

# REPORTE DE PRÁCTICA NO. 1

Flotillas de Autos(Base de Datos)

ALUMNO: Cesar Martinez Andrade  
Dr. Eduardo Cornejo-Velázquez

18/08/24 — UAEH — LCC



# 1. Introducción

Este proyecto de base de datos tiene como objetivo centralizar y gestionar de manera eficiente la información relacionada con una flota de vehículos, sus operadores, los servicios prestados, los clientes, el mantenimiento de los autos y los seguros. La implementación de esta base de datos permitirá un control detallado y organizado de todos estos aspectos, facilitando la consulta y manipulación de los datos mediante el uso de SQL (Structured Query Language). A través de modelos entidad/relación y modelos relacionales, se ilustrará la estructura y las relaciones entre las diferentes tablas que componen esta base de datos.

A través de esta base de datos, buscamos centralizar la información y mejorar la eficiencia en la gestión de los datos, asegurando un control detallado y organizado de todos los aspectos relacionados con la flota de vehículos y sus operaciones. La implementación de esta base de datos no solo facilitará la consulta y manipulación de los datos, sino que también permitirá una mejor toma de decisiones basada en información precisa y actualizada.

Todo esto gracias a la relaciones que existen entre si en las tablas, y tambien por los conocimientos dados en la clase de Base de datos distribuidas en el presente semestre del presente año. Aplicando los conocimientos dados en estas clases.

## 2. Marco teórico

### Análisis de requerimientos

El presente proyecto tiene como objetivo desarrollar una base de datos integral para gestionar y organizar información crucial relacionada con una flota de vehículos, sus operadores, los servicios prestados, los clientes, el mantenimiento de los autos y los seguros. La implementación de esta base de datos permitirá centralizar y estructurar de manera efectiva todos estos datos, facilitando su consulta y manipulación.

#### Conceptos Clave

- **Gestión de Bases de Datos:** Proceso de almacenar, organizar y gestionar datos de manera eficiente.
- **Gestion de Bases de Clientes/Operadores:**Administracion de datos de empleados y usuarios para su consulta rapida.
- **Gestión de Flotas de Vehículos:** Administración y mantenimiento de una flota de vehículos para asegurar su operatividad y eficiencia.
- **Mantenimiento de Vehículos:** Conjunto de actividades realizadas para mantener los vehículos en buen estado de funcionamiento.
- **Seguros de Vehículos:** Contratos que proporcionan cobertura financiera en caso de daños o pérdidas de los vehículos.

### Modelo Entidad - Relación

Para la generación de los modelos relacionales y entidad-relación, utilizamos EDR Plus[1], lo que nos permitió visualizar y estructurar adecuadamente las relaciones entre las diferentes tablas que componen la base de datos. Las tablas principales incluyen:

- **Autos:** Información detallada sobre cada vehículo, incluyendo placa, marca, modelo, año y estado actual.
- **Operador:** Datos de los conductores, como nombre, licencia de conducir, información de contacto e historial de servicios.
- **Servicios:** Registro de cada servicio prestado, con detalles como fecha, hora, punto de partida y destino, operador asignado y vehículo utilizado.

- **Cliente:** Información de los clientes, incluyendo nombre, contacto e historial de servicios utilizados.
- **Mantenimiento:** Control del mantenimiento de cada vehículo, registrando fechas, tipo de servicio y costos asociados.
- **Seguros:** Información sobre las pólizas de seguro de cada vehículo, incluyendo compañía aseguradora, número de póliza, fechas de cobertura y detalles de la misma.

## Modelo relacional

La teoría de bases de datos relacionales, propuesta por Edgar F. Codd en 1970[5], establece que los datos deben organizarse en tablas que se relacionan entre sí mediante claves primarias y foráneas. Este modelo relacional guía la estructura de nuestra base de datos, asegurando que los datos se organicen de manera lógica y eficiente. Las relaciones entre las tablas se definen de la siguiente manera:

- **Auto** se relaciona con **Mantenimiento**, **Seguro** y **Servicio** a través de la clave primaria (**idAuto**).
- **Operador** se relaciona con **Servicios** mediante la clave primaria (**idOperador**).
- **Cliente** se relaciona con **Servicios** mediante la clave primaria (**idCliente**).

## SQL

Para la compilación y ejecución de líneas de código en lenguaje SQL, utilizamos SQL OnLine[6]. Esta plataforma nos permitió definir, manipular y consultar los datos de manera eficiente. Además, empleamos el bloc de notas para agrupar y generar rápidamente las tuplas necesarias, optimizando el tiempo y esfuerzo invertido en la inserción de datos.

