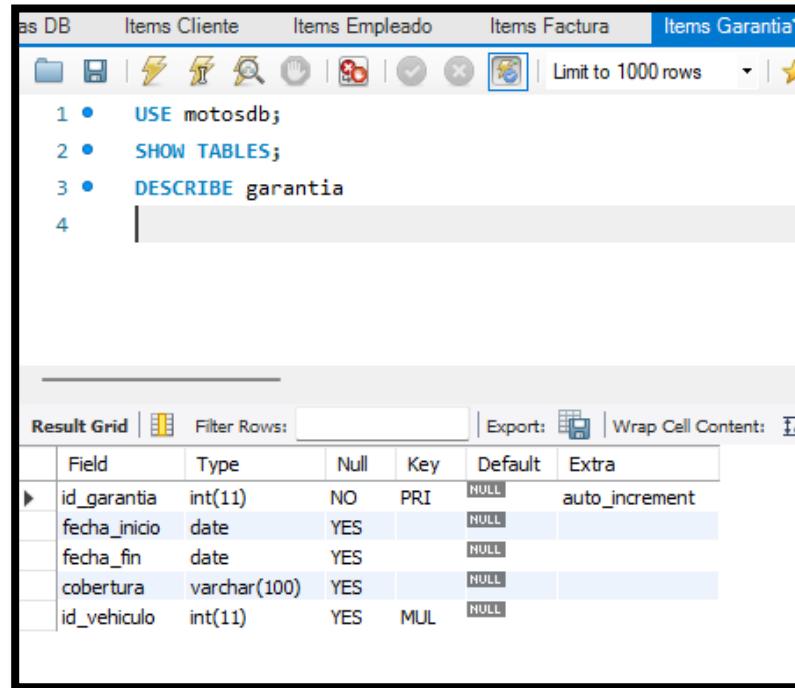
Field	Type	Null	Key	Default	Extra
id_empleado	int(11)	NO	PRI	NULL	auto_increment
nombre	varchar(100)	YES		NULL	
rol	varchar(50)	YES		NULL	
correo	varchar(100)	YES		NULL	
salario	decimal(10,2)	YES		NULL	
id_sucursal	int(11)	YES	MUL	NULL	

- *Tabla de Factura:*

The screenshot shows the MySQL Workbench interface with the 'factura' table selected. The table has the following structure:

Field	Type	Null	Key	Default	Extra
id_factura	int(11)	NO	PRI	NULL	auto_increment
monto_total	decimal(10,2)	YES		NULL	
fecha_emision	date	YES		NULL	
id_venta	int(11)	YES	MUL	NULL	

- *Tabla de garantía:*



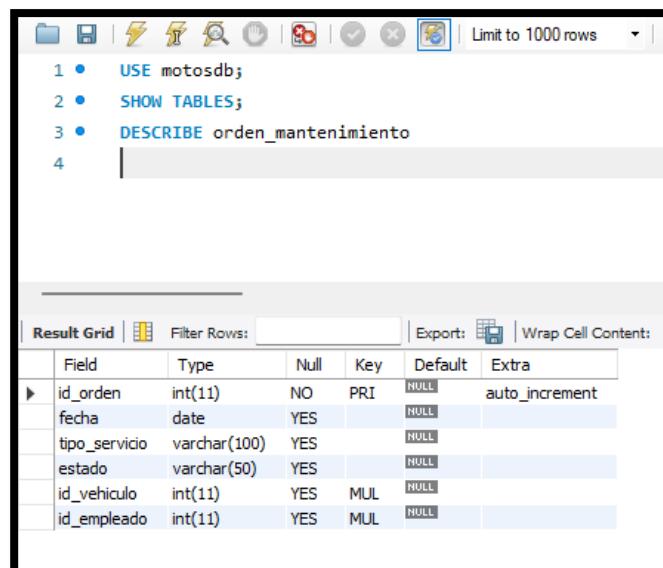
The screenshot shows the MySQL Workbench interface with the 'Items Garantia' tab selected. In the SQL editor pane, the following commands are displayed:

```
1 • USE motosdb;
2 • SHOW TABLES;
3 • DESCRIBE garantia
4
```

