



# ViewCarTransit

Diseño y administración de bases de datos

Difonti Domé, Ramiro - [alu0101425030@ull.edu.es](mailto:alu0101425030@ull.edu.es)  
García Martín, Ruymán - [alu0101408866@ull.edu.es](mailto:alu0101408866@ull.edu.es)  
Medinilla Hernández, Marcos - [alu0101211206@ull.edu.es](mailto:alu0101211206@ull.edu.es)



1. Introducción	2
2. Objetivos del proyecto	2
3. Contexto de la base de datos	2
3.1 Entidades	2
3.2 Relaciones	2
3.3 Consideraciones adicionales	2
4. Scripts SQL	2
4.1 Creación de Base de Datos	2
4.2 Datos de la Base de Datos	2
4.3 Triggers	2
4.4 Checkers	2
5. Ejemplos de consulta	2
6. Repositorio GitHub	2
7. Bibliografía	2



# 1. Introducción

Este proyecto tiene como objetivo desarrollar una base de datos para gestionar la logística de reparto para una empresa (ViewCarTransit) que ofrece servicios de envío de paquetería a clientes de empresas asociadas. La gestión de logística es una parte fundamental para garantizar el éxito de las operaciones de envío, desde la recogida de paquetes hasta la entrega final. En este contexto, la base de datos proporciona una plataforma estructurada para organizar y almacenar información sobre los recursos y procesos involucrados (vehículos, paquetes...).

La base de datos está diseñada para ser escalable, permitiendo a la empresa manejar un gran volumen de datos relacionado con sedes, empleados, empresas asociadas, contratos, vehículos, talleres, paquetes y conductores. Además, incorpora controles de integridad y automatizaciones mediante triggers para garantizar que los datos se gestionan de manera eficiente y con alta calidad. Este sistema no solo simplifica la operación diaria, sino que también permite realizar análisis sobre el rendimiento y la efectividad de los procesos logísticos.

## 2. Objetivos y requisitos

### 2.1 Objetivos del proyecto

<b>OBJ - 01</b>	<b>Sistema estructurado</b>
<b>Descripción</b>	Crear un sistema estructurado para registrar información clave relacionada con la logística.

<b>OBJ - 02</b>	<b>Relaciones coherentes</b>
<b>Descripción</b>	Establecer relaciones entre entidades para representar adecuadamente el flujo de trabajo

<b>OBJ - 03</b>	<b>Consultas eficientes</b>
<b>Descripción</b>	Facilitar consultas eficientes y consistentes para la gestión operativa.



<b>OBJ - 04</b>	<b>Generación de informes</b>
<b>Descripción</b>	Crear un módulo que permita generar informes detallados.

<b>OBJ - 05</b>	<b>Integración con sistemas de gestión externos</b>
<b>Descripción</b>	Desarrollar un sistema donde el usuario pueda realizar un seguimiento de su paquete.

## 2.2 Requisitos funcionales del proyecto

### Gestión de Vehículos:

- Registrar nuevos vehículos con datos como matrícula, modelo, color, estado, y su asociación con sede o taller.
- Consultar la lista de vehículos existentes en el sistema.
- Actualizar información de un vehículo específico.
- Eliminar registros de vehículos cuando ya no sean necesarios.

### Gestión de Empresas:

- Permitir el registro de empresas con nombre, dirección y teléfono.
- Visualizar un listado de empresas registradas.
- Actualizar los datos de una empresa existente.
- Eliminar empresas del sistema.

### Gestión de Conductores

- Permite agregar nuevos conductores con dni, nombre apellidos y licencia de conducir.
- Los vehículos asociados a esos conductores.

### Gestión de Paquetes:

- Registrar paquetes con información como descripción, peso, destinatario y estado.
- Consultar y listar los paquetes registrados.
- Eliminar paquetes que ya no vayan a ser enviados.