## 3. Herramientas empleadas

Este proyecto de base de datos tiene como objetivo centralizar y gestionar de manera eficiente la información relacionada con una flota de vehículos, sus operadores, los servicios prestados, los clientes, el mantenimiento de los autos y los seguros. Para lograr esto, utilizamos las siguientes herramientas:

1. **EDR Plus:** Para la generación de los modelos relacionales y entidad-relación, lo que nos permitió visualizar y estructurar adecuadamente las relaciones entre las diferentes tablas..
2. **SQL Online:** Para la compilación de líneas de código en lenguaje SQL, facilitando la definición, manipulación y consulta de los datos.
3. **Bloc de notas o Notepad++:** Para agrupar y generar más rápidamente las tuplas necesarias usando las tabulaciones y comandos de copiado y pegado.

A través de estos recursos, logramos una implementación eficiente y organizada de la base de datos.

## 4. Desarrollo

### Análisis de requisitos

1. **Autos:** Esta tabla contendrá información detallada sobre cada vehículo de la flota, incluyendo el número de identificación del vehículo (Placa), la marca, el modelo, el año, el color, y el estado actual del vehículo.
2. **Operador:** Aquí se almacenarán los datos de los conductores, como su nombre, licencia de conducir, información de contacto, y el historial de servicios realizados.
3. **Servicios:** Esta tabla registrará cada uno de los servicios prestados, incluyendo la fecha, la hora, el punto de partida y destino, el operador asignado, y el vehículo utilizado.
4. **Cliente:** En esta tabla se guardará la información de los clientes, como su nombre, información de contacto, y el historial de servicios que han utilizado.
5. **Mantenimiento:** Esta tabla será crucial para llevar un control del mantenimiento de cada vehículo, registrando las fechas de mantenimiento, el tipo de servicio realizado, y los costos asociados.
6. **Seguros:** Finalmente, esta tabla contendrá información sobre las pólizas de seguro de cada vehículo, incluyendo la compañía aseguradora, el número de póliza, la fecha de inicio y fin de la cobertura, y los detalles de la cobertura.

Para la implementación de esta base de datos, utilizaremos el lenguaje SQL (Structured Query Language), que nos permitirá definir, manipular y consultar los datos de manera eficiente. Además, se presentarán modelos de entidad/relación y modelos relacionales para ilustrar la estructura y las relaciones entre las diferentes tablas de la base de datos.

## Modelo Entidad - Relación

En la Tabla 1 se presenta la propuesta de Modelo Entidad - Relación para la flotilla de automoviles estableciendo una X donde existe una relacion.

Table 1: Matriz de realaciones.

Entidades	Autos	Operador	Servicios	Cliente	Mantenimiento	Seguros
Autos			X		X	X
Operador			X			
Servicios	X	X		X		
Cliente			X			
Mantenimiento	X					
Seguros	X					

En la Figura 1 se presenta la propuesta de Modelo Entidad - Relación para. el caso.....

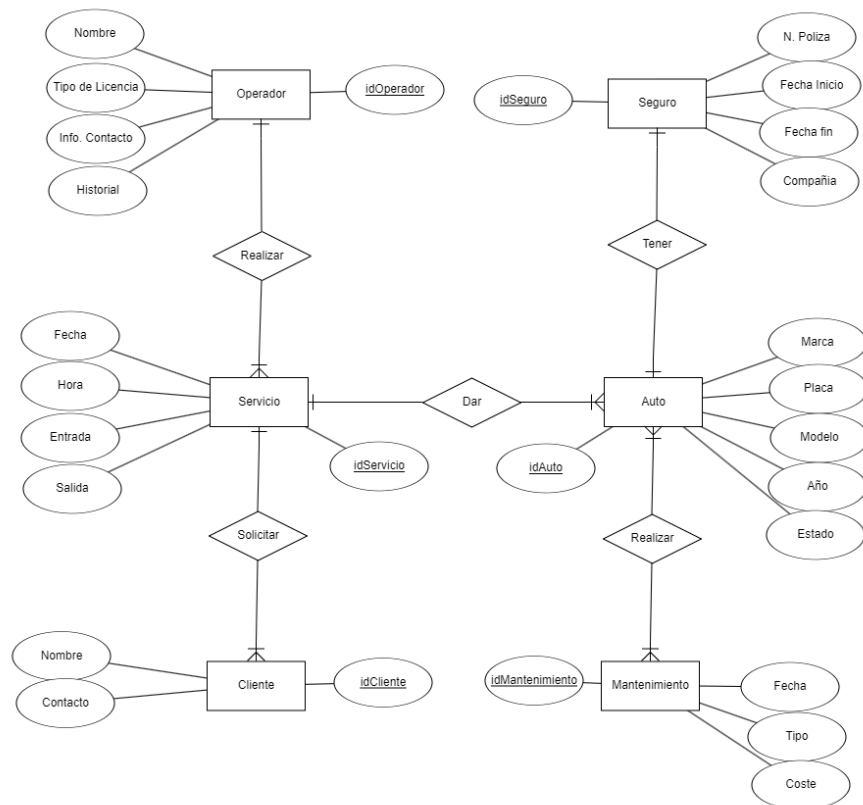


Figure 1: Modelo Entidad - Relación propuesto.

