



# **Universidad del Valle**

**Taller**

**Proyecto en clase**

**Move\_to\_fast**

**Federico Barbetti Ruiz, código 2181247-2724**

**Universidad del Valle**

**Asignatura: Bases de datos**

**Profesor: Jefferson A. Peña Torres**

**Fecha de entrega: Abril 27 de 2025**

## INFORME

En el desarrollo de este proyecto ejecuté lo aprendido acerca de modelación de bases de datos relacionales así como también el uso de restricciones para lograr la integralidad en el manejo de los datos.

El proceso empieza a partir del despliegue de los contenedores de docker correspondientes a las versiones de Postgres 12 y Pgadmin 4, este despliegue se hace en el puerto local.

Supuesto: 1 vehículo pertenece a una sola sucursal.

Abordando el tema de la definición de la PRIMARY KEY para las relaciones, se encuentra que las relaciones “alquiler” y “pago” no tienen una PK natural, a diferencia de las relaciones “cliente” y “vehículo” que naturalmente la poseen.

De tal manera que podemos asegurar que las relaciones alquiler y pago son DÉBILES, pues dependen de su existencia de otras relaciones. Aun así, se les debe generar su PK compuesta, en ambos casos fue de gran ayuda el atributo SERIAL puesto que complementa e individualiza la PK en cada una de las estas relaciones DÉBILES.

La PK en las relaciones que es natural que la lleven, se configuró como NOT NULL; no así en las relaciones DÉBILES, a mi manera de ver, no hace falta esa restricción puesto que al ser una PK compuesta por atributos NOT NULL, sería redundante otra definición más. No obstante se sabe que la PK nos provee la restricción de integridad de la entidad o relación (duda).

En el proyecto se crea un tipo de dato denominado “carro” y se define su dominio; por tanto este atributo se configura como NOT NULL para lograr una restricción de integridad de dominio.

En las entidades débiles, se generaron las FOREIGN KEY para asegurar la restricción de integridad referencial, esto se logra al configurar la PK como NOT NULL. Así se asegura que la FK siempre estará compuesta completamente.

Se usa la restricción CHECK para asegurar un rango determinado en el dominio, esto se muestra en la tabla “vehículo” en su campo “tarifa dia”; también, en la tabla “pago” en su campo “importe”.

En ambos casos, con el uso de las restricciones, se fortalece el control en los datos para asegurar las reglas del negocio, en este caso las tarifas mínimas cobradas por la renta de vehículos.

**El uso de valor DEFAULT para un campo de una tabla, se presenta en la tabla “alquiler”, en el campo “disponible” (atributo booleano) se configura un valor default TRUE, simbolizando que un vehículo siempre estará disponible para el servicio, lógicamente, es natural que en un “momento” no sea del todo cierto, puesto que los vehículos “entran y salen”, así que lo recomendable es siempre hacer una actualización de los datos con una cláusula de consulta.**

**Las cláusulas ON UPDATE CASCADE y ON DELETE CASCADE se usaron a fin de actualizar la información del negocio ante los eventos de la pérdida de un cliente (se eliminan sus registros) o el cambio del id de una sucursal (se actualiza el dato en los vehículos asignados a esa sucursal), estas herramientas, hacen que los datos estén actualizados y así el negocio pueda contar con la integralidad en su información.**

**Nota: Se adjunta diagrama conceptual E--R, agradecería mucho cualquier retroalimentación.**

**Mil Gracias Profe.**