## 3. Contexto de la base de datos

### 3.1 Entidades

Las principales entidades del modelo son:

- **SEDE:** Representa las oficinas o sedes donde opera la empresa.
  - **Atributos:**
    - **id\_sede** (clave primaria): Identificador único de la sede.
    - **nombre:** Nombre de la sede.
    - **localidad:** Localidad donde se encuentra la sede.
    - **calle:** Calle donde está ubicada la sede.
    - **numero:** Número de la dirección.
    - **telefono:** Teléfono de contacto, con formato validado.
    - **correo\_contacto:** Correo electrónico de contacto, validado.
- **ENCARGADO:** El encargado de cada sede.
  - **Atributos:**
    - **dni** (clave primaria): Identificador del encargado.
    - **nombre:** Nombre del encargado.
    - **apellidos:** Apellidos del encargado.
    - **id\_sede** (clave foránea): Relación única con la sede correspondiente.
- **EMPRESA:** Empresas clientes que requieren los servicios logísticos.
  - **Atributos:**
    - **id\_empresa** (clave primaria): Identificador único de la empresa.
    - **nombre:** Nombre de la empresa.
    - **tipo\_empresa:** Tipo de negocio (e.g., minorista, mayorista).
    - **telefono:** Teléfono de contacto.
    - **correo\_contacto:** Correo electrónico de la empresa.
- **CONTRATO:** Contratos entre la empresa logística y sus clientes.
  - **Atributos:**
    - **id\_contrato** (clave primaria): Identificador único del contrato.



- **fecha\_ini:** Fecha de inicio del contrato.
- **fecha\_fin:** Fecha de finalización del contrato.
- **VEHICULO:** Incluye detalles de vehículos usados para transporte.
  - **Atributos:**
    - **matricula** (clave primaria): Identificador del vehículo.
    - **modelo:** Modelo del vehículo.
    - **estado:** Estado operativo del vehículo (e.g., "Apto", "No apto").
    - **color:** Color del vehículo.
    - **tipo:** Tipo de vehículo (e.g., furgoneta, camión).
- **TALLER:** Talleres asociados para mantenimiento de vehículos.
  - **Atributos:**
    - **id\_taller** (clave primaria): Identificador único del taller.
    - **nombre:** Nombre del taller.
    - **telefono:** Teléfono de contacto.
    - **direccion:** Dirección del taller (localidad, calle, número).
- **INFORME:** Informes generados tras revisiones en talleres.
  - **Atributos:**
    - **id\_informe** (clave primaria): Identificador único del informe.
    - **descripcion:** Descripción del trabajo realizado.
    - **fecha:** Fecha del informe.
    - **nombre\_mecanico:** Nombre del mecánico responsable.
    - **apellidos:** Apellidos del mecánico.
- **CONDUCTOR:** Conductores asignados a los vehículos.
  - **Atributos:**
    - **dni** (clave primaria): Identificador del conductor.
    - **nombre:** Nombre del conductor.
    - **apellidos:** Apellidos del conductor.
    - **licencia:** Tipo de licencia del conductor.
- **PAQUETE:** Paquetes enviados por las empresas clientes.
  - **Atributos:**
    - **id\_paquete** (clave primaria): Identificador único del paquete.
    - **descripcion:** Descripción del contenido del paquete.
    - **peso:** Peso del paquete (validado para ser mayor a cero).



- **destino:** Dirección de destino del paquete.
- **fecha:** Fecha del envío.
- **TEST:** Registros de pruebas de los vehículos.
  - **Atributos:**
    - **id\_test** (clave primaria): Identificador de la prueba.
    - **nota:** Resultado de la prueba.
    - **matricula** (clave foránea): Relación con el vehículo probado.

