



Nombre: Hernandez Sanchez  
Juan German

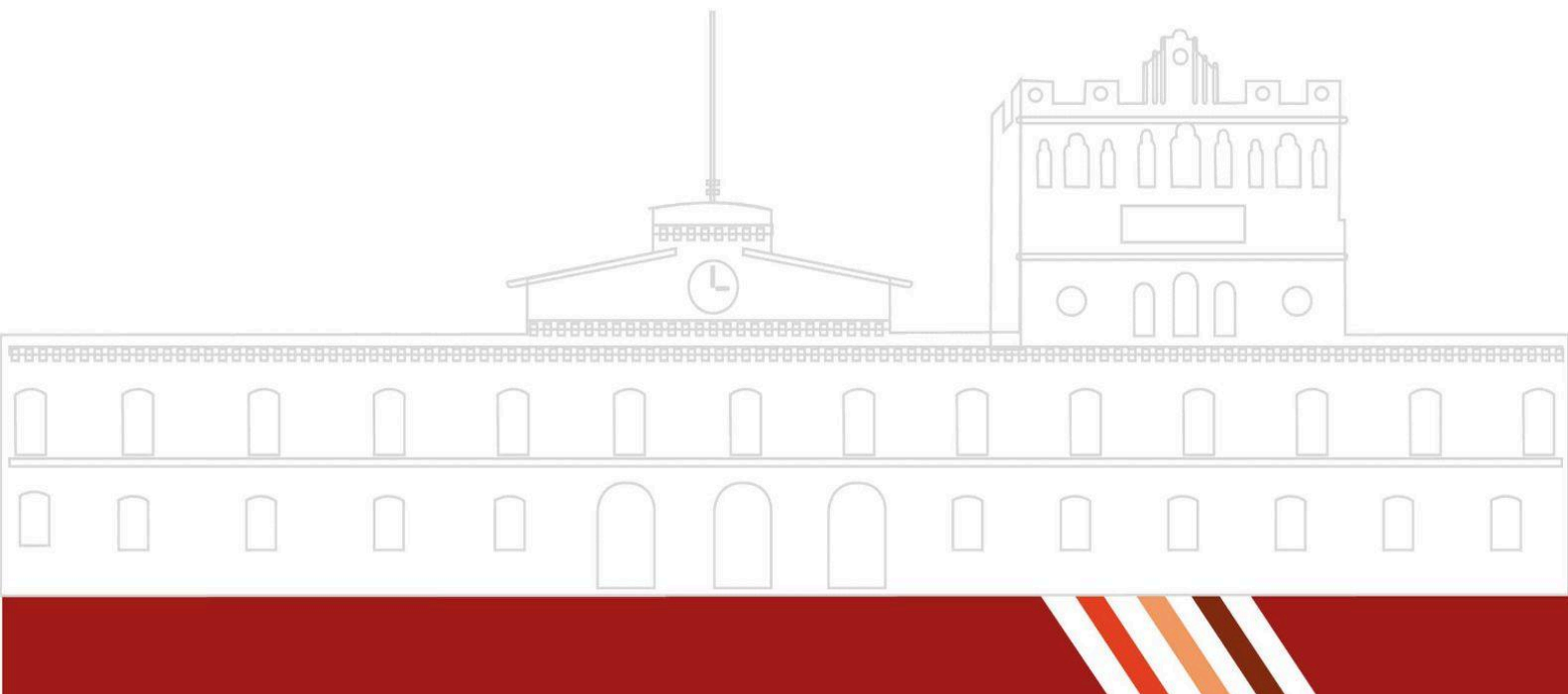
Materia: Base de datos  
distribuidas

No. De Cuenta: 472549

Catedrático: Eduardo Cornejo  
Velazquez

Carrera: Licenciatura en Ciencias  
Computacionales

Semestre: Sexto Grupo: "2"



## Introducción

El proceso sistemático y metodológico en el modelado de bases de datos relacionales es crucial por varias razones:

