

# REPORTE DE PRÁCTICA NO. 1

Gestión de flotilla de autos

ALUMNO: José Ángel Castro Paredes  
Dr. Eduardo Cornejo-Velázquez



## 1. Introducción

Una flotilla de autos es un conjunto de automóviles con los que dispone una empresa, pueden ser de muchos tipos diferentes, además de contar con otras características las cuales es importante tener el control de cada una de ellas para que así el manejo de nuestra flotilla sea más eficiente

## 2. Marco teórico

### Análisis de requerimientos

Es el conjunto de técnicas y procedimientos que nos permiten conocer los elementos necesarios para definir un proyecto de software, permite especificar las características operacionales del software, indicar la interfaz del software con otros elementos del sistema y establecer las restricciones que debe cumplir el software. Fuentes, M. D. (2011). MATERIAL DIDÁCTICO NOTAS DEL CURSO ANÁLISIS DE REQUERIMIENTOS. [?, ?]

### Modelo Entidad - Relación

Es un modelo logico de datos que está basado en una percepción de un mundo real que consiste en una colección de objetos básicos, denominados entidades, y de relaciones entre estos objetos.(De Lourdes, R. A. M. Modelo Entidad-Relación.) [?, ?] para incorporarla al documento.

### Modelo relacional

El modelo relacional es un modelo de datos que debe permitir representar la información que nos interesa del mundo real, manipula informacion mediante las operaciones de actualización y consulta de los datos, ademas establece reglas de integridad; es decir, condiciones que los datos deben cumplir. (Costa, D. C. (2002). El modelo relacional y el álgebra relacional. UOC, la universidad virtual.)[?, ?, ?]

### SQL

El SQL es el lenguaje estándar ANSI/ISO de definición, manipulación y control de bases de datos relacionales. Es un lenguaje declarativo: sólo hay que indicar qué se quiere hacer. [?, ?, ?] (Escofet, C. M. (2002). El lenguaje SQL. UOC, la universidad virtual.)

### **3. Herramientas empleadas**

Las herramientas que se utilizaron para la realización del proyecto fueron las siguientes:

1. ERD Plus, como apoyo para la realización de los esquemas relacionales.
2. MySQL workbench, como apoyo para la realización de la base de datos.

## 4. Desarrollo

### Análisis de requisitos

Para la realización de la base de datos lo primordial es analizar los datos propuestos para que a partir de los requisitos recopilados, se deben identificar las entidades principales que van a componer la base de datos. A partir de las entidades se definen los atributos de las entidades y después identificar las relaciones entre las entidades. Una vez contando con estos datos se puede realizar los esquemas que nos ayudan a la programación de la base de datos con ayuda de sentencias en lenguaje sql.

Los requisitos funcionales del proyecto es:

1. Registro de vehiculos.
2. Gestión de manetimiento.
3. Reporte de costos
4. Gestion de rutas

Los requisitos no funcionales se podrian considerar como:

1. Seguridad
2. Escalabilidad
3. Tiempo de respuesta
4. Control

### Modelo Entidad - Relación

En la Tabla 1 se presenta la propuesta de Modelo Entidad - Relación para.....

Table 1: Matriz de realaciones.

Entidades	1	2	3	4	5	6	7	8	9	10
1 vehículo	O	X	X	X	O	O	O	O	O	O
2 documentación	X	O	O	O	O	O	O	O	O	O
3 conductor	X	O	O	O	O	O	O	O	O	O
4 tanque	X	O	O	O	X	X	O	O	O	O
5 detallesCompraGasolina	O	O	O	X	X	X	O	O	O	O
6 compraGasolina	O	O	O	O	O	X	O	O	O	O
7 reporteFallas	X	O	X	O	O	O	O	X	O	O
8 solicitudRefacción	O	O	O	O	O	O	O	X	X	X
9 detallesCompraRefacciones	O	O	O	O	O	O	O	X	O	X
10 distribuidorRefacciones	O	O	O	O	O	O	O	O	X	O

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

### Modelo relacional

En la Figura 2 se presenta la propuesta de Modelo Entidad - Relación para. el caso.....