Below the SQL editor is the 'Result Grid' pane, which displays the structure of the 'garantia' table:

	Field	Type	Null	Key	Default	Extra
▶	id_garantia	int(11)	NO	PRI	NULL	auto_increment
	fecha_inicio	date	YES		NULL	
	fecha_fin	date	YES		NULL	
	cobertura	varchar(100)	YES		NULL	
	id_vehiculo	int(11)	YES	MUL	NULL	

- *Tabla de orden Mantenimiento:*



The screenshot shows the MySQL Workbench interface with the 'Items Garantia' tab selected. In the SQL editor pane, the following commands are displayed:

```
1 • USE motosdb;
2 • SHOW TABLES;
3 • DESCRIBE orden_mantenimiento
4
```

Below the SQL editor is the 'Result Grid' pane, which displays the structure of the 'orden\_mantenimiento' table:

	Field	Type	Null	Key	Default	Extra
▶	id_orden	int(11)	NO	PRI	NULL	auto_increment
	fecha	date	YES		NULL	
	tipo_servicio	varchar(100)	YES		NULL	
	estado	varchar(50)	YES		NULL	
	id_vehiculo	int(11)	YES	MUL	NULL	
	id_empleado	int(11)	YES	MUL	NULL	

- *Tabla de Proveedor:*

The screenshot shows the MySQL Workbench interface with the 'proveedor' table selected. The table has four columns: id\_proveedor (int(11), primary key, auto\_increment), nombre (varchar(100)), contacto (varchar(100)), and direccion (varchar(150)).

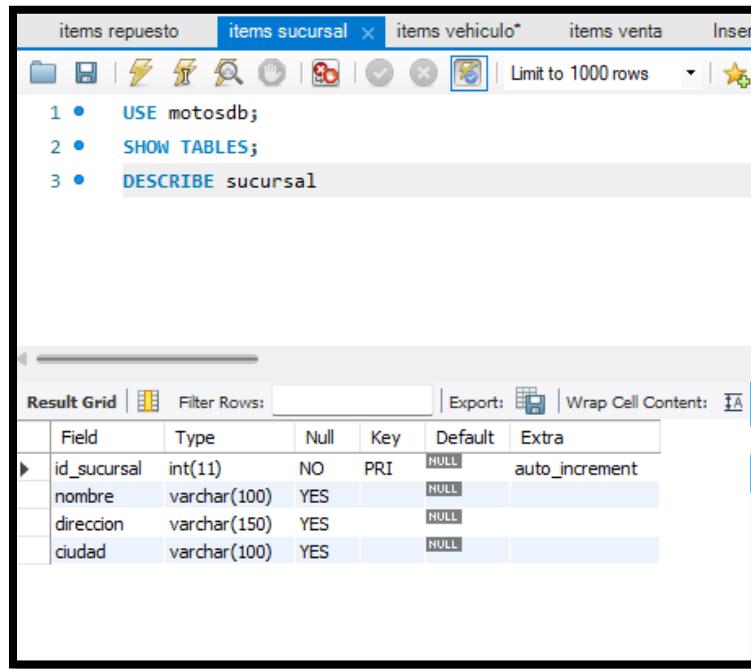
Field	Type	Null	Key	Default	Extra
id_proveedor	int(11)	NO	PRI	NULL	auto_increment
nombre	varchar(100)	YES		NULL	
contacto	varchar(100)	YES		NULL	
direccion	varchar(150)	YES		NULL	

- *Tabla de Repuesto:*

The screenshot shows the MySQL Workbench interface with the 'repuesto' table selected. The table has six columns: id\_reuesto (int(11), primary key, auto\_increment), nombre (varchar(100)), categoria (varchar(50)), precio (decimal(10,2)), stock (int(11)), and id\_proveedor (int(11), foreign key). The foreign key constraint is defined as MUL.

Field	Type	Null	Key	Default	Extra
id_reuesto	int(11)	NO	PRI	NULL	auto_increment
nombre	varchar(100)	YES		NULL	
categoria	varchar(50)	YES		NULL	
precio	decimal(10,2)	YES		NULL	
stock	int(11)	YES		NULL	
id_proveedor	int(11)	YES	MUL	NULL	

- *Tabla de Sucursal:*



The screenshot shows the MySQL Workbench interface with the 'items sucursal' tab selected. The SQL editor pane contains the following code:

```

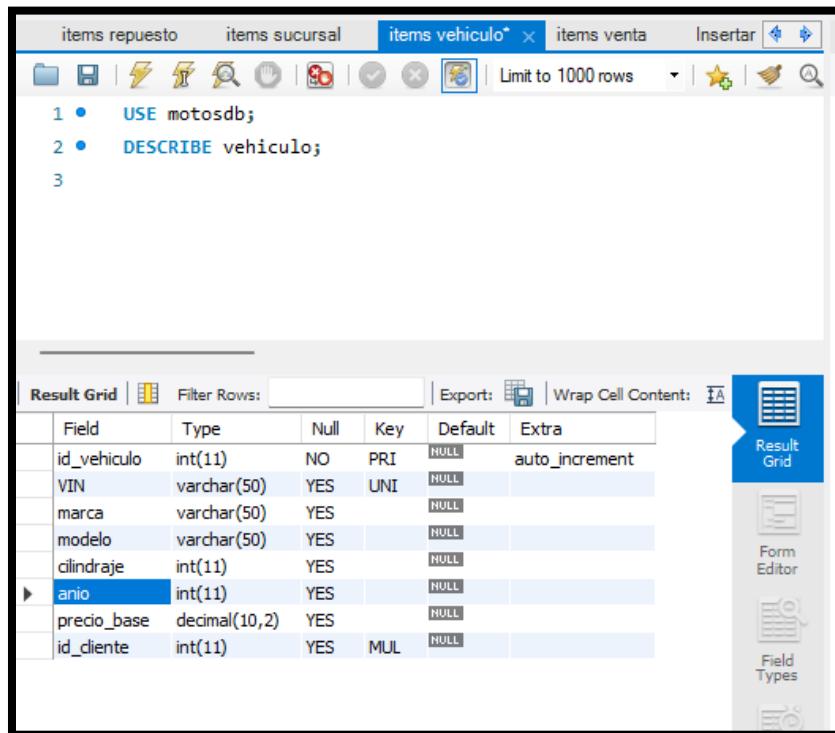
1 • USE motosdb;
2 • SHOW TABLES;
3 • DESCRIBE sucursal

```

The Result Grid pane displays the structure of the 'sucursal' table:

	Field	Type	Null	Key	Default	Extra
▶	id_sucursal	int(11)	NO	PRI	NULL	auto_increment
	nombre	varchar(100)	YES		NULL	
	direccion	varchar(150)	YES		NULL	
	ciudad	varchar(100)	YES		NULL	

- *Tabla de Vehículo:*



The screenshot shows the MySQL Workbench interface with the 'items vehiculo\*' tab selected. The SQL editor pane contains the following code:

```

1 • USE motosdb;
2 • DESCRIBE vehiculo;
3

```

The Result Grid pane displays the structure of the 'vehiculo' table:

	Field	Type	Null	Key	Default	Extra
▶	id_vehiculo	int(11)	NO	PRI	NULL	auto_increment
	VIN	varchar(50)	YES	UNI	NULL	
	marca	varchar(50)	YES		NULL	
	modelo	varchar(50)	YES		NULL	
	cilindraje	int(11)	YES		NULL	
	anio	int(11)	YES		NULL	
	precio_base	decimal(10,2)	YES		NULL	
	id_cliente	int(11)	YES	MUL	NULL	

A sidebar on the right shows icons for 'Result Grid', 'Form Editor', and 'Field Types'.

- *Tabla de Venta:*

The screenshot shows the MySQL Workbench interface. At the top, there are tabs for 'items repuesto', 'items sucursal', 'items vehiculo\*', and 'items venta'. The 'items venta' tab is active. Below the tabs is a toolbar with icons for file operations, search, and other database functions. A status bar at the bottom indicates 'Limit to 1000 rows'. The main area contains the following SQL code:

```
1 • USE motosdb;
2 • SHOW TABLES;
3 • DESCRIBE venta
4
```

Below the code, a 'Result Grid' is displayed showing the structure of the 'venta' table:

	Field	Type	Null	Key	Default	Extra
▶	id_venta	int(11)	NO	PRI	NULL	auto_increment
	fecha	date	YES		NULL	
	forma_pago	varchar(50)	YES		NULL	
	estado	varchar(50)	YES		NULL	
	id_cliente	int(11)	YES	MUL	NULL	
	id_vehiculo	int(11)	YES	MUL	NULL	

- *Tabla de Roles:*

The screenshot shows the MySQL Workbench interface. The 'items venta' tab is active. Below the tabs is a toolbar with icons for file operations, search, and other database functions. A status bar at the bottom indicates 'Limit to 1000 rows'. The main area contains the following SQL code:

```
1 • describe roles;
```

Below the code, a 'Result Grid' is displayed showing the structure of the 'roles' table:

	Field	Type	Null	Key	Default	Extra
▶	rol_id	int(11)	NO	PRI	NULL	auto_increment
	nombre	varchar(50)	NO		NULL	
	descripcion	text	YES		NULL	

On the right side of the interface, there is a sidebar with two buttons: 'Result Grid' (which is highlighted in blue) and 'Form Editor'.

- **Tablas de Usuarios:**

```
1 •    describe usuarios;
```

Field	Type	Null	Key	Default	Extra
usuario_id	int(11)	NO	PRI	NULL	auto_increment
nombre	varchar(100)	YES		NULL	
correo	varchar(100)	YES	UNI	NULL	
rol_id	int(11)	YES	MUL	NULL	
contraseña	text	YES		NULL	

- **Interfaz:**

- Esta interfaz web permite registrar y visualizar motos almacenadas en la base de datos MotosDB. La página está estructurada en dos bloques principales:

#### **Formulario de Registro**

- Incluye campos como VIN, marca, modelo, cilindraje, año, precio base e ID del cliente.
- Cuando el usuario llena los datos y hace clic en "Guardar Moto", la información se envía mediante método POST al archivo process.php, que se encarga de insertar el registro en la base de datos.
- También muestra mensajes visuales como “ Moto registrada con éxito” o “ Hubo un error”, dependiendo del resultado de la operación, lo que mejora la experiencia del usuario.

### Tabla de Visualización

- Debajo del formulario aparece una tabla titulada “Listado de Motos”.
- Esta tabla se genera automáticamente al incluir el archivo mostrar.php, que realiza una consulta a la base de datos y muestra los registros existentes de manera ordenada.
- Los estilos aplicados dan una apariencia moderna con efectos de hover, colores de contraste y diseño tipo tarjeta para mayor legibilidad.

ID	VIN	Marca	Modelo	Cilindrada	Año	Precio Base	ID
1	FY599D	Yamaha	FZ25	250	2022	12240000.00	1
2	DS971H	Honda	CB190R	190	2021	10500000.00	2
3	QBWBDC	Suzuki	Gixxer	150	2023	9905000.00	3
16	VN002345678	KTM	Duke	200	2023	16275000.00	14
17	VN003456789	TVS	Apache	160 AV	2022	11730000.00	15
18	VN004567890	Yamaha	MT-15	150	2024	17400000.00	16

Esta página web permite registrar nuevos clientes y visualizar los registros existentes en la base de datos, facilitando la administración de clientes vinculados a las motos registradas.

La interfaz se divide en dos secciones principales:

## 1. Formulario de Registro de Clientes

- a. Sigue los datos clave como nombre, documento, teléfono, correo y dirección
- b. Los campos están validados como requeridos para evitar registros incompletos.
- c. Al presionar “Guardar Cliente”, la información se envía mediante método POST al archivo process\_cliente.php, que se encarga de guardar los datos en la base.
- d. El sistema muestra mensajes visuales de éxito o error, informando al usuario si el registro fue procesado correctamente, lo que mejora la experiencia de uso.

## 2. Listado Dinámico de Clientes

- a. Despues del formulario, se muestra una tabla titulada “Listado de Clientes”.
- b. Esta tabla se carga automáticamente desde el archivo mostrarClientes.php, que consulta la base de datos y muestra los registros de manera organizada.
- c. Se aplican estilos visuales para hacer la tabla más legible y profesional, con efectos hover y alternancia de colores entre filas.

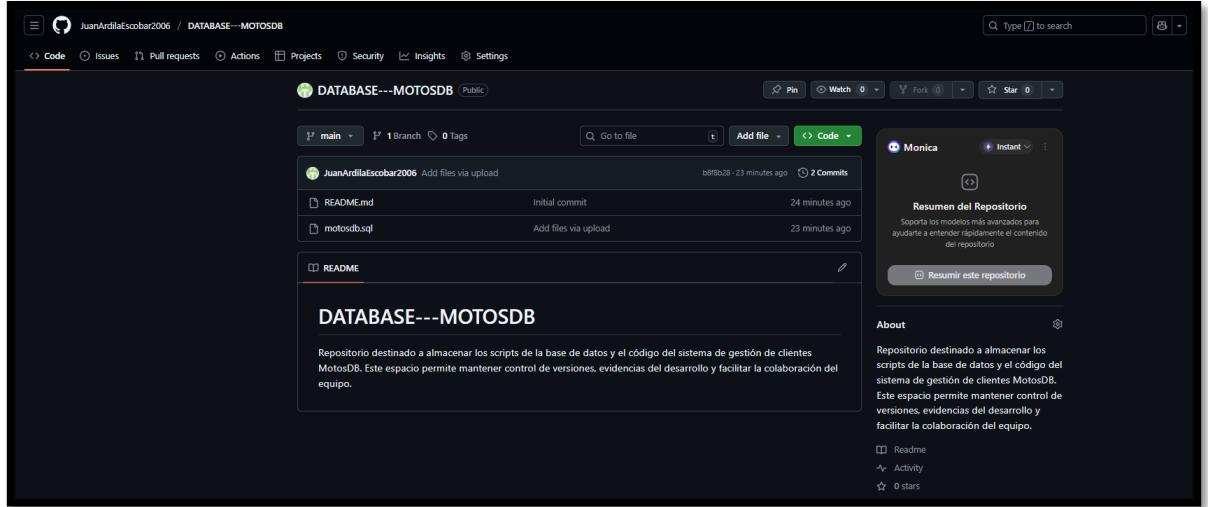
## 3. Diseño y experiencia de usuario

- a. Se utiliza un estilo moderno con degradados y sombras, para simular una interfaz profesional tipo sistema administrativo.
- b. Los campos tienen efecto de foco luminoso, lo que guía la interacción del usuario.
- c. El botón tiene animación al pasar el cursor, mejorando la percepción visual del sistema.

The screenshot shows a web application interface. At the top, there is a header bar with the URL 'localhost/MotosDB/indexCliente.php'. Below the header, there is a 'Registro de Clientes' form containing fields for Nombre, Documento, Teléfono, Correo, and Dirección, each with a corresponding input field. Below the form is a blue button labeled 'Guardar Cliente'. Underneath the button is a section titled 'Listado de Clientes' which includes a message 'Conexión exitosa a la base de datos.' and a table displaying client data. The table has columns: ID, Nombre, Documento, Teléfono, Correo, and Dirección. The data in the table is as follows:

ID	Nombre	Documento	Teléfono	Correo	Dirección
1	Juan Pérez	12345678	3001112233	juanperez@gmail.com	Calle 10 #12-34
2	Maria Gómez	98765432	3109876543	mariagomez@hotmail.com	Calle 5 #45-22
3	Carlos Ramírez	45678912	3204567890	carlosramirez@yahoo.com	Calle 12 #8 sur #20-21
14	Andrés Torres	11223344	3019876543	andrestorres@gmail.com	Carrera 8 #14-25
15	Laura Mendoza	22334455	3201122345	lauramendoza@hotmail.com	Avenida 9 #23-41
16	Franco Gómez	33445566	3112233445	francogomez@yahoo.com	Calle 12 #9-18
17	Sofía Ramírez	44556677	3023344556	softaramirez@gmail.com	Transversal 5 #67-20
18	Carmela Herrera	55667788	3004455667	carmelaherrera@outlook.com	Diagonal 21 #45-40

- *Repository:*



- <https://github.com/JuanArdilaEscobar2006/DATABASE---MOTOSDB>