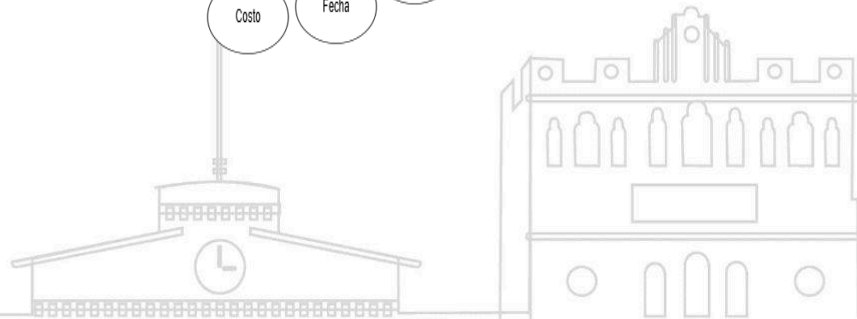
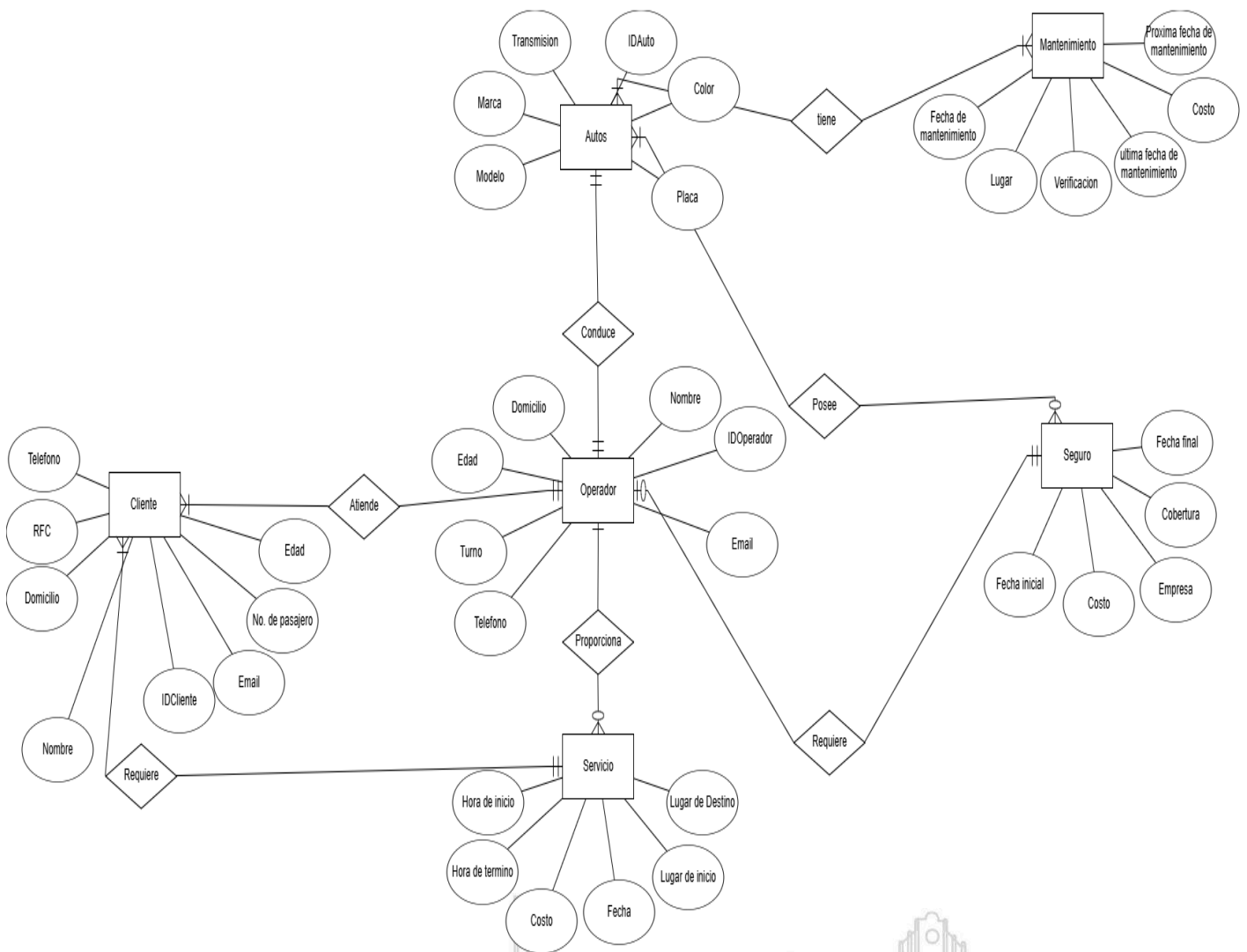
1. Coherencia y precisión: Un enfoque metodológico asegura que los datos se estructuren de manera coherente y precisa, lo que facilita la integridad y la consistencia de la información.
2. Reducción de redundancia: Al seguir un proceso sistemático, se minimizan los datos redundantes, lo que optimiza el almacenamiento y mejora el rendimiento de la base de datos.
3. Mejora de la comunicación: Un modelo bien definido facilita la comunicación entre los diferentes actores del proyecto, como desarrolladores, analistas y usuarios finales. Esto asegura que todos tengan una comprensión clara de la estructura y el propósito de la base de datos.
4. Eficiencia en consultas: Un diseño adecuado permite ejecutar consultas de manera más eficiente, lo que mejora el rendimiento general del sistema.
5. Calidad del sistema: Un proceso metodológico reduce la posibilidad de errores y aumenta la calidad del sistema al proporcionar una base sólida para el desarrollo y mantenimiento de la base de datos.
6. Facilidad de mantenimiento: Un modelo bien estructurado facilita el mantenimiento y la actualización de la base de datos, lo que es esencial para adaptarse a los cambios en los requisitos del negocio.

En resumen, el proceso sistemático y metodológico en el modelado de bases de datos relacionales es fundamental para garantizar la eficiencia, precisión y calidad del sistema, además de facilitar la comunicación y el mantenimiento.

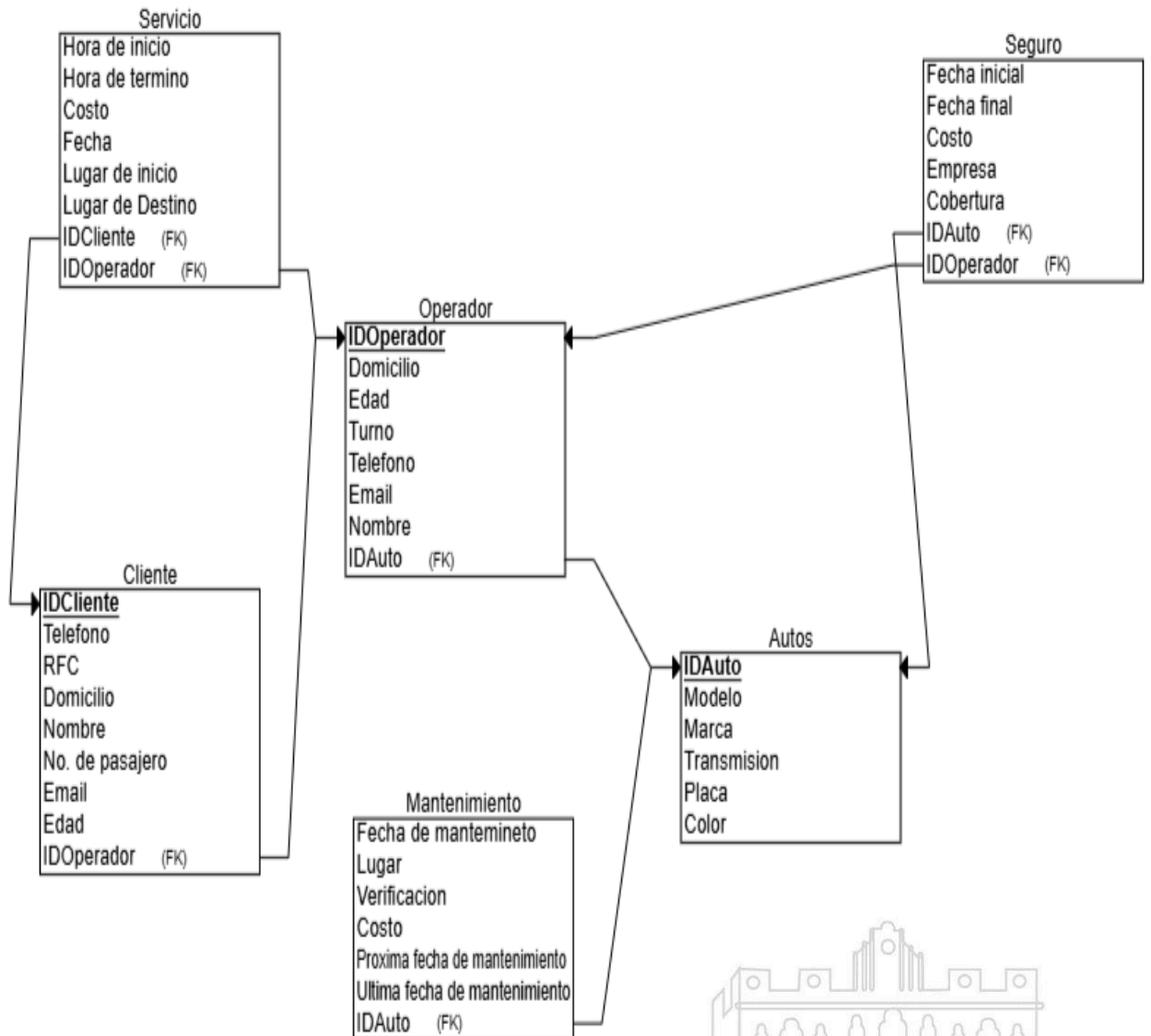
## Marco teórico



## Modelo Entidad-Relación



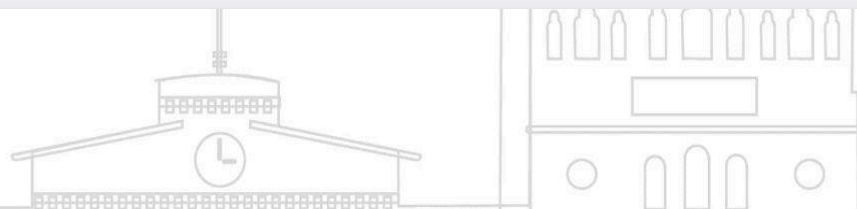
## Modelo Relacional



## Sentencias SQL

```
FlotilladeTaxiUber x
Limit to 1000 rows
1 • create database uber_taxi;
2
3 • use uber_taxi;
4
5 • create table Auto (
6     idAuto INT auto_increment primary key,
7     Marca varchar(45) not null,
8     Placa varchar(45) not null,
9     Color varchar(45) not null,
10    Transmission varchar(45) not null,
11    Modelo varchar(45) not null
12 );
13
14 • insert into Auto (idAuto, Marca, Placa, Color, Modelo,Transmission)values
15 (1, 'camaro', 'F5G9', 'verde','1993','Estandar'),
16 (2, 'nissan', 'C7T9', 'rosa','2003','Automatico'),
17 (3, 'ferrari', 'K9R6', 'morado','1980','Estandar');
18
19 • create table Operador (
```

```
FlotilladeTaxiUber x
Limit to 1000 rows
19 • create table Operador (
20     idOperador INT auto_increment primary key,
21     Nombre varchar(30) not null,
22     Edad varchar(30) not null,
23     Domicilio varchar(30) not null,
24     Telefono varchar(30) not null,
25     Email varchar(30) not null,
26     Turno varchar(30) not null,
27     idAuto int not null,
28     FOREIGN KEY (idAuto) REFERENCES Auto(idAuto)
29 );
30
31 • insert into Operador (idOperador, Nombre, Edad, Domicilio, Telefono, Email, Turno, idAuto)values
32 (12, 'gerardo', '18','soriana del valle','7713654567','gerardo@uber.com', 'nocturno', '3111'),
33 (23, 'aimar', '47','tulipanes','771234589','aimar2@taxi.com', 'nocturno', '6222'),
34 (34, 'david', '75','centro','7724680641','david7@uber.com', 'matutino', '9333');
35
36 • select *from Operador;
37
```



```
FlotilladeTaxiUber
Limit to 1000 rows

39 • create table Cliente (
40     idCliente INT auto_increment primary key,
41     Nombre varchar(45) not null,
42     Domicilio varchar(45) not null,
43     Telefono varchar(45) not null,
44     Email varchar(45) not null,
45     no_de_pasajero varchar(45) not null,
46     RFC varchar(45) not null,
47     Edad int(10) not null,
48     idOperador int not null,
49     FOREIGN KEY (idOperador) REFERENCES Operador(idOperador)
50 );
51
52
53 • insert into Cliente (idCliente, Nombre, Domicilio, Telefono, Email, no_de_pasajero, RFC, edad,idOperador)values
54 (198, 'Rufino','camelia','5544668849','Rufino555@gmail.com', '3', '4892',19,12),
55 (297, 'Lulu','rojo gomez','7235678912','Lulu478@gmail.com', '2', '8903',22,23),
56 (396, 'Bertha','hacienda margarita','7711357830','Bertha909@.com', '2', '1892',38,34);
57
```

```
FlotilladeTaxiUber
Limit to 1000 rows

58 • select *from cliente;
59
60 • create table Mantenimiento (
61     ultima_fecha_de_mantenimiento varchar(45) not null,
62     verificacion varchar(45) not null,
63     costo varchar(45) not null,
64     fecha_de_mantenimiento varchar(45) not null,
65     proxima_fecha_de_mantenimiento varchar(45) not null,
66     lugar varchar(45) not null,
67     idAuto int not null,
68     FOREIGN KEY (idAuto) REFERENCES Auto(idAuto)
69 );
70
71 • insert into Mantenimiento (ultima_fecha_de_mantenimiento, verificacion, costo, fecha_de_mantenimiento, proxima_fecha_de_mantenimie
72 ('04-07-23','llantas', '300', '26-04-24', '12-03-25','torreon', 111),
73 ('11-15-22','aire acondicionado', '1200', '17-01-24', '19-09-25','guadalajara', 222),
74 ('01-02-20','motor', '15000', '31-08-24', '25-10-25','cancun', 333);
75
76 • select*from Mantenimiento;
77
```

```
FlotilladeTaxiUber
Limit to 1000 rows

77 • create table Servicio (
78     lugar_de_inicio varchar(45) not null,
79     lugar_destino varchar(45) not null,
80     hora_de_inicio varchar(45) not null,
81     Hora_de_termino varchar(45) not null,
82     costo varchar(45) not null,
83     fecha varchar(45) not null,
84     idAuto int not null,
85     idOperador int not null,
86     FOREIGN KEY (idAuto) REFERENCES Auto(idAuto),
87     FOREIGN KEY (idOperador) REFERENCES Operador(idOperador)
88 );
89
90 • insert into Servicio (lugar_de_inicio, lugar_destino, hora_de_inicio, Hora_de_termino , costo, fecha, idAuto, idOperador)values
91 ('reloj monumental', 'cenhies', '07:58', '09:00','80', '24-05-21', 111,12),
92 ('tuzos', 'galerias', '02:25', '03:59','70', '26-05-24', 222,23),
93 ('mercado 1 de mayo', 'fraccionamiento colosio', '05:00', '05:40','60', '13-10-26', 333,34);
94
95 • select*from Servicio;
```

```
FlotilladeTaxiUber
Limit to 1000 rows
96
97 • create table Seguro (
98     fecha_inicial varchar(45) not null,
99     fecha_final varchar(45) not null,
100     costo varchar(45) not null,
101     empresa varchar(45) not null,
102     cobertura varchar(45) not null,
103     idAuto int not null,
104     idOperador int not null,
105     FOREIGN KEY (idAuto) REFERENCES Auto(idAuto),
106     FOREIGN KEY (idOperador) REFERENCES Operador(idOperador)
107 );
108
109 • insert into Seguro (fecha_inicial, fecha_final , costo, empresa, cobertura, idAuto, idOperador) values
110 ('05-01-22', '05-01-27', '3500', 'FIDUM', 'Asistencia vial', 111, 12),
111 ('29-08-23', '29-08-28', '3800', 'BBVA', 'Robo total', 222, 23),
112 ('15-07-24', '15-07-29', '4200', 'INBURSA', 'Gastos medicos', 333, 34 );
113
114 • select*from Seguro;
```

## Herramientas empleadas

Para la realización de este proyecto se implementaran como herramientas de trabajo LáTeX con el compilador TeXstudio para la creación del presente documento, igualmente ERDPlus el cual nos permite la creación de los diagramas de Modelo Entidad-Relación y del modelo relacional para comprender el funcionamiento del sistema y por ultimo My SQL para la creación y administración remota de la base de datos, todo esto nos fue muy útil para poder realizar este trabajo.

## Desarrollo

### Listado de requerimientos y funciones de datos

Auto: Marca, Placa, Color, IdAuto, modelo, IdOperador

Operador: IdOperador, nombre, edad, domicilio, teléfono, IdAuto, Correo, Teléfono

Servicio: Lugar inició, lugar, destino, costo, fecha, hora inicial, hora final

Cliente: idCliente, nombre, correo, telefono, No. de pasajeros, Domicilio, RFC

Mantenimiento: verificación, fecha actual, fecha próxima, costo, lugar

Seguro: Costo, duracion, empresa, poliza, ficha inicio, fecha final





## Matriz de relaciones

	Auto	Operador	Servicio	Cliente	Mantenimiento	Seguro	
Auto		X	O	X	X	O	O
Operador		O	X	O	O	X	O
Servicio		X	O	X	O	X	X
Cliente		X	O	O	X	X	X
Mantenimiento		O	X	X	X	X	X
Seguro		O	O	X	X	X	X

## Conclusiones

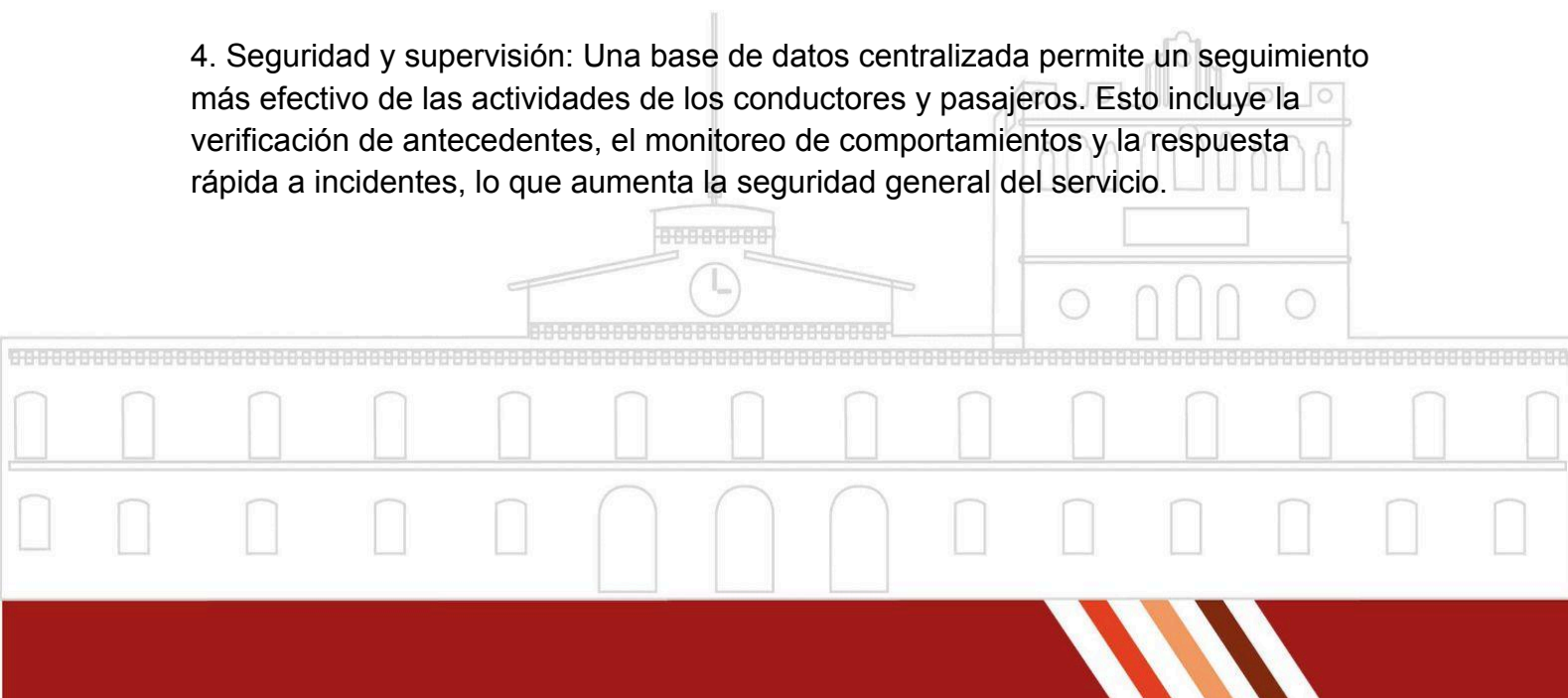
La creación e implementación de esta base de datos aborda muchos puntos y dará muchas ventajas como las siguientes:

**Mejora de la eficiencia operativa:** Una base de datos bien estructurada permite gestionar grandes volúmenes de datos de manera eficiente. Esto incluye información sobre pasajeros, conductores, rutas y tiempos de viaje, lo que facilita la toma de decisiones rápidas y precisas.

**2. Optimización de rutas y tiempos de espera:** Con datos en tiempo real, el sistema puede identificar áreas de alta demanda y ajustar la disponibilidad de conductores en consecuencia. Esto reduce los tiempos de espera para los pasajeros y optimiza las rutas para los conductores.

**3. Personalización de la experiencia del usuario:** Al analizar los patrones de viaje de los usuarios, el sistema puede ofrecer recomendaciones personalizadas, como destinos frecuentes y opciones de viaje preferidas. Esto mejora la satisfacción del cliente y fomenta la lealtad.

**4. Seguridad y supervisión:** Una base de datos centralizada permite un seguimiento más efectivo de las actividades de los conductores y pasajeros. Esto incluye la verificación de antecedentes, el monitoreo de comportamientos y la respuesta rápida a incidentes, lo que aumenta la seguridad general del servicio.







## Referencias

<https://ahorraseguros.mx/inbursa>. (s. f.). ▷ Inbursa Seguro de Auto - ¡Cotiza en AhorraSeguros.mx! <https://ahorraseguros.mx/inbursa/>

BBVA. (2024). BBVA MÉXICO.

<https://www.bbva.mx/educacion-financiera/seguros/seguro-auto-tipos-de-seguros.html>