**Entidades fuertes:** SEDE, CONDUCTOR, VEHICULO, TEST, TALLER, INFORME, EMPRESA, CONTRATO.

**Entidades débiles:** ENCARGADO, PAQUETE.

## 3.2 Relaciones

1. **Administra:** Es una relación 1:1 entre SEDE y ENCARGADO dónde se especifica que una sede tiene un Encargado que la administra.
2. **Gestiona:** Es una relación 1:N entre SEDE y EMPRESA. Ya que una o varias empresas pueden estar gestionadas por una sede.
3. **Tiene:** Es una relación 1:N, entre SEDE y VEHÍCULO. Dónde se especifica que una sede puede tener 1 o varios coches.
4. **Conduce:** Es una relación N:M entre CONDUCTOR y VEHÍCULO. Que permite que varios conductores puedan conducir varios vehículos.
5. **Pasa:** Es una relación 1:1 entre CONDUCTOR y TEST, esta específica que un conductor tiene un test que puede o no pasar.
6. **Asociado:** Es una relación 1:N entre VEHICULO y TALLER. Dónde 1 o varios vehículos pueden estar asociados a talleres.
7. **Pertenece:** Es una relación 1:N entre VEHICULO e INFORME donde un vehículo puede tener varios informes asociados.
8. **Genera:** Es una relación 1:N entre TALLER e INFORME. Donde cada taller puede generar muchos informes.



9. **Participa:** Relación 1:N entre EMPRESA y CONTRATO. Donde cada empresa puede participar en varios contratos.
10. **Participa:** Relación 1:N entre VEHICULO y CONTRATO. Donde un vehículo puede llegar a participar en varios contratos.
11. **Envía:** Es una relación ternaria cuya cardinalidad es 1:1:N entre EMPRESA, PAQUETE y VEHÍCULO. Donde una empresa puede enviar a través de 1 vehículo varios paquetes.
  - **Atributos:** Fecha (fecha de envío del producto) y Destino (lugar al que se envía).

### 3.3 Consideraciones adicionales

- **IS A:** En la entidad relación tenemos un IS A de exclusividad y total donde se establece el tipo de un vehículo además de que si un vehículo es una furgoneta no puede ser un camión a la vez y viceversa.
- **Inclusividad:** Entre las relaciones conduce y pasa que implica que un conductor puede conducir si previamente ha pasado un test.

El resto de consideraciones adicionales serán añadidas mediante checks y triggers.

## 4. Scripts SQL

### 4.1 Creación de Base de Datos

- **Creación de la base de datos.**

```
Unset
SET statement_timeout = 0;
SET lock_timeout = 0;
SET idle_in_transaction_session_timeout = 0;
SET client_encoding = 'UTF8';
SET standard_conforming_strings = on;
SELECT pg_catalog.set_config('search_path', '', false);
SET check_function_bodies = false;
SET xmloption = content;
SET client_min_messages = warning;
SET row_security = off;
DROP DATABASE viewcartransit;
```





```
CREATE DATABASE viewcartransit WITH TEMPLATE = template0 ENCODING = 'UTF8'
LOCALE = 'es_ES.UTF-8';

\connect viewcartransit

SET statement_timeout = 0;
SET lock_timeout = 0;
SET idle_in_transaction_session_timeout = 0;
SET client_encoding = 'UTF8';
SET standard_conforming_strings = on;
SELECT pg_catalog.set_config('search_path', '', false);
SET check_function_bodies = false;
SET xmloption = content;
SET client_min_messages = warning;
SET row_security = off;
```

## 4.2 Creación de las talas

- Ejemplo de creación de la tabla sede con sus respectivos checks

```
Unset
CREATE TABLE SEDE (
    id_sede SERIAL PRIMARY KEY,
    nombre VARCHAR(50),
    localidad VARCHAR(50),
    calle VARCHAR(50),
    numero VARCHAR(10),
    telefono VARCHAR(20) CHECK (telefono ~ '^\\d{3}-\\d{3}-\\d{3}$'),
    correo_contacto VARCHAR(50) CHECK (correo_contacto LIKE '%@%.%')
);
```

El resto de tablas creadas se encuentran en el siguiente [enlace](#).



## 4.3 Datos de la Base de Datos

- **Ejemplo de Inserción de los datos de una sede.**

```
Unset
INSERT INTO SEDE (nombre, localidad, calle, numero, telefono,
correo_contacto)
VALUES ('Sede Central Tenerife', 'Tenerife', 'Los Majuelos', '100',
'922-123-456', 'central@viewcartransit.com'),
      ('Sede Sur Tenerife', 'Adeje', 'Sur', '200', '922-456-789',
'sur@viewcartransit.com'),
      ('Sede Norte Tenerife', 'La Orotava', 'Norte', '300', '922-789-123',
'norte@viewcartransit.com');
```

El resto de inserciones utilizadas se encuentran en el siguiente [enlace](#).

## 4.4 Triggers

- **Ejemplo de trigger para comprobar si un vehículo está disponible antes de ser asignado a un contrato**

```
Unset
-- Trigger para verificar disponibilidad del vehiculo al asignar contrato
CREATE OR REPLACE FUNCTION check_vehiculo_disponible()
RETURNS TRIGGER AS $$
BEGIN
    IF (SELECT estado FROM VEHICULO WHERE matricula = NEW.matricula LIMIT 1)
    != 'Disponible' THEN
        RAISE EXCEPTION 'El vehiculo no esta disponible';
    END IF;

    RETURN NEW;
END;
```

El resto de triggers creados se encuentran en el siguiente [enlace](#).



## 5. Ejemplos funcionamiento de la base de datos.

- Mostrar las sedes y las empresas que tienen.

Unset

```
SELECT sede.nombre nombre_sede, empresa.nombre empresa
FROM SEDE sede
JOIN EMPRESA empresa ON sede.id_sede = empresa.id_sede;
```

Resultado de la consulta:

	nombre_sede character varying (50) 🔒	empresa character varying (50) 🔒
1	Sede Central Tenerife	Supermercado La Colmena
2	Sede Central Tenerife	Tienda ElectroMax
3	Sede Sur Tenerife	Moda y Complementos SRL
4	Sede Sur Tenerife	Jugueteria HappyKids
5	Sede Norte Tenerife	Hogar Decoracion

- Mostrar sede y encargado de la sede

Unset

```
SELECT sede.nombre nombre_sede, encargado.nombre nombre_encargado
FROM SEDE sede
JOIN ENCARGADO encargado ON sede.id_sede = encargado.id_sede;
```

Resultado de la consulta:

	nombre_sede character varying (50) 🔒	nombre_encargado character varying (50) 🔒
1	Sede Central Tenerife	Juan
2	Sede Sur Tenerife	Maria
3	Sede Norte Tenerife	Pedro



- **Mostrar sede y vehículos que tiene**

Unset

```
SELECT sede.nombre nombre_sede, vehiculo.matricula matricula_vehiculo
FROM SEDE sede
JOIN VEHICULO vehiculo ON sede.id_sede = vehiculo.id_sede;
```

Resultado de la consulta:

	nombre_sede character varying (50) 🔒	matricula_vehiculo character varying (20) 🔒
1	Sede Central Tenerife	0000AAA
2	Sede Central Tenerife	0000AAB
3	Sede Central Tenerife	0000AAC
4	Sede Central Tenerife	0000AAD
5	Sede Central Tenerife	0000AAE
6	Sede Sur Tenerife	0000AAF
7	Sede Central Tenerife	0000AAG
8	Sede Central Tenerife	0000AAH
9	Sede Central Tenerife	0000AAI
10	Sede Central Tenerife	0000AAJ

- **Mostrar los contratos que tienen las empresas actualmente y el vehículo que tienen contratado**

Unset

```
SELECT empresa.nombre nombre_empresa, vehiculo.modelo, contrato.fecha_ini
fecha_inicio, contrato.fecha_fin fecha_fin
FROM EMPRESA empresa
JOIN CONTRATO contrato ON empresa.id_empresa = contrato.id_empresa
JOIN VEHICULO vehiculo ON contrato.matricula = vehiculo.matricula
```



Resultado de la consulta:

	nombre_empresa character varying (50)	modelo character varying (50)	fecha_inicio date	fecha_fin date
1	Supermercado La Colmena	Renault Kangoo	2024-12-20	2025-01-20
2	Tienda ElectroMax	Mercedes Sprinter	2024-12-20	2025-01-20
3	Moda y Complementos SRL	Iveco Daily	2024-12-20	2025-01-20
4	Jugueteria HappyKids	Scania R450	2025-01-21	2025-02-20
5	Jugueteria HappyKids	Scania R450	2024-12-20	2025-01-20
6	Hogar Decoracion	Volvo FH16	2025-01-21	2025-02-20
7	Hogar Decoracion	Volvo FH16	2024-12-20	2025-01-20
8	Supermercado La Colmena	Volkswagen Crafter	2025-01-21	2025-02-20
9	Supermercado La Colmena	Volkswagen Crafter	2024-12-20	2025-01-20
10	Tienda ElectroMax	Ford Transit	2025-01-21	2025-02-20