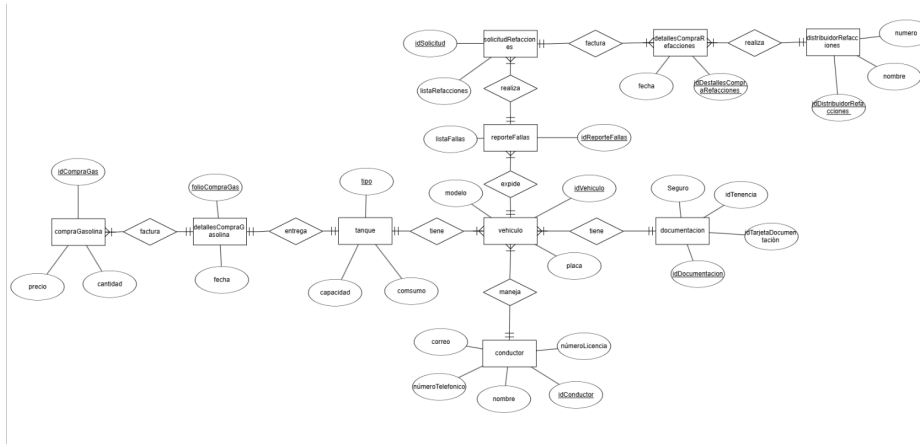


Figure 1: Modelo Entidad - Relación propuesto.

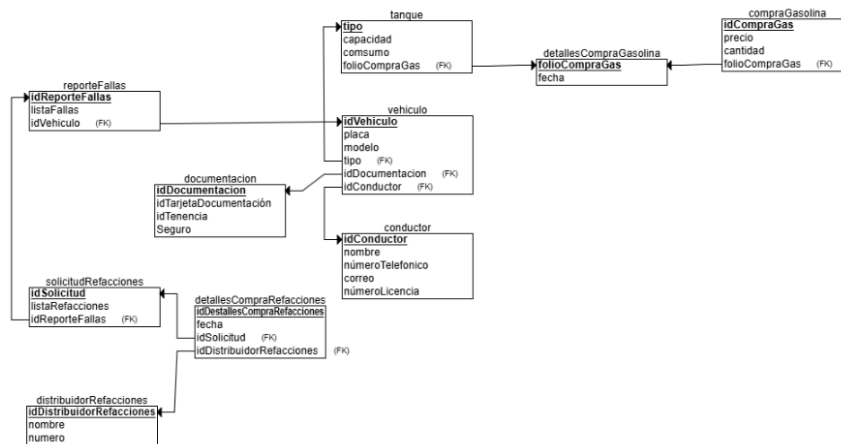


Figure 2: Modelo Relacional propuesto.

## Sentencias SQL

Las sentencias utilizadas en sql que se utilizaron fueron:

Listing 1: Crear base de datos competencia.

```
CREATE TABLE conductor
(
  idConductor INT NOT NULL,
  nombre VARCHAR(80) NOT NULL,
  numeroTelefono VARCHAR(10) NOT NULL,
  correo VARCHAR(80) NOT NULL,
  numeroLicencia INT NOT NULL,
  PRIMARY KEY (idConductor)
);

CREATE TABLE documentacion
(
```

```

    idDocumentacion INT NOT NULL,
    idTarjetaDocumentacion INT NOT NULL,
    idTenencia INT NOT NULL,
    Seguro INT NOT NULL,
    PRIMARY KEY (idDocumentacion)
);

```

```

CREATE TABLE distribuidorRefacciones
(
    idDistribuidorRefacciones INT NOT NULL,
    nombre VARCHAR(50) NOT NULL,
    numero INT NOT NULL,
    PRIMARY KEY (idDistribuidorRefacciones)
);

```

```

CREATE TABLE detallesCompraGasolina
(
    folioCompraGas INT NOT NULL,
    fecha DATE NOT NULL,
    PRIMARY KEY (folioCompraGas)
);

