**Requerimientos funcionales de consulta**

**RFC 1 CONSULTAR EL HISTÓRICO DE TODOS LOS SERVICIOS PEDIDOS POR UN USUARIO**

Tablas usadas: Servicio, DetallesServicio y Usuario

Columnas: Usario; IdUsuario

Servicio; IdServicio

DetallesServicio; fecha, longitudTrayecto, costoTotal, tipoServicio

ConductorVehiculo; idVehiculo

JOIN: Se usó el INNER JOIN para en primera instancia unir el servicio con su detalle, y siguiendo el mismo razonamiento unir el servicio con su usuario asignado. Y por último se unió al conductor del servicio junto con el usuario. Como todos son elementos que coinciden y nunca serán null, no hace falta usar un JOIN que admita estos.

Ejemplo consulta:

SELECT s.idServicio, ds.fecha, ds.longitudTrayecto, ds.costoTotal, ds.tipoServicio, cv.idVehiculo FROM DatosUsuario du

INNER JOIN Servicio s ON du.IDUSUARIO = s.IDUSUARIO

INNER JOIN DetallesServicio ds ON s.IDSERVICIO = ds.IDSERVICIO

INNER JOIN ConductorVehiculo cv ON s.idVehiculo = cv.idVehiculo

WHERE du.IDUSUARIO = 'U00003';

**RFC2 MOSTRAR LOS 20 USUARIOS CONDUCTORES QUE MÁS SERVICIOS HAN PRESTADO EN LA APLICACIÓN**

Tablas: DatosUsuario y Servicio

Columnas: Datos usuario; idUsuario, nombre

Servicio; idServicio

JOIN: Siguiendo la misma lógica, al solo necesitar unir elementos en común, se usó INNER JOIN

Ejemplo de consulta:

SELECT du.idUsuario AS idConductor, du.nombre, COUNT(\*) AS totalServiciosPrestados FROM Servicio s

INNER JOIN DatosUsuario du ON s.IDUSUARIO = du.IDUSUARIO

INNER JOIN DetallesServicio ds ON s.IDSERVICIO = ds.IDSERVICIO

INNER JOIN ConductorVehiculo cv ON s.idVehiculo = cv.idVehiculo

WHERE cv.idUsuario = 'C00002'

GROUP BY du.idUsuario, du.nombre

ORDER BY totalServiciosPrestados DESC;

**RFC3 MOSTRAR EL TOTAL DE DINERO OBTENIDO POR USUARIOS CONDUCTORES PARA CADA UNO DE SUS VEHÍCULOS, DISCRIMINADO POR SERVICIOS**

Tablas: DetallesServicio y Servicio

Columnas: Datos usuario; costoTotal, comisión, tipoServicio

Servicio; idServicio.

ConductorVehiculo: idUsuario

JOIN: Siguiendo la misma lógica, al solo necesitar unir elementos en común, se usó INNER JOIN

Ejemplo de consulta:

SELECT SUM(ds.Costototal- ds.comision) AS ganancia, ds.tiposervicio, s.idVehiculo, cv.idUsuario FROM Servicio s

INNER JOIN DetallesServicio ds ON s.IDSERVICIO = ds.IDSERVICIO

INNER JOIN ConductorVehiculo cv ON s.idVehiculo = cv.idVehiculo

WHERE cv.idUsuario = 'C00002'

GROUP BY s.idVehiculo, ds.tiposervicio, cv.idUsuario;

**RFC4 - MOSTRAR LA UTILIZACIÓN DE SERVICIOS DE ALPESCAB EN UNA CIUDAD DURANTE UN RANGO DE FECHAS INDICADO.**

Tablas: DetallesServicio, Servicio, Ciudad y PuntoUbicacion

Columnas: PuntoUbicación; idCiudad

Ciudad: nombre

Servicio; idServicio.

DetallesServicio; idServicio, puntoUbicacion.

JOIN: Siguiendo la misma lógica, al solo necesitar unir elementos en común, se usó INNER JOIN

Ejemplo de consulta:

SELECT p.idciudad, c.nombre, COUNT(\*) AS total\_servicios, ROUND( 100 \* COUNT(\*) / SUM(COUNT(\*)) OVER (), 2 ) AS porcentaje FROM DetallesServicio ds

INNER JOIN PuntoUbicacion p ON ds.PUNTOUBICACION = p.idPunto

INNER JOIN Ciudad c ON p.idCiudad = c.idCiudad

WHERE ds.fecha BETWEEN TO\_DATE('2024-01-01', 'YYYY-MM-DD') AND TO\_DATE('2025-12-31', 'YYYY-MM-DD')

GROUP BY p.idciudad, c.nombre

ORDER BY total\_servicios DESC;