

# Estrategia

1. Utilizamos claves subrogadas para todas las tablas ya que nos ayudan a independizarnos del dominio. En el caso de que exista alguna modificación en el mismo, no se verán afectadas una o más tablas.
2. Elegimos como pk el telefono de los choferes y de los clientes de la tabla maestra para migrar.
3. Elegimos como contraseña de los usuarios, el usuario (excepto para admin), y elegimos como usuario, el teléfono de los clientes y choferes, ya que, como restricción del enunciado, el teléfono debe ser único para cada usuario, y no se aceptan usuarios repetidos. Esto nos da una doble ventaja ya que al dar de alta un nuevo Cliente o un Chofer, en caso de que el teléfono no exista, se crea el nuevo usuario y se lo asocia al rol donde se lo esté dando de alta, en cambio, si el usuario ya existe en el sistema, se verifica si es para el mismo rol, en caso de ser para el mismo rol, hay un error de usuario existente, en caso de no ser para el mismo rol, se crea un registro en la tabla USUARIO\_POR\_ROL en la que se le agrega el nuevo rol.
4. Elegimos la patente del auto como pk de la tabla maestra para migrar los automóviles
5. Utilizamos la descripción del turno como pk de la tabla maestra para migrar los turnos.
6. Para validar que no se realicen facturas a clientes inhabilitados, pagos a choferes inhabilitados, y viajes con turnos/clientes/choferes/autos inhabilitados, utilizamos triggers en los inserts de las respectivas tablas. Estos triggers los creamos luego de haber realizado la migración ya que por default todos los usuarios que migramos al sistema están habilitados.
7. Utilizamos cursores para verificar que no haya solapamientos entre turnos. Estos cursores se encuentran en el store procedure Alta\_Turno.
8. Dado que los viajes no tenían fecha de fin, realizamos un estimativo del mismo teniendo en cuenta la cantidad de kilómetros recorridos. Consideramos que un auto tarda 1 minuto por kilómetro.
9. Utilizamos la función UPPER para los nombres de los clientes y los choferes, para que queden consistentes con el apellido.
10. Casteamos los turnos de int a time ya que necesitamos tratarlo como tal para poder tener minutos en los mismos.
11. Tomamos como pk de los viajes, el cliente y la fecha para migrar.
12. Como existen muchos viajes repetidos, lo que hicimos fue tomar el máximo valor de auto/chofer/turno/cantidad de km para cada uno. No pudimos definir un criterio lógico ya que no tenemos información acerca de los mismos y no sabemos cuál es el verdadero.
13. Tomamos como turno activo el turno en el cual se realizó el ultimo viaje para ese chofer. Existen viajes realizados en el mismo día, para el mismo chofer, en diferentes horarios.
14. Los errores lógicos con los que venía la tabla maestra fueron migrados igualmente, y se limitan a partir de los nuevos datos que se vayan a ingresar al sistema. Por ejemplo, que un chofer tenga viajes realizados en diferentes turnos de un mismo día, fechas de nacimiento invalidas ( gente nacida en 1902 por ej) etc.
15. No migramos los números de factura y renidicon que vienen en la tabla maestra ya que son discontinuos y no nos sirven como pk de nuestra nueva tabla. Para tener una tabla limpia, decidimos crear nuestra propia pk subrogada
16. Para disminuir la cantidad de validaciones y puntos de falla, se decidió usar un ComboBox no editable en todas las pantallas de baja
17. En la modificación de funcionalidades por rol, optamos por borrar todas ante cada cambio, y así evitar validar la presencia de ellas, una por una