```

```

CREATE TABLE tanque
(
    tipo VARCHAR(30) NOT NULL,
    capacidad INT NOT NULL,
    consumo INT NOT NULL,
    folioCompraGas INT NOT NULL,
    PRIMARY KEY (tipo),
    FOREIGN KEY (folioCompraGas) REFERENCES detallesCompraGasolina(folioCompraGas)
);

```

```

CREATE TABLE compraGasolina
(
    idCompraGas INT NOT NULL,
    precio FLOAT NOT NULL,
    cantidad FLOAT NOT NULL,
    folioCompraGas INT NOT NULL,
    PRIMARY KEY (idCompraGas),
    FOREIGN KEY (folioCompraGas) REFERENCES detallesCompraGasolina(folioCompraGas)
);

```

```

CREATE TABLE vehiculo
(
    idVehiculo INT NOT NULL,
    placa VARCHAR(10) NOT NULL,
    modelo VARCHAR(30) NOT NULL,
    tipo VARCHAR(30) NOT NULL,
    idDocumentacion INT NOT NULL,
    idConductor INT NOT NULL,
    PRIMARY KEY (idVehiculo),
    FOREIGN KEY (tipo) REFERENCES tanque(tipo),
    FOREIGN KEY (idDocumentacion) REFERENCES documentacion(idDocumentacion),
    FOREIGN KEY (idConductor) REFERENCES conductor(idConductor)
);

```

);

**CREATE TABLE** reporteFallas

(  
    idReporteFallas **INT NOT NULL**,  
    listaFallas **VARCHAR**(100) **NOT NULL**,  
    idVehiculo **INT NOT NULL**,  
    **PRIMARY KEY** (idReporteFallas),  
    **FOREIGN KEY** (idVehiculo) REFERENCES vehiculo(idVehiculo)  
);

**CREATE TABLE** solicitudRefacciones

(  
    idSolicitud **INT NOT NULL**,  
    listaRefacciones **VARCHAR**(100) **NOT NULL**,  
    idReporteFallas **INT NOT NULL**,  
    **PRIMARY KEY** (idSolicitud),  
    **FOREIGN KEY** (idReporteFallas) REFERENCES reporteFallas(idReporteFallas)  
);

**CREATE TABLE** detallesCompraRefacciones

(  
    idDestallesCompraRefacciones **INT NOT NULL**,  
    fecha **DATE NOT NULL**,  
    idSolicitud **INT NOT NULL**,  
    idDistribuidorRefacciones **INT NOT NULL**,  
    **PRIMARY KEY** (idDestallesCompraRefacciones),  
    **FOREIGN KEY** (idSolicitud) REFERENCES solicitudRefacciones(idSolicitud),  
    **FOREIGN KEY** (idDistribuidorRefacciones) REFERENCES distribuidorRefacciones(idDistribuidor)  
);



## 5. Conclusiones

Para finalizar, la implementación de esta base de datos para la gestión de una flotilla de vehículos ha demostrado ser una herramienta muy útil para organizar y mejorar los procesos dentro de la empresa. Al estructurar de manera adecuada tablas que contienen información clave sobre los conductores, los vehículos, la documentación, las compras de gasolina, las refacciones y los reportes de fallas, se facilita el acceso a los datos de forma rápida y clara. Esto ayuda a optimizar el trabajo diario y a tomar decisiones más informadas.

## Referencias Bibliográficas

## References

- [1] Fuentes, M. D. (2011). MATERIAL DIDÁCTICO NOTAS DEL CURSO ANÁLISIS DE REQUERIMIENTOS. *7th ed McGraw-Hill*
- [2] Costa, D. C. (2002). El modelo relacional y el álgebra relacional. UOC, la universidad virtual. *7th ed Pearson*
- [3] Date, C. J. (2004). An Introduction to Database Systems *8th ed Addison-Wesley*
- [4] De Lourdes, R. A. M. Modelo Entidad-Relación. *11th ed Pearson*