- **Mostrar los paquetes que se han enviado desde una furgoneta**

Unset

```
SELECT paquete.descripcion paquete, empresa.nombre nombre_empresa,  
furgoneta.modelo coche  
FROM ENVIA envia  
JOIN PAQUETE paquete ON envia.id_paquete = paquete.id_paquete  
JOIN EMPRESA empresa ON envia.id_empresa = empresa.id_empresa  
JOIN FURGONETA furgoneta ON envia.matricula = furgoneta.matricula;
```

Resultado de la consulta:

	paquete character varying (255)	nombre_empresa character varying (50)	coche character varying (50)
1	Televisor LED 55 pulgadas	Supermercado La Colmena	Renault Kangoo
2	Ropa de invierno para niños	Tienda ElectroMax	Mercedes Sprinter
3	Set de muebles de jardin	Moda y Complementos SRL	Iveco Daily
4	Set de herramientas electricas	Jugueteria HappyKids	Scania R450
5	Juguetes educativos	Jugueteria HappyKids	Scania R450
6	Juguetes educativos	Jugueteria HappyKids	Scania R450
7	Cafetera de ultima generacion	Hogar Decoracion	Volvo FH16
8	Decoracion para el hogar	Hogar Decoracion	Volvo FH16
9	Decoracion para el hogar	Hogar Decoracion	Volvo FH16
10	Laptop Ultrabook	Supermercado La Colmena	Volkswagen Crafter



- **Mostrar los paquetes que se han enviado desde un camión**

Unset

```
SELECT paquete.descripcion paquete, empresa.nombre nombre_empresa,  
camion.modelo coche  
FROM ENVIA envia  
JOIN PAQUETE paquete ON envia.id_paquete = paquete.id_paquete  
JOIN EMPRESA empresa ON envia.id_empresa = empresa.id_empresa  
JOIN CAMION camion ON envia.matricula = camion.matricula;
```

Resultado de la consulta:

	paquete character varying (255) 🔒	nombre_empresa character varying (50) 🔒	coche character varying (50) 🔒
1	Juego de sábanas premium	Moda y Complementos SRL	Renault Trafic
2	Set de muebles de jardín	Moda y Complementos SRL	Renault Trafic
3	Laptop Ultrabook	Supermercado La Colmena	Volkswagen Caddy
4	Televisor LED 55 pulgadas	Supermercado La Colmena	Volkswagen Caddy
5	Refrigerador 300L	Tienda ElectroMax	Iveco Eurocargo
6	Ropa de invierno para niños	Tienda ElectroMax	Iveco Eurocargo

- **Informe de un taller y el vehículo que ha sido reparado**

Unset

```
SELECT taller.nombre nombre_taller, informe.descripcion, vehiculo.modelo  
modelo_vehiculo, informe.fecha fecha_reparacion  
FROM INFORME informe  
JOIN TALLER taller ON informe.id_taller = taller.id_taller  
JOIN ONLY VEHICULO vehiculo ON informe.matricula = vehiculo.matricula;
```

Resultado de la consulta:

	nombre_taller character varying (50) 🔒	descripcion character varying (255) 🔒	modelo_vehiculo character varying (50) 🔒	fecha_reparacion date 🔒
1	Taller Central	El vehículo presenta un fallo en el motor	Renault Kangoo	2024-12-20
2	Taller Sur	El vehículo presenta un fallo en el sistema de frenos	Mercedes Sprinter	2024-12-20
3	Taller Norte	El vehículo presenta un fallo en el sistema de dirección	Iveco Daily	2024-12-20
4	Taller Este	El vehículo presenta un fallo en el sistema de luces	Scania R450	2024-12-20
5	Taller Oeste	El vehículo presenta un fallo en el sistema de climatizaci...	Volvo FH16	2024-12-20
6	Taller Paco	El vehículo presenta un fallo en el sistema de suspensión	Volkswagen Crafter	2024-12-20



- **Mostrar los conductores aptos para nuestra empresa y la nota que sacaron en su test**

Unset

```
SELECT conductor.nombre nombre_conductor, conductor.apellidos  
apellidos_conductor, test.nota nota_conductor  
FROM CONDUCTOR conductor  
JOIN TEST test ON conductor.dni = test.dni  
WHERE test.nota >= 5;
```

Resultado de la consulta:

	nombre_conductor character varying (50)	apellidos_conductor character varying (50)	nota_conductor numeric (5,2)
1	Juan	Perez	8.50
2	Maria	Lopez	7.00
3	Pedro	Garcia	9.00
4	Laura	Martinez	6.50
5	Carlos	Sanchez	8.00
6	Ana	Gonzalez	5.50

- **Insertar un vehículo en una sede**

Unset

```
INSERT INTO CAMION (matricula, modelo, color, estado, id_sede, id_taller,  
tiene_trailer)  
VALUES ('1000FAA', 'Mercedes Gama Actros', 'Gris', 'Disponible', 1, 1,  
FALSE);
```

Ejecución:

```
1  -- Insertar un vehiculo en una sede  
2  ▾ INSERT INTO CAMION (matricula, modelo, color, estado, id_sede, id_taller, tiene_trailer)  
3  VALUES ('1000FAA', 'Mercedes Gama Actros', 'Gris', 'Disponible', 1, 1, FALSE);
```

Data Output [Messages](#) Notifications

INSERT 0 1

Query returned successfully in 55 msec.



Comprobación:

`SELECT * FROM CAMION Where matricula = '1000FAA';`

Output Messages Notifications

matricula	modelo	color	estado	id_sede	id_taller	tiene_trailer
character varying (20)	character varying (50)	character varying (20)	character varying (10)	integer	integer	boolean
1000FAA	Mercedes Gama Actros	Gris	Disponible	1	1	false

- Añadir un contrato a una empresa

Unset

```
INSERT INTO CONTRATO (id_empresa, matricula, fecha_ini, fecha_fin)
VALUES (1, '1000FAA', CURRENT_DATE + INTERVAL '3 month', CURRENT_DATE +
INTERVAL '4 month');
```

Ejecución:

Query Query History

```
1 INSERT INTO CONTRATO (id_empresa, matricula, fecha_ini, fecha_fin)
2 VALUES (1, '1000FAA', CURRENT_DATE + INTERVAL '3 month', CURRENT_DATE + INTERVAL '4 month');
```

Data Output Messages Notifications

INSERT 0 1

Query returned successfully in 69 msec.

Comprobación:

Query Query History

```
1 SELECT * FROM CONTRATO Where matricula = '1000FAA';
2
```

Data Output Messages Notifications

	id_contrato	id_empresa	matricula	fecha_ini	fecha_fin
	[PK] integer	integer	character varying (20)	date	date
1	29	1	1000FAA	2025-03-20	2025-04-20

- Añadir un nuevo envío

Unset

```
INSERT INTO ENVIA (matricula, id_paquete, id_empresa, destino, fecha)
```





```
VALUES ('1000FAA', 31, 1, 'Santa Cruz de Tenerife', CURRENT_DATE + INTERVAL '100 day');
```

Ejecución:

Query Query History

```
1 INSERT INTO ENVIA (matricula, id_paquete, id_empresa, destino, fecha)
2 VALUES ('1000FAA', 31, 1, 'Santa Cruz de Tenerife', CURRENT_DATE + INTERVAL '100 day');
```

Data Output Messages Notifications

INSERT 0 1

Query returned successfully in 80 msec.

Comprobación:

Query Query History

```
1 SELECT * FROM ENVIA WHERE matricula = '1000FAA';
```

Data Output Messages Notifications

	matricula character varying (20)	id_paquete [PK] integer	id_empresa integer	destino character varying (100)	fecha date
1	1000FAA	31	1	Santa Cruz de Tenerife	2025-03-30

- **Mostrar nuevo envío y contrato**

Unset

```
SELECT paquete.descripcion paquete, empresa.nombre nombre_empresa,
camion.modelo coche, camion.matricula
FROM ENVIA envia
JOIN PAQUETE paquete ON envia.id_paquete = paquete.id_paquete
JOIN EMPRESA empresa ON envia.id_empresa = empresa.id_empresa
JOIN CAMION camion ON envia.matricula = camion.matricula
WHERE paquete.id_paquete = 31;
SELECT contrato.id_empresa, contrato.matricula, contrato.fecha_ini,
contrato.fecha_fin
FROM CONTRATO contrato
WHERE contrato.id_empresa = 1 AND contrato.mantricula = '1000FAA';
```



Ejecución:

```
SELECT paquete.descripcion paquete, empresa.nombre nombre_empresa, camion.modelo coche, camion.matricula
FROM ENVIA envia
JOIN PAQUETE paquete ON envia.id_paquete = paquete.id_paquete
JOIN EMPRESA empresa ON envia.id_empresa = empresa.id_empresa
JOIN CAMION camion ON envia.matricula = camion.matricula
WHERE paquete.id_paquete = 31;
SELECT contrato.id_empresa, contrato.matricula, contrato.fecha_ini, contrato.fecha_fin
FROM CONTRATO contrato
WHERE contrato.id_empresa = 1 AND contrato.matricula = '1000FAA';
```

Comprobación:

Output Messages Notifications				
				SQL
id_empresa	matricula	fecha_ini	fecha_fin	
integer	character varying (20)	date	date	
1	1000FAA	2025-03-20	2025-04-20	

- Eliminar un camión

Unset

```
DELETE FROM CAMION WHERE matricula = '1000FAA';
```

Ejecución:

```
Query Query History
1 -- Eliminar vehiculo
2 DELETE FROM CAMION WHERE matricula = '1000FAA';
3
Data Output Messages Notifications
DELETE 1
Query returned successfully in 59 msec.
```

Comprobación:

1

SELECT \* FROM CAMION WHERE matricula = '1000FAA';

Data Output

Messages

Notifications

SQL

matricula

character varying (20)

modelo

character varying (50)

color

character varying (20)

estado

character varying (10)

id\_sede

integer

id\_taller

integer

tiene\_trailer

boolean

- Reasignación de paquete al haber eliminado el otro vehículo.



Unset

```
SELECT paquete.descripcion paquete, empresa.nombre nombre_empresa,  
camion.modelo coche, camion.matricula  
FROM ENVIA envia  
JOIN PAQUETE paquete ON envia.id_paquete = paquete.id_paquete  
JOIN EMPRESA empresa ON envia.id_empresa = empresa.id_empresa  
JOIN CAMION camion ON envia.matricula = camion.matricula  
WHERE paquete.id_paquete = 31;
```

Comprobación:

Query		Query History	
1	SELECT	paquete.descripcion paquete, empresa.nombre nombre_empresa, camion.modelo coche, camion.matricula	
2	FROM	ENVIA envia	
3	JOIN	PAQUETE paquete ON envia.id_paquete = paquete.id_paquete	
4	JOIN	EMPRESA empresa ON envia.id_empresa = empresa.id_empresa	
5	JOIN	CAMION camion ON envia.matricula = camion.matricula	
6	WHERE	paquete.id_paquete = 31;	

Data Output		Messages		Notifications	
+	SQL				
	paquete	nombre_empresa	coche	matricula	
	character varying (255)	character varying (50)	character varying (50)	character varying (20)	
1	Aspiradora robot	Supermercado La Colmena	Fiat Ducato	0000ABX	

## 6. Repositorio GitHub

Todo el proyecto ha sido elaborado de forma conjunta en el siguiente repositorio:

<https://github.com/marcosxmh/Proyecto-ADBD-ViewCarTransit.git>

## 7. Bibliografía