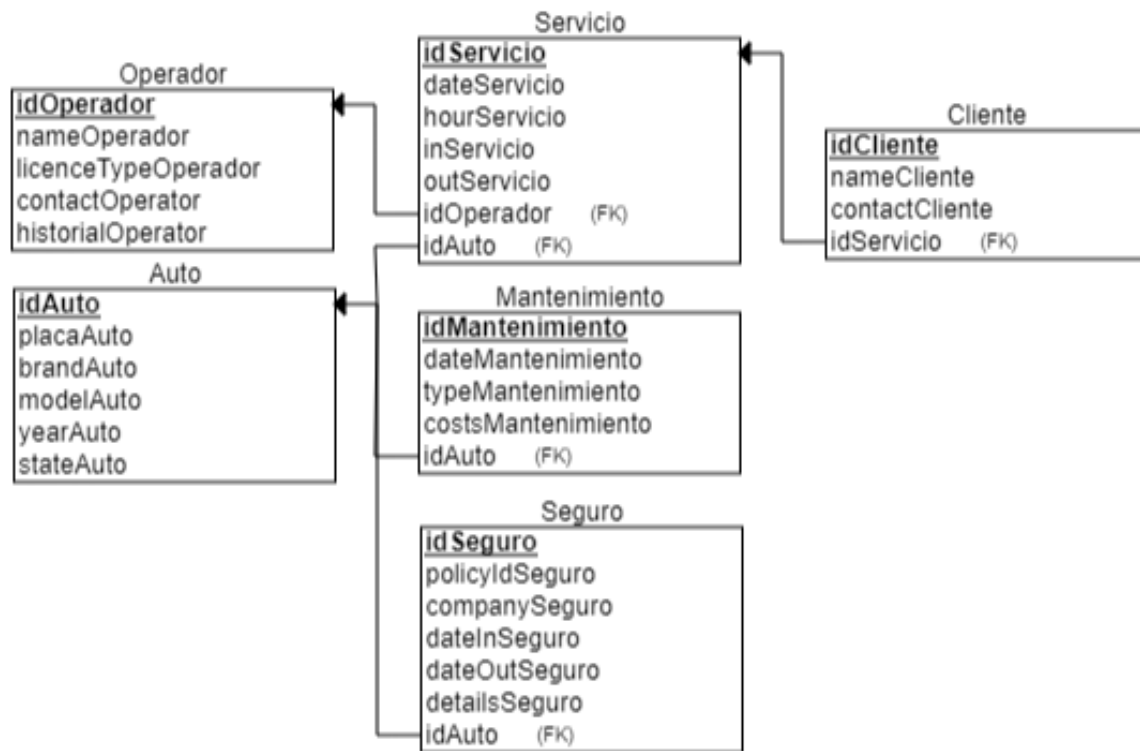


Figure 2: Modelo relacional.

## Modelo relacional

En la Figura 2 se presenta la propuesta de Modelo Entidad - Relación para. el caso de la flota de autos, con las tablas **Operador**, **Auto**, **Servicio**, **Mantenimiento**, **Seguro** y **Cliente** con sus respectivas relaciones.

## Sentencias SQL

En el siguiente apartado se presentara el codigo en formato SQL para crear la base de datos, crear las tablas con sus respectivas llaves principales y llaves foraneas, y la poblacion de la misma con tuplas individuales con datos genericos.

Listing 1: Base de datos en SQL "Flotilla de Autos".

```
CREATE DATABASE flotillaAutos;
```

— *Creacion de las tablas*

```
CREATE TABLE Operador  
(idOperador INT PRIMARY KEY, nameOperador CHAR(100),  
licenseTypeOperador CHAR(5), contactOperador CHAR(50),  
historialOperador CHAR(50));
```

```
CREATE TABLE Auto  
(idAuto INT PRIMARY KEY, placaAuto CHAR(15),  
brandAuto CHAR(15), modelAuto CHAR(15),  
yearAuto INT, stateAuto CHAR(15));
```

```
CREATE TABLE Servicio  
(idServicio INT PRIMARY KEY,  
dateServicio CHAR(15), hourServicio CHAR(15),  
inServicio CHAR(15), outServicio CHAR(15),  
idOperador INT, idAuto INT, FOREIGN KEY (idAuto)  
REFERENCES Auto(idAuto), FOREIGN KEY (idOperador)  
REFERENCES Operador(idOperador));
```

```
CREATE TABLE Mantenimiento  
(idMant INT PRIMARY KEY, dateMant CHAR(15),  
typeMant CHAR(15), costsMant INT, idAuto INT,  
FOREIGN KEY (idAuto) REFERENCES Auto(idAuto));
```

```
CREATE TABLE Seguro  
idSeguro INT PRIMARY KEY, policyIdSeguro char(15),  
companySeguro char(15), dateInSeguro char(15),  
dateOutSeguro char(15), detailsSeguro char(15),  
idAuto INT, FOREIGN KEY (idAuto) REFERENCES Auto(idAuto));
```

```
CREATE TABLE Cliente  
(idCliente INT PRIMARY KEY, nameCliente CHAR(30),  
contactCliente CHAR(30), idServicio INT,  
FOREIGN KEY (idservicio) REFERENCES Servicio(idServicio));
```

— *Poblacion de Tuplas*

```
INSERT INTO Operador VALUES  
(1, 'Cesar-Martinez', 'A', '7713320132'),  
(2, 'Pedro-Martinez', 'B', '7713520962', 'Bueno'),  
(3, 'Juan-Perez', 'A', '77668435962', 'Malo');
```

```
INSERT INTO Auto VALUES  
(1, '45f-d85', 'Ford', 'Expedition', '2024', 'Excelente'),  
(2, '1d3-1ff', 'Ibiza', 'Seat', '2020', 'Regular'),
```

```
(3, 'r4g-4vf', 'Tesla', 'CYBERTRUCK', '2023', 'Excelente');
```

**INSERT INTO Servicio VALUES**

```
(1, '17/08/24', '22:53', 'Pachuca', 'Zempoala', 1, 3),  
(2, '22/08/24', '17:00', 'Pachuca', 'Real-Del-Monte', 2, 1),  
(3, '20/08/24', '14:00', 'Zempoala', 'Ecatepec', 1, 3);
```

**INSERT INTO Mantenimiento Values**

```
(1, '14/08/24', 'Enllantado', '5000', 2),  
(2, '15/08/24', 'Cambio-Aceite', '2000', 1),  
(3, '14/08/24', 'Amortiguadores', '3000', 2);
```

**INSERT INTO Seguro Values**

```
(1, '1AS4D50', 'BBVA-Seguros', '14/08/23', '14/08/25', 'Total', 3),  
(2, 'SA1F5SS', 'AXA', '14/08/23', '14/08/25', 'Total', 1),  
(3, 'SA1F5SS', 'Mapfre', '14/08/23', '14/08/25', 'Parcial', 2);
```

**INSERT INTO Cliente Values**

```
(1, 'Melisa-Mendoza', '5577448890', 3),  
(2, 'Juan-Perez', '5564813291', 1),  
(3, 'Cesar-Martinez', '7713320132', 2);
```

*— Consultas de Tablas*

**SELECT \* FROM Auto;**

**SELECT \* FROM Cliente;**

**SELECT \* FROM Mantenimiento;**

**SELECT \* FROM Operador;**

**SELECT \* FROM Seguro;**

**SELECT \* FROM Servicio;**



## 5. Conclusiones

En conclusión, esta práctica ha sido sumamente enriquecedora. Aunque inicialmente no recordaba algunas líneas de código SQL, esta experiencia me permitió refrescar mis conocimientos sobre las propiedades de los modelos relacionales y el modelo entidad-relación (E/R). Además, he comprobado que el lenguaje SQL es bastante accesible una vez que se comprende su lógica. Lo que más me agradó y entendí en esta práctica fue el uso de las llaves foráneas y su importancia en las consultas tanto horizontales como verticales. Asimismo, el modelo relacional me proporcionó una base sólida para iniciar este proyecto, ya que partir de este concepto es fundamental para la correcta generación de una base de datos.

## References

- [1] info@erdplus.com. (s. f.). ERDPlus. Mosor, Inc. <https://erdplus.com/>
- [2] *W3Schools.com*. (s. f.). [https://www.w3schools.com/sql/sql\\_intro.asp](https://www.w3schools.com/sql/sql_intro.asp)
- [3] Guasch, C. G. (2023, 12 julio). What is ER Modeling? A Comprehensive Guide With Practical Examples - SQL Knowledge Center. *SQL Easy Tutorial*. <https://www.sql-easy.com/learn/what-is-er-modeling/>
- [4] Orakzai, R. U., & Orakzai, R. U. (2023, 16 mayo). *What Is an Entity Relationship Diagram? — Baeldung on Computer Science*. Baeldung On Computer Science. <https://www.baeldung.com/cs/erd>
- [5] colaboradores de Wikipedia. (2024, July 14). *Edgar Frank Codd*. Wikipedia, La Enciclopedia Libre. [https://es.wikipedia.org/wiki/Edgar\\_Frank\\_Codd](https://es.wikipedia.org/wiki/Edgar_Frank_Codd)
- [6] *SQL OnLine IDE*. (n.d.). <https://sqliteonline